Pacific Graphics 2013
B. Levy, X. Tong, and K. Yin
(Guest Editors)

Volume 32 (2013), Number 7

Guided Real-Time Scanning of Indoor Objects

Young Min Kim[⋆] Niloy J. Mitra[†] Qixing Huang[⋆] Leonidas Guibas[⋆]

[⋆]Stanford University [†]University College London

Abstract

Advances in 3D acquisition devices provide unprecedented opportunities for quickly scanning indoor environments. Such raw scans, however, are often noisy, incomplete, and significantly corrupted, making semantic scene understanding difficult, if not impossible. Unfortunately, in most existing workflows, scan quality is assessed after *the scanning stage is completed, making it cumbersome to correct for significant missing data by additional scanning. In this work, we present a* guided real-time scanning *setup, wherein the incoming 3D data stream is continuously analyzed, and the data quality is automatically assessed. While the user is scanning an object, the proposed system discovers and highlights potential missing parts, thus guiding the operator (or an autonomous robot) as* where to scan next. *The proposed system assesses the quality and completeness of the 3D scan data by comparing to a large collection of commonly occurring indoor man-made objects using an efficient, robust, and effective scan descriptor. We have tested the system on a large number of simulated and real setups, and found the guided interface to be effective even in cluttered and complex indoor environments.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

Navigating, acquiring, and understanding complex environments is a critical component for any autonomous robotics system. While for outdoor environments such a system can proceed based on provided maps, road markings, or GPS data, for indoor environments such useful information is either unreliable or simply unavailable. Furthermore, indoor environments tend to be both more cluttered and more variable over time, further complicating the task of reconstruction. Thus, for successful autonomous navigation and task completion in novel indoor scenes, it is necessary to simultaneously scan the environment, interpret the incoming data stream, and plan subsequent data acquisition, all in real-time.

It is now possible to obtain real-time 3D scans using portable commercial scanners (e.g., the Microsoft Kinect scanner); such 3D data can be very valuable in building useful, semantically-meaningful models of the environment. The challenge is, however, that individual frames from such scanners are often of poor quality (i.e., noisy point-clouds, with outliers, large regions missing, etc.). Typically, complex geometry can only be acquired by accumulating multiple scans. Information integration is done in a post-scanning phase, when individual scans are registered and merged, leading eventually to useful object models. Such a workflow, however, is limited by the fact that poorly scanned or missing regions are only identified *after* the scanning process is

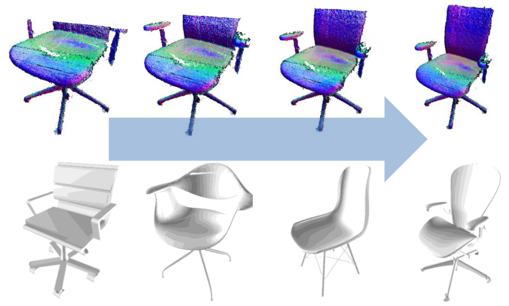

Figure 1: *We introduce a real-time guided scanning system. As streaming 3D data is progressively accumulated (top), the system retrieves the top matching models (bottom) along with their pose to act as geometric proxies to assess the current scan quality, and provide guidance for subsequent acquisition frames. Only a few intermediate frames with corresponding retrieved models are shown in this figure.*

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd. Published by John Wiley & Sons Ltd.

finished, when it may be costly to revisit the environment being acquired to perform additional scans. We focus on real-time 3D model quality assessment and data understanding in order to provide immediate feedback for guidance in subsequent acquisition.

Evaluating acquisition quality without having any prior knowledge about an unknown object is an ill-posed task. We observe that although the target object itself maybe unknown, in many cases, it comes from a well-prescribed predefined set of object categories. Moreover, these categories are well represented in existing shape repositories (e.g., Trimble 3D Warehouse). For example, an office setting typically consists of various tables, chairs, monitors, etc., all of which have thousands of instances in the Trimble 3D Warehouse. Hence, instead of trying to recover detailed 3D geometry from low-quality inconsistent 3D measurements, we focus on parsing the input object scans into simpler geometric entities, and use publicly available 3D model repositories like Trimble 3D warehouse as proxies to assist the process of assessing data quality. To this end, we have to overcome two key challenges: (i) Given a partially scanned object, reliably and efficiently retrieve appropriate proxy models from the database; and (ii) position the retrieved models in the scene and provide real-time feedback (e.g., missing geometry that still needs to be scanned) to guide subsequent data gathering.

We introduce A2h, a novel shape descriptor for retrieval of similar shapes of a query partial scan. Subsequently, we use the A2h descriptor, to enable guided real-time scanning using a Microsoft Kinect scanners. The proposed approach, which combines both descriptor-based retrieval and registration-based verification, is able to search in a database of thousands of models, in real-time. To account for partial similarity between the input scan and the models in a database, we created simulated scans of each database model and compared a scan of real setting to a scan of simulated setting. This allowed us to efficiently compare shapes using global descriptors even in the presence of only partial similarity; and the approach remains robust in the case of occlusions or missing data about the object being scanned.

In order to identify problematic regions of the current scan, our system aligns the retrieved match with the partial scan and highlights potential missing parts or places where the scan density is low. This visual feedback allows the operator to quickly adjust the scanning device for subsequent scans. In effect, our 3D model database and matching algorithms make it possible for the operator to assess the quality of the data being acquired and discover badly scanned or missing areas *while* the scan is being performed, thus allowing corrective actions to be taken immediately.

We extensively evaluated the robustness and accuracy of our system using synthetic data sets with available ground truth. Further, we tested our system on typical office environments to achieve real-time object scanning (see the supplementary video that includes the actual scanning session recorded). In summary, we present a novel *guided scanning interface* and introduce a relation-based light-weight A2h descriptor for fast and accurate model retrieval and positioning to provide real-time guidance for scanning.

2. Related Works

Interactive acquisition. Fast, accurate, and autonomous model acquisition has long been primary goals in computer graphics and computer vision. With the introduction of affordable, portable, commercial RGB-D cameras, there has been a pressing need to simplify scene acquisition workflows to allow less experienced individuals to acquire scene geometries around them. Recent efforts fall into two broad categories: (i) combining individual frames of low-quality point-cloud data with SLAM algorithms [EEH*11, HKH*12] to improve scan quality [IKH*11]; and (ii) using supervised learning to train classifiers for scene labeling [RBF12] with applications to robotics [KAJS11]. Previously, [RHHL02] aggregated scans at interactive rates to provide visual feedback such that a user can deliberately fill in the missing area and overcome registration errors. This work was recently expanded by [DHR*11] to use RGB-D cameras. However, there very few attempts to provide feedback in terms of higher-level interpretation of the scan. Kim et al. [KDS*12] extract simple planes and reconstruct floor plans with guidance from a projector pattern. While our goal is also to provide real-time feedback, our system differs from previous efforts in using retrieved proxy models to automatically access the current scan quality, enabling object-level understanding and provide guidance accordingly.

Scan completion. Various strategies have been proposed to improve noisy scans or plausibly fill in missing data due to occlusion (off-line): researchers have exploited repetition [PMW*08], symmetry [TW05, MPWC12], or used primitives to complete missing parts [SWK07]. Other approaches have focused on using geometric proxies and abstractions including curves, skeletons, planar abstractions, etc. In the context of image understanding, indoor scenes have been abstracted and modeled as a collection of simple cuboids [LGHK10, ZCC*12, KLM*13] to capture a variety of man-made objects.

Part-based modeling. Simple geometric primitives, however, are not always sufficiently expressive for complex shapes. However, such objects can still be split into simpler parts that aid shape understanding. For example, parts can act as entities for discovering repetitions [TSS10], training classifiers [SFC*11, XS12], or facilitating shape synthesis [JTRS12]. Multiple objects of a single category can also be represented by a smaller set of part-based template [KLM*13]. Such methods often rely on expensive matching, and thus do not lend themselves to low-memory footprint real-time realizations.

Template-based completion. Real 3D scans suffer from

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

noise and occlusion, which could be completed by geometrically matching templates [HCI*11], or using templates to complete missing parts [PMG*05] A similar line of recent works includes detecting pre-defined 3-D models (usually 100-200 models) for indoor understanding [SXZ*12, NXS12, KMYG12], or scan-based 3-D shape retrieval [BP13, SPT12]. Our system also uses database of 3D models (e.g., chairs, lamps, tables) to retrieve shape from 3D scans. However, by introducing A2h, a novel simple descriptor, compared to previous efforts, our system reliably handles much larger model databases (thousands of models instead of hundreds of models).

Shape descriptors. In the context of shape retrieval, various descriptors have been investigated for grouping, classification, or retrieval of 3D geometry [FKMS05], and we do not attempt to name them all. Among them, only a small number of descriptor might be suitable for compact for real-time application and robust to noise enough to be used in real-data. For example, the method proposed by [CTSO03] uses light-field descriptors based on silhouettes, the method by [OFCD02] uses shape distributions to categorize different object classes, etc. Both methods assume access to nearly complete models to match against. The silhouette method requires an expensive rotational alignment search, limiting its usefulness in our setting to a small number of models. Our system searches for consistency in *distribution of relation* among primitive faces, similar to [RE11], but our representation is simpler and specifically designed for noisy, partial scans instead of full 3-D models.

3. Overview

Our goal is to quickly assess the quality of the current scan and guide the user in subsequent scans. However, there are several challenges: (i) the system has to assess model quality without necessarily knowing he model; (ii) the scans are potentially incomplete, with large parts of data missing; and (iii) the system should respond in real-time.

First, we observe that existing database models such as Trimble 3D warehouse models can be used as *proxies* for evaluating scan quality of similar objects being scanned, thus addressing the first challenge. Hence, for any merged query scan (i.e., pointcloud) S, the system looks for a match among similar models in the database $\mathcal{M} = \{M_1, \cdots M_N\}$. For simplicity, we assume that the models to have a consistent upright orientation as commonly found in existing databases.

To handle the second challenge, we note that missing data, even in large chunks, are mostly the result of self occlusion, and hence are predictable. Specifically, our system synthetically scans the models M_i from different viewpoints to simulate such self occlusions. This greatly simplifies the problem by allowing us to directly compare S to the simulated scans of M_i, thus automatically accounting for missing data in S.

Finally, to achieve real-time performance, we propose a simple, robust, yet effective descriptor to match S to view-dependent scans of M_i. Subsequently, the system performs registration to verify the match between each matched simulated scan and the query scan, and returns the most similar simulated scan and the corresponding model M_i.

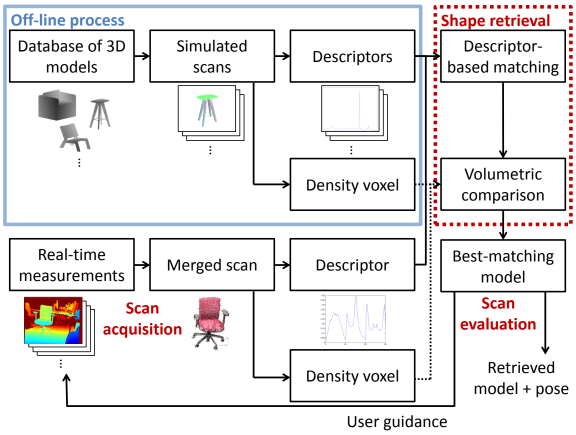

Figure 2: *Pipeline of the proposed system.*

Figure 2 illustrates the pipeline of our guided real-time scanning system, which consists of a scanning device (Kinect in our case) and a database of 3D shapes containing the categories of the shapes present in the environment. For both domains, we produce pointcloud data and calculate the A2h descriptors and density voxels (see Section 4). Partial shape retrieval proceeds in two stages: candidate matches are quickly found using the proposed descriptor followed by comparison using density voxels. The retrieved shape is then compared with the scanned point set to provide guidance to the user, if necessary. The process iterates until a sufficiently good match is found (see supplementary video).

3.1. Interface Design

The real-time system guides the user to scan an object and retrieve the closest match. In our study, we used the Kinect scanner for the acquisition and the retrieval process took 5-10 seconds/iteration on our unoptimized implementation. The user scans an object from an operating distance of about 1-3m and we assume that the query object is separated from background and placed on the ground plane, thus upright orientation can be easily acquired. The sensor data of real-time video stream of depth pointcloud and color images are visible to the user at all times (see Figure 3, bottom-left).

Initialization. The user starts scanning by pointing the sensor to the ground plane. The ground plane is detected if the sensor captures a dominant plane that covers more than 50% of the scene. Our system uses this plane to extract the upright direction of the captured scene. When the ground plane is successfully detected, the user receives an indication on the screen (Figure 3 top-right).

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

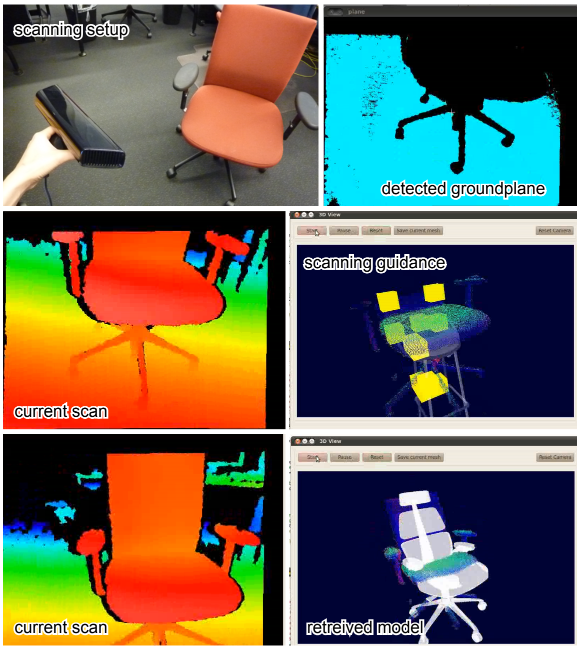

Figure 3: *The proposed guided real-time scanning setup is simple to use. The user starts by scanning using a Microsoft Kinect (top-left). The system first detects the ground plane and the user is notified (top-right). The current pointcloud corresponding to the target object is displayed in the 3D view window, the best matching database model is retrieved (overlaid in transparent white), and the predicted missing voxels are highlighted as yellow voxels (middle-right). Based on the provided guidance, the user acquires the next frame of data, and the process continues. Our method stops when the retrieved shape explains well the captured pointcloud. Finally, the overlaid 3D shape is highlighted in white (bottom-right). Note that the accumulated scans have significant parts missing in most scanning steps.*

Scan acquisition. The input stream is collected and processed using an open-source implementation [EEH*11] that calibrates the color and depth measurements and outputs the pointcloud data. The color features of individual frames are then extracted and matched from consecutive frames. The corresponding depth values are used to incrementally register the depth measurements [HKH*12]. Note that the viewpoint of the sensor cannot drastically change from the initial point-of-view due to the limitation of real-time registration, and we thus we cannot simply extend existing shape retrieval methods using full models. The pointcloud that belongs to the object is segmented as the system detects the ground plane and exclude the points that belong to the plane and depth thresholding. We will refer to the segmented, registered set of depth measurements as a *m*erged scan S. In a separate window, the pointcloud data corresponding to the object being captured is continuously displayed.

Partial shape retrieval. Whenever each new frame is processed, the system calculates the A2h descriptor and the density voxels from the pointcloud data for the merged scan. For shape retrieval, our system first performs a descriptor-based similarity search against the entire database to obtain a candidate set of similar models. Finally, the system performs registration of each candidate model with the merged scan and returns the model with the best alignment score. The challenge is how to maintain real-time performance during the retrieval process. The technical details of the shape retrieval process is described in Section 4.

Scan evaluation. Once the best matching model is retrieved, the proxy is displayed for the user overlaid with yellow voxels (Figure 3 middle-right). The yellow voxels indicate where the missing data is, and the user can then acquire the next scan around the area. Specifically, the system compares the density voxels of the best-matching model and the current merged scan S, and highlights the voxels that the matched model has more than average number of points, but the current measurement has low density. Based on this guidance, the user can then acquire the next scan. The system automatically finishes when the retrieved best match model is close enough to the current measurement (when the missing voxels are less than 1% of total number of voxels) (Figure 3 bottom-right).

4. Partial Shape Retrieval

As mentioned in Chapter 3, we use simulated scans to predict self-occlusion and compare database of models against partial, noisy scan data (Section 4.1). We take a two-stage process to achieve both accuracy and efficiency. First, candidate matches are retrieved from a large database of model with help of our suggested new descriptor (Section 4.2). From the first stage, the system keeps a few top candidate matches that have to be verified by more detailed volumetric comparison (Section 4.3) based on density voxels to find the best match.

While we observed the A2h descriptor is efficient and good at capturing local geometric features, the entire shape structure can better be captured by aligning the 3-D shapes and comparing relative 3-D locations. Both alignment step and volumetric comparison step require more computation time given the size of our database models, and we perform them only for top 25 candidate matches extracted using the proposed descriptor. The number of candidate matches (25) after descriptor comparison is chosen empirically based on our performance on the real scan data, that contain at least 3-5 good matched models after about 10 seconds of scanning session while still maintaining real-time performance.

4.1. View-Dependent Simulated Scans

For each model M_i, the system generates simulated scans $S^k(M_i)$ from multiple camera positions, where the super-

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

script k represent the index to the camera location. Let $\vec{d_{up}}$ denote the up-right orientation for model M_i. Our system takes $\vec{d_{up}}$ as the z-axis and arbitrarily fixes any orthogonal direction $\vec{d_i}$ (i.e., $\vec{d_i}^T \vec{d_{up}} = 0$) as the x-axis. The system also translates the centroid of M_i to the origin.

The system then virtually positions the cameras at the surface of a view-sphere around the origin. Specifically, the camera is placed at

$$\vec{c_i} := (2d\cos\theta\sin\phi, 2d\sin\theta\sin\phi, 2d\cos\phi)$$

where d denotes the length of the diagonal of the bounding box of M_i, and ϕ denotes the camera latitude. The camera up-vector is defined as

$$\vec{u_i} := \frac{\vec{d_{up}} - < \vec{d_{up}}, \bar{\vec{c_i}} > \bar{\vec{c_i}}}{\|\vec{d_{up}} - < \vec{d_{up}}, \bar{\vec{c_i}} > \bar{\vec{c_i}}\|} \quad \text{with } \bar{\vec{c_i}} = \vec{c_i}/\|\vec{c_i}\|$$

and the gaze point is defined as the origin. The fields of view are set to $\pi/2$ in both the up and horizontal directions.

For each such camera location, our system obtains a synthetic scan using the z-buffer with a grid setting of 200×200. Our system places K samples of θ, i.e., $\theta = 2k\pi/K$ where $k \in [0, K)$ and $\phi = \{\pi/6, \pi/3\}$ to obtain view-dependent simulated scans for each model M_i. Empirically, we set $K = 6$ to balance between efficiency and quality when comparing simulated scans and the merged scan S. Note that this is a very coarse sampling compared to conventional settings for light-field descriptor or view-point based approach, but $K = 6$ was good enough to capture necessary self-occlusion utilizing the depth information.

4.2. A2h Scan Descriptor

Our goal is to design a descriptor that (i) is efficient to compute, (ii) is robust to noise and outliers, and (iii) has a low-memory footprint. We draw inspiration from shape distributions [OFCD02] that computes statistics about geometric quantities that are invariant to global transforms, e.g., distances between pairs of points on the models. Shape distribution descriptors, however, were designed to be resilient to local geometric changes. Hence, they are ineffective in our setting, where shapes are distinguished by subtle local features. Instead, our system computes the distributions of angles between point normals, which better capture the local geometric features. Further, since the system knows the up-right direction of each shape,this information is incorporated into the design of the descriptor.

Specifically, for each scan S (real or simulated), our system first allocates the points into three bins based on their height along the z-axis, i.e., the up-right direction. Then, among the points within each bin, the system computes the distribution of angles between all pairs of points. Recall that the points have associated normal directions. The angle space is discretized using 50 bins between $[0, \pi]$, e.g., each bin counts the frequency of normal angles within each bin.

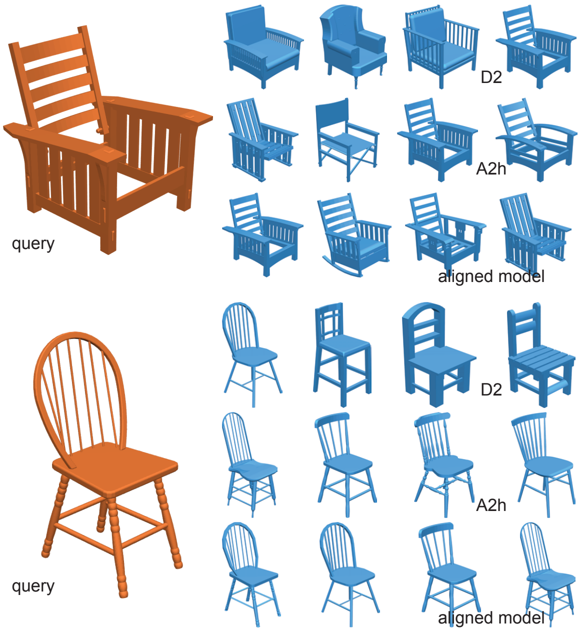

Figure 4: *Representative shape retrieval results using the D2 descriptor ([OFCD02], first row), the A2h descriptor introduced in this chapter (Section 4.2, second row), and the aligned models after scan registration (Section 4.3, third row). For each method, we only show the top 4 matches. The D2 and A2h descriptor (first two rows) are compared by histogram distributions, which is a quick and efficient. Empirically, we observed the A2h descriptor to better capture local geometric features compared to the D2 descriptor, with local registration further improving the retrieval quality. The comparison based on 3D alignment (third row) is more accurate, but require more computation time, and cannot be performed in real-time given the size of our database of models.*

We call this the *A2h scan descriptor*, which for each point cloud is a $50 \times 3 = 150$ dimensional vector; this collects the angle distribution within each height bin.

In practice, for pointclouds belonging to any merged scan, our system randomly samples 10,000 pairs of points within each height bin to speed-up the computation. In our extensive tests, we found this simple descriptor to perform better than distance-only histograms in distinguishing fine variability within a broad shape class (see Figure 4).

4.2.1. Descriptor-Based Shape Matching

A straightforward way to compare two descriptor vectors $\vec{f_1}$ of $\vec{f_2}$ is to take the L_p norm of their difference vector $\vec{f_1} - \vec{f_2}$. However, the L_p norm can be sensitive to noise and does not account for the similarity of distribution between similar curves. Instead, our system uses the Earth Mover's distance (EMD) to compare a pair of distributions [RTG98]. Intu-

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

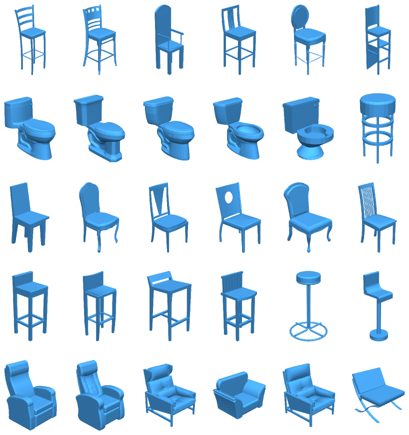

Figure 5: *Retrieval results with simulated data using a chair data set. Given the model in the first column, the database of 2138 models are matched using the A2h descriptor, and the top 5 matches are shown.*

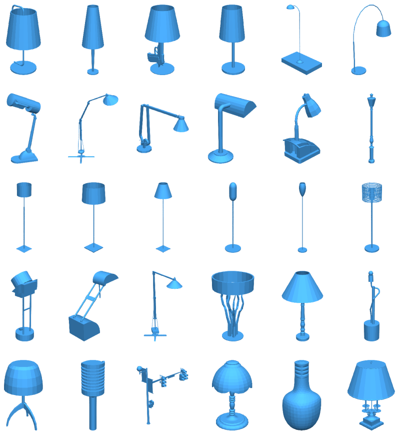

Figure 6: *Retrieval results with simulated data using a lamp data set. Given the model in the first column, the database of 1805 models are matched using the A2h descriptor, and the top 5 matches are shown.*

itively, given two distributions, one distribution can be seen as a mass of earth properly spread in space, the other distribution as a collection of holes that need to be filled with that earth. Then, the EMD measures the least amount of work needed to fill the holes with earth. Here, a unit of work corresponds to transporting a unit of earth by a unit of ground distance. The costs of "moving earth" reflect the notion of nearness between bins; therefore the distortion due to noise is minimized. In a 1D setting, EMD with L_1 norms is equivalent to calculating an L_1 norm for cumulative distribution functions (CDF) of the distribution [Vil03]. Hence, our system achieves robustness to noise at the same time complexity as calculating an L_1 norm between the A2h distributions. For all of the presented results, our system used EMD with L_1 norms of the CDFs computed from the A2h distributions.

Because there are $2K$ view-dependent pointclouds associated with each model M_i, the system matches the query S with each such pointcloud $S^k(M_i)$ $(k = 1, 2, ..., 2K)$ and records the best matching score. In the end, the system returns the top 25 matches across the models in $\mathcal{M}$.

4.3. Volumetric Comparison

We create density voxels from the merged scan S and compare it against the density voxels of the candidate matches of the database models. The density voxels of database models are calculated during the pre-processing stage. Specifically, the bounding box of M_i is discretized into uniform voxel grids and the density of points that falls within the voxel location is calculated. The resolution of the voxel is chosen considering the trade-off between the accuracy and the speed. In our tests, we used a $9 \times 9 \times 9$ voxel grid.

The merged scan S and the retrieved model M_i can be aligned as follows: the system first aligns the centroid of the simulated scan $S^k(M_i)$ to match the centroid of S (note that we do not force the model M_i to touch the ground), while scaling model M_i to match the height of data. To fix the remaining 1DOF rotational ambiguity, the angle space is discretized into 10° intervals, and total 36 density voxels are created for a given scan S. The system compares the volume density between S and M_i in terms of the cross correlation of the density voxels and picks the angle for which the rotated model best matches the scan S. In practice, we found this refinement step necessary since our view-dependent scans have coarse angular resolution ($K = 6$).

Finally, the system uses the positioned proxy model M_i to assess the quality of the current scan and provide guidance, as described in Section 3.1.

5. Evaluation

We tested the robustness of the proposed A2h descriptor on synthetically generated data against available groundtruth. Further, we let novice users use our system to scan different indoor environments. The real-time guidance allowed the users to effectively capture the indoor scenes (see supplementary video).

Table 1: *Database and scan statistics.*

dataset	# models	average # points/scan
chair	2138	45068
couch	1765	129310
lamp	1805	11600
table	5239	61649

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

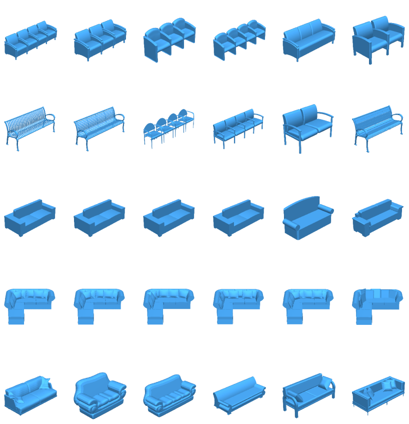

Figure 7: *Retrieval results with simulated data using a couch data set. Given the model in the first column, the database of 1765 models are matched using the A2h descriptor, and the top 5 matches are shown.*

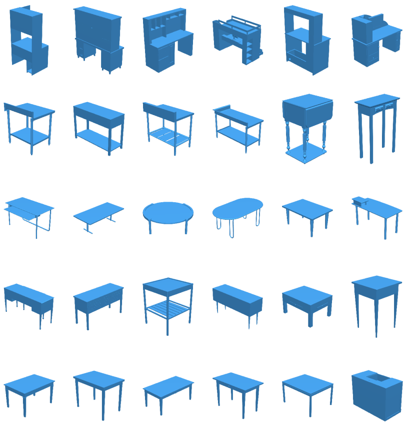

Figure 8: *Retrieval results with simulated data using a table data set. Given the model in the first column, the database of 5239 models are matched using the A2h descriptor, and the top 5 matches are shown.*

Model database. We considered four categories of objects (i.e., chairs, couches, lamps, tables) in our implementation. For each category, we downloaded a large number of models from the Trimble 3D Warehouse (see Table 1) to act as proxy geometry in the online scanning phase. The models were pre-scaled and moved to the origin. We synthetically scanned each such model from 12 different viewpoints and computed the A2h descriptor for each such scan. Note that we placed the camera only above the objects (latitudes of $\pi/6$ and $\pi/3$) as the input scans rarely capture the underside of the objects. We used the Kinect scanner to gather streaming data and used an open source library [EEH*11] to accumulate the input data to produce merged scans.

Retrieval results with simulated data. The proposed A2h descriptor is effective in retrieving similar shapes in fractions of seconds. Figure 5, 7, 6, and 8 show typical retrieval results. In our tests, we found the retrieval results to be useful for chairs and couches, which have a wider variation of angles compared to lamps or tables, the shape of which is almost always very symmetric.

Effect of viewpoints. The scanned data often have significant parts missing, mainly due to self-occlusion. We simulated this effect on the A2h descriptor-based retrieval by comparing the performance of the retrieval using $2K = 12$ scans separately against retrieval with merged all $2K$ scans, Figure 9. We found the retrieval results to be robust and the models sufficiently representative to be used as proxies for subsequent model assessment.

Comparison with other descriptors. We also tested existing shape descriptors: silhouette-based light field descriptor [CTSO03], local spin image [Joh97], and the D2 descriptor [OFCD02]. In all the cases, we found our A2h descriptor to be more effective in quickly resolving local geometric changes, particularly for low quality partial pointclouds. In contrast, we found the light field descriptor to be more susceptible to noise and cannot be easily applied to the partial

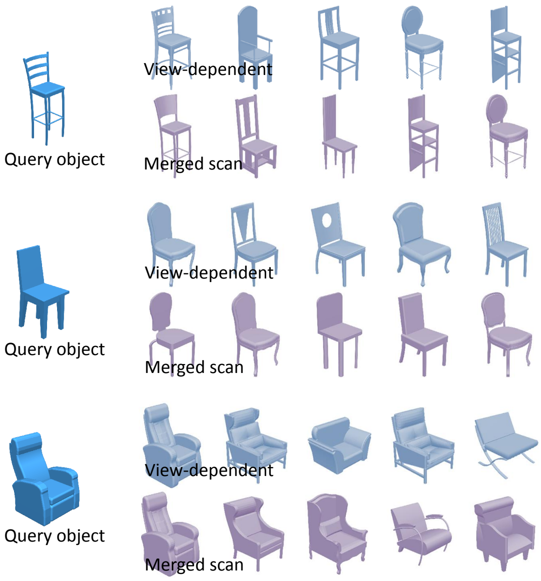

Figure 9: *Comparison between retrieval with view-dependant and merged scans. The models are sorted by matching scores, with lower scores denoting better matches. The leftmost images show the query model. Note that the view-dependent scan-based retrieval are robust even with significant missing regions (∼30-50%).*

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

pointcloud data. Local spin image was more expensive to compute to achieve the real-time performance. The D2 descriptor was less able to distinguish between local variations than our A2h descriptor. We show sample retrieval results in Figure 4.

We next evaluated the degradation in the retrieval results under perturbations in sampling density and noise.

Effect of density. During scanning, points are sampled uniformly on the sensor grid, instead of uniformly on the model surface. This uniform sampling on the sensor grid results in varying densities of scanned points depending on the viewpoint. Our system compensates for this effect by assigning probabilities that are inversely proportional to the density of sample points.

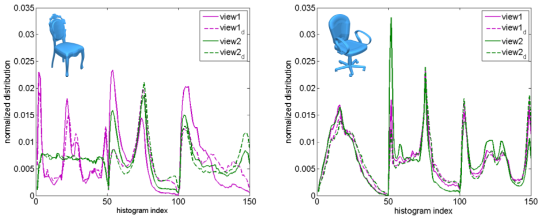

Figure 10: *Effect of density-aware sampling on two different combination of views (comb1 and comb2). The sampling that considers the density of points are comb1$_d$ and comb2$_d$, respectively.*

Figure 10 shows the effect of density compensation on the histogram distributions. We tested two different combination of viewpoints and compared the distributions, using sampling based on uniform distribution or inversely proportional to the density. Density-aware sampling are indicated by dotted lines. The overall shapes of the graphs are similar for uniform and density-aware samplings. However, the absolute values on the peaks are observed at similar heights while using density-aware sampling. Hence, our system uses density-aware sampling to achieve robustness to sampling variations.

Effect of noise. In Figure 11, we show the robustness of A2h histograms under noise. Generally, the histograms become smoother under increasing noise as subtle orientation variations get masked. For reference, the Kinect measurements from a distance range of 1-2m have noise perturbations comparable to 0.005 noise in the simulated data. We added synthetic Gaussian noise on the simulated data to calculate the A2h descriptors to better simulate the shape of the histogram.

Retrieval results with real data. Figure 13 shows retrieval results on a range of objects (i.e., chairs, couches, lamps, and tables). Overall we found the guided interface to work well in practice. The performance was better for chairs and couches, while for lamps and tables, the thin and symmetric structures led to some failure cases. In all cases, the system successfully handled missing data as high as 40-60% of the object surface (or half of the object surface invisible) and the response of the system was at interactive rates. Note that for testing purposes we manually pruned the input database models to leave out models (if any) that looked very similar to the target objects to be scanned. Please refer to the supplementary video for the system in action.

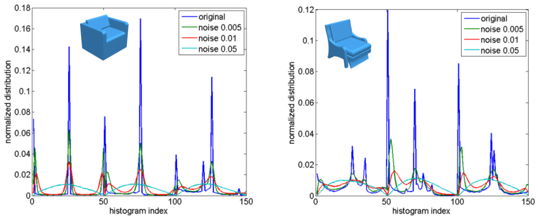

Figure 11: *Effect of noise. The shape of histogram becomes smoother as the level of noise increases.*

Comparison. There is no prior work that exactly does real-time retrieval as we propose. We performed limited comparison with the segmented scans provided by [NXS12], which segments real scans and matches a limited number of objects from database. The overall goal is different because we are retrieving models for individual objects in real-time, while the work by [NXS12] interprets a scene with multiple objects. However, both approaches share the process of matching segments of pointcloud with a database of objects. For the particular process, our approach has the benefit of having a larger size of database and quick retrieval results. We are using thousands of models in real-time while the approach by [NXS12] used only 8 chairs and 8 desks. According to their paper, the query time for randomized decision forest taking a fraction of second, but entire deformation and align-

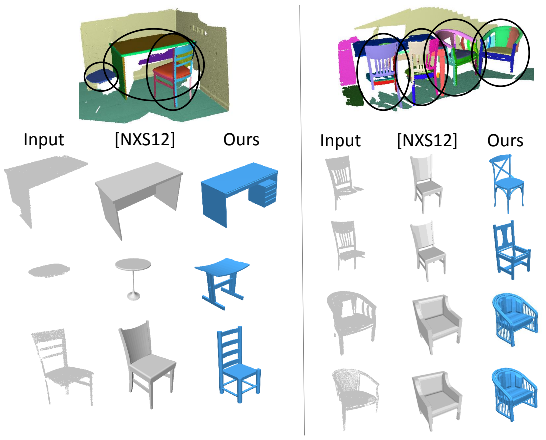

Figure 12: *Comparison with [NXS12].*

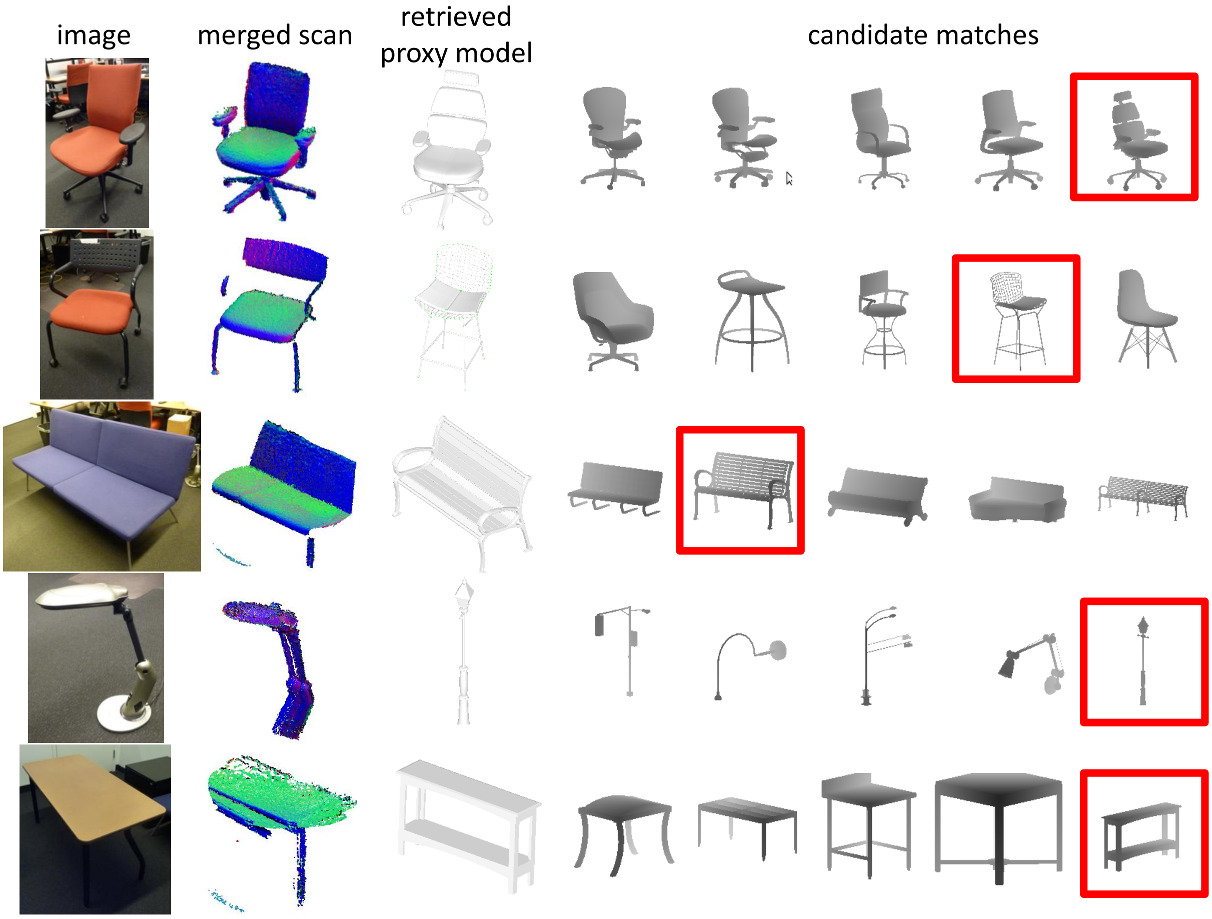

Figure 13: *Real-time retrieval results on various datasets. For each set, we show the image of the object being scanned, the accumulated pointcloud, and the closest shape retrieved model, along with the 5 out of the top 25 candidates that are picked from the database of thousands of models using the proposed A2h descriptor.*

ing the object takes 10 seconds. On the other hand, our retrieval takes less than a second to compare against thousands of models given the merged scan. (From the supplementary video, you can see that the entire retrieval can take a few seconds, but it is actually performed for every frame added to the scan.) Figure 12 shows some sample matched models provided by their pipeline and ours. Also we can possibly incorporate user feedback for the objects with severe occlusion to get a better reconstruction result.

6. Conclusions

We presented a real-time guided scanning setup for online quality assessment of streaming RGBD data obtained while acquiring indoor environments. The proposed approach is motivated by three key observations: (i) indoor scenes largely consist of a few different types of objects, each of which can be reasonably approximated by commonly available 3D model sets; (ii) data is often missed due to self-occlusions, and hence such missing regions can be predicted by comparisons against synthetically scanned database models from multiple view-points; and (iii) streaming scan data can be robustly and effectively compared against simulated scans by a direct comparison of the distribution of relative local orientations in the two types of scans. The best retrieved model is then used as a proxy to evaluate the quality of the current scan and guide subsequent acquisition frames. We have demonstrated the real-time system on a large number of synthetic and real-world examples with a database of 3D models, often ranging in a few thousands.

In the future, we would like to extend our guided system to create online reconstructions while specifically focusing on generating semantically valid scene models. Using context information in the form of structural cues [MWZ*13] can prove to be effective.

© 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and John Wiley & Sons Ltd

Acknowledgements

We are grateful to the anonymous reviewers for their feedback and Han Liu for some initial testing. The project was supported in parts by NSF grants CCF 1011228 and 1161480, ONR MURI award N00014-13-1-0341, a Google research award, the Max Planck Center for Visual Computing and Communications, a Marie Curie CIG, and ERC Starting grant SmartGeometry.

References

[BP13] BAE M. S., PARK I. K.: Content-based 3d model retrieval using a single depth image from a low-cost 3d camera. *The Visual Computer 29*, 6-8 (2013), 555–564. 3

[CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3d model retrieval. *CGF 22*, 3 (2003), 223–232. 3, 7

[DHR*11] DU H., HENRY P., REN X., CHENG M., GOLDMAN D. B., SEITZ S. M., FOX D.: Interactive 3d modeling of indoor environments with a consumer depth camera. In *Proc. Ubiquitous computing* (2011), pp. 75–84. 2

[EEH*11] ENGELHARD N., ENDRES F., HESS J., STURM J., BURGARD W.: Real-time 3D visual SLAM with a hand-held RGB-D camera. In *RGB-D Workshop on 3D Perception in Robotics* (2011). 2, 4, 7

[FKMS05] FUNKHOUSER T., KAZHDAN M., MIN P., SHILANE P.: Shape-based retrieval and analysis of 3d models. *Commun. ACM 48*, 6 (June 2005), 58–64. 3

[HCI*11] HINTERSTOISSER S. HOLZER S., CAGNIART C., ILIC S., KONOLIGE K., NAVAB N., LEPETIT V.: Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. *ICCV* (2011). 3

[HKH*12] HENRY P., KRAININ M., HERBST E., REN X., FOX D.: RGBD mapping: Using kinect-style depth cameras for dense 3D modeling of indoor environments. *I. J. Robotic Res. 31*, 5 (2012), 647–663. 2, 4

[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 *Proc. UIST* (2011), pp. 559–568. 2

[Joh97] JOHNSON A.: *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, Robotics Institute, CMU, 1997. 7

[JTRS12] JAIN A., THORMAHLEN T., RITSCHEL T., SEIDEL H.-P.: Exploring shape variations by 3d-model decomposition and part-based recombination. *CGF (EUROGRAPHICS) 31*, 2 (2012), 631–640. 2

[KAJS11] KOPPULA H., ANAND A., JOACHIMS T., SAXENA A.: Semantic labeling of 3D point clouds for indoor scenes. In *NIPS* (2011), pp. 244–252. 2

[KDS*12] KIM Y. M., DOLSON J., SOKOLSKY M., KOLTUN V., THRUN S.: Interactive acquisition of residential floor plans. In *ICRA* (2012), pp. 3055–3062. 2

[KLM*13] KIM V. G., LI W., MITRA N. J., CHAUDHURI S., DIVERDI S., FUNKHOUSER T.: Learning part-based templates from large collections of 3d shapes. *ACM TOG (SIGGRAPH) 32*, 4 (July 2013). 2

[KMYG12] KIM Y. M., MITRA N. J., YAN D.-M., GUIBAS L.: Acquiring 3d indoor environments with variability and repetition. *ACM TOG 31*, 6 (2012). 3

[LGHK10] LEE D. C., GUPTA A., HEBERT M., KANADE T.: Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS* (2010), pp. 1288–1296. 2

[MPWC12] MITRA N. J., PAULY M., WAND M., CEYLAN D.: Symmetry in 3d geometry: Extraction and applications. In *EUROGRAPHICS State-of-the-art Report* (2012). 2

[MWZ*13] MITRA N. J., WAND M., ZHANG H., COHEN-OR D., BOKELOH M.: Structure-aware shape processing. In *EUROGRAPHICS State-of-the-art Report* (2013). 9

[NXS12] NAN L., XIE K., SHARF A.: A search-classify approach for cluttered indoor scene understanding. *ACM TOG (SIGGRAPH Asia) 31*, 6 (2012). 3, 8

[OFCD02] OSADA R., FUNKHOUSER T., CHAZELLE B., DOBKIN D.: Shape distributions. *ACM Transactions on Graphics 21*, 4 (Oct. 2002), 807–832. 3, 5, 7

[PMG*05] PAULY M., MITRA N. J., GIESEN J., GROSS M., GUIBAS L. J.: Example-based 3D scan completion. In *Symp. on Geometry Proc.* (2005), pp. 23–32. 3

[PMW*08] PAULY M., MITRA N. J., WALLNER J., POTTMANN H., GUIBAS L.: Discovering structural regularity in 3D geometry. *ACM TOG (SIGGRAPH) 27*, 3 (2008), 43:1–43:11. 2

[RBF12] REN X., BO L., FOX D.: RGB-D scene labeling: Features and algorithms. In *CVPR* (2012), pp. 2759 – 2766. 2

[RE11] RAMEZANI M., EBRAHIMNEZHAD H.: 3d object categorization based on histogram of distance and normal vector angles on surface points. In *MVIP* (2011), pp. 1–5. 3

[RHHL02] RUSINKIEWICZ S., HALL-HOLT O., LEVOY M.: Real-time 3D model acquisition. *ACM TOG (SIGGRAPH) 21*, 3 (2002), 438–446. 2

[RTG98] RUBNER Y., TOMASI C., GUIBAS L. J.: A metric for distributions with applications to image databases. In *ICCV* (1998), pp. 59–. 5

[SFC*11] SHOTTON J., FITZGIBBON A., COOK M., SHARP T., FINOCCHIO M., MOORE R., KIPMAN A., BLAKE A.: Real-time human pose recognition in parts from a single depth image. In *CVPR* (2011), pp. 1297–1304. 2

[SPT12] SFIKAS K., PRATIKAKIS I., THEOHARIS T.: 3D object retrieval via range image queries based on sift descriptors on panoramic views. In *Proc. 3DOR* (2012). 3

[SWK07] SCHNABEL R., WAHL R., KLEIN R.: Efficient RANSAC for point-cloud shape detection. *CGF (EUROGRAPHICS) 26*, 2 (2007), 214–226. 2

[SXZ*12] SHAO T., XU W., ZHOU K., WANG J., LI D., GUO B.: An interactive approach to semantic modeling of indoor scenes with an RGBD camera. *ACM TOG (SIGGRAPH Asia) 31*, 6 (2012). 3

[TSS10] TRIEBEL R., SHIN J., SIEGWART R.: Segmentation and unsupervised part-based discovery of repetitive objects. In *Proc. of Robotics: Science and Systems* (2010). 2

[TW05] THRUN S., WEGBREIT B.: Shape from symmetry. In *ICCV* (2005), pp. 1824–1831. 2

[Vil03] VILLANI C.: *Topics in Optimal Transportation*. Graduate Studies in Mathematics. American Mathematical Society, 2003. 6

[XS12] XIANG Y., SAVARESE S.: Estimating the aspect layout of object categories. In *CVPR* (2012), pp. 3410–3417. 2

[ZCC*12] ZHENG Y., CHEN X., CHENG M.-M., ZHOU K., HU S.-M., MITRA N. J.: Interactive images: Cuboid proxies for smart image manipulation. *ACM TOG (SIGGRAPH) 31*, 4 (2012), 99:1–99:11. 2