Numer. Math. Theor. Meth. Appl.
doi: 10.4208/nmtma.2013.mssvm12

Vol. **6**, No. 1, pp. 223-244
February 2013

# Partial Shape Matching Without Point-Wise Correspondence

Jonathan Pokrass[1,*], Alexander M. Bronstein[1] and
Michael M. Bronstein[2]

[1] *School of Electrical Engineering, Tel Aviv University, Israel.*
[2] *Institute of Computational Science, Faculty of Informatics, Università della Svizzera Italiana (USI), Lugano, Switzerland.*

Received 15 December 2011; Accepted (in revised version) 1 July 2012

Available online 11 January 2013

**Abstract.** Partial similarity of shapes is a challenging problem arising in many important applications in computer vision, shape analysis, and graphics, e.g. when one has to deal with partial information and acquisition artifacts. The problem is especially hard when the underlying shapes are non-rigid and are given up to a deformation. Partial matching is usually approached by computing local descriptors on a pair of shapes and then establishing a point-wise non-bijective correspondence between the two, taking into account possibly different parts. In this paper, we introduce an alternative correspondence-less approach to matching fragments to an entire shape undergoing a non-rigid deformation. We use region-wise local descriptors and optimize over the integration domains on which the integral descriptors of the two parts match. The problem is regularized using the Mumford-Shah functional. We show an efficient discretization based on the Ambrosio-Tortorelli approximation generalized to triangular point clouds and meshes, and present experiments demonstrating the success of the proposed method.

**AMS subject classifications**: 65D18, 68U05

**Key words**: Deformable shapes, partial matching, partial correspondence, partial similarity, diffusion geometry, Laplace-Beltrami operator, shape descriptors, heat kernel signature, Mumford-Shah regularization.

## 1. Introduction

Many shape analysis applications arising in computer and vision and graphics require matching of partially similar shapes [33]. Such problems typically arise in two scenarios. On the one hand, partial similarity may be the right description of the similarity relationship between two shapes (for example, consider a centaur shape: the centaur is partially

[*]Corresponding author. *Email addresses:* evgenyfo@post.tau.ac.il (J. Pokrass), bron@eng.tau.ac.il (A. M. Bronstein), michael.bronstein@usi.ch (M. M. Bronstein)

similar to a human because they share the human-like upper body, and at the same time, partially similar to a horse because the share the horse-like lower body [29]). On the other hand, many real-world data are degraded by acquisition imperfections and noise (missing views, holes, etc) that are especially acute when acquiring 3D scenes using a range sensing device such as Microsoft Kinect, resulting in the need to work with partially given objects. Such cases are common, for example, in face recognition, where the facial surface may be partially occluded by hair [14], or in computational archeology where one often has to deal with missing pieces [28].

In rigid shape analysis, introducing weights that reject corresponding points that are too distant or whose normals are misaligned into the popular iterative closest point (ICP) algorithm [7,20] are able to deal with partial shape alignment. However, it is impossible to guarantee how large and regular the resulting corresponding parts will be. In order to cope with this problem, a Mumford-Shah [19,39]-like regularization allowing to explicitly control the size of the rejected part and its regularity was used in [10], resulting in a robust ICP algorithm allowing to match rigid shapes with significant dissimilar parts.

This approach is, in fact, a particular setting of the framework introduced by Bronstein *et al.* [11], in which non-rigid partial similarity is formulated as a multi-criterion optimization problem, wherein one tries to find the corresponding parts in two shapes by simultaneously maximizing significance and similarity criteria. The framework allows plugging in different similarity (e.g., some intrinsic metric distortion [11,13,26,37] for non-rigid shapes, or Hausdorff-like distance [10] for rigid shapes) and significance (e.g. part area [10,11] or statistical occurrence of local shape descriptors [16]) criteria to address different settings of the problem. The solution proposed in the methods above requires the knowledge of correspondence between the shapes, and in the absence of a given correspondence, can be solved by alternating between weighted correspondence finding and maximization of part area. Such an alternating optimization scheme is computationally very expensive.

A different class of methods not relying on correspondence are bags of features [46] approaches popular in image analysis recently adopted in 3D shape analysis [12,38,40,49]. The main idea is to represent the shape as a collection of some local feature descriptors [17,21,25,30,32,36,38,41,45,48,51,52] and quantize them in some vocabulary of "geometric words" in order to compute a histogram representing the occurrence of different geometric words in the shape (this method follows the "bag of words" approach in text retrieval, hence the name). If the geometric vocabulary is large enough and the shapes have significant common parts, it is possible to compare partially similar shapes. However, if the similar parts are small, this method usually does not work. Furthermore, since the bag of features representation looses the spatial information, it does not allow to identify the parts that are similar in two shapes.

In this paper, we present an approach for correspondence-less partial matching of non-rigid 3D point clouds and shapes that is, in a sense, a combination of the two aforementioned methods but because our method doesn't require to calculate point-wise correspondence it is less computationally expensive. Our work is inspired by the recent work on partial matching of images [22]. The main idea is to find similar parts by comparing partwise distributions of local descriptors (that can be thought of as "local bags of features"). This removes the need of correspondence knowledge and greatly simplifies the problem.

This paper is an extended version of [42] where the framework was presented. Here, our main focus is on computation on point clouds – a type of geometry representation very popular in computer vision applications, e.g., when acquiring 3D objects with a range scanner. We provide more detailed development of the theoretical and numerical parts of the method and show extensive experimental results. The rest of the paper is organized is as follows. In Section 2, we review the mathematical background of diffusion geometry, which is used for the construction of local descriptors. Section 3 deals with the partial matching problem and Section 4 addresses its discretization. Section 5 presents experimental results. Finally, Section 6 concludes the paper.

## 2. Background

We model a shape as a two-dimensional compact Riemannian manifold $X$ (possibly with a boundary) equipped with a *metric tensor* $g$, defined as a positive-definite bilinear form (inner product) $g_x(\cdot,\cdot) = \langle\cdot,\cdot\rangle_x$ on the tangent space $T_xX$. Given a smooth scalar field $f : X \to \mathbb{R}$ on the manifold, its *gradient* is defined as the vector field $\nabla f$ satisfying $f(x+dx) = f(x) + g_x(\nabla f(x), dx)$ for every point $x$ and every infinitesimal tangent vector $dx \in T_xX$. The metric tensor $g$ defines the *Laplace-Beltrami operator* $\Delta_g$ that satisfies

$$\int f \Delta_g h da = -\int g_x(\nabla f, \nabla h) da \tag{2.1}$$

for any pair of smooth scalar fields $f, h : X \to \mathbb{R}$; here $da$ denotes integration with respect to the standard area measure on $X$. This formula is also known as the *Stokes identity*. The Laplace-Beltrami operator is positive semi-definite and self-adjoint. Furthermore, it is an *intrinsic* property of $X$, i.e., it is expressible solely in terms of $g$. In the case when the metric $g$ is Euclidean, $\Delta_g$ becomes the standard Laplacian. In the following, we use the shorter notation $\Delta$ omitting the reference to the metric $g$, which is implied from context.

By virtue of the spectral theorem, there exists an orthonormal basis on $L_2(X)$ consisting of the eigenfunctions $\phi_0, \phi_1, \cdots$ of the Laplace-Beltrami operator (i.e., solutions to $\Delta\phi_i = \lambda_i\phi_i$, where $0 = \lambda_0 \leq \lambda_1 \leq \cdots$ are the corresponding eigenvalues). This basis can be interpreted analogously to the Fourier basis, and the eigenvalues $\lambda_i$ as frequencies. Reuter *et al.* [43] used the spectrum (eigenvalues) of the Laplace-Beltrami operator as global deformation-invariant shape descriptors, referred to as *shapeDNA*. However, there exists a non-trivial class of shapes that are iso-spectral but not isometric; the existence of such shapes was formulated by Kac as a famous question "can we hear the shape of the drum?"

### 2.1. Diffusion equation and heat kernel signatures

The Laplace-Beltrami operator gives rise to the *heat equation*,

$$\left(\Delta + \frac{\partial}{\partial t}\right) u(x,t) = 0, \tag{2.2}$$

which describes diffusion processes and heat propagation on the manifold. Here, $u(x,t)$ denotes the distribution of heat at time $t$ at point $x$. The initial condition to the equation is some heat distribution $u(x,0)$, and if the manifold has a boundary, appropriate boundary conditions (e.g. Neumann or Dirichlet) must be specified. The solution of (2.2) with a point initial heat distribution $u_0(x) = \delta(x,x')$ is called the *heat kernel* and denoted here by $h_t(x,x')$. The value of $h_t(x,x')$ is the amount of heat transported from $x'$ to $x$ over time $t$, and can also be interpreted as the transition probability density of a Brownian motion of length $t$ on the manifold. Using a signal processing analogy, $h_t$ can be thought of as the "impulse response" of the heat equation. Using spectral expansion, the heat kernel can be represented as [31],

$$h_t(x,x') = \sum_{k\geq 0} e^{-\lambda_k t}\phi_k(x)\phi_k(x), \tag{2.3}$$

which gives an efficient ways to compute the heat kernel numerically: one only needs to find the discretization of the Laplace-Beltrami operator eigenfunctions. This can be done efficiently on different representations of the shape, in particular, triangular meshes and point clouds. This issue is discussed in detail in Section 4.

The diagonal of the heat kernel $h_t(x,x)$ expresses the probability density of remaining at a point $x$ after time $t$. The value $h_t(x,x)$, sometimes referred to as the *auto-diffusivity function*, is related to the Gaussian curvature $K(x)$ through

$$h_t(x,x) = \sum_{k\geq 0} e^{-\lambda_k t}\phi_k^2(x) \approx \frac{1}{4\pi t}\left(1+\frac{1}{6}K(x)t+\mathcal{O}(t^2)\right). \tag{2.4}$$

This relation coincides with the well-known fact that heat tends to diffuse slower at points with positive curvature, and faster at points with negative curvature. Under mild technical conditions, the set $\{h_t(x,x)\}_{t>0}$ is fully informative in the sense that it allows to reconstruct the Riemannian metric of the manifold [48].

Sun *et al.* [48] and Gebal *et al.* [25] proposed constructing point-wise descriptors referred to as *heat kernel signatures* (HKS) by taking the values of the discrete auto-diffusivity function at point $x$ at multiple times, $\mathbf{p}(x) = (h_{t_1}(x,x),\cdots,h_{t_d}(x,x))$, where $t_1,\cdots,t_d$ are some fixed time values. Such a descriptor is a vector of dimensionality $d$ at each point. Since the heat kernel is an intrinsic quantity, the HKS is invariant to isometric transformations of the shape.

A scale-invariant version of the HKS descriptor (SI-HKS) was proposed in [17]. First, the heat kernel is sampled logarithmically in time. Next, the logarithm and a derivative with respect to time of the heat kernel values are taken to undo the multiplicative constant. Finally, taking the magnitude of the Fourier transform allows to undo the scaling of the time variable. This yields the *modified heat kernel* of the form

$$\hat{h}_\omega(x,x') = \left|\mathscr{F}\left\{\frac{\partial \log h_t(x,x')}{\partial \log t}\right\}(\omega)\right|, \tag{2.5}$$

where $\omega$ denotes the frequency variable of the Fourier transform. The SI-HKS is obtained by replacing $h_t$ with $\hat{h}_\omega$ from (2.5), yielding $\mathbf{p}(x) = (\hat{h}_{\omega_1}(x,x),\cdots,\hat{h}_{\omega_d}(x,x))$, where $\omega_1,\cdots,\omega_d$ are some fixed frequency values.

## 2.2. Schrödinger equation and wave kernel signatures

Aubry *et al.* [4] proposed a different physical model, considering a quantum mechanical particle on the manifold governed by the *Schrödinger equation*,

$$\left(i\Delta + \frac{\partial}{\partial t}\right)\psi(x,t) = 0, \tag{2.6}$$

where $\psi(x,t)$ denotes the complex wave function. The squared absolute value of the wave function $|\psi(x,t)|^2$ represents the probability to find the particle at point $x$ at time $t$.

Let us assume the quantum particle has initial energy distribution $f(e)$. Since energy is directly related to frequency (i.e., the eigenvalues of the Laplace-Beltrami operator), we will denote $f(\lambda)$ with some abuse of physical terminology. The solution of the Schrödinger equation can be expressed in the spectral domain as

$$\psi(x,t) = \sum_{k\geq 1} e^{-i\lambda_k t} f(\lambda_k)\phi_k(x). \tag{2.7}$$

Integrating over all times, the average probability

$$p(x) = \lim_{T\to\infty}\frac{1}{T}\int_0^T |\psi(x,t)|^2 dt = \sum_{k\geq 1} f^2(\lambda_k)\phi_k^2(x), \tag{2.8}$$

to measure the particle at point $x$ is obtained. Fixing some family of energy distributions, e.g.,

$$f_e(\lambda) \propto \exp\left(-\frac{(\log e - \log\lambda)^2}{2\sigma^2}\right), \tag{2.9}$$

it is possible to compute the *wave kernel signature* (WKS) $\mathbf{p}(x) = (p_{e_1}(x), \cdots, p_{e_d}(x))$ for each point of the shape, where $e_1, \cdots, e_d$ are some fixed energy values.

## 2.3. Optimal kernels and descriptor learning

Observing Eqs. (2.4) and (2.8) from which the HKS and WKS descriptors are derived, one can see that both have the form

$$p(x) = \sum_{k\geq 1} K(\lambda_k)\phi_k^2(x), \tag{2.10}$$

and can be represented as a family of*filters* $K(\lambda)$ applied to the eigenvalues of the Laplace-Beltrami operator [1,8].

Aflalo *et al.* [1] and Bronstein [8] showed that a family of optimal filters can be constructed for a specific class of shapes an deformations from the spectral characteristics of the discriminative features (the "signal") and the influence introduced by the deformations (the "noise"). This construction resembles in its spirit the Wiener filter that passes frequency bands with high signal-to-noise ratio, while attenuating those where the signal content is low compared to the energy of the noise. While it is usually difficult to model these statistical properties axiomatically, the authors showed that they can be learned from examples using standard metric learning algorithms [47].

## 2.4. Point Feature Histograms

Besides descriptors based on diffusion geometry for meshes and point clouds there are other local descriptors that rely on the curvature and normals of the underlying surface, like spin images [30] and curvature maps [24]. Wahl *et al.* [50] proposed a descriptor that is based on a histogram of pairwise features at each point's k nearest neighbours. The basic features that are chosen are angles between normals and distances between points, but more features can be used moments and curvatures. The construction of the descriptor is described in detail in Section 4. Rusu *et al.* [44] showed that these local descriptors based on normal's information only are informative about the underlying surface and rigid transformations invariant. They also proposed a slight modification to the PFH descriptor (that they called FPFH [44]) in order to be able to calculate it efficiently while retaining its discriminative properties.

## 2.5. Bags of features

Ovsjanikov *et al.* [40] and Toldo *et al.* [49] proposed constructing global shape descriptors from local descriptors using the *bag of features* paradigm [46]. In this approach, a fixed "geometric vocabulary" $\{\mathbf{p}_1, \cdots, \mathbf{p}_v\}$ is computed by means of an off-line clustering of the descriptor space. Next, each point descriptor is represented in the vocabulary using vector quantization, yielding a point-wise $v$-dimensional distribution of the form

$$\pi(x) \propto e^{-\|\mathbf{p}(x)-\mathbf{p}_l\|^2/2\sigma^2}. \tag{2.11}$$

The distribution is normalized in such a way that the elements of $\pi(x)$ sum to one. In the case of $\sigma = 0$, hard vector quantization is used, and $\pi_l(x) = 1$ for $\mathbf{p}_l$ being the closest element of the geometric vocabulary to $\mathbf{p}(x)$ in the descriptor space, and zero elsewhere. The bag of features global shape descriptor is then computed as the histogram of quantized descriptors over the entire shape. of dimensionality $v$ is then computed as

$$\bar{\mathbf{p}}(X) = \int_X \pi(x) da. \tag{2.12}$$

In [12], this construction was extended to joint distributions of spatially-close descriptors, referred to as spatially-sensitive bags of features or "geometric expressions".

## 3. Partial matching

In what follows, we assume to be given two shapes $X$ and $Y$ with corresponding point-wise descriptor fields $\mathbf{p}$ and $\mathbf{q}$ defined on them (here we adopt the raw HKS descriptors, though their quantized variants or any other intrinsic point-wise descriptors can be used as well). Assuming that $Y$ is a part of an unknown shape that is intrinsically similar to $X$, we aim at finding a part $X' \subseteq X$ having the same area $A$ of $Y$ such that the integral shape descriptors computed on $X'$ and $Y$ coincide as closely as possible. In order to prevent the parts from being fragmented and irregular, we penalize for their boundary length. The entire problem can be expressed as minimization of the following energy functional

$$E(X') = \left\| \int_{X'} \mathbf{p} da - \overline{\mathbf{q}} \right\|^2 + \lambda_\mathrm{r} L(\partial X') \tag{3.1}$$

under the constraint $A(X') = A$, where $A$ denotes area and $\overline{\mathbf{q}} = \int_Y \mathbf{q} da$. The first term of the functional constitutes the data term while the second one is the regularity term whose influence is controlled by the parameter $\lambda_\mathrm{r}$.

Discretization of the above minimization problem with a crisp set $X'$ results in combinatorial complexity. To circumvent this difficulty, in [9, 10] it was proposed to relax the problem by replacing the crisp part $X'$ by a fuzzy membership function $u$ on $X$, replacing the functional $E$ by a generalization of the Mumford-Shah functional [39] to surfaces. Here, we adopt this relaxation as well as the approximation of the Mumford-Shah functional proposed by Ambrosio and Tortorelli [2]. This yields the problem of the form

$$\min_{u,\rho,\sigma} D(u) + \lambda_\mathrm{r} R(u;\rho) \quad \text{s.t.} \quad \int_X u da = A, \tag{3.2}$$

with the data term

$$D(u) = \left\| \int_X \mathbf{p} u da - \overline{\mathbf{q}} \right\|^2 \tag{3.3}$$

and the Ambrosio-Tortorelli regularity term

$$R(u;\rho) = \frac{\lambda_\mathrm{s}}{2} \int_X \rho^2 \|\nabla u\|^2 da + \lambda_\mathrm{b} \epsilon \int_X \|\nabla \rho\|^2 da + \frac{\lambda_\mathrm{b}}{4\epsilon} \int_X (1-\rho)^2 da, \tag{3.4}$$

where $\rho$ is the so-called phase field indicating the discontinuities of $u$, and $\epsilon > 0$ is a parameter.

The first term of $R$ above imposes piece-wise smoothness of the fuzzy part $u$ governed by the parameter $\lambda_\mathrm{s}$. By setting a sufficiently large $\lambda_\mathrm{s}$, the parts become approximately piece-wise constant as desired in the original crisp formulation (3.1). The second term of $R$ is analogous to the boundary length term in (3.1) and converges to the latter as $\epsilon \to 0$.

We minimize (3.2) using alternating minimization comprising the following two iteratively repeated steps:

**Step 1:** Fix $\rho$ and solve for $u$

$$\min_u \left\| \int_X \mathbf{p} u da - \overline{\mathbf{q}} \right\|^2 + \lambda_\mathrm{r} \frac{\lambda_\mathrm{s}}{2} \int_X \rho^2 \|\nabla u\|^2 da \quad \text{s.t.} \quad \int_X u da = A. \tag{3.5}$$

We initialize this step with $\rho = 1$ thus achieving the Tikhonov regularization from the first step!

**Step 2:** Fix the part $u$ and solve for $\rho$

$$\min_{\rho} \frac{\lambda_{\mathrm{s}}}{2} \int_X \rho^2 \|\nabla u\|^2 da + \lambda_{\mathrm{b}} \epsilon \int_X \|\nabla \rho\|^2 da + \frac{\lambda_{\mathrm{b}}}{4\epsilon} \int_X (1-\rho)^2 da. \tag{3.6}$$

## 4. Discretization and numerical aspects

We represent a compact manifold $X$ isometrically embedded in $R^3$, as the sample points $\{\mathbf{x}_1, \cdots, \mathbf{x}_m\}$ on $X$, and denote by $\mathbf{a} = (a_1, \cdots, a_m)^{\mathrm{T}}$ the corresponding area elements at each point (the computation of the $a_i$'s is described later). $\mathbf{A} = \mathrm{diag}\{\mathbf{a}\}$ denote the diagonal $m \times m$ matrix created out of $\mathbf{a}$. The membership function $u$ is sampled at each point and represented as the vector $\mathbf{u} = (u_1, \cdots, u_m)^{\mathrm{T}}$. Similarly, the phase field is represented as the vector $\boldsymbol{\rho} = (\rho_1, \cdots, \rho_m)^{\mathrm{T}}$.

We quantify the sampling density of the underlying smooth manifold in terms of its extrinsic geometry. For that purpose, we define the *medial axis* of $X$ as the closure of the set of points that have at least two closest points in $X$. For any point $\mathbf{x} \in X$, the *local feature size* at $\mathbf{x}$, is the closest distance from $\mathbf{x}$ to the medial axis of $X$. The reach $\varrho$ of $X$ is the infimum of the local feature size over entire $X$. We say that the point cloud $\{\mathbf{x}_1, \cdots, \mathbf{x}_m\}$ is an $\varepsilon$-sampling of $X$ if for any point $\mathbf{y} \in X$, there exists $\mathbf{x}_i$ such that $\|\mathbf{y} - \mathbf{x}_i\| \leq \varepsilon \varrho$. The point cloud is an $(\varepsilon, \delta)$-sampling of $X$ if it is an $\varepsilon$-sampling of $X$ and for every $\mathbf{x}_i, \mathbf{x}_j$, $\|\mathbf{x}_j - \mathbf{x}_i\| \geq \delta \varrho$. In other words, a $(\varepsilon,\, \delta)$-sampling of $X$ is a $\delta\varrho$-separated $\varepsilon\varrho$-net in it.

### 4.1. Descriptors

We tried several descriptors including HKS, SI-HKS, WKS, PFH, FPFH. We have seen best results using HKS, PFH and FPFH.

**Heat Kernel Signatures (HKS)** The computation of the discrete heat kernel $h_t(\mathbf{x}_i, \mathbf{x}_j)$ requires computing discrete eigenvalues and eigenfunctions of the discrete Laplace-Beltrami operator. The latter can be computed directly using the finite elements method (FEM) [43], or by discretization of the Laplace operator on the point cloud followed by its eigendecomposition. Here, we adopt the latter approach following the spirit of the point cloud data (PCD) Laplacian discretizations [6, 35]. Both approaches are based on the findings of Belkin *et al.* [5] that the Laplace operator can be approximated using the Gaussian kernel in the ambient Euclidean space,

$$\Delta f(\mathbf{x}) = \lim_{h \to 0} \frac{1}{4\pi h^2} \int_X \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{4h}\right) \big(f(x) - f(y)\big)\, da(y), \tag{4.1}$$

where $f$ is a scalar function defined on the point cloud and $da$ is the standard area measure of the underlying manifold (volume form induced by the Riemannian structure). Thus, in the discrete setting it is needed to approximate the above integral on a point cloud. It is shown in [18] that $\int_X g da$ can be approximated by the discrete sum $\sum_{i=1}^{m} g(\mathbf{y}_i) a_i$ with the $a_i$'s being the area measure elements calculated by projecting $\mathbf{x}_i$'s neighbouring points to its local tangent plane and taking the area of the Voronoi cell containing $\mathbf{x}_i$. It is shown in [18] that for an $(\varepsilon, \delta)$-sampling and $h = \varepsilon^{1/(2+\xi)}$ with any positive constant $\xi$,

$$\left\| \sum_{i=1}^{m} g(\mathbf{y}_i) a_i - \int_X g(y) d\mu(y) \right\|_\infty = \mathcal{O}(\varepsilon + \varepsilon^3/\delta^2), \tag{4.2}$$

implying that for $\delta = \Omega(\varepsilon^{3/2-\xi})$ with any positive constant $\xi$, $\sum_{i=1}^{m} g(\mathbf{y}_i) a_i$ converges to $\int_X g(y) da(y)$ in the sense of the $L_\infty$ norm as $\varepsilon \downarrow 0$.

Rewriting Eq. (4.1) in discrete form we get,

$$(\Delta f)_i = \sum_j w_{ij} (f_i - f_j) a_j \,, \tag{4.3}$$

where $f_i = f(\mathbf{x}_i)$, and $w_{ij}$ are weights defined as

$$w_{ij} = \frac{1}{4\pi h^2} \exp\left( -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{4h} \right) \tag{4.4}$$

for $\mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i)$, and vanish elsewhere. Here, as the neighborhood $\mathcal{N}(\mathbf{x}_i)$ we use the $k$-th nearest neighbors of $\mathbf{x}_i$ (we used $k = 8$ and $h$ set to square root of the average distance between samples in all the neighborhoods).

The area elements $a_i$ are calculated in the following way: At each point $\mathbf{x}_i$, a tangent plane to the surface was constructed by fitting a plane to a local neighbourhood of $\mathbf{x}_i$ using the algorithm described in [27]. Then, $\mathbf{x}_i$'s neighbouring points were projected to the tangent plane, and triangulated using Delaunay triangulation. The area elements $a_i$ were defined as one third of the resulting $\mathbf{x}_i$ 1-ring area.In case the point cloud is endowed with a known mesh structure, it can be used instead in the calculation of the area elements.

In matrix notation, (4.3) can be written as $\Delta \mathbf{f} = \mathbf{L}\mathbf{f}$, where $\mathbf{L}$ is the $m \times m$ discretized Laplace-Beltrami operator, and $\mathbf{f} = (f_1, \cdots, f_m)^{\mathrm{T}}$. As by construction of $\mathcal{N}(\mathbf{x}_i)$ the weights $w_{ij}$ are not symmetric, we use the symmetrized version $\tilde{w}_{ij} = \frac{1}{2}(w_{ij} + w_{ji})$. Then, we define $\mathbf{W} = \left(\mathrm{diag}\{\sum_{l \neq j} \tilde{w}_{il}\} - \tilde{w}_{ij}\right)$ obtaining a symmetric matrix $\mathbf{W}$.

The discrete eigenfunctions and eigenvalues of $\mathbf{L}$ are found by solving the *generalized eigendecomposition* [34]

$$\mathbf{W}\boldsymbol{\Psi} = \mathbf{A}^{-1}\boldsymbol{\Psi}\boldsymbol{\Lambda}, \tag{4.5}$$

where $\mathbf{A} = \mathrm{diag}\{a_i\}$, $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_l\}$ is the diagonal matrix of the generalized eigenvalues, and $\boldsymbol{\Psi} = (\psi_j(x_i))$ is the matrix of the corresponding eigenvectors. It is easy to verify that every generalized eigenvalue $\lambda_j$ is also an eigenvalue of the Laplacian $\mathbf{L}$ with the corresponding eigenvector $\boldsymbol{\phi}_j = \mathbf{A}^{-1}\psi_j$. Once the eigendecomposition of the Laplacian has been discretized, the HKS descriptor is calculated according to (2.4). Other spectral descriptors are computed in a similar way. In our experiments we used 300 eigenvalues, and normalized the HKS descriptor vector.

**Point Feature Histogram (PFH)** PFH at point $\mathbf{x}_l$ is constructed by calculating a histogram of angular variations of normals between all pairs of points in the local neighbourhood $\mathcal{N}(\mathbf{x}_l)$. The process consists of the following steps. Normals $\mathbf{n}_l$, are calculated for each point $\mathbf{x}_l$ using the algorithm described in [27]. Then, for each pair $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_l)$ and their estimated normals $\mathbf{n}_i, \mathbf{n}_j$, where $\mathbf{x}_i$ is chosen to be the point with the smaller angle between $\mathbf{n}_i$ and $\mathbf{x}_j - \mathbf{x}_i$. Define

$$\mathbf{u} = \mathbf{n}_i, \quad \mathbf{v} = (\mathbf{x}_j - \mathbf{x}_i) \times \mathbf{u}, \quad \mathbf{w} = \mathbf{u} \times \mathbf{v}. \tag{4.6}$$

And the pairwise feature $\mathbf{F^l_{ij}}$ as

$$\alpha = \arctan(\mathbf{w} \cdot \mathbf{n}_j, \mathbf{u} \cdot \mathbf{n}_i), \quad \beta = \mathbf{v} \cdot \mathbf{n}_j\ , \tag{4.7a}$$

$$\gamma = \mathbf{u} \cdot \frac{\mathbf{x}_j - \mathbf{x}_i}{||\mathbf{x}_j - \mathbf{x}_i||_2}. \tag{4.7b}$$

Finally, the set $\{\mathbf{F^l_{ij}}\}_{\mathrm{ij} \in \mathcal{N}(\mathrm{x_l})}$ is binned into a histogram. Since the coordinates of $\mathbf{F^l}$ are measures of angles between normals. It is easy to divide each coordinate into same number ($b$) of bins. We chose $b = 5$, thus resulting in $5^3 = 125$ fully correlated feature space.

**Fast Point Feature Histogram (FPFH)** The FPFH [44] is a more relaxed version of PFH that is faster to compute. Instead of computing the histogram in the $b^3$ space, the histograms of $\alpha, \beta, \gamma$ are calculated separately and than concatenated to form a descriptor. The feature $\mathbf{F^l}$ (4.7) is not calculated for every pair in $\mathcal{N}(\mathbf{x}_l)$ but only between $\mathbf{x}_l$ and other points in $\mathcal{N}(\mathbf{x}_l)$. A histogram as described above is calculated only from these points and the resulting descriptor is called $SPFH(\mathbf{x}_l)$ from these descriptors the FPFH is constructed using the following formula

$$FPFH(\mathbf{x}_l) = SPFH(\mathbf{x}_l) + \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_l)} \frac{1}{w_i} SPFH(\mathbf{x}_i), \tag{4.8}$$

where the weight $w_i$ represents the distance between $\mathbf{x}_l$ and $\mathbf{x}_i$. The number of bins chosen for each coordinate of $\mathbf{F^l}$ was 11 thus resulting in descriptor of length 33.

Both FPFH and FPH are rotation and translation invariant thus are better choice for matching rigid shapes as they are faster to calculate than the HKS descriptor less sensitive to boundary effects and from our tests produce good results.

## 4.2. Data term

We define $\mathbf{P}$ as the $d \times m$ matrix of point-wise descriptors on $X$ (stored in columns), thus the sum of all the descriptors according to the membership function $u$ can be written as,

$$\int_X \mathbf{p} u da \approx \mathbf{PAu}. \tag{4.9}$$

Hence, data term (3.3) discretization takes the following form

$$D(\mathbf{u}) = ||\mathbf{PAu} - \overline{\mathbf{q}}||^2 = \mathbf{u}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{PAu} - 2\overline{\mathbf{q}}^{\mathrm{T}}\mathbf{PAu} + \overline{\mathbf{q}}^{\mathrm{T}}\overline{\mathbf{q}}. \tag{4.10}$$

### 4.3. Gradient norm

Let us denote by $\mathcal{N}(\mathbf{x}_i)$ the 1-ring of some point $\mathbf{x}_i$ on the shape formed by $t$ vertices $\mathbf{x}_1, \cdots, \mathbf{x}_t$ ordered e.g. in clock-wise order (to simplify notation, we assume without loss of generality consecutive indices). For meshes, the 1-ring or $\mathbf{x}_i$ is defined by all the vertices of triangles that share $\mathbf{x}_i$. For point clouds, the 1-ring is calculated by estimating the tangent plane at $\mathbf{x}_i$, projecting its neighbours to the plane, performing Delunay triangulation on the plane and selecting the vertices that resulted being in the 1-ring on the plane. Afterwards, we pick some $j$-th triangle in $\mathcal{N}(\mathbf{x}_i)$ formed by the central vertex $\mathbf{x}_i$ and the vertices $\mathbf{x}_j$ and $\mathbf{x}_k$ for $k = j \mod t + 1$. Let $\mathbf{X}_j = (\mathbf{x}_j - \mathbf{x}_i, \mathbf{x}_k - \mathbf{x}_i)$ be the vectors describing the triangle, hence, $\mathbf{X}_j$ is a $3 \times 2$ matrix. We calculate triangle's area from $\mathbf{X}_j$ by defining $\alpha_j = \frac{1}{2}\sqrt{\det(\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)}$.

In order to calculate, the gradient of the function $u$ on triangle $j$. We first, construct a matrix $\mathbf{D}_j$ that calculates the difference of values of $u$ on the vertices of triangle $j$ i.e. $\mathbf{D}_j\mathbf{u} = (u_j - u_i, u_k - u_i)^{\mathrm{T}}$. To do this we set $\mathbf{D}_j$ to be a sparse $2 \times m$ matrix with $+1$ at indices $(1, j)$ and $(2, k)$, and $-1$ at $(1, i)$ and $(2, i)$. The membership function $u$ at a point $\mathbf{x}$ within the triangle can be expressed in the above notation as $u(\mathbf{x}) = (\mathbf{D}_j\mathbf{u})^{\mathrm{T}}\boldsymbol{\delta}$, where $\boldsymbol{\delta} = (\delta_1, \delta_2)^{\mathrm{T}}$ is the vector of barycentric coordinates of the point $\mathbf{x}$, such that $\mathbf{x} = \mathbf{X}_j\boldsymbol{\delta}$. Since the function $u(\mathbf{x})$ is linear within the triangle, its gradient is constant and its norm can be expressed as

$$\|\nabla u\|^2 = \max \frac{(\mathbf{D}_j\mathbf{u})^{\mathrm{T}}\mathbf{d}\boldsymbol{\delta}}{\mathbf{dx}} = \max_{\|\mathbf{dx}\|=1} (\mathbf{D}_j\mathbf{u})^{\mathrm{T}}\mathbf{d}\boldsymbol{\delta}, \tag{4.11}$$

where $\mathbf{d}\boldsymbol{\delta}$ is a small displacement in barycentric coordinates, and $\mathbf{dx}$ is the corresponding displacement on the triangle. Observing that $\mathbf{dx} = \mathbf{X}_j\mathbf{d}\boldsymbol{\delta}$, by multiplying both sides by the pseudo-inverse of $\mathbf{X}_j$ one has $\mathbf{d}\boldsymbol{\delta} = (\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)^{-1}\mathbf{X}_j^{\mathrm{T}}\mathbf{dx}$, yielding

$$\|\nabla u\|^2 = \max_{\|\mathbf{dx}\|=1} (\mathbf{D}_j\mathbf{u})^{\mathrm{T}}(\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)^{-1}\mathbf{X}_j^{\mathrm{T}}\mathbf{dx} = (\mathbf{D}_j\mathbf{u})^{\mathrm{T}}(\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)^{-1}\mathbf{D}_j\mathbf{u}. \tag{4.12}$$

The latter can be in turn cast as the norm $\|\nabla u\|^2 = \mathbf{g}_j^{\mathrm{T}}\mathbf{g}_j$ of the gradient vector

$$\mathbf{g}_j = (\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)^{-1/2}\mathbf{D}_j\mathbf{u} = \mathbf{E}_j\mathbf{u}. \tag{4.13}$$

The area element corresponding to $\mathbf{x}_i$ is given by $a_i = \frac{1}{3}(\alpha_1 + \cdots + \alpha_t)$, and finally the gradient at $\mathbf{x}_i$ can be expressed by averaging the $\mathbf{g}_j$'s with the weights $\alpha_j$. This yields

$$\begin{aligned}\nabla u\, da &\approx \frac{1}{3}\sum_{j=1}^{t} \alpha_j\mathbf{g}_j = \frac{1}{3}\sum_{j=1}^{t} \alpha_j\mathbf{E}_j\mathbf{u} = \frac{1}{3}\mathbf{1}^{\mathrm{T}}\begin{pmatrix}\alpha_1\mathbf{E}_1\\ \vdots \\ \alpha_t\mathbf{E}_t\end{pmatrix}\mathbf{u}\\ &= \frac{1}{3}((\alpha_1, \cdots, \alpha_t) \otimes \mathbf{I})\mathbf{Eu},\end{aligned} \tag{4.14}$$

where $\mathbf{1}$ is a $2t \times 1$ vector of ones, $\mathbf{E}$ is the $2t \times m$ matrix stacking $\mathbf{E}_j$'s, $\mathbf{I}$ is the $2 \times 2$ identity matrix, and $\otimes$ denotes the Kroenecker product $(\alpha_1, \cdots, \alpha_t) \otimes \mathbf{I} = (\alpha_1 \mathbf{I}, \cdots, \alpha_t \mathbf{I})$. Denoting by $\mathbf{G}_i$ the $2 \times m$ matrix $\frac{1}{3}((\alpha_1, \cdots, \alpha_t) \otimes \mathbf{I})\mathbf{E}$ corresponding to the vertex $\mathbf{x}_i$, we can write $\nabla u\, da \approx \mathbf{G}_i \mathbf{u}$.

Let us now consider all the points of the shape. We have

$$\int_X \rho^2 \|\nabla u\|^2 da \approx \sum_{i=1}^m \frac{\rho_i^2}{a_i} \mathbf{u}^{\mathrm{T}} \mathbf{G}_i^{\mathrm{T}} \mathbf{G}_i \mathbf{u}. \tag{4.15}$$

Introducing a $2m \times m$ matrix

$$\mathbf{G} = \left( \operatorname{diag} \left\{ \frac{1}{\sqrt{a_1}}, \cdots, \frac{1}{\sqrt{a_m}} \right\} \otimes \mathbf{I} \right) \begin{pmatrix} \mathbf{G}_1 \\ \vdots \\ \mathbf{G}_m \end{pmatrix} \tag{4.16}$$

allows to rewrite the former integral as

$$\int_X \rho^2 \|\nabla u\|^2 da \approx \|(\operatorname{diag}\{\boldsymbol{\rho}\} \otimes \mathbf{I}) \mathbf{G} \mathbf{u}\|^2. \tag{4.17}$$

## 4.4. Discretized alternating minimization

Using the tools we described above we can finally construct the two steps of the alternating minimization problem (3.5)-(3.6). The first step of the discretized minimization problem (3.5), is preformed when $\boldsymbol{\rho}$ is fixed, and optimal $u$ is being searched for that $\boldsymbol{\rho}$

$$\min_{\mathbf{u}} \mathbf{u}^{\mathrm{T}} \left( \mathbf{A}^{\mathrm{T}} \mathbf{P}^{\mathrm{T}} \mathbf{P} \mathbf{A} + \lambda_{\mathrm{r}} \frac{\lambda_{\mathrm{s}}}{2} \mathbf{G}^{\mathrm{T}} (\operatorname{diag}\{\boldsymbol{\rho}^2\} \otimes \mathbf{I}) \mathbf{G} \right) \mathbf{u} - 2 \overline{\mathbf{q}}^{\mathrm{T}} \mathbf{P} \mathbf{A} \mathbf{u} \quad \text{s.t.} \quad \mathbf{a}^{\mathrm{T}} \mathbf{u} = A. \tag{4.18}$$

The second step is preformed when $\mathbf{u}$ is fixed, thus yielding the optimal $\boldsymbol{\rho}$. Recall that, at a vertex $\mathbf{x}_i$,

$$\|\nabla u\|^2 da \approx \frac{1}{a_i} \mathbf{u}^{\mathrm{T}} \mathbf{G}_i^{\mathrm{T}} \mathbf{G}_i \mathbf{u} = s_i \tag{4.19}$$

Using this notation, we obtain the following discretization of the integrals in the regularization term

$$\int_X \rho^2 \|\nabla u\|^2 da \approx \sum_{i=1}^m \rho_i^2 s_i = \boldsymbol{\rho}^{\mathrm{T}} \operatorname{diag}\{s_1, \cdots, s_m\} \boldsymbol{\rho} \;=\; \boldsymbol{\rho}^{\mathrm{T}} \mathbf{S}(\mathbf{u}) \boldsymbol{\rho}, \tag{4.20}$$

$$\int_X \|\nabla \rho\|^2 da \approx \|\mathbf{G} \boldsymbol{\rho}\|^2 = \boldsymbol{\rho}^{\mathrm{T}} \mathbf{G}^{\mathrm{T}} \mathbf{G} \boldsymbol{\rho}, \tag{4.21}$$

$$\int_X (1 - \rho)^2 da \approx \boldsymbol{\rho}^{\mathrm{T}} \mathbf{A} \boldsymbol{\rho} - 2 \mathbf{a}^{\mathrm{T}} \boldsymbol{\rho} + \mathbf{1}^{\mathrm{T}} \mathbf{a}. \tag{4.22}$$

The discretized minimization problem (3.6) with respect to $\boldsymbol{\rho}$ becomes

$$\min_{\rho} \boldsymbol{\rho}^{\mathrm{T}} \left( \frac{\lambda_{\mathrm{s}}}{2} \mathbf{S}(\mathbf{u}) + \lambda_{\mathrm{b}} \epsilon \mathbf{G}^{\mathrm{T}} \mathbf{G} + \frac{\lambda_{\mathrm{b}}}{4\epsilon} \mathbf{A} \right) \boldsymbol{\rho} - \frac{\lambda_{\mathrm{b}}}{2\epsilon} \mathbf{a}^{\mathrm{T}} \boldsymbol{\rho}. \tag{4.23}$$

This is an unconstrained quadratic problem, and has the following closed-form solution

$$\boldsymbol{\rho} = \left( 2 \frac{\lambda_{\mathrm{s}} \epsilon}{\lambda_{\mathrm{b}}} \mathbf{S}(\mathbf{u}) + 4\epsilon^2 \mathbf{G}^{\mathrm{T}} \mathbf{G} + \mathbf{A} \right)^{-1} \mathbf{a}. \tag{4.24}$$

## 5. Results

In order to test our approach, we performed several partial matching experiments on data from the SCAPE dataset [3][†], TOSCA dataset [15][‡], range scans from NIST SHREC'10 range dataset [23] and range scans obtained by the Kinect sensor. TOSCA set contains synthetic shapes undergoing simulated articulation transformations, SCAPE contains high-quality scans of real people assuming real poses, SHREC'10 range scans contains various models captured from arbitrary view directions. Groundtruth correspondence between the transformed shapes is available for the TOSCA dataset. Though some datasets contained triangulated data, in all experiments the processing was performed on the raw point clouds only.

In our experiments, all the shapes were downsampled to approximately 2500 vertices. Parts were cut by taking a geodesic circle of random radius around a random center point. For each part, the normalized HKS, PFH, FPFH descriptors were calculated at each point belonging to the part. To avoid boundary effects (Fig. 1), descriptors close to the boundary were ignored when calculating $\overline{\mathbf{q}}$ in (3.1). The distance from the boundary was selected in accordance to the time scales of the HKS and the sampling density. For small scale values the descriptors differ significantly between the full and partial shapes due to insufficient sample density. For large scale values, the influence of the boundary, affects the heat kernel at larger distances (Fig. 1). We used ten linearly spread samples of the heat kernel in the range $[30, 50]$ for the HKS, and neighbourhood of 8 (and radius 15) for the FPFH and the PFH descriptors, as visualized in Fig. 2. This range of time scales and neighbourhood sizes was found to be informative enough while minimizing the boundary effects.

We observed fast convergence of the alternating minimization procedure (Fig. 3). After only two iterations, the membership function $\mathbf{u}$ typically ceased changing significantly. This is due to the initialization of step one with $\rho = 1$, and achieving the Tikhonov regularization on the first step, hence, having a smooth solution in the very beginning of the process. The phase map $\rho$ assumed the values close to 1 in places of low gradient of the membership function $\mathbf{u}$, and less than 1 in high gradient areas (Fig. 3). The importance of the regularization step is evident observing the change in $\mathbf{u}$ in Fig. 3.

Fig. 4 shows the influence of the parameter $\lambda_{\mathrm{r}}$, controlling the impact of the regularization. For too small values of $\lambda_{\mathrm{r}}$, no good match is obtained because the searched part is

[†] Available online at `http://ai.stanford.edu/~drago/Projects/scape/`

[‡] Available online at `http://tosca.cs.technion.ac.il`

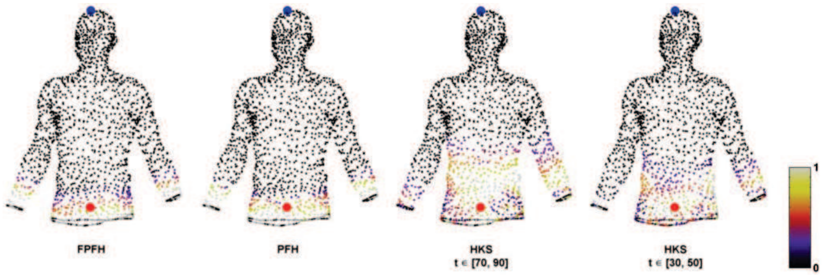

Figure 1: The $L_2$ difference between descriptors computed on the full shape and the partial shape. Note that the difference is maximal on the boundary decaying away from it; the error decay speed depends on the neighbourhood size chosen for computation of the features. For the HKS it also depends on the chosen scale.

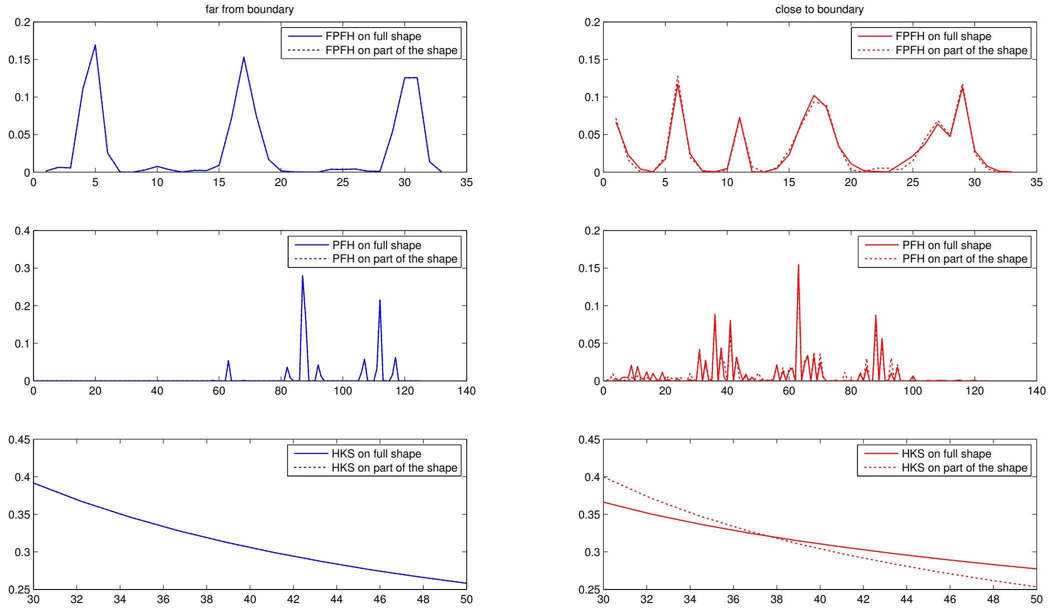

Figure 2: A comparison of the descriptors at different locations, when calculated on part of shape (dashed line) and on full shape (solid line). The blue and red colors refer to points far and close to the boundary (see Fig. 1).

not exactly the same in the shape. As the influence of the regularization increase the membership functions becomes smoother and closer to the correct match. Matches obtained due to symmetry decrease. However, increasing $\lambda_r$ further causes incorrect matching due to low data term influence. Increasing it even more starts smoothing the result (rightmost column) until eventually making the membership function uniform over the entire shape.

Fig. 5 shows examples of matching results using the FPFH and HKS descriptors. It can be seen that due to invariance of the descriptors to transformations the symmetric parts in the shape are also matched. This phenomena is greater for the HKS descriptor as it is also invariant to isometric transformations. In order to visualize this we've adjusted the

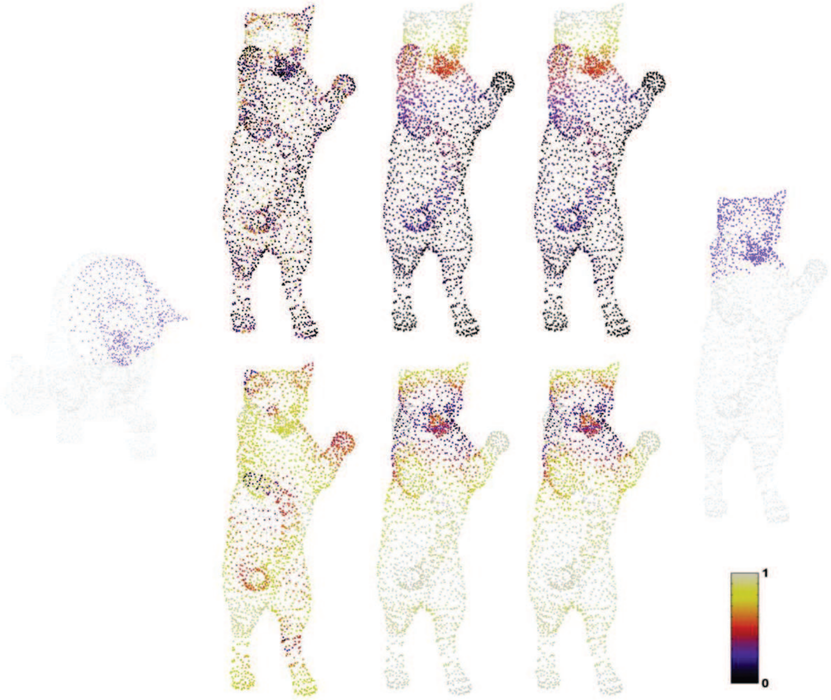

Figure 3: Convergence of the alternating minimization procedure. Depicted are the membership function $\mathbf{u}$ (top row) and the phase field $\rho$ (bottom row). The first column displays result of step one when $\rho$ is initialized to zero (no regularization at all). The second column is the result of the first step when $\rho = 1$ is used as the initialization. The third column depicts the result after completing two full iterations of the alternating minimization.

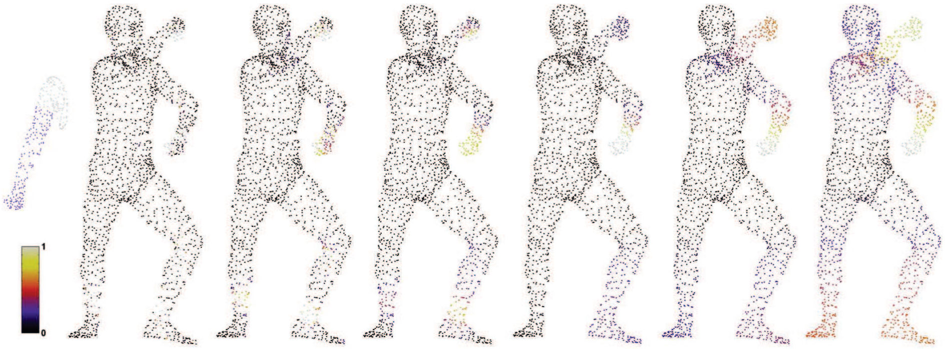

Figure 4: The influence of the parameter $\lambda_r$, controlling the impact of regularization. The leftmost figure depicts a query part; figures on the right are the membership function $u$ for increasing values of $\lambda_r$.

threshold for the membership function to be 0.25. By adjusting the regularization term correctly it is possible to remove the matching of similar as parts as in Fig. 4. Throughout the experiments we didn't notice much difference in matching results between the FPFH and PFH descriptors.

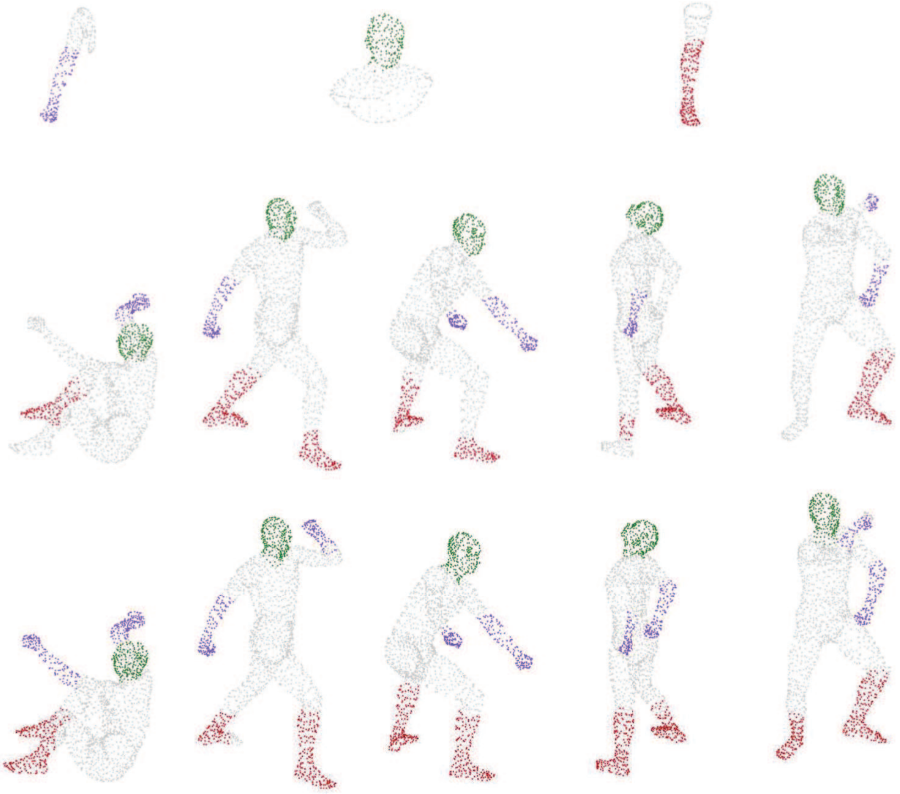

Figure 5: Examples of matching of random parts of SCAPE human shapes (first row) to approximately isometric deformations of the shapes using the FPFH descriptor (second row) and the HKS descriptor third row. Color code indicates different parts.

Fig. 6 shows the result of the membership function when trying to match range scans representing an object captured from different views. Using the FPFH descriptor the method matched a side facing head to a front facing head and other side facing head. It is interesting to see that although the front side differ in area, has occlusion of points due to the viewing angle, the method still matched it due to regularization and rough similarities of the descriptors on the head scanned from different angles. Same experiment with the HKS descriptor produced poor results this is due to high influence of the boundary effect on the HKS descriptor.

Figs. 7-9 show examples of matching results after thresholding. The membership function **u** was thresholded in such a way to make the resulting area as close as possible to the area of the query region.

The robustness of the method is shown in Figs. 6-9, where it is capable to find correct matching even if the shape undergoes deformations or is degraded by geometric and topological noise (Fig. 9). Furthermore, correct matching is obtained between shapes that have only roughly similar features (e.g. male and female shapes, or shapes from TOSCA and SCAPE datasets, scans from different angles), as can be seen in Figs. 6 and 8. This

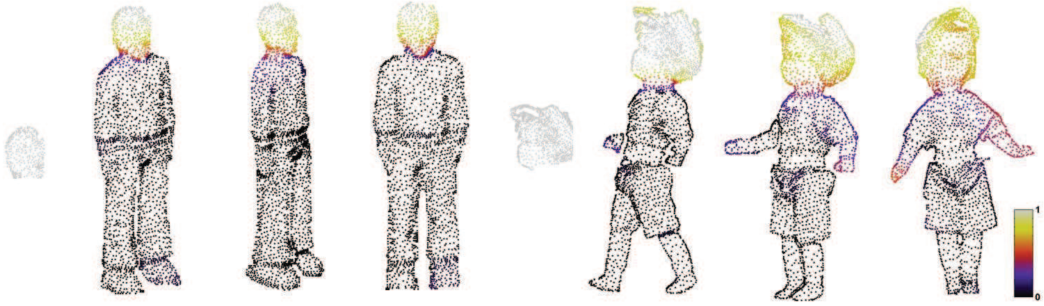

Figure 6: The resulting membership function for two experiments trying to match three range scans of the same shape from different angles. For both shapes the top part of the side facing shape was taken as reference (first and fourth column). Than it was matched to the original shape (second and fifth column), to shape scanned from the other side (third and sixth column), and to the front facing scan (fourth and seventh column).

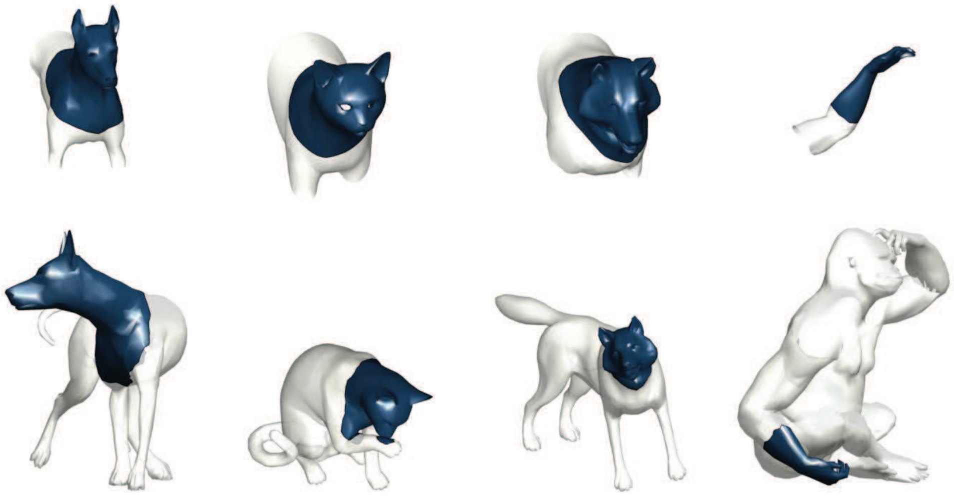

Figure 7: Examples of matching of random parts of TOSCA animal shapes (first row) to approximately isometric deformations of the shapes (second row).

behavior largely depends on the typical feature size, which is determined by selecting the scale parameter of the HKS descriptor and the neighbourhood sizes of the PFH and FPFH descriptors.

Table 1 summarizes a quantitative evaluation that was performed on a subset of the TOSCA database, for which ground truth correspondence and its bilaterally symmetric counterpart are available. This subset included a male, a dog and a horse shape classes with different geometric, topological and noise deformations (98 shapes in total). The query set was generated by selecting a part from a deformed shape (1000 queries in each deformation category) and matched to the null shape with parameters and thresholds as described above.

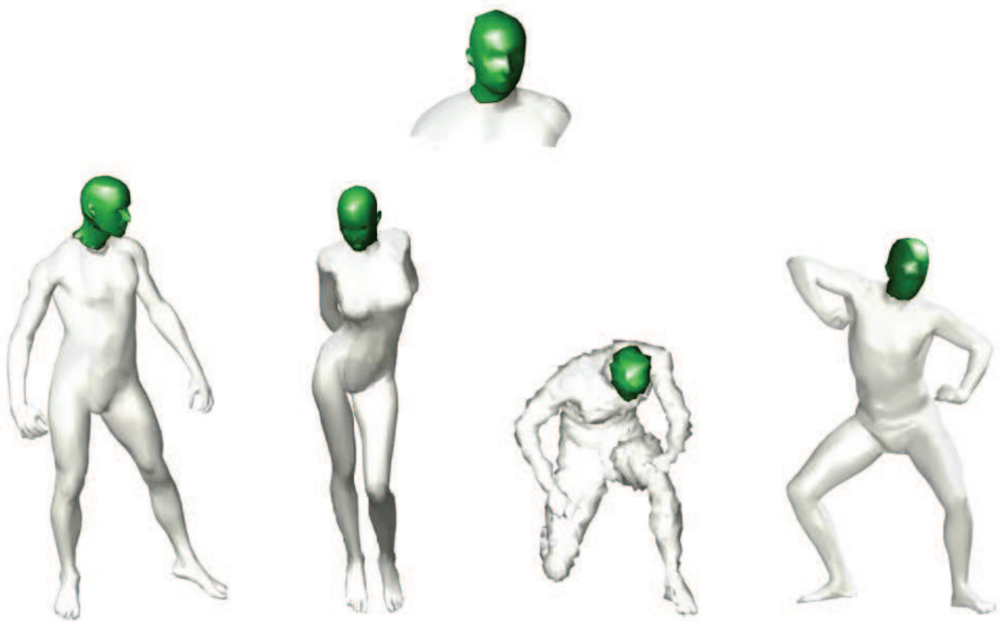

Figure 8: Results of matching a part of shape to other shapes that contain a similar part. Shown a head from SCAPE shape matched to heads of TOSCA shapes (second row, first to third columns) and SCAPE shape (second row, fourth column).

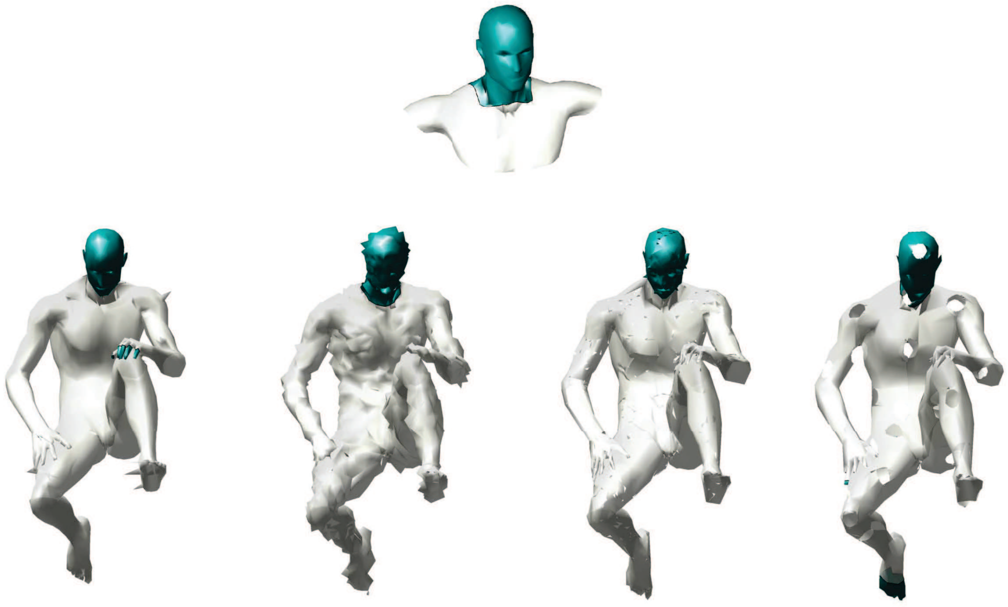

Figure 9: Results of matching a part of TOSCA shape (first row) to shapes distorted by different transformations (second row). Shown left-to-right are: shot noise, noise, micro holes and holes.

Table 1: Part matching performance on transformed shapes from the TOSCA database. At each query a random part (location and size) was selected from a deformed shape and matched to the null shape. Overlap is reported compared to the groundtruth correspondence between the shapes (in parentheses taking into account the intrinsic bilateral symmetry).

| Transformation | Queries | Avg. overlap |
|---|---|---|
| Isometry | 1000 | 75% (85%) |
| Isometry + Shotnoise | 1000 | 75% (85%) |
| Isometry + Noise | 1000 | 71% (82%) |
| Isometry + Microholes | 1000 | 68% (82%) |
| Isometry + Holes | 1000 | 66% (76%) |
| All | 5000 | 71% (82%) |

### 5.1. Complexity

The code used in the experiments was implemented in Matalb with time-critical parts written in C with the Mex interface. The quadratic programming problem (3.5) in Step 1 was solved using the QPC[§] solver implementing a dual active set method. The experiments were run on a 2.3GHz Intel Core2 Quad CPU, 2GB RAM in Win7 32bit environment. The running time (including re-sampling and descriptor calculation) per part was about 40–120 seconds; a detailed breakdown of the running time is summarized in Table 2.

Table 2: Average runtime (in seconds) for different stages in the proposed method: Laplacian construction, eigendecomposition, cost matrix generation for the QP solver, and two iterations of the minimization process. Note, that for FPFH/PFH descriptors we don't need the Laplacian and the eigen values decomposition.

| Points | Laplacian | Eigen. decomp. | QP generation | Minimization |
|---|---|---|---|---|
| 1000 | 1.06 | 5.15 | 3.31 | 4.36 |
| 2000 | 1.74 | 19.92 | 9.61 | 18.15 |
| 3000 | 2.57 | 39.45 | 19.45 | 47.04 |

## 6. Conclusions

We presented a framework for finding partial similarity between range scans, point clouds and shapes which does not rely on explicit correspondence. The method is based on regularized matching of region-wise local descriptors, and can be efficiently implemented. Experimental results show that our approach performs well with different kind of descriptors and in challenging matching scenarios, such as the presence of geometric and topological noise and occlusions. In the future work, we will efficiently implement this method and test in on full cluttered scenes, then extend the method to the setting of two partially-similar full shapes, in which two similar parts have to be found in each shape, and then consider a multi-part matching (puzzle) scenario.

[§] Available online at `http://sigpromu.org/quadprog`

**Acknowledgments** The author would like to thank the referees for the helpful suggestions. This work has been supported in part by the Israeli Science Foundation grant 615/11, the German-Israeli Foundation grant 2269/2010, and the Swiss High Performance and High Productivity Computing (HP2C) grant.

## References

[1] J. Aflalo, A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Deformable shape retrieval by learning diffusion kernels. In *Proc. SSVM*, 2011.

[2] L. Ambrosio and V.M. Tortorelli. Approximation of functionals depending on jumps by elliptic functionals via-convergence. *Comm. Pure Appl. Math*, 43(8):999–1036, 1990.

[3] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. In *Proceedings of the SIGGRAPH Conference*, 2005.

[4] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *Proc. Workshop 4DMOD*, 2011.

[5] M. Belkin and P. Niyogi. Towards a theoretical foundation for laplacian-based manifold methods. *COLT*, pages 486–500, 2005.

[6] M. Belkin, J. Sun, and Y. Wang. Constructing laplace operator from point clouds in $r^d$. In *20th ACM-SIAM Sympos. Discrete Algorithms*, pages 1031–1040. ACM-SIAM, 2009.

[7] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *Trans. PAMI*, 14:239–256, 1992.

[8] A. M. Bronstein. Spectral descriptors for deformable shapes. Technical Report arXiv:1110.5015v1, 2011.

[9] A. M. Bronstein and M. M. Bronstein. Not only size matters: regularized partial matching of nonrigid shapes. In *Prof. NORDIA*, 2008.

[10] A. M. Bronstein and M. M. Bronstein. Regularized partial matching of rigid shapes. In *Proc. ECCV*, pages 143–154, 2008.

[11] A. M. Bronstein, M. M. Bronstein, Y. Carmon, and R. Kimmel. Partial similarity of shapes using a statistical significance measure. *IPSJ Trans. Computer Vision and Applications*, 1:105–114, 2009.

[12] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov. Shape google: geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics (TOG)*, 30(1):1, 2011.

[13] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Science (PNAS)*, 103(5):1168–1172, 2006.

[14] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Robust expression-invariant face recognition from partially missing data. In *Proc. European Conf. Computer Vision (ECCV)*, pages 396–408, 2006.

[15] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer-Verlag New York Inc, 2008.

[16] A.M. Bronstein, M.M. Bronstein, Y. Carmon, and R. Kimmel. Partial similarity of shapes using a statistical significance measure. *Trans. Computer Vision and Applications*, 1(0):105–114, 2009.

[17] M. M. Bronstein and I. Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. CVPR*, 2010.

[18] Luo C., Sun J., and Wang Y. Integral estimation from point cloud in d-dimensional space: A geometric view.

[19] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10(2):266–277, 2001.

[20] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proc. Conf. Robotics and Automation*, 1991.

[21] U. Clarenz, M. Rumpf, and A. Telea. Robust feature detection and local classification for surfaces based on moment analysis. *Trans. Visualization and Computer Graphics*, 10(5):516–524, 2004.

[22] C. Domokos and Z. Kato. Affine Puzzle: Realigning Deformed Object Fragments without Correspondences. In *Proc. ECCV*, pages 777–790, 2010.

[23] Helin Dutagaci, Afzal Godil, Chun Pan Cheung, Takahiko Furuya, Ulrich Hillenbrand, and Ryutarou Ohbuchi. Shrec'10 track: Range scan retrieval. In *3DOR*, pages 109–115, 2010.

[24] Timothy Gatzke, Cindy Grimm, Michael Garland, and Steve Zelinka. Curvature maps for local shape comparison. In *In Shape Modeling International*, pages 244–256, 2005.

[25] K. Gebal, J.A. Bærentzen, H. Aanæs, and R. Larsen. Shape analysis using the auto diffusion function. In *Computer Graphics Forum*, volume 28, pages 1405–1413, 2009.

[26] M. Gromov. *Structures Métriques Pour les Variétés Riemanniennes*. Number 1 in Textes Mathématiques. 1981.

[27] S. Har-Peled and K. R. Varadarajan. Projective clustering in high dimensions using core-sets. In *18th Annu.ACM Sympos. Comput. Geom.*, pages 312–318. ACM, 2002.

[28] Q.X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. *ACM Trans. Graphics*, 25(3):569–578, 2006.

[29] D. Jacobs, D. Weinshall, and Y. Gdalyahu. Class representation and image retrieval with non-metric distances. *Trans. PAMI*, 22(6):583–600, 2000.

[30] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *Trans. PAMI*, 21(5):433–449, 1999.

[31] P. W. Jones, M. Maggioni, and R. Schul. Manifold parametrizations by eigenfunctions of the Laplacian and heat kernels. *PNAS*, 105(6):1803, 2008.

[32] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, and R. Kimmel. Photometric heat kernel signature. In *Proc. Scale Space and Variational Methods (SSVM)*, 2011.

[33] L. J. Latecki, R. Lakaemper, and D. Wolter. Optimal Partial Shape Similarity. *Image and Vision Computing*, 23:227–236, 2005.

[34] B. Lévy. Laplace-Beltrami eigenfunctions towards an algorithm that "understands" geometry. In *Proc. Shape Modeling and Applications*, 2006.

[35] Chuanjiang Luo, Issam Safa, and Yusu Wang. Approximating gradients for meshes and point clouds via diffusion metric. In *Eurographics Symposium on Geometry Processing*, 2009.

[36] S. Manay, B.W. Hong, A.J. Yezzi, and S. Soatto. Integral invariant signatures. *Lecture Notes in Computer Science*, pages 87–99, 2004.

[37] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5:313–346, 2005.

[38] N. J. Mitra, L. J. Guibas, J. Giesen, and M. Pauly. Probabilistic fingerprints for shapes. In *Proc. SGP*, 2006.

[39] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42(5):577–685, 1989.

[40] M. Ovsjanikov, A.M. Bronstein, L.J. Guibas, and M.M. Bronstein. Shape Google: a computer vision approach to invariant shape retrieval. In *Proc. NORDIA*. Citeseer, 2009.

[41] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces.

In *Computer Graphics Forum*, volume 22, pages 281–289, 2003.

[42] J. Pokrass, A. M. Bronstein, and M. M. Bronstein. A correspondence-less approach to matching of deformable shapes. In *Proc. SSVM*, 2011.

[43] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-spectra as fingerprints for shape matching. In *Proc. ACM Symp. Solid and Physical Modeling*, pages 101–106, 2005.

[44] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast Point Feature Histograms (FPFH) for 3D Registration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, May 12-17 2009.

[45] I. Sipiran and B. Bustos. A robust 3D interest points detector based on Harris operator. In *Proc. 3DOR*, pages 7–14. Eurographics, 2010.

[46] J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. In *Proc. CVPR*, 2003.

[47] C. Strecha, A. M. Bronstein, M. M. Bronstein, and P. Fua. LDAHash: improved matching with smaller descriptors. 35(1):66–78, 2012.

[48] J. Sun, M. Ovsjanikov, and L. Guibas. A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. In *Computer Graphics Forum*, volume 28, pages 1383–1392, 2009.

[49] R. Toldo, U. Castellani, and A. Fusiello. Visual vocabulary signature for 3D object retrieval and partial matching. In *Proc. 3DOR*, 2009.

[50] E. Wahl, U. Hillenbrand, and G. Hirzinger. Surflet-pair-relation histograms: A statistical 3d-shape representation for rapid classification. In *3DIM03*, pages 474–481, 2003.

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

[52] C. Zhang and T. Chen. Efficient feature extraction for 2D/3D objects in mesh representation. In *Proc. ICIP*, volume 3, 2001.