Scale-Dependent 3D Geometric Features

John Novatnack Ko Nishino
Department of Computer Science
Drexel University
{jmn27,kon}@cs.drexel.edu

Abstract

Three-dimensional geometric data play fundamental roles in many computer vision applications. However, their scale-dependent nature, i.e. the relative variation in the spatial extents of local geometric structures, is often overlooked. In this paper we present a comprehensive framework for exploiting this 3D geometric scale variability. Specifically, we focus on detecting scale-dependent geometric features on triangular mesh models of arbitrary topology. The key idea of our approach is to analyze the geometric scale variability of a given 3D model in the scale-space of a dense and regular 2D representation of its surface geometry encoded by the surface normals. We derive novel corner and edge detectors, as well as an automatic scale selection method, that acts upon this representation to detect salient geometric features and determine their intrinsic scales. We evaluate the effectiveness and robustness of our method on a number of models of different topology. The results show that the resulting scale-dependent geometric feature set provides a reliable basis for constructing a rich but concise representation of the geometric structure at hand.

1. Introduction

Three-dimensional geometric data play increasingly vital roles in various computer vision applications, ranging from navigation to inverse rendering. Despite their ubiquitous use, little attention has been given to the fact that real-world 3D geometric data can contain significant scale variation in their local geometric structures. For instance, in a 3D human face model, both the tip of the nose and dimples are discriminative geometric features suitable for representing the underlying surface. The spatial extents of such geometric features, however, significantly differ from one another – they lie at entirely different scales. While this geometric scale variability can be deemed to be another cause of error in subsequent processing, it can in turn be exploited as an additional source of information to enrich the representation of the actual object/scene geometry. In this paper we focus on extracting scale-dependent 3D geometric features, a unified set of geometric features detected at their own intrinsic scales, in geometric data given as mesh models.

Several methods have been proposed in the past that account for the geometric scale variability in 3D feature detection. Most of these methods are loosely based on the 2D scale-space theory – the analysis of scale variability in intensity images. In 2D scale-space theory, the space of images across different scales, the scale-space, is constructed by successively convolving the image with Gaussian kernels of increasing standard deviation [11, 15, 26, 27]. Rich visual features, including corners, edges, and blobs, can then be detected in this scale-space and their intrinsic scales can be identified [16].

Previous methods essentially apply the 2D scale-space theory to 3D geometric data by replacing pixel intensities with the 3D vertex coordinates of the mesh model. However, directly "smoothing" the 3D points, for instance with Gaussian kernels [18] or mean curvature flow [22], modifies the extrinsic geometry of the original model. This can lead to alterations of the global topology of the geometric data, in particular through fragmentation of the original model [25], which leads to an erroneous scale-space representation. Most past methods also use the Euclidean distance between 3D points as the distance metric in the operator for constructing the scale-space representation [8, 12, 14, 21]. This, however, can lead to the creation of erroneous features in the scale space, due to local topological changes within the support region of the operator [1]. Finally, past methods are tailored to detect one type of a feature, either corners [8] or edges [20], or use empirical measures for detecting them [19].

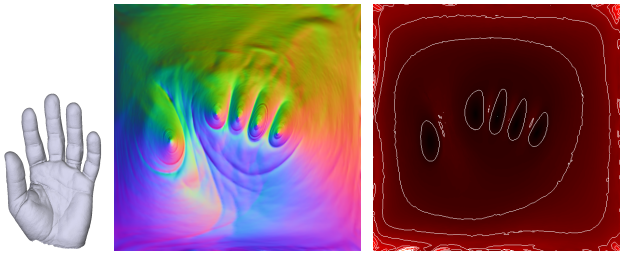
In this paper, we present a comprehensive method for extracting scale-dependent 3D geometric features. We aim to construct a sound scale-space type representation from which a variety of scale-dependent features can be extracted. The key insight behind our method is that we

[1] For instance, if two surfaces from different parts of the model lie close to each other, the support region of the scale-space operator will mistakenly include both surfaces.

should construct a scale-space representation of the surface geometry of the 3D model at hand, since we are interested in the scale-dependent geometric features lying on the surface. For this reason, we consider the normal field on the 2D surface of the 3D model as the base representation of the 3D geometry. While higher-order geometric entities such as the mean curvature [13] can be used instead, surface normals are a better choice since it is the first-order partial derivatives of the raw geometric data which is less affected by noise compared to higher-order partial derivatives.

We first parameterize the surface of the mesh model on a 2D plane. We then interpolate over the surface normals at each 2D-embedded vertex of the mesh model to obtain a dense and regular 2D representation (vector field) of the original surface, which we refer to as the *normal map*. This representation is independent of the mesh resolution and thus the results are robust to different samplings of the surface. Since the parameterization introduces distortion to the relative distances between the surface points, we compute a *distortion map* which encodes the relative change in the model edge lengths. We compute the discrete scale-space of the normal map (*geometric scale-space*) by convolving the vector field with Gaussian kernels of increasing standard deviation. The Gaussian kernel is modified to account for the distortion induced by the unwrapping. This ensures the use of the geodesic distances as the distance metric. As a result, constructing and analyzing the geometric scale-space is equivalent to analyzing the scale-space of the normal field on the surface of the 3D model, however, algorithmically in a much simpler way since we compute it in a regular and dense 2D domain.

A rich set of scale-dependent features can be extracted from the resulting scale-space representation. In particular, we derive detectors to extract geometric corners and edges at different scales. In order to establish these detectors we carefully derive the first- and second-order partial derivatives of the normal map. Finally, we derive an automatic scale selection method analogous to that of 2D scale-space theory to identify the natural scale of each feature and to unify all features into a single set. The result is a set of scale-dependent 3D geometric features that provide a rich and unique basis for representing the 3D geometry of the original data. For instance, these scale-dependent geometric features may directly be used for matching and aligning different 3D models robustly against local occlusions and distortions. They can also be used for deriving various local and global geometric representations, such as scale-invariant local geometric descriptors that can be useful in 3D object recognition or hierarchical feature-based global representations that can enable efficient 3D registration. The effectiveness of the proposed method is evaluated on several 3D models of different topology and its robustness to noise and mesh sampling density variation is demonstrated.



(a) (b) (c)

Figure 1. 2D normal and distortion map of a 3D model. (a) shows the original model. (b) illustrates the dense 2D normal map. Observe that geometric features such as the creases on the palm are clearly visible. (c) shows the distortion maps corresponding to the normal maps. Darker regions have been shrunk relative to the brighter regions. Iso-contour lines illustrate the various levels of distortion induced by the embedding.

2. Scale-Space of a 3D Model

Geometric features that represent a given 3D mesh model reside on the model's surface. For this reason, we must construct a scale-space type of a representation that faithfully encodes the scale variability of its surface geometry. We represent the given geometry with its surface normals and compute a dense and regular 2D representation of it by parameterizing the surface on a 2D plane. We then build a scale-space of this surface normal field by deriving and applying a scale-space operator that correctly accounts for the geodesic distances on the surface.

2.1. 2D Representation of the Surface Geometry

We construct a 2D representation of the 3D geometry given as a 3D mesh model by first unwrapping the surface of the model onto a 2D plane. Specifically, given a mesh M and the planar domain D we seek a bijective parameterization $\phi : D \rightarrow M$ from a discrete set of planar points to the mesh vertex set. Ideally, this mapping would be isometric, however in general an isometric embedding cannot be achieved and shrinkage and expansion of mesh edges will occur. Since we later accurately account for the introduced distortion in the distance metric, any embedding algorithm may be used [2]. For this work, we compute an initial parameterization based on the estimation of the harmonic map [4], and iteratively refine it using a method proposed by Yoshizawa et al. [28] which minimizes the distortion in the surface area of the triangulation.

The results of the above embedding is a 2D sparse "image" of the 3D mesh vertices. In order to construct a regular and dense representation of the original surface, we interpolate a geometric entity associated with each of the vertex points in the 2D domain. Surface normals are a natural choice for this entity due to the fact that they are less affected by noise as compared to higher-order derivative

[2]Readers are referred to [5] for a recent survey on various $\mathbb{R}^3$ to $\mathbb{R}^2$ parameterization algorithms.

quantities such as curvature. Furthermore, they convey directional information of the surface geometry as opposed to scalar quantities such as mean or Gaussian curvature. Note that 3D coordinates cannot be used since they form the extrinsic geometry of the surface. Given a parameterization ϕ we construct a dense normal map $\mathbf{N} : \mathbb{R}^2 \rightarrow \mathbb{S}^2$, where $\mathbf{N}$ maps points in D to the corresponding 3D normal vector. Figure 1 (b) shows an example of such a normal map. The resulting normal map provides a dense and regular 2D representation of the original 3D geometry. Most importantly, the density of the normal map is independent of the mesh resolution of the given 3D model, therefore subsequent feature detection can be achieved at arbitrary precisions on the surface and is robust to changes in the sampling density on the original 3D model.

In order to accurately construct a scale-space representation of the original surface geometry, we require the relative geodesic distance between any two points on the normal map [3]. This geodesic distance can be computed by accounting for the distortion introduced by the embedding. Given a point $\mathbf{u} = (s,t) \in D$ that maps to a 3D mesh vertex $\phi(\mathbf{u})$ we may define its distortion $\varepsilon(\mathbf{u})$ as the average change in edge lengths connected to each vertex:

$$\varepsilon(\mathbf{u}) = \frac{1}{|\mathcal{A}(\mathbf{u})|} \sum_{\mathbf{v} \in \mathcal{A}(\mathbf{u})} \frac{\| \mathbf{u} - \mathbf{v} \|}{\| \phi(\mathbf{u}) - \phi(\mathbf{v}) \|} , \qquad (1)$$

where $\mathcal{A}(\mathbf{u})$ is a set of vertices adjacent to $\mathbf{u}$. We construct a dense distortion map by again interpolating over the values defined at each vertex. Figure 1(c) depicts the distortion map of the hand model in Figure 1(a).

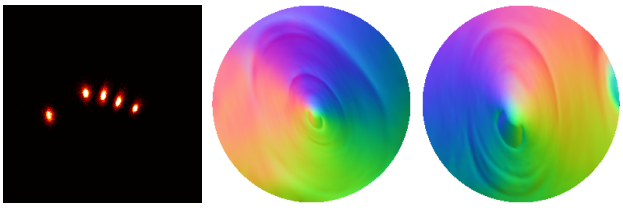
2.2. Multiple Normal Maps

Mesh parameterization algorithms often require 3D models to contain a natural boundary loop that is mapped to the boundary of the planar domain. For 3D models that do not contain such boundaries, we need to introduce cuts that map to the boundaries of the 2D domains, for instance one boundary cut for a genus-0 model, and compute multiple normal maps corresponding to different surface regions. At the same time, in general, surface regions mapped close to the boundary of the 2D domain are significantly distorted. In such areas the local neighbor of a 3D vertex is mapped into a highly skewed region, which can introduce errors in the subsequent filtering.

We introduce boundary cuts such that they avoid surface regions that are most likely to contain discriminative features. Specifically, for each 3D boundary cut, we manually select end points and automatically trace vertices with low curvature [4]. Furthermore, we construct a complimentary parameterization where portions of the model that were mapped to the perimeter of the original embedding are mapped to the central region of D. By using this complimentary parameterization together with the original embedding, we can ensure that every surface region is mapped to a region in the normal map where the local structure is well preserved.

[3] Note that there is also a global scaling between the 3D mesh model and its parameterized 2D image.

[4] This is the exact opposite of methods that try to minimize the overall distortion in the resulting parameterization [9].



(a) (b) (c)

Figure 2. Representing a 3D model using multiple normal maps. The heat map (a) illustrates the density of 3D vertices mapped to each point in the normal map shown in Figure 1(b). Many surface points on each finger are mapped to identical 2D points. (b) and (c) show two supplementary normal maps corresponding to two fingers of the model shown in Figure 1, which can be automatically computed given (a).

There can also be a considerable loss of information due to the finite resolution of the planar domain D. Often when a model has large appendages the discretization of the normal map results in 2D points which correspond to multiple mesh vertices. To compensate for this many-to-one mapping, we first compute a density map of vertices as shown in Figure 2(a). By clustering the points of this histogram into disjoint sets and parameterizing the corresponding portions of the mesh in each cluster, we construct a comprehensive set of supplementary normal maps. Figure 2(b) and (c) show two such supplementary normal maps corresponding to two fingers of the model shown in Figure 1. By representing the 3D model using multiple normal maps, we ensure that all portions of the surface are covered by the representation.

2.3. Geometric Scale-Space

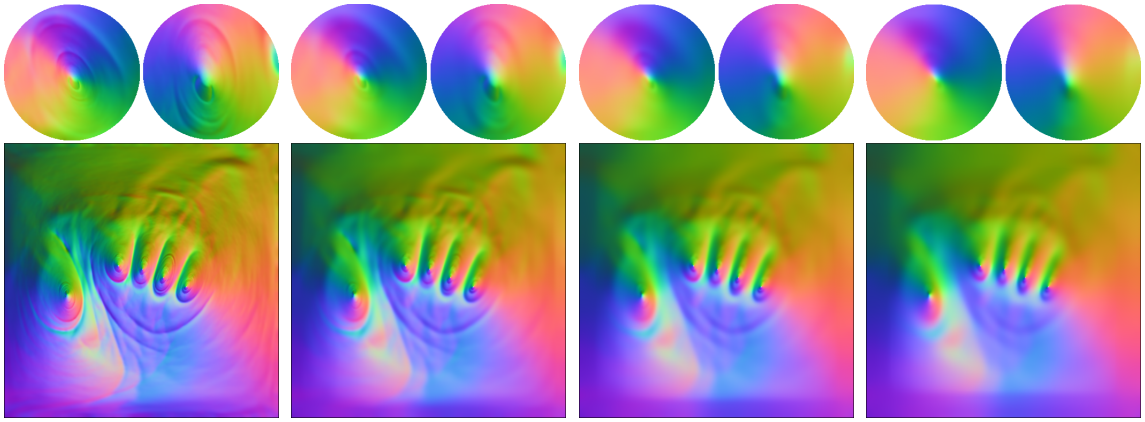
Given the normal map(s) $\mathbf{N}$ and its distortion map(s) ε we construct a scale-space representation of the original surface geometry. This geometric scale-space should encode the evolution of the surface normals on the 3D model while it is gradually smoothed. In other words, it is equivalent to computing the (2-)harmonic flow of a harmonic map from $\mathbb{R}^2$ to $\mathbb{S}^2$ [24]. The harmonic map is the minimizer of the harmonic energy,

$$\min_{\mathbf{N}:\mathbb{R}^2 \rightarrow \mathbb{S}^2} \int \int_D \| \nabla \mathbf{N} \|^2 \, dsdt , \qquad (2)$$

and the harmonic flow corresponds to the gradient-descent flow of the Euler-Lagrange equation of the harmonic energy,

$$\frac{\partial N_i}{\partial t} = \Delta N_i + N_i \| \nabla \mathbf{N} \|^2 \quad (i = 1,2,3) , \qquad (3)$$

where N_i is the i-th component of $\mathbf{N}$ and t corresponds to the scale level in the geometric scale-space.



(a) Scale $\sigma = 1$ (b) Scale $\sigma = 3$ (c) Scale $\sigma = 5$ (d) Scale $\sigma = 7$

Figure 3. The scale-space representation of the 3D hand shown in Figure 1 and Figure 2(b,c). As the standard deviation increases fine model details are smoothed away leaving only coarse geometric structures. For example the finger nail is quickly smoothed away, while the prominent creases on the palm remain visible even at the coarsest scale. Although the sizes of the finger nails in the two supplementary normal maps are different, the rate of smoothing is consistent due to the use of the geodesic Gaussian kernel that accounts for the distortion induced by the embedding.

The existence and uniqueness of the harmonic map for $\mathbb{R}^2 \rightarrow \mathbb{S}^2$ has been shown [6, 23]. Thus we are able to construct a unique geometric scale-space based on the normal map [5]. However, it has been shown that the harmonic flow is only partially regular and can create singularities in finite time. This means that the geometric scale-space computed based on the normal map may not satisfy the causality assumption – *"any feature at a coarse level of resolution is required to possess a cause at a finer level of resolution"* [11]. However, the cases where the harmonic flow is known to blowup are when the initial data (original normal map in our case) is highly symmetric and at least C^1-continuous [1, 10], which is very rare for real-world geometric data. Deriving the exact conditions that lead to non-causal geometric scale-space is a difficult problem which we leave as future work. For all the models in our experiments, we did not observe any singularities created in the computed geometric scale-space.

To construct a (discrete) geometric scale-space, instead of iteratively computing the gradient-descent flow of Equation 3, we convolve the normal map with a Gaussian kernel and renormalize the normals at each level [6]. As in 2D scale-space theory [15], the standard deviation of the Gaussian is monotonically increased from fine to coarse scale levels.

We use the geodesic distance as the distance metric to accurately construct a geometric scale-space that encodes the surface geometry. Given a 2D isotropic Gaussian centered at a point $\mathbf{u} \in D$, we define the value of the *geodesic Gaussian kernel* at a point $\mathbf{v}$ as

$$g(\mathbf{v}; \mathbf{u}, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{d(\mathbf{v}, \mathbf{u})^2}{2\sigma^2}\right] \tag{4}$$

where $d : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$ is the geodesic distance between the 3D surface points $\boldsymbol{\phi}(\mathbf{v})$ and $\boldsymbol{\phi}(\mathbf{u})$. The geodesic distance between two 3D points $\boldsymbol{\phi}(\mathbf{v})$ and $\boldsymbol{\phi}(\mathbf{u})$ is defined as the discretized line integral $d(\mathbf{u}, \mathbf{v})$ between $\mathbf{v}$ and $\mathbf{u}$ in the distortion map, which can be computed as

$$d(\mathbf{v}, \mathbf{u}) \approx \sum_{\mathbf{v}_i \in \mathcal{P}(\mathbf{v},\mathbf{u}), \neq \mathbf{u}} \frac{\varepsilon(\mathbf{v}_i)^{-1} + \varepsilon(\mathbf{v}_{i+1})^{-1}}{2} \parallel \mathbf{v}_i - \mathbf{v}_{i+1} \parallel , \tag{5}$$

where $\mathcal{P}(\mathbf{v}, \mathbf{u}) = [\mathbf{v}, \mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n, \mathbf{u}]$ is a list of points sampled on the line between $\mathbf{v}$ and $\mathbf{u}$. The density of this sampling determines the quality of the approximation of the original geodesic distance.

Using this geodesic Gaussian kernel, we compute the normal at point $\mathbf{u}$ for scale level σ as

$$\mathbf{N}^\sigma(\mathbf{u}) = \frac{\sum_{\mathbf{v} \in \mathcal{W}} \mathbf{N}(\mathbf{v}) g(\mathbf{v}; \mathbf{u}, \sigma)}{\parallel \sum_{\mathbf{v} \in \mathcal{W}} \mathbf{N}(\mathbf{v}) g(\mathbf{v}; \mathbf{u}, \sigma) \parallel}, \tag{6}$$

where $\mathcal{W}$ is a set of points in a window centered at $\mathbf{u}$. The window size is also defined in terms of the geodesic distance and is set proportional to σ at each scale level. In our implementation, we grow the window from the center point while evaluating each point's geodesic distance from the center to correctly account for the points falling inside the window. Figure 3 shows the normal map of the hand model and two supplementary normal maps at four scale levels. As the standard deviation of the Gaussian increases fine model details are smoothed away, leaving only coarse geometric structures.

[5] On the other hand, the harmonic energy for $\mathbb{B}^3 \rightarrow \mathbb{S}^2$ has infinite number of solutions [3] and hence a unique geometric scale-space cannot be constructed if the Euclidean distance is used.

[6] Observe that Equation 3 can be seen as a diffusion equation with an additional term rooting from the unit vector constraint. The iterative computation of harmonic flow is usually computed by first computing the gradient-descent flow for the diffusion term and renormalizing the vectors at each step [2].

3. Feature Detection in Geometric Scale-Space

In order to detect salient features in the geometric scale-space, we first derive the first- and second-order partial derivatives of the normal map $\mathbf{N}^\sigma$. Novel corner and edge detectors are then derived using these partial derivatives. An automatic scale-selection algorithm is introduced to unify the features detected at each scale into a single set of scale-dependent geometric features.

3.1. Derivatives of the Normal Map

We first derive the first-order partial derivatives of the 2D normal map in the horizontal (s) and vertical (t) directions. In the following we describe them only for the horizontal (s) direction. The partial derivatives in the vertical direction (t) may be derived by simply replacing s with t.

At any point in the normal map the horizontal direction corresponds to a unique direction on the tangential plane at the corresponding 3D point. The first-order derivative is thus the directional derivative of the normal along this specific direction in the tangential plane, known as the normal curvature. In the discrete domain D the normal curvature in the horizontal ($\mathcal{C}^s$) direction at a point $\mathbf{u} = (s, t)$ may be computed by numerical central angular differentiation:

$$\mathbf{N}_s(\mathbf{u}) = \frac{\partial \mathbf{N}(\mathbf{u})}{\partial s} = \mathcal{C}^s(\mathbf{u}) \approx \frac{\sin(\frac{1}{2}\theta(\mathbf{u}_{-1}, \mathbf{u}_{+1}))}{L(\mathbf{u}_{-1}, \mathbf{u}_{+1})} , \quad (7)$$

where $\mathbf{u}_{\pm 1} = (s \pm 1, t)$, $\theta(\mathbf{u}_{-1}, \mathbf{u}_{+1})$ is the angle between the normal vectors $\mathbf{N}(\mathbf{u}_{-1})$ and $\mathbf{N}(\mathbf{u}_{+1})$, and $L(\mathbf{u}_{-1}, \mathbf{u}_{+1})$ is the chord length between the 3D points $\boldsymbol{\phi}(\mathbf{u}_{-1})$ and $\boldsymbol{\phi}(\mathbf{u}_{+1})$. Because the normal curvature is a function of adjacent points in the 2D domain D the chord length L is simply the geodesic distance between these points. After applying the discrete geodesic distance in Equation 5 we obtain

$$\mathbf{N}_s(\mathbf{u}) \approx \frac{\sin(\frac{1}{2}\theta(\mathbf{u}_{-1}, \mathbf{u}_{+1}))}{d(\mathbf{u}_{-1}, \mathbf{u}_{+1})} . \quad (8)$$

Note that because the angle between the two normal vectors is in the range $[0, \pi]$, the first-order derivative is nonnegative at both convex and concave surface points – it is unsigned.

The second-order derivative of the normal map can be derived as

$$\mathbf{N}_{ss}(\mathbf{u}) = \frac{\partial^2 \mathbf{N}(\mathbf{u})}{\partial s^2} = \frac{\partial C^s(\mathbf{u})}{ds} . \quad (9)$$

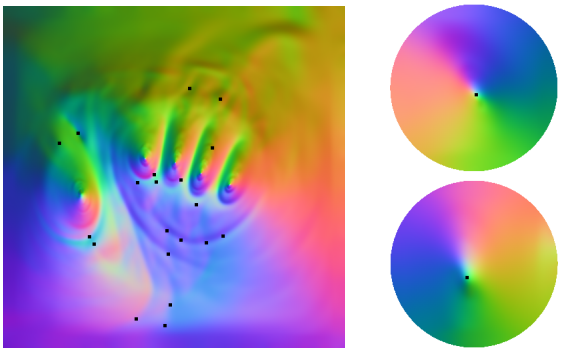
After applying the chain rule to Equation 7 we obtain

$$\begin{aligned} \mathbf{N}_{ss}(\mathbf{u}) \approx & \frac{\partial \theta(\mathbf{u}_{-1}, \mathbf{u}_{+1})}{\partial s} \frac{\cos\left(\frac{1}{2}\theta(\mathbf{u}_{-1}, \mathbf{u}_{+1})\right)}{L(\mathbf{u}_{-1}, \mathbf{u}_{+1})} \\ & - \frac{\partial L(\mathbf{u}_{-1}, \mathbf{u}_{+1})}{\partial s} \frac{2\sin\left(\frac{1}{2}\theta(\mathbf{u}_{-1}, \mathbf{u}_{+1})\right)}{L(\mathbf{u}_{-1}, \mathbf{u}_{+1})^2} . \end{aligned}$$

Since we can safely assume that the parameterization induces a uniform distortion between every adjacent point in D, the derivative of the chord length L will be zero, and the second term vanishes. After applying numerical central differentiation to θ and using the half angle formula, the second-order derivative reduces to

$$\mathbf{N}_{ss}(\mathbf{u}) \approx \frac{\theta(\mathbf{u}_{-2}, \mathbf{u}) - \theta(\mathbf{u}_{+2}, \mathbf{u})}{d(\mathbf{u}_{-1}, \mathbf{u}_{+1})} \frac{\sqrt{\frac{1}{2}\left(1 + \mathbf{N}(\mathbf{u}_{-1}) \cdot \mathbf{N}(\mathbf{u}_{+1})\right)}}{d(\mathbf{u}_{-1}, \mathbf{u}_{+1})} . \quad (10)$$

This form is particularly attractive as it enables us to compute the second-order derivative in terms of the original normal vectors, and the change in the local angle. The noise associated with higher-order derivatives is reduced as we have avoided an additional numerical differentiation of the first-order derivatives.



(a) $\sigma = 3$ (b) $\sigma = 7$

Figure 4. Corners detected on the 2D normal map. (a) illustrates the 20 strongest corners on the 2D representation of the hand model at scale $\sigma = 3$. Observe that the corner points on the palm are primarily located where two creases converge, or where there is an acute bend in one crease. (b) shows the strongest corner on two of the finger normal maps at scale $\sigma = 7$. At this coarse scale the corners are detected on the tip of the finger.

3.2. Corners

Consider, the hand model shown in Figure 1. We wish to detect geometrically meaningful corners such as the finger tips as well as the points on the sharp bends of the palm prints. In other words, we are interested in detecting two different types of geometric corners, namely points that have high curvature isotropically or in at least two distinct tangential directions. The rich geometric information encoded in the normal maps enable us to accurately detect these two types of 3D corners using a two-phase geometric corner detector.

We begin by computing the Gram matrix $\mathcal{M}$ of first-order partial derivatives of the normal map $\mathbf{N}^\sigma$ at each point. The Gram matrix at a point $\mathbf{u}$ is defined as

$$\mathcal{M}(\mathbf{u}; \sigma, \tau) = \sum_{\mathbf{v} \in \mathcal{W}} \begin{bmatrix} \mathbf{N}^\sigma_s(\mathbf{v})^2 & \mathbf{N}^\sigma_s(\mathbf{v})\mathbf{N}^\sigma_t(\mathbf{v}) \\ \mathbf{N}^\sigma_s(\mathbf{v})\mathbf{N}^\sigma_t(\mathbf{v}) & \mathbf{N}^\sigma_t(\mathbf{v})^2 \end{bmatrix} g(\mathbf{v}; \mathbf{u}, \tau) , \quad (11)$$

where $\mathcal{W}$ is the local window around the point $\mathbf{u}$. $\mathcal{M}$ has two parameters, one that determines the particular scale in the scale-space representation (σ), and one that determines the weighting of each point in the Gram matrix (τ)[7]. The

[7] In our experiments we set $\tau = \sigma/2$

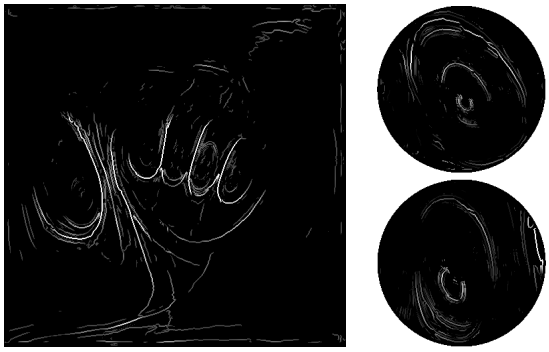
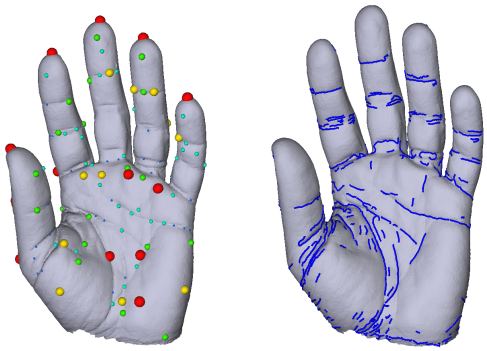


Figure 5. Edges detected at one scale level ($\sigma = 1$). The edges are detected accurately on surface points with locally maximum curvature, namely 3D ridges and valleys. Here the the creases of the palm form the edges.



(a) (b)

Figure 6. Scale-dependent geometric corners (a) and edges (b) detected on the hand model. The corners are represented with 3D spheres which are colored and sized according to their respective scale (blue and red correspond to the finest and coarsest scales, respectively). The corners accurately represent the geometric scale variability of the model, for instance with fine corners on the palm creases and coarse corners at the tips of the fingers. The edges also encode the geometric scale variability, tracing edge segments that arise at different scales.

corner response at a point $\mathbf{u}$ is defined as the maximum eigenvalue of $\mathcal{M}$. However, due to the unsigned first-order derivative the resulting corner set will contain not only the aforementioned two desired types of geometric corners, but will also contain points lying on 3D edges.

The second-order derivatives of the normal map can be used to prune the corners lying along the 3D edges. We first prune the corner points that are not centered on zero crossings in both the horizontal and vertical directions. Next we keep only those points where the variance of the second-order partial derivatives around the point $\mathbf{u}$ are within a constant factor of each other. The closer this constant factor is to 1, the greater the geometric variance of the selected corner points in both tangential directions. Figure 4 illustrates corners detected on the hand model shown in Figure 1 at one scale level. Once the corners are detected in 2D they can be mapped back to the 3D model. Because the 2D normal map is dense, the corresponding location of the corners in 3D are independent of the input model's triangulation.

3.3. Edges

In order to find edges at each scale level in the geometric scale-space we use the second-order derivatives of the normal map. Although the first-order derivative is unsigned, locating edges using the zero crossing of the second-order derivative is sufficient, as the sign only affects the profile of the derivative values around the zero crossing, and not the actual location of the zero crossing.

Similar to the classic work of Marr and Hildreth [17] for 2D images, given a normal map, we begin by computing the Laplacian, defined as

$$\nabla^2 \mathbf{N}^\sigma = \mathbf{N}^\sigma_{ss} + \mathbf{N}^\sigma_{tt} \tag{12}$$

Next we construct a binary image that contains the zero-crossing of the Laplacian. This set of zero crossings contains points centered on curvature maxima, as well as spurious edge points arising from uniform or slow changing curvature regions. We remove the spurious edge points by thresholding the magnitude of the first-order derivative, and the variance of the second-order derivative. This ensures that edges are detected in high curvature regions, and lie along portions of the surface with a significant variation in the surface geometry.

Figure 5 shows an example result of edge detection on the hand model at one scale level ($\sigma = 1$). Again, these edges are localized on the surface of the 3D model independent of the mesh resolution. Additional post-processing may also be applied to the edges once they are mapped onto the 3D model. In latter experiments, we first compute a minimum spanning tree of the 3D edge points, where the magnitude of the edge response in 2D determines the weight of the 3D point similar to [20]. Then we decompose the tree into a set of disjoint edge paths via caterpillar decomposition and fit NURBs curves to each of these paths to obtain smooth parametric 3D edges.

3.4. Automatic Scale Selection

Once features are detected in each of the scale-space normal maps $\mathbf{N}^\sigma$, they can be unified into a single feature set. Although a feature may have a response at multiple scales, it intrinsically exists at the scale where the response of the feature detector is maximized. By determining this intrinsic scale for each feature we obtain a comprehensive scale-dependent 3D geometric feature set.

In order to find the intrinsic scale of a feature we search for local maxima of the normalized feature response across a set of discrete scales, analogous to the 2D automatic scale selection method [16]. The derivatives are normalized to account for a decrease in the derivative magnitude as the normal maps are increasingly blurred. We define the normalized first-order derivatives $\widetilde{\mathbf{N}}^\sigma_s$ and $\widetilde{\mathbf{N}}^\sigma_t$ as

$$\widetilde{\mathbf{N}}_s^\sigma = \sigma^\gamma \mathbf{N}_s^\sigma \quad \text{and} \quad \widetilde{\mathbf{N}}_t^\sigma = \sigma^\gamma \mathbf{N}_t^\sigma \,, \tag{13}$$

where γ is a free parameter that is empirically set for each particular feature detector. The corresponding normalized second-order derivatives are defined as

$$\widetilde{\mathbf{N}}_{ss}^\sigma = \sigma^{2\gamma} \mathbf{N}_{ss}^\sigma \quad \text{and} \quad \widetilde{\mathbf{N}}_{tt}^\sigma = \sigma^{2\gamma} \mathbf{N}_{tt}^\sigma \,. \tag{14}$$

Normalized feature responses are computed by substituting the normalized derivatives into the corner and edge detectors presented in the previous two sections. The final scale-dependent geometric feature set is constructed by identifying the points in scale-space where the normalized feature response is maximized along the scale axis and locally in a spatial window. Figure 6 illustrates the scale-dependent geometric corners and edges of the hand model. The scale-dependent geometric features accurately encode the geometric scale-variability and can clearly be used as a unique representation of the underlying geometry.

4. Experiments

We evaluated the effectiveness and robustness of the proposed method for computing scale-dependent geometric features on several 3D models. The method was applied to 3 different models, in addition to the hand model shown in Figure 6. One of these models, the Julius Caesar, is of disk topology and two, the armadillo and Buddha, have a genus of zero.

4.1. Corner and Edge Detection

Figure 7 illustrates the corners and edges detected on the three models. The armadillo model has appendages that were significantly distorted in the 2D representation and therefore multiple normal maps were used. Additionally the large distortion at the boundaries of the armadillo was accounted for using a complimentary parameterization. The set of scales used to detect the corners and edges depend on the geometry of the model and were set empirically. Observe that the set of corners is distributed across scales, and that the scale of a particular corner reflects the scale of the underlying surface geometry. For instance the tip of Caesar's nose is detected at the coarsest scale, while the corners of the mouth are detected at a relatively finer scale. The edges are detected along ridges and valleys of the 3D models existing at different scales, for example the edges on the prominent creases of the Buddha's robe, as well as edges along the finer details of the base.

4.2. Noisy Surface Normals

We tested the resilience of our framework to noisy input data by applying Gaussian random noise with standard deviation 0.05, 0.075, and 0.1 to the surface normals of the Julius Caesar model. The features were detected with the identical parameters used to detect the original corner set. Figure 8(a) illustrates the results. Although fine scale corners can arise from the input noise, the detected scale-dependent geometric corner set are highly consistent with the those detected on the original model and localized accurately (Figure 7).

4.3. Varying Sampling Densities

We demonstrate the independence of our framework from surface sampling density by computing scale-dependent geometric corners on three simplified Julius Caesar models. Specifically, we applied a surface simplification algorithm [7] to construct Julius Caesar models with $30,000$, $20,000$ and $10,000$ faces from the original model with $50,000$ triangle faces. Corners were detected at each sampling density using the parameters from the original experiment (Figure 7). Figure 8(b) illustrates the results. Although the number of faces changes substantially, the location and scale of the corners remain largely constant. This demonstrates that the density of the 2D representation of the surface geometry ensures that the framework is independent of the surface sampling.

5. Conclusion

In this paper we presented a comprehensive framework for detecting scale-dependent 3D geometric features, in particular corners and edges, of a 3D model of arbitrary topology. This was achieved by constructing and analyzing a scale-space representation of the surface geometry encoded by the surface normals embedded in a regular and dense 2D domain. Experimental results show that we can robustly detect and localize scale-dependent features with our method.

The detected scale-dependent geometric features faithfully encode the scale variability of the underlying geometry. Our framework is general and enables one to exploit this hidden dimension of geometric data in any application or computational task that relies on 3D geometric data. The scale-dependent geometric features can also be used to define various local or global representations of the 3D geometry. For instance, we are currently investigating the derivation of scale-invariant descriptors based on the analysis of the geometric scale-space and concise global representations of scale-dependent features that encode the spatial distribution of them.

References

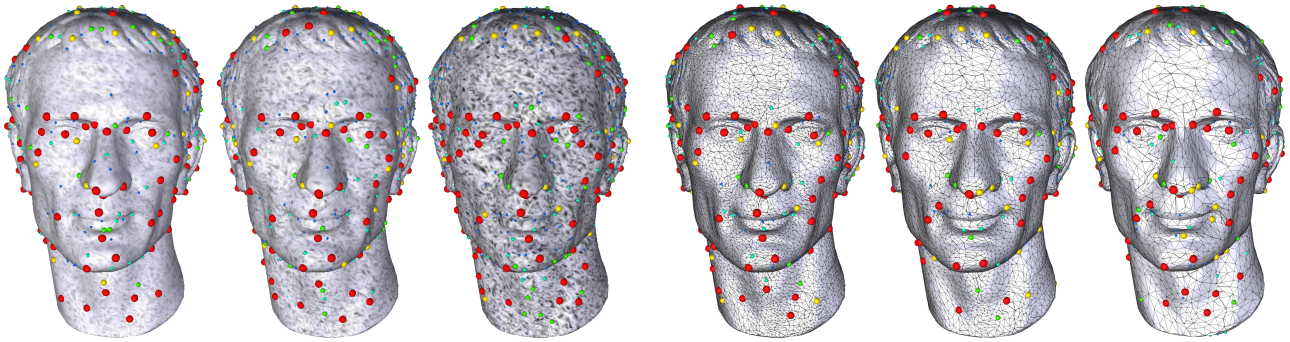
[1] K-C. Chang, W.Y. Ding, and R. Ye. Finite-Time Blow-Up of the Heat Flow of Harmonic Maps from Surfaces. *Journal of Differential Geometry*, 36(2):507–515, 1992.

[2] R. Cohen, R.M. Hardt, D. Kinderlehrer, S.Y. Lin, and M. Luskin. *Minimum Energy Configurations for Liquid Crystals: Computational Results*, pages 99–121. Springer-Verlag: New York, 1987.

[3] J.-M. Coron. Nonuniqueness for the Heat Flow of Harmonic Maps. *Annales de l'institut Heri Poincaré*, 7(4):335–344, 1990.

[4] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzl. Multiresolution Analysis of Arbitrary Meshes. In *EUROGRAPHICS 2002*, pages 209–218, 2002.

Figure 7. Scale-dependent geometric corner and edge detection results on a disc topology model (Caesar) and two genus zero models (Buddha and armadillo). The corner and edges are accurately detected across different scales. The resulting scale-dependent geometric feature set encode the scale variability of the underlying surface geometry resulting in a unique representation of each model.



(a) Models with input noise of 0.05, 0.075 and 0.1

(b) Models with 30,000, 20,000 and 10,000 faces

Figure 8. Scale-dependent geometric corner detection with the presence of noisy surface normals (a), and with varying surface sampling densities (b). When compared with the corners shown in Figure 7 the results demonstrate that the scale-dependent corners detected with our framework are largely invariant to significant input noise and variations in the sampling density.

[5] M.S. Floater and K. Hormann. Surface Parameterization: A Tutorial and Survey. In M. S. Floater and M. A. Sabin, editors, *Advances in Multiresolution for Geometric Modelling*, pages 157–186. Springer-Verlag, 2005.

[6] A. Freire. Uniqueness for the Harmonic Map Flow in Two Dimensions. *Calculus of Variations and Partial Differential Equations*, 3(1):95–105, 1995.

[7] M. Garland and P. Heckbert. Surface Simplification Using Quadric Error Metrics. In *ACM SIGGRAPH*, pages 209–216, 1997.

[8] N. Gelfand, N.J. Mitra, L.J. Guibas, and H. Pottmann. Robust Global Registration. In *Symposium on Geometry Processing*.

[9] X. Gu, S. Gortler, and H. Hoppe. Geometry Images. In *ACM SIGGRAPH*, pages 355–361, 2002.

[10] R.M. Hardt. Singularities of Harmonic Maps. *Bulletin of The American Mathematical Society*, 34:15–34, 1991.

[11] J.J. Koenderink. The Structure of Images. *Biological Cybernetics*, 50:363–370, 1984.

[12] J. Lalonde, R. Unnikrishnan, N. Vandapel, and M. Hebert. Scale Selection for Classification of Point-sampled 3-d Surfaces". In *Int'l Conf. on 3-D Digital Imaging and Modeling*, 2005.

[13] C.H. Lee, A. Varshney, and D.W. Jacobs. Mesh Saliency. *ACM Trans. on Graphics*, 24(3):659–666, 2005.

[14] X. Li and I. Guskov. Multi-scale Features for Approximate Alignment of Point-based Surfaces. In *Symposium on Geometry Processing*, 2005.

[15] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academics Publishers, 1994.

[16] T. Lindeberg. Feature Detection with Automatic Scale Selection. *Int'l Journal of Computer Vision*, 30:77–116, 1998.

[17] D. Marr and E. Hildreth. Theory of Edge Detection. *Proc. Royal Society London*, 207:187 – 217, 1980.

[18] F. Mokhtarian, N. Khalili, and P. Yuen. Cuvature Computation on Free-Form 3-D Meshes at Multiple Scales. *Computer Vision and Image Understanding*, 83:118–139, 2001.

[19] J. Novatnack, K. Nishino, and A. Shokoufandeh. Extracting 3D Shape Features in Discrete Scale-Space. In *Third Int'l Symposium on 3D Data Processing, Visualization and Transmission*, 2006.

[20] M. Pauly, R. Keiser, and M. Gross. Multi-scale Feature Extraction on Point-sampled Surfaces. *EUROGRAPHICS*, 21(3), 2003.

[21] M. Pauly, L. P. Kobbelt, and M. Gross. Point-Based Multi-Scale Surface Representation. *ACM Trans. on Graphics*, 25(2), 2006.

[22] M. Schlattmann. Intrinsic Features on Surfaces. In *Central European Seminar on Computer Graphics*, pages 169–176, 2006.

[23] M. Struwe. On the Evolution of Harmonic Mappings of Riemannian Surfaces. *Commentarii Mathematici Helvetici*, 60(1):558–581, 1985.

[24] B. Tang, G. Sapiro, and V. Caselles. Diffusion of General Data on Non-Flat Manifolds via Harmonic Maps Theory: The Direction Diffusion Case. *Int'l Journal of Computer Vision*, 36(2):149–161, 2000.

[25] G. Taubin. A Signal Processing Approach to Fair Surface Design. In *ACM SIGGRAPH*, pages 351–358, 1995.

[26] J. Weickert, S. Ishikawa, and A. Imiya. Linear Scale-Space has First been Proposed in Japan. *Journal of Mathematical Imaging and Vision*, 10(3):237–252, 1999.

[27] A.P. Witkin. Scale-Space Filtering: A New Approach to Multi-Scale Description. In *IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, pages 150–153, 1984.

[28] S. Yoshizawa, A. Belyaev, and H-P. Seidel. A Fast and Simple Stretch-Minimizing Mesh Parametrization. In *Int'l Conf. on Shape Modeling and Applications*, pages 200–208, 2004.