

3D features, surface descriptors, and object descriptors

Alexander M. Bronstein Michael M. Bronstein
Maks Ovsjanikov

September 6, 2010

Abstract

The computer vision and pattern recognition communities have recently witnessed a surge of feature-based methods in numerous applications including object recognition and image retrieval. Similar concepts and analogous approaches are penetrating the world of 3D shape analysis, in a variety of areas including non-rigid shape retrieval and matching. In this chapter, we present the state-of-the-art of feature-based approaches in 3D shape analysis.

1 Introduction

In computer vision and pattern recognition jargon, the term *features* is often used to refer to persistent elements of a shape (such as corners or sharp edges), which capture most of the relevant information, and based on which one is able to perform object analysis. In the last decade, feature-based methods have become a standard and broadly used paradigm in various applications, including retrieval and matching (e.g. for multiview geometry reconstruction), due to their relative simplicity, flexibility and excellent per-

formance in practice. An important milestone was the introduction of scale invariant feature transform (SIFT) [31] and similar algorithms [34, 4].

The success of SIFT and similar approaches in a variety of practical scenarios, as well as the public availability of the software have made feature-based approaches a *de facto* standard in computer vision.

A similar trend is emerging in 3D shape analysis in a variety of areas including non-rigid shape retrieval and shape matching, where fundamental differences between 2D and 3D shapes make it difficult to apply existing techniques from computer vision.

Like in image analysis, two archetype applications are *correspondence*, where the goal is to find matches between points in two 3D shapes, and *similarity*, in which one has to quantify the degree of similarity or dissimilarity of two 3D shapes. The advantage of using features for correspondence problems is the ability to reliably identify similar points on two shapes, thereby reducing the set of potential correspondence candidates. In finding similarity, especially in large-scale shape search and retrieval applications, one of the strengths of feature-based approaches is that they allow to represent a shape as a collection of primitive elements (geometric or visual “words”), the same way as text can be represented as a collection of words, and use the well-developed methods from text search [47, 17].

One of the distinguishing characteristics that make computer vision techniques inapplicable in 3D shape processing is the difference in shape representations. In computer vision, it is common to work with an image of a physical object, representing both its geometric and photometric properties. Such a representation simplifies the task of shape analysis by reducing it to simple image processing operations, at the cost of losing information about the object’s 3D structure, which cannot be unambiguously captured in a 2D image. In computer graphics and geometry processing, it is assumed that the 3D geometry of the object is given. Depending on application, the geometric representation of the object can differ significantly. For example, in graphics it is common to work with triangular meshes or point clouds; in medical

applications with volumes and implicit representations.

Most feature-based approaches can be logically divided into two main stages: location of stable points that capture most of the relevant shape information (*feature detection*) and representation of the shape properties at these points (*feature description*). Both processes depend greatly on shape representation as well as on the application at hand.

In image analysis, the typical use of features is to describe an object independently of the way it is seen by a camera. Features found in images are geometric discontinuities in the captured object (edges and corners) or its photometric properties (texture). Since the difference in viewpoint can be locally approximated as an affine transformation, feature detectors and descriptors in images are usually made *affine invariant*.

In 3D shape analysis, features are typically based on geometry rather than appearance. The problems of shape correspondence and similarity require the features to be stable under natural transformations that an object can undergo, which may include not only changes in pose, but also non-rigid bending. If the deformation is inelastic, it is often referred to as *isometric* (distance-preserving), and feature-based methods coping with such transformations as *isometry-invariant*; if the bending also involves connectivity changes, the feature detection and description algorithms are called *topology-invariant*.

The main challenge of feature-based 3D shape analysis can be thus summarized as finding a set of features that are not susceptible to some class of shape transformations and carry sufficient information to allow using these features for finding correspondence and similarity, among other tasks. In this chapter, we present an overview of feature-based methods in 3D shape analysis and their applications, classical as well as most recent approaches, and future challenges.

2 Mathematical Background

For the remainder of the discussion, an object is some subset of the ambient Euclidean space, $\Omega \subset \mathbb{R}^3$. In many cases (e.g. data acquired by a range scanner), we can access only the *boundary* $\partial\Omega$ of the object, which can be modeled as a two-dimensional smooth *manifold* or *surface*, denoted here by X . Photometric information is given as a scalar or a vector field $\alpha : X \rightarrow \mathbb{R}^d$ on the manifold and referred to as *texture*. If the surface is sampled at some discrete set of points $\{x_1, \dots, x_N\} \subset X$ then this representation is called a *point cloud*; if in addition connectivity information is available in a form of a simplicial complex (*triangulation*, consisting of a set of *edges* $(x_i, x_j) \in E$ and *faces* $(x_i, x_j, x_k) \in F$), such a representation is called a *mesh*.

In medical applications, such as tomographic data analysis, information about the internal structure of the object in addition to its boundary is often available. A common representation in such applications is a *volumetric image*, which can be represented as a 3D matrix, where each voxel (3D pixel) describes the properties of the object (e.g. its penetrability by X-ray radiation). Segmentation algorithms applied to volumetric data used in medical applications often extract boundaries of 3D objects in *implicit form*, represented as level-sets of some function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$.

3 Feature detectors

The goal of a feature detector is to find stable points or regions on a shape. The main requirements of a feature detector are that the points that it selects are (i) *repeatable*, i.e., in two instances of a shape, ideally the same set of corresponding points is detected, and (ii) *informative*, i.e., descriptors built upon these points contain sufficient information to e.g. distinguish the shape from others.

Since there is no single way to define a feature, the construction of the detector depends very much on the shape representation and the application

at hand, or more specifically, the desired invariance properties.

3.1 A taxonomy

In the most trivial case, no feature detection is performed and the feature descriptor is computed at all the points of the shape (e.g. [13]) or at some densely sampled subset thereof. The descriptor in this case is usually termed *dense*. Dense descriptors bypass the problem of repeatability, at the price of increased computational cost and potentially introducing many unimportant points that clutter the shape representation.

Many detectors assume to be given some scalar- or vector-valued function defined on the surface. The function can be either *photometric* information (texture) or a *geometric* quantity such as curvature. With this concept in mind, feature detection on shapes resembles very much that in images, and many attempts to import methods from image processing and computer vision have been described in the literature. Several methods for feature detection have been inspired by the the *difference of Gaussians* (DOG), a classical feature detection approach used in computer vision. Zaharescu *et al.* [56] introduce the *mesh DOG* approach by first applying Gaussian filtering to scalar functions (e.g. mean or Gauss curvature) defined on the shape. This allows to represent the function in scale space, and feature points are prominent maxima of the scale space across scales. Castellani *et al.* [16] apply Gaussian filtering directly on the mesh geometry, and create a scale space describing the displacement of the mesh points in the normal direction.

Because many feature detectors operate locally on a function defined on the shape, they are usually not very susceptible to non-rigid deformations. Nevertheless, there exist several geometric quantities based on the intrinsic properties of the manifold and thus theoretically invariant to isometric deformations by construction. Feature detectors based on such quantities are called *intrinsic* and also *isometry-* or *bending-invariant*. Examples of intrinsic geometric quantities are the Gaussian curvature (which has been used in several settings of [56]), and heat kernels [51, 22]. Feature detection meth-

ods based on the heat kernel define a function on the shape, measuring the amount of heat remaining at a point x after large time t given a point source at x at time 0, and detect features as local maxima of this function.

Another type of transformations of interest in practical applications are changes in topology, manifested as the presence of holes, missing parts, or changes in connectivity. Feature detectors insensitive to such changes (typically, a simpler case of point-wise connectivity change) are referred to as *topology-invariant*.

Table 3.1 summarizes the properties of known feature detectors, some of which are detailed in what follows.

Descriptor	Representation	Invariance			
		Scale	Rigid	Bending	Topology
Dense	Any	Yes	Yes	Yes	Yes
Harris 3D [46]	Any	No	Yes	Approx	Approx
Mesh DOG [56]	Mesh	No	Yes	Approx ¹	Approx
Salient features [16]	Mesh	No	Yes	Approx	Approx
Heat kernel [51]	Any	No	Yes	Yes	Approx

Table 1: Comparison of 3D feature detectors.

3.2 Harris 3D

An efficient feature detection method, called Harris operator, first proposed in images [25] was extended to 3D shapes by Glomb [24] and Sipiran and Bustos [46]. This method is based on measuring variability of the shape in a local neighborhood of the point, by fitting a function to the neighborhood, and identifying feature points as points where the derivatives of this function are high [7]. Unlike images, 3D data might have arbitrary topology and sampling, which complicates the computation of derivatives.

¹Unless truly intrinsic quantities are used.

For each point x on the shape, a neighborhood of radius ρ (typically, a k -ring in mesh representation) is selected. The neighborhood points are brought into a canonical system of coordinates by first subtracting the centroid. Next, a plane is fit into the translated points by applying PCA and choosing the direction corresponding to the smallest eigenvalues as the direction of the normal. The points are rotated so that the normal is aligned with the z -axis. A quadratic function of the form $f(u, v) = a^T(u^2, uv, v^2, u, v, 1)$ is then fit to the set of transformed points, yielding a parametric representation of the local extrinsic surface properties.

A 2×2 symmetric matrix

$$E = \frac{1}{\sqrt{2\pi\sigma}} \int_{\mathbb{R}^2} e^{-\frac{u^2+v^2}{2\sigma^2}} \begin{pmatrix} f_u^2(u, v) & f_u(u, v)f_v(u, v) \\ f_u(u, v)f_v(u, v) & f_v^2(u, v) \end{pmatrix} dudv \quad (1)$$

is computed. The *3D Harris operator* is defined as the map assigning $H(x) = \det(E) - 0.04\text{tr}^2(E)$ to each point x on the shape. A fixed percentage of points with the highest values of $H(x)$ are selected as the feature points.

In [46], the neighborhood radius ρ (alternatively, the k -ring width k) and the Gaussian variance σ are performed adaptively for each point in order to make the method independent on sampling and triangulation.

3.3 Mesh DOG

The Mesh DOG descriptor introduced in Zaharescu *et al.* [56] assumes the shape in mesh representation and in addition to be given some function f defined on the mesh vertices. The function can be either photometric information (texture) or a geometric quantity such as curvature.

Given a scalar function f on the shape, its *convolution* with a radially-symmetric kernel $k(r)$ is defined as

$$(f * k)(x) = \int k(d(x, y))f(y)dy, \quad (2)$$

where $d(x, y)$ is the geodesic distance between points x and y . Zaharescu *et*

al. [56] propose the following r -ring approximation:

$$(f * k)(x) = \frac{\sum_{y \in \mathcal{N}_r(x)} k(\|x - y\|) f(y)}{\sum_{y \in \mathcal{N}_r(x)} k(\|x - y\|)}, \quad (3)$$

which assumes a uniformly sampled mesh.

By subsequently convolving a function f with a Gaussian kernel g_σ of width σ , a scale space $f_0 = f$, $f_k = f_{k-1} * g_\sigma$ is constructed. The *difference of Gaussians* (DOG) operator at scale k is defined as $DOG_k = f_k - f_{k-1}$.

Feature points are selected as the maxima of the DOG scale space across scales, followed by non-maximum suppression, using the one ring neighborhood in the current and the adjacent scales. A fixed percentage of points with the highest values of DOG are selected. To further eliminate unstable responses, only features exhibiting *corner-like* characteristics are retained. For this purpose, the *Hessian* operator at every point x is computed as

$$H = \begin{pmatrix} f_{uu}(x) & f_{uv}(x) \\ f_{uv}(x) & f_{vv}(x) \end{pmatrix}, \quad (4)$$

where f_{uu} , f_{uv} and f_{vv} are the second-order partial derivatives of f at x . Second order derivatives are estimated w.r.t. some local system of coordinates u, v (obtained, e.g., by fixing u to be the direction of the gradient, $u = \nabla_X f(x)$, and v perpendicular to it) by applying the directional derivative twice,

$$f_{uv}(x) = \langle \nabla_X \langle \nabla_X f(x), u \rangle, v \rangle. \quad (5)$$

The condition number $\lambda_{\max}/\lambda_{\min}$ of H (typically, around 10) is independent of the selection of the local system of coordinates and is used to threshold the features.

3.4 Salient features

In Mesh DOG, the scale space is built by filtering a scalar function on the mesh while keeping the mesh geometry intact. Castellani *et al.* [16] proposed to create a scale space by filtering the shape itself.

Let $\mathbf{x} \in \mathbb{R}^3$ denote the extrinsic coordinates of a point x on a surface. Applying a Gaussian kernel g_σ to \mathbf{x} several times creates the vector-valued DOG scale space, $DOG_k = \mathbf{x}_k - \mathbf{x}_{k-1}$, where $\mathbf{x}_0 = \mathbf{x}$ and $\mathbf{x}_k = (\mathbf{x}_{k-1} * g_\sigma)$. By projecting $DOG_k(x)$ onto the normal $n(x)$ at the point x , a scalar-valued scale space (referred to as the *scale map* by the authors) is created. From this stage on, an approach essentially identical to Mesh DOG is undertaken. The authors do not use filtering by Hessian operator response, and propose to use a robust method inspired by [26] to detect the feature points.

3.5 Heat kernel features

Recently, there has been increased interest in the use of *diffusion geometry* for shape recognition [44, 38, 35, 32, 12, 41]. This type of geometry arises from the *heat equation*,

$$\left(\Delta_X + \frac{\partial}{\partial t} \right) u = 0, \quad (6)$$

which governs the conduction of heat u on the surface X (here, Δ_X denotes the positive semi-definite *Laplace-Beltrami operator*, a generalization of the Laplacian to non-Euclidean domains). The fundamental solution $K_t(x, y)$ of the heat equation, also called the *heat kernel*, is the solution of (6) at time t with a point heat source at x used as the initial condition. Probabilistically, the heat kernel can also be interpreted as the transition density function of a Brownian motion (continuous analog of a random walk). By virtue of the spectral decomposition theorem, the heat kernel can be expressed as

$$K_t(x, y) = \sum_i e^{-\lambda_i t} \phi_i(x) \phi_i(y), \quad (7)$$

where λ_i are the eigenvalues of the Laplace-Beltrami operator and ϕ_i are the corresponding eigenfunctions. This relation makes heat kernels especially attractive as there exists efficient and stable methods to discretize the Laplace-Beltrami operator and its eigendecomposition.

The diagonal of the heat kernel at different scales, $K_t(x, x)$, referred to as the *heat kernel signature* (HKS), can be interpreted as a multi-scale notion

of the Gaussian curvature. Local maxima of the HKS for a large time parameter correspond to tips of protrusions that can be used as stable features as recently proposed by Sun *et al.* [51] and Gebal *et al.* [22].

In the simplest setting, feature points are found as two-ring local maxima of $K_t(x, x)$ at a sufficiently large scale t [51]. In a more sophisticated setting, the persistence diagram of $K_t(x, x)$ is computed, and features with insufficiently large distance between birth and death times are filtered out [48, 19, 7].

3.6 Topological Features

A different variety of feature-based techniques have been inspired by topological, rather than geometrical, shape analysis (see e.g. [37] for a survey of some methods for the analysis of biomolecular data). The most common tool used in applying topological methods to feature-based shape analysis is the notion of *topological persistence* introduced and formalized by Edelsbrunner *et al.* [21]. In its most basic form, topological persistence allows to define a pairing between critical values of a function defined on a topological domain (such as a simplicial complex) in a canonical way. This pairing defines a persistence value associated with each critical point, which provides a principled way of distinguishing prominent local maxima and minima from noise. Thus, these techniques fit naturally into the feature-based shape analysis framework, where both feature detection and description is often obtained via analysis of critical values of some function. Several techniques have been recently proposed for finding stable feature points by applying topological persistence to different functions defined on the shape, including the Heat Kernel Signature [48, 19] and the eigenfunctions of the Laplace-Beltrami operator [43].

An excellent application of topological persistence to shape analysis and shape matching was demonstrated by Agarwal *et al.* [1], who used it to define a feature detector and descriptor, by defining a function on a surface, which approximately captures the concavity and convexity at each point in

a parameter-free way. For every point x on the surface, the authors use topological persistence to find a canonical pair y which shares the normal direction with x . Then the *elevation function* at x is simply the difference of the height values of x and y in this normal direction. Elevation function is invariant to rigid deformations and allows to analyze both concavities and convexities in a unified fashion. Prominent minima and maxima of the elevation function can also be used as natural stable features of a shape. Applying methods from computational topology to feature-based shape analysis is an active and potentially fruitful area of research, and we refer an interested reader to a recent book [20].

3.7 Benchmarks

An ideal feature detector should be repeatable under the desired class of shape transformations and also “rich” or informative. While the latter is largely application and data-dependent, the repeatability of the detector can be evaluated quantitatively on a set of representative shape transformations. SHREC’10 robust feature detection and description benchmark [7] evaluates the detector repeatability by running the detector on a set of reference shapes. The detected features are used as reference locations. Then, detection is performed on the same shapes undergoing simulated transformations of different types (non-rigid bending, different types of noise, holes, etc.), for which groundtruth correspondence with the reference shapes is known. Repeatability is evaluated by counting the fraction of features that are consistently detected in the proximity of the reference locations. Different varieties of the heat kernel methods achieved the best results on this benchmark.

4 Feature descriptors

Given a set of feature points (or, in the case of a dense descriptor, all the points on the shape), a local descriptor is then computed. The way in which the descriptor is constructed depends very much on the representation in

which the shape is given and the kind of information available, and the application in mind, or more specifically, the desired invariance properties.

Descriptors can be categorized as *geometric* or *photometric*, depending whether they rely only on the 3D geometry of the shape, or also make use of the texture. Some photometric descriptors can be adapted to work with geometric information, where some geometric property (e.g. curvature) is used in place of the texture [56]. A wide variety of geometric quantities such as local patches [36], local moments [18] and volume [23], spherical harmonics [45], and contour and edge structures [39, 30] trying to emulate comparable features in images, can be used for geometric descriptors.

4.1 A taxonomy

Multiscale descriptors (e.g. [51, 15]) look at the shape at multiple levels of resolution, thus capturing different properties manifested at these scales. Descriptors which are not altered by global scaling of the shape are called *scale-invariant*.

Because typically a descriptor operates locally around the feature point, feature descriptors are usually not very susceptible to non-rigid deformations of the shape. Nevertheless, there exist several geometric descriptors which are based on intrinsic properties of the manifold and thus theoretically invariant to isometric deformations by construction. Examples of intrinsic descriptors include histograms of local geodesic distances [40, 14], conformal factors [6], some settings of [56], and heat kernels [51, 15]. Such descriptors are called *intrinsic* and also *isometry-* or *bending-invariant*.

Another type of transformations of interest in practical applications are changes in topology, manifested as the presence of holes, missing parts, or changes in connectivity. Descriptors insensitive to such changes (typically, a simpler case of point-wise connectivity change) are referred to as *topology-invariant*.

Finally, some authors [23] make a distinction between *high-dimensional* (or *rich*) and *low-dimensional* descriptors. The former refers to descriptors

providing a fairly detailed description of the shape properties around the point such as [5, 27], while the latter compute only a few values per point and typically are curvature-like quantities such as *shape index* [28] and *curvedness* [29]. We find this division somewhat misleading, as there is no direct relation between the descriptor “richness” and dimensionality (recent works in computer vision on descriptor hashing and dimensionality reduction [50] demonstrate that rich descriptors such as SIFT can be compactly represented in much lower dimensions without losing much information). The question whether the “richness” of a descriptor is sufficient depends in general on the application and the data.

Table 4.1 summarizes the properties of known descriptors, some of which are detailed in what follows. We devote particular attention to different varieties of the recently introduced heat kernel signatures, which we consider one of the most versatile descriptors currently available, as well as a promising and interesting field for future research.

Descriptor	Representation	Invariance			
		Scale	Rigid	Bending	Topology
Curvature	Any	No	Yes	Yes	No
Integral volume [23]	Volume, Mesh ⁰	No	Yes	No	No
Local histograms [40]	Any	No ¹	Yes	Yes	No ¹
HKS [51]	Any	No	Yes	Yes	Approx ²
SIHKS [15]	Any	Yes	Yes	Yes	Approx ²
VHKS [42]	Volume, Mesh ⁰	No	Yes	Yes	Approx ²
Spin image [27]	Any	a	a	a	a
Shape context [5]	Any	No	Yes	No	Yes
MeshHOG [56]	Mesh (+Texture)	Yes ³	Yes	Approx ⁴	Approx ³
Conformal factor [6]	Mesh	No	Yes	Yes	No ⁵

Table 2: Comparison of 3D feature descriptors.

4.2 Spin images

Spin image descriptor [27, 2, 3] represents the neighborhood of a point on a shape by fitting an oriented coordinate system at the point. The local system of cylindrical coordinates at point x is defined using the normal and tangent plane: the radial coordinate α defined as the perpendicular distance to the line through the surface normal $n(x)$, and the elevation coordinate β , defined as the signed perpendicular distance to the tangent plane. The cylindrical angular coordinate is omitted because it cannot be defined robustly and unambiguously on planar surfaces.

A spin image is a histogram of points in the *support region* represented in α, β coordinates. The support region is defined by limiting the range of the values of α and β (thus looking at points y within some distance from x) and requiring that $\cos^{-1}\langle n(x), n(y) \rangle < \epsilon$ (limiting self occlusion artifacts). The histogram can be represented as a 2D image, hence the name of the descriptor. Spin image is applicable to any shape representation in which the point coordinates are explicitly given and normals and tangent planes can be computed, e.g., meshes or point clouds. Because of dependence on the embedding coordinates, such a descriptor is not deformation-invariant.

4.3 Shape context

The concept of *shape context descriptor* was first introduced in [5] for image analysis, though it is directly applicable to 3D shapes. The shape context describes the structure of the shape as relations between a point to the rest of the point. Given the coordinates of a point x on the shape, the shape con-

⁰Involving mesh rasterization.

¹Assuming geodesic distances. Different invariance properties can be achieved using diffusion or commute-time distances.

²Point-wise connectivity changes have only a local effect and do not propagate to distant descriptors.

³If photometric texture is used; in general, depending on the texture choice.

⁴Triangulation-dependent.

⁴Defined for shapes with fixed topology (e.g. watertight).

text descriptor is constructed as a histogram of the direction vectors from x to the rest of the point, $y - x$. Typically, a log-polar histogram is used. The descriptor is applicable to any shape representation in which the point coordinates are explicitly given, such as mesh, point cloud, or volume. Because of dependence on the embedding coordinates, such a descriptor is not deformation-invariant.

4.4 Integral volume descriptor

The *integral volume descriptor* used in [23] is an extension to 3D shapes in of the concept of integral invariants introduced for image description in [33]. Given a solid object Ω with a boundary $X = \partial\Omega$, the descriptor measures volume contained in a ball of fixed radius r ,

$$V_r(x) = \int_{B_r(x) \cap \Omega} dx. \quad (8)$$

If $B_r(x) \cap \Omega$ is simply connected, the volume descriptor can be related to the mean curvature $H(x)$ as $V_r(x) = \frac{2\pi}{3}r^3 - \frac{\pi}{4}Hr^4 + \mathcal{O}(r^5)$ [23]. Since the mean curvature is not intrinsic, the descriptor is sensitive to deformations of the shape. Varying the value of r , a multi-scale descriptor can be computed. Numerically, the descriptor is efficiently computed in a voxel representation of the shape by means of convolution with the ball mask.

4.5 Mesh HOG

MeshHOG [56] is a shape descriptor emulating SIFT-like image descriptors [31], referred to as *histograms of gradients* or *HOG*. The descriptor assumes the shape in mesh representation and in addition to be given some function f defined on the mesh vertices. The function can be either photometric information (texture) or a geometric quantity such as curvature. The descriptor at point x is computed by creating a local histogram of gradients of f in an r -ring neighborhood of x . The gradient ∇f is defined extrinsically as a

vector in \mathbb{R}^3 but projected onto the tangent plane at x which makes it intrinsic. The descriptor support is divided into four polar slices (corresponding to 16 quadrants in SIFT). For each of the slices, a histogram of 8 gradient orientations is computed. The result is a 32-dimensional descriptor vector concatenating the histogram bins.

MeshHOG descriptor works with mesh representations and can work with photometric or geometric data or both. It is intrinsic in theory, though the specific implementation in [56] depends on triangulation.

4.6 Heat kernel signatures

The *heat kernel signature* (HKS) was proposed in [51] as an intrinsic descriptor based on the properties of heat diffusion and defined as the diagonal of the heat kernel. Given some fixed time values t_1, \dots, t_n , for each point x on the shape, the HKS is an n -dimensional descriptor vector

$$p(x) = (K_{t_1}(x, x), \dots, K_{t_n}(x, x)). \quad (9)$$

Intuitively, the diagonal values of the heat kernel indicate how much heat remains at a point after certain time (or alternatively, the probability of a random walk to remain at a point if resorting to the probabilistic interpretation of diffusion processes), and is thus related to the “stability” of a point under diffusion process.

The HKS descriptor is intrinsic and thus isometry-invariant, captures local geometric information at multiple scales, insensitive to topological noise, and informative (if the Laplace-Beltrami operator of a shape is non-degenerate, then any continuous map that preserves the HKS at every point must be an isometry). Since the HKS can be expressed in the Laplace-Beltrami eigenbasis as

$$K_t(x, x) = \sum_{i \geq 0} e^{-t\lambda_i} \phi_i^2(x), \quad (10)$$

it is easily computed across different shape representations for which there is a way to compute the Laplace-Beltrami eigenfunctions and eigenvalues.

4.7 Scale-invariant heat kernel signatures

A disadvantage of the HKS is its dependence on the global scale of the shape. If X is globally scaled by β , the corresponding HKS is $\beta^{-2}K_{\beta^{-2}t}(x, x)$. In some cases, it is possible to remove this dependence by *global* normalization of the shape. A *scale-invariant HKS* (SI-HKS) based on *local* normalization was proposed in [15]. By using a logarithmic scale-space $t = \alpha^\tau$, the scaling of X by β results in HKS amplitude scaling and shift by $2 \log_\alpha \beta$. This effect is undone by the following sequence of transformations,

$$\begin{aligned} p_{dif}(x) &= (\log K_{\alpha^{\tau_2}}(x, x) - \log K_{\alpha^{\tau_1}}(x, x), \dots, \log K_{\alpha^{\tau_m}}(x, x) - \log K_{\alpha^{\tau_{m-1}}}(x, x)), \\ \hat{p}(x) &= |(\mathcal{F}p_{dif}(x))(\omega_1, \dots, \omega_n)|, \end{aligned} \quad (11)$$

where \mathcal{F} is the discrete Fourier transform, and $\omega_1, \dots, \omega_n$ denotes a set of frequencies at which the transformed vector is sampled. Taking differences of logarithms removes the scaling constant, and the Fourier transform converts the scale-space shift into a complex phase, which is removed by taking the absolute value.

4.8 Volumetric heat kernel signatures

The idea of heat kernel descriptor can be applied to volumetric shape representations [42]. In this case, given a solid object Ω , the heat diffusion inside the volume is given by the heat equation with Neumann boundary conditions on the boundary $\partial\Omega$,

$$\begin{aligned} \left(\Delta + \frac{\partial}{\partial t} \right) U(x, t) &= 0 & x \in \text{int}(\Omega), \\ \langle \nabla U(x, t), n(x) \rangle &= 0 & x \in \partial\Omega \end{aligned} \quad (12)$$

where n is the normal to the boundary surface $\partial\Omega$, Δ is the positive-semidefinite Laplacian operator in \mathbb{R}^3 , and $U : \Omega \times [0, \infty) \rightarrow \mathbb{R}$ is the volumetric heat distribution in Ω . The *volumetric heat kernel signature* (VHKS) is defined as

the diagonal of the heat kernel of (12) at a set of time values t , expressible in the eigenbasis of the Laplacian as

$$K_t(x, x) = \sum_{l=0}^{\infty} e^{-\lambda_l t} \Phi_l(x)^2, \quad (13)$$

where λ_l , Φ_l are the eigenvalues and eigenfunctions of the Laplacian operator with the above boundary conditions,

$$\begin{aligned} \Delta \Phi_l(x) &= \lambda_l \Phi_l(x); \\ \langle \nabla \Phi_l(x), n(x) \rangle &= 0 \quad x \in \partial\Omega. \end{aligned} \quad (14)$$

The descriptor can be computed on any volumetric representation of the shape allowing for efficient computation of the Laplacian eigenvalues and eigenfunctions. For meshes and other surface representations, it is necessary to perform rasterization to convert them into voxel representation [42].

4.9 Benchmarks

An ideal feature descriptor should be invariant under the desired class of shape transformations and also “rich” or informative. While the latter is largely application and data-dependent, the invariance of the descriptor can be evaluated quantitatively on a set of representative shape transformations. SHREC’10 robust feature detection and description benchmark [7] evaluates the descriptor invariance as the variability (measured as average normalized L_2 distance) of the descriptor under simulated transformations of different types (non-rigid bending, different types of noise, holes, etc.). Different varieties of the heat kernel signature (HKS) descriptor achieved the best results on this benchmark.

5 Applications

5.1 Shape retrieval

The availability of large public-domain databases of 3D models such as Google 3D Warehouse has created the demand for shape search and retrieval algorithms capable of finding similar shapes in the same way a search engine responds to text queries. *Content-based shape retrieval* using the shape itself as a query and based on the comparison of the geometric and topological properties of shapes is complicated by the fact that many 3D objects manifest rich variability, and shape retrieval must often be *invariant* under different classes of transformations. A particularly challenging setting is the case of *non-rigid* or *deformable* shapes, which includes a wide range of shape transformations such as bending and articulated motion. Shape retrieval is a particular instance of *shape similarity*, a fundamental problem of shape analysis.

One of the notable advantages of feature-based approaches is the possibility of representing a shape as a collection of primitive elements (“geometric words”), and using the well-developed methods from text search such as the *bag of features* (BOF) (or *bag of words*) paradigm [47, 17]. Such approaches are widely used in image retrieval, and have been introduced more recently to shape analysis [13, 53]. The bag of features representation is usually compact, easy to store and compare, which makes such approaches suitable for large-scale shape retrieval.

The construction of a bag of features is usually performed in a few steps, depicted in Figure 1. First, the shape is represented as a collection of local feature descriptors (either dense or computed at a set of stable points following an optional stage of feature detection). Second, the descriptors are represented by *geometric words* from a *geometric vocabulary* using vector quantization. The geometric vocabulary is a set of representative descriptors, precomputed in advance. This way, each descriptor is replaced by the index of the closest geometric word in the vocabulary. Computing the histogram

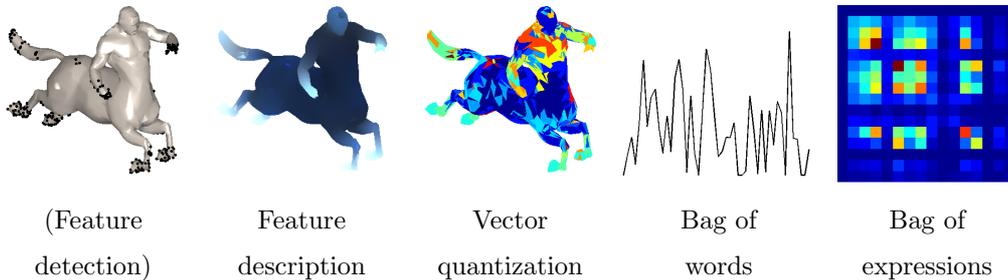


Figure 1: Feature-based shape analysis algorithm.

of the frequency of occurrence of geometric words gives the bag of features. Alternatively, a two-dimensional histogram of co-occurrences of pairs of geometric words (*geometric expressions*) can be used [13]. Shape similarity is computed as a distance between the corresponding bags of features.

5.2 Correspondence

Another fundamental problem in shape analysis is that of *correspondence* consisting of finding relations between similar features on two or more shapes. Defining optimal correspondence based on some structure preservation criterion, one can obtain a criterion of shape similarity as the amount of structure distortion, making shape correspondence intimately related to shape similarity problems. Finding correspondence between two shapes that would be invariant to a wide variety of transformations is usually referred to as *invariant shape correspondence*.

Correspondence problems are often encountered in shape synthesis applications such as morphing. In order to morph one shape into the other, one needs to know which point on the first shape will be transformed into a point on the second shape, in other word, establishing a correspondence between the shapes. A related problem is *registration*, where the deformation bringing one shape into the other is explicitly sought for.

Let us be given two shapes X and Y with the feature points $\{x_i\}_{i=1}^M$ and $\{y_j\}_{j=1}^N$, and corresponding descriptors $\{p_i\}$ and $\{q_j\}$, respectively. Feature-

based correspondence problem can be formulated as finding a (partial) map $C : \{1, \dots, M\} \rightarrow \{1, \dots, N\}$ that maximizes the similarity between corresponding descriptors while keeping as similar as possible some global structure, usually expressed in terms of the geodesic or diffusion metrics, d_X and d_Y , on X and Y . Correspondence problems can be therefore written as a generic minimization problem, e.g.,

$$\min_C \sum_i \|p_i - q_{C(i)}\|^2 + \eta \sum_{i,k} (d_X(x_i, x_j) - d_Y(y_{C(i)}, y_{C(j)}))^2$$

where η is a Lagrange multiplier.

The minimum-distortion correspondence can be found by an extension of the GMDS algorithm [10, 52] or graph labeling [49, 54, 55].

5.3 Benchmarks

SHREC10 robust large-scale retrieval benchmark [8] simulates a retrieval scenario, in which the queries include multiple modifications and transformations of the same shape which is placed into a large corpus of “negatives”. The quality of the retrieval is quantified in terms of mean average precision. Different varieties of the heat kernel feature methods achieved the best results on this benchmark.

SHREC10 robust correspondence benchmark [9] simulates a one-to-one shape matching scenario, in which one of the shapes undergoes multiple modifications and transformations. The quality of the correspondence is evaluated as the distance on the shape between the found matches and the known groundtruth correspondence.

6 Conclusions

In this chapter, we overviewed feature-based methods in 3D shape analysis and their applications such as content-based shape retrieval and invariant shape correspondence.

7 Further reading

For a broad overview of geometric foundations and algorithms in shape analysis, we refer the reader to [11].

Details of SHREC'10 benchmarks mentioned in this chapter appear in [7, 8, 9].

Meta-algorithms for dimensionality reduction and hashing of descriptors are discussed in [50].

References

- [1] P. K. Agarwal, H. Edelsbrunner, J. Harer, and Y. Wang, *Extreme elevation on a 2-manifold*, SCG '04: Proceedings of the twentieth annual symposium on Computational geometry, 2004, pp. 357–365.
- [2] M. Andreetto, N. Brusco, and G. M Cortelazzo, *Automatic 3D modeling of textured cultural heritage objects*, Trans. Image Processing **13** (2004), no. 3, 335–369.
- [3] J. Assfalg, M. Bertini, and P. Pala A. Del Bimbo, *Content-based retrieval of 3d objects using spin image signatures*, Trans. Multimedia **9** (2007), no. 3, 589–599.
- [4] H. Bay, T. Tuytelaars, and L. Van Gool, *SURF: Speeded up robust features*, Proc. ECCV, 2006, pp. 404–417.
- [5] S. Belongie, J. Malik, and J. Puzicha, *Shape matching and object recognition using shape contexts*, Trans. PAMI (2002), 509–522.
- [6] M. Ben-Chen, O. Weber, and C. Gotsman, *Characterizing shape using conformal factors*, Proc. 3DOR, 2008.
- [7] A. M. Bronstein, M. M. Bronstein, B. Bustos, U. Castellani, M. Crisani, B. Falcidieno, L. J. Guibas, I. Isipiran, I. Kokkinos, V. Murino, M. Ovs-

- janikov, G. Patané, M. Spagnuolo, and J. Sun, *Shrec 2010: robust feature detection and description benchmark*, Proc. 3DOR, 2010.
- [8] A. M. Bronstein, M. M. Bronstein, U. Castellani, B. Falcidieno, A. Fusiello, A. Godil, L. J. Guibas, I. Kokkinos, Z. Lian, M. Ovsjanikov, G. Patané, M. Spagnuolo, and R. Toldo, *Shrec 2010: robust large-scale shape retrieval benchmark*, Proc. 3DOR, 2010.
- [9] A. M. Bronstein, M. M. Bronstein, U. Castellani, A. Dubrovina L. J. Guibas, R. P. Horaud, R. Kimmel, D. Knossow, E. von Lavante, D. Mateus, M. Ovsjanikov, and A. Sharma, *Shrec 2010: robust correspondence benchmark*, Proc. 3DOR, 2010.
- [10] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, *Numerical geometry of non-rigid shapes*, Springer-Verlag New York Inc, 2008.
- [11] ———, *Numerical geometry of non-rigid shapes*, Springer-Verlag New York, 2008, First systematic treatment of non-rigid shapes.
- [12] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro, *A Gromov-Hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching*, IJCV (2010).
- [13] A. M. Bronstein, M. M. Bronstein, M. Ovsjanikov, and L. J. Guibas, *Shape google: a computer vision approach to invariant shape retrieval*, Proc. NORDIA, 2009.
- [14] A.M. Bronstein, M.M. Bronstein, A.M. Bruckstein, and R. Kimmel, *Partial similarity of objects, or how to compare a centaur to a horse*, International Journal of Computer Vision **84** (2009), no. 2, 163–183.
- [15] M. M. Bronstein and I. Kokkinos, *Scale-invariant heat kernel signatures for non-rigid shape recognition*, Proc. CVPR, 2010.

- [16] U. Castellani, M. Cristani, S. Fantoni, and V. Murino, *Sparse points matching by combining 3D mesh saliency with statistical descriptors*, Computer Graphics Forum **27** (2008), 643–652.
- [17] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, *Total recall: Automatic query expansion with a generative feature model for object retrieval*, Proc. ICCV, 2007.
- [18] 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** (2004), no. 5, 516–524.
- [19] T. K. Dey, K. Li, C. Luo, P. Ranjan, I. Safa, and Y. Wang, *Persistent Heat Signature for Pose-oblivious Matching of Incomplete Models*, Proc. SGP, 2010, pp. 1545–1554.
- [20] H. Edelsbrunner and J. Harer, *Computational Topology. An Introduction*, Amer. Math. Soc., Providence, Rhode Island, 2010.
- [21] H. Edelsbrunner, D. Letscher, and A. Zomorodian, *Topological persistence and simplification*, Discrete and Computational Geometry **28** (2002), no. 4, 511–533.
- [22] K. Gebal, J. A. Bærentzen, H. Aanæs, and R. Larsen, *Shape analysis using the auto diffusion function*, Computer Graphics Forum **28** (2009), no. 5, 1405–1413.
- [23] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, *Robust global registration*, Proc. SGP, 2005.
- [24] P. Glomb, *Detection of interest points on 3D data: Extending the harris operator*, Computer Recognition Systems 3, Advances in Soft Computing, vol. 57, Springer Berlin / Heidelberg, May 2009, pp. 103–111.
- [25] C. Harris and M. Stephens, *A combined corner and edge detection*, Proc. Fourth Alvey Vision Conference, 1988, pp. 147–151.

- [26] L. Itti, C. Koch, and E. Niebur, *A model of saliency-based visual attention for rapid scene analysis*, Trans. PAMI **20** (1998), no. 11.
- [27] A. E. Johnson and M. Hebert, *Using spin images for efficient object recognition in cluttered 3D scenes*, Trans. PAMI **21** (1999), no. 5, 433–449.
- [28] P. Koehl, *Protein structure similarities*, Current Opinion in Structural Biology **11** (2001), no. 3, 348–353.
- [29] J.J. Koenderink and A.J. van Doorn, *Surface shape and curvature scales*, Image and vision computing **10** (1992), no. 8, 557–564.
- [30] M. Kolomenkin, I. Shimshoni, and A. Tal, *On edge detection on surfaces*, Proc. CVPR, 2009.
- [31] D. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV (2004).
- [32] M. Mahmoudi and G. Sapiro, *Three-dimensional point cloud recognition via distributions of geometric distances*, Graphical Models **71** (2009), no. 1, 22–31.
- [33] S. Manay, B.W. Hong, A.J. Yezzi, and S. Soatto, *Integral invariant signatures*, Computer Vision-ECCV 2004 (2004), 87–99.
- [34] J. Matas, O. Chum, M. Urban, and T. Pajdla, *Robust wide-baseline stereo from maximally stable extremal regions*, Image and Vision Computing **22** (2004), no. 10, 761–767.
- [35] D. Mateus, R. P. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer, *Articulated shape matching using laplacian eigenfunctions and unsupervised point registration*, Proc. CVPR (2008).
- [36] N. J. Mitra, L. J. Guibas, J. Giesen, and M. Pauly, *Probabilistic fingerprints for shapes*, Proc. SGP, 2006.

- [37] V. Natarajan, P. Koehl, Y. Wang, and B. Hamann, *Visual analysis of biomolecular surfaces*, Visualization in Medicine and Life Sciences, Mathematics and Visualization, 2008, pp. 237–255.
- [38] M. Ovsjanikov, J. Sun, and L. J. Guibas, *Global intrinsic symmetries of shapes*, Computer Graphics Forum, vol. 27, 2008, pp. 1341–1348.
- [39] M. Pauly, R. Keiser, and M. Gross, *Multi-scale feature extraction on point-sampled surfaces*, Computer Graphics Forum, vol. 22, 2003, pp. 281–289.
- [40] D. Raviv, A. M. Bronstein, M. M. Bronstein, and R. Kimmel, *Symmetries of non-rigid shapes*, Proc. NRTL, 2007.
- [41] D. Raviv, A. M. Bronstein, M. M. Bronstein, R. Kimmel, and G. Sapiro, *Diffusion symmetries of non-rigid shapes*, Proc. 3DPVT, 2010.
- [42] D. Raviv, M. M. Bronstein, A. M. Bronstein, and R. Kimmel, *Volumetric heat kernel signatures*, Proc. ACM Multimedia Workshop on 3D Object Retrieval, 2010.
- [43] M. Reuter, *Hierarchical shape segmentation and registration via topological features of laplace-beltrami eigenfunctions*, International Journal of Computer Vision **89** (2010), no. 2, 287–308.
- [44] R. M. Rustamov, *Laplace-Beltrami eigenfunctions for deformation invariant shape representation*, Proc. SGP, 2007, pp. 225–233.
- [45] P. Shilane and T. Funkhouser, *Selecting distinctive 3D shape descriptors for similarity retrieval*, Proc. Shape Modelling and Applications, 2006.
- [46] I. Sipiran and B. Bustos, *Robust 3D Harris operator*, submitted, 2010.
- [47] J. Sivic and A. Zisserman, *Video google: A text retrieval approach to object matching in videos*, Proc. CVPR, 2003.

- [48] P. Skraba, M. Ovsjanikov, F. Chazal, and L. Guibas, *Persistence-based segmentation of deformable shapes*, CVPR Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment, June 2010.
- [49] J. Starck and A. Hilton, *Correspondence labelling for widetimeframe free-form surface matching*, Proc. ICCV, 2007.
- [50] C. Strecha, A. M. Bronstein, M. M. Bronstein, and P. Fua, *Ldahash: Improved matching with smaller descriptors*, Technical report, EPFL, 2010.
- [51] J. Sun, M. Ovsjanikov, and L. J. Guibas, *A concise and provably informative multi-scale signature based on heat diffusion*, Proc. SGP, 2009.
- [52] N. Thorstensen and R. Keriven, *Non-rigid shape matching using geometry and photometry*, Proc. CVPR, 2009.
- [53] R. Toldo, U. Castellani, and A. Fusiello, *Visual vocabulary signature for 3D object retrieval and partial matching*, Proc. 3DOR, 2009.
- [54] L. Torresani, V. Kolmogorov, and C. Rother, *Feature correspondence via graph matching: Models and global optimization*, Proc. ECCV, 2008, pp. 596–609.
- [55] C. Wang, M. M. Bronstein, and N. Paragios, *Discrete minimum distortion correspondence problems for non-rigid shape matching*, Research Report 7333, INRIA, 2010.
- [56] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, *Surface feature detection and description with applications to mesh matching*, Proc. CVPR, 2009.