

# Multi-scale Interest Regions from Unorganized Point Clouds

Ranjith Unnikrishnan      Martial Hebert  
Carnegie Mellon University, Pittsburgh, PA 15213  
ranjith,hebert@cs.cmu.edu

## Abstract

Several computer vision algorithms rely on detecting a compact but representative set of interest regions and their associated descriptors from input data. When the input is in the form of an unorganized 3D point cloud, current practice is to compute shape descriptors either exhaustively or at randomly chosen locations using one or more preset neighborhood sizes. Such a strategy ignores the relative variation in the spatial extent of geometric structures and also risks introducing redundancy in the representation. This paper pursues multi-scale operators on point clouds that allow detection of interest regions whose locations as well as spatial extent are completely data-driven. The approach distinguishes itself from related work by operating directly in the input 3D space without assuming an available polygon mesh or resorting to an intermediate global 2D parameterization. Results are shown to demonstrate the utility and robustness of the proposed method.\*

## 1. Introduction

Many approaches to solving computer vision problems work with compact intermediate representations of the input data. A popular approach to problems like object recognition and scan registration from 3D point clouds is to start by selecting a set of regions in the data, and then reducing the input data to a collection of descriptors computed over those regions [4, 12]. This strategy of using local regions has proven to be well-suited to take on the practical challenges of occlusion, clutter and intra-class variation.

Similar approaches exist for the same problems in the 2D image domain, one example of which is the popular bag-of-words model. The success of these local representations in image processing has been largely driven by developments in building multi-scale representations of images [10] and by their application to finding distinguished

\*We gratefully acknowledge collaborative participation in the Robotics Consortium sponsored by the U.S Army Research Laboratory under the Collaborative Technology Alliance Program (Cooperative Agreement DAAD19-01-209912) that made this research possible.

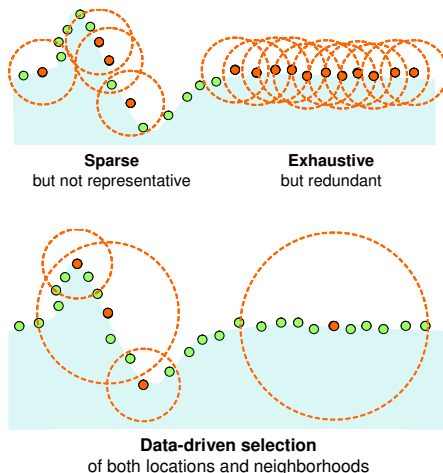


Figure 1. Random selection of locations for computing shape descriptors can miss geometric structures whose spatial extent are not comparable to the preset choice of support radius. Exhaustive selection strategies introduce redundancy in areas with little shape variation. A data-driven approach can judiciously choose both locations and associated support radius if guided by local shape.

scale-invariant interest regions automatically [11, 13].

However, when the input is in the form of an unorganized 3D point cloud, the interest region selection methods currently available are relatively unprincipled in comparison. Current practice [4, 12] consists of computing a shape descriptor [6, 1] either exhaustively at each point or at randomly distributed locations in the point cloud. Furthermore, the descriptor is computed over a spatial extent that is usually preset to one or more values that are based largely on intuition, instead of being guided by the available data.

Why is there such a huge difference between approaches for processing 2D images and 3D point clouds?

The main reason for this inconsistency is that while a rigorous theory of scale selection exists for images [5, 10], a direct equivalent does not really exist for unorganized point clouds. Scale theory for images has focused largely on analyzing functions observed on a regular 2D (or even  $n$ -D) lattice. However point clouds lack any organized lattice structure. Unlike image intensity, the observed geometry

“signal” in point clouds is implicit and is represented only through the spatial arrangement of the observed locations. Thus the applicability of traditional scale theory for images to discrete point cloud data is unclear.

This paper addresses the above inconsistency through two contributions. First, we derive multi-scale filtering operator for point clouds that captures variation in shape at a point relative to its neighborhood, working analogously to the Laplacian filter in 2D images, but with the property of being invariant to local changes in sampling density.

Second, we show how this filter may be used in an algorithm for interest region detection working *directly* in the input point cloud domain and *without* relying on polygonal meshes or accurate surface normals. We wish to emphasize that our goal is not to develop yet another kind of 3D shape descriptor, but to provide a principled way of *repeatably* selecting regions over which descriptors should be computed. By this, we enable a change in current practice from using random locations and preset neighborhood sizes to using a completely data-driven strategy that finds locations and their associated support radii automatically.

### 1.1. Related work

To distinguish our proposed approach from other related methods, it is necessary to understand how other methods draw connections to scale-space axioms from the 2D image domain. It is also useful to identify cases where these interpretations are insufficient or even incorrect for point clouds. To do this, we begin by summarizing the development of scale theory for images and mention its salient points.

**Scale theory in 2D images.** The beginning of scale theory in images is usually attributed to Witkin [19], who proposed a scale space representation as a transformation of an input signal  $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$  to a one-parameter family of signals  $f(\mathbf{x}, t) : \mathbb{R}^d \times \mathbb{R}^+ \rightarrow \mathbb{R}$ , where the non-negative  $t$  denotes the *scale parameter*. Witkin obtained a *Gaussian scale space* representation by convolving  $f(\mathbf{x})$  with Gaussians of increasing width as  $f(\mathbf{x}, t) = f(\mathbf{x}) * G(\mathbf{x}, t)$  where  $G(\mathbf{x}, t) = (2\pi t^2)^{-1/2} \exp(-\|\mathbf{x}\|^2/2t^2)$  and the asterisk denotes convolution.

Koenderink [7] pointed out the connection between Gaussian scale space and the *diffusion equation*

$$\frac{\partial f(\mathbf{x}, t)}{\partial t} = \Delta f(\mathbf{x}, t) \equiv \sum_i^d \frac{\partial^2 f(\mathbf{x}, t)}{\partial x_i^2}, \quad (1)$$

where  $\Delta$  denotes the *Laplacian* operator. When subject to the initial condition  $f(\mathbf{x}, 0) = f(\mathbf{x})$ , the solution of (1) can be shown to be Gaussian convolution with  $G$  as the unique Green’s function for the above system.

Lindeberg [10] later showed that the amplitude of scale-normalized derivative operators assumes a unique maximum for a value of  $t$  proportional to the “wavelength” of the

signal at that point. This property provides a way to choose feature support regions in proportion to this characteristic length of the signal.

Lowe [11] proposed to linearize the diffusion equation with a simple forward explicit Euler scheme so that successive convolutions of the signal may be obtained as

$$f(\mathbf{x}, t + \delta t) = f(\mathbf{x}, t) + \delta t \Delta f(\mathbf{x}, t). \quad (2)$$

**3D Mesh processing.** Several methods have been proposed to extend the above connection of 2D scale space with the diffusion equation to surfaces represented by a *connectivity mesh*. Seminal work by Taubin [17] replaced the continuous Laplacian operator  $\Delta$  from the diffusion equation by its discrete counterpart, the *graph Laplacian*  $L_g$ . For a set of sample points  $\{\mathbf{x}_i\}$  forming vertices of a graph, the unnormalized graph Laplacian is defined as the operator over a function  $f$  on the points as

$$L_g f(\mathbf{x}_i) = \sum_{j, \{i,j\} \in \mathbb{E}} (f(\mathbf{x}_i) - f(\mathbf{x}_j)) w_{ij}, \quad (3)$$

where the summation is over graph edges  $(i, j)$  in edge set  $\mathbb{E}$ , and  $w_{ij}$  are positive edge weights.

By using the graph Laplacian  $L_g$  and replacing the scalar image intensity function  $f(\mathbf{x})$  in (2) by the Euclidean coordinate vector  $\mathbf{x}$ , one can progressively “smooth” the 3D points by successive application of (2).

Our main criticism of this line of work is that, by modifying the 3D points directly, we believe it is trying to solve a different problem. Altering the extrinsic geometry as part of a multi-scale representation is akin to changing not only the pixel values but also the pixel coordinates in images, and incorrectly changes the observations. This different problem of mesh simplification, while still useful for tasks like progressive rendering and compression, is not directly applicable to multi-scale interest region detection.

**Other relevant work.** Pauly *et al.* [15] measure a quantity termed *surface variation*, given by  $\sigma_n(x) = \lambda_0 / (\lambda_0 + \lambda_1 + \lambda_2)$ , where the  $\lambda_i$ ’s are eigenvalues of the sample covariance matrix computed in a local  $n$ -point neighborhood of sample point  $x$ . They propose the natural scale at a point to be the neighborhood size for which the corresponding  $\sigma_n$  achieves a local extremum. However the variation in  $\sigma_n$  is extremely noisy and heuristic pre-smoothing procedures have to be applied in order to recover any trends in its variation.

Recent work [14] presented a method to detect multi-scale corner and edge features by constructing a global 2D parameterization of the data and filtering surface normals in the 2D space. The approach relies strongly on having a connectivity mesh to faithfully construct the 2D parameterization, and also on the prior availability of good surface normals, as does [9]. Also none of the above methods address the *variability* in the distribution of the point samples.

## 2. Multi-scale operators for point clouds

From studying the related work, we may conclude that discretization of the 2D diffusion equation is not the appropriate starting point for developing scale-space analogies in unorganized point clouds. In this section we will pursue a different line of reasoning to develop multi-scale operators while preserving the advantages of working in the original input space and not relying on good initial estimates of surface normals or meshes. We will build on these operators in Section 3 to devise an algorithm for estimating at each point a natural scale that is representative of its local shape. Results using this algorithm are presented in Section 4 followed by discussion of the approach in Section 5.

As was first reported by Weickert [18], Gaussian scale space theory was axiomatically derived by Iijima [5] as far back as 1959. Iijima's starting point [18] was to define the mathematical form of an *integral operator* that maps input functions to their one-parameter multi-scale representations. We will follow this route by first defining the form of such an operator for continuous surfaces, and then progressively modify the operator to satisfy the various requirements of our problem domain.

Our general strategy will be to determine not point locations but *values* defined on the points that will reflect the variation in local mean curvature of the underlying shape at each point. As importantly, we will also show how these values can be computed independently of the sampling distribution generating those points.

### 2.1. Case of 2D curves

We start by considering the integral operator  $A : \mathbb{R}^d \times \mathbb{R}^+ \rightarrow \mathbb{R}^d$ , with  $d = 2$  for the 2D case.  $A$  is defined to have the form

$$A(\alpha(s), t) = \int_{\Gamma} \phi(s, u, t) \alpha(u) du \quad (4)$$

that operates on positions on a *smooth* 2D curve  $\Gamma$  parameterized by distance as the function  $\alpha(s) : \mathbb{R} \rightarrow \mathbb{R}^2$ . Note that (i) the form of the transformation is linear, and (ii) the integration is performed *along* the curve  $\Gamma$  and not in the Euclidean  $\mathbb{R}^2$  space. The implication of the latter is that by defining the operator in terms of intrinsic geometry, there is no reference to an extrinsic coordinate system. Thus the operator is invariant to rigid transformations in the 2D space.

For the operator to be translation invariant in the 1D intrinsic system, the kernel  $\phi$  must be representable in the form  $\phi(s - u, t)$ . We fix the kernel to be a Gaussian

$$\phi(s, u, t) = (2\pi t^2)^{-\frac{1}{2}} \exp\left(-\frac{(s-u)^2}{2t^2}\right), \quad (5)$$

so that  $\int_{-\infty}^{\infty} \phi(s, u, t) du = 1$ .

How does the above operator relate to the geometry of the curve? To answer this, let us fix a local coordinate system at a point  $\mathbf{x} = \alpha(0)$ , for convenience, and examine the effect of the operator at the point. By Taylor expansion of  $\alpha(u)$  around  $u = 0$ ,

$$\alpha(u) = \mathbf{x} + u\dot{\alpha}(0) + \frac{u^2}{2}\ddot{\alpha}(0) + \dots,$$

we can derive

$$\begin{aligned} A(\mathbf{x}, t) &= \int_{-\infty}^{\infty} (2\pi t^2)^{-\frac{1}{2}} e^{-\frac{u^2}{2t^2}} \alpha(u) du \\ &\approx \mathbf{x} + (2\pi t^2)^{-\frac{1}{2}} \int_{-\infty}^{\infty} \frac{u^2}{2} e^{-\frac{u^2}{2t^2}} \ddot{\alpha}(0) du \quad (6) \\ &= \mathbf{x} + \ddot{\alpha}(0) \frac{t^2}{2} = \alpha(0) + \kappa_{\mathbf{x}} \mathbf{n}_{\mathbf{x}} \frac{t^2}{2}. \end{aligned}$$

Hence the effect of the operator is to displace the point  $\mathbf{x}$  in direction of the normal  $\mathbf{n}_{\mathbf{x}}$  to the curve, and in proportion to its curvature  $\kappa_{\mathbf{x}}$ .

### 2.2. Extension to 3D surfaces

Using the previous result, it is straightforward to extend the operator from (4) to the case of a surface in 3D. Consider now the neighborhood around the origin  $\mathbf{x}$  on a surface  $\mathcal{M}$  having normal direction  $\mathbf{n}_{\mathbf{x}}$ . There then exists a family of planes  $\Pi_{\theta}$  that all contain  $\mathbf{n}_{\mathbf{x}}$  and whose normals lie in the tangent space of  $\mathbf{x}$  at angle  $\theta$  to some reference tangent. The intersection of each plane  $\Pi_{\theta}$  with the surface  $\mathcal{M}$  is a *normal curve*  $\Gamma_{\theta}$  parameterized by the function  $\alpha_{\theta}(0)$ .

Using the fact that the normal to each curve  $\alpha_{\theta}(0)$  at  $\mathbf{x}$  is  $\mathbf{n}_{\mathbf{x}}$ , we can deduce for each normal curve that

$$A(\mathbf{x}, t) = A(\alpha_{\theta}(0), t) \approx \mathbf{x} + \kappa_{\mathbf{x}, \theta} \mathbf{n}_{\mathbf{x}} \frac{t^2}{2},$$

where  $\kappa_{\mathbf{x}, \theta}$  is the *normal curvature* associated with tangent direction  $\theta$ . We can average over all tangent vector directions  $\theta$  to get

$$A(\mathbf{x}, t) \approx \mathbf{x} + \frac{1}{2\pi} \int \kappa_{\mathbf{x}, \theta} \mathbf{n}_{\mathbf{x}} t^2 d\theta = \mathbf{x} + H \mathbf{n}_{\mathbf{x}} t^2, \quad (7)$$

where  $H$  is the *mean curvature* at the origin.<sup>1</sup>

Any two orthogonal tangent directions at angles  $\theta$  and  $\theta + \pi/2$  may be used to form a local coordinate system at  $\mathbf{x}$ . Using the property of mean curvature on surfaces  $H = (\kappa_{\theta} + \kappa_{\theta + \pi/2})/2$  gives

$$A(\mathbf{x}, t) \approx \mathbf{x} + \left[ \ddot{\alpha}_{\theta}(0) + \ddot{\alpha}_{\theta + \frac{\pi}{2}}(0) \right] \frac{t^2}{2} = \mathbf{x} + \frac{t^2}{2} L_{\mathcal{M}} \mathbf{x},$$

where  $L_{\mathcal{M}}$  is the **Laplace-Beltrami (LB) operator** [3, 16]. The LB operator is the natural analogue of the Laplacian  $\Delta$  from Euclidean space, but operates in an *intrinsic* coordinate system defined on a manifold. The modified interpretation of the  $A$  operator is that of displacing a point along its normal direction  $\mathbf{n}$  in proportion to its *mean curvature*  $H$ .

<sup>1</sup>It should be clear that  $H$  varies with location  $\mathbf{x}$ , and an explicit subscript is omitted for clarity.

### 2.3. Non-uniform sampling

So far, our continuous domain analysis was done under the assumption that the points on the curve or surface were uniformly distributed. In reality, the nature of sensor geometry induces a variation in sampling density that needs to be accounted for. Our method will follow a structure similar to Lafon's [8] analysis of the discrete Laplacian operator, with the important differences that our chosen operator  $A(\alpha(s), t)$  integrates over the sub-manifold instead of in Euclidean  $\mathbb{R}^3$  space, and is analyzed as specifically applied to the extrinsic surface coordinate function  $\alpha(s)$ .

We consider again from Section 2.1 the case where points are sampled from the 2D curve  $\Gamma = \alpha(s)$  but now follow an unknown but smooth probability distribution  $p(s)$ . The expected value of the operator  $A$  may then be obtained by integrating over the modified measure  $\mu(s) = p(s)ds$ , so that

$$A(\alpha(0), t) = \frac{1}{d(0, t)} \int_{-\infty}^{\infty} (2\pi t^2)^{-\frac{1}{2}} e^{-\frac{u^2}{2t^2}} \alpha(u) p(u) du, \quad (8)$$

where the normalization factor

$$d(0, t) = \int_{-\infty}^{\infty} (2\pi t^2)^{-\frac{1}{2}} e^{-\frac{u^2}{2t^2}} p(u) du \quad (9)$$

ensures the effective kernel weights sum to 1.

Linearizing the now combined function  $\alpha(s)p(s)$  as before yields

$$\begin{aligned} A(\alpha(0), t) &\approx \left[ p(0) + \ddot{p}(0) \frac{t^2}{2} \right]^{-1} \left[ p(0)\alpha(0) \right. \\ &\quad \left. + \frac{t^2}{2} [p(0)\ddot{\alpha}(0) + 2\dot{\alpha}(0)\dot{p}(0) + \alpha(0)\ddot{p}(0)] \right] \\ &\approx \alpha(0) + \frac{t^2}{2} \ddot{\alpha}(0) \\ &\quad + \frac{t^2}{2p(0)} \left[ 2\dot{\alpha}(0)\dot{p}(0) + \alpha(0)\ddot{p}(0) \right]. \end{aligned} \quad (10)$$

Comparing with (6), it can be seen that effect of the variation in sampling density  $p(s)$  is felt through the additional last term in (10) that corrupts the estimate of the curvature vector. A similar expression may be obtained for surfaces.

**Invariance to sampling distribution.** In this section, we show how to remove this additional term [8] by modifying the kernel function  $\phi$  through an estimate of the density  $p(s)$ . In particular, consider

$$\tilde{\phi}(s, u, t) = \frac{\phi(s, u, t)}{p_t(s)p_t(u)}, \quad (11)$$

where  $p_t(s)$  is the kernel density estimate at  $s$  with kernel bandwidth  $t$  as

$$p_t(s) = \int \phi(s, u, t) p(u) du. \quad (12)$$

Note that the above continuous domain expression for  $p_t(s)$  must be approximated in practice with finite samples as  $\sum_i \phi(s, u_i)$  where  $\mathbf{x}_i = \alpha(u_i)$ . We now show that the above modification of (11) eliminates the dependence on the sampling distribution.

We know from linearizing  $p(u)$  around  $s$  that

$$p_t(s) = \int (2\pi t^2)^{-\frac{1}{2}} e^{-\frac{(s-u)^2}{2t^2}} p(u) du \approx p(s) + \ddot{p}(s) \frac{t^2}{2}.$$

Therefore

$$\begin{aligned} &\int_{-\infty}^{\infty} \tilde{\phi}(s, u, t) \alpha(u) p(u) du \\ &\approx \frac{(2\pi t^2)^{-\frac{1}{2}}}{p_t(s)} \int_{-\infty}^{\infty} e^{-\frac{(s-u)^2}{2t^2}} \alpha(u) \left[ 1 - \frac{\ddot{p}(u) t^2}{p(u) 2} \right] du \\ &\approx \frac{1}{p_t(s)} \left[ \alpha(s) + \frac{t^2}{2} \ddot{\alpha}(s) - \frac{\ddot{p}(s) t^2}{p(s) 2} \right], \end{aligned} \quad (13)$$

and the normalizing factor transforms to

$$\tilde{d}(s, t) = \int_{-\infty}^{\infty} \tilde{\phi}(s, u, t) p(u) du \approx \frac{1}{p_t(s)} \left[ 1 - \frac{\ddot{p}(s) t^2}{p(s) 2} \right]. \quad (14)$$

Dividing (13) by (14) gives

$$\begin{aligned} \tilde{A}(\alpha(s), t) &= \frac{1}{\tilde{d}(s, t)} \int_{\Gamma} \tilde{\phi}(s, u, t) \alpha(u) du \\ &\approx \alpha(s) + \frac{t^2}{2} \ddot{\alpha}(s). \end{aligned} \quad (15)$$

Thus the use of the density normalized kernel  $\tilde{\phi}$  removes the dependence of the result on variations in sampling density  $p(s)$ . A similar result can be obtained for surfaces, with the  $\ddot{\alpha}(s)$  term replaced by the mean curvature normal  $H\mathbf{n}_p$ .

### 3. A scale-selection algorithm

In order to try and define what a *characteristic scale* at a point may be, it is useful to recall its analogy in 2D intensity images. A pixel location is considered salient or "interesting" by virtue of the distinctiveness of its intensity relative to that of its neighboring pixels. A change in the definition of this neighborhood can make the pixel seem less or more salient. Thus a point in a periodic intensity pattern of monotone frequency is most distinctive relative to a neighborhood of radius equal to its wavelength.

In our problem since the available information is of object shape, the distinctiveness at a point may be captured through the *variation in shape* at that point relative to its neighborhood. For example, on a perfect plane or sphere no point is salient with respect to its neighbors as the local shape is identical for any choice of neighborhood. Consider the addition of a small "bump" caused by perturbing the surface of a sphere. A point on the bump is now salient due its increased shape variation with respect to the sphere. However, relative to a neighborhood much larger than the spatial extent of the bump, the perturbation is not significant.

### 3.1. Scale-space extrema

We propose to capture this *locality* in shape variation by inspecting the variation in  $\tilde{A}(\mathbf{x}, t)$  as a function of the size of the neighborhood  $t$  used to estimate it. The relationship between the  $\tilde{A}(\mathbf{x}, t)$  operator and the mean curvature suggests a way to do this, and also provides a simple interpretation for the case of constant curvature surfaces.

We know from (7) and the derivation in previous section with the density normalized kernel that

$$\|\mathbf{x} - \tilde{A}(\mathbf{x}, t)\| = H \frac{t^2}{2} \sqrt{\mathbf{n}_x^T \mathbf{n}_x} = H \frac{t^2}{2} \quad (16)$$

or that the norm of the shift induced by the  $\tilde{A}$  operator is proportional to the mean curvature.

Consider the scalar function  $F$  formed by exponential damping of the above expression

$$F(\mathbf{x}, t) = \frac{2\|\mathbf{x} - \tilde{A}(\mathbf{x}, t)\|}{t} e^{-\frac{2\|\mathbf{x} - \tilde{A}(\mathbf{x}, t)\|}{t}}. \quad (17)$$

Differentiating (17) with respect to the scale parameter  $t$  and using (16) gives

$$\frac{\partial F(\mathbf{x}, t)}{\partial t} = \frac{\partial}{\partial t} (Hte^{-Ht}) = He^{-Ht} (1 - Ht)$$

which achieves a maximum at  $t_{\max} = \frac{1}{H}$  for  $H \neq 0$ .

Thus the characteristic scale may be defined in the case of a perfect sphere or plane simply as its radius of curvature, in direct analogy to the wavelength for monotone intensity patterns in images. In the case where the shape is not of constant mean curvature,  $t_{\max}$  loses its simple interpretation, as is true of characteristic scale in images for non-monotone frequencies [10]. In the previous example of the perturbation on a sphere, the local shape at a point is a composition of two constant curvature surfaces, and there will exist two characteristic scales reflecting the two components.

### 3.2. Implementation

Algorithm 1 outlines the procedure based on the extrema property of (17). The set of scales  $\{t_k\}$  were fixed as  $t_k = t_0(1.6)^k$  where  $t_0$  is a base scale. We draw attention to a few implementation details below.

**Graph construction.** (Step 1) Since the integral operator  $A$  requires knowledge of geodesic distances between each point pair, we require a way to compute them from the few observed points. A common strategy is to construct a sparse Euclidean graph on the points and approximate the geodesic distances by distance along the shortest path in the graph. We choose to use disjoint minimum spanning trees (DMST) [2], although other valid constructions include  $\epsilon$ -graphs and mutual  $k$ -nearest neighbor graphs. The construction using DMSTs has some desirable properties over traditional  $k$ -nearest neighbor or  $\epsilon$ -ball schemes. In practice, it produces connected graphs without undesirable gaps

---

#### Algorithm 1 Scale selection algorithm

---

**Data:** Points  $X = \{\mathbf{x}_i\} \in \mathbb{R}^3$  with  $i = 1 \dots n$ , and a set of scales  $T = \{t_k\}$  to be considered.

1: Construct a graph  $\mathbb{G}$  on the points from which approximate geodesic distances may be computed as graph path distances. Distance  $d_{\mathbb{G}}(x_i, x_j)$  between any pair of points  $\mathbf{x}_i, \mathbf{x}_j$  can be computed efficiently using Dijkstra's algorithm.

2: **for**  $t \in \{t_k\}$  **do**

3:   **for**  $x \in \{\mathbf{x}_i\}$  **do**

4:     Compute estimate of density

$$p_t(\mathbf{x}_i) = \sum_j \phi(\mathbf{x}_i, \mathbf{x}_j, t)$$

where  $\phi(\mathbf{x}_i, \mathbf{x}_j, t) = (2\pi t^2)^{-\frac{1}{2}} \exp\left(-\frac{d_{\mathbb{G}}^2(\mathbf{x}_i, \mathbf{x}_j)}{2t^2}\right)$

5:     Compute weights for each pair  $(i, j)$  as

$$\tilde{\phi}(\mathbf{x}_i, \mathbf{x}_j, t) = \phi(\mathbf{x}_i, \mathbf{x}_j, t) / [p_t(\mathbf{x}_i)p_t(\mathbf{x}_j)]$$

6:     Evaluate the operator  $\tilde{A}$

$$\tilde{A}(\mathbf{x}_i, t) = \frac{\sum_j \tilde{\phi}(\mathbf{x}_i, \mathbf{x}_j, t)\mathbf{x}_j}{\sum_j \tilde{\phi}(\mathbf{x}_i, \mathbf{x}_j, t)}$$

7:     Compute the invariant  $F$

$$F(\mathbf{x}_i, t) = \frac{2\|\tilde{A}(\mathbf{x}_i, t) - \mathbf{x}_i\|}{t} \exp\left(-\frac{2\|\tilde{A}(\mathbf{x}_i, t) - \mathbf{x}_i\|}{t}\right)$$

8:    **end for**

9:    Declare interest points to be those having extremum values of  $F(\mathbf{x}_i, t_k)$  both in a geodesic neighborhood of radius  $t_k$  as well as over a range of scales  $(t_{k-1}, t_{k+1})$ .

10:    Designate points in the geodesic  $t_k$ -neighborhood of each interest point as forming its interest region at that scale.

11: **end for**

---

and does not induce edges to clump together in noisy regions having relatively higher point density.

**Region extraction** (Step 10) for an extremum detected at scale  $t$  is currently done by grouping together points that are within a graph distance proportional to  $t$  from the extremum point. Note this is analogous to the use of a multiple of the scale  $\sigma$  as the patch radius in 2D images [13].

## 4. Experiments

In this section we present supporting experimental results to demonstrate the utility of the proposed method. Since region selection is done separately from the step of constructing shape descriptors, we focus only on perfor-

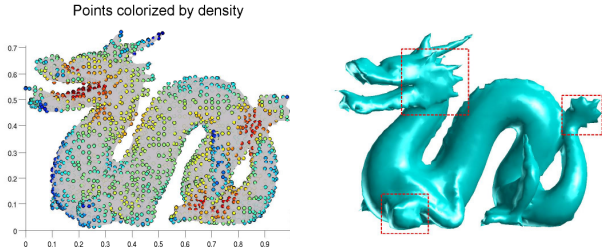


Figure 2. (left) Colored plot showing variation of point density on the dragon model colored in gray. (right) Rendering of underlying mesh. Results in boxed regions are analyzed in Figure 3.

mance metrics that are *independent* of the choice of descriptor. Some of the datasets presented in this section were obtained from 3D models that were available in the form of a triangulated connectivity mesh. We wish to emphasize here that the algorithm does *not* have any knowledge of the mesh during runtime, and the mesh only serves as a visual aid for presenting our results.

**Qualitative behavior:** To illustrate the behavior of the multi-scale operator, we focus on its application to a low-resolution version of the ‘dragon’ model, originally from the Stanford Scanning Repository. One important characteristic of this model is its complex shape with both finer spike-like features on its limbs, tail and back, as well as coarser features such as the stubby feet and tail. Figure 2 shows the 5205 points on the model colored by point density, and illustrates the non-uniformity of the samples.

Figure 4 shows the norm of the density normalized Laplacian  $\|\mathbf{x}_i - \tilde{A}(\mathbf{x}_i, t)\|$  for a few choices of scale  $t$ . The regions associated with their scale-space extrema for the corresponding computed value of  $F$  are shown in Figure 5. For each scale-space extremum in Figure 5, the mesh faces formed by points belonging to its interest region are given the same random color. Note that for the purpose of comparison with Figures 4 all the associated interest regions in Figure 5 are shown without the use of a threshold. A suitable choice of threshold may be used in practice to remove the least salient detections.

For the smallest values of  $t$  considered, the extrema correspond largely to noise due to discretization. When the value of  $t$  is increased, it can be seen (Figure 5) that features reflecting the geometry of the model become visible, such as the spikes on the feet and tail-end, and the features on the claws and mouth are progressively detected as hoped. Figure 3 shows zoomed views of some interesting parts of the model where complex shapes exist in close proximity and are correctly detected at different scales. For example, the first row shows the tail having fine short spikes at its edges that are detected for small values of  $t$ , while the coarser overall club-like shape is detected at a larger scale.

Figure 6 shows the results from processing a sparse outdoor scan of vehicles in a parking lot. While the geometric

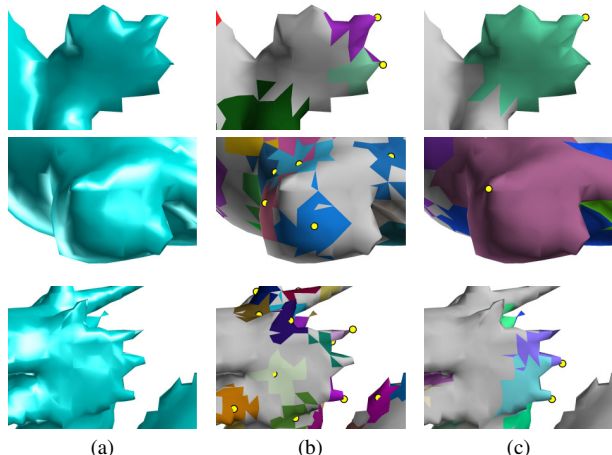


Figure 3. Interest regions detected in parts of the ‘dragon’ model from Figure 2 rendered in the column (a). Columns (b) and (c) show colored patches for each detected region corresponding to finer and coarser shapes, respectively. Yellow dots mark extrema locations. Figures are best seen in color.

interpretation of the results is not as vivid as the previous example, the detected intermediate regions correctly segment points on the vehicle into front and side components based on the difference in local curvature of those regions.

**Repeatability:** To quantitatively measure the robustness of the algorithm to noisy data and sampling density, we computed the repeatability of the detections for low-resolution versions of the ‘bunny’ and ‘dragon’ models from the Stanford database. The data was controllably perturbed by adding Gaussian random noise with multiple values of standard deviation  $\sigma$  to points sampled with replacement from the models.

We define the overlap score between two interest regions as the ratio of the intersection to the union of the two smallest 3D spheres containing the points corresponding to those regions. Note that this metric for computing overlap is the direct 3D analog of the metric that is well accepted as the benchmark for 2D interest region detection in images [13]. Each detected interest region in the noise-free reference model is matched with a region in the noisy, resampled test model as the region with which it has the maximum overlap score. We then analyze the repeatability of the detection by computing the average of the overlap score between matched regions.

Figure 7 plots the average overlap of matched interest regions for different noise levels  $\{\sigma\}$ . The x-axis is the scale level number  $k$  indexing  $\{t_k\}$ , the set of scales considered. Two observations are noteworthy. First, as can be expected, the overall repeatability score decreases with increasing noise level  $\sigma$ . Second, the repeatability of finer scale features (low scale level numbers) is consistently lower than that of coarser large-scale features. This should also not be surprising, as noise in point positions is indistinguish-



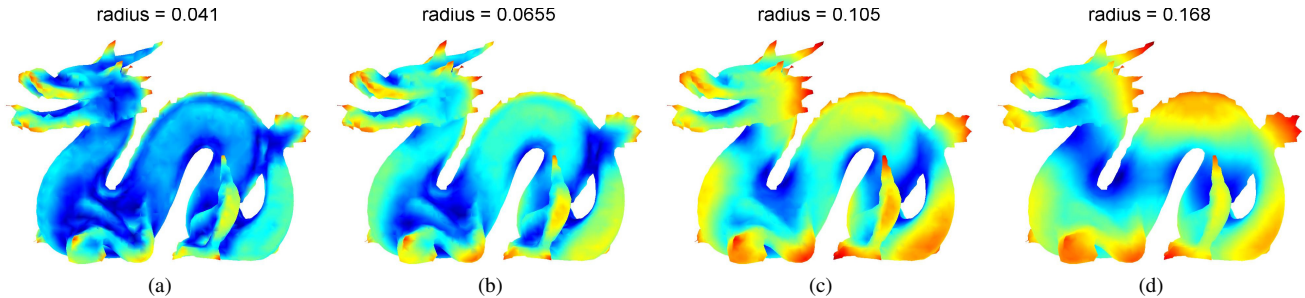


Figure 4. Plot of norm of density-normalized Laplacian operator on the ‘dragon’ model for increasing kernel widths. Plots use the ‘jet’ colormap. Note that knowledge of the underlying mesh was not used in the computation.

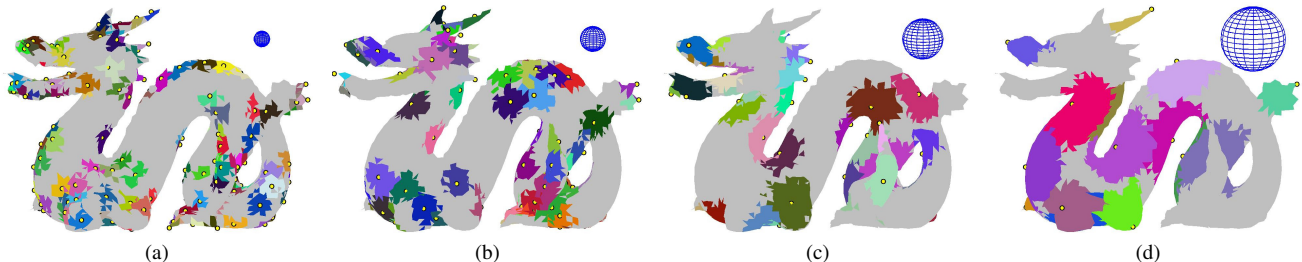


Figure 5. Plot showing extrema and corresponding sizes for increasing kernel widths, corresponding to  $F$  values computed from Figure 4. Each colored patch on the model corresponds to the region associated with a extremum point detected at that scale. Yellow dots mark extrema locations. The blue spheres are drawn with radii equal to the value of kernel width used to detect the regions in the image.

able from high-frequency shape details. Thus addition of Gaussian noise should adversely affect the repeatability of a geometric feature whose spatial extent is comparable to the standard deviation of the noise distribution.

When interpreting the repeatability score values, it should be noted that the overlap metric is far more stringent when used with 3D points than with 2D images for two reasons. First, due to the difference in dimensionality, an overlap score of 60%, for example, corresponds to a difference in radius of only 15% for two 3D spheres centered at the exact same location while it corresponds to a 23% difference in circular patch radius in 2D. It has been verified in the 2D image domain [13] that an overlap error of 50% can be handled with a sufficiently robust descriptor. This is equivalent to an error of 65% in 3D, or an overlap score of just 35%. Second, the ability to find extrema at precisely the same locations in 3D point samples of the same shape is severely limited by several factors such as point density and noise. Thus, for the 3D point cloud domain, a low repeatability score is not as much a negative indicator, when compared to 2D images, of the ability to match descriptors computed in those regions.

**Redundancy:** One drawback of using fixed-scale quantities such as surface variation [15] to find interest regions is that, although the measure may correlate with change in surface geometry, it is unclear how to relate the computed values across different neighborhood sizes. Detectors based on these measures tend to select regions centered at the same

points for multiple scales. Thus, the naive approach of simply choosing every interest region detected for a range of preset scales usually results in regions that have high degree of overlap, and thus a high degree of redundancy in the representation.

We quantify the redundancy in a given set of detected multi-scale interest regions by computing the average overlap between any pair of regions in the set. Overlap was computed in the same manner as in the repeatability experiment, and the comparison was made between a detector based on the surface variation score [15] (defined in Section 1.1) and our proposed invariant  $F(\mathbf{x}, t)$ . A lower overlap score is an indicator of higher variation in the detected interest regions and thus lower redundancy.

Using our proposed method, the redundancy score for the ‘dragon’ model reduced from 0.166 to 0.114, and reduced from 0.165 to 0.152 for the ‘bunny’ model, corresponding to difference of 32% and 8% respectively. Although this difference is dataset-dependent, we have observed this lowering of redundancy score to be consistent across datasets and noise levels. At the same time, our proposed method also gives a larger number of interest regions. Detailed results are omitted due to space limitations.

## 5. Discussion

This work presented a filtering operator that works directly in the input point cloud domain to generate multi-scale representations that reflect the underlying geometry of

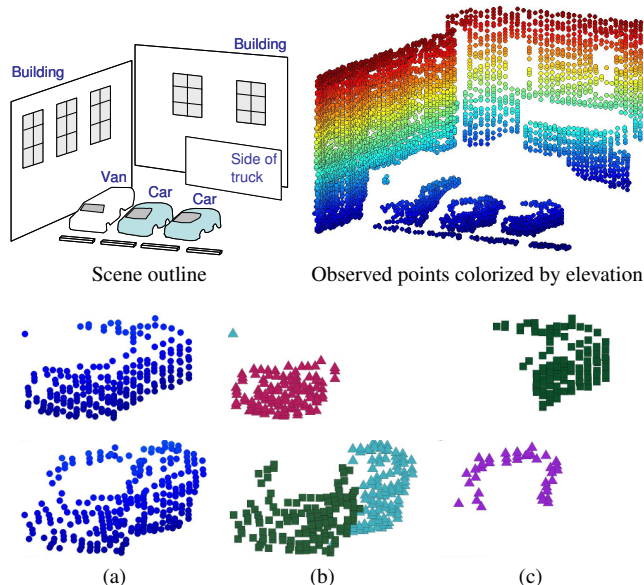


Figure 6. Top figure plots points from a sparse 3D scan of vehicles in a parking lot. Column (a) shows magnified view of the points on the two cars highlighted in the scene outline. Columns (b) and (c) plot points colored by interest regions detected at different scales for the input points in the corresponding row.

the data. Our approach has deviated from traditional multi-scale analysis in that we construct an operator  $\hat{A}$  that does *not* have the property of being a semi-group. The impact of this is largely reflected as increased computational cost. One way to scale the proposed algorithm efficiently to large datasets may be through use of an appropriate quadrature rule to evaluate the integrals in  $\hat{A}(\mathbf{p}, t)$ . This could be done so that a far fewer number of points would need to be evaluated at the expense of a small loss in numerical accuracy.

This approach also leaves open the possibility of richer classes of operators that could be obtained by modifying the kernel  $\phi$  or using its higher order derivatives. A more exhaustive characterization of the possible choices could lead to a useful larger family of interest region detectors and deserves further study.

## References

- [1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4), 2002.
- [2] M. A. Carreira-Perpinán and R. S. Zemel. Proximity graphs for clustering and manifold learning. In *Neural Information Processing Systems (NIPS)*, pages 225–232, 2004.
- [3] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
- [4] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik. Recognizing objects in range data using regional point descriptors. In *Proc. European Conf. on Computer Vision (ECCV)*, May 2004.

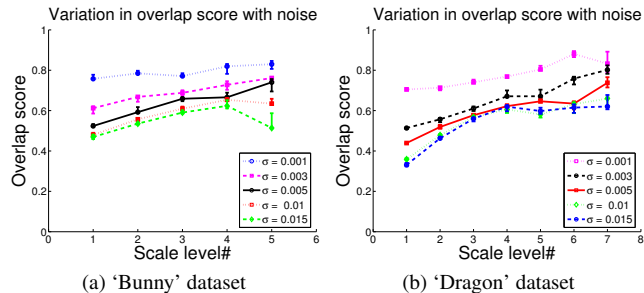


Figure 7. Variation in average overlap score of matched interest regions for various scales across increasing score noise levels. Error-bars show upper and lower quartiles from 10 trials for each standard deviation ( $\sigma$ ).

- [5] T. Iijima. Basic theory of pattern observation. In *Papers of Technical Group on Automata and Automatic Control, IECE*, 1959.
- [6] A. Johnson and M. Hebert. Using spin images for efficient object recognition in 3D scenes. *PAMI*, 21(5), 1999.
- [7] J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.
- [8] S. Lafon. *Diffusion maps and geometric harmonics*. PhD thesis, Yale University, 2004.
- [9] X. Li and I. Guskov. Multi-scale features for approximate alignment of point-based surfaces. In *Symposium on Geometry Processing (SGP)*, 2005.
- [10] T. Lindeberg. Feature detection with automatic scale selection. *Intl. Journal of Computer Vision (IJCV)*, 30(2):77–116, 1998.
- [11] D. G. Lowe. Distinctive image features from scale-invariant viewpoints. *Intl. Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.
- [12] B. Matei, Y. Shan, H. Sawhney, Y. Tan, R. Kumar, D. Huber, and M. Hebert. Rapid object indexing using locality sensitive hashing and joint 3D-signature space estimation. *PAMI*, 28(7):1111–1126, July 2006.
- [13] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *Intl. Journal of Computer Vision (IJCV)*, 65(1/2):43–72, 2005.
- [14] J. Novatnack and K. Nishino. Scale-dependent 3D geometric features. In *Proc. Intl. Conf. on Computer Vision (ICCV)*, 2007.
- [15] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces. In *Eurographics*, 2003.
- [16] R. C. Reilly. Mean curvature, the laplacian, and soap bubbles. *The American Mathematical Monthly*, 89(3), 1982.
- [17] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH*, 1995.
- [18] 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.
- [19] A. Witkin. Scale filtering. In *IJCAI*, volume 2, pages 1019–1022, 1983.