Exploring Collections of 3D Models using Fuzzy Correspondences

Vladimir G. Kim, Princeton University
Wilmot Li, Adobe Systems
Niloy J. Mitra, UCL
Stephen DiVerdi, Adobe Systems
Thomas Funkhouser, Princeton University

Abstract

Large collections of 3D models from the same object class (e.g., chairs, cars, animals) are now commonly available via many public repositories, but exploring the range of shape variations across such collections remains a challenging task. In this work, we present a new exploration interface that allows users to browse collections based on similarities and differences between shapes in user-specified regions of interest (ROIs). To support this interactive system, we introduce a novel analysis method for computing similarity relationships between points on 3D shapes across a collection. We encode the inherent ambiguity in these relationships using fuzzy point correspondences and propose a robust and efficient computational framework that estimates fuzzy correspondences using only a sparse set of pairwise model alignments. We evaluate our analysis method on a range of correspondence benchmarks and report substantial improvements in both speed and accuracy over existing alternatives. In addition, we demonstrate how fuzzy correspondences enable key features in our exploration tool, such as automated view alignment, ROI-based similarity search, and faceted browsing.

Keywords: database analysis, fuzzy correspondence, shape analysis, exploration, model collections.

Links: DL PDF WEB VIDEO DATA CODE

1 Introduction

Increasing availability of powerful modeling software and 3D acquisition devices has led to rapidly growing repositories of 3D models (e.g., TurboSquid, Google 3D Warehouse, etc.). Yet, the task of exploring such large 3D repositories remains an important and challenging problem. In particular, while most online databases make it easy for users to select sets of similar models (which we refer to as *model collections*) using text-based filtering, understanding the range of variations *within* such collections is typically much more difficult (see also [Ovsjanikov et al. 2011]).

For many object classes, one key challenge is that the shape can vary in many different ways, and users may be interested in exploring different types of variations. For example, within a collection of chair models, one user may want to see how the backs of chairs attach to the seat (smoothly merge, right angle, etc.) while another user may only want to see chairs whose back legs are at a certain angle (see Figure 1). Even within a single exploration session, a user may want to define the exploration space using multiple different attributes (e.g., chairs with a stem base and curved back). Given the spectrum of different possible exploration criteria, a static predefined organization of the data is clearly not sufficient. This is especially true for very large and diverse collections, like the ones we find for many object classes in the Google 3D Warehouse (e.g., chairs, cars, animals).

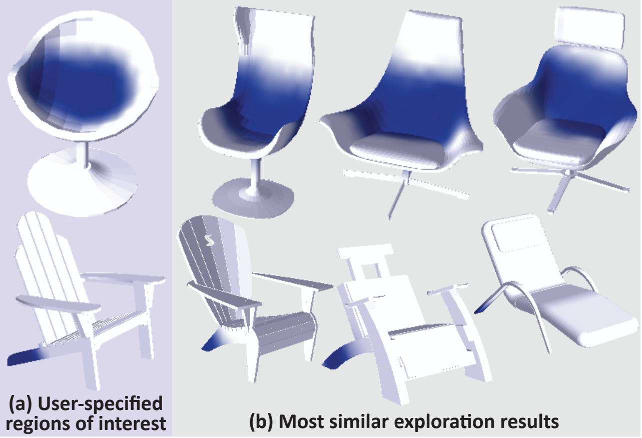

(a) User-specified regions of interest
(b) Most similar exploration results

Figure 1: Region-based exploration of chairs. *The user specifies exploration criteria by selecting regions of interest (a), and our system sorts models based on their similarity within those regions (b). Corresponding regions are highlighted in dark-blue.*

In this work, we present a new analysis tool and exploration interface for 3D model collections. As a key feature, we allow users to directly specify regions of interest (ROI) on example shapes in order to guide subsequent exploration actions. Thus we can support the browsing scenarios described above; the user selects the appropriate ROIs on one or more chairs, and the system automatically organizes the rest of the chairs based on their similarity to the specified region. From a Human Computer Information Retrieval perspective, our system can be described as a type of *faceted browsing* interface, which has proven to be a very effective and popular method for exploring diverse collections of items with a wide variety of attributes [Hearst 2006]. For example, faceted interfaces are very common for online shopping websites where attributes like price, date of release, popularity, etc. act as predefined facets for navigation. In contrast, our facets are defined interactively as the user selects specific ROIs (see supplementary video).

The main technical challenge in realizing our proposed interface is how to relate any arbitrary user selected region on one shape to all the other models in the collection. Since user-selected ROIs do not necessarily match pre-segmented parts, consistent part-level segmentation is not sufficient. Rather, we prefer correspondences for individual points. Yet, automatically establishing one-to-one point correspondences across a model collection is difficult and often ambiguous, especially when the collection includes diverse shapes. For example, how do the chair bases in the the top row of Figure 1 correspond to the chair legs in the bottom row?

To address this issue, we encode the ambiguity of correspondences in model collections with *fuzzy correspondences*[1]. Specifically, given a collection of N shapes $\mathbf{S} := \{S_1, S_2, \ldots, S_N\}$, we use fuzzy

[1] We use the word "fuzzy" as previously used in non-rigid surface matching [Chui and Rangarajan 2003] rather than fuzzy set theory.

correspondences as a function $f(p_i, p_j) : \mathbf{S} \times \mathbf{S} \rightarrow \mathbb{R}$ to denote a continuous similarity measure between points $p_i \in S_m$ and $p_j \in S_n$.

To estimate fuzzy correspondences, $f(p_i, p_j)$, we utilize geometric matching methods that align pairs of shapes. Although these methods are computationally expensive and often produce noisy alignments, we observe that for collections of shapes from the same class a correspondence matrix that stores high values for corresponding pairs of points is (i) sparse, (ii) low-rank, and (iii) its rank does not depend on the number of models. We propose a method based on diffusion maps [Nadler et al. 2006] to reconstruct f from sparse and noisy samples (i.e., pairwise alignments) and an iterative procedure to refine f by adaptively sampling based on the current estimate.

We test the accuracy of our estimate of f using the correspondence benchmark for intrinsically-similar shapes [Kim et al. 2011]. We also introduce a new correspondence benchmark of 111 chairs and 86 commercial airplanes using data obtained from the Google 3D Warehouse. Our method successfully utilizes the collection to improve alignments of shapes in comparison to existing methods (see Figures 11, 12, and supplementary material).

Contributions. In summary, we

- introduce an approach for using fuzzy correspondences to understand similarity relations across 3D model collections,
- propose a robust and efficient algorithm to compute fuzzy correspondences from sparse and noisy pairwise alignments,
- evaluate our algorithm on correspondence benchmarks and report substantial improvement over existing alternatives, and
- present an interactive exploration tool for large model collections that uses fuzzy correspondences to support view alignment, ROI-based similarity search, and faceted exploration.

2 Related Work

Analyzing model pairs. Various algorithms have been proposed to align model pairs under different conditions, such as rigid or isometric deformations, partial overlap, and missing data (e.g., [Besl and McKay 1992; Mitra et al. 2004; Li et al. 2008]). In the absence of a good initial alignment, researchers have used manual correspondence for initial placement [Kraevoy and Sheffer 2004; Schreiner et al. 2004], or proposed purely automatic methods to search for rigid [Aiger et al. 2008] and as-isometric-as-possible maps [Bronstein et al. 2006; Lipman and Funkhouser 2009; Ovsjanikov et al. 2010; Sun et al. 2010; Kim et al. 2011]. Such methods, however, make various assumptions about model quality and geometric similarity across the models (e.g., near isometry, same topology, etc.). Hence, most existing point-to-point methods (see [van Kaick et al. 2011]) focus on databases of human bodies, body parts, and quadruped animals, rather than heterogeneous collections with diverse shape variations (e.g., in Google 3D Warehouse). For example, even manually prescribing point-to-point correspondences among the chairs in Figure 2 is ambiguous and difficult.

Analyzing model collections. Extracting relationships across an entire collection of models from the same class often yields more robust results than analyzing isolated model pairs. Several recent efforts investigate this strategy for a variety of applications, including analyzing collections of segmented images to identify interesting relationship pathways [Heath et al. 2010] and analyzing 3D scene graphs to extract placement relationships across various objects in a scene [Fisher et al. 2011]. Here, we focus on previous work that analyzes model collections to compute or refine correspondences.

Some recent work leverages collections to extract part-level segmentations. For example, Kalogerakis et al. [2010] propose a machine learning approach to simultaneously segment and label parts in 3D meshes that uses a collection of hand-labeled models as training data. Other approaches jointly analyze a set of shapes to establish consistent part-to-part correspondences across the collection [Golovinskiy and Funkhouser 2009; Huang et al. 2011; Sidi et al. 2011]. In a concurrent effort, Xu et al. [2012] propose a fuzzy representation for encoding relationship between parts. While such part-level relationships can facilitate exploration (as in the part-based modeling system of Funkhouser et al. [2004]) extracting parts *a priori* (i.e., before the user explores the database) does not allow the user to guide exploration by prescribing arbitrary regions of interest at runtime. Furthermore, computing part-level correspondence maps can be ambiguous for collections with large topological variations and is widely considered to be a very challenging problem (see also van Kaick et al. [2011]).

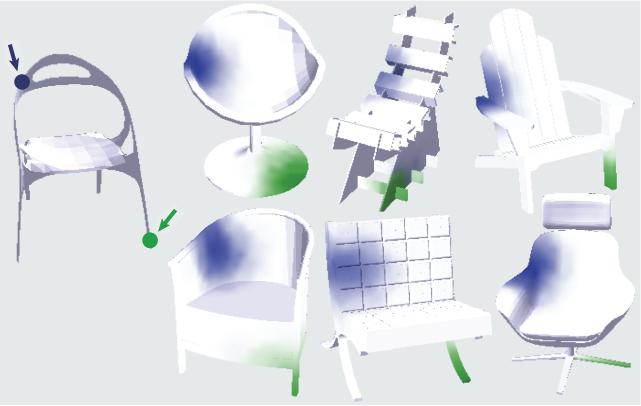


Figure 2: Fuzzy correspondence values for two points. *The blue and green regions of the chairs on the right have the largest fuzzy correspondence to the two selected points on the chair on the left.*

In a related attempt, Nguyen et al. [2011] propose an interesting algorithm to improve point-to-point mappings between model pairs belonging to collections of shapes. Their method is based on the assumption that all cycles of consistent maps must return to identity. They start with $O(N^2)$ pairwise maps between surfaces and then perform an optimization that iteratively improves the consistency of 3-cycles, thus leveraging information from the whole model collection. The method, however, has four important limitations: (i) it requires $O(N^2)$ pairwise alignments, (ii) it computes point-to-point correspondences and thus is not applicable to heterogeneous quality models, (iii) it propagates information only across 3-cycles and thus converges slowly, and (iv) it only aligns pairs of models by concatenating full maps, which limits applicability of this method to heterogeneous datasets where most pairs of models might not have a bijective map between them. We extend their optimization strategy to support fuzzy correspondences, which work with fewer pairwise alignments and more diverse datasets.

Diffusion maps. Introduced by Nadler et al. [2006], diffusion maps provide a probabilistic interpretation of spectral clustering and dimension reduction algorithms and have previously been used for analyzing image collections [Heath et al. 2010], establishing symmetric correspondences [Lipman et al. 2010], and clustering similar segments for consistent segmentation [Sidi et al. 2011]. We use diffusion maps to compute fuzzy correspondences. Our approach has some similarities to that of Lipman et al. [2010] and Sidi et al. [2011], since we use the manifold induced by spectral embedding to estimate correspondences in shapes. In contrast to Lipman et al., our method works on model collections. In contrast to Sidi et al., we focus on point correspondences (rather than parts). As such, our method is well-suited for exploring arbitrary variations of shapes in model collections.

Exploring model collections. Many early efforts to facilitate exploration of 3D model databases aim to group models into meaningful object classes (e.g., horse, chair, car) by clustering in various shape descriptor spaces (see [Shilane et al. 2004] and references therein). Some recent work that builds on this approach allows users to interactively refine search results via relevance feedback [Giorgi et al. 2010]. While these methods categorize models into different classes, our goal is to help users explore and understand variations within a class, which are often more subtle.

Some existing strategies for intra-class exploration learn a deformation model that explains continuous variations across the collection. However, many of these methods focus on a specific shape class, such as human bodies, and assume it is possible to compute consistent point-to-point correspondences between all models (e.g., [Allen et al. 2003; Anguelov et al. 2005]). More recently, Ovsjanikov et al. [2011] present a correspondence-free method that uses a coupled spatial-descriptor space analysis to extract a template-based deformation model that the user can directly manipulate to explore the collection. In contrast, we enable region-based exploration where users can interactively add and adapt navigation criteria. Such guided browsing is particularly useful for diverse collections where the different shape regions vary in distinct ways.

We also draw inspiration from previous work on exploring collections of text-based documents, such as Web pages. In this domain, most traditional interfaces support hierarchical navigation based on predefined categories along with text search using a bag-of-words approach. More recently, *faceted browsing* [Hearst 2006] allows users to dynamically prescribe multiple filters or *facets* to narrow the exploration space. One of our key contributions is to enable a form of faceted browsing for large 3D model collections and demonstrate its potential for interactive exploration.

3 Overview

We introduce an interactive exploration tool that helps users undertand local variations within large collections of shapes. Our system allows users to select an arbitrary ROI on an example shape (a single point, a large fraction of the surface, several disjoint patches, or all of the above) and explore the rest of the collection within the context of that selection. For example, users can see what other shapes have similar/disimilar corresponding regions or get a sense for the range of variation within that region across the collection. Our interface allows multiple ROI-based queries, which enables a form of faceted browsing.

To support this functionality, our system must compute geometric relationships, such as similarity and alignment, between shapes for arbitrary user-specified surface patches. More specifically, when the user paints an ROI, the system should be able to instantly retrieve and align the most (or least) similar shapes from the collection. However, searching a large database to satisfy arbitrary partial shape similarity queries is difficult at interactive rates. On the other hand, precomputing and storing similarities and alignments for all possible ROIs is unrealistic.

Instead, we take a hybrid approach. In an off-line precomputation, we construct *fuzzy correspondences* between points sampled discretely on surfaces in the collection. Then, during any interactive session, our system uses fuzzy correspondences to the ROI to retrieve relevant surfaces and weight the contributions of point samples when computing ROI alignments. Effectively, we perform most of the matching computation in the off-line phase, which infers and leverages low-dimensional structure in the space of shapes spanned by the collection, while deferring only the fine-scale refinement of alignments and the similarity evaluation to the interactive phase.

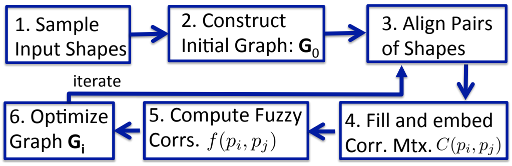

1. Sample Input Shapes
2. Construct Initial Graph: $\mathbf{G}_0$
3. Align Pairs of Shapes
iterate
4. Fill and embed Corr. Mtx. $C(p_i, p_j)$
5. Compute Fuzzy Corrs. $f(p_i, p_j)$
6. Optimize Graph $\mathbf{G}_i$

Figure 3: Computational Pipeline. *In our optimization procedure, we first construct an initial alignment graph $\mathbf{G}_0$, which is further used to fill the correspondence matrix C by aligning shapes connected by an edge. The spectral embedding of C defines the fuzzy correspondence function f, which is used to optimize the alignment graph. We iterate until the process converges.*

The following two sections describe how the off-line system computes fuzzy correspondences (Section 4) and how the interactive system uses them to support interactive exploration (Section 5).

4 Computing Fuzzy Correspondences

We cast the problem of computing fuzzy correspondences as a sampling problem, where the goal is to reconstruct the fuzzy correspondence function $f(p_i, p_j)$. We represent each shape by K discrete points, and thus a discrete representation of f for a database of N shapes is an $NK \times NK$ matrix. To sample entries in f, we use automatic pairwise matching techniques that predict point correspondences based on geometric alignments. While these matching algorithms are often effective at finding semantic correspondences between pairs of similar shapes, they have two primary limitations: (i) geometric shape matching is slow and (ii) matching based on geometry alone can result in semantically incorrect alignments for pairs that differ significantly in geometry or topology, differ by extreme non-homogeneous deformations, and/or have missing or extraneous parts with non-uniform proportions. Thus, the challenge is to reconstruct f with the fewest possible and most robust samples.

Our approach is based on diffusion. First, let us denote an approximate correspondence matrix $C \in \mathbb{R}^{NK \times NK}$ to store computed samples for matched pairs of points. Note that in an ideal case, if we assume that a point on a model corresponds to exactly one unique point on every other model, then the rank of C is independent of the number of models (and equals to the number of points, K). We use diffusion maps to compute a spectral embedding of C that maps each point on a shape to a Euclidean space whose coordinates are the eigenvectors of C scaled by the eigenvalues. We expect the embedded points to lie on a low-dimensional manifold where corresponding points are close to each other, and thus we estimate fuzzy correspondence based on distances in the embedded space (also called diffusion distances). This approach has two main advantages: (i) the embedding (fuzzy correspondences) can be computed without aligning all point pairs, and (ii) it is robust with respect to noise, and thus it overcomes the problems of methods based solely on pairwise alignments.

We now describe our computational pipeline (see Figure 3).

Step 1: Sample input shapes. We represent any shape S_i by a discrete set of sample points P_i. The set is produced by starting with a random vertex and then iteratively adding vertices that are farthest from the set until $|P_i| = K = 128$ [Eldar et al. 1997]. We only estimate the fuzzy correspondence function for pairs of points in P_i across different models, and then interpolate the function to the rest of the shape using nearest-neighbor interpolation.

Step 2: Construct an initial alignment graph. Next, we construct an initial graph $\mathbf{G}_0(\mathbf{S}, \mathbf{A}_0)$ that has the following two properties.

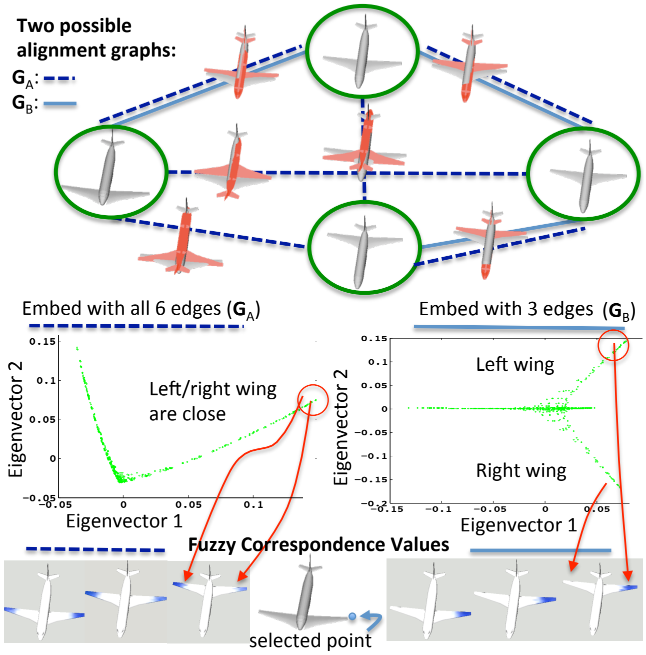


Figure 4: ***Example alignment graphs.*** *A denser alignment graph might result in more noise due to alignment of dissimilar models. A complete graph* $\mathbf{G}_A$ *has more misalignments that result in blending of wings in the embedded space (left), while the linear graph* $\mathbf{G}_B$ *resolves the issue (right).*

First, we want edges in $\mathbf{A}_0$ to only connect shapes that are similar enough to be matched automatically. The importance of this property is illustrated in Figure 4 where airplane models are the nodes of the graph and edges are produced by an affine alignment that minimizes surface distance. Note that the airplanes with wings closer to the nose are misaligned with airplanes that have wings closer to the tail. These misalignments add noise to the complete graph $\mathbf{G}_A$, which results in an erroneous blend of correspondences from one wing to the other (left column). In contrast, a noise-free alignment graph $\mathbf{G}_B$ where only similar models are matched leads to a more accurate embedding (right column). The second desirable property for $\mathbf{G}_0$ is that every pair of shapes should have multiple paths between them so that the embedding is robust to misalignments.

Given these properties, we construct the initial alignment graph as follows. First, we initialize $\mathbf{G}_0$ as the complete graph over all shapes. We then compute the spherical harmonics shape descriptor of the GEDT function [Kazhdan et al. 2003] for every shape and use the L_2 distance between descriptors as edge weights. These weights predict whether the connected shapes are similar enough to be matched automatically (smaller weights suggest higher similarity). Based on these weights, we update $\mathbf{G}_0$ to be the minimal spanning tree. Next, we add edges to improve the graph connectivity. For each node, we select the $3M$ lowest weight edges as candidates, and for each candidate, we compute its *edge rank*, which is a metric proposed by Heath et al. [2010] that estimates the importance of an edge to the overall connectivity of a graph. We then add the highest ranking $N \cdot M$ edges (roughly M edges per shape) to $\mathbf{G}_0$. Choosing a larger value for M improves the connectivity of the alignment graph; we use $M = 5$ in all of our experiments.

Step 3a: Align pairs of shapes extrinsically. Our method can use any pairwise alignment method, and some algorithms may be more appropriate for certain types of collections. For diverse datasets with varying topology (e.g., typical data from Google 3D Warehouse), we find affine transformations to be an effective alignment method. Step 3b describes how to use an intrinsic mapping technique for other types of collections.

Given two shapes S_k and S_l, an alignment score $a_{S_k,S_l,\phi}(p_i \in S_k, p_j \in S_l) : S_k \times S_l \to \mathbb{R}$ is defined over some parameter space ϕ (e.g., all affine transformations T). The score a is defined per pair of points and depends on the quality of the local alignment (L_a) and the global alignment (B_a):

$$a_{S_k,S_l,\phi}(p_i, p_j) := L_a(p_i, p_j) \cdot B_a(S_k, S_l). \tag{1}$$

We define the local term based on Euclidean distance after the transformation $\phi = T$, while the global term represents how well T aligns two shapes globally:

$$L_a(p_i, p_j) := \exp\left(-\frac{D_{\text{Eucl.}}(p_i, T(p_j))^2}{\sigma(S_l)^2}\right) \tag{2}$$

$$B_a(S_k, S_l) := \left(\frac{1}{K}\sum_{p_1 \in P_k} \exp\left(-\frac{D_{\text{Eucl.}}(p_1, T(S_l))^2}{(0.5 \cdot \sigma(S_l))^2}\right)\right) \times \left(\frac{1}{K}\sum_{p_2 \in P_l} \exp\left(-\frac{D_{\text{Eucl.}}(p_2, T^{-1}(S_k))^2}{(0.5 \cdot \sigma(S_k))^2}\right)\right). \tag{3}$$

Note that $\sigma(S)$ depends on the expected ambiguity of pairwise alignments (higher σ captures fuzzier pairwise matching), we use $\sigma(S) = \text{Diam}(S)/5$ for all examples, where $\text{Diam}(S)$ is an average distance between all pairs of points, i.e., $\text{Diam}(S) := \sum_{p_j,p_j \in S} D_{\text{Eucl.}}(p_i, p_j)/K^2$. The global term roughly estimates what fraction of surfaces align under a tighter threshold $\sigma(S)/2$.

Finally, we find the aligning transformation T for a pair of models. In our experience, the models in Google Warehouse have consistent upward orientation, so we only look for optimal 2D rotation and a scale. To avoid optimizing over the space of real-valued parameters, we test all 4 alignments of principal components for a pair of shapes as an initial guess, and locally refine the transformation using ICP optimization [Golovinskiy and Funkhouser 2009]. We choose the transform that maximizes the global alignment score, i.e., $\arg\max_T B_a$. Note that we store all the four aligning transforms with the edge and the transformation can change during the graph optimization (see Step 6).

Step 3b: Align pairs of shapes intrinsically. In cases where the input collection is known to contain smooth, near-isometric manifold surfaces, intrinsic mapping techniques are more suitable. For such datasets, we use blended conformal maps [Kim et al. 2011] as pairwise alignment method. We further define an alignment score based on an intrinsic map $m : S_k \to S_l$. Similarly to the original work, we use a map's area distortion to quantify alignment confidence at a point: $c_m(p \in S_k) := 2 \Big/ \left[\frac{area(N_p)}{area(f(N_p))} + \frac{area(f(N_p))}{area(N_p)}\right]$, where N_p is a 1-ring vertex neighborhood around vertex p. We define local and global terms for the alignment score in Equation 1, as:

$$L_a^{\text{blend}}(p_i, p_j) := c_m(p_i) \cdot \exp\left(-\frac{D_{\text{geod.}}(m(p_i), p_j)^2}{\sigma_{\text{intrinsic}}(S_l)^2}\right) \tag{4}$$

$$B_a^{\text{blend}}(S_k, S_l) := \frac{1}{K}\sum_{p_1 \in P_k} c_m(p_1) \tag{5}$$

where, $\sigma_{\text{intrinsic}}(S_l) = 0.3 \cdot \sqrt{area(S_l)}$.

Note that the method of Kim et al. [2011] produces several intrinsic maps in a presence of near-symmetry (5-10 for examples presented in this paper). Although during the initial alignment we pick a map with the highest score B^{blend}, we make use of all the candidate maps in the graph optimization (see Step 6).

Step 4: Fill and embed the correspondence matrix C. We populate a correspondence matrix C using alignment scores for pairs of

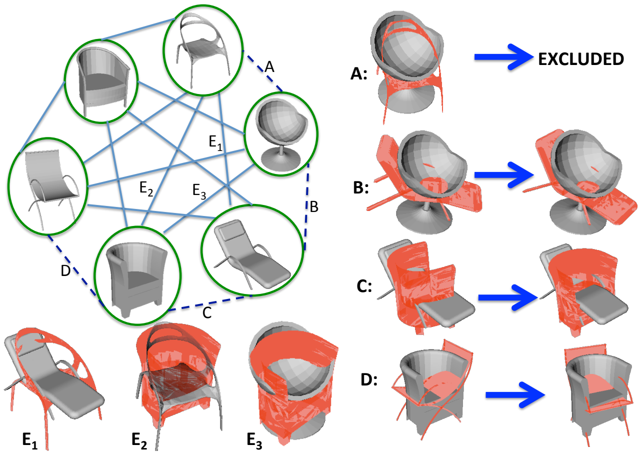


Figure 5: ***Graph optimization.*** *For a small example graph of 6 nodes we show how the edges change during the graph optimization. Edges that correctly align models (solid lines) compensate for the noise introduced by edges where pairwise matching fails (A-D). The graph optimization either re-aligns some models (B-D) or excludes an alignment (A) if no good re-alignment was found. The main issue with (A) is that all possible alignments map seat too close to other chair's base.*

shapes connected by an edge in the alignment graph $\mathbf{G}$. As each row in C is associated with a mapped point, we row-normalize C, labeling it $\tilde{C}$, so that every point is mapped somewhere in the collection with a constant energy:

$$\forall (k,l) \in \mathbf{G}: \qquad \tilde{C}(p_i \in S_k, p_j \in S_l) = \frac{a_{S_k,S_l,\phi}(p_i,p_j)}{\sum_{p_j} a_{S_k,S_l,\phi}(p_i,p_j)}. \tag{6}$$

We further perform spectral analysis of $\tilde{C}$. Let ψ_n be n^{th} eigenvector and λ_n be n^{th} eigenvalue of the matrix $\tilde{C}$. The eigenvectors ψ are normalized as proposed by Nadler et al. [2006], setting $\Psi := D^{-1/2}U$, where U are the original eigenvectors and D is a diagonal matrix with the row sums of C as its entries. The diffusion map at time t ($t = 10$ in our examples, set to a higher value to reduce the influence of noisy alignments) defines the spectral embedding:

$$\Pi_t(p_i) := (\lambda_1^t \psi_1(p_i), \lambda_2^t \psi_2(p_i), \ldots, \lambda_{NK-1}^t \psi_{NK-1}(p_i)) \tag{7}$$

where, $\psi_n(p_i)$ is the i^{th} component of n^{th} eigenvector corresponding to the point p_i. We further define the diffusion distance as Euclidean distance in the embedded space:

$$D_t(p_i,p_j)^2 := \sum_n \lambda_n^{2t} (\psi_n(p_i) - \psi_n(p_j))^2. \tag{8}$$

Step 5: Compute fuzzy correspondence. We define the fuzzy correspondence function as

$$f(p_i,p_j) := \exp\left(-D_t(p_i,p_j)^2/\tau(p_i)^2\right) \tag{9}$$

where, $\tau(p_i)$ is point-specific normalization set equal to the distance to the farthest of 15% of the nearest points (to p_i) in the embedded space. To efficiently find f from the spectral embedding, we set the fuzzy correspondence of all points that are farther than $2\tau(p_i)$ to 0, and based on the low-rank assumption only consider the top K eigenvectors. We search for the nearest neighbors in K-dimensional space using approximate nearest neighbor search (FLANN) [Muja and Lowe 2009].

Step 6: Optimize alignment graph. Given the fuzzy correspondence function f_0 created by the embedding the graph $\mathbf{G}_0$, we update the graph $\mathbf{G}_0$ such that pairwise alignments (i.e., samples) are consistent with the estimated embedding, i.e., f_0. Thus, we want to optimize the graph by (i) detecting and pruning noisy samples, and (ii) adding new samples to the under-sampled areas.

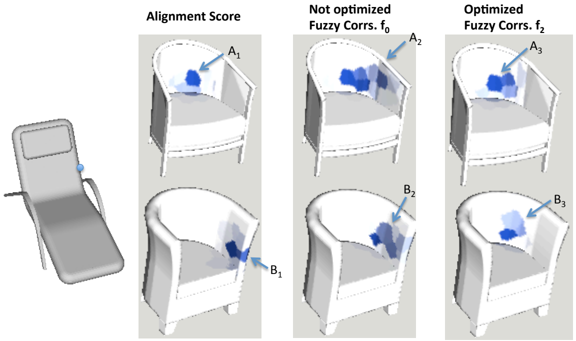


Figure 6: ***Graph optimization: fuzzy correspondences.*** *Fuzzy correspondences before and after the graph optimization shown in Figure 5 – note that the bottom chair is mis-aligned, see Figure 5C. Note that diffusion improves the result for the bottom chair by using indirect paths (B_1 to B_2), however, the results for the top chair get fuzzier due to incorrect alignments (A_1 to A_2). Finally, after the pairwise alignments are fixed, the fuzzy correspondences on the right are more accurate (A_3 and B_3).*

To detect noisy samples, we search for pairwise alignments that are inconsistent with the aggregated contribution of all other paths in the alignment graph. More specifically, given an alignment graph $\mathbf{G}_i$ and resulting fuzzy correspondences f_i, we define a *consistency score* for each aligning edge a as a simple correlation:

$$\text{score}_{f_i}(a_{S_l,S_k,\phi}) := \sum_{p_1 \in P_l} \sum_{p_2 \in P_k} \frac{a(p_1,p_2) f_i(p_1,p_2)}{\sum_{p \in P_l} |f(p_1,p)|}. \tag{10}$$

Since our choice for the alignment parameter ϕ in the initial graph only depended on the quality of the pairwise matching, it is possible that another parameter ϕ that would align models with higher consistency score. We compute score_{f_i} for all precomputed alternative alignments (e.g., 4 ICP initializations for extrinsic matching, or other low-distortion blended maps). For each edge, we pick the parameter ϕ that maximizes the consistency score. If the best score

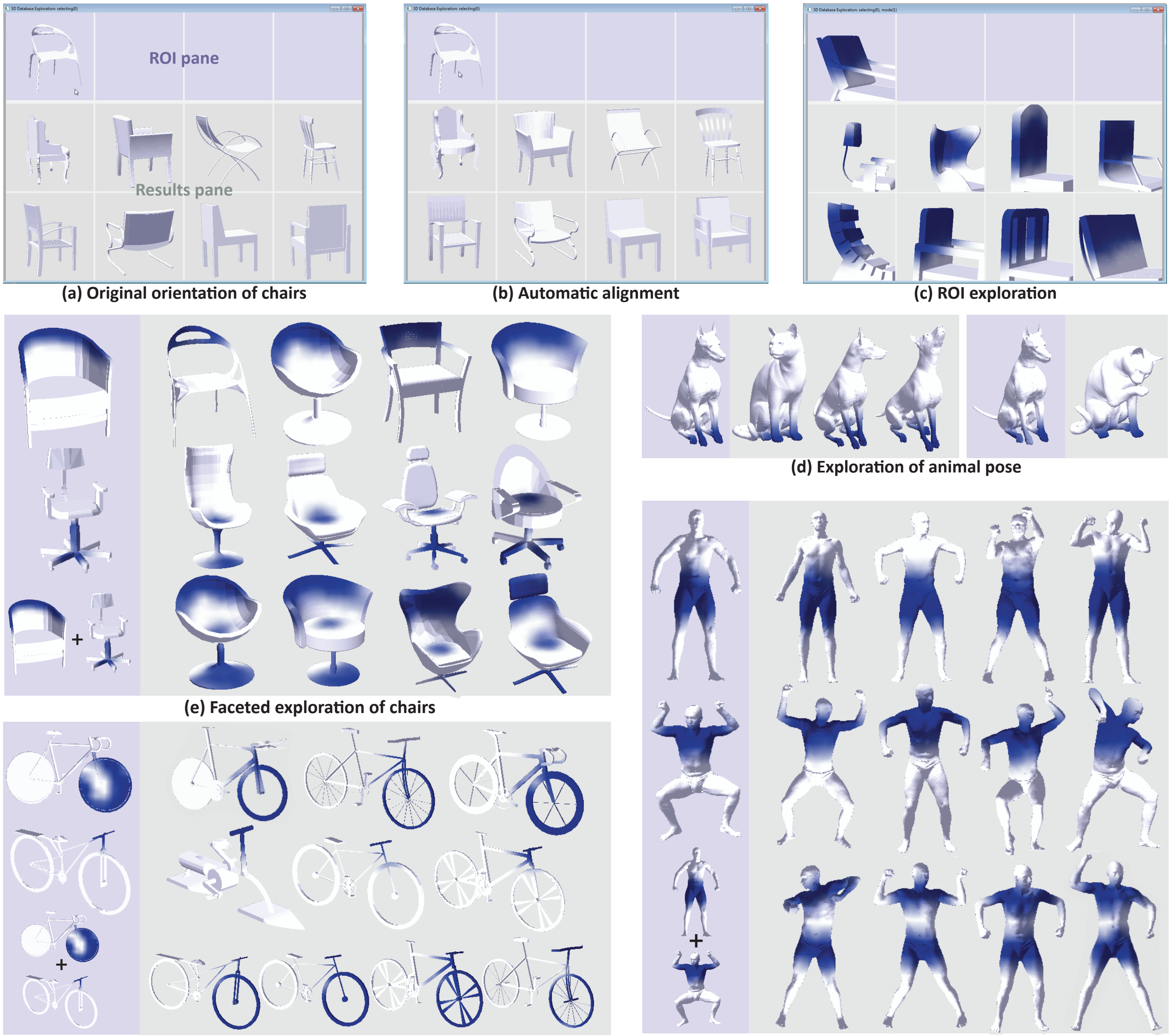
Algorithm 1 Optimize $\mathbf{G}_i$ to create new graph $\mathbf{G}_{i+1}$

```
$\mathbf{G}_{i+1} \leftarrow (\mathbf{G}_i.\mathbf{S}, \emptyset)$

*Improve consistency of all edges*
for $a \in \mathbf{G}_i.\mathbf{A}$ do
    $a_{S_k,S_l}.\phi \leftarrow \arg\max_{\phi' \in \Phi}(\text{score}_{f_i}(a_{\phi'}))$
end for

*Only keep consistent edges*
$\text{NE}_{\text{add}} \leftarrow 0$
for $a_{S_k,S_l} \in \mathbf{G}_i.\mathbf{A}$ do
    $E_{\text{adj.}} \leftarrow \{a_{S_m,S_n} \in \mathbf{G}_i.\mathbf{A} \text{ s.t. } m = k \text{ or } n = l\}$
    $best \leftarrow \max_{a \in E_{\text{adj.}}}(\text{score}_{f_i}(a))$
    if $\text{score}_{f_i}(a) \geq 0.3 \cdot \text{best}$ then
        $\mathbf{G}_{i+1}.\mathbf{A} \leftarrow \mathbf{G}_{i+1}.\mathbf{A} \cup a$
    else
        $\text{NE}_{\text{add}} \leftarrow \text{NE}_{\text{add}} + 1$
    end if
end for

*Sample new alignments*
$candidates \leftarrow a_{S_l,S_k} \text{ s.t. ShortestPath}_{\mathbf{G}_{i+1}}(S_l,S_k) > 3$
$candidates \leftarrow \text{IFCBest}_{f_{i+1}}(candidates, 3 \cdot \text{NE}_{\text{add}})$
$candidates \leftarrow \text{EdgeRankBest}(candidates, \text{NE}_{\text{add}})$
$\mathbf{G}_{i+1}.\mathbf{A} \leftarrow \mathbf{G}_{i+1}.\mathbf{A} \cup candidates$
```



(a) Original orientation of chairs

(b) Automatic alignment

(c) ROI exploration

(d) Exploration of animal pose

(e) Faceted exploration of chairs

(f) Faceted exploration of bikes

(g) Faceted exploration of human poses

***Figure 7:** **Exploration interface and exploration results.** The original collection of chairs (a) is automatically aligned to a canonical viewpoint (b) and then to a selected region of interest (c). While exploring a collection of animals (d), the user queries for a specific arrangement of paws and the system returns all animals in a sitting pose as the most similar results. If the user does not select the right paw, a cat with one paw up appears among the top matching results. To combine exploration criteria, the user selects regions on multiple example shapes. For example, she browses for chairs with high curved backs and stems (e), bikes with large front wheels and straight handlebars (f), and humans with an upright posture and arms extended away from the torso (g).*

for an edge is below 30% of the node's best score (for both nodes) we assume that shapes are too dissimilar and prune the edge.

Finally, we want to include the same number of edges that was pruned (NE$_{\text{add}}$), preferably choosing pairs of models that can be matched more robustly, and such that matching them improves the embedding. Unlike during initialization, at this point we already have an approximation to the function f which we use to guide the sampling. We expect to get more information by aligning pairs that are separated by long paths in the alignment graph $\mathbf{G}$, thus we use a candidate set of all edges such that the shortest path in the pruned graph between two nodes has more than 3 edges. Additionally, we rank all the edges by the integrated fuzzy correspondence value $IFC(S_k, S_l) := \sum_{p_1 \in S_k, p_2 \in S_l} f(p_1, p_2)$ and pick the 3NE$_{\text{add}}$ highest-ranked edges. Finally, we use edge rank to order them and add the top NE$_{\text{add}}$ that improve the graph connectivity (see Figures 5, 6). Algorithm 1 summarizes a single iteration of this optimization.

5 Exploring Model Collections

We leverage the computed fuzzy correspondences to enable guided exploration of the model collection. Specifically, we introduce a browsing interface that allows users to paint regions of interest on any example shape, which then act as navigation criteria to determine how the other models in the collection are organized and presented to the user. Our interface consist of two panes: the ROI pane shows the set of user-selected regions of interest, and the re-

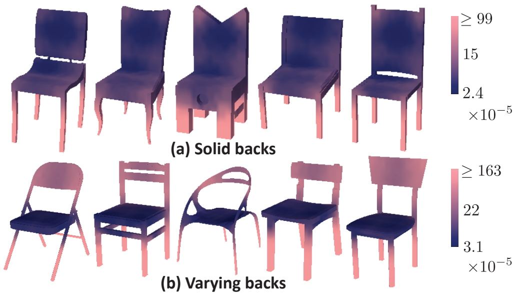


Figure 8: Variance function for two example collections. *The chairs in (a) have more similar backs than the chairs in (b) (dark blue regions indicate less variance). In both collections, seats exhibit less variation than the legs, which have a variety of styles.*

sults pane shows a sorted view of the collection on one or more result pages (see Figure 7a). By selecting various ROIs and examining the results, users can quickly explore many different types of variations within the collection. In order for this type of interface to be effective, it must satisfy a few key design requirements:

Finding where variations occur. Faced with an unfamiliar collection, users may not know *a priori* which regions of a shape to explore. Thus, the browsing interface should convey where interesting variations occur within a collection to help users decide what ROIs to specify.

Visual comparison. To help users understand the similarities and differences across a set of shapes, the interface should facilitate visual comparison of the models. In particular, it should be easy for users to see shapes from a consistent viewpoint and focus on the ROI of each model.

Interactive sorting. Finally, to enable guided exploration using the selected ROIs, the browsing system must be able to interactively sort models based on their similarity to the example shape(s).

The rest of this section describes the main features of our interface with respect to these requirements.

5.1 Finding Variations

To visualize where variations occur across a collection, we measure the amount of variation within different corresponding regions of the shapes. For each sample point on a shape, we compute an average distance to 15% of the nearest neighbors in the embedded space. This average represents an estimate of how much variance there is in a particular region of the shape with respect to the rest of the collection. We normalize these variance values within each model and then visualize this function with a colour map (see Figure 8), where brighter colors indicate more variance. By examining this visualization, users can quickly see what regions of a shape vary the most or the least.

5.2 Visual Comparison

To help users understand the similarities and differences between the models presented in our interface, we provide several features to facilitate visual comparison. First, the system allows the user to align all the models to a consistent viewpoint, as shown in Figures 7a–c. We compute automatic alignments for either the entire shape (see Figure 7b) or just the selected region (see Figure 7c) using fuzzy correspondences as described in the next subsection. Once the models are aligned, the user can interactively orbit, pan, and zoom the viewpoints of all the models in conjunction, which makes it easy to inspect and compare different portions of the shapes. Furthermore, to facilitate orbiting around a selected ROI, our system places the orbit centre for each result model at the centroid of the corresponding ROI points, which we compute as an average of the the point locations weighted by their correspondence values. Finally, to emphasize the corresponding ROIs in the result models, we highlight these regions in blue (see Figures 7c–f). The intensity of the highlight is scaled by the correspondence values at each sample point, interpolated across the surface. To ensure that the highlight is sufficiently visible on each model, we normalize the correspondence values by the maximum correspondence value within the model.

5.3 Interactive Sorting

To sort the collection based on a user-selected ROI, we need a way to measure the geometric similarity between the selected region of an example shape and all other target models in the collection. To this end, we introduce a selection-aware similarity function that uses fuzzy correspondences to determine how to align and compare target models to example shapes.

Given a selected region $R \subset S_i$ on example shape S_i we compute the distance $D_R(S_j)$ to target shape S_j as follows. First, we determine the best rigid alignment $T_R(S_j)$ of the target shape to the selected region. Specifically, for every selected point on the example shape, we find all fuzzy correspondences on the target shape and then compute $T_R(S_j)$ by finding the optimal affine transformation that aligns the corresponding points, where the error is weighted by the correspondence values and minimized in the least-squares sense. To ensure that the alignment remains stable even for small selected regions, we add fuzzy correspondences for *unselected* points to the computation as well, but we multiply their values with a damping factor $\alpha \ll 1$ set to $1/4K$, where K is the number of point samples per model. Next, we find a single correspondence value f_R for each point on the target shape by taking the maximum correspondence value to any point in the selected region $f_R(p) := \max_{p' \in R}(f(p', p)f(p, p'))$. Finally, we use these correspondence values as weights to compute the Euclidean distance between the aligned target shape and the example shape as $D_R(S_j) := \sum_{p \in S_j} D_{\text{Eucl.}}(T_R(p), S_i) f_R(p) / \sum f_R(p)$.

Based on this distance function, our interface supports two sorting modes. The *ordered* mode sorts all target models in the collection based on decreasing similarity to the example shape within the selected ROI. This allows the user to quickly browse the models from most to least similar by scrolling through pages in the results pane (see Figures 7d–f). The *variations* mode populates the results pane with 8 target models that are most dissimilar to each other in the ROI, which gives the user a sense for the range of variations across the collection (see Figure 7c and the supplemental video). Paging the ordered results simply fills the results pane with the next 8 models from the sorted list of shapes, while paging the variations retrieves the next 8 models that are most dissimilar to each other.

If the user paints multiple ROIs on an example shape, the system sorts the remaining models based on their distance to all of the selected ROIs. In particular, for each target model, we compute its distance by summing its distances to each example shape. This allows the user to treat the ROIs as facets that apply multiple sorting criteria to the collection. For example, in Figure 7e, the user specifies multiple facets that return bikes with large front wheels and a straight handlebar, while in Figure 7f, the user applies facets to explore a collection of human poses.

6 Results

In this section, we report evaluation of the proposed methods. Since there are two phases of execution, off-line and on-line, and the results of the first phase (fuzzy correspondences) can be evaluated quantitatively with benchmarks, we first discuss them. Then, later, we describe qualitative results achieved with the interactive system.

6.1 Fuzzy Correspondence Results

To evaluate the quality and speed of our methods for computing fuzzy correspondences, we ran a set of experiments with benchmark data comprising ground truth one-to-one surface correspondences. While our system is not designed for such data (we allow more ambiguous correspondence relationships), the experiments provide means to compare with previous work. Using this benchmark data, we investigate if our algorithms indeed discover a low-dimensional manifold in shape space and leverage it effectively.

Data. We test our method on two benchmarks. The first involves smooth manifold surfaces with similar intrinsic structure proposed by Kim et al. [2011] that contains near-isometric surfaces from three datasets: (i) SCAPE (71 meshes representing a human body in different poses) [Anguelov et al. 2005], (ii) TOSCA (80 meshes representing people and animals in a variety of poses) [Bronstein et al. 2008], and (iii) watertight models from the SHREC dataset [Giorgi et al. 2007]. Each model comes with a set of feature points that correspond across the collection (e.g., 17 points for hands, and all 12,500 vertices for SCAPE).

We specifically created a second benchmark by downloading diverse collections of 111 chairs and 86 commercial airplanes from Google 3D Warehouse. As ground truth, we manually annotated 10 and 6 feature points respectively on each model of the chairs and airplanes datasets (see Figures 10, 12, and supplementary material). In addition, we use a collection of 66 bikes with our exploration tool. Note that these models have large variations in the number of connected components (from 1 to 74,796) and polygons (from 816 to 2,622,379).

Evaluation metric. Semantic correctness of fuzzy correspondence values is hard to quantify since that requires us to prescribe values for semantic similarity for all pairs of corresponding feature points. Instead, we assume that annotated feature points are in perfect correspondence and simply project the fuzzy correspondences to the space of point-to-point maps by choosing the points with the best correspondence value. More specifically, given a pair of shapes S_k, S_l for any point $p_i \in P_k$ we assign the closest point in the embedded space: $corr(p_i) := \arg\min_{p_j \in P_l} D_t(p_i, p_j)$. We use nearest-neighbor interpolation based on such a map of K samples to capture large-scale correspondences for the benchmark.

We measure the quality of a map as proposed in Kim et al. [2011]: we record distance from the predicted correspondence to the true correspondence $d_{S_l}(corr(p_i), f_{\text{true}}(p_i))$, and plot our results as a curve showing the fraction of correspondences mapped correctly within a threshold $d_{S_l} < D$, where different thresholds D are depicted on the x-axis. We use geodesic distances for the benchmark of intrinsically similar shapes and Euclidean distances for a diverse set of shapes. We now report the findings of our experiments based on these evaluation methods.

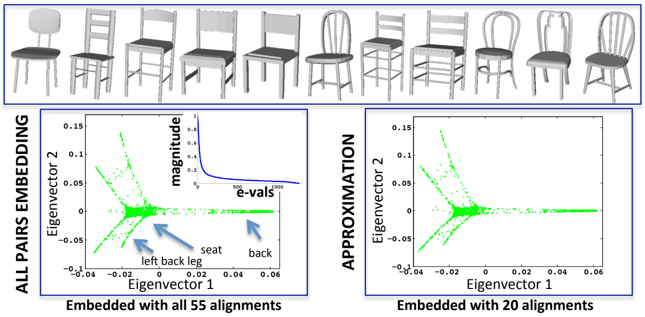


***Figure 9:** **Top two non-constant eigenvectors of** C. This figure demonstrates eigenvalues and the embedding of a collection of 11 chairs. A green dot in the embedded space corresponds to one of* 128×11 *points in the data. The correspondence matrix is sampled by aligning all 55 pairs, and arbitrary 20 pairs of shapes.*

Correspondence space is low-dimensional. First, we investigate if correspondences among diverse models can indeed be effectively embedded in a low-dimensional space. For this test, we pick a relatively uniform set of 11 chairs (Figure 9) and use our method to compute the correspondence matrix C (note that all pairs are matched with default parameters and all alignments are good). Although the correspondence matrix is very high-dimensional $(11 \cdot 128 \times 11 \cdot 128)$, note the spectral gap in the distribution of the eigenvalues. On the left we further map each of the $11 \cdot 128$ points on all shapes to a 2D plane using just the top two non-constant eigenvectors. The structure of that manifold resembles the shape of a generic chair, and corresponding points are close to one another in the embedded space, which agrees with our hypothesis that the correspondence space is low-dimensional $O(K)$.

A small subset of pairwise alignments suffices. Next, we test how sparsity of the alignment graph **G** affects the fuzzy correspondences. In Figure 9-right we demonstrate embedding due to an alignment graph with 20 pairs of models. Note that the embedding is robust to a sparser sampling. In another experiment (see Figure 10), we only use correct affine transformations that best align the prescribed feature correspondences in the least squares sense (note that this is the only experiment where we use the ground truth to align models). The dashed curve (manual affine) shows the accuracy obtained just by the ground truth alignment. We further compute fuzzy correspondences from the alignment graph created with 150, 250, 350, and 500 edges. The results indicate that even with 500 alignments we already reach the accuracy of using ground truth affine alignment for all 6105 pairs.

Fuzzy correspondences improve pairwise point-to-point maps. We also compare our method to work on optimizing collections of maps by Nguyen et al. [2011] who also optimize blended intrinsic maps for consistency. Our evaluation is based on four classes in the SHREC dataset: animals, humans, teddy bears, and hands (we only use the collections of maps produced by the authors). For animals and humans datasets we successfully resolve *all* flips caused by symmetry confusion except for two higher genus human models where all blended maps fail (also Nguyen's failure case). Similar to Nguyen et al. [2011], our method is confused by the symmetry in teddy bears, and we also consistently misalign some of the models. See supplemental materials for these datasets.

The hand dataset shows the real benefit (see Figure 11). Note that this dataset is different from the other three in the main source of error for the pairwise matching: instead of globally inconsistent alignments due to symmetry, it usually has local inconsistencies in mapping fingers. Note that in such cases their method is limited to finding a concatenation of full model-to-model maps that eventually aligns fingers correctly — this may be impossible if none of the maps are perfect. On the contrary, we optimize per point, and thus aggregate several maps to produce a consistent alignment. Thus we can correctly align *all* the hands, as opposed to their method (see hands 11 and 17 in Figure 11 and these hands are consistently misaligned to *all* the other 18 hands in their results).

In Figure 12, we do a pairwise matching using just the best affine transformation on a commercial dataset. Similarly to blended

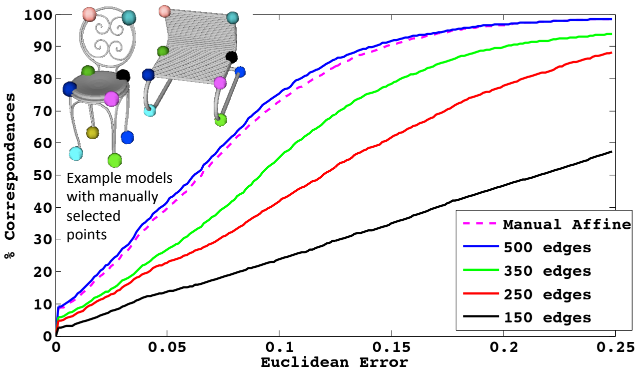


Figure 10: Chairs with manual alignments*. We compare our method to naively aligning all pairs (dashed line) using ground truth pairwise alignments. Using fuzzy correspondences, even with 500 alignments, we get comparable results to those obtained by matching all the 6105 pairs.*

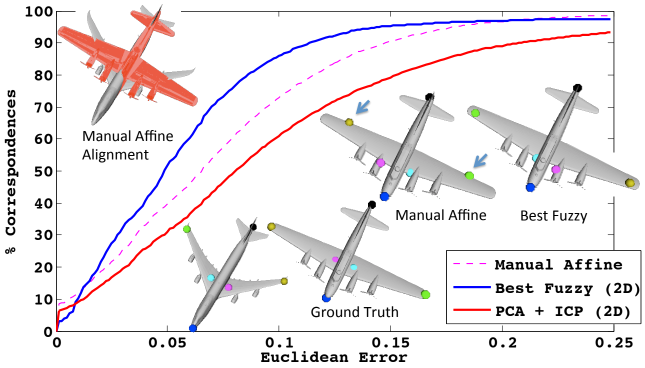


Figure 12: Airplanes dataset*. We compare taking best fuzzy correspondence to geometric matching of all pairs. Note that fuzzy correspondences also do better than the best manual pairwise alignment because diffusion expands the allowable aligning deformations beyond affine transformations by using indirect alignments.*

intrinsic maps, we improve over pairwise matching due to consistency optimization (avoiding matching the front of an airplane to the tail). Interestingly, for some error intervals, fuzzy correspondences outperform the affine alignment with the ground-truth points. The reason for this is that wings and the body generally have very different proportions and position along the body of the airplane, thus even with ground truth alignment, the semantic points might not align (i.e., the deformations within the class of airplanes are beyond our pairwise alignment method); however, diffusion allows these correspondences to propagate closer to semantically correct values. That is, it extends the allowable deformations beyond the capability of our simple pairwise matching algorithm.

Please refer to the supplemental material for evaluation results on other intrinsically similar collections (51 animals, 71 SCAPE humans), as well as extrinsic matching of 111 chairs — fuzzy correspondences improve the accuracy for *all* these datasets.

More data improves correspondences. In our final experiment, we investigate whether leveraging the "power of the set" aids computation of fuzzy correspondences between individual pairs. Figure 13 shows the accuracy curves as the number of models in the database changes. We selected a subset of 5 chairs from the full collection aiming at higher variation within the subset, and show the error only for the selected 5 models; similarly for a dataset with 20 additional random models from the chair dataset, and all the 111 models. These results demonstrate that increasing the size of the database improves the correspondences due to a denser sampling of shape variations within a class.

Timing. We executed all off-line computations of fuzzy correspondences on SunFire X4100 computer with an AMD Opteron 275 Dual-Core 2.2GHz processor. For the largest dataset of 111 chairs the whole analysis (including pairwise matching) requires about 900s, of which about 300s are spent on spectral analysis of the correspondence matrix, 500s on graph optimization, and the rest on pairwise matching. Our algorithm used only 602 out of 6105 pairs for chairs dataset, 335 out of 2145 bike pairs, and 430 out of 3655 airplane pairs. Although, the optimization timings are comparable to the SCAPE and animals datasets, the intrinsic pairwise matching takes up to 5 minutes per pair for SCAPE (we match 355 of 2485 pairs), and 10 to 30 minutes per pair for animals (we match 632 of 1275 pairs). We compute these maps in parallel on a cluster. In all examples our method converged within 8 iterations.

On the hands dataset (Figure 11) fuzzy correspondences were computed with 100 intrinsic maps and the optimization process converged within 70s. Nguyen et al. [2011] used all 380 pairwise alignments, and the optimization took 140s. Finding the intrinsic maps takes about 5-10 minutes. Our exploration interface runs interactively on a laptop with 2.4GHz Intel Core 2 Duo processor.

Parameters. We use the same parameters for all the datasets. We choose $K = 128$ samples per model based on the desired resolution of correspondences: a larger K offers higher precision in fuzzy cor-

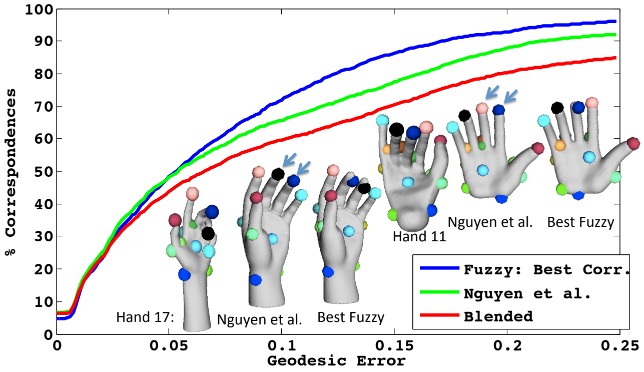


Figure 11: SHREC hands dataset*. Comparison of three methods: taking the best fuzzy correspondence (blue), method of Nguyen et al. (green) and just using blended intrinsic map (red). Note that hands 11 and 17 are consistently misaligned by the method of Nguyen et al. to all the other 18 models in the database.*

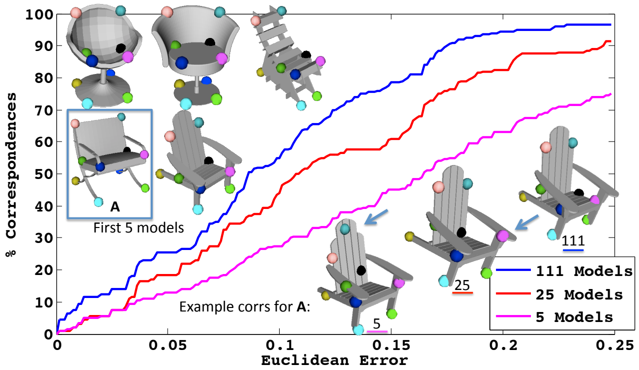


Figure 13: Chairs: 5, 25, 111 models*. We analyze the error for 5 selected models present in all datasets and add more chair models for joint analysis. Note that increasing the size of the database improves the quality of the correspondences, as shown in the example correspondences at the bottom right.*

respondences at the cost on increased compute times.

We chose $t = 10$ based on the desired trade-off between robustness and fuzziness of correspondences: higher diffusion times reduce noise due to misalignment errors and sparse connectivity in the alignment graph, while lower diffusion times provide higher discriminative power of correspondence values. In the limit, as t goes to infinity, f approaches a uniform distribution for every point.

Limitations. In the off-line step, we found that fuzzy correspondences are vulnerable to the presence of a strong sampling bias (e.g., due to near-symmetry). This issue usually manifests in two ways: (i) A model might be aligned incorrectly (but consistently) across the collection (e.g., a nose of an airplane is mapped to tails of all other airplanes in a collection). Among all examples datasets, we found that 1 of 86 airplanes, and 5 of 66 bikes were consistently misalgned. (ii) If a bias is too common for a collection, diffusion might propagate information along incorrect paths causing undesirable blending of correspondences (see Figure 4).

The largest collection we analyzed with our current implementation has 111 models. Scaling our method to thousands of models would require parallel implementation of some steps of the algorithm like eigenvalue decomposition and iterative graph optimization. We are also restricted to collections such that for every pair of dissimilar models there is a continuous path of pairwise-similar models. While datasets we analyzed have various types of noise, multiple disconnected components, intersecting surfaces, large holes, micro holes, etc., our method is unlikely to be suitable for processing raw scans.

6.2 Exploration Results

We use our interactive tool to explore and visualize variations within several example collections (see Figure 7 and supplementary video): chairs, bikes, commercial airplanes, animals, and SCAPE humans. The ability to browse the collection based on specific regions of interest enables us to uncover interesting characteristics within all of these datasets. For example, by selecting the curved back and arms of a chair (Figure 7e) we discover that several other chairs have a similar arrangement of these features. Selecting the straight handlebars of a bike (Figure 7f) reveals a variety of shapes that share this property, including the "bike" with no wheels that is ranked as the most similar result. In addition to emphasizing similarities in the collection, our system also reveals diversity. For example, by sorting the tops of chair backs in variations mode, we immediately see a wide range of results that include boxy, organic, upright and tilted shapes (see Figure 7c).

Note that the ability for users to select arbitrary regions rather than just predefined parts represents a significant advantage for our exploration system. For diverse collections such as the chairs, selecting portions or combinations of standard "parts" often yields informative and discriminative navigation criteria, such as the seat/back region in Figure 1. Furthermore, for collections where our analysis considers intrinsic geometry, such as the animals and humans, selecting regions that span several limbs or joints is an intuitive way to guide exploration based on pose. For instance, Figure 7g shows how selecting the mid-section of a human can restrict navigation to upright poses, while a selection across the shoulders and chest returns humans with their arms extended away from their sides. As another example, selecting all four paws of the dog in Figure 7d retrieves other sitting animals, but if we do not select one of the front paws, we also discover the sitting cat with one paw in the air.

Finally, the ability to combine exploration criteria into facets provides additional flexibility and control during navigation (see Figures 7e–g). Faceted browsing not only allows the user to narrow the exploration space (e.g., only show chairs with both a stem and high curved back) it also gives the user a sense for what types of variations are independent of each other within the collection. For example, the sequence of facets in Figure 7e shows that the shape of chair backs and the type of base vary independently, and the queries in Figure 7g reveal a similar independence between the different aspects of human poses in the SCAPE dataset. These types of insights are especially valuable for understanding the variations across diverse shape collections.

Limitations. In some cases our exploration tool might return unintuitive results due to limitations of the interactive sorting algorithm, which does not capture variations in some interesting geometric features (see Figure 14). Developing tunable and more discriminative geometric descriptors that employ fuzzy correspondences is an interesting topic for future work.

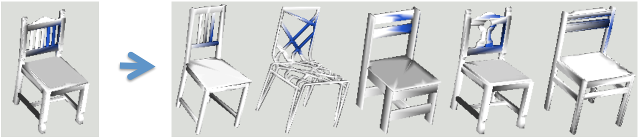


*Figure 14: **Sorting limitation**. The similarity metric we use for the interactive sorting is too simple to capture some aspects of geometry like vertical bars on chair's back.*

7 Conclusion

We make two main contributions: (i) fuzzy correspondences, a new computational tool to find and encode semantic relationships between points in a large, diverse collection of 3D models, and (ii) a novel browsing interface for exploring relationships and variations in collections of models based on interactive selection of regions of interests. We evaluate our computational framework on two benchmarks, showing that it is efficient, i.e., generates all correspondences by matching only a subset of pairs of models, and accurate, i.e., improves semantic alignment of points in comparison to existing mapping methods.

Future work. This paper considers only initial ideas on potential applications of fuzzy correspondences. Future efforts may include a more rigorous study of applicability of our exploration tool and fuzzy correspondences in other use scenarios like searching for parts for modeling-by-example applications [Funkhouser et al. 2004], learning other types of variations (e.g., texture, high-frequency geometry properties), and/or for transfering properties between models. For example, Figure 15 shows initial results for transferring functional labelings between models. In this example, the contact region on one human and one chair is marked interactively by a user. Then, fuzzy correspondences are used to find a best affine alignment between all chairs in the database, which can be used to place the human model in an appropriate position in *all* other chairs in the database just from a single example alignment

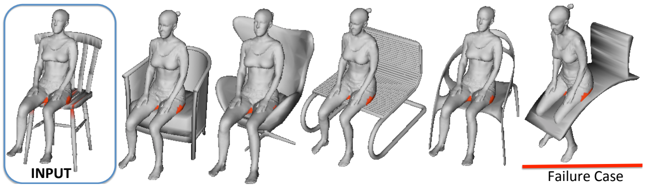


*Figure 15: **Functional labeling.** The user provides an example alignment for two selected regions (a human on a chair), and our system automatically aligns human to all other chairs using fuzzy correspondences.*

(see supplementary material). Further investigation of applications of this type seems like a fruitful direction for future work.

Acknowledgements. We thank Marc Alexa, Yaron Lipman, Amit Singer, and the anonymous reviewers for their comments and suggestions. The project was partially supported by NSERC, NSF (CNS-0831374, CCF- 0702672, CCF-0937139, and CNS-0831374), AFOSR (1096101), Intel (ISTC-VC), Google, and Marie Curie Career Integration Grant 303541.

References

AIGER, D., MITRA, N. J., AND COHEN-OR, D. 2008. 4-points congruent sets for robust surface registration. *ACM TOG 27*, 3, #85, 1–10.

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2003. The space of human body shapes: reconstruction and parameterization from range scans. In *ACM SIGGRAPH*, 587–594.

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005. Scape: shape completion and animation of people. In *ACM SIGGRAPH*, 408–416.

BESL, P. J., AND MCKAY, N. D. 1992. A method for registration of 3-d shapes. *IEEE PAMI 14*, 2 (Feb.), 239–256.

BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2006. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. *PNAS 103*, 5, 1168–1172.

BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2008. *Numerical geometry of non-rigid shapes*. Springer.

CHUI, H., AND RANGARAJAN, A. 2003. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst. 89*, 2-3, 114–141.

ELDAR, Y., LINDENBAUM, M., PORAT, M., AND ZEEVI, Y. 1997. The farthest point strategy for progressive image sampling. *IJCV 40*, 2, 99–121.

FISHER, M., SAVVA, M., AND HANRAHAN, P. 2011. Characterizing structural relationships in scenes using graph kernels. *ACM SIGGRAPH 30*, 34:1–34:12.

FUNKHOUSER, T., KAZHDAN, M., SHILANE, P., MIN, P., KIEFER, W., TAL, A., RUSINKIEWICZ, S., AND DOBKIN, D. 2004. Modeling by example. *ACM SIGGRAPH*, 652–663.

GIORGI, D., BIASOTTI, S., AND PARABOSCHI, L. 2007. Shrec:shape retrieval contest: Watertight models track. *http://watertight.ge.imati.cnr.it/*.

GIORGI, D., FROSINI, P., SPAGNUOLO, M., AND FALCIDIENO, B. 2010. 3D relevance feedback via multilevel relevance judgements. *Vis. Comput. 26*, 10, 1321–1338.

GOLOVINSKIY, A., AND FUNKHOUSER, T. 2009. Consistent segmentation of 3D models. *Proc. SMI 33*, 3, 262–269.

HEARST, M. A. 2006. Clustering versus faceted categories for information exploration. *Commun. ACM 49*, 4 (Apr.), 59–61.

HEATH, K., GELFAND, N., OVSJANIKOV, M., AANJANEYA, M., AND GUIBAS, L. J. 2010. Image webs: Computing and exploiting connectivity in image collections. In *IEEE CVPR*.

HUANG, Q., KOLTUN, V., AND GUIBAS, L. 2011. Joint shape segmentation with linear programming. In *ACM SIGGRAPH Asia*, 125:1–125:12.

KALOGERAKIS, E., HERTZMANN, A., AND SINGH, K. 2010. Learning 3D mesh segmentation and labeling. In *ACM SIGGRAPH*, 102:1–102:12.

KAZHDAN, M., FUNKHOUSER, T., AND RUSINKIEWICZ, S. 2003. Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proc. SGP*, 156–164.

KIM, V. G., LIPMAN, Y., AND FUNKHOUSER, T. 2011. Blended intrinsic maps. In *ACM SIGGRAPH*, 79:1–79:12.

KRAEVOY, V., AND SHEFFER, A. 2004. Cross-parameterization and compatible remeshing of 3D models. In *ACM SIGGRAPH*, 861–869.

LI, H., SUMNER, R. W., AND PAULY, M. 2008. Global correspondence optimization for non-rigid registration of depth scans. In *Proc. SGP*, 1421–1430.

LIPMAN, Y., AND FUNKHOUSER, T. 2009. Mobius voting for surface correspondence. In *ACM SIGGRAPH*, 72:1–72:12.

LIPMAN, Y., CHEN, X., DAUBECHIES, I., AND FUNKHOUSER, T. 2010. Symmetry factored embedding and distance. In *ACM SIGGRAPH*, 103:1–103:12.

MITRA, N. J., GELFAND, N., POTTMANN, H., AND GUIBAS, L. 2004. Registration of point cloud data from a geometric optimization perspective. In *SGP*, 22–31.

MUJA, M., AND LOWE, D. G. 2009. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proc. VISSAPP*, 331–340.

NADLER, B., LAFON, S., COIFMAN, R. R., AND KEVREKIDIS, I. G. 2006. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis 21*, 1, 113 – 127.

NGUYEN, A., BEN-CHEN, M., WELNICKA, K., YE, Y., AND GUIBAS, L. 2011. An optimization approach to improving collections of shape maps. *SGP 30*, 5, 1481–1491.

OVSJANIKOV, M., MÉRIGOT, Q., MÉMOLI, F., AND GUIBAS, L. J. 2010. One point isometric matching with the heat kernel. *SGP 29*, 5, 1555–1564.

OVSJANIKOV, M., LI, W., GUIBAS, L., AND MITRA, N. J. 2011. Exploration of continuous variability in collections of 3D shapes. *ACM SIGGRAPH 30*, 4, 33:1–33:10.

SCHREINER, J., ASIRVATHAM, A., PRAUN, E., AND HOPPE, H. 2004. Inter-surface mapping. *ACM SIGGRAPH 23*, 3, 870–877.

SHILANE, P., MIN, P., KAZHDAN, M., AND FUNKHOUSER, T. 2004. The princeton shape benchmark. In *Proc. SMI*, 167–178.

SIDI, O., VAN KAICK, O., KLEIMAN, Y., ZHANG, H., AND COHEN-OR, D. 2011. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM SIGGRAPH Asia 30*, 6, 126:1–126:9.

SUN, J., CHEN, X., AND FUNKHOUSER, T. A. 2010. Fuzzy geodesics and consistent sparse correspondences for: eformable shapes. *CGF 29*, 5, 1535–1544.

VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2011. A survey on shape correspondence. *CGF 30*, 6, 1681–1707.

XU, K., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2012. Fit and diverse: Set evolution for inspiring 3D shape galleries. *ACM Trans. on Graph (Proc. of SIGGRAPH) 31*, to appear.