Pacific Graphics 2016
E. Grinspun, B. Bickel, and Y. Dobashi
(Guest Editors)

Volume 35 (2016), Number 7

Spatial Matching of Animated Meshes

H. Seo[1] and F. Cordier[2,1]

[1]ICube, University of Strasbourg, France
[2]LMIA, University of Haute-Alsace, France

Abstract

This paper presents a new technique which makes use of deformation and motion properties between animated meshes for finding their spatial correspondences. Given a pair of animated meshes exhibiting a semantically similar motion, we compute a sparse set of feature points on each mesh and compute spatial correspondences among them so that points with similar motion behavior are put in correspondence. At the core of our technique is our new, dynamic feature descriptor named AnimHOG, which encodes local deformation characteristics. AnimHOG is ob-tained by computing the gradient of a scalar field inside the spatiotemporal neighborhood of a point of interest, where the scalar values are obtained from the deformation characteristic associated with each vertex and at each frame. The final matching has been formulated as a discreet optimization problem that finds the matching of each feature point on the source mesh so that the descriptor similarity between the corresponding feature pairs as well as compatibility and consistency as measured across the pairs of correspondences are maximized. Consequently, reliable correspondences can be found even among the meshes of very different shape, as long as their motions are similar. We demonstrate the performance of our technique by showing the good quality of matching results we obtained on a number of animated mesh pairs.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems; I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Shape matching constitutes a key task in higher level algorithms, such as attribute transfer, similarity based shape retrieval, and statistical shape modeling. However, robust and efficient matching still remains a challenge sometimes, we need to go beyond dealing with large variations between individual shapes and understand the semantics or functionalities of the shape, in order to infer meaningful matching.

While there is a large amount of research done on static surfaces with a proliferation of algorithms and a solid theoretical background, this has rarely been the case for dynamic, time-varying dataset. Indeed, it is only recently that techniques enabling densely sampled dynamic shape capture start to appear [MFO*07] [WAO*09] [WLVGP09]. Dealing with the large amount of noisy geometric information produced by high-speed shape acquisition devices, these works often deal with the problem of intra-subject correspondence, i.e. finding the correspondences between consecutive frames or the motion of the moving object across the frames.

Despite the increasing availability and relevance of time-varying shapes, current inter-subject correspondence techniques handle mostly the geometric features of static dataset [ZBVH09] [OMMG10]. Although the use of geometric feature is still a golden standard, it is quite obvious that it may guide to correspondence computations with limited reliability. Indeed, commonly observed subjects like human body are highly mobile and drastically change their spatial arrangement, thus geometric properties. In this paper, we investigate a new shape matching technique that exploits rich set of motion information exhibited on animated meshes. Our key idea is to use kinematic properties of animated surfaces so as to place parts with similar deformation behaviors in correspondence. By the deforming shape of a subject contains much more information than a single static posture of it, our approach allows estimating robust, reliable, and sensible matching. Given a pair of meshes exhibiting semantically same motions with moderate inter-subject variation, our goal is to find reliable correspondences among them by making use of their functional (i.e. motional) properties. We first extract a set of dynamic feature points, based on deformation characteristics that can effectively encode the local movement on the animated mesh surface. We then compute dynamic feature descriptor named AnimHOG (Animated Mesh Histogram-of-Gradients) on each dynamic feature point. Finally, we employ a graph-based discrete optimization paradigm for establishing reliable feature correspondences between on the two animated meshes. We demonstrate robustness and effectiveness of dynamic feature matching on a number of examples of varying subjects, movements, complexities and details.

This paper makes the following contributions:

- We present the first method that examines motion properties of time-varying shapes in spatial matching.
- We introduce a new dynamic point descriptor which encodes the local deformation behavior of an ani-mated surface.
- We adopt a feature correspondence algorithm based on graph matching [TKR13] to efficiently and robustly compute the spatial matching.

1.1. Input animation data

An animated mesh is defined as an artifact-free surface changing its shape over time, with a fixed topology. We assume that we are given a pair of animated meshes exhibiting semantic similarity in their motions. Formally, let a source animated mesh $\mathcal{M} = \{\mathcal{M}_f \,|\, f = 1 \ldots M\}$ be defined at each frame $\mathcal{M}_f = (V_f, T)$, represented as a set of vertices $V_f = \left\{v_f^i \in \mathbb{R}^3 | i = 1..N\right\}$ sharing the same connectivity T. The respective vertices v_i across the frames are assumed to be in correspondence. Similarly, a target animated mesh $\mathcal{M}' = \left\{\mathcal{M}_f^{'} | f = 1..M'\right\}$ is defined with $\mathcal{M}_f^{'} = \left(V_f^{'}, T'\right)$. Typically, our dataset originates either from animation softwares or optical sensor devices [LWP10] [CWZ*14]. Some examples of animated meshes used in our work are shown in Section 6.3.

1.2. Overview

Given a pair of animated meshes that exhibit a similar motion, we want to find the reliable correspondences that make use of motion properties on the surface. We start by detecting spatiotemporal feature points in the scale space representation of the deformation characteristic, which represents the local changes in stretching and bending at each point, at each frame (Section 3). Additional feature points are detected at end-effector locations (Section 5), and added to the feature point set. Then in Section 4 we describe our AnimHOG descriptor. Armed with the feature descriptor along with some energy terms to ensure the consistency across the correspondences, we compute the correspondences between the two sets of features (Section 6) by adopting a binary graph labeling technique. Our experimental results show that our method is able to compute reliable matching on a range of different animated meshes of varying subjects, movements, and complexities (Section 6.3).

2. Related work

While computing the correspondences for static meshes is an active area of research, there is almost no existing work that computes correspondences between animated meshes. We review correspondence computing methods based on feature descriptors. Also reviewed are spatio-temporal feature descriptors for animated surfaces or video.

Matching methods. With a few of exceptions, the majority of existing methods perform their matchings on pre-selected feature points on the given surfaces, whose similarities are compared in the form of descriptor distances. As the pairwise similarity alone is not sufficient to condition the combinatorial problem, additional constraints have been developed in many of robust correspondences. One often used constraint is the approximately isometric deformation (where geodesic distances are nearly invariant) between the source and the target, which can be observed in many real-world deformation data (such as the surface of articulated subject). Many of robust correspondences have been demonstrated using this constraint, including methods based on clustering in the transformation space [CZ08], embedding in the bending-invariant representation [JZvK07], random sampling [TBW*09] [TBW*11], efficient pruning of search-tree [FS06] [ZSCO*08], and graph matching [MCS13]. When the problem involves surfaces with large, non-rigid shape variations, the isometric deformation condition is relaxed to minimum distortion or deformation energy. Existing methods usually make use of the structural information assuming the structural similarity between shapes, and aim at maximum collective consistency as measured by using the spatial arrangement of the feature points. Examples include spectral clustering method that maximizes the inter-cluster compatibility (in 2D) [LH05], or embedding in the low-dimensional space [LF09] [KLF11]. We refer to the article by Van Kaick et al [[vKZHC10] for an extensive survey on correspondence techniques for surface.

Local shape descriptors. Local shape descriptor is one of the most fundamental problems in surface matching. Many existing methods are based on geometric properties. Spin image [Joh97] and its scale-invariant extension [DK12] encode the local surface properties in the form of 2D histograms accumulating points by spinning around the normal axis. Although not bending invariant, spin image has been successfully adopted in correspondence computation between poses [ASP*04] [CZ08]. Geodesic fan proposed by Zelinka et al. [ZG04] consists of a set of spokes and a set of samples on each spoke. Samples equally spaced along each spoke form a local geodesic polar map around a vertex. Gatzke et al. [GGGZ05] propose the curvature map that associates each ring neighborhood of a given point with the corresponding average Gaussian or average mean curvature over the ring. They extend the curvature map to 2D, where each spoke of the fan is associated with Gaussian/mean curvature as a function of distance from the point. Gelfand et al. [GMGP05] takes the intersection of a ball centered at the given point with the interior of the surface. The resulting integral volume descriptor is shown to be closely related to the mean curvature, but much more robust to the noise. The HMM-based statistical descriptor that appears in the work of Castellani et al [CCFM08] sample a number of existing local point descriptors along the spiral path on the surface, around a given point. The repeatedly occurring entries have been modeled as a stochastic process by a discrete HMM. The recently developed Heat Kernel Signature (HKS) and its variants [SOG09] [TBW*11] are related to the diffusion of heat across the surface over time. HKS provides a multi-scale representation of the surface and inherently invariant to isometric deformations and thus can be applied for the matching of shapes with isometric deformations [DLL*10] [OMMG10]. Some of the shape descriptors have been inspired by the work carried out in image processing. For instance, Local Depth SIFT (LD-SIFT) [DK12] and [MBO10] propose to extend the Scale-Invariant Feature Transform (SIFT) [Low04] to compute point signatures for surfaces. Similarly, Zaharescu et al. [ZBVH09] have developed

MeshHOG derived from the well-known Histogram of Oriented Gradients (HOG), a feature descriptor developed in computer vision [DT05]. The HoG The 2D shape contexts introduced by Belongie et al. [BMP00] has also been extended to signatures for 3D shapes [KPNK03].

Spatio-temporal feature descriptors. To the best of our knowledge the problem of computing point signature on animated meshes has not been addressed in the context of matching. However, there are several indirectly related works in animated mesh processing. A few works use vertex trajectories as per-vertex descriptor [SSK05] [DATTS08] or vertex displacement signature [AKH*10] [ACH*13] in their animated mesh segmentation algorithms. Others [LWC06] rely on per-triangle deformation gradients computed over subsequent frames of animation. The deformation gradients are computed in each frame, which essentially encode rotation and scale/shear components of the local surface deformation. In animated mesh compression, Mamou et al. [MZP06] use vertex Affine transformations in homogeneous coordinates in order to capture the surface motion characteristic. More recently, Tung and Matsuyama have proposed timing-based descriptors for dynamic surface [TM12] [TM14]. Their timing-based local descriptor is computed from surface intrinsic property variation dynamics, which are modeled using hybrid linear dynamical systems (HDS). Their work distinguishes itself from dynamic event modeling works in dynamic texture [DCWS03] [RCV09], which rely on dynamical system state parameters to characterize dynamic events (of dynamic texture). Also related to our approach is a variety of descriptors from image processing and computer vision. In particular, many descriptors for dynamic event modeling in video sequences are based on orientations of spatio-temporal 3D gradients. Examples include cuboid descriptor [DRCB05], 3D SIFT [SAS07], HOG3D [KMS08], and 3D SURF [WTG08]. Dollár et al. [DRCB05] suggest to transform pixels inside a feature cuboid (spatio-temporal neighborhood of a feature point) into a number of modalities, on which gradient vectors are computed and binned into a feature vector descriptor. Scovanner et al. [SAS07] extends 2D SIFT by adding the temporal dimension; the 3D SIFT essentially describes a histogram, constructed by binning the gradient orientations in the spatio-temporal neighbourhood of a feature point. The HOG3D [KMS08] employs regular polyhedrons instead of polar coordinate for the orientation quantization, so as to avoid problems of singularities at the poles. The 3D SURF [WTG08], an extention of the Speed Up Robust Features (SURF) [BETVG08] image feature descriptor, describes a point of interest by accumulating into vector the responses of axis-aligned Haar-wavelets for each sub-volume of a rectangular volume around it.

3. Deformation feature points

At the first stage, a set of spatio-temporal feature points are extracted from each animated mesh, based on a previous work [MSC15]. The deformation characteristic is first computed for each vertex $\mathbf{p}$ at each frame f, which measures the amount of non-rigid deformation with respect to the reference pose, the rest shape of the mesh before deformation. More specifically, a weighted summation of the curvature change and maximum principal strain defines the deformation characteristic: $\mathrm{d}\left(\mathbf{p}^f\right) = \alpha \cdot \mathrm{c}\left(\mathbf{p}^f\right) + (1-\alpha) \cdot s\left(\mathbf{p}^f\right)$, α a coefficient to determine the relative contribution of strain $s(.)$ and curvature $c(.)$, has been set to 0.5 in all our experiments. Algorithm details on the strain- and curvature-computations are provided in Appendix A.

Next, a space-time scale space as well as a Difference-of-Gaussian (DoG) pyramid [Lin98] is constructed based on the deformation characteristic values defined for each vertex at each frame. The feature points are then extracted by identifying local maxima in the spatiotemporal neighborhood and in the (scale-)normalized, space-time DoG (Difference-of-Gaussian) pyramid. Associated with each spatiotemporal feature point are a frame at which the vertex has been detected, and the two (temporal- and spatial-) scale values. For further details we redirect the reader to the original article [MSC15].

With our assumption that the animated meshes exhibit semantically same motions with moderate variation, the spatiotemporal feature sets on the source and the target animated meshes are presumably consistent to a certain degree, though computed independently. This will allow us to perform reliable matching on them. In practice, however, the feature sets extracted from the two given meshes can be noticeably different, due to the motion variation, approximation error, or inherent noise. In Section 6, we describe our correspondence technique which deals with inconsistent feature point sets in general.

4. AnimHOG

At the core of our animated mesh matching is our new dynamic feature descriptor, AnimHOG (Histogram-of-Gradients for animated mesh). With its primary purpose being useful in reliable correspondence based on the motion semantics, it is important to make the descriptor invariant to absolute degrees of deformation, and account for the relative characteristics in the local spatiotemporal neighborhood. To this end, we define our descriptor so as to characterize the dynamic property of a feature point in its local spatiotemporal neighborhood. As the name suggests, AnimHOG is based on the histogram of oriented gradients of the deformation characteristic. The use of histogram of gradient has been around for a while, which has been successfully adopted for static meshes, namely MeshHoG [ZBVH09] and 3D SIFT [SAS07]. However, we believe that our descriptor has a valuable aspect, since it allows for use with animated meshes. The computation of AnimHoG proceeds in a number of steps, which we describe as below.

4.1. Flattening of the spatial neighborhood

For a feature point $\mathbf{p}$ we identify its *spatial neighborhood* $N_s^k(\mathbf{p})$ by taking its k neighbor-rings with k proportional to the characteristic scale of $\mathbf{p}$ in space domain (Figure 1(a)). We then flatten the surface patch $N_s^k(\mathbf{p})$ in a quasi-isometric way (Figure 1(b)) by using the isomap [TdSL00]. Note that in certain cases, such as the spatial neighborhood with a large value of k on a cylindrical surface, we may obtain a patch which cannot be flattened without a self-overlap. In order to cope with such case, we retain from $N_s^k(\mathbf{p})$ only those neighboring vertices whose normal vectors are no more than $\frac{\pi}{2}$ apart from the normal vector $\mathbf{n}_\mathrm{p}$ at $\mathbf{p}$ (see Figure 2), i.e. the front-facing spatial neighborhood:

$$\tilde{N}_s^k(\mathbf{p}) = \{v \in N_s^k(\mathrm{p}) \,|\, \angle(\mathbf{n}_v, \mathbf{n}_\mathrm{p}) < \frac{\pi}{2}\}$$

where $\angle()$ denotes the angle between two vectors.

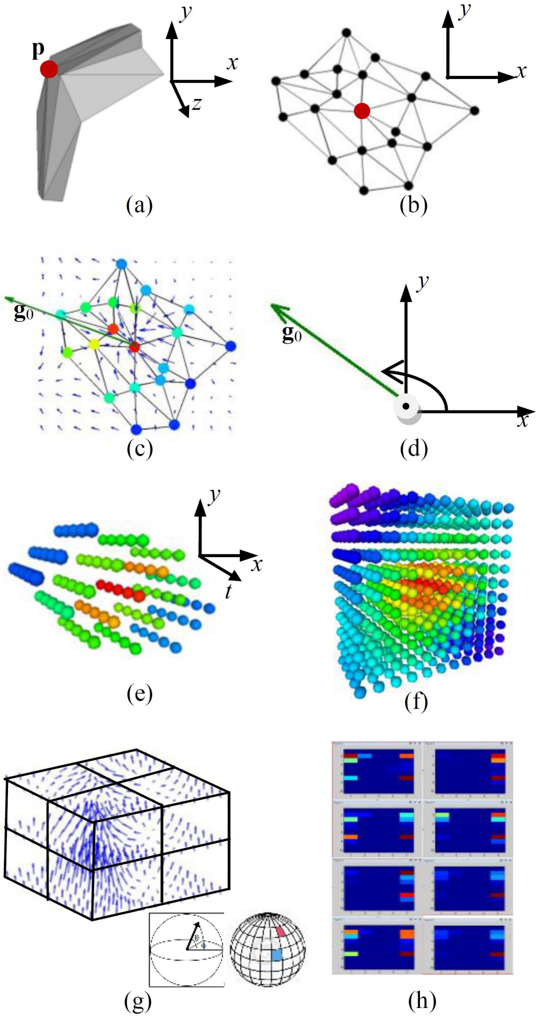


Figure 1: *The steps of AnimHoG computation. (a) The patch (the neighborhood proportional to the spatial scale) of a feature point is identified. (b) The patch is flattened to a 2D plane. (c) The dominant direction $\mathbf{g}_0$ of local gradient directions is estimated. (d)The x-axis of the patch is aligned with $\mathbf{g}_0$. (e) Spatiotemporal volume is defined proportionally to the feature's characteristic scales. (f) Densely interpolated scalar values. (g) Subdivision of the volume into octets and (h) 2D histograms for the octet are computed by binning gradient directions in the spherical coordinate system.*

4.2. Gradient vectors computation

In the next step, we compute the gradient vector field of the deformation characteristic (Figure 1(c)) on the plane where the flattened patch resides. For the sake of rotation-invariance, we define a coordinate system by using the dominant gradient direction $\mathbf{g}_0$. After binning into a polar histogram all gradient vectors according to their orientations, the direction corresponding to the bin of global maximum is taken as $\mathbf{g}_0$. Then we define a new coordinate system by first translating $\mathbf{p}$ to the origin point, and by aligning its x-axis with $\mathbf{g}_0$ (Figure 1(d)).

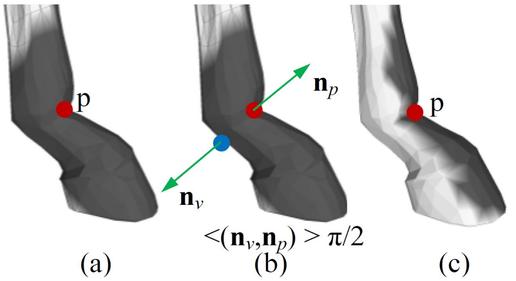


Figure 2: *(a) Spatial neighborhood $N_s^k(\mathbf{p})$ of p is shown in the dark color. By considering the angle between the surface normals (b), we obtain $N_s^k(\mathbf{p})$, a front-facing spatial neighborhood (c).*

4.3. Volume construction with temporal neighborhood

Next, we define a temporal neighborhood $N_t(\mathbf{p})$ of $\mathbf{p}$ proportional to its temporal characteristic scale τ. Given the time t^* (in frame number) at which $\mathbf{p}$ was detected as feature, $N_t(\mathbf{p})$ is defined as $N_t(\mathbf{p}) = \{t^* + \iota\Delta t | \iota \in [-\tau,..,\tau]\}$, where Δt is a constant proportional to the frame rate of the animation. To handle the case where $\mathbf{p}$ is detected either at the beginning or the end of the animation sequence, boundary conditions are added:

$$N_t(p) = \begin{cases} t^* + \iota\Delta t \ : \iota \in [0,..,\tau], \text{if } t^* < \tau \\ t^* + \iota\Delta t \ : \iota \in [-\tau,..,|M-t^*|], \text{ if } |M-t^*| < \tau \end{cases}$$

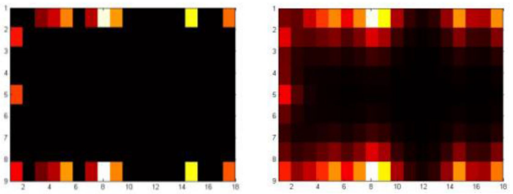
The combination of spatial and temporal neighborhoods defines the *characteristic volume* $V(\mathbf{p})$:

$$V(\mathbf{p}) = \{(\mathbf{x}, t) \,|\, \mathbf{x} \in \tilde{N}_s^k(\mathbf{p}), t \in N_t(\mathbf{p})\}$$

Intuitively, $V(\mathbf{p})$ can be interpreted as a stack of flattened patches with fixed spatial coordinates, within the given time interval. The time axis t is chosen according to the right-hand rule with respect to x- and y-axes (Figure 1(e)).

4.4. Histogram of gradients

Given the characteristic volume $V(\mathbf{p})$ with scalar values of deformation characteristics inside, we proceed with the numerical computation [FB13] of gradients (Figure 1(g)). Often, the deformation characteristic values are given only at a relatively small number of scattered locations (Figure 1(e)) inside the characteristic volume, which is undesirable. To make the scalar field appropriate for the gradient computation, we densely and regularly sub-sampling the volume, and interpolate deformation characteristic values at every sub-sampled location. For the interpolation we use RBF (Radial Basis Function) with Gaussian function. Once we obtain a dense, regular sampling of deformation characteristic values in $V(\mathbf{p})$), we compute the AnimHoG descriptor at $\mathbf{p}$ by computing orientation histograms $H_p = \{H_p^i | i = 1,..,8\}$ in each of its octet (Figure 1(g-h)). Each of H_p^i is essentially a 2d histogram computed by binning the gradient directions in a spherical coordinate system (φ,θ) with $\varphi \in [0,2\pi]$ and $\theta \in [0,\pi]$. The number of bins is a configurable parameter, which we set as $\varphi = 8$, $\theta = 4$ in our experiments. Similarly to 3D SIFT [SAS07], we further normalize the values added to each radial bin in H_p by the area of corresponding bin's solid angle Ω. Solid angle normalization compensates the fact that the bins near sphere equator tend to be significantly larger than the bins around sphere poles. Finally, in order to improve the anti-aliasing in the histograms, we applied the fast 2d histogram smoothing technique used in [EG04] (Figure 3). We found that reducing aliasing in the histograms helps to make the signature more robust during the matching stage.



(a) (b)

Figure 3: *(a) Original histograms from the Cylinder's feature point; (b) Smoothed histograms.*

Distance metric. Given two AnimHOG signatures $H_p = \{H_p^i | i = 1,..,8\}$ and $H_{p'} = \{H_{p'}^i | i = 1,..,8\}$, the distance between them is defined as follows:

$$D_{H^{\mathbf{p}}}\left(\mathbf{p},\mathbf{p}'\right) = \sum_{i=1}^{8} \left\| H_p^i - H_{p'}^i \right\|_{\mathrm{H}} \quad (1)$$

where $\|.\|_{\mathrm{H}}$ is the histogram earth moverŠs distance also known as Wasserstein metric [LO07].

5. Displacement feature points

Depending on the motion presented by the animated mesh, the number of deformation feature points extracted in Section 3 can be very small. For instance, bending cylinders shown in Figure 5 have only two or three feature points detected at the joints. When we want to extend the sparse feature correspondence to a full matching, it is definitely advantageous to have more feature points with distinctive features, in the light of more reliable correspondence. To this end, we extend the feature point set by extracting additional feature points in highly mobile regions in a translational sense, which supplements the *deformation feature* point set we found previously. These features are referred to as *displacement features*.

Consider the displacement curve $\Delta_{\mathrm{V}}(f)$ of vertex $v \in \mathcal{M}$ that encodes the displacement of v at each frame (see Appendix B for the definition). Then the sum of vertex displacements over all frames $\int \Delta_{\mathrm{V}} = \sum_{\mathrm{f}=1}^{M} \Delta_{\mathrm{V}}(\mathrm{f})$ yields the total length of the vertex trajectory. Displacement feature points are defined as local maxima of the trajectory length, subject to a threshold.

Definition 1. Vertex $v \in \mathcal{M}$ is a *displacement feature point* if and only if

$$\forall u \in N_s(v) : (\int \Delta_{\mathrm{v}} > \int \Delta_{\mathrm{u}}) \wedge (\int \Delta_{\mathrm{v}} > \mathrm{h}^{\Delta})$$

where threshold h^{Δ} controls the minimum allowed trajectory length of the displacement feature points.

In our experiments we usually set h^{Δ} to 0.1 of global maximum in trajectory length. Typically, displacement feature points represent the end-effectors of animated characters, which cannot be detected as features using the deformation characteristic. As is the case with the dynamic feature points, a same static mesh can yield different sets of displacement feature points, depending on the types of motions it undergoes. Note that in case of animated meshes with no skeleton structure, such as faces, no displacement feature point is detected.

For the correspondence finding, we use a full set P of dynamic feature points as the union of deformation feature points and displacement ones $P = P_d \bigcup P_{\Delta}$

6. Feature correspondence as graph labeling

We now use the computed feature descriptor to search for correspondences among them. We formulate our feature matching problem as a graph matching, where the feature points from the source and the target animation are viewed as vertices of the two graphs and edges. Graph matching is a challenging discrete optimization problem which received considerable attention in the literature. In our work, we adopt the dual decomposition technique introduced by Torresani et al [TKR08] [TKR13]. The original problem, which is too complex to solve directly, is decomposed into a series of subproblems that are smaller and solvable. The key idea is, instead of minimizing directly the energy of the original problem, to maximize the lower bound of it by solving the dual to the LP (linear programming) relaxation of the integer problem. After solving each subproblem, the solutions are combined using a projected-subgradient scheme to obtain the solution of the original problem. This technique has been chosen for several practical reasons: First, the method has a low computational complexity, and therefore is fast. Second, it performs inexact matching, allowing us to handle the possibly inconsistent feature point sets between the source and the target animated meshes. Finally, the method can produce correct matching results in most cases, and so is able to robustly handle the possible variations of the corresponding feature point descriptors inherited from the motion variation.

In what follows, we describe how we construct the feature graph to be labeled (Section 6.1), define the energy terms of our matching problem (Section 6.2), and compute the solution (Section 6.3).

6.1. Graph construction

Given a set of feature points $P = \{(\mathbf{p},t)\}$, $P' = \{(\mathbf{p}',t')\}$ extracted from the source $\mathcal{M}$ and target $\mathcal{M}'$ animations respectively, we construct a *feature graph* as follows. First we discard the time coordinate by projecting the feature point sets on the space domain: $P = \{\mathbf{p}\}$, $P' = \{\mathbf{p}'\}$. Next, we define a set of possible correspondences or matching pairs $A \subseteq P \times P'$. Each correspondence

$a = \left(\mathbf{p},\, \mathbf{p}'\right) \in A$ is associated with a cost θ_a equal to the feature descriptor distance $D_{\mathcal{H}}\left(\mathbf{p},\, \mathbf{p}'\right)$ as defined in Section 4.4). When a same spatial point has been detected as feature at different times, we compute AnimHoG descriptor at each frame of detection, and the cost of correspondence is computed by taking the minimum of all pairwise distances with respect to the descriptor(s) of the corresponding feature.

For each feature point $\mathbf{p} \in P$, l features $P_l^{'}\left(\mathbf{p}\right) \subset P'$ on the target that are closest to $\mathbf{p}$ in terms of feature descriptor distance are considered for potential matching. We further populate A by including pairs of features on target and l features on the source $P_l\left(\mathbf{p}'\right) \subset P$ with the most similar signatures. We typically set l=4 throughout our experiments.

6.2. Energy function

We now formulate our objective function to be minimized as a weighted sum of four energy terms:

$$E\left(\mathbf{x}\right) = \lambda^{dscr}E^{dscr}\left(\mathbf{x}\right) + \lambda^{geod}E^{geod}\left(\mathbf{x}\right) + \lambda^{unmat}E^{unmat}\left(\mathbf{x}\right) + \lambda^{coh}E^{coh}\left(\mathbf{x}\right) \quad (2)$$

where $\mathbf{x}$ is a binary-valued vector $\mathbf{x} \in \{0,1\}^A$ representing a feature matching solution. Each value x_a in $\mathbf{x}$ is associated with a correspondence a: If $x_a = 1$ then the correspondence a is included in feature matching solution, and $x_a = 0$ otherwise. The E^{dscr} term estimates nodes dissimilarities, E^{geod} describes edge dissimilarities and E^{unmat} E^{coh} are stabilizing terms enforcing valid graph matching results. We detail each of the energy terms below.

Descriptor energy term measures the descriptor distance (Section 4.4) between a pair of feature points involved in an active correspondence and is defined as

$$E^{dscr}\left(\mathbf{x}\right) = \sum_{a\in A} \theta_a x_a$$

with $\theta_a = D_{\mathcal{H}}\left(\mathbf{p},\, \mathbf{p}'\right)$. This term favors matching of features with similar descriptors. Note that we set θ_a to a very large value when the types of the two features in-volved are different (i.e. displacement and deformation features).

Geodesic distortion term E^{geod} penalizes the matching of the features belonging to different structural parts. Considering the semantically similar motion between the two subjects, it would be reasonable to assume a similar structural relation among the feature points. Let $\Gamma^{\mathcal{M}}_{\mathbf{p},\mathbf{q}}$ be the Dijkstra's shortest path approximating discrete geodesic path from $\mathbf{p}$ to $\mathbf{q}$ on mesh $\mathcal{M}$. As in [HSKK01], we define average geodesic distance from vertex $\mathbf{v}$ to all other vertices of $\mathcal{M}$ as $\left|\Gamma^{\mathcal{M}}_{\mathbf{v}}\right| = \frac{1}{|\mathcal{M}|}\sum_{\forall \mathbf{p}\in\mathcal{M}} \left|\Gamma^{\mathcal{M}}_{\mathbf{v},\mathbf{p}}\right|$. In general, $\left|\Gamma^{\mathcal{M}}_{\mathbf{v}}\right|$ reaches global minimum when $\mathbf{v}$ is near the center of an object and global maximum near "end-effector" locations. Thus we can distinguish two feature points belonging to distinct body parts by observing how $\left|\Gamma^{M}_{\mathbf{v}_i}\right|$ change along the geodesic path for $\forall \mathbf{v}_i \in \Gamma^{\mathcal{M}}_{\mathbf{p},\mathbf{q}}$ where i indexes the order in which vertices are encountered along the geodesic path. Specifically, given two feature points $\mathbf{p}$ and $\mathbf{q}$ on a mesh, we see that $\gamma\left(\Gamma^{\mathcal{M}}_{\mathbf{p},\mathbf{q}}\right) = \left|\Gamma^{M}_{\mathbf{v}_i}\right|$ for $\forall \mathbf{v}_i \in \Gamma^{\mathcal{M}}_{\mathbf{p},\mathbf{q}}$ is monotonously increasing or decreasing if $\mathbf{p}$ and $\mathbf{q}$ belong to the same semantic part (Figure 4). On the other hand, γ is non-monotonous with one global minimum if they belong to different body parts, as $\Gamma^{\mathcal{M}}_{\mathbf{p},\mathbf{q}}$ always passes through the neighborhood of the geodesic center of the mesh [PSR89], where γ reaches its global minimum [Moe05] [NAJ11].

Now let $\mathcal{N}$ be a set of neighboring correspondence pairs

$$\mathcal{N} = \{\left\langle\left(\mathbf{p},\mathbf{p}'\right),\left(\mathbf{q},\mathbf{q}'\right)\right\rangle \in A\times A|\mathbf{p}\in N_s^k\left(\mathbf{q}\right)\vee \mathbf{q}\in N_s^k\left(\mathbf{p}\right)\vee \mathbf{p}'\in N_s^k\left(\mathbf{q}'\right)\vee \mathbf{q}'\in N_s^k\left(\mathbf{p}'\right)\}$$

where $N_s^k\left(\mathbf{p}\right)$ is a set of $k\in\mathbb{N}^+$ nearest (spatial) neighbors of feature point $\mathbf{p}$. For our matching experiments k has been set to a value from 3 to 6. Then we define the geodesic distortion energy term over the pairs of active correspondences $(a,b)\in\mathcal{N}$ as

$$E^{geod}\left(\mathbf{x}\right) = \sum_{(a,b)\in\mathcal{N}} \theta^{geod}_{ab} x_a x_b$$

where θ^{geod}_{ab} is set according to **Algorithm 1**. Intuitively, θ^{geod}_{ab} is a scalar representing the distortion between correspondences a and b as measured by using the average geodesic distance curves, normalized by the largest geodesic distances of respective meshes. In Figure 4, for example, the value of θ^{geod}_{ab} is set to $+\infty$, indicating that a and b involve highly distorted (inconsistent) matching.

Algorithm 1. Geodesic distortion term computation.

```
Input: A pair of correspondences <$a$=$(p,p')$,$b$=$(q,q')$>, and the normalized average geodesic curves $\tilde{\gamma}(\Gamma^{\mathcal{M}}_{p,q})$, $\tilde{\gamma}(\Gamma^{\mathcal{M}'}_{p',q'})$.
Output: Geodesic distortion term $\theta^{geod}_{ab}$ // $\nearrow(\cdot)$: monotonously increasing; $\searrow(\cdot)$: monotonously decreasing.
begin
    if ($\nearrow(\tilde{\gamma}(\Gamma^{\mathcal{M}}_{p,q}))$ and $\nearrow(\tilde{\gamma}(\Gamma^{\mathcal{M}'}_{p',q'}))$ ) or
      ($\searrow(\tilde{\gamma}(\Gamma^{\mathcal{M}}_{p,q}))$ and $\searrow(\tilde{\gamma}(\Gamma^{\mathcal{M}'}_{p',q'}))$ ) then
          $\theta^{geod}_{ab} \leftarrow \sum\left(\tilde{\gamma}(\Gamma^{\mathcal{M}}_{p,q}) - \tilde{\gamma}(\Gamma^{\mathcal{M}'}_{p',q'})\right)$
      else
          $\theta^{geod}_{ab} \leftarrow (+\infty)$
end
```

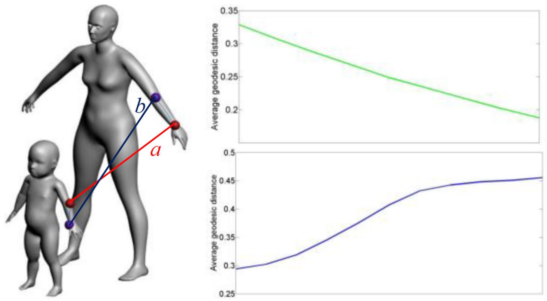


Figure 4: *Average geodesic distance changes along the geodesic path from a feature assigned by correspondence a to another one assigned by correspondence b. Different behaviors of two curves on the woman (above) and the baby (below) indicate that a and b are not compatible.*

Penalty term for unmatched feature pairs. This term imposes a penalty on the number of unmatched features. The penalty can be effectively defined as a ratio of unmatched features in the smaller set among P and P'. More precisely the unmatched ratio is defined as

$$E^{unmat}(\mathbf{x}) = 1 - \frac{1}{\min\{|P|,|P'|\}} \sum_{a \in A} x_a$$

Without losing generality we can assume that the number of source features is less than the number target features, i.e. $\min\{|P|,|P'|\} = |P|$. Then $E^{unmat}(\mathbf{x}) = 0$ is minimal when all features in P are matched, and $E^{unmat}(\mathbf{x}) = 1$ is maximal when none of the features are matched. Note that $E^{unmat}(\mathbf{x})$ is especially useful in inexact graph matching, since it balances with the proportion of discrepancies in feature point sets that the matching can tolerate.

Coherency term. The last term measures the coherency in neighborhoods of matched regions by favoring the feature matching solution $\mathbf{x}$ that preserves spatial proximity of matched features. Intuitively the term is interpreted as a proportion of neighboring features with different matched/unmatched status. More precisely, given neighboring features $\mathbf{p}$ (involved in the correspondence x_a and $\mathbf{q}$ (involved in correspondence x_b), the matching status coherency $V_{\mathbf{p},\mathbf{q}}$ is 0 if x_a **xor** $x_b = 0$, and 1 otherwise. Formally the consistency term is defined as a fraction of all neighboring features with different matched/unmatched status, yielding the following formula:

$$E^{coh}(\mathbf{x}) = \frac{1}{|N_{\mathbf{p},\mathbf{q}}|} \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{N}} V_{\mathbf{p},\mathbf{q}}(\mathbf{x})$$

where $N_{\mathbf{p},\mathbf{q}}$ is a set of neighbouring features $N_{\mathbf{p},\mathbf{q}} = \{(\mathbf{p},\mathbf{q}) \in (P \times P) \bigcup (P' \times P') \mid \mathbf{p} \in N_s^k(\mathbf{q}) \vee \mathbf{q} \in N_s^k(\mathbf{p})\}$.

6.3. Solution computation

To complete the description of our technique, we specify how to obtain solution $\mathbf{x}$. By choosing the dual approach we do not attempt to accurately minimize the energy function $E(\mathbf{x}|\theta)$ directly. Instead, the lower bound of its dual function $\Phi(\theta)$ is maximized. The dual problem is then decomposed into $|\bar{P}|$ subproblems for each point $\mathbf{p} \in \bar{P} = P \bigcup P'$ as follows. Given a set of k nearest neighbours $N_{\mathbf{p}}$ for each point $\mathbf{p} \in \bar{P}$, a subproblem is considered with a set of correspondences: $A(N_{\mathbf{p}}) = \{(\mathbf{q},\mathbf{q}') \in A | \mathbf{q} \in N_{\mathbf{p}} \vee \mathbf{q}' \in N_{\mathbf{p}}\}$. Intuitively, for the local subproblem of point p, we consider only assignments to its k nearest neighbourhood $N_{\mathbf{p}}$. With these assignments in set $A(N_{\mathbf{p}})$, we set the values of the graph matching terms $\theta^{\mathbf{p}}$ as follows: $\theta_a^{\mathbf{p}} = 0$ if $a \notin A(N_{\mathbf{p}})$, $\theta_{ab}^{\mathbf{p}} = 0$ if $a \notin A(N_{\mathbf{p}})$ or $b \notin A(N_{\mathbf{p}})$. The correspondence set $A(N_{\mathbf{p}})$ and corresponding terms $\theta^{\mathbf{p}}$ define a local graph matching problem over the local neighbourhood $N_{\mathbf{p}}$.

As demonstrated in the work of [TKR08], the sum of the solutions to all local subproblems $\sum_{\varsigma \in I} \min_{\mathbf{x}} E(\mathbf{x}|\theta^{\varsigma})$ can serve as a lower bound optimization function $\Phi(\theta)$ as defined by:

$$\Phi(\theta) = \sum_{\varsigma \in I} \min_{\mathbf{x}} E(\mathbf{x}|\theta^{\varsigma}) \leq \min_{\mathbf{x}} E(\mathbf{x}|\theta)$$

where I is the set of subproblem indices. The global minimum for each subproblem $\min_{\mathbf{x}} E(\mathbf{x}|\theta)$ is computed by an exhaustive search. Then, given the solutions for all subproblems, the lower bound $\Phi(\theta)$ is *maximized* via the classical optimization subgradient method. For the complete description of the dual decomposition technique, we refer to the original papers of Torresani et al [TKR08] [TKR13].

7. Experimental results and discussion

We now present the results of feature correspondence using the graph matching. We used 'bending cylinders', galloping 'horse and camel', and walking 'woman and baby' animations (Figure 5). Experiments were performed on a workstation with 16GB of memory and Intel Core i7-2600 processor running at 3.40 GHz.

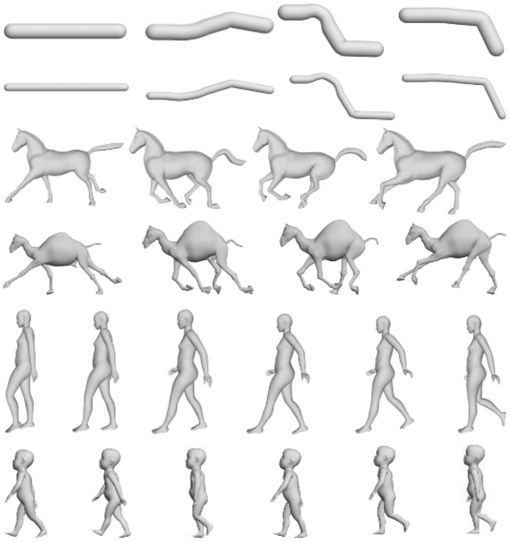


Figure 5: *Animation sequences used in our experiments: bending cylinders, galloping 'horse and camel', and walking 'woman and baby'.*

To validate the effectiveness of our method, we first build ground truths on the correspondences. As opposed to the case of static meshes, ground truth on dynamic data is rarely available, which makes it difficult to perform validation and comparative analysis of our work. To obtain the ground truths, we have asked a user to manually define correspondences, who has examined the dynamic feature points by using our viewer specially developed for the visualization of spatiotemporal feature points. Table 1 lists the manually selected ground truths for our dataset.

Table 1. Ground truths of feature correspondence.

	No. features on $\mathcal{M}$ and $\mathcal{M}'$		No. unmatched points on $\mathcal{M}$ and $\mathcal{M}'$	
Cylinders	5	5	0	0
Horse & Camel	19	24	0	5
Woman & Baby	23	30	8	15

We then compare the performance of AnimHOG with two other dynamic descriptors: normalized displacement curve, and normalized deformation. Finally, we experiment with a composite descriptor which combines all these descriptors. We describe the details of our experiments in next sections.

7.1. Algorithms

Normalized displacement curve. This descriptor is based on the displacement vector, which has been introduced in Arcila et al [ACH*13] for spatial segmentation of mesh sequence. It encodes the inter-frame displacement of a (feature) point with respect to the previous frame, in the world coordinate system. We propose to use it for matching as follows. First, we normalize displacement values by using the global maximum displacement of the given animation, in order to achieve scale-invariant property. We then define the distance between two normalized displacement curves as *l2*-norm of the difference between the two curves. A complete description on this descriptor can be found in Appendix B.

In Figure 6, we depict two normalized displacement curves for the corresponding features in the 'walking woman' and 'walking baby' animations.

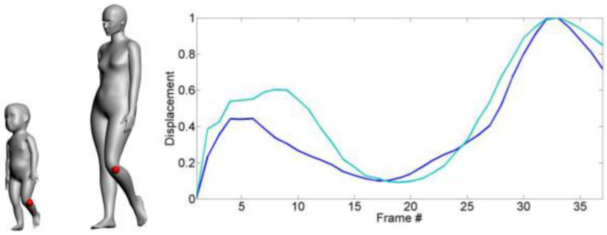


Figure 6: *Normalized displacement curves for two feature points on the walking woman (in blue color) and baby animations (in cyan color).*

Normalized deformation characteristic curve. The second feature descriptor we compare with AnimHOG encodes the temporal evolution of the deformation characteristic as defined in Section 3. In order to obtain the invariance to absolute degree of motion variability, we normalize the deformation characteristic values by using the global maximum of the animated mesh. A complete description on this descriptor can be found in Appendix C.

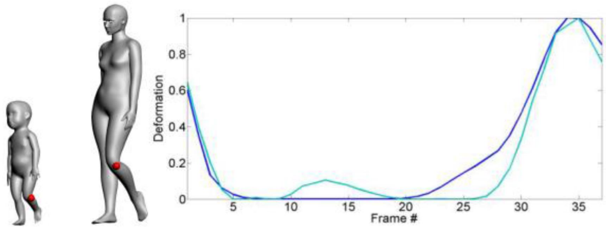


Figure 7: *Deformation functions of the corresponding features for the walking woman (blue) and baby (turquoise) animations.*

In Figure 7, we show the plots of normalized deformation characteristic curves for feature points in the walking woman and baby animations. Note that despite notable differences between the shapes and motions of the woman and baby models, the curve shows similar behavior for the corresponding feature points.

The computation of AnimHOG descriptor has the storage complexity of O(1), but a **heavy time complexity**. Here we discuss time and storage complexities of the descriptors under comparison. Both the displacement and deformation characteristic curves have the linear time and storage complexity of O(M), where M is the number of frames in an animation. Given the number of points in the spatial $\left|N_s^k(\mathbf{p})\right|$, temporal $\left|N_t^k(\mathbf{p})\right|$, and spatiotemporal $\left|N_{st}^k(\mathbf{p})\right|$ neighborhoods of a feature point p, the computational time for each step of AnimHOG descriptor can be estimated as follows:

- Isomap: $\mathrm{O}\left(\left|N_s^k(\mathbf{p})\right|\right) + \mathrm{O}(\left|N_s^k(\mathbf{p})\right| \cdot \log\left(\left|N_s^k(\mathbf{p})\right|\right)) + \mathrm{O}\left(\left|N_s^k(\mathbf{p})\right|^2\right)$.
- Normal orientation check: linear $\mathrm{O}\left(\left|N_s^k(\mathbf{p})\right|\right)$.
- RBF interpolation: $\mathrm{O}\left(\left|N_{st}(\mathbf{p})\right| \cdot \log\left(\left|N_{st}(\mathbf{p})\right|\right)\right)$.
- Histogram binning: $\mathrm{O}\left(\left|N_{st}(\mathbf{p})\right|\right)$.

7.2. Comparative analysis

We have performed the comparative computations by computing E^{dscr} energy term with respective descriptors while fixing other energy terms. Table 2 summarizes the accuracy of the correspondences obtained by using different descriptors. We observe that the overall performance of AnimHOG descriptor is better than the other two descriptors. The commonly observed mismatch by AnimHOG is the symmetric confusion caused by the left-right symmetry of the motion. The curve based descriptors seem to show relatively better performance in such case.

On the other hand, depending on the animation, the displacement curve descriptor can be overly normalized by a dominant value of $\Delta_{\max}$. For instance, the features located at the end of the hierarchy chain can have much larger displacements than those in the upper level. In such case, only those features near end effectors remain distinctive.

Table 2. Accuracy (error rate) of the correspondences obtained by using different descriptors.

	Normalized displacement curve	Normalized deformation characteristic curve	AnimHOG
Cylinders	5/5 (0%)	5/5 (0%)	5/5 (0%)
Horse & Camel	14/19 (26.3%)	12/19 (36.8%)	17/19 (10.5%)
Woman & Baby	12/15 (20%)	13/15 (13.3%)	12/15 (20%)

In all of our experiments the graph matching algorithm produces feature correspondence solutions in less than a few seconds (0.14s for the cylinders, and 1.89s for the woman and baby).

7.3. Composite descriptor

Although AnimHOG encodes well the local deformation characteristics of the feature point, it is not sufficiently distinctive for a complete matching in some cases, as have seen above. On the other hand, the two curve descriptors incorporate the temporal information on the features. Given such complimentary properties of descriptors, we propose a composite dynamic descriptor of a feature point $\mathbf{p}$ as a triple composed of the dynamic descriptors described above: $\mathcal{H} = \left(\Delta^{\mathbf{P}}, c_d^{\mathbf{P}}, H^{\mathbf{P}}\right)$. The proposed descriptor maximizes the amount of information about distinctive deformation and dynamic properties gained around spatial and temporal extents around a feature point. Given the composite descriptor $\mathcal{H}$, the distance metric between feature point signatures is defined as

$$D_{\mathcal{H}}\left(\mathbf{p},\mathbf{p}'\right) = w_1 D_{\Delta^{\mathbf{P}}}\left(\mathbf{p},\mathbf{p}'\right) + w_2 D_{c_d^{\mathbf{P}}}\left(\mathbf{p},\mathbf{p}'\right) + w_3 D_{H^{\mathbf{P}}}\left(\mathbf{p},\mathbf{p}'\right) \quad (3)$$

where $D_{\Delta^{\mathbf{P}}}$ ((Equation (5)) and $D_{c_d^{\mathbf{P}}}$ ((Equation (6)) are the distances defined for the normalized displacement curves and normalized deformation characteristics curves, respectively, and w_{i} are positive weights that control the influence of each signature. In our settings we usually set $w_i = 1$. As shown in Figure 8, we obtain very encouraging results by using the composite descriptor. It produces very good quality of correspondence results, showing a perfect matching accuracy with respect to the ground truths in all animation dataset.

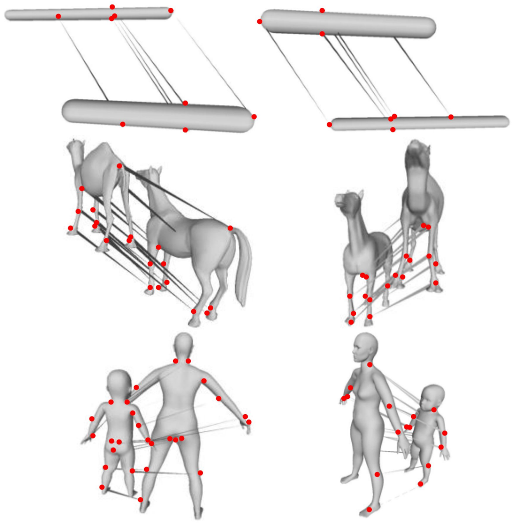


Figure 8: *Correspondence results on the animated meshes. Feature points are shown with red dots.*

7.4. Comparison with geometric descriptors

Since our technique makes use of the deformation/motion properties, it is clear that it can give better results compared to those that solely rely on the geometric properties. We illustrate this in the Figure 9, which shows two moving arms whose forearms have different length. Our method successfully identifies the location of the elbow on both arms, whereas the PLANSAC [TBW*11] fails to match the exact locations of the two elbows. This is because the length of the other forearm is shorter and the PLANSAC locates the elbow point so that its relative distance to the shoulder and the hand is preserved.

7.5. Dense matching

Once we have computed the matching among feature vertices, we can propagate the computed correspondence to all points on the source mesh. Here, we demonstrate this via the bijective mapping in the spherical parameterization. Our algorithm takes as input the spherical mesh and moves its vertices on the surface of the unit sphere such that the result mesh on the sphere is fold-over free, and its feature vertices are aligned with their corresponding counterparts of the target mesh. Similar in spirit to [SC10], we compute this warping function as a sequence of iterative warpings based on radial basis function, where each warping step incrementally moves the source feature vertices to their counterpart. We show the results of the full matching Figure 10.

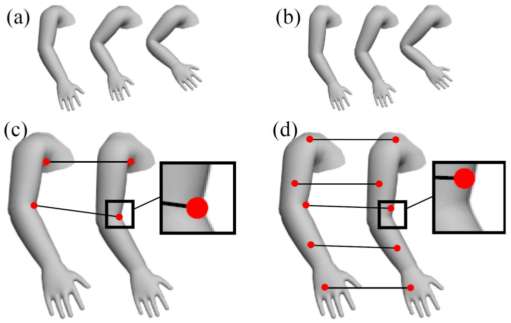


Figure 9: *Matching between two animated arms; the forearm in (a) is slightly shorter than the one shown in (b). Our method successfully matches (c), while RANSAC misses (d), the corresponding location of the elbow on both arms.*

7.6. Application

Our work can be employed in applications such as medicine, where inter-subject correspondence is the key element for the diagnosis and detection of abnormalities. Although the use of geometric feature is still a golden standard, it may generate results with limited capability of reliable correspondence computation since the heart drastically changes its spatial arrangement over time. With our techniques, one can use not only the geometric features of some key shapes but also the functional (spatio-temporal) features for the correspondence computation. This is particularly relevant for organs with mobility such as heart, the observation of which becomes increasingly available. However, this would require extending the proposed technique to handle volumes as input.

8. Conclusion

In this paper we presented a shape correspondence technique for animated meshes, a problem that hasn't been addressed before. Motivated by the pioneer idea of exploiting available motion information for the matching, we have developed new dynamic descriptors that capture local deformation pattern and temporal motional behavior of a feature point on the animated mesh. We have used these descriptors successfully in finding a robust spatial matching between animated meshes, by adopting a graph matching based optimization scheme. Our goal function is defined in such a way that

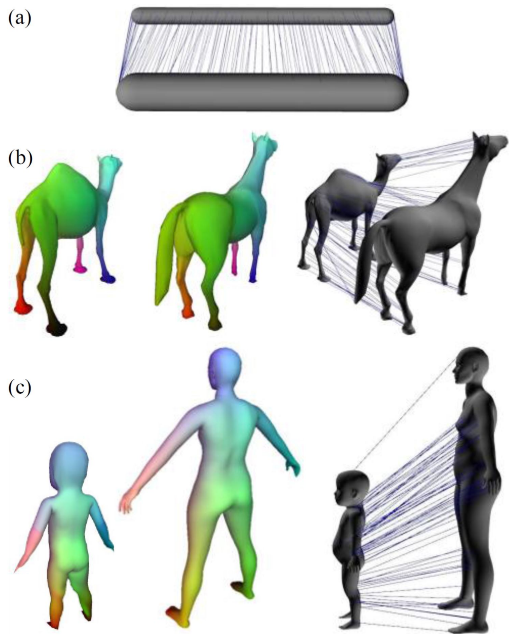


Figure 10: *Fine matching between bending cylinders (a), galloping horse/camel (b) and walking woman/baby (c) animations. Subsets of matching pairs of vertices are depicted by lines. The full dense correspondence is depicted in color.*

the descriptor similarity between the corresponding feature pairs as well as compatibility and consistency as measured across the pairs of correspondences are maximized.

Since our technique makes use of the deformation/motion properties, it can produce more reliable results compared to those that solely rely on the geometric properties. Additionally, it can help avoid symmetry confusion in the matching with the use of motion properties of deforming meshes, given appropriate motion data. However, our work relies on the given motion and thus it cannot perform well if the motions exhibited are insufficient or semantically different.

Acknowledgements

We would like to thank Vasyl Mykhalchuk for his help with the implementation and Alexis Renault for having made the ground-truth analysis.

This work has been supported by the French national project SHARED (Shape Analysis and Registration of People Using Dynamic Data, No.10-CHEX-014-01).

References

[ACH*13] Arcila R., Cagniart C., HéTroy F., Boyer E., Dupont F.: Segmentation of temporal mesh sequences into rigidly moving components. *Graph. Models 75*, 1 (Jan. 2013), 10–22.

[ACSD*03] Alliez P., Cohen-Steiner D., Devillers O., Lévy B., Desbrun M.: Anisotropic polygonal remeshing. *ACM Trans. Graph. 22*, 3 (July 2003), 485–493.

[AKH*10] Arcila R., Kartik B. S., Hétroy F., Denis F., Dupont F.: A framework for motion-based mesh sequence segmentation. In *WSCG'10: International Conference on Computer Graphics, Visualization and Computer Vision* (Plzen, Tchèquie, Feb. 2010).

[ASP*04] Anguelov D., Srinivasan P., Pang H.-C., Koller D., Thrun S., Davis J.: The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. In *Proceedings of the 17th International Conference on Neural Information Processing Systems* (Cambridge, MA, USA, 2004), NIPS'04, MIT Press, pp. 33–40.

[BETVG08] Bay H., Ess A., Tuytelaars T., Van Gool L.: Speeded-up robust features (surf). *Comput. Vis. Image Underst. 110*, 3 (June 2008), 346–359.

[BMP00] Belongie S., Malik J., Puzicha J.: Shape context: A new descriptor for shape matching and object recognition. In *Neural Information Processing Systems Conference (NIPS)* (2000), pp. 831–837.

[CCFM08] Castellani U., Cristani M., Fantoni S., Murino V.: Sparse points matching by combining 3d mesh saliency with statistical descriptors. *Computer Graphics Forum* (2008).

[CWZ*14] Cao C., Weng Y., Zhou S., Tong Y., Zhou K.: Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics 20*, 3 (Mar. 2014), 413–425.

[CZ08] Chang W., Zwicker M.: Automatic registration for articulated shapes. *Computer Graphics Forum (Proceedings of SGP 2008) 27*, 5 (2008), 1459–1468.

[DATTS08] De Aguiar E., Theobalt C., Thrun S., Seidel H.-P.: Automatic Conversion of Mesh Animations into Skeleton-based Animations. *Computer Graphics Forum* (2008).

[DB01] De Boor C.: *A practical guide to splines; rev. ed.* Applied mathematical sciences. Springer, Berlin, 2001.

[DCWS03] Doretto G., Chiuso A., Wu Y., Soatto S.: Dynamic textures. *International Journal of Computer Vision 51*, 2 (2003), 91–109.

[DK12] Darom T., Keller Y.: Scale-invariant features for 3-d mesh models. *IEEE Transactions on Image Processing 21*, 5 (May 2012), 2758–2769.

[DLL*10] Dey T. K., Li K., Luo C., Ranjan P., Safa I., Wang Y.: Persistent heat signature for pose-oblivious matching of incomplete models. *Computer Graphics Forum* (2010).

[DRCB05] Dollar P., Rabaud V., Cottrell G., Belongie S.: Behavior recognition via sparse spatio-temporal features. In *Proceedings of the 14th International Conference on Computer Communications and Networks* (Washington, DC, USA, 2005), ICCCN '05, IEEE Computer Society, pp. 65–72.

[DT05] Dalal N., Triggs B.: Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01* (Washington, DC, USA, 2005), CVPR '05, IEEE Computer Society, pp. 886–893.

[EG04] Eilers P. H. C., Goeman J. J.: Enhancing scatterplots with smoothed densities. *Bioinformatics 20*, 5 (2004), 623–628.

[FS06] Funkhouser T., Shilane P.: Partial matching of 3d shapes with priority-driven search. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP '06, Eurographics Association, pp. 131–142.

[GGGZ05] Gatzke T., Grimm C., Garland M., Zelinka S.: Curvature maps for local shape comparison. In *International Conference on Shape Modeling and Applications 2005 (SMI' 05)* (June 2005), pp. 244–253.

[GMGP05] Gelfand N., Mitra N. J., Guibas L. J., Pottmann H.: Robust global registration. In *Proceedings of the Third Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2005), SGP '05, Eurographics Association.

[HSKK01] Hilaga M., Shinagawa Y., Kohmura T., Kunii T. L.: Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 203–212.

[Joh97] Johnson A.: *Spin-images: A Representation for 3-D Surface Matching*. Technical report. Carnegie Mellon University, The Robotics Institute, 1997.

[JZvK07] Jain V., Zhang H., van Kaick O.: Non-rigid spectral correspondence of triangle meshes. *International Journal on Shape Modeling 13*, 1 (2007), 101–124.

[KLF11] Kim V. G., Lipman Y., Funkhouser T.: Blended intrinsic maps. *ACM Trans. Graph. 30*, 4 (July 2011), 79:1–79:12.

[KMS08] Klaeser A., Marszalek M., Schmid C.: A spatio-temporal descriptor based on 3d-gradients. In *Proceedings of the British Machine Vision Conference* (2008), BMVA Press, pp. 99.1–99.10.

[KPNK03] Körtgen M., Park G. J., Novotni M., Klein R.: 3d shape matching with 3d shape contexts. In *The 7th Central European Seminar on Computer Graphics* (Apr. 2003).

[LF09] Lipman Y., Funkhouser T.: MÖbius voting for surface correspondence. *ACM Trans. Graph. 28*, 3 (July 2009), 72:1–72:12.

[LH05] Leordeanu M., Hebert M.: A spectral technique for correspondence problems using pairwise constraints. In *Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2* (Washington, DC, USA, 2005), ICCV '05, IEEE Computer Society, pp. 1482–1489.

[Lin98] Lindeberg T.: Feature detection with automatic scale selection. *Int. J. Comput. Vision 30*, 2 (Nov. 1998), 79–116.

[LO07] Ling H., Okada K.: An efficient earth mover's distance algorithm for robust histogram comparison. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 29*, 5 (2007), 840–853.

[Low04] Lowe D. G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2 (Nov. 2004), 91–110.

[LWC06] Lee T.-Y., Wang Y.-S., Chen T.-G.: Segmenting a deforming mesh into near-rigid components. *The Visual Computer 22*, 9 (2006), 729–739.

[LWP10] Li H., Weise T., Pauly M.: Example-based facial rigging. *ACM Trans. Graph. 29*, 4 (July 2010), 32:1–32:6.

[MBO10] Mian A., Bennamoun M., Owens R.: On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes. *Int. J. Comput. Vision 89*, 2-3 (Sept. 2010), 348–361.

[MCS13] Mykhalchuk V., Cordier F., Seo H.: Landmark transfer with minimal graph. *Computers Graphics 37*, 5 (2013), 539 – 552.

[MFO*07] Mitra N. J., Flöry S., Ovsjanikov M., Gelfand N., Guibas L., Pottmann H.: Dynamic geometry registration. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2007), SGP '07, Eurographics Association, pp. 173–182.

[Moe05] Moenning C.: *Intrinsic Point-based Surface Processing*. University of Cambridge, 2005.

[MSC15] Mykhalchuk V., Seo H., Cordier F.: On spatio-temporal feature point detection for animated meshes. *The Visual Computer 31*, 11 (2015), 1471–1486.

[MZP06] Mamou K., Zaharia T., Prêteux F.: A skinning prediction scheme for dynamic 3D mesh compression. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* (Aug. 2006), vol. 6315 of , p. 631502.

[NAJ11] Noyel G., Angulo J., Jeulin D.: Fast computation of all pairs of geodesic distances. *Image Analysis & Stereology 30*, 2 (2011), 101–109.

[OMMG10] Ovsjanikov M., Mérigot Q., Mémoli F., Guibas L. J.: One point isometric matching with the heat kernel. *Comput. Graph. Forum 29*, 5 (2010), 1555–1564.

[PSR89] Pollack R., Sharir M., Rote G.: Computing the geodesic center of a simple polygon. *Discrete & Computational Geometry 4*, 6 (1989), 611–626.

[RCV09] Ravichandran A., Chaudhry R., Vidal R.: View-invariant dynamic texture recognition using a bag of dynamical systems. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA* (2009), pp. 1651–1657.

[SAS07] Scovanner P., Ali S., Shah M.: A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th ACM International Conference on Multimedia* (New York, NY, USA, 2007), MM '07, ACM, pp. 357–360.

[SC10] Seo H., Cordier F.: Constrained texture mapping using image warping. *Computer Graphics Forum* (2010).

[SOG09] Sun J., Ovsjanikov M., Guibas L.: A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2009), SGP '09, Eurographics Association, pp. 1383–1392.

[SP04] Sumner R. W., Popović J.: Deformation transfer for triangle meshes. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 399–405.

[SSK05] Sattler M., Sarlette R., Klein R.: Simple and efficient compression of animation sequences. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 209–217.

[TBW*09] Tevs A., Bokeloh M., Wand M., Schilling A., Seidel H.: Isometric registration of ambiguous and partial data. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA* (2009), pp. 1185–1192.

[TBW*11] Tevs A., Berner A., Wand M., Ihrke I., Seidel H.: Intrinsic shape matching by planned landmark sampling. *Comput. Graph. Forum 30*, 2 (2011), 543–552.

[TdSL00] Tenenbaum J. B., de Silva V., Langford J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319.

[TKR08] Torresani L., Kolmogorov V., Rother C.: Feature correspondence via graph matching: Models and global optimization. In *Proceedings of the 10th European Conference on Computer Vision: Part II* (Berlin, Heidelberg, 2008), ECCV '08, Springer-Verlag, pp. 596–609.

[TKR13] Torresani L., Kolmogorov V., Rother C.: A dual decomposition approach to feature correspondence. *IEEE Trans. Pattern Anal. Mach. Intell. 35*, 2 (2013), 259–271.

[TM12] Tung T., Matsuyama T.: Invariant surface-based shape descriptor for dynamic surface encoding. In *Computer Vision - ACCV 2012 - 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I* (2012), pp. 486–499.

[TM14] Tung T., Matsuyama T.: Timing-based local descriptor for dynamic surfaces. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014* (2014), pp. 414–421.

[vKZHC10] van Kaick O., Zhang H., Hamarneh G., Cohen-Or D.: A survey on shape correspondence. In *Eurographics 2010 - State of the Art Reports, Norrköping, Sweden, May 3-7, 2010* (2010), pp. 61–82.

[WAO*09] Wand M., Adams B., Ovsjanikov M., Berner A., Bokeloh M., Jenke P., Guibas L., Seidel H.-P., Schilling A.: Efficient reconstruction of nonrigid shape and motion from real-time 3d scanner data. *ACM Trans. Graph. 28*, 2 (May 2009), 15:1–15:15.

[WLVGP09] Weise T., Li H., Van Gool L., Pauly M.: Face/off: Live facial puppetry. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2009), SCA '09, ACM, pp. 7–16.

[WTG08] Willems G., Tuytelaars T., Gool L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In *Proceedings of the 10th European Conference on Computer Vision: Part II* (Berlin, Heidelberg, 2008), ECCV '08, Springer-Verlag, pp. 650–663.

[ZBVH09] Zaharescu A., Boyer E., Varanasi K., Horaud R. P.: Surface feature detection and description with applications to mesh matching. In *International Conference on Computer Vision and Pattern Recognition, CVPR'09, June, 2009* (Miami, Etats-Unis, June 2009), IEEE, pp. 373–380.

[ZG04] Zelinka S., Garland M.: Similarity-based surface modelling using geodesic fans. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (New York, NY, USA, 2004), SGP '04, ACM, pp. 204–213.

[ZSCO*08] Zhang H., Sheffer A., Cohen-Or D., Zhou Q., Van Kaick O., Tagliasacchi A.: Deformation-driven shape correspondence. *Computer Graphics Forum* (2008).

Appendix A: Strain and curvature computation

Strain computation. We consider the degree of deformation associated to each triangle on the mesh at each frame. Let $\mathbf{v}_\mathrm{i}$ and $\tilde{\mathbf{v}}_\mathrm{i}$ be the vertices of a triangle before and after the deformation, respectively. A 3 by 3 affine matrix $\mathbf{F}$ and displacement vector $\mathbf{d}$ transforms $\mathbf{v}_\mathrm{i}$ into $\tilde{\mathbf{v}}_\mathrm{i}$ as: $\mathbf{F}\cdot\mathbf{v}_\mathrm{i}+\mathbf{d}=\tilde{\mathbf{v}}_\mathrm{i}, \mathrm{i}=1,\ldots 3$. Similarly to [SP04], we add a fourth vertex in the direction of the normal vector of the triangle and subtract the first equation from the other to eliminate $\mathbf{d}$. Then, we get $\mathbf{F}=\tilde{\mathbf{V}}\cdot\mathbf{V}^{-1}$ where $\mathbf{V}=[\mathbf{v}_2-\mathbf{v}_1\ \mathbf{v}_3-\mathbf{v}_1\ \mathbf{v}_4-\mathbf{v}_1]$ and $\tilde{\mathbf{V}}=[\tilde{\mathbf{v}}_2-\tilde{\mathbf{v}}_1\ \tilde{\mathbf{v}}_3-\tilde{\mathbf{v}}_1\ \tilde{\mathbf{v}}_4-\tilde{\mathbf{v}}_1]$.

Non-translational component of F encodes the change in orientation, scale, and skew induced by the deformation. Note that this representation specifies the deformation in per-triangle basis, so it is independent of the specific position and orientation of the mesh in world coordinates. Without loss of generality, we assume that the triangle is stretched first and then rotated. Then we have **F**=**RU**, where **R** denotes rotation tensor and **U** stretch tensor. Since we want to describe the triangle only with its degree of stretch, we eliminate the rotation component of **F** by computing the right Cauchy deformation tensor $\mathbf{C}=\mathbf{F}^\mathrm{T}\mathbf{F}=(\mathbf{RU})^\mathbf{T}(\mathbf{RU})$. Because of the orthogonality property of the rotation matrix **R** ($\mathbf{R}^\mathrm{T}\mathbf{R}=\mathbf{I}$, **I** being the identity matrix), matrix **C** can be simplified to $\mathbf{C}=\mathbf{U}^\mathrm{T}\mathbf{R}^\mathrm{T}\mathbf{R}\mathbf{U}=\mathbf{U}^\mathrm{T}\mathbf{U}$. It can be shown that **C** is equal to the square of the right stretch tensor. We obtain principal stretches by Eigen-analysis on **C**, and use the largest eigenvalue as the in-plane deformation of the triangle. The strain at a vertex is computed by taking the average strain of its adjacent triangles.

Curvature computation. Computing the curvature at the vertices of a mesh is known to be non-trivial because of the piecewise-linear nature of meshes. One simple way of computing the curvature would be to compute the angle between two neighboring triangles along an edge. However, such curvature measurement is too sensitive to the noise on the surface of the mesh because its computation relies on two triangles only. Instead, we compute the curvature over a set of edges as described in [ACSD*03]. Given a vertex $\mathbf{v_i}$, we first compute the set of edges E_j whose two vertices are within a user-defined geodesic distance to $\mathbf{v_i}$. Next we compute the curvature at each E_j. The curvature at $\mathbf{v_i}$ is calculated as the average of the curvatures computed at each E_j.

Appendix B: Normalized displacement curve

The *displacement curve* of a spatiotemporal feature point $\mathbf{p}\in\mathcal{M}$ is defined as a function $\Delta_\mathrm{p}:[1,M]\to\mathbb{R}$ which maps each frame of the animated mesh to the corresponding norm of displacement value

$$\Delta_\mathrm{p}(f)=\left\|\mathbf{p}^{f+1}-\mathbf{p}^{f}\right\|, f\in[1,M]$$

where $\mathbf{p}^f$ is the global coordinate of $\mathbf{p}$ at frame f and M is the number of frames of the animated mesh $\mathcal{M}$.

The normalized displacement curve descriptor is defined as $\Delta_\mathrm{p}(f)$ above, normalized by the global maximum displacement of $\mathcal{M}$:

$$\tilde{\Delta}_\mathrm{p}(f)=\frac{\Delta_\mathrm{p}(f)}{\Delta_\mathrm{max}} \quad (4)$$

where $\Delta_\mathrm{max}=\max\limits_{\mathrm{v_i}\in\mathcal{M}}\Delta_{\mathrm{v_i}}$.

Distance metric. Given a feature point p from the source and feature point p' from the target, the distance between two normalized displacement curve signatures is defined as *l2*-norm of the difference between the two curves:

$$D_{\Delta^\mathbf{P}}=\left\|\Delta_\mathrm{p}-\Delta_\mathrm{p'}\right\|_2=\sum_{f=1}^{M}\left|\Delta_\mathrm{p}(f)-\Delta_\mathrm{p'}(f)\right| \quad (5)$$

Frame-subsampling. In order to cope with the case of different number of frames M and $M^{'}$ ($M\neq M^{'}$) in the source and target animations, we proceed as follows. First, we estimate the least common multiple $\bar{M}=\mathrm{LCM}(M,M')$ of the source and target number of frames M and $M^{'}$. Out of the least common multiple $\bar{M}$ we consider that the source is sampled at frames $\left\{\frac{\bar{M}}{M}i|i=1..M\right\}$,and the target at frames $\left\{\frac{\bar{M}}{M'}i|i=1..M'\right\}$. Then we apply cubic spline interpolation [DB01] to obtain samplings on a new, common uniform domain $\bar{M}$.

Appendix C: Normalized deformation characteristics curve

Normalized deformation characteristics curve $c_d^\mathbf{p}$ at a feature point p is defined as a real-valued function on indices of animation frames $c_d^\mathbf{p}:[1,M]\to\mathbb{R}$ such that $c_d^\mathbf{p}(f)=\mathrm{d}\left(\mathbf{p}^\mathbf{f}\right)/\mathrm{d}_\mathrm{max}$ where $\mathrm{d}_\mathrm{max}=\max\limits_{v\in\mathcal{M}}\mathrm{d}(v)$.

Distance metric. Given a deformation characteristics curve $c_d^\mathbf{p}$ for a feature point $\mathbf{p}$ on the source and another $\mathrm{c}_d^{\mathbf{p}'}$ for a feature point $\mathbf{p}'$ on the target, the metric is defined as *l2*-norm of the difference between the two curves, as given by:

$$D_{c_d^\mathbf{p}}(\mathbf{p},\mathbf{p}')=\sum_{f=1}^{M}\left(c_d^\mathbf{p}(f)-c_d^{\mathbf{p}'}(f)\right)^2 \quad (6)$$

In order to make Equation (6) valid for the case when the source and target come with different number of frames, we apply a similar frame-subsampling interpolation as described in Appendix B.