Deformation-Driven Shape Correspondence

H. Zhang[1], A. Sheffer[2], D. Cohen-Or[3], Q. Zhou[2], O. van Kaick[1], and A. Tagliasacchi[1]

[1]Simon Fraser University, [2]University of British Columbia, [3]Tel Aviv University

Abstract

Non-rigid 3D shape correspondence is a fundamental and difficult problem. Most applications which require a correspondence rely on manually selected markers. Without user assistance, the performances of existing automatic correspondence methods depend strongly on a good initial shape alignment or shape prior, and they generally do not tolerate large shape variations. We present an automatic feature correspondence algorithm capable of handling large, non-rigid shape variations, as well as partial matching. This is made possible by leveraging the power of state-of-the-art mesh deformation techniques and relying on a combinatorial tree traversal for correspondence search. The search is deformation-driven, prioritized by a self-distortion energy measured on meshes deformed according to a given correspondence. We demonstrate the ability of our approach to naturally match shapes which differ in pose, local scale, part decomposition, and geometric detail through numerous examples.

1. Introduction

Establishing a meaningful shape correspondence is a fundamental task in geometry processing. In applications such as object recognition, statistical shape modeling, shape morphing, and deformation transfer, shape correspondence is often the first step. However, the problem is essentially ill-posed as one generally needs to understand the shape semantics to truly infer what a meaningful correspondence is.

In this paper, we develop an automatic, feature-based correspondence algorithm for shapes represented by triangle meshes. It is aimed at handling large, non-rigid shape variations, as well as partial matching. Removing the rigidity constraint complicates the problem significantly; see Figure 1. Recognizing that the models are both dinosaurs, a human can easily find a correspondence. However, the large variations in the models' pose, local scale, and geometric detail, e.g., those around the hands, feet, and belly areas, would challenge an automatic algorithm. There is no longer a low-dimensional pose space, as in the rigid case. Also, a higher tolerance for local feature dissimilarity and shape distortion must be allowed, which not only enlarges the solution search space but also increases the likelihood of a false match.

Several well-known problems in graphics which require feature correspondence, e.g., cross-parameterization, have mostly relied on user input. When dealing with a large model set, such user assistance may become tedious and impracti-

Figure 1: *The dino-skeleton is deformed to match the raptor (red markers indicate features). Top two candidate correspondences are shown. Switching between symmetric parts, highlighted in circle, is detected by the distortion cost.*

cal. The majority of existing approaches for automatic correspondence either adopt greedy local search, making them sensitive to initial alignment and prone to local minima, or handle only rigid or affine transforms. Non-rigid schemes have been proposed in the context of statistical shape modeling. They often benefit from given shape priors, e.g., operating only on human faces, bodies, brain surfaces, corpus callosi shapes, or other medical data. Many works in computer vision and medical image analysis are in this category, where the considered shapes do not significantly differ.

To deal with large shape variations, we rely on non-rigid mesh deformation to drive a combinatorial search for the best correspondence. Our notion of a good feature correspondence requires that each corresponding feature pair is

sufficiently similar — a local criterion — and that deforming the shapes into each other while respecting the feature pairing incurs a small *self-distortion* cost — a global criterion. It is important to note that the cost we use is only measured on the deformed mesh and not between two meshes. This not only gives us efficiency but also facilitates partial matching.

We further stipulate that effective shape correspondence should be carried out in a coarse-to-fine manner. A small number of features, perhaps up to 10-12, are sufficient to represent the prominent parts of most objects. A correspondence between them can significantly narrow down the search space for a more refined mapping, e.g., a dense cross-parameterization, which is a more local problem. The main challenge is to find the initial correspondence automatically; this typically involves an expensive global search.

Our algorithm returns a coarse correspondence by operating on a small number of automatically selected, well-spread, and prominent features on two meshes. The local criterion serves as constraints for a search which is is *deformation-driven*, i.e., it is completely prioritized by a distortion energy associated with mesh deformation. Figure 1 highlights the utility of the deformation-based approach. Since the two dinosaurs differ rather significantly in their geometric detail, local feature-to-feature similarities become less reliable in correspondence search. Without any thresholding by feature similarity, the deformation-driven search still correctly finds the most natural correspondence. It also properly differentiates symmetry switchings in a shape, such as the crossed legs and arms shown in Figure 1; these are generally difficult to detect by any approach that is completely intrinsic.

The main contributions of our work are:

- A deformation-driven approach to correspondence which can handle shapes that vary in pose, local scale, part decomposition, and geometric detail effectively.

  - Our distortion cost is intrinsic to the shape and suitable to use for non-rigid correspondence.

  - Measuring the distortion on deformed meshes allows us to detect switching between symmetric parts.

  - The deformation-based criterion for correspondence is global and can tolerate local shape variations.

  - Use of self-distortion cost facilitates partial matching since spurious parts do not constrain cost estimates.

- Correspondence search via combinatorial tree traversal to enable partial matching and exploration of a large solution space so as to avoid reliance on initial shape alignment and erroneous local minima reached by greedy search.

With a few initial parameters set, our algorithm runs fully automatic feature selection and correspondence. We show that various models can be matched correctly with fixed parameters and demonstrate the use of the computed matches as initial correspondences for dense cross-parameterization.

2. Related work

A good example where feature correspondence is needed is cross-parameterization, where a continuous mapping between two surfaces is sought. Existing methods, e.g., [SAPH04, KS04], require a sparse initial correspondence given by the user. Although such a correspondence may be removed eventually [SAPH04], the guidance it provides is indispensable as it immediately brings the starting point of the solution search near the global optimum to facilitate gradient descent. In related applications such as shape morphing [COSL98, GSL*98, ACOL00], statistical shape modeling [ACP03, ASK*05], and deformation transfer [SP04, ZRKS05], user assistance is similarly required. In practice, it is desirable to minimize user interaction as manually specifying markers is an iterative process, often including a corrective loop that can be tedious [ZRKS05]. User involvement is especially burdensome when dealing with a large set of models, i.e., during preparation of training data.

Existing methods for automatic feature correspondence work on salient features extracted based on some local shape signature [GCO06, GMGP05, LG05]. A proper feature-to-feature similarity provides a means for local feature fitting, followed by a more expensive global search for the top correspondences. Our review focuses on correspondence search.

Correspondence search: Branch-and-bound-based combinatorial search [GMGP05], graph optimization [HH03], voting in pose space [GCO06, LG05], and forward search [HFG*06], have been proposed for rigid registration. These schemes all benefit from the rigidity constraint, as it implies a low-dimensional pose space, more reliable feature picking, and more stringent global consistency thresholds which lead to more aggressive pruning in branch-and-bound. Iterated closest point (ICP) [BM92] is applicable in both the rigid and non-rigid [CR03] settings, but as a greedy approach ICP is sensitive to initial alignment.

Matching of articulated shapes is naturally supported by skeletal [SSGD03] or Reeb graph [TS04, BMSF06] representations, whose nodes describe geometric features over a shape by a signature. Also, spectral embeddings [EK03] based on intrinsic shape information such as geodesics can provide pose normalization. However, these structural descriptions are expensive to compute accurately. They are also sensitive to topological changes while being oblivious to extrinsic information which should be accounted for in correspondence, e.g., a switching between the symmetric parts of an animal. User assistance is typically required to break such symmetries [ASK*05]. Our deformation-based approach relies on intrinsic representations to factor out rigid transforms. It also addresses the above issues with the structural descriptions as our distortion cost is measured on deformed meshes, which accounts for extrinsic geometric quantities.

Partial matching: Partial matching allows shapes to correspond based only on their sub-parts or sub-regions, offering

a powerful and versatile option for object recognition and retrieval [GCO06, FS06]. Techniques have been proposed to relax the combinatorial search for partial matchings into a continuous one, so that constrained non-linear optimization can be applied. Relaxation labeling [RHZ76] and graduated assignments [GR96] fall into this category. Other methods for partial matching include geometric hashing [GCO06] and combinatorial searches which rely on upper-bounding k, the number of unmatched features, e.g., [SH81], fixing k in advance, e.g., [FS06], or heuristically searching for k. For example, Gelfand et al. [GMGP05] run their branch-and-bound search of global registrations for each k and choose one which causes a sharp increase in the minimum registration error returned. The above approaches have all been applied to rigid or affine shape registration.

Deformation-based approaches: Early work of Sederberg and Greenwood [SG92] defines a good 2D shape blending as one which minimizes "work," combining bending and stretching. Finding an optimal 2D contour correspondence is greatly simplified by the order preservation constraint (along the contours). Blanz and Vetter [BV99] develop morphable models for 3D faces and rely on planar parameterizations to reduce the problem to image registration. Sheldon [She00] extends morphable models to surfaces, but the model-fitting term of the deformation energy is based on Euclidean closest points and the energy is minimized via gradient decent. Also, all the test models are already quite similar and roughly aligned. Much the same can be said about other deformation-based works in computer vision and medical imaging, e.g., [VG05], where the corresponded shapes, e.g., corpus callosum data or human faces, do not possess the kind of large shape variabilities as the type of models we consider.

3. Overview

The uncanny ability of humans to match complex shapes can be attributed to the initial shape recognition process, which heavily draws upon prior knowledge [Heb49]. One can hardly expect a machine to perform recognition adequately, which makes automatic correspondence difficult.

3.1. Difficulty of non-rigid shape correspondence

In the case of rigid or affine shape registration, the space of allowable transformations is low-dimensional and can be characterized analytically. One can then define an optimization criterion accordingly. Although loss of precision can occur in the presence of noise or sampling artifacts, such errors are relatively predictable and easy to bound. Non-rigid correspondence, on the other hand, is an ill-posed problem as we do not have a definition of what a meaningful matching is. The amount of shape variations that should be tolerated is model-dependent and can take on arbitrary values.

Consider the two features on the belly of the dino-skeleton and the raptor, as shown in Figure 1. Geometrically, their

local neighborhoods are not similar at all, yet semantically speaking, they should correspond. Similar claims can also be made for the fingers or mouths of the two models. Resorting to scale-space theory [Lin94], one may detect a higher degree of similarity between these features after sufficient smoothing. However, too much smoothing hinders our ability to differentiate. Indeed, if our non-rigid correspondence algorithm is to tolerate rather large shape variations, finding a high-quality local shape signature becomes quite difficult, as there is always a conflict between shape discrimination and tolerance of shape variations.

Consequently, our approach is to rely more on global criteria, as local shape similarity may become less reliable. Our deformation-driven algorithm offers a number of advantages for non-rigid shape correspondence (see Section 1).

3.2. Overview of deformation-driven correspondence

Due to correlation between object recognition and shape correspondence, as well as the general consensus that our visual object recognition is likely guided by the the objects' part structures [Heb49], we approach the non-rigid correspondence problem by considering a coarse set of features residing on the prominent parts of a shape. The features are detected shape extremities, as described in Section 4.

We define a correspondence or (partial) matching between the features extracted from two meshes as a set of *one-to-one* feature pairs. The correspondence search is carried out by a combinatorial tree traversal, which we cover in Section 5. The search is prioritized by a symmetric self-distortion energy (Section 5.1), serving as the correspondence cost. The distortion energies are evaluated on deformed meshes with deformation anchored by the given correspondence. For this task, we adopt the linear, rotation-invariant differential mesh deformation scheme of Lipman et al. [LSLCO05].

Figure 2 shows an example of our correspondence cost at work. The correct matching (b), between 9 features on the wolves, has the lowest cost. Leg or ear switches lead to higher distortion on the deformed meshes, which is captured by our cost. Note that the same does not hold for the purely intrinsic geodesic distortion measure β, which accounts for the total differences between pairwise geodesic distances (see Section 5.2) measured on the *original* meshes. We reiterate that a key point of our deformation-driven approach is to measure distortion on the *deformed* meshes.

To search through the partial matchings between features, we organize them into a search tree. Each path from the root of the tree encodes a candidate solution to the correspondence problem, where each node along the path adds a new feature pair. We offer the option of early termination when a desired set of solutions has been obtained. To return a single best solution automatically, we again rely on a distortion cost driven approach to choose a matching size (Section 5.3). To narrow the search space, we employ several
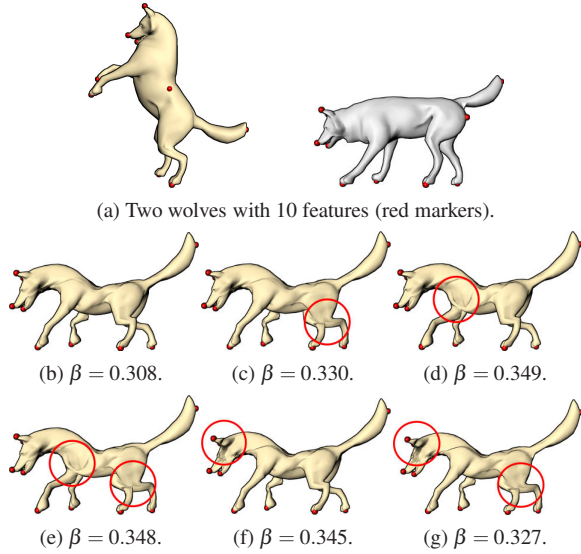
(a) Two wolves with 10 features (red markers).

(b) $\beta = 0.308$.   (c) $\beta = 0.330$.   (d) $\beta = 0.349$.

(e) $\beta = 0.348$.   (f) $\beta = 0.345$.   (g) $\beta = 0.327$.

Figure 2: *Top six matchings sorted by distortion cost* η*: (b) - (g). The left wolf in (a) is deformed into the other, based on 9 matched features. Deformation is anchored by extrinsic geometry and properly sorts out switches (highlighted in circle) between the legs or ears. Total geodesic distortion* β*, fails to accomplish that; observe (g) vs. (c) and (e) vs. (d).*

pruning techniques, e.g., those based on pairwise geodesic distances and feature-to-feature similarity defined by curvature maps [GGGZ05]; this is covered in Section 5.2.

4. Automatic feature selection for correspondence

Geometric features on a shape can be defined in different ways, e.g., ridge and valley lines [OBS04], prominent tip points [ZMT05], or points with most unusual signatures [GMGP05]. Line-type features are generally not stable under shape articulation. The most prominent features of a model part are arguably near its extremity. Extremities are stable under bending and stretching, making them suitable to use for feature correspondence. Using shape extremities as features also reinforces the correlation between correspondence analysis and object recognition by parts. Extremity features have been utilized for mesh parameterization [ZMT05] and segmentation [KLT05].

We rely on critical point detection over the *average squared geodesic distance field* [HSKK01, ZMT05] defined over a mesh to extract extremities. On each mesh, we find a small number of uniformly distributed samples $s_1, \ldots, s_k$ via geodesic farthest point sampling. The geodesic distance field $\mathscr{G}(v) = \frac{1}{k} \sum_{i=1}^{k} g(v, s_i)^2$ approximates the average squared geodesic distances from vertex v to all other vertices. The local maxima of $\mathscr{G}$ correspond to shape extremities (either convex or concave) while the local minima lie near the center of the shape. We extract the latter as well, since they tend
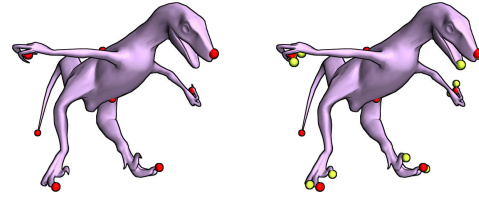

Figure 3: *Features extracted from the T-rex. Left: a coarse set of 8 features, where* $\sigma = 0.05$ *and* $\gamma = 0.2$*. Right: with* $\gamma = 0.1$*, additional features (shown in yellow) are selected.*

to be stable representatives of the central part of an object and make the features well-spread.

To keep the feature set small, we smooth $\mathscr{G}$ via a polynomial, $\frac{2t^3}{\sigma^3} - \frac{3t^2}{\sigma^2} + 1$, having kernel size σ, where t is geodesic distance to the center of the smoothing kernel, and apply geodesic Poisson disk sampling with radius γ to the critical points returned. The σ and γ are constrained to $[0,1]$ and multiplied by the largest pairwise geodesic distance over the mesh. Whenever two critical points are found in the same Poisson disk, we throw away the one that is less prominent, i.e., one having a smaller $\mathscr{G}$ value. The remaining critical points are sorted by $\mathscr{G}$ and selected from atop the sorted list.

We observe that the most prominent extremities can always be reliably selected. While different choices γ, σ, and feature count can induce spurious features, the ability of our algorithm to perform partial matching ensures that such features would be excluded in the result. Figure 3 shows features automatically extracted from a T-rex. With γ to 0.1, we obtain a more refined set of features. For coarse feature correspondence however, we do not attempt to match up the fine-detail features, e.g., all the individual fingers of the T-rex and the raptor (see Figure 1). Instead, our correspondence search only considers results with a small number of feature pairs.

5. Correspondence search

Let M and M' be two input meshes with extracted features. We wish to search for the best correspondences between (a subset of) these features in an exponentially large search space. Our algorithm traverses a search tree, whose root is the empty set. Any other node is marked by a feature pair and a path from the root specifies a correspondence. A node (i, i') can be expanded by a feature pair (j, j'), yielding a child node, if the pairs satisfy the pruning constraints and feature j is *after* feature i according to some arbitrary feature ordering. This way, all possible (partial) correspondences that satisfy the constraints are *uniquely* represented in the tree. Our correspondence search is based on a *best-first* strategy and is implemented by a priority queue. At each iteration, the lowest-cost node is extracted from the queue and expanded. The cost of a correspondence is defined by a distortion energy. We seek correspondences which lead to low-distortion deformations anchored by the corresponding features.
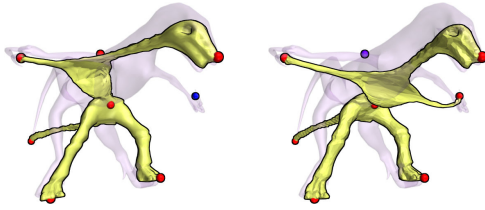
Figure 4: *Importance of* symmetric *cost. The dino-skeleton from Figure 1 (7 features) is matched with the T-rex from Figure 3 (8 features). There is no feature selected on the back of the dino-skeleton. Blue markers indicate unmatched features. Left: optimal correspondence based on a one-sided cost, i.e., cost of deforming the T-rex into the dino-skeleton is unaccounted for. We see a mismatch. Right: with a symmetric cost, the optimal correspondence is the natural one.*

5.1. Mesh deformation and correspondence cost

Given a correspondence π, we deform mesh M into mesh M' using the differential deformation scheme of Lipman et al. [LSLCO05]. As we compute deformations multiple times during our tree search, we require a method which is both efficient and reasonably robust; the differential deformation scheme addresses both our concerns. The input to the method consists of anchor points with their associated coordinate frames after the deformation. In our case, the anchor points are the matched feature vertices in π. The anchors are associated with the corresponded feature vertex normals, smoothed via polynomial filtering. Given the new anchor positions and normals, the new frames are estimated using a rigid transformation mapping the original positions of the anchors to the new ones.

We use the deformation result to establish a deformation-based cost for the correspondence π. We observe that an error metric based on rotated input normals [LSLCO05] is less accurate than a metric using the actual normals after deformation, as the accuracy of the former depends on the accuracy of the normal rotation. Thus we use the error functional defined in [KS06] to measure distortion. It uses the deformed mesh to estimate the normals, providing a more accurate error lower bound. We note that the actual deformation method described in [KS06] is not suitable for our needs as it solves a non-linear optimization and is fairly slow.

Denote the resulting distortion error by $\eta_1(\pi)$. We *symmetrize* the cost by also deforming M' into M, resulting in an error $\eta_2(\pi)$. The final correspondence cost is $\eta(\pi) = \max\{\eta_1(\pi), \eta_2(\pi)\}$. Figure 4 gives an example illustrating the importance of a symmetric cost measure.

Note that we do not combine distortion energy with feature similarity or other terms into the cost as that would require weighting between different terms. Also, the distortion is measured between the deformed mesh M and its original, i.e., a *self-distortion*, and not between M and M' since:

1. π is only a sparse feature correspondence. Estimating a distortion between the two meshes M and M' via a dense correspondence is non-trivial and expensive; it requires solving the cross-parameterization problem [KS04].

2. Using a global shape descriptor to compare the deformed and target meshes is not suitable in the presence of spurious object parts. The same can be said about the Hausdorff distance, which is also more expensive to compute.

5.2. Tree pruning

Tree pruning is implemented by enforcing constraints whenever a node is expanded. A key requirement is that if a node fails a constraint check then so will all of its descendants.

Local feature similarity: Let (i, i') be the feature pair defining a new node and denote by s^* the maximum pairwise feature dissimilarity. If the dissimilarity distance between i and i' exceeds $\varepsilon_{sim} s^*$, then the node is pruned. Here $\varepsilon_{sim} \in [0, 1]$ is a tolerance on feature dissimilarity. To compute feature dissimilarity in the non-rigid setting, we use a slight modification of the curvature map signature of Gatzke et al. [GGGZ05]. For a mesh vertex p, we construct b equally spaced geodesic rings around p — two adjacent rings bound a geodesic bin. We compute curvature averages in each bin, resulting in a b-dimensional signature. The dissimilarity between two signatures is given by the L_2 norm.

Geodesic distortion: Given a new node which extends a correspondence π, we check whether the new pair incurs a *geodesic distortion* with some feature pair in π that exceeds ε_{geod}, where $\varepsilon_{geod} \in [0, 1]$ is a tolerance value to control the level of tolerable stretching. If this is the case, the new node is pruned. We define the geodesic distortion for two feature pairs (i, i') and (j, j') by $\max\{||G_{ij} - G'_{i'j'}||, ||G_{ji} - G'_{j'i'}||\}$, where G and G' are the normalized pairwise geodesic distance matrices for features on M and M', respectively. To reduce possible influence of outlier features, we normalize the original geodesic distances into $[0, 1]$ on a per-row basis. As a result, G and G' are not symmetric in general.

Tolerance values: For non-rigid correspondence, one cannot possibly predict, under all circumstances, the level of shape variations that should be tolerated to produce a reasonable matching. In our current work, we determine the tolerances ε_{sim} and ε_{geod} empirically and leave fully automatic determination of their values for future work. Experimentally, we have found that setting the tolerances conservatively allows us to correctly match a wide variety of shapes, due to the effectiveness of the distortion energy.

5.3. Optimal matching size

Each node in the search tree represents a candidate solution: a partial matching between subsets of features from two input shapes. All solution candidates of the same size are
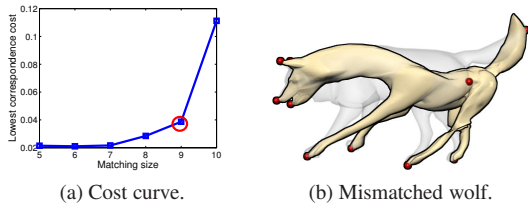
(a) Cost curve.   (b) Mismatched wolf.

Figure 5: *Finding the right matching size. (a) Plot of lowest correspondence costs against matching size, for the wolf pair in Figure 2. (b) Deformed wolf based on 10 feature pairs. There is a mismatch between a feature on the belly and a feature on a leg, causing a large jump in the curve.*

Table 1: *Benefit of deferred deformation calls. Result is from matching the wolves in Figure 2, with matching size $N = 9$.*

	$L_{def} = 6$	$L_{def} = 7$	$L_{def} = 8$
# Nodes expanded	2,803	4,332	4,926
# Deformations	1,600	602	73

ordered by their correspondence costs. For automatic correspondence, we determine the optimal matching size by a fairly standard scheme. We plot the lowest correspondence costs, denoted by c_i, as more features are included, where i runs from 5 (the smallest matching size we request) to the size of the largest candidate solution found. We look for a point i^* that is the *first peak* in the difference curve defined by $d_i = c_{i+1} - c_i$; this represents the first significant jump in the correspondence cost. Indeed, we expect a pair of unmatchable features to incur such a jump in distortion.

Figure 5(a) shows the cost curve for the wolf pair from Figure 2. The optimal matching size detected is 9 and the correspondence is shown in Figure 2(c). The lowest-cost matching of size 10 is shown in Figure 5(b), where we see a pair of mismatched features; they cause a large distortion and a jump in the cost curve. To obtain a meaningful cost curve, we require at least three cost values.

5.4. Implementation issues

While the original input meshes can be dense, we find that performing mesh deformation and measuring costs on reduced models to be sufficient; this can greatly reduce search time. In our implementation, we decimate a high-resolution input mesh down to having between 2,500 and 5,000 vertices. Note that estimation of normals, curvatures, and geodesics, as well as smoothing and curvature map computations, are performed on higher-resolution models.

To reduce search time further, we can trade deformation calls for node expansion and pruning checks, as the latter operations are less costly. Specifically, we use a parameter L_{def} to indicate the first tree level at which mesh deformation is enabled. Before this level, all nodes are assigned zero cost and only tree pruning applies. Table 1 shows how different values of L_{def} can affect the number of deformation calls. With a small L_{def}, many deformation calls were made for correspondences which would end up being pruned. It is much more efficient to prune them without computing a deformation, as is done by setting a higher L_{def}. It is important to note however that any solution returned by the search must

have an actual cost computed via mesh deformation. After all, tree pruning does not rate the correspondences. Therefore, if we seek matchings of size at least N, then we must have $L_{def} < N$. This ensures that all the returned solutions are evaluated by our correspondence cost.

6. Results

We have tested our deformation-driven correspondence algorithm on various models with pronounced features. These consist mainly of animals and human-like figures, which can vary significantly in pose, relative scale, part composition, and geometric detail. In the following experiments, we use 10 geodesic bins and a neighborhood size that is 0.3 of the maximum pairwise geodesic distance to construct the curvature maps. As we are interested in coarse feature correspondence, we fix the feature count at 10. For automatic feature selection, we first set $\sigma = 0.05$ and $\gamma = 0.2$. If at least 10 features are found, we select the 10 most prominent ones (refer to Section 4). Otherwise, we relax γ to 0.1 and repeat. This enables us to automatically select 10 features from each test model. Both tolerance values ε_{sim} and ε_{geod} are set conservatively at 0.4. The matching size is determined automatically by examining the cost curve, as described in Section 5.3. For a given matching size N, we set $L_{def} = N - 1$ to minimize the number of deformation calls.

We first take the five wolf models from the ISDB database and match one of them, shown in Figure 6(a), to the other four. No initial alignment is applied to any model pairs in our experiments, except for a uniform scale normalization. The final correspondence results are shown in Figure 6. They demonstrate the ability of our algorithm to handle pose; there is only minor stretching and little change in the local geometry of the models. In each case, 9 out of 10 features are properly matched and the one spurious feature is ignored by partial matching. We use the matching results to smoothly morph between wolves in different poses (Figure 7) by first utilizing the feature correspondences as input for dense cross-parameterization [KS04] and then interpolate the corresponding Laplacian coordinates between the models to obtain the intermediate geometries.

Moving on to more challenging examples, we match the triceratops (top-left model in Figure 8) to a rhino, two pigs which differ a great deal in the scale of their parts, and finally the feline with wings to exemplify partial matching in the presence of spurious parts. Figure 8 shows the automatically picked features on each model, the cost curves used
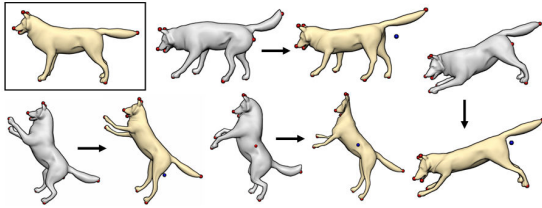
Figure 6: *Correspondence results between the wolf given in the box and four other wolves from the ISDB database, each with 10 automatically extracted features. Again, blue markers indicate unmatched features.*
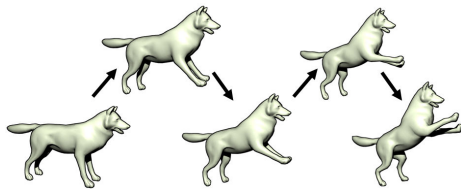

Figure 7: *Morphing result on a pair of wolves, based on dense cross-parameterization driven by initial correspondences (Figure 6) computed by our algorithm.*

to find the matching size, and the final correspondences returned. The cost curve starts at matching size 5 and stops when no solution is returned, as dictated by tree pruning based on the threshold parameters chosen. As we can see, results from automatic correspondence may miss some feature pairings which would have been detected by a human, e.g., all the tails of the animals. Nevertheless, these results are all correct partial matchings, which can be further refined by matching more features. This latter process would be a more localized and thus efficient search.

To showcase another dense cross-parameterization example using the computed feature correspondences as input, we cross-parameterized the triceratops and one of the pigs and use the result to obtian a blended model, shown in Figure 9, where the front of the body is taken from the pig and the back from the triceratops. The same procedure was also applied to the dino-skeleton and the raptor from Figure 1.

The ability of our algorithm to perform partial matching is evident in many of the visual examples shown so far: spurious features or parts (e.g., see the triceratops vs. the feline), on one or both input models, are appropriately ignored in the correspondence. Figure 10 gives another such example, where we note that if we switch the order of the two input meshes, the same correspondence is returned, since the pruning process and the correspondence cost are both symmetric. Also, observe that the models we match all exhibit symmetry. In no case has our algorithm produced a wrong correspondence resulting from symmetry switching.

The search for the best matching size is quite expensive computationally, since we maintain the same feature count, 10 in
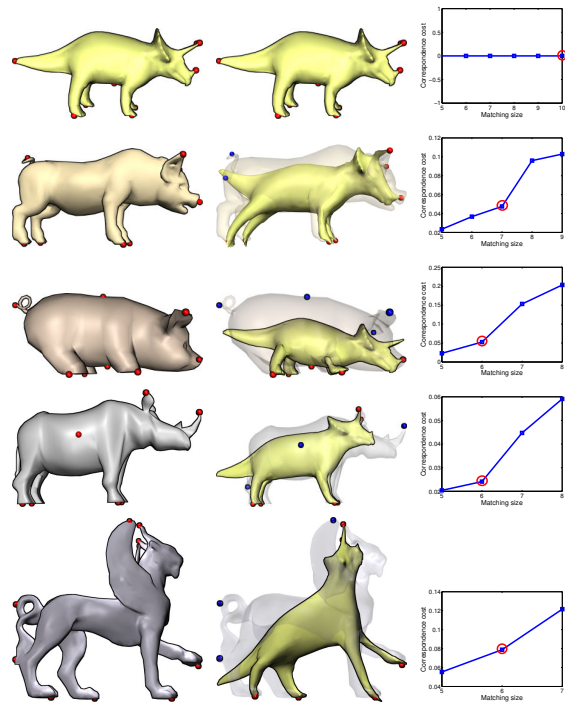

Figure 8: *Automatic feature correspondence between the triceratops (top left) and four different animals, as well as itself. First column shows the animals with selected features. Second column shows the correspondences (red markers are matched features and blue markers are unmatched ones). Cost curves shown in the last column are used to determine the matching sizes (circled in the plots).*
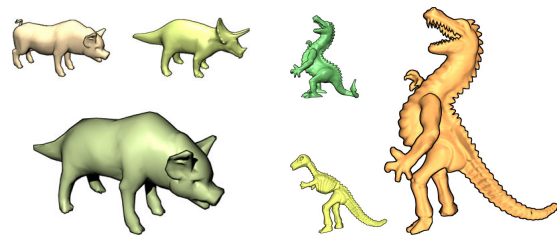

Figure 9: *Blending the front of one animal with the back of another created via dense cross-parameterization. Left: a "prehistoric pig" (triceratops and pig in the second row of Figure 8). Right: dino-skeleton and raptor from Figure 1.*

our case, and need to perform tree search for small values of L_{def}. The vertex counts for this series of models range between 606 and 3,552; the triceratops has 2,832 vertices. It took from 20 minutes to more than one hour to perform these fully automatic correspondences. We observe that majority of the search time was spent when $L_{def} \leq 5$. Therefore, it should be more efficient to construct the cost curve in reverse order, i.e., computing matchings of larger sizes first.
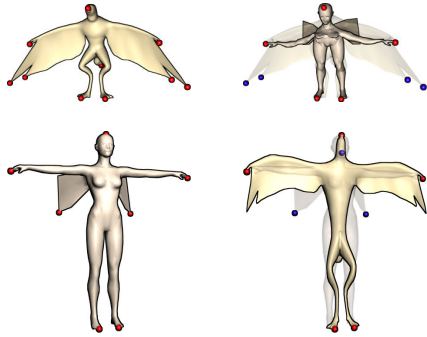
Figure 10: *Partial matching with spurious object parts. A woman with wings is matched with a dino-bird model.*

Improving the speed of the solver, e.g., by setting a higher tolerance for convergence, should also help. Another factor behind the high search cost is the conservative tolerance values we chose. A more stringent set of tolerances can provide many-fold speed-ups, but they are not always easy to find automatically. One possible approach worth investigating is to start with low tolerances and then gradually increase them while observing the correspondence costs.

7. Discussions and limitations

Refinement: A number of factors influence the performance of a correspondence algorithm. We first need a sufficient set of features to make the shapes matchable. However, the high cost of global search requires the feature count to be low, at least for the initial coarse correspondence. A low feature count may lead to locally inaccurate results, e.g., when some features are close to each other. We can then examine several candidate solutions (e.g., the top few from the our tree search based on correspondence cost) and use refinement to provide a final ranking. The refinement procedure inserts more features on two shapes and can simply match them up based on geodesic distances to the initial correspondences. Meshes are deformed based on the refined correspondence and the resulting costs provide the final ranking. Taking this to the limit, we would arrive at cross-parameterization, e.g., [SAPH04].

Dependent feature picking: Shape extremities are not always reliable as features for correspondence, e.g., consider the unlikely case of matching to a part embedded in a larger model where none of the model's extremities lie on that part. A possible remedy is to make use of local shape signatures. For example, Gelfand et al. [GMGP05] extract feature points from one shape that have the most unusual signatures. Each feature is matched against all points on the second shape, where features are identified via thresholding the similarity between shape signatures. For this to work well, we need high-quality local shape signatures and as discussed in Section 3.1, this is a difficult problem in the non-rigid setting.
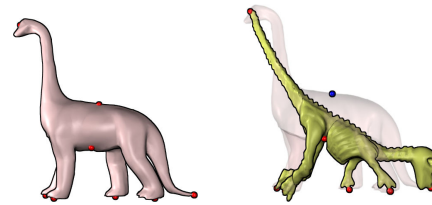

Figure 11: *Matching the dino-skeleton with a brontosaurus. The least-cost solution, shown on the right, is a wrong one.*

Choosing tolerance values: It is generally difficult to select the tolerance values automatically and conservative settings lead to high search cost. In practice, there are applications, e.g., statistical modeling [ASK*05], where feature correspondences are needed for a large set of models belonging to the same class. As the shape variations are not expected to be too large, our algorithm should work efficiently under a relatively stringent tolerance setting.

Limitation of the deformation approach: While the deformation view of shape correspondence is shown to be effective in many cases, as demonstrated by the results so far, it cannot be expected to replicate the way humans perform correspondence. Figure 11 shows the result of matching the dino-skeleton with a brontosaurus model via deformation. Indeed, it is the least-distorting way to correspond them, yet it is wrong. We argue that the reason we are able to match these two models correctly is that we can recognize them. Based purely on geometry, it can be an impossible task.

8. Conclusion and future work

It is generally believed that shape correspondence by human involves an object recognition phase. That is, two shapes are first recognized as belonging to certain classes before identifying feature correspondences. Without relying on this phase, we develop a purely geometric approach which emphasizes the global characteristics of shape correspondence. When large shape variations are tolerated, local shape descriptors can become unreliable. We take a deformation view of shape correspondence and compute partial matchings between two meshes which lead to low-distortion deformation. The ability of our algorithm to handle shape symmetry and shape variations in pose, local scale, part composition, and geometric detail is demonstrated through various examples.

In future work, we would first like to improve the speed of tree search, while retaining an automatic algorithm. Therefore, we need effective means to find proper tolerance values and techniques to prune the tree search more aggressively. Pruning and feature selection can both benefit from having a high-quality local shape descriptor. The key question appears to be how to deal with local stretching or scale changes near a feature point. We would like to look into the use of *feature-sensitive* neighborhood traversals [LZH*07] when constructing local shape descriptors.

References

[ACOL00] ALEXA M., COHEN-OR D., LEVIN D.: As-rigid-as-possible shape interpolation. In *SIGGRAPH* (2000), pp. 157–164.

[ACP03] ALLEN B., CURLESS B., POPOVIC Z.: The space of human body shapes: reconstruction and parameterization of range scans. *ACM Trans. on Graphics 22*, 3 (2003), 587–594.

[ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: shape completion and animation of people. *ACM Trans. on Graphics 24*, 3 (2005), 408–416.

[BM92] BESL P. J., MCKAY N. D.: A method for registration of 3-d shapes. *IEEE Trans. Pat. Ana. & Mach. Int. 14*, 2 (1992), 239–256.

[BMSF06] BIASOTTI S., MARINI S., SPAGNUOLO M., FALCIDIENO B.: Sub-part correspondence by structural descriptors of 3d shapes. *Computer-Aided Design 38*, 9 (2006), 1002–1019.

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3d faces. In *SIGGRAPH* (1999), pp. 187–194.

[COSL98] COHEN-OR D., SOLOMOVIC A., LEVIN D.: Three-dimensional distance field metamorphosis. *ACM Trans. on Graphics 17*, 2 (1998), 116–141.

[CR03] CHUI H., RANGARAJAN A.: A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding 89* (2003), 114–141.

[EK03] ELAD A., KIMMEL R.: On bending invariant signatures for surfaces. *IEEE Trans. Pat. Ana. & Mach. Int. 25*, 10 (2003), 1285–1295.

[FS06] FUNKHOUSER T., SHILANE P.: Partial matching of 3d shapes with priority-driven search. In *Proc. Eurographics Symp. on Geometry Processing* (2006), pp. 131–142.

[GCO06] GAL R., COHEN-OR D.: Salient geometric features for partial shape matching and similarity. *ACM Trans. on Graphics 25*, 1 (2006), 130–150.

[GGGZ05] GATZKE T., GRIMM C., GARLAND M., ZELINKA S.: Curvature maps for local shape comparison. In *Proc. IEEE Conf. on Shape Modeling and Applications* (2005).

[GMGP05] GELFAND N., MITRA N., GUIBAS L., POTTMANN H.: Robust global registration. In *Proc. Eurographics Symp. on Geometry Processing* (2005), pp. 197–206.

[GR96] GOLD S., RANGARAJAN A.: A graduated assignment algorithm for graph matching. *IEEE Trans. Pat. Ana. & Mach. Int. 18*, 4 (1996), 377–388.

[GSL*98] GREGORY A. D., STATE A., LIN M. C., MANOCHA D., LIVINGSTON M. A.: Feature-based surface decomposition for correspondence and morphing between polyhedra. In *Proc. Computer Animation* (1998), pp. 64–71.

[Heb49] HEBB D. O.: *The Organization of Behavior*. John Wiley, 1949.

[HFG*06] HUANG Q.-X., FLÖRY S., GELFAND N., HOFER M., POTTMANN H.: Reassembling fractured objects by geometric matching. In *SIGGRAPH* (2006), pp. 569–578.

[HH03] HUBER D., HEBERT M.: Fully automatic registration of multiple 3d data sets. *Image and Vision Computing 21*, 7 (2003), 637–650.

[HSKK01] HILAGA M., SHINAGAWA Y., KOHMURA T., KUNII T. L.: Topology matching for fully automatic similarity estimation of 3d shapes. In *SIGGRAPH* (2001), pp. 203–212.

[KLT05] KATZ S., LEIFMAN G., TAL A.: Mesh segmentation using feature point and core extraction. *The Visual Computer (Pacific Graphics) 21*, 8-10 (2005), 649–658.

[KS04] KRAEVOY V., SHEFFER A.: Cross-parameterization and compatible remeshing of 3D models. In *SIGGRAPH* (2004), pp. 861–869.

[KS06] KRAEVOY V., SHEFFER A.: Mean-value geometry encoding. *International Journal of Shape Modeling 12*, 1 (2006).

[LG05] LI X., GUSKOV I.: Multi-scale features for approximate alignment of point-based surfaces. In *Proc. Eurographics Symp. on Geometry Processing* (2005), pp. 217–226.

[Lin94] LINDEBERG T.: Scale-space theory: A basic tool for analysing structures at different scales. *J. of Applied Statistics 21*, 2 (1994), 224–270.

[LSLCO05] LIPMAN Y., SORKINE O., LEVIN D., COHEN-OR D.: Linear rotation-invariant coordinates for meshes. In *SIGGRAPH* (2005), pp. 479–487.

[LZH*07] LAI Y.-K., ZHOU Q.-Y., HU S.-M., WALLNER J., POTTMANN H.: Robust feature classification and editing. *IEEE Trans. Vis. & Mach. Comp. Graphics 13*, 1 (2007), 34–45.

[OBS04] OHTAKE Y., BELYAEV A., SEIDEL H.-P.: Ridge-valley lines on meshes via implicit surface fitting. In *SIGGRAPH* (2004), pp. 609–612.

[RHZ76] ROSENFELD A., HUMMEL R. A., ZUCKER S. W.: Scene labelling by relaxation operations. *IEEE Trans. Systems, Man, & Cybernetics 6*, 6 (1976), 420–433.

[SAPH04] SCHREINER J., ASIRVATHAM A., PRAUN E., HOPPE H.: Inter-surface mapping. In *SIGGRAPH* (2004), pp. 870–877.

[SG92] SEDERBERG T. W., GREENWOOD E.: A physically based approach to 2-D shape blending. In *SIGGRAPH* (1992), pp. 25–34.

[SH81] SHAPIRO L., HARALICK R. M.: Structural descriptions and inexact matching. *IEEE Trans. Pat. Ana. & Mach. Int.* (1981), 504–519.

[She00] SHELTON C. R.: Morphable surface models. *Int. J. Comput. Vision 38*, 1 (2000), 75–91.

[SP04] SUMNER R., POPOVIC J.: Deformation transfer for triangle meshes. *ACM Trans. on Graphics 23*, 3 (2004), 399–405.

[SSGD03] SUNDAR H., SILVER D., GAGVANI N., DICKINSON S.: Skeleton based shape matching and retrieval. In *Proc. IEEE Conf. on Shape Modeling and Applications* (2003), pp. 130–142.

[TS04] TUNG T., SCHMITT F.: Augmented reeb graphs for content-based retrieval of 3d mesh models. In *Proc. IEEE Conf. on Shape Modeling and Applications* (2004), pp. 157–166.

[VG05] VAILLANT M., GLAUNÉS J.: Surface matching via currents. In *Info. Proc. in Med. Imaging* (2005), pp. 381–392.

[ZMT05] ZHANG E., MISCHAIKOW K., TURK G.: Feature-based surface parameterization and texture mapping. *ACM Trans. on Graphics 24*, 1 (2005), 1–27.

[ZRKS05] ZAYER R., RÖSSL C., KARNI Z., SEIDEL H.-P.: Harmonic guidance for surface deformation. *Computer Graphics Forum 24*, 3 (2005), 601–609.