The Correlated Correspondence Algorithm for Unsupervised Registration of Nonrigid Surfaces ** Stanford AI Lab Technical Report SAIL-2004-100 ***

Dragomir Anguelov* Stanford University
Daphne Koller† Stanford University
Praveen Srinivasan‡ Stanford University
Sebastian Thrun§ Stanford University
Hoi-Cheung Pang¶ Stanford University
James Davis‖ Honda Research Labs

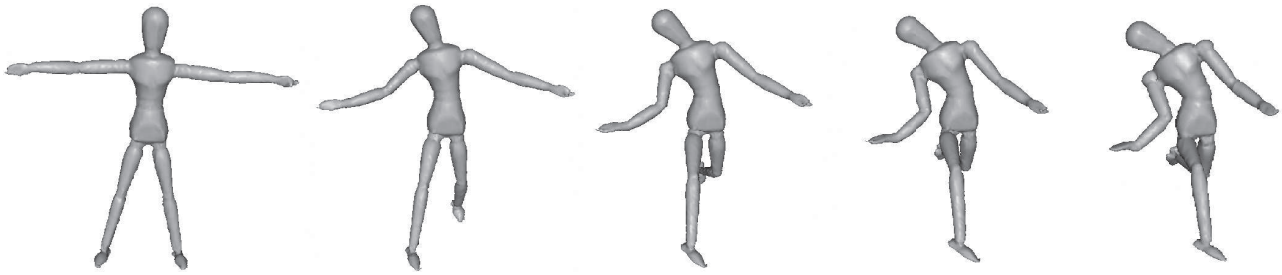

Figure 1: Several frames from a motion animation generated by interpolating two scans of a puppet (far left and far right), which were automatically registered using the Correlated Correspondence algorithm.

Abstract

We present an unsupervised algorithm for registering 3D surface scans of a deformable object in very different configurations. Our algorithm does not use markers, nor does it assume prior knowledge about object shape, the dynamics of its deformation, or its alignment. The algorithm finds the correspondences between points in the two meshes using a joint probabilistic model over all point correspondences. This model combines preservation of local mesh geometry with more global constraints that capture the preservation of geodesic distance between corresponding pairs of points in the two meshes. Our approach successfully registers scans that exhibit large transformations, including both movement of articulate parts and non-rigid surface deformations. It applies even when one of the meshes is an incomplete range scan; thus, it can be used to automatically fill in the remaining surfaces for this partial scan, even if those surfaces were previously only seen in a different configuration. We also show how our results can be used to interpolate between two scans of a non-rigid object in a way that preserves surface geometry, leading to natural motion paths. Finally, we show that a registration of multiple scans in different configurations allows us to automatically identify components in articulate objects.

1 Introduction

The construction of 3D object models is a key task for many graphics applications. It is becoming increasingly common to acquire these models from a range scan of a physical object. This report deals with an important subproblem of this acquisition task — the problem of registering two different (partial or complete) scans of an object.

In the case of rigid objects, the surface registration problem is typically solved using the well-known ICP algorithm [Besl and McKay 1992; Chen and Medioni 1991; Rusinkiewicz and Levoy 2001]. However, many real-world objects are non-rigid, either because they contain multiple parts whose configuration can vary, or because the surface itself can deform. Some work has attempted to address the surface registration problem for deformable objects. So far, however, these methods do not give a general-purpose solution. Some are *supervised*, relying on the presence of markers on the object [Allen et al. 2002] for registering multiple poses, or on a good initial alignment [Shelton 2000; Hähnel et al. 2003], or knowledge of the object dynamics [Lin 1999]. Others assume significant prior knowledge on the object's shape [Leventon 2000; Blanz and Vetter 1999]. Algorithms that make neither restriction generally apply only in cases where the object deformation is relatively small. This report describes what, to our knowledge, is the first algorithm that allows the surface registration of a deformable 3D object where the object configurations can vary significantly, there is no prior knowledge about object shape or dynamics of deformation, and nothing whatsoever is known about the object alignment. Moreover, unlike many methods, our algorithm can be used to register a partial scan to a complete model, greatly increasing its applicability.

As we show in this report, the main difficulty in the registration problem is determining the *correspondences* of points on one mesh to points on the other. Local regions on the are rarely distinctive enough to determine the correct correspondence, whether because of noise in the scans, or because of symmetries in the object shape. Thus, the set of candidate correspondences to a given point is usually large. As we need to determine the correspondence for all points in one object, the resulting space of possibilities is combinatorially large, rendering the problem computationally infeasible. Algorithms that rely on markers, or that only apply to very similar configurations, obtain much of their power because they greatly simplify the correspondence problem.

*e-mail: drago@cs.stanford.edu
†e-mail:koller@cs.stanford.edu
‡e-mail:praveens@cs.stanford.edu
§e-mail:thrun@stanford.edu
¶e-mail:hcpang@cs.stanford.edu
‖e-mail:jedavis@graphics.stanford.edu

Many algorithms [Besl and McKay 1992; Shelton 2000] address the correspondence problem by decomposing the problem, choosing a correspondence for each point in the scan separately. However, this approach allows nearby points in one scan to be mapped to far-away regions in the other, leading to poor registrations. The key idea behind our approach is that we avoid this decomposition. Rather, our *Correlated Correspondence* algorithm defines a joint probabilistic model over all point correspondences, which explicitly encodes the correlations between nearby points — specifically, the fact that nearby points on one mesh should correspond to nearby points on the other. Importantly, the notion of "nearby" used in our model is defined in terms of geodesic distance over the mesh, a more appropriate measure in this context than the standard Euclidean distance. We define a probabilistic model over the set of correspondences, that encodes these geodesic distance constraints as well as penalties for link twisting and stretching, and high-level local surface features [Johnson 1997]. We then apply probabilistic inference techniques [Pearl 1988] to this model, in order to solve for the entire set of correspondences simultaneously. The result is a registration that respects the surface geometry.

We apply our approach to two different registration problems. The first involves two scans of a human puppet, acquired using a range finder previously described in [Davis et al. 2003]. This data set poses challenges, because the meshes are in significantly different configurations, and because the puppet model is symmetrical, leading to ambiguities in the registration. Nevertheless, our approach correctly aligns the two configurations. The second data set contains different configurations of a flexing arm [Allen et al. 2002], which exhibits both articulated parts and significant deformation in the surface of the parts.

We also describe three applications of our method that showcase its utility. In the first, we use the correspondences found by our algorithm to smoothly interpolate between two different poses of an object. We define a motion path that preserve (to the extent possible) the local geometry of the surface points, obtaining motion paths that minimally deform the object. Our approach applies without any assumption about the existence or parameterization of an articulated skeleton. In our second application, we show how a partial scan of an object can be registered onto a fully specified model in a different configuration. The resulting registration allows us to use the model to "complete" the partial scan in a way that preserves the local surface geometry. This technique essentially provides a marker-free motion capture method, and as such should find use in the entertainment and gaming industries. In our final application, we use a set of scans of the same object to determine which points in the surface tend to have similar deformation dynamics, thereby determining the partition into parts of an articulated object as well as the configurations of its skeleton in the different scans. Importantly, all of these applications are done in an unsupervised way, using only the output of our Correlated Correspondence algorithm applied to pairs of poses with widely varying deformations, and unknown intiial alignments. These results demonstrate the value of a high-quality solution to the registration problem to a range of graphics tasks.

2 Previous Work

Modeling applications that require the registration of complex 3D surfaces often use a sparse set of markers placed on the scanned objects [Allen et al. 2002; Allen et al. 2003]. The markers allow the registration algorithm to obtain a good initial surface alignment, simplifying the correspondence problem.

There are also several approaches whose goal is to register surfaces in an unsupervised manner. The classic algorithm addressing this task is the Iterated Closest Point (ICP) algorithm [Besl and McKay 1992; Chen and Medioni 1991]. ICP iterates between estimating the correspondences and estimating the transformation between the two meshes. ICP assumes that the object is rigid, so that all of the points in a mesh undergo the same transformation. When applied to objects that violate this assumption in a significant way, ICP breaks down.

More recently, there has been work extending ICP to the non-rigid case [Chui and Rangarajan 2000; Shelton 2000; Hähnel et al. 2003]. Shelton [2000] and Hähnel *et al.* [Hähnel et al. 2003] both use an approach similar to ICP, iterating between estimating the correspondences and estimating the transformation. However, as we show below, such approaches are likely to converge to local minima in the absence of a good initialization. Standard techniques can be used to improve the convergence, such as the use of coarse-to-fine resolution strategies, and the use of local features such as color, if available [Shelton 2000]. However, these solutions only partly address the problem of incorrect correspondences, where points in one part of the mesh are mapped to distant points in the other.

The TPS-RPM method of Chui and Rangarajan [2000] takes a more global approach. It maintains beliefs over the correspondence estimates, which are annealed to become more deterministic. This makes the algorithm more tolerant of incorrect initialization than the others in its class. However, TPS-RPM algorithm attempts to find a registration that preserves the Euclidean distances between all pairs of points, making it inappropriate for articulated objects where global Euclidean distances can change drastically.

Another set of approaches uses prior knowledge about the space of transformations an object can undergo. Given previously registered meshes from the same object class, they create a parametric representation of the surface variability. For this, principal component analysis is applied either to a set of registered meshes [Blanz and Vetter 1999; Allen et al. 2003] or to aligned volumetric representations such as active level sets [Leventon 2000]. A registration of a new surface in the same class to the model can be established by optimizing for the best alignment and for the best set of principal components describing the deformation of the model. Such algorithms are often quite sensitive to the initial alignment. Moreover, the types of deformations that can be well-encoded through linear PCA is quite restricted. Thus, these approaches often work well for largely convex objects, but are unsuccessful at representing the deformation space of surfaces that have branching parts such as arms.

Our algorithm is most closely related to computer vision algorithms for non-rigid template matching. Systems for detection of articulated objects in images [Huttenlocher and Felzenszwalb 2003; Yu et al. 2002] assume that the object articulation is known, reducing the general correspondence problem to one where we need only find correspondences for the parts composing the shape. They also assume that the available detectors provide a set of discrete hypotheses for the location of each part, greatly reducing the space of possible matches for each part. Like our algorithm, these approaches exploit correlations between the correspondences of adjacent parts. With these assumptions, the correspondence problem reduces to the task of determining a position for each part (within the circumscribed set of hypotheses) so that the resulting configuration is consistent with the object's articulation. When the graph of pairwise correlations is a tree, efficient dynamic programming algorithms can give the most likely match [Huttenlocher and Felzenszwalb 2003]. When the correlation graph has loops, graph-partitioning algorithms can be applied [Yu et al. 2002]. When the orientation of an articulated 3D human body template is being inferred from image data, reasoning for both correspondence and orientation can be performed simultaneously for a body model consisting of nine parts [Sigal et al. 2003].

Template matching approaches have also been applied to deformable 2D objects. The method of Felzenszwalb [2003]. finds a globally-optimal embedding of a morphable 2D template, repre-

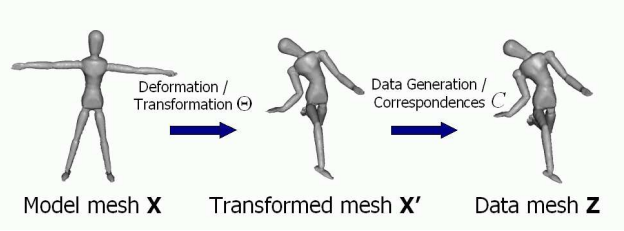

Figure 2: Probabilistic generative model for the non-rigid ICP algorithm. The data mesh Z is generated by applying a non-rigid transformation Θ to the model mesh X, and then resampling the transformed mesh (guided by the correspondence variables C between the meshes). Given the two meshes X and Z, the task of registration is to recover the correspondences C and the nonrigid transformation Θ.

sented as a set of deformable triangles, in an image. The algorithm solves for the optimal value of the correspondence variables via dynamic programming. However, the model uses a deformation parameterization which is particular to 2D; it also relies on a strong prerequisite that the template triangulation belongs to a constrained set of triangulations, preventing its practicality for 3D meshes. The triangulation constraint is relaxed in the work of Coughlan and Ferreira [2002]. They register a morphable 2D template to image data by employing a framework which has similarities to the work presented in this report. Our algorithm differs from their method because it addresses important issues that arise in 3D registration and in dealing with range data. In particular, our model enforces approximate preservation of geodesic distances during registration and allows the registration of partial views of an object to the object template shape.

3 Nonrigid-ICP for Deformable Surface Registration

The assumptions about the registration process can be expressed in a *generative* probabilistic framework, which has been used to produce state-of-the art 3D registration algorithms [Allen et al. 2002; Shelton 2000; Hähnel et al. 2003]. In this section we describe the framework and demonstrate that the existing algorithms fail to register of surfaces undergoing large deformations in an unsupervised manner. In the next section we will leverage the insights obtained by analyzing the current algorithms to define a new algorithm which performs well even for strongly-deforming surfaces.

3.1 Deformable Surfaces

The current 3D registration framework is based on the premise that the model is a deformable surface, and the data is a set of points sampled from this surface. The generative process assumes that the model is deformed using some unknown nonrigid transformation Θ, and then the set of data points Z is sampled from the deformed model (see Fig. 2). We are interested to recover transformations which align the model and data points while minimizing the amount of model surface deformation necessary for the alignment.

More concretely, the model is a complete specification of the shape of the object involved, in some particular configuration. This specification is usually given as a mesh $X = (V^X, E^X)$, where $V^X = (x_1, \ldots, x_N)$ is the set of mesh points, and E^X is a set of *links*. In general, links may be edges in the original mesh; however, we sometimes sub-sample the original mesh, in which case the links

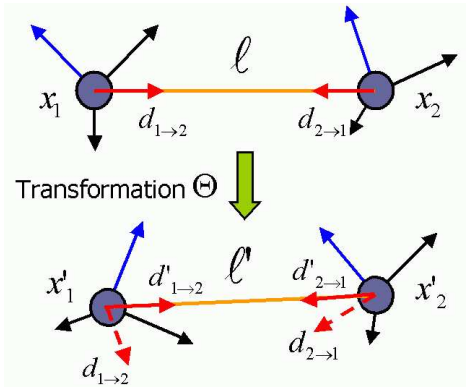

Figure 3: Illustration of the link deformation process. The nonrigid transformation Θ moves the locations and rotates the local coordinate systems of the link endpoints.

connect points in the sub-sampled data that are nearby on the object surface.

In order to quantify the amount of deformation applied to the model by Θ, we will follow the ideas of Hähnel *et al.* [Hähnel et al. 2003] and treat the links in the set E^X as springs, which resist stretching and twisting at their endpoints. Stretching is easily quantified by looking at changes in the link length induced by the transformation Θ. Link twisting, however, is ill-specified by looking only at the cartesian coordinates of the points alone. We adopt the solution of Hähnel *et al.*, who addressed this issue by attaching an imaginary *local coordinate system* to each point on the model, and augmenting the Cartesian coordinates of each point with an orientation, represented as a triple of Euler angles. This local coordinate system allows them to quantify the "twist" of a point x_j relative to a neighbor x_i. A non-rigid transformation Θ defines, for each point x_i, a translation of its coordinates and a rotation of its local coordinate system.

To evaluate the deformation penalty, we parameterize each link in the model in terms of its length and its direction relative to its endpoints (see Fig. 3. Specifically, we define $l_{i,j}$ to be the distance between x_i and x_j; $d_{i\to j}$ is a unit vector denoting the direction of the point x_j in the coordinate system of x_i (and vice versa). We use $e_{i,j}$ to denote the set of edge parameters $(l_{i,j}, d_{i\to j}, d_{j\to i})$.

It is now straightforward to specify the penalty for model deformations. Let Θ be a transformation, and let $\tilde{e}_{i,j}$ denote the triple of parameters associated with the link between x_i and x_j in the transformed model mesh. Our model penalizes twisting and stretching independently, using a zero-mean Gaussian noise model for each one separately:

$$P(\tilde{e}_{i,j} \mid e_{i,j}) = P(\tilde{l}_{i,j} \mid l_{i,j})\, P(\tilde{d}_{i\to j} \mid d_{i\to j})\, P(\tilde{d}_{j\to i} \mid d_{j\to i}) \qquad (1)$$

The data surface is a sample of the transformed surface. It is defined as a mesh $Z = (V^Z, E^Z)$, which we assume to be a subset of the transformed model mesh $\tilde{X}$. Each point z_k in the data mesh is therefore associated with a *correspondence variable* c_k, which specifies the model point that generated the data point z_k. Specifically, $c_k = i$ when z_k was sampled from $\tilde{x}_i$, which is the transformed location of model point x_i. We assume that each data mesh point z_k gets generated with zero mean Gaussian noise around z_i: $p(z_k \mid c_k = i, \tilde{x}_i) = \mathcal{N}(\tilde{x}_i, \mathrm{diag}(\sigma_Z))$, where $\mathrm{diag}(\sigma_Z)$ is a known diagonal covariance matrix.

This framework allows for smooth deformations of the model to account for the points seen in the data mesh, and hence goes beyond the standard rigid ICP framework. The deformation penalty is defined locally, over individual links, allowing for arbitrary transformations of the model surface, but larger deformations are penalized

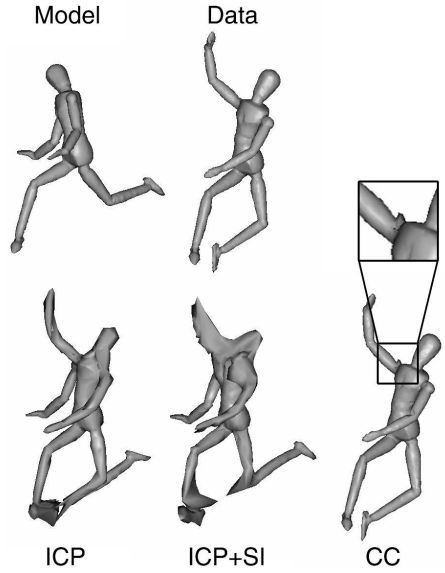

Figure 4: Registration results for meshes ((a) and (b)) using different algorithms. (c) Nonrigid ICP gets stuck in a local minimum, due to incorrect initial correspondences. Points on the head are mapped to the right arm, while points on the right shoulder are mapped to the head. (d) Incorporating spin-images in the nonrigid ICP distance function does not address the problem of incorrect correspondences. (e) The Correlated Correspondence algorithm produces a largely correct registration, although with an artefact in the right shoulder (inset).

more than small ones. The key problem in solving the registration problem is finding the correspondences between model points and data points.

3.2 Applying ICP to the Correspondence Problem

The formulation of the non-rigid registration process described in the section above is used as a basis for an iterative algorithm similar to the original ICP algorithm. This *nonrigid ICP* approach, taken by both Hähnel *et al.* and by Shelton [2000], iterates between two phases. It assumes that an initial alignment between the meshes is known (equivalent to having an estimate of transformed model mesh $\tilde{X}$ in Fig. 2). Given this initial alignment, the first phase computes a maximum likelihood correspondence for each point in the data mesh, simply based on least-squares distance to the model. This process is done separately for each correspondence variable. Based on the estimated correspondences, the second phase of the algorithm calculates the most likely transformation of the model. This process involves a minimization of the deformation penalty as defined in the previous section. The result is a new estimate of the deformed model $\tilde{X}$, which is used in another iteration of the process.

The decomposition of the original problem into two subproblems — estimating the transformations and inferring the correspondences — allows an efficient iterative solution to the non-rigid registration problem. However, this decomposition also induces the algorithm's main limitation. By assigning points in the data to closest points on the model, nearby points in the data frequently get associated to remote regions in the model. The subsequent alignment optimization only reinforces the false associations by bending the model accordingly. This problem is clearly illustrated in Fig. 4, which shows the results of running this ICP algorithm in an attempt to register two views of an articulate puppet of a human body. In the initial alignment, points on the head in the data mesh are mapped to the puppet's right arm on the model mesh, whereas points on the left shoulder of the data mesh are mapped to points on the head. This bad initial mapping is only reinforced during the course of the iterations, leading to incorrect correspondences.

One might think that a more informed selection of correspondences might remedy the problem. To this end, we included in the model a richer description of surface properties. In particular, we use surface signatures called *spin images* [Johnson 1997] to refine our probabilistic model. Spin images are commonly used to register scans for rigid objects. In contrast to other surface descriptors, they are robust against the clutter and occlusion generally present in range data. Experimentally, spin images have been shown to perform well when the normal vectors of the object surfaces can be accurately estimated [Ruiz-Correa et al. 2001]. Unfortunately, as shown in Fig. 4, spin images do not address the fundamental problem with this approach. Although these high-level features help eliminate some of the worst correspondences, nearby points in the data mesh can still be mapped to different regions in the model.

These problems limit the applicability of the nonrigid ICP approach to problems where the deformation is local, and the initial alignment is approximately correct.

4 The Correlated Correspondence Algorithm

The heart of the problem with the ICP approach is that it decouples the choice of correspondence for the different points in the data mesh, treating them as independent (given the current alignment). This allows two points that are adjacent in one mesh to be mapped to points that are far away in the other. The key idea behind our approach is that we explicitly correlate the choice of the different correspondence variables in the data mesh, eliminating this problem.

We formulate the problem as one of finding an embedding of the data mesh Z into the model mesh X, which is encoded as an assignment to all correspondence variables C. Specifically, for each pair of adjacent data points z_k, z_l, we want to define a probabilistic potential $\psi(c_k, c_l)$ that constrains this pair of correspondences to be a reasonable one. We can define a joint distribution which is of the form $p(C) \propto \prod_{k,l \text{ adjacent}} \psi(c_k, c_l)$, containing only pairwise potentials. Performing probabilistic inference to find the most likely joint assignment to the entire set of correspondence variables C should yield a good registration.

Note that unlike nonrigid ICP, this framework subsumes the reasoning about the nonrigid transformations into the combinatorial optimization problem over the correspondences. We avoid the need to maintain an explicit hypothesis about the transformed model $\tilde{X}$ as a whole, which would place us in a local minimum with regards to finding correspondences.

4.1 Geodesic Distances

Our proposed approach raises the question as to what constitutes the best constraint between neighboring correspondence variables. The literature on scan registration — for rigid and non-rigid models alike — relies on the Euclidean distance: the closer the Euclidean distance of two data points to that of the corresponding model points, the more likely this correspondence. While Euclidean distance is meaningful for rigid objects, it is relatively meaningless for non-rigid objects. This is because the Euclidian distance between points is sensitive to deformations, especially those induced by moving parts. For example, in Fig. 5, we see that the two legs in one configuration of our puppet are fairly close together, allowing

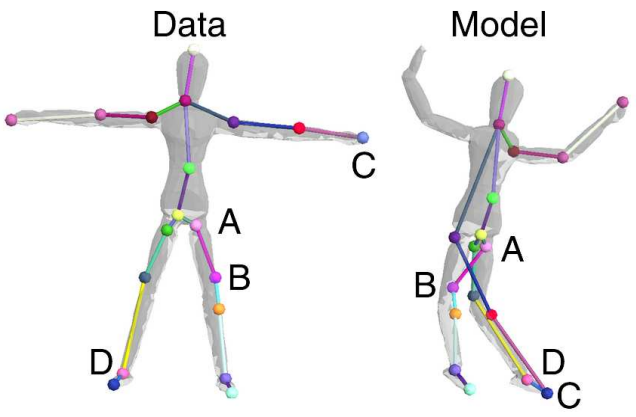

Figure 5: A Correlated Correspondence Algorithm which only models the preservation of inter-point distances in Euclidean space can violate mesh geometry. Links can cross between two parts of the model mesh which are close in Euclidean space, but distant along the mesh surface (link between points A and B). Conversely, two distant parts of the data mesh can map to the same part of the model mesh (points C and D). A run of the Correlated Correspondence algorithm with the same settings as before, but introducing the geodesic distance preservations constraints produces a good registration.

the algorithm to map two adjacent points in the data mesh to the two separate legs, with minimal penalty. In the complementary situation, the local geometry around two points might be quite similar, a common situation in objects with significant symmetries. In this case, two distant yet similar points might get mapped to the same region. For example, in the same figure, we see that points in both an arm and a leg in the data mesh get mapped to a single leg in the model mesh.

The insight that geodesic distance is the right way of parameterizing mesh surfaces has already been extensively used in graphics, one example is the system of Krishnamurthy and Levoy [2002], which allows the geometric detail manipulation in dense polygon meshes. Our approach also relies on geodesic distance — instead of the common Euclidean distance — to constrain neighboring correspondence variables. The geodesic distance is the distance of the shortest path between two points, measured along the surface of the mesh. It is easily calculated using standard shortest path algorithms (e.g., Dijkstra's algorithm). The key motivation for using geodesic distance is that it is invariant to most shape deformations. Thus, it is a good measure of consistency of registrations.

Our approach easily naturally us to encode consistency constraints that use geodesic distance to avoid problems like the ones in Fig. 5. These constraints manifest as correlations between the assignment to adjacent correlation variables. In particular, we introduce *nearness preservation* constraints, that enforce the fact that adjacent points on the data mesh cannot be mapped to points that are geodesically far apart in the model. We also introduce *farness preservation* constraints to enforce the complementary constraint, that points that are geodesically far apart in the data cannot map to nearby points in the model. Both constraints are naturally implemented within our Correlated Correspondence framework as pairwise potentials between correspondence variables.

4.2 Probabilistic Model

Our overall probabilistic model is a joint distribution over the correspondence variables $c_1, \ldots, c_M$, represented as an undirected graphical model, or a *Markov network* [Pearl 1988]. The network encodes a joint probability distribution as a product of potentials over the correspondence variables:

$$p(C) = \frac{1}{Z} \prod_k \psi(c_k) \prod_{k,l} \psi(c_k, c_l) \tag{2}$$

where Z is a normalizing constant. There are three kinds of pairwise potentials which we will encode: link potentials, nearness preservation potentials and farness preservation potentials. We will also define single variable potentials which will enforce preference for matching regions with similar local surface descriptors.

The link potentials encode the link deformation penalties. For each pair of adjacent points z_k, z_l in the data mesh, we include a potential $P(\tilde{e}_{i,j} \mid e_{i,j})$, defined as in Eq. (1). However, the link parameters $\tilde{e}_{i,j}$ depend on the knowledge of the nonrigid transformation, which is not given as part of the input. (Indeed, estimating it is part of the goal of the algorithm.) The key issue is estimating the rotation of a point's coordinate system that is induced by the (unknown) transformation. In effect, this rotation is an additional latent variable, which must also be part of the probabilistic model. To remain within the realm of discrete Markov networks, allowing the application of standard probabilistic inference algorithms, we discretize the space of the possible rotations, and fold it into the domains of the correspondence variables. For each possible value of the correspondence variable, we select a small set of candidate rotations, consistent with local geometry. Specifically, for $c_k = i$, we align the local surface around x_i and z_k using the surface normal, and then run ICP on these local regions from a number of different starting points (we have found experimentally that two diametrically opposite points suffice) . The rotations produced by ICP from these starting points defines our set of candidate rotations for the correspondence $c_k = i$. In this extended model, the correspondence variable c_k can take on a value which specifies not only the point i, but also a particular rotation in this preselected set. Given this rotation, the set of edge parameters $\tilde{e}_{i,j}$ is fully determined, and so is the probabilistic potential.

The nearness preservation potentials define, for each pair of *adjacent* points z_k, z_l in the data mesh, a 0-1 potential that encodes a nearness preservation constraint:

$$\psi(c_k = i, c_l = j) = \begin{cases} 0 & \text{dist}_{\text{Geodesic}}(x_i, x_j) > \alpha \, \text{dist}_{\text{Euclidean}}(x_i, x_j) \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

where α is some constant, chosen to be 3.5ρ, where ρ is the data mesh resolution. To understand this constraint, recall that our basic link deformation penalties already penalize cases where the Euclidean distances between corresponding pairs of points are not preserved. This constraint augments that basic penalty by testing whether the geodesic distance between the model mesh points x_i and x_j is similar to their Euclidean distance. If it is much larger, than these two points are far in the mesh, and should not be mapped to the much closer pair of data mesh points z_k, z_l.

The farness preservation potentials encode the complementary constraint. For *every* pair of points z_k, z_l whose geodesic distance is more than 5ρ on the data mesh, we have a potential:

$$\phi(c_k = i, c_l = j) = \begin{cases} 0 & \text{dist}_{\text{Geodesic}}(x_i, x_j) < \beta\rho \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

where β is also a constant, chosen to be 2 in our implementation. The intuition behind this constraint is fairly clear: If z_k, z_l are far apart on the data mesh, then their corresponding points must be far apart on the model mesh. Note that our use of correlated correspondence variables easily allows us to accomodate both of these types of geodesic constraints.

Finally, we encode a set of potentials that correspond to the preservation of local surface descriptors between the model mesh and data mesh. The use of local surface descriptors is important, because it helps to guide the optimization in the exponential space of assignments. We use spin images, which are two-dimensional histograms computed at an oriented point x on the surface mesh of an object. We apply principal components analysis to the spin images to produce a low-dimensional *signature* s_x of the local surface geometry around x. When data and model points correspond, we expect their local signatures to be similar. We thus introduce a potential which defines a zero-mean Gaussian penalty for discrepancies between s_{x_i} and s_{z_k} when $c_k = i$.

4.3 Optimization

In the previous section, we defined a Markov network, which encodes a joint probability distribution over the correspondence variables as a product of single and pairwise potentials. Our goal is to find a joint assignment to these variables that maximizes this probability. This problem is one of standard probabilistic inference over the Markov network. However, the Markov network is quite large, and contains a large number of loops, so that exact inference is computationally infeasible. We therefore apply an approximate inference method known as *loopy belief propagation (LBP)* (see, for example, [Yedidia et al. 2003]), which has been shown to work well in a wide variety of applications. LBP is a message passing algorithm over the variables in the Markov network. Roughly speaking, it maintains for each variable a probability distribution over its possible values. In each iteration, each variable sends its distribution to its neighbors — those variables to which it is directly connected via a probabilistic potential — and uses the distributions it receives to update its beliefs. Running LBP until convergence results in a set of probabilistic assignments to the different correspondence variables, which are locally consistent. We then simply extract the most likely assignment for each variable to obtain a correspondence.

One remaining complication arises from the form of our farness preservation constraints. Recall that we introduce a farness potential between every pair of points in the data mesh that are some minimal distance apart. In general, most pairs of points in the mesh are not close, so that the total number of such potentials grows as $O(M^2)$, where M is the number of points in the data mesh. (The total number of other potentials is $O(M)$). When the meshes are large, this quadratic growth can result in a high computational cost. However, these potentials are only intended to steer the algorithm away from bad correspondence assignments. Thus, we only need to activate them if such a correspondence arises. In particular, rather than introducing all these potentials into the Markov net from the start, we introduce them as needed. First, we run LBP without any farness preservation potentials. We then check if any farness preservation constraints are violated. If not, we are done. Otherwise, we add the violated constraints into the Markov net and rerun LBP. This process repeats until we converge to a valid solution. In practice, we only need to add a small number of farness preservation potentials until convergence occurs.

We note that the LBP algorithm is an approximate inference algorithm, and may encounter some difficulties. In certain cases, LBP may not converge, and when it does, there are no theoretical guarantees on the quality of the results. In particular, although the algorithm does find correspondences that are locally consistent, it may not produce the optimal alignment. We discuss this issue further in the next section.

5 Experimental Results

In this section, we show some results for the Correlated Correspondence algorithm. We first show that it successfully solves the surface registration problem, even for challenging data sets. We then show that the high-quality correspondences obtained by the algorithm enable us to provide completely unsupervised solutions to several different challenging graphics tasks.

5.1 Basic Registration

We applied our registration algorithm to meshes from two different datasets. In one data set, we used a range scanner to acquire a set of seven different complete surface meshes of a wooden puppet in different positions. Each mesh was composed from ten range scans taken from different viewing angles, and composed using the method of Curless and Levoy [1996]. Our second data set consisted of seven meshes of a human arm, first acquired and used by Allen *et al.* [2002]. These meshes are not complete surfaces, but several techniques exist for filling the holes (e.g., [Davis et al. 2002; Liepa 2003]).

We ran the Correlated Correspondence algorithm using precisely the same probabilistic model and the same parameters on both data sets. Our algorithm first runs on a coarse subsampling of the point locations in the original meshes. We then apply our algorithm to a finer-grained mesh, using the results of the first phase to constrain the possible correspondences of the points in the first sub-mesh to a general region. This coarse-to-fine approach helps the LBP algorithm avoid local minima, similarly to the traditional non-rigid ICP. The resulting mesh was then refined by running the non-rigid ICP algorithm of Hähnel *et al.* [2003], which, unlike our algorithm, allows the points on the data mesh to be mapped to points that are on the model surface, but not necessarily in the model mesh.

The Correlated Correspondence algorithm successfully aligned all pairs of meshes in the human arm data set. In the puppet data set the algorithm correctly registered four out of six data meshes to the model mesh. In the two remaining cases, the algorithm produced a registration where the torso was rotated, so that the front was mapped to the back. This problem arises from ambiguities induced by the symmetries of the puppet, whose front and back are almost identical. Importantly, however, our probabilistic model assigns a higher score to the correct solution, so that the incorrect registration is a consequence of local minima in the LBP algorithm.

This fact allows us to address this issue in an unsupervised way simply by running the algorithm several times, with different initialization. The initialization conditions were obtained by automatically partitioning the puppet data mesh into parts. Our algorithm looks for extremal points in the data mesh, and then extends the regions around the extremal points to be of a predefined size, which is set to be set to be a fraction of the total object size. Each part thus obtained is then aligned to several different places in the model mesh by using the Correlated Correspondence algorithm. We computed the probability for each part and each of its candidate alignments. We then selected the six non-overlapping part assignments whose total probability was highest. These alignments were used to initialize our algorithm by restricting the set of possible correspondences for the mesh points in the different parts, as dictated by the part-level alignment. We ran the algorithm for each of these six initializations, and selected the one which gave the highest score.

We ran this algorithm to register one puppet mesh to the remaining six meshes in the dataset, obtaining the correct registration in all cases. In particular, as shown in Fig. 4, we successfully deal with the case on which the straightforward nonrigid ICP algorithm failed. Note, however, that the results of the algorithm do contain a small artefact in the puppet's right shoulder. This artefact is a consequence of the large deformation in the right arm configuration between the two meshes. In this case, the correct registration of the arm cannot be determined from the data, and the algorithm makes an arbitrary decision, leading to the observed effect.

An unoptimized version of the Correlated Correspondence al-

Figure 6: Several frames from a motion animation generated by interpolating two scans of an arm (far left and far right), which were automatically registered using the Correlated Correspondence algorithm. The animation was produced by linear interpolation in link transformation space.

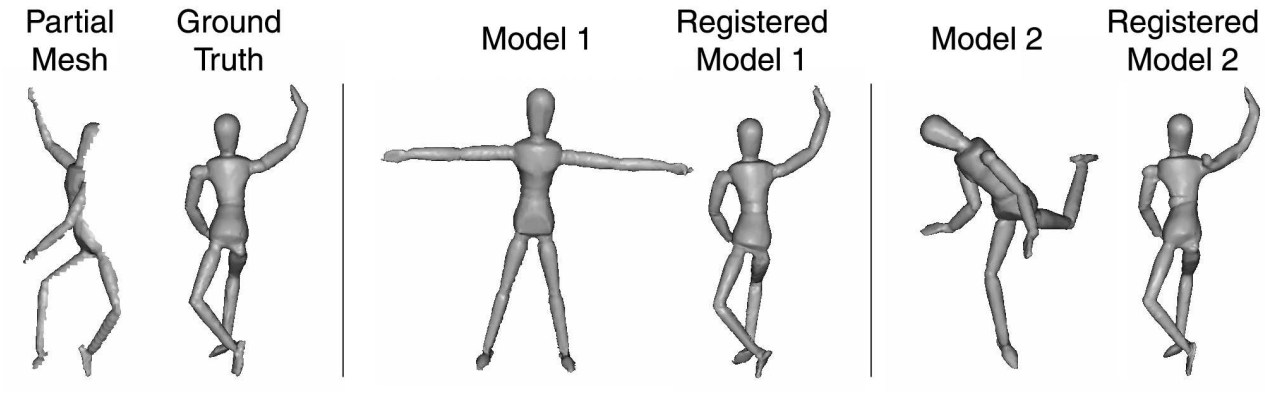

Figure 7: Partial mesh completion. (a) Partial view used as a data mesh. (b) Complete model from which (a) was taken, which serves as the ground truth. The complete model is oriented to display the hidden part of the surface. Registration of the model mesh (c) to (a) produces a completion, displayed in (d), which closely approximates the ground truth. Registration of model mesh (e) to (a) also produces a reasonable reconstruction, shown in (f). The reconstruction in (f) contains an artefact in the right shoulder, reseulting from an attempt to preserve the original geometry.

gorithm runs for 1.5 minutes on an Intel Xeon 2.4GHz processor to register a pair of arm meshes. This process includes all the pre-processing steps, including the mesh subsampling phase and the spin-image computation. The algorithm applied to the puppet data, which also involves the computation of the different part embeddings and the execution of the Correlated Correspondence algorithm for the different initialization points, takes a total of 10 minutes per puppet pair.

Overall, the algorithm performs robustly, producing a close-to-optimal registrations even for pairs of meshes that involve large deformations. It deals successfully both with transformations resulting from articulation, where entire parts undergo large motion transformations, and with non-rigid surface transformations. The registration is accomplished in an unsupervised way, without any prior knowledge about object shape, dynamics, or alignment.

5.2 Interpolation between Two Meshes

The task of interpolation between different object poses has been extensively studied in graphics and animation. For example, Allen *et al.* [2002] build a model for articulated upper torso deformations from range scan data. They obtain multiple scans of the arms and torso in different positions, and use prior knowledge in terms of a body skeletal structure and markers to build a model of the deformations. In general, there are well-known solutions to the interpolation problem in cases where the object's skeleton is known (e.g., [Chadwick et al. 1989; Wang and Phillips 2002]).

As we now show, our Correlated Correspondence algorithm can provide an alternative method for interpolation, which applies directly to meshes. It is therefore applicable even in cases where an object's articulation structure is unknown and in cases where the object is not articulated. Our approach uses the Correlated Correspondence algorithm to register two meshes, which recovers the non-rigid transformation Θ deforming the model mesh. The transformation Θ can be expressed in terms of local edge geometry, by using the local transformations of the mesh links, as opposed to movement of mesh points in Cartesian space. We can now interpolate linearly between the two meshes, where the interpolation is done *in the space of link transformations*.

Specifically, for each link $e_{i,j}$ in the model mesh, the Correlated Correspondence algorithm recovers the coordinate system rotations of the link endpoints, and the new link parameters $\tilde{e}_{i,j}$. Any intermediate mesh between the two can be obtained by linearly interpolating the local edge parameters. In particular, we interpolate the rotations of the link endpoints in Euler angle space, and we interpolate the directions $d_{i\to j}, d_{j\to i}$ and the lengths $l_{i,j}$. This form of interpolation tries to preserve both the link lengths and their local geometry, to the extent possible. Thus, links whose configuration in both meshes is unchanged will be unchanged throughout the interpolation.

The resulting linear interpolation, executed independently for each link, may not result in a consistent mesh. We therefore solve for a consistent mesh, which is closest (in squared distance) to the linearly interpolated model. Solving for a consistent mesh is equivalent to applying non-rigid ICP (see [Hähnel et al. 2003]) with a deformation prior specified by the linear interpolation defined above. The interpolation process tends to result in natural shapes, generating correct-looking anumation sequences, as shown in Fig. 1 and Fig. 6.

5.3 Partial View Completion

The Correlated Correspondence algorithm allows us to register a data mesh containing only a scan of part of an object to a known complete surface model of the object, which serves as a template. We can then transform the template mesh to the partial scan, a process which leaves undisturbed the links that are not involved in the partial mesh. The result is a mesh that matches the data on the ob-

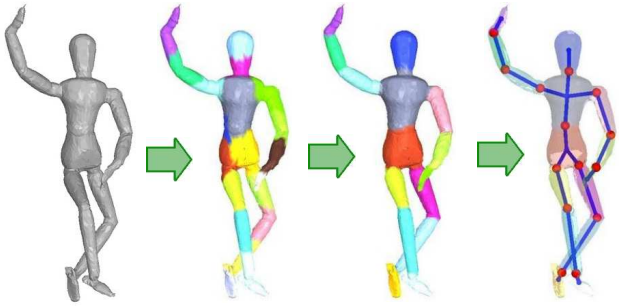

Figure 8: Illustration of the part-finding process: (a) A template mesh is registered to all other meshes by the Correlated Correspondence algorithm. (b) The mesh is randomly divided into small patches of approximately equal areas, different parts are color-coded. (c) Results in (b) are used to initialize an iterative algorithm which estimates the rigid parts and their transformations. (d) The joints linking the rigid parts are estimated.

Figure 9: Four different poses from the puppet dataset display the 15 rigid parts and the articulated skeleton, both of which are recovered automatically.

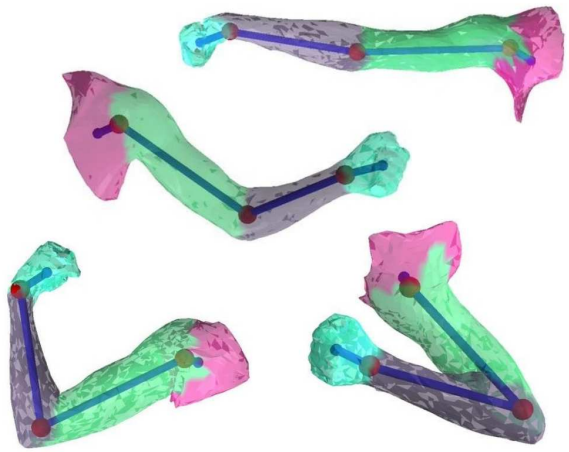

Figure 10: Four different poses from the arm dataset display four (approximately) rigid parts and the articulated skeleton, both of which are recovered automatically.

served points, while completing the unknown portion of the surface using the template.

We take a partial mesh, which is missing the entire back part of the puppet in a particular pose. The resulting partial model is displayed in Fig. 7(a); for comparison, the correct complete model in this configuration (which was not available to the algorithm), is shown in Fig. 7(b). We register the partial mesh to models of the object in two different poses (Fig. 7(c) and (e)), and compare the completions we obtain (Fig. 7(d) and (f)), to the ground truth represented in Fig. 7(b). The results demonstrate a largely correct reconstruction of the complete surface geometry from the partial scan and the deformed template.

The experiment also demonstrates the limitations of this approach. The completion method that we described leaves unchanged links that do not appear in the data mesh. In cases where the template is significantly different from the data configuration on parts that are not visible in the data, this assumption can lead to incorrect completions. For example, the configuration of the left arm in Fig. 7(e) changes significantly, leading to an artefact in ths in the right shoulder of Fig. 7(f). The reason for the artefact is that the shoulder links in the model prefer the original orientation, and no data is available to additionally constrain them.

5.4 Recovering Articulated Models

Articulated object models have a number of applications in animation and motion capture, and there has been work on recovering them automatically from 3D data [Cheung et al. 2003] and from feature tracking in video [Song et al. 2003].

We show that our unsupervised registration capability can greatly assist articulated model recovery from meshes corresponding to different configurations of an object. First, we register one mesh to all the remaining meshes of the object using the Correlated Correspondence algorithm. Subsequently, we perform *Expectation Maximization* by iterating between finding a decomposition of the object into rigid parts, and finding the location of the parts in the object instances. Finally, we use the recovered rigid parts and their transformations to automatically estimate the joints. The steps of the algorithm are visualized in Fig. 8. The part-finding algorithm [Anguelov et al. 2004] is a separate contribution from the registration work described in this report, and you can refer to `http://robotics.stanford.edu/~drago/Parts/uai2004.pdf` for additional details.

The Correlated Correspondence algorithm successfully registers all seven poses of the puppet, giving us information about the dynamics of the different points in the mesh. As a result of applying the algorithm for clustering the object surface into rigid parts, we automatically recover all 15 rigid parts of the puppet, as well as the joints between them. Several poses of the puppet, along with the recovered skeleton structure and position, are displayed in Fig. 9. To our knowledge, this is the first implementation that estimates such a complex skeleton from real world data with very few poses, in a completely unsupervised way.

In comparison, the algorithm of Cheung *et al.* [2003] is applied to sequences where only 2 parts move at a time, recovering an articulated human model with 9 parts by composing the results of the various sequences. Their approach is essentially a generalization of the ICP algorithm to multiple rigid parts. As we argued in Sec. 3.2 of this report, ICP is known to be prone to local minima. We hypothesize that the additional degrees of freedom provided by the possible part decompositions make the problem more severe, preventing them from dealing with multiple parts. Solving the registration problem with Correlated Correspondences constrains us enough to allow the use of global inference technique for rigid part clustering, which is robust even in the presence of multiple parts.

Our algorithm for recovering articulation works well even when

the object parts are not purely rigid, as is the case with the human arm. Even in this case, however, we get the intuitive articulated decomposition by using the meshes from our arm data set (see Fig. 10).

6 Conclusion

In this report, we describe an algorithm for unsupervised registration of non-rigid 3D surfaces in significantly different configurations. Our results show that the algorithm can deal with articulated objects subject to large joint movements, as well as with non-rigid surface deformations. The algorithm is not provided with markers or other cues regarding corresponence, and makes no assumptions about object shape, dynamics, or alignment.

We show that a solution to the registration problem can be used as a component in several applications. In our first application, we show that it allows a smooth interpolation between two different meshes of an object, in a way that tends to preserve the local geometry. We note that this interpolation process does not rely on the knowledge of even the existence of an underlying articulated skeleton. In our second application, we show that a partial data mesh (e.g., one arising from a single-view scan) can be registered to a complete model mesh, allowing the missing part of the data mesh to be completed using the model. Finally, we can use a set of scans of an articulate object in different configurations to determine its partition into parts.

The most important limitation of our approach is the fact that it makes the assumption of (approximate) preservation of geodesic distance. Although this assumption is a good heuristic in many cases, it is not always warranted. In some cases, the mesh topology may change, for example, when an arm touches the body. In these cases, our nearness preservation constraints are violated. In other cases, occlusions may eliminate paths in our data mesh, making nearby points appear geodesically distant, and violated our farness preservation constraints. We can try to extend our approach to handle these cases by trying to detect when they arise, and eliminating the associated constraints. However, even this solution is likely to fail on some cases.

A second limitation of our approach is that it assumes that the data mesh is a subset of the model mesh. Therefore, points on the data mesh are forced to correspond only to mesh points on the model surface, whereas a better correspondence may be with a non-mesh point. We address this problem by a post-processing phase that allows the 3-D coordinates of the transformed mesh points to be refined (e.g., using the algorithm of Hähnel *et al.* [2003]).

Third, as our results show, there are limitations to what can be achieved using only the raw mesh data. For example, when we have two meshes of the puppet where the arm configuration changes significantly, the algorithm cannot correctly determine the appropriate registration of mesh points on the shoulder, which can lead to artefacts (as shown in Fig. 4). Given only the raw data, the problem is under-specified, and this registration appears to be equally valid. To address situations such as this, we must give the algorithm more information, whether in the form of additional meshes, or in the form of prior knowledge. It would be interesting, for example, to incorporate into our probabilistic model an explicit notion of articulated parts. We can then weaken the link deformation penalties for links that cross joints, making deformations that correspond to part movement more likely. Given enough scans of the same object, we can even try to estimate the deformation penalty associated with different links, learning the extent to which they allow local deformations. Such an extension would be a step towards the goal of learning models of object shape and dynamics from raw data.

References

ALLEN, B., CURLESS, B., AND POPOVIC, Z. 2002. Articulated body deformation from range scan data. In *Proc. SIGGRAPH*.

ALLEN, B., CURLESS, B., AND POPOVIC, Z. 2003. The space of human body shapes:reconstruction and parameterization from range scans. In *Proc. SIGGRAPH*.

ANGUELOV, D., KOLLER, D., PANG, H., SRINIVASAN, P., AND THRUN, S. 2004. Recovering articulated object models from 3d range data. In *In Proc. UAI*.

B.CURLESS, AND M.LEVOY. 1996. A volumetric method of building complex models from range images. In *Proc. SIGGRAPH*.

BESL, P., AND MCKAY, N. 1992. A method for registration of 3d shapes. *Transactions on Pattern Analysis and Machine Intelligence 14*, 2, 239–256.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *Proc. SIGGRAPH*.

BLANZ, V., AND VETTER, T. 2002. Face recognition based on 3d shape estimation from single images. In *CG Technical Report No.2, University of Freiburg*.

CHADWICK, J. E., HAUMANN, D. R., AND PARENT, R. E. 1989. Layered construction for deformable animated characters. In *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, ACM Press, 243–252.

CHEN, Y., AND MEDIONI, G. 1991. Object modeling by registration of multiple range images. In *Proc. IEEE Conf. on Robotics and Automation*.

CHEUNG, K., BAKER, S., AND KANADE, T. 2003. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *Proc. IEEE CVPR*.

CHUI, H., AND RANGARAJAN, A. 2000. A new point matching algorithm for non-rigid registration. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*.

COUGHLAN, J., AND FERREIRA, S. 2002. Finding deformable shapes using loopy belief propagation. In *In Proc. ECCV*, vol. 3, 453–468.

DAVIS, J., MARSCHNER, S., GARR, M., AND LEVOY, M. 2002. Filling holes in complex surfaces using volumetric diffusion. In *Symposium on 3D Data Processing, Visualization, and Transmission*.

DAVIS, J., RAMAMOOTHI, R., AND RUSINKIEWICZ, S. 2003. Spacetime stereo : A unifying framework for depth from triangulation. In *Proc. CVPR*.

FELZENSZWALB, P. 2003. Representation and detection of shapes in images. In *PhD Thesis*, Massachusetts Institute of Technology.

FISCHLER, M. A., AND BOLLES, R. C. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Comm. of the ACM*, vol. 24, 381–395.

HÄHNEL, D., THRUN, S., AND BURGARD, W. 2003. An extension of the ICP algorithm for modeling nonrigid objects with mobile robots. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, IJCAI.

HUTTENLOCHER, D., AND FELZENSZWALB, P. 2003. Efficient matching of pictorial structures. In *Proc. CVPR*.

JOHNSON, A. 1997. *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

KRISHNAMURTHY, V., AND LEVOY, M. 2002. Fitting smooth surfaces to dense polygon meshes. In *Symposium on 3D Data Processing, Visualization, and Transmission*.

KRISHNAMURTHY, V. 2000. Fitting smooth surfaces to dense polygon meshes. In *PhD Thesis*, Massachusetts Institute of Technology.

LEVENTON, M. 2000. Statistic models in medical image analysis. In *PhD Thesis*, Massachusetts Institute of Technology.

LIEPA, P. 2003. Filling holes in meshes. In *Proc. of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, Eurographics Association, 200–205.

LIN, M. H. 1999. Tracking articulated objects in real-time range image sequences. In *ICCV (1)*, 648–653.

PEARL, J. 1988. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

RUIZ-CORREA, S., SHAPIRO, L., AND MEILA, M. 2001. A new signature-based method for efiicient 3-d object recognition. In *Proc. IEEE CVPR*, vol. 1.

RUSINKIEWICZ, S., AND LEVOY, M. 2001. Efficient variants of the ICP algorithm. In *Proc. Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, IEEEComputer Society, Quebec City, Canada.

SHELTON, C. 2000. Morphable surface models. In *International Journal of Computer Vision*.

SIGAL, L., ISARD, M., SIGELMAN, B. H., AND BLACK, M. J. 2003. Attractive people: Assembling loose-limbed models using non-parametric belief propagation. In *In Proc. NIPS*.

SONG, Y., GONCALVES, L., AND PERONA, P. 2003. Unsupervised learning of human motion. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

TAYCHER, L., FISHER, J. W., III, AND DARRELL, T. Recovering articulated model topology from observed motion.

WANG, X. C., AND PHILLIPS, C. 2002. Multi-weight enveloping: least-squares approximation techniques for skin animation. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ACM Press, 129–138.

YEDIDIA, J., FREEMAN, W., AND WEISS, Y. 2003. Understanding belief propagation and its generalizations. In *Exploring Artificial Intelligence in the New Millennium*, Science & Technology Books.

YU, S., GROSS, R., AND SHI, J. 2002. Concurrent object recognition and segmentation with graph partitioning. In *Proc. NIPS*.

ZHANG, D., AND HEBERT, M. 1997. Multi-scale classification of 3-d objects. In *Proc. CVPR*.