

Topologically-Robust 3D Shape Matching Based on Diffusion Geometry and Seed Growing

Avinash Sharma, Radu Horaud, Jan Cech, and Edmond Boyer
INRIA Grenoble Rhône-Alpes, Montbonnot Saint-Martin, France
{avinash.sharma, radu.horaud, jan.cech, edmond.boyer}@inria.fr

Abstract

3D Shape matching is an important problem in computer vision. One of the major difficulties in finding *dense* correspondences between 3D shapes is related to the topological discrepancies that often arise due to complex kinematic motions. In this paper we propose a shape matching method that is robust to such changes in topology. The algorithm starts from a sparse set of *seed* matches and outputs dense matching. We propose to use a shape descriptor based on properties of the heat-kernel and which provides an intrinsic scale-space representation. This descriptor incorporates (i) heat-flow from already matched points and (ii) self diffusion. At small scales the descriptor behaves locally and hence it is robust to global changes in topology. Therefore, it can be used to build a vertex-to-vertex matching score conditioned by an initial correspondence set. This score is then used to iteratively add new correspondences based on a novel *seed-growing* method that iteratively propagates the seed correspondences to nearby vertices. The matching is farther densified via an EM-like method that explores the congruency between the two shape embeddings. Our method is compared with two recently proposed algorithms and we show that we can deal with substantial topological differences between the two shapes.

1. Introduction

In the era of perpetually increasing computational capabilities, multi-camera acquisition systems have become popular to capture parameterization-free articulated 3D shapes. These systems allow marker-less shape acquisition and are useful for a wide range of applications in entertainment, sport, surveillance, interactive, and augmented reality systems. In the past decade, many 3D acquisition methods have been proposed which successfully provides a frame-wise reliable visual-hull or mesh representation for real 3D animation sequences [8, 22, 20, 17].

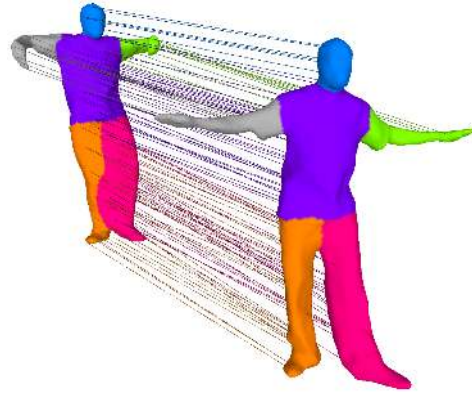


Figure 1. 3D shape matching in the presence of a topological merge. On the left shape the legs are completely merged together while on the right shape they are well separated. The colors emphasize the fact that our method correctly assigns the body parts, in particular the left/right legs are correctly matched even in the presence of a topological merge. (Only one percent of matches are shown for the purpose of visualization).

However, obtaining 3D animation sequences with spatio-temporal coherence, based on these independently reconstructed shapes, is a challenging task. It inherently involves estimation of *dense* 3D correspondences. This is particularly difficult in the case of articulated shapes due to complex kinematic poses. These poses induce self-occlusions and shadow effects which cause topological changes along the sequence, such as merging and splitting.

In this work, we propose a scale-space analysis of 3D shapes in order to achieve robustness with respect to topological changes. Starting with an initial set of sparse correspondences we propose to *grow* them in a small neighborhood defined by a scale parameter. Figure 1 shows a typical case of topological merging and the robust matching obtained with our method.

The main contribution of this paper is a dense 3D shape matching method that is robust to topological changes in the shape. The method starts from *sparse* one-to-one corre-

spendences and produces as output *dense* correspondences. We analyze a recently proposed shape-matching descriptor that is based on the heat-kernel matrix that may well be viewed as a *shape operator*. The eigenvalues and eigenvectors of this matrix/operator provide a scale-space *intrinsic* shape representation. The descriptor incorporates both (i) *heat flows* from the already matched points and (ii) *self diffusion*. At small scales this descriptor is fairly local and hence it is robust to changes in topology. It can therefore be used to build a matching score between a point on the first shape and a point of the second shape conditioned by the initial correspondences. This score is then used to iteratively add new point-to-point correspondences based on a novel *seed-growing* method that propagates current correspondences to nearby ones. The final set of dense correspondences is obtained via a point registration method that uses a variant of the EM algorithm.

There are two main classes of approaches that achieve dense 3D shape matching. The first class consists of model-based approaches which by-pass the problem of topological issues by starting with a prior shape model. This model is locally deformed at each time step of the sequence in order to obtain a globally consistent shape representation. This is achieved at the cost of losing detailed geometric and texture information obtained at each independent reconstruction. Another problem is the accumulation of deformation errors over time. Moreover, in the case of dynamic scenes, the assumption of a prior model is not realistic.

The second class consists of model-free approaches which do not impose any shape priors. Initially, sparse correspondences are computed between two independently reconstructed shapes using local cues based either on texture or on geometry. These correspondences are then propagated to obtain a dense shape matching. However, it is often the case that these initial sparse correspondences are not uniformly distributed over the shapes, and hence the propagation of these correspondences is a challenging task. It is even more difficult when the two shapes differ significantly. Indeed, one major limitation of this class of methods is that they use a geodesic distance onto the shape manifold, which is not robust to changes in topology.

The rest of the paper is structured as follows. In the next section, we discuss the related work. In section 3, we present a detailed outline of the proposed method. In section 4, we describe a multi-scale heat diffusion descriptor that is used in the novel seed growing algorithm introduced in section 5. In section 6, we outline an EM algorithm for the computation of dense correspondences. In section 7, we present experimental results that illustrate the robustness of our approach. Finally, section 8 concludes the paper with brief discussion and future directions of work.

2. Related Work

The problem of finding dense 3D shape matching has challenged many researchers in computer vision and computer animation. There exists a class of solutions based on the iterative closest point (ICP) method adapted to 3D shape matching [23, 24, 5]. These methods compute shape registration in the Euclidean space and hence do not provide a pose invariant matching.

Another class of methods relies on the assumption that articulated poses in the nonrigid shapes are isometric deformations. Hence, pose invariance is achieved by embedding the shape into an isometric subspace [21, 12, 4, 29, 25, 15, 16]. This is a very strong assumption for multi-camera acquisition systems, as the independently reconstructed shapes can be non-isometric due to presence of topological merges and splits.

Recent methods for shape tracking, that inherently performs dense shape matching employed initial sparse matches to achieve robustness to the topological issues [27, 10, 26]. In [26], a mesh evolution was performed by locally deforming the 3D shape according to the sparse 3D correspondences obtained by minimizing an error function evaluating the texture and geometric consistency. The consistency check was performed using the geodesic metric which is vulnerable to large shape deformations.

In [1], initial sparse correspondences were used to compute a set of harmonic functions and each shape vertex was represented by the coordinates of these functions. Then a dense matching was performed based upon computation of the level set of closest initial correspondences on two shapes. However, these harmonic functions are the solution of stationary heat equation and hence are globally affected by the topological issues. In a recent work, hierarchical assembly of independently reconstructed shapes was performed in [14] for computing a globally consistent space-time reconstruction.

The closest work to our method in terms of the shape feature descriptors is [13]. In their work, a dense shape matching method using a single (or multiple) initial correspondence is used. They propose a detailed theoretical justification for using heat diffusion maps. However, the proposed descriptor, when used in conjunction with a greedy matching method, is vulnerable to the topological issues. In our work, instead, we use a robust seed growing approach for matching which locally propagate sparse matches and is robust to outliers in the initial correspondences.

3. Approach Outline

Figure 2 sketches the pipeline of our approach. Given two 3D shapes, e.g, Figure 2.a (notice the difference in

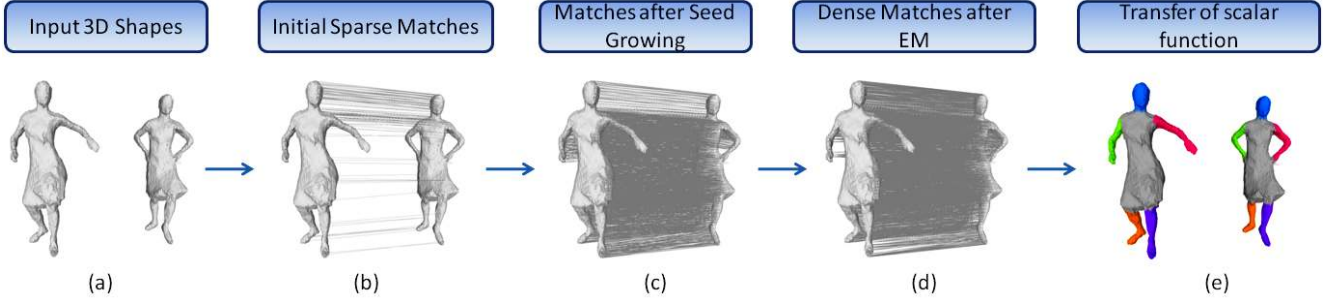


Figure 2. Method outline : (a) Input shapes (notice the difference in their topology). (b) Initial sparse matches. (c) Matches obtained with the seed-growing algorithm. (d) Probabilistic dense matches obtained with the EM algorithm. (e) Probabilistic transfer of a scalar function (coloring of shape parts).

their topology) we compute a set of initial sparse correspondences by matching features computed over any scalar function defined on the shape surface, e.g., local curvature, texture, color, etc. We call these initial sparse correspondences the *anchor correspondences*, e.g., figure 2.b. Then we describe a multi-scale heat kernel descriptor based on these anchors. Hence, a descriptor is associated with each shape vertex. A seed growing algorithm uses these descriptors to propagate initial correspondences over the shape, e.g., Figure 2.c. The final dense vertex-to-vertex registration result is obtained using an EM algorithm, e.g., Figure 2.d. In order to illustrate the quality of the final result, we use the matching to transfer a scalar function, e.g. color, texture, etc., from one shape to another, Figure 2.e.

4. Shape Description using Heat Diffusion

In this section we introduce heat diffusion over 3D shapes and analyze the heat-kernel descriptor which exploits the interesting property of capturing the local geometry of the surface. 3D shapes are described by meshes which are discretizations of closed Riemannian manifolds (compact and without boundaries). However, in this work we treat these meshes as undirected weighted graphs: Each vertex in this graph corresponds to a mesh vertex or, equivalently, to a 3D point on the shape. Each edge corresponds to a mesh edge. Graph edges are weighted by a similarity measure. Hence, each shape is a graph $\mathcal{M} = \{\mathcal{V}, \mathcal{E}, n\}$, where $\mathcal{V} = \{v_1, \dots, v_i, \dots, v_j, \dots, v_n\}$ is the set of vertices and \mathcal{E} is the set of edges. Each edge is weighted by $w_{ij} = \exp(-d_{ij}^2/\sigma)$, where d_{ij} denotes the Euclidean distance between the 3D points i and j associated with vertices v_i and v_j , and σ is a positive scalar parameter. Given two graphs \mathcal{M} and $\mathcal{M}' = \{\mathcal{V}', \mathcal{E}', n'\}$, the set of initial correspondences is denoted by $\mathcal{A} = \{a_1, \dots, a_m, \dots, a_M\}$ where each correspondence a_m is a vertex pair $\{v_{p_m}, v'_{q_m}\}$ with $v_{p_m} \in \mathcal{V}, v'_{q_m} \in \mathcal{V}'$.

4.1. Heat Diffusion on Graphs

Heat diffusion is a fundamental concept in physics. The heat diffusion equation is a partial differential equation which describes the distribution of heat (or the variation in temperature) in a given location and over time. Heat diffusion is generally studied on Euclidean spaces but it can be generalized to non-Euclidean ones such as manifolds. Heat diffusion on graphs is exactly the parallel of diffusion on closed Riemannian manifolds. The graph's heat operator or the *heat-kernel* matrix is [7]:

$$\mathbf{H}(t) = e^{-t\mathbf{L}}, \quad (1)$$

where $t > 0$ is a time parameter and \mathbf{L} is the combinatorial (or unnormalized) graph Laplacian. Each entry of this matrix is a Mercer kernel that has a very simple physical interpretation, as follows.

We consider real-valued functions f over \mathcal{V} , $f : \mathcal{V} \rightarrow \mathbb{R}$ and we note that $\mathbf{f} = (f_1 \dots f_n)^\top$ is simply a vector indexed by the vertices of \mathcal{M} . The vector $\mathbf{F}(t) = \mathbf{H}(t)\mathbf{f}$ is a solution to the heat-diffusion equation $(\partial/\partial t + \mathbf{L})\mathbf{F}(t) = 0$.

Hence, \mathbf{f} corresponds to some initial heat distribution over the vertices of \mathcal{M} and $\mathbf{F}(t)$ is the heat distribution at time t starting from $\mathbf{F}(0) = \mathbf{f}$. Notice that starting with a point heat distribution at vertex j , $\mathbf{g}_j = (0 \dots g_j \dots 0)^\top$, $g_j = 1$, the heat distribution at time t is given by the j^{th} column of the heat matrix which is denoted by $\mathbf{H}(\cdot, j; t)$:

$$\mathbf{G}(t) = \mathbf{H}(t)\mathbf{g}_j = \mathbf{H}(\cdot, j; t). \quad (2)$$

From (2) we obtain a straightforward interpretation of the entries of the heat matrix, namely each entry $h(i, j; t)$ of $\mathbf{H}(t)$ corresponds to the amount of heat available at vertex v_i at time t , starting with a point heat distribution at vertex v_j , i.e., \mathbf{g}_j :

$$h(i, j; t) = \mathbf{H}(i, j; t). \quad (3)$$

The symmetric function $h : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$ is the heat kernel of a graph \mathcal{M} . Each diagonal term $h(i, i; t)$ of the heat-kernel matrix has an interesting interpretation as well: it corresponds to the amount of heat remaining at i at time t . To conclude, the heat-kernel matrix encapsulates important *intrinsic* information about how heat “travels” from one part of the graph (or of the shape) to another part.

Let $\mathbf{L} = \mathbf{U}\mathbf{A}\mathbf{U}^\top$ be the spectral decomposition of \mathbf{L} with $\mathbf{U}\mathbf{U}^\top = \mathbf{I}$. Each column of \mathbf{U} , $\mathbf{u}_k = (u_{1k} \dots u_{ik} \dots u_{nk})^\top$ is an eigenvector and by omitting the zero eigenvalue and the associated eigenvector, we have the spectral decomposition of the Laplacian $\mathbf{L} = \sum_{k=2}^n \lambda_k \mathbf{u}_k \mathbf{u}_k^\top$, from which we easily obtain the spectral decomposition of the heat-kernel matrix:

$$\mathbf{H}(t) = \sum_{k=2}^n e^{-t\lambda_k} \mathbf{u}_k \mathbf{u}_k^\top. \quad (4)$$

Each entry $h(i, j; t)$ of this matrix is a *heat kernel*:

$$h(i, j; t) = \sum_{k=2}^n e^{-t\lambda_k} u_{ik} u_{jk} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \quad (5)$$

where

$$\mathbf{x}_i = [e^{-t\lambda_2/2} u_{i2}, \dots, e^{-t\lambda_n/2} u_{in}]^\top. \quad (6)$$

Hence, \mathbf{x}_i is an element of a *feature space*, or an *embedding* of the graph in \mathbb{R}^{n-1} . In practice one can use a reduced dimension $d \ll n - 1$. The heat-kernel can be used to define distances and norms in the feature space, namely:

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 = h(i, i; t) + h(j, j; t) - 2h(i, j; t) \quad (7)$$

$$\|\mathbf{x}_i\|^2 = h(i, i; t). \quad (8)$$

4.2. Heat Kernel Descriptor

Figure 3 depicts the heat diffusion phenomenon on 3D shapes. For small values of the time parameter t (which may well be viewed as scale parameter), the heat diffusion is limited to a local neighborhood (Figure 3.a), whereas for large values of t , the heat diffusion is global (Figure 3.b). This scale-dependent behavior of heat diffusion allows us to envisage a scale-space representations of 3D shapes. At small scales, diffusion behaves similarly across two topologically different shapes, while at larger scales, the behavior is affected by topological discrepancies between the shapes. This motivates the choice of a local descriptor based on heat diffusion at small scales.

We adapt the descriptor recently proposed in [13] to capture the local geometry by considering heat diffusion at small values of t . Each vertex on the shape is characterized by the amount of heat transferred from each of the

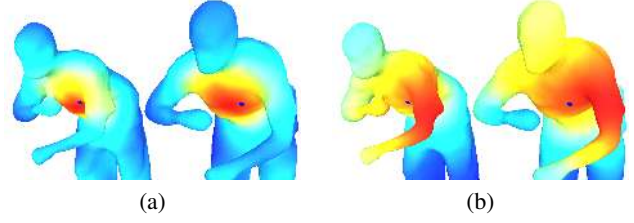


Figure 3. Visualization of heat diffusion on 3D shapes. The color at each vertex v_i encodes the value of the kernel $h(i, j; t)$, where v_j is the blue dot on the torso. (a): For small values of t , the heat diffusion map is very similar on both shapes and it is not affected by their topological differences (hand merging with body). (b): For large values of t , the behavior of the diffusion process drastically depends on the shape’s topology.

point heat sources placed at every anchor correspondence vertex in t time steps. We consider heat diffusion at the K times t_1, \dots, t_K . We suppose that M anchor correspondences are provided. For each anchor correspondence $a_m = (v_{p_m}, v'_{q_m})$, we compute two K -dimensional vectors:

$$\mathbf{d}_m^{v_i} = [h(i, p_m; t_1), \dots, h(i, p_m; t_K)]^T \quad \forall v_i \in \mathcal{V},$$

$$\mathbf{d}_m^{v'_j} = [h'(j, q_m; t_1), \dots, h'(j, q_m; t_K)]^T \quad \forall v'_j \in \mathcal{V}'.$$

Each coordinate of these vectors stores the amount of heat diffusion at times t_1, \dots, t_K . In addition to the heat diffusion from these M anchor correspondences, we also consider *self diffusion* at multiple times to capture the local geometry. This is encapsulated in an additional vector:

$$\mathbf{d}_{M+1}^{v_i} = [h(i, i; t_1), \dots, h(i, i; t_K)]^T \quad \forall v_i \in \mathcal{V},$$

$$\mathbf{d}_{M+1}^{v'_j} = [h'(j, j; t_1), \dots, h'(j, j; t_K)]^T \quad \forall v'_j \in \mathcal{V}'.$$

Finally, considering the heat diffusion from all the anchor correspondences and *self diffusion*, we define a $K \times (M + 1)$ descriptor matrix for each vertex $v_i \in \mathcal{V}$ on the shape \mathcal{M} as:

$$\mathbf{D}^{v_i} = [\mathbf{d}_1^{v_i} \dots \mathbf{d}_M^{v_i} \dots \mathbf{d}_{M+1}^{v_i}]. \quad (9)$$

Similarly, we can define a descriptor $\mathbf{D}^{v'_j}$ for each vertex $v'_j \in \mathcal{V}'$ from \mathcal{M}' . In practice we use $K = 5$ and set $\{t_1, \dots, t_5\} = \{0, 20, 40, 80, 100\}$. In some cases where the cardinality of two graphs are very different, we compute compatible time scales using the formulation presented in section 2.7 of [18].

Computation of Matching Score: The matching score between two descriptors is obtained as:

$$score(v_i, v'_j) = \|\mathbf{d}_{M+1}^{v_i} - \mathbf{d}_{M+1}^{v'_j}\|_2 + \sum_{m=1}^M \|\mathbf{d}_m^{v_i} - \mathbf{d}_m^{v'_j}\|_\infty. \quad (10)$$

Thus, the descriptors from two different shapes will have a low matching error score if the corresponding vertices are consistent with input anchor correspondences.

5. Correspondence Propagation

Once we have defined a multi-scale descriptor for every vertex on two shapes and a matching score, we can employ existing feature matching techniques that work either locally or globally and which exploit the neighborhood information. Therefore, a natural way to formulate the feature matching problem is to cast it as a graph matching problem. Finding a global optimum to the graph matching problem is an NP-hard problem. Nevertheless, several suboptimal solutions are available.

Energy minimization solutions based on Markov random fields (MRF) have been previously employed for 3D surface matching [21] but have a high computational cost. One can also cast the matching problem into an integer quadratic problem (IQP) that is equivalent to find a set of mutually compatible nodes in an association graph [2]. Other sub-optimal solutions use various possible relaxations, e.g., spectral relaxation [11]. These methods are tractable when the correspondence set is small, namely of the order of 10^2 and hence they are not suitable when one seeks dense matching.

Another interesting class of methods is seed-growing that starts with a set of initial matches and grows them in a local neighborhood. In our case, it is highly desirable to grow the seed matches locally, due to presence of topological issues and we already have a set of anchor correspondences which can be easily used as the seed matches. Hence, we propose a seed-growing algorithm similar in spirit to the one proposed in [6], for propagating anchor correspondences over 3D shapes.

Let's define a set of binary variables $\Gamma = \{\gamma_1, \dots, \gamma_i, \dots, \gamma_n\}$ for shape \mathcal{M} where γ_i is set to 1 if a vertex v_i from the first shape is assigned to a vertex v'_j from the second shape and 0 otherwise. Similarly, we can define Γ' for \mathcal{M}' . Let $\mathcal{S} = \{s_1, \dots, s_m, \dots, s_M\}$ be the set of initial seed matches, with $s_m = (v_i, v'_j)$, e.g., $\gamma_i = \gamma'_j = 1$. \mathcal{S} is stored in a priority queue data structure where each seed correspondence s_m is associated with a matching score. The matching score $score(s_m)$ is the heat descriptor matching error computed between the descriptors of v_i and v'_j (10). Each time an element is drawn from \mathcal{S} , it returns the seed with the minimum matching error score. Initially, the scores of all the seed correspondence are set to zero. This will ensure that all the seed matches are part of the output binary matching. We define a matching threshold α and consider only the correspondences with a matching score less than this threshold. We designate by

$Nei(v_i)$ the set of 2-ring neighbors of vertex v_i on the shape graph. The output of our algorithm is the set of accepted binary matches represented as $\Delta = \{\delta_{ij}\}$.

The proposed seed-growing algorithm proceeds as follows. We iteratively draw a seed match with the minimum matching error from \mathcal{S} and accept it as a correct binary match (add to set Δ) if the constituting vertices are not yet assigned to an existing match in Δ . Otherwise, we drop this seed correspondence from \mathcal{S} . Once a seed correspondence is accepted, the algorithm searches for all the neighboring vertices of this seed correspondence on each shape that are not yet assigned to any existing binary matches in Δ . It then computes the matching score between every pair of corresponding vertex descriptors and adds the current pair to priority queue structure \mathcal{S} if the matching score is less than α . Pseudo code of the seed-growing method is outlined in Algorithm 1.

Algorithm 1 Seed-Growing for Match Propagation

input : Two sets of vertex descriptors $\mathcal{D}, \mathcal{D}'$; seed matches \mathcal{S} ; a set of binary variables Γ, Γ' ; the matching score threshold α .

output : Dense matches $\Delta = \{\delta_{ij}\}$ where $\delta_{ij} = (v_i, v'_j)$.

- 1: **while** \mathcal{S} is not empty **do**
- 2: Draw the seed $s \in \mathcal{S}$ with the minimum matching error, $s = (v_i, v'_j)$.
- 3: **if** $\gamma_i = \gamma'_j = 0$ **and** $score(s) < \alpha$ **then**
- 4: $\Delta = \Delta \cup \{(v_i, v'_j)\}$ and set $\gamma_i = \gamma'_j = 1$.
- 5: **for each** $v_a \in Nei(v_i)$ **and** $\gamma_a = 0$ **do**
- 6: $v'_* = \underset{v'_k \in Nei(v'_j) \ \gamma'_{v'_k} = 0}{\operatorname{argmin}} \ score(v_a, v'_k)$.
- 7: **if** v'_* exists **and** $score(v_a, v'_*) < \alpha$ **then**
- 8: $\mathcal{S} = \mathcal{S} \cup \{(v_a, v'_*)\}$.
- 9: **end if**
- 10: **end for**
- 11: **end if**
- 12: **end while**

The proposed algorithm is robust to the initial outlier seeds as those seeds have a low score and will not be propagated due to the priority queue structure and provision of threshold. However, these outlier matches will still exist as part of output matches as we assign them a zero matching error score. Another problem is that the local heat diffusion properties will not be the same in the areas of topological merging/splitting. This will lead to unmatched set of vertices on two shapes. In the next section, we propose a dense probabilistic matching method to overcome these problems.

6. Dense Matching with EM

In this section we briefly describe a method that takes as input a sparse set of correspondences between the two shapes and provides as output a dense set of correspondences. The method is based on a parametric probabilistic model, namely maximum likelihood with missing data. Let us consider the Laplacian embedding of two shapes, i.e., (6) with $t = 0$: $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$, $\mathbf{X}' = \{\mathbf{x}'_j\}_{j=1}^{n'}$, with $\mathbf{X}, \mathbf{X}' \subset \mathbb{R}^k$, where $k \ll \min\{n, n'\}$ is the common dimension of the two embeddings, $3 \leq k \leq 10$ in our experiments. Without loss of generality, we assume that the points in the first set, \mathbf{X} are cluster centers of a Gaussian mixture model (GMM) with n clusters and an additional uniform component that accounts for outliers and unmatched data. The matching $\mathbf{X} \leftrightarrow \mathbf{X}'$ will consist in fitting the Gaussian mixture to the set \mathbf{X}' .

Let this Gaussian mixture undergo a $k \times k$ transformation \mathbf{Q} with $\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_k$, $\det(\mathbf{Q}) = \pm 1$, more precisely $\mathbf{Q} \in O(k)$, the group of orthogonal matrices acting on \mathbb{R}^k . Hence, each cluster in the mixture is parametrized by a prior p_i , a cluster mean $\boldsymbol{\mu}_i = \mathbf{Q}\mathbf{x}_i$, and a covariance matrix $\boldsymbol{\Sigma}_i$. It will be assumed that all the clusters in the mixture have the same priors, $\{p_i = \pi_{in}\}_{i=1}^n$, and the same isotropic covariance matrix, $\{\boldsymbol{\Sigma}_i = \sigma \mathbf{I}_k\}_{i=1}^n$. This parametrization leads to the following *observed-data log-likelihood* (with $\pi_{out} = 1 - n\pi_{in}$ and \mathcal{U} is the uniform distribution):

$$P(\mathbf{X}') = \sum_{j=1}^{n'} \log \left(\sum_{i=1}^n (\pi_{in} \mathcal{N}(\mathbf{x}'_j | \boldsymbol{\mu}_i, \sigma)) + \pi_{out} \mathcal{U} \right) \quad (11)$$

It is well known that the direct maximization of (11) is not tractable and it is more practical to maximize the *expected complete-data log-likelihood* using the EM algorithm, where “complete-data” refers to both the observed data (the points \mathbf{X}') and the missing data (the point-to-point assignments). In our case, this expectation writes (see [9] for details):

$$\mathcal{E}(\mathbf{Q}, \sigma) = -\frac{1}{2} \sum_{j=1}^{n'} \sum_{i=1}^n \alpha_{ji} (\|\mathbf{x}'_j - \mathbf{Q}\mathbf{x}_i\|^2 + k \log \sigma), \quad (12)$$

where α_{ji} denotes the posterior probability of an assignment: $\mathbf{x}'_j \leftrightarrow \mathbf{x}_i$:

$$\alpha_{ji} = \frac{\exp(-\|\mathbf{x}'_j - \mathbf{Q}\mathbf{x}_i\|^2/2\sigma)}{\sum_{q=1}^n \exp(-\|\mathbf{x}'_j - \mathbf{Q}\mathbf{x}_q\|^2/2\sigma) + \emptyset \sigma^{k/2}}, \quad (13)$$

where \emptyset is a constant term associated with the uniform distribution \mathcal{U} . Notice that one easily obtains the posterior probability of a data point to remain unmatched, $\alpha_{jn+1} = 1 - \sum_{i=1}^n \alpha_{ij}$. This leads to the dense matching procedure outlined in Algorithm 2.

Algorithm 2 Dense matching with EM

input : Two embedded shapes \mathbf{X} and \mathbf{X}' ;

output : Dense correspondences $\mathbf{X} \leftrightarrow \mathbf{X}'$ between the two shapes;

- 1: *Initialization*: Set $\mathbf{Q}^{(0)}$ and $\sigma^{(0)}$;
 - 2: *E-step*: Compute the posteriors $\alpha_{ij}^{(q)}$ using (13);
 - 3: *M-step*: Estimate the transformation $\mathbf{Q}^{(q)} = \arg \min_{\mathbf{Q}} \sum_{i,j} \alpha_{ij}^{(q)} \|\mathbf{x}'_j - \mathbf{Q}\mathbf{x}_i\|^2$ and the variance $\sigma^{(q)} = \sum_{i,j} \alpha_{ij}^{(q)} \|\mathbf{x}'_j - \mathbf{Q}^{(q)}\mathbf{x}_i\|^2 / k \sum_{i,j} \alpha_{ij}^{(q)}$
 - 4: *Assignment*: Match $\mathbf{x}'_j \leftrightarrow \mathbf{x}_i$ if $\max_i \alpha_{ij}^{(q)} > 0.5$.
-

However, the proposed EM algorithm can be easily trapped in a local minimum and the final result crucially depends on initialization. Hence, we use the dense binary matching obtained with seed propagation algorithm to initialize our probabilistic dense matching algorithm.

7. Results

The matching results are obtained with publicly available 3D shapes (meshes) captured using multi-camera systems, such as the *flashkick* sequence from University of Surrey [22] and the *samba* dance sequence from MIT [27]. In the latter case, we use a simple voxel carving algorithm to compute a visual hull represented as a mesh.

Matching results are presented in Figures 4 and 5. In Figure 4, we show *wide-time-frame* matching obtained in the presence of topological merging. For the purpose of visualization, we color code the body parts of one shape and we transfer the corresponding vertex-to-part labels to the other shape, as proposed in [19]. This makes use of the fact that EM outputs a posterior probability for each vertex of one shape to be matched with each vertex of the second shape.

	$ \mathcal{V} $	$ \mathcal{V}' $	$ \mathcal{A} $	$ \Delta $
<i>flashkick 016-017</i>	12006	12005	379	10191
<i>flashkick 019-020</i>	11864	12035	303	9836
<i>flashkick 024-025</i>	12231	12282	410	10214
<i>flashkick 113-117</i>	12582	12041	58	7197
<i>flashkick 230-236</i>	12240	11943	153	9596
<i>flashkick 117-130</i>	12041	12656	91	7118
<i>samba 042-048</i>	3881	4760	171	911
<i>samba 118-123</i>	4284	4226	167	2641
<i>samba 118-121</i>	4284	4254	200	2432
<i>samba 121-123</i>	4254	4226	200	2955

Table 1. Dataset details: Number of vertices of the input meshes ($|\mathcal{V}|$ and $|\mathcal{V}'|$), the initial number of anchor correspondences ($|\mathcal{A}|$), and the number of correspondences found with the seed-growing algorithm ($|\Delta|$).

Figure 6 shows the results of matching obtained with our method and with two other methods: with [12] (which is the best performing method in the recent shape matching benchmark [3]), and with [13]. The matching based on histograms of Laplacian eigenvectors is not reliable when the two shapes have different topologies and hence the dense matching method presented in [12] fails to provide good results, e.g., Figure 6.b. As mentioned earlier, greedy matching [13] does not consider the neighborhood consistency of matches and leads to wrongly matched patches on shapes, e.g., Figure 6.c.

Table 1, summarizes the mesh size ($n = |V|$ and $n' = |V'|$), the initial number of anchor correspondences $|\mathcal{A}|$ and the number of matches obtained with our seed-growing algorithm before applying EM.

Finding anchor correspondences. There are different methods proposed in the past that use local geometry/texture cues to find a set of sparse anchor correspondences [28, 1, 18], and any one of these methods can be used. In practice, we computed the sparse anchor correspondences using the method proposed in [1].

8. Conclusion

We proposed a dense shape matching method using a descriptor based on heat-diffusion on 2D manifolds, a seed-growing algorithm that locally propagates only the good matches and a variant of the EM algorithm that eventually registers the two shapes. The key feature of our method is that it considers an intrinsic scale-space representation based on the heat-kernel. This provides a principled framework for defining features at small scales and for robustly propagating correspondences locally in spite of topological changes. We have shown very good matching results on shapes in the presence of topological merging. Our method can be used to perform dense shape registration which stands at the basis of transferring any scalar function defined one shape, to the other one. In the future it will be interesting to employ local heat-kernel descriptors in a dynamic 3D environment where any prior assumption about shape topology is not valid. Additionally, automatic detection and correction of topological issues will be an important direction to explore.

Acknowledgments We thank Antoine Letouzey, Benjamin Petit and Simone Gasparini for fruitful discussions and providing feedback on our work.

References

- [1] N. Ahmed, C. Theobalt, C. Rössl, S. Thrun, and S. Hans-Peter. Dense correspondence finding for parameterization-free animation reconstruction from video. In *CVPR*, 2008.
- [2] R. C. Bolles and R. A. Cain. Recognizing and locating partially visible objects: The local-feature-focus method. *IJRR*, 1982.
- [3] A. Bronstein and *et. al.* Shrec 2010: Robust correspondence benchmark. In *Eurographics Workshop on 3D Object Retrieval*, 2010.
- [4] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Topology invariant similarity of nonrigid shapes. *IJCV*, 2009.
- [5] C. Cagniard, E. Boyer, and S. Ilic. Probabilistic deformable surface tracking from multiple videos. In *ECCV*, 2010.
- [6] J. Čech, J. Matas, and M. Perđoch. Efficient sequential correspondence selection by cosegmentation. *IEEE PAMI*, 2010.
- [7] F. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
- [8] J.-S. Franco and E. Boyer. Efficient Polyhedral Modeling from Silhouettes. *IEEE PAMI*, 2009.
- [9] R. Horaud, F. Forbes, M. Yguel, G. Dewaele, and J. Zhang. Rigid and articulated point registration with expectation conditional maximization. *IEEE PAMI*, 2011.
- [10] Q.-X. Huang, B. Adams, M. Wicke, and L. J. Guibas. Non-rigid registration under isometric deformations. In *SGP*, 2008.
- [11] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *ICCV*, 2005.
- [12] D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer. Articulated shape matching using Laplacian eigenfunctions and unsupervised point registration. In *CVPR*, 2008.
- [13] M. Ovsjanikov, Q. Merigot, F. Memoli, and L. Guibas. One point isometric matching with the heat kernel. In *SGP*, 2010.
- [14] T. Popa, I. South-Dickinson, D. Bradley, A. Sheffer, and W. Heidrich. Globally consistent space-time reconstruction. In *SGP*, 2010.
- [15] M. Reuter. Hierarchical shape segmentation and registration via topological features of Laplace-Beltrami eigenfunctions. *IJCV*, 2010.
- [16] Y. Sahillionglu and Y. Yemez. 3d shape correspondence by isometry-driven greedy optimization. In *CVPR*, 2010.
- [17] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006.
- [18] A. Sharma and R. Horaud. Shape matching based on diffusion embedding and on mutual isometric consistency. In *NORDIA*, 2010.
- [19] A. Sharma, E. von Lavante, and R. Horaud. Learning shape segmentation using constrained spectral clustering and probabilistic label transfer. In *ECCV*, 2010.
- [20] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer. A survey of methods for volumetric scene reconstruction from photographs. In *WVG*, 2001.

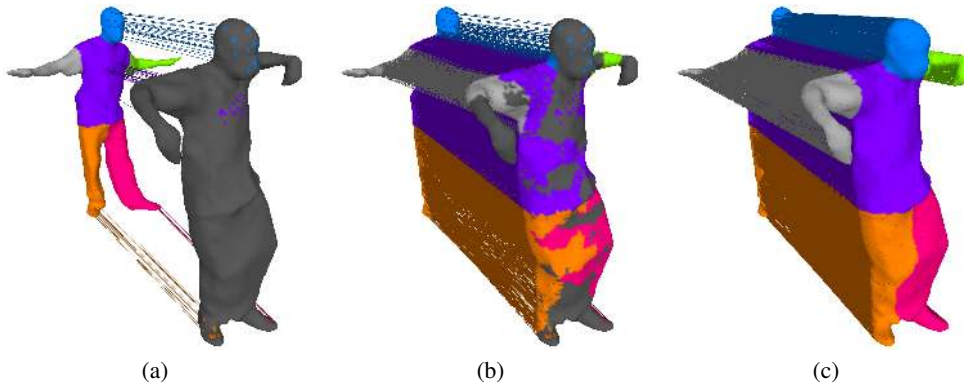


Figure 4. Dense shape matching: (a) Initial sparse matches, (b) Matches obtained with seed growing, (c) Final matching after EM.

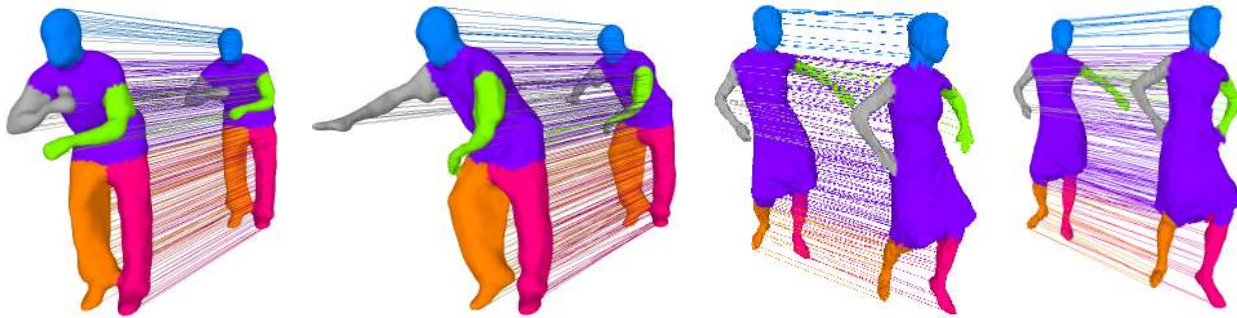


Figure 5. Shape matching results in the presence of topological issues. Only 1% of the total matches are actually shown.

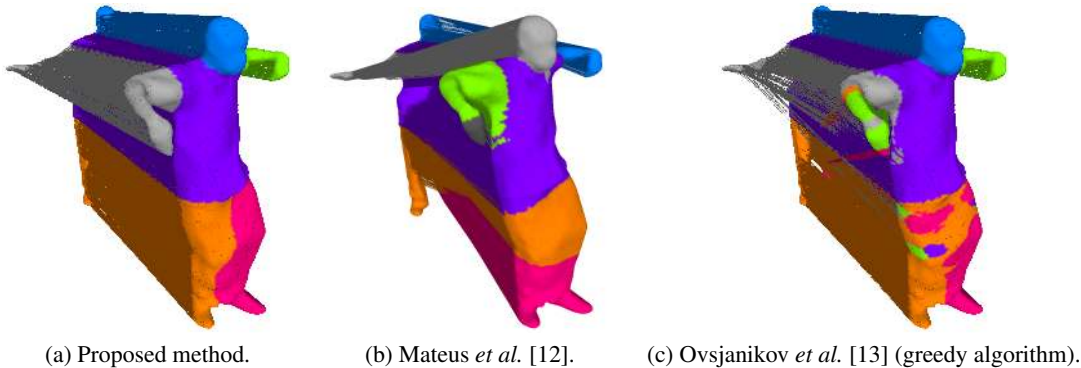


Figure 6. Comparison with two existing methods. The colors encode the body-part labels transferred from a segmented shape, i.e., Figure 1.

[21] J. Starck and A. Hilton. Correspondence labelling for wide-timeframe free-form surface matching. In *ICCV*, 2007.

[22] J. Starck and A. Hilton. Surface capture for performance based animation. *IEEE Computer Graphics and Applications*, 2007.

[23] J. Sussmuth, M. Winter, and G. Greiner. Reconstructing animated meshes from time varying point clouds. In *SGP*, 2008.

[24] A. Tevs, M. Bokeloh, M. Wand, A. Schilling, and H.-P. Seidel. Isometric registration of ambiguous partial data. In *CVPR*, 2009.

[25] T. Tung and T. Matsuyama. Dynamic surface matching by geodesic mapping for 3d animation transfer. In *CVPR*, 2010.

[26] K. Varanasi, A. Zaharescu, E. Boyer, and R. Horaud. Temporal surface tracking using mesh evolution. In *ECCV*, 2008.

[27] D. Vlastic, I. Baran, W. Matusik, and J. Popovi. Articulated mesh animation from multi-view silhouettes. In *SIGGRAPH*, 2008.

[28] A. Zaharescu, E. Boyer, K. Varanasi, and R. P. Horaud. Surface feature detection and description with applications to mesh matching. In *CVPR*, 2009.

[29] Y. Zeng, C. Wang, Y. Wang, D. Samaras, X. Gu, and N. Paragios. Dense non-rigid registration using high order graph matching. In *CVPR*, 2010.