International Journal of Computational Geometry & Applications

Scale Selection for Geometric Fitting in Noisy Point Clouds

Ranjith Unnikrishnan
Jean-François Lalonde
Nicolas Vandapel
Martial Hebert

Robotics Institute, Carnegie Mellon University,
Pittsburgh, PA 15213, USA
{*ranjith, jlalonde, vandapel, hebert*}*@cs.cmu.edu*

Received (received date)
Revised (May 13, 2010)
Communicated by (Name)

ABSTRACT

In recent years, there has been a resurgence in the use of raw point cloud data as the geometric primitive of choice for several modeling tasks such as rendering, editing and compression. Algorithms using this representation often require reliable additional information such as the curve tangent or surface normal at each point. Estimation of these quantities requires the selection of an appropriate scale of analysis to accommodate sensor noise, density variation and sparsity in the data. To this goal, we present a new class of *locally semi-parametric* estimators that allows analysis of accuracy with finite samples, as well as explicitly addresses the problem of selecting optimal support volume for local fitting. Experiments on synthetic and real data validate the behavior predicted by the model, and show competitive performance and improved stability over leading alternatives that require a preset scale.

Keywords: scale selection, point cloud processing, tangent estimation, normal estimation, perturbation analysis, semi-parametric estimator

1. Introduction

With advances in sensor technology, it is now feasible to acquire detailed scans of complex scenes with millions of data points at high sampling rates. This possibility brings with it the question of how best to process such large amounts of point data to extract meaningful information such as the underlying shape of the scene being scanned.

In the past, approaches to process such point-sampled data consisted of using an intermediate representation such as a 2D range-image or a triangulated mesh constructed from the input data. Range-images allow easy, though not always appropriate, substitution of operators from traditional image processing to the domain of 3D point processing. However, because the sensor geometry of laser range scan-

ners need not confirm with a regular lattice structure of an image, the construction of range-images usually involves some loss of information. It is also not easy to combine information from multiple range-images of the same scene without reverting to the original 3D input space. Meshes are piece-wise approximations of the underlying surface and mesh processing has been the subject of much study. The construction of a mesh from noisy data, however, is not straightforward and requires denoising and additional pre-processing that can undesirably remove geometric detail.

In contrast, working directly with point clouds in the input space offers several advantages. Point clouds are a natural sensor output and there is no assumption of available connectivity information. It is also better suited for dynamic applications requiring data addition and deformation. Recent years have also seen a revival amongst the graphics community in the use point clouds directly as a rendering primitive, as it circumvents the need for error-prone mesh construction procedures.

Most applications of point cloud processing require some additional knowledge of the underlying shape represented by the point samples. Rendering requires knowledge of surface normals at each point for visibility and lighting computation. Some shape-compression algorithms utilize estimates of tangents to curves as predictors for shape-outline encoding and iso-contour compression schemes in triangle-meshes [22]. For solving the path planning problem in mobile robot navigation, there is a frequent need to evaluate the traversability of terrain by reconstructing its shape from observed sparse laser data. Accuracy in the reconstruction is crucial in order to reliably determine *a priori* whether the vehicle will make all-wheel contact with the ground at each point of a candidate trajectory.

All the above applications require fitting a surface of some form to observed data. Due to the nature of sensing modalities, some immediate concerns arise from the above applications such as data sparseness, irregularity in sampling and range-dependent noise. The subject of this paper is a mathematically sound approach to geometric fitting that addresses these challenges with finite-sample guarantees of accuracy.

In what follows, we describe an approach to local geometric fitting that enjoys the benefits of both finite-sample error analysis as well as asymptotic efficiency. Section 2 presents related work. Section 3 formulates the problem and makes the case for the locally semi-parametric approach employed in the remainder of the paper. We point out that Section 3 presents a generalization of our previous work in [36] whose analysis was restricted to 2D and 3D curves. Section 4 and Section 5 will then detail the application of the approach to the analysis of points lying on curves and its extension to surfaces respectively. We then present results in Section 6 to validate the behavior predicted by the model on finite real data and demonstrate its accuracy and stability with some applications. We then conclude in Section 7 with discussion and some directions for future work.

2. Related Work

There are several approaches to curve and surface fitting, both non-parametric (tensor voting [32], radial basis functions, etc.) and parametric [22] (moving least-squares approximations [19], implicit parabolic fitting, b-splines, etc.). This section will outline some of the popular approaches in the literature.

There are many approaches to surface and curve reconstruction motivated by techniques from computational geometry. These include algorithms based on Delaunay triangulations [3,6], such as the crust algorithm [2], the cocone algorithm [4] and its extension called tight cocone [9], and algorithms based on alpha shapes [11, 12]. As summarized in [13], the more successful approaches are based on the construction of Delaunay triangulations. Under the somewhat restrictive assumption of a closed bounded shape, the problem may be transformed into one of filtering the Delaunay tetrahedra whose union approximates the shape interior. The different approaches promise differing extents of theoretical guarantees on the reconstructed shape varying with assumptions on sampling density and smoothness. However, the fact that the reconstruction in these methods can only interpolate through the observed points affects the quality of their results with noisy and sparse data.

Most practical curve and surface reconstruction algorithms are based on local polynomial fitting and its variants. Recent work by Lewiner *et al.* [22] computed the coefficients of an arc-length parameterized third-order approximation to a curve by solving a weighted least-squares problem at each point using only the points in its local neighborhood. The implicit parameter in the algorithm was the considered neighborhood radius, which was preset by fixing the number of neighbors considered at each point. A similar neighborhood selection strategy was used by Cazals *et al.* [7] who fit the local representation of a manifold using coefficients of a truncated Taylor expansion, termed a jet. Hoppe *et al.* [14] and others [42] compute normals to a surface at each point by fitting a plane to its k-nearest neighbors. The success of these algorithms depends crucially on the chosen value of k, and there is little guidance in the literature on how to make that choice.

In the computer vision community, much work has been done on geometric reconstruction using non-parametric tensor voting [32, 33]. A key step of this is a voting procedure used to aggregate local information at each point or voxel of interest. The vote is in the form of a $d \times d$ tensor, where d is the data dimensionality, indicating preferred direction of normal (or tangent), and the eigen decomposition of the aggregate tensor at a point gives the desired result. Again, a crucial parameter is the choice of the size of the support region for vote collection, usually chosen heuristically. Work in [33] proposed a fine-to-coarse approach in which points likely to form curves are linked together at fine scale to form fragments, and then linked together incrementally as the scale is increased using a heuristic inspired by perceptual grouping. Work in this paper focuses on sparser point sets than used in [33] and thus requires guarantees on the small sample behavior of the choice of estimator.

Closely related theoretical work by Mitra *et al.* [26] addresses the choice of opti-

mal neighborhood size for normal estimation in surfaces using PCA. They derived a bound on the angular error between the estimated normal and true normal, and proposed the optimal radius as the value that minimized that bound. An iterative procedure was suggested that first estimated the local density and curvature, then computed the optimal radius for those values, and repeated the procedure until convergence. However, the obtained closed-form expression had two parameters that relied on knowledge of the observed data distribution and had to be fixed *a priori*. Furthermore, it was unclear whether the behavior predicted by the obtained closed-form expression agreed with real data for finite samples.

In the Computer Graphics community, a large body is dedicated to geometric fitting of noisy point cloud, especially based on Least Square techniques pioneered by Levin [1, 20, 21]. The work of [40] addresses the problem of scale selection in the context of surface reconstruction. The geometric fitting is also cast into a statistical framework in the recent work of [16, 28].

In the machine learning literature, there has been renewed interest in the use of max-margin methods as well as Gaussian Processes (GP) for solving non-linear non-parametric regression problems [23,29,38,39]. Recent work in [30] demonstrated how algorithms based on these models can be implemented efficiently and scaled to large datasets. However their application to surface reconstruction by global fitting of an implicit function requires specifying additional constraints to avoid a degenerate solution. Currently, these constraints have to be specified by adding off-surface points, generated by projecting along an estimate of the surface normal, and specifying function values at those points. More generally, it is unclear how the specification of these locations and values at off-manifold points affects the accuracy of the reconstruction.

3. Approach

In this section, we formulate the geometric fitting problem and develop our solution to it in a manner that satisfies the requirements of our domain. In the process of doing so, we will argue that traditional estimators from classical statistics are insufficient to deal with point sample data, and that both new estimators as well new methods for evaluating these estimators are necessary. Table 1 presents the mathematical notations used throughout this document, grouped by the sections in which they first appear.

3.1. *Overview*

Before presenting our proposed solution to the geometric fitting problem, we briefly motivate our approach. Our goal is two-fold. First, we wish to estimate the parameters of a geometric model that best fit our observed data. Second, we wish to simultaneously obtain some guarantee of the accuracy of our solution that is valid for the sparse datasets we may expect to work with.

Notation	Meaning
$\mathcal{M}$	Unknown manifold
$\mathbf{x}_i^o$	Point lying on $\mathcal{M}$ $(i = 1 \dots n)$
$\mathbf{x}_i$	Noisy observation of $\mathbf{x}_i^o$ with $\mathbf{x}_i = (x_i, y_i, z_i)$
η_i	Noise associated with observation $\mathbf{x}_i$ with $\eta_i = (\eta_{x,i}, \eta_{y,i}\eta_{z,i})$
Λ_i	Covariance matrix corresponding to noise η_i
θ	Unknown model parameters
$f(\mathbf{x}, \theta)$	Function of position $\mathbf{x}$ whose zero-level set represents a surface with model parameters θ
$\gamma(\mathbf{x})$	Problem-specific map from coordinates $\mathbf{x}$ to a higher dimensional vector
$\mathcal{N}(\mathbf{x}_i, r_i)$	Neighborhood around point $\mathbf{x}_i$ defined by radius r_i
$\mathbf{s}_i$	Intrinsic coordinates of point $\mathbf{x}_i^o$ on $\mathcal{M}$
$\bar{\theta}_n$	Estimate of θ from n observations
κ $(\dot{\kappa})$	Curvature (derivative) of line or normal curve (at origin of interest)
τ $(\dot{\tau})$	Torsion (derivative) of line (at point of interest)
r	Radius of neighborhood (around point of interest) considered for geometric fitting
σ_0	Std. deviation of noise in each coordinate
X	Random variable for the x-coordinate. Y and Z are defined similarly.
$\bar{X}_n$	Estimate of $\mathbb{E}(X)$ from n samples of X.
μ_X	$= \mathbb{E}(X)$, mean of distribution of random variable X
$d_n(X)$	$= \mathbb{E}(X - \mu_X)^n$, capturing centered statistical dispersion of random variable X
$c_m(X, Y)$	$= \mathbb{E}[(X - \mu_X)(Y - \mu_Y)]^m$, capturing generalized covariance of two random variables X and Y.
$\hat{M}_n$	Estimate of the scatter (covariance) matrix from the n points in a local neighborhood.
$\bar{M}$	$= \mathbb{E}(\hat{M}_n)$, the expected value of the scatter matrix
Q	Perturbation matrix that deviates estimate of tangent (normal) away from its true value
δ	Spectral gap of $\hat{M}$
$B(r)$	Error bound in estimate of tangent (normal)
Π_α	Normal plane at angle α to some reference vector in the tangent plane, and containing the normal at that point.
κ_1, κ_2	Principal curvatures at point of interest. $\kappa_1 > \kappa_2$

Table 1. Mathematical notation used throughout this document, grouped by the sections in which they first appear.

To pursue the first objective, we make a design choice of modeling the scene as a combination of local compact regions, each represented by an implicit function having a small number of parameters. This design choice offers the flexibility of modeling scenes that may otherwise be too complex for a function to fit globally. However, this comes at the cost of requiring a principled method to automatically choose the neighborhoods.

One way to go about the second objective of deriving finite-sample guarantees is to first pursue the intermediate goal of deriving asymptotic guarantees that are valid when the number of data samples increases to infinity. Through analyzing this hypothetical scenario, we may hope to relate the estimation error with infinite data to the practical case of finite data. The framework of regression from classical statistics offers several tools for deriving these asymptotic guarantees. Hence, we approach the geometric fitting problem by first mapping it to the regression problem and deriving the asymptotic error for our chosen estimator.

To fold both the above objectives into one approach, we make an assumption on how points in a small neighborhood are spatially distributed. This step introduces the neighborhood size as another variable in the system along with the other unknown variables. By then measuring the deviation of the solution obtained with finite data from the ideal asymptotic data case, we obtain an *error bound* that involves both the unknown parameters encoding the underlying geometry as well as the neighborhood size. The best estimate of the model parameters may then be obtained by simply minimizing this error bound for both sets of unknown variables.

3.2. *Formulation*

Our starting point will be the set of available point samples $\{\mathbf{x}_i\} \in \mathbb{R}^d$. The points are assumed to be noisy observations of an unknown underlying curve or surface and follow the noise model

$$\mathbf{x}_i = \mathbf{x}_i^o + \eta_i \qquad \text{where} \quad \eta_i \sim N(0, \Lambda_i), \tag{1}$$

with η_i denoting heteroscedastic (or point-dependent) zero-mean Gaussian noise with variance Λ_i. The points $\mathbf{x}_i^o$ represent the unknown true points lying on the manifold. Throughout the paper, we will assume that the manifold (curve or surface) under study is smooth, and that the noise variance Λ_i is available through an error model of the sensor used to acquire the points.

One way to represent the underlying manifold mathematically is through a parameterized implicit equation

$$f(\mathbf{x}_i^o; \theta) = 0 \qquad \forall i, \tag{2}$$

where θ represent the unknown model parameters. In particular, we will be interested in the bilinear form $\gamma(\mathbf{x}_i^o)^{\mathrm{T}}\theta = 0$ where γ denotes a problem-specific vector map.

3.3. *Local versus Global Fitting*

At this stage, the problem specification is identical to several others in multi-view geometry. For instance, in the problem of estimating the 3×3 fundamental matrix F, the observed point pairs (x, y) and (x', y') are required to satisfy the epipolar constraint $\begin{bmatrix} x & y & 1 \end{bmatrix}^{\mathrm{T}} F \begin{bmatrix} x' & y' & 1 \end{bmatrix}$. By expanding the constraint in terms of the entries of F, its equivalent bilinear form may be obtained as $\gamma(x, y, x', y') = \begin{bmatrix} xx' & xy' & x & yx' & yy' & y & x' & y' & 1 \end{bmatrix}^{\mathrm{T}}$, with model parameters θ representing the 9-vector of F-matrix coefficients. Problems of this type have been studied by several researchers [15, 24, 34], often by the name of the non-linear errors-in-variables model.

However, there is one very important difference. In problems such as fundamental matrix estimation, all the observations satisfy a single *global* constraint equation as a natural consequence of epipolar geometry. However, in the domain of geometric fitting it is unreasonable to expect that a single equation to capture the geometric complexity of an arbitrary scene. Conversely, such a model would potentially require far too many parameters, perhaps exceeding the number of available data points, thus making the fitting problem ill-posed.

A more tractable approach is that of *local* geometric fitting, in which the chosen geometric model is assumed sufficient to explain a subset of observed points in a small neighborhood $\mathcal{N}(\mathbf{x}_i, r_i)$ of each point $\mathbf{x}_i$ where the r_i is the radius defining the neighborhood. Thus the implicit equation models the surface *locally* at *each* point $\mathbf{x}_i$ as

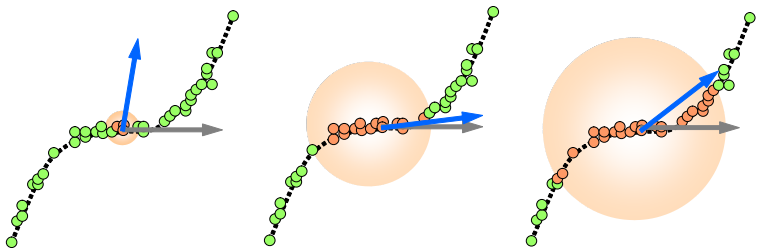
$$f(\mathbf{x}_j^o, \theta_i) = 0 \qquad \text{where} \quad \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i, r_i), \tag{3a}$$

$$\text{and} \qquad \mathbf{x}_j = \mathbf{x}_j^o + \eta_j \qquad \text{where} \quad \eta_j \sim N(0, \Lambda_j) \tag{3b}$$

Note that j indexes points in the neighborhood $\mathcal{N}(\mathbf{x}_i, r_i)$, and that there is now one model equation for *each* point $\mathbf{x}_i$ that is satisfied only by points in its neighborhood $\mathcal{N}(\mathbf{x}_i, r_i)$. The subscript i in θ_i emphasizes that the model parameters may be different at each point $\mathbf{x}_i$.

Several useful signal processing tasks may be positioned in the above framework of local geometric fitting. The task of **denoising** a point cloud may be accomplished by simply projecting each observed point $\mathbf{x}_i$ to the zero-level set of its locally fit surface $f(\mathbf{x}, \theta_i)$. The task of **surface reconstruction** can be done in a two-step process. First, divide the input space into a regular grid and perform the geometric fitting procedure in a neighborhood centered around each grid point. The fitted function evaluated at the grid location gives the unsigned distance of the point to the surface. Performing this at each grid point gives an unsigned distance map, whose zero-level isosurface may be extracted using, say, a marching-cubes algorithm, to give the underlying surface.

Given the functional form of f, the remaining problems to be addressed are: (A) **scale selection**, or how to choose a neighborhood $\mathcal{N}(\mathbf{x}_i, r_i)$ at each point, and (B) **parameter estimation** how to compute the model parameters θ_i for that neighborhood. Traditionally these two problems have been addressed separately in



(a) $r \ll r_{opt}$ (b) $r = r_{opt}$ (c) $r \gg r_{opt}$

Fig. 1. Effect of varying neighborhood radius (r) considered for computing tangent at a point on a sampled curve with respect to the optimal radius (r_{opt}). Estimated and true tangents are shown by the blue and grey arrows respectively

the literature.

The first problem of scale selection is usually addressed by arbitrarily fixing the neighborhood size or the value of k in k-nearest neighborhood at each point [14, 22, 42], perhaps guided by some knowledge of the extent of the scene. This however, can have undesirable consequences as illustrated in Figure 1 for the case of tangent estimation (or equivalently, local line fitting) in a 2D curve. Using too small a radius can compromise the quality of the estimate due to the use of smaller number of noisy data points, while using too large a radius can permit a potentially dissimilar points in the neighborhood to adversely influence the estimate. Thus, it is crucial to make a choice of scale in model fitting that reflects the underlying geometry.

We will show later how the above two problems of model parameter estimation and scale selection may be solved *jointly*. For now we shall focus on getting good answers to the parameter estimation problem.

3.4. *Reduction to statistical inference*

In this section, we compare and draw connections between our problem of geometric fitting and the regression problem from classical statistics. This is done for two reasons - first, by mapping our problem into another that is well-studied, we hope to leverage solutions for the latter to be able to compute the model parameters θ_i at each point. Secondly, because our revised problem formulation involves with local neighborhoods having potentially a small number of points, we require a way of evaluating the accuracy of our solution within the established framework of statistical inference.

In what follows, we will drop the subscript i on θ_i and r_i with the understanding that these two parameters depend on a neighborhood around $\mathbf{x}_i$ and that this point of interest is fixed.

A comparison of the geometric fitting equations (3) and the standard regression

equation of the form

$$y_i = \beta x_i + \eta_i, \tag{4}$$

with model parameter β reveals some crucial differences. First, the model parameters θ in the geometric fitting problem satisfy an implicit equation through $f(\mathbf{x}, \theta) = 0$, whereas the relationship is explicit in the regression problem. Second, unlike in standard regression, there is no distinction into abscissa and ordinate variables in the fitting problem. Finally, the observation errors occur in all variables unlike in regression where the observation errors are assumed to exist only in the ordinate.

The first step in converting the geometric fitting problem into the classical regression framework is to convert the implicit equation (3a) into an *explicit* equation. This can be done through the introduction of local coordinate system as

$$\mathbf{x}_j^o = \mathbf{x}_j^o(\mathbf{s}_j, \theta), \tag{5}$$

where $\mathbf{s}_j$ denote unknown intrinsic coordinates on the surface $\mathcal{M}$. Note that the $\mathbf{s}_j$'s have the (lower) dimensionality of the manifold while the $\mathbf{x}_j$'s have the dimensionality of the input space.

The effect of introducing surface coordinates $\mathbf{s}_j$ is to convert the system of equations (3) into the explicit system

$$\mathbf{x}_j = \mathbf{x}_j^o(\mathbf{s}_j, \theta) + \eta_j, \tag{6}$$

The equation (6) now has the same functional form as the standard regression problem (4) but with the addition of nuisance parameters $\mathbf{s}_j$. Thus, the unknowns in the system are $\{\theta, \mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$ which exceed the number of equations.

Does it matter that the system of equations is under-determined? After all, techniques such as Expectation-Maximization (EM) are routinely used to solve systems through the introduction of nuisance parameters. As shall be shown, the problem arises not in solving the above equations, but instead in evaluating the *accuracy* of the obtained solution.

A well-accepted paradigm to evaluate the accuracy of an estimator $\bar{\theta}_n$ of θ from n observations is through its asymptotic efficiency. One way of evaluating this is through the question: does $\bar{\theta}_n \to \theta$ as $n \to \infty$? This paradigm of asymptotic behavior is a concept central to classical statistical reasoning. Can this paradigm be applied to a solution to the geometric fitting problem?

Unfortunately, classical asymptotic analysis cannot be directly applied to the geometric fitting problem, at least in the form that we have presented so far, for two reasons. The first reason, as also pointed out by Kanatani [15], is that classical asymptotic analysis is applicable to systems having a fixed number of unknown parameters, whereas the number of unknowns in the geometric fitting problem increases with the number of observations. Another way to interpret this effect is that each additional observation $\mathbf{x}_{n+1}$ is not one more observation of a system with a fixed number of parameters, but an observation of new system with one more

parameter $\mathbf{s}_{n+1}$. Thus the number of effective observations of the system is always one.

The second reason is that of insufficiency. A claim of asymptotic behavior as $n \to \infty$ does not necessarily translate to good results for finite n. In reality, we work with finite samples and expect n to be quite small for each neighborhood. Thus, in addition to asymptotic convergence, a guarantee of how *wrong* the estimate could be for finite n would be more useful in real application.

The next section will present a technique that enjoys the benefits of both, the ability to perform asymptotic analysis as well as to analyze finite sample behavior, and in doing so will simultaneously address the problem of choosing the support region size.

3.5. *Locally semi-parametric analysis*

From the last section, we saw that a key hindrance to performing asymptotic analysis was the introduction of nuisance parameters $\{\mathbf{s}_i\}$. One way to circumvent this obstacle is to assume a parametric form for the distribution that generates the samples $\{\mathbf{s}_i\}$. i.e. Assume that the random variable T that generates samples $\{\mathbf{s}_i\}$ follows $T \sim \text{pdf}(\nu)$ where ν denotes the unknown hyperparameters of the chosen form of distribution. The net effect of this model is to replace the set of unknowns $\{\theta, \mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$ by $\{\theta, \nu\}$. Because this fixes the number of unknowns, the standard asymptotic analysis may be performed without difficulty.

Now in general, the imposition of a distribution on samples $\{\mathbf{s}_i\}$ is a bad idea because the distribution over the unknown manifold $\mathcal{M}$ need not take a simple form. Indeed, for the same reasons cited earlier for local fitting, the form of the distribution may require too many parameters making the problem of determining the hyperparameters difficult.

However, in a *small* local neighborhood, the distribution will appear nearly uniform. Therefore one appealing form of the distribution is

$$T \sim \text{Uniform}(r), \tag{7}$$

where r is the radius of the neighborhood under consideration. Note that the fixed set of parameters in the system are now θ and r, and that radius r has been introduced as a hyperparameter.

The impact of introducing radius r is that we are now able to map the problem into the standard statistical framework of regression while incorporating knowledge of the locality of the fitting problem. Thus, any error bounds we may obtain for a given estimator will involve the free variable r which may be optimized to improve the accuracy of the solution. Since the salient feature of this method is the combination of a local parametric model for the point sample distribution with a different model at each sample point, we refer to it as a **locally semi-parametric** approach.

To contrast this approach with related work by Kanatani [15], we wish to point out that his analysis of geometric fitting demonstrated the inadequacy of classical

statistics using arguments similair to those in the previous section. However, the analysis in [15] deviate from ours in that its focus is on global fitting and hence the problem of choosing an appropriate neighborhood size does not arise. Its proposed analysis of estimator convergence rate as noise $\eta_i \to \infty$ exhibits a dual nature to classical asymptotic analysis, but by itself is also asymptotic in nature. In contrast, the locally semi-parametric approach benefits from the simple forms of distributions locally to allow computation of 2^{nd} order statistics and associated finite-sample error bounds, as well as compute asymptotic values to check for any possible bias of the estimator.

The next two sections will demonstrate the application of the above approach to the problems of reconstructing 2D and 3D curves, and extend the results to reconstructing 2D surfaces.

4. Problem: Reconstruction of Curves

In this section, we illustrate the method of locally semi-parametric analysis through the task of reconstructing curves from sample points. Evaluating the accuracy of a reconstruction algorithm by comparing two curves is not straightforward. To make this evaluation more easily quantifiable, we define the objective to be that of estimating the tangent to the curve at each observed point, which is much easier to compare and quantify with other algorithms.

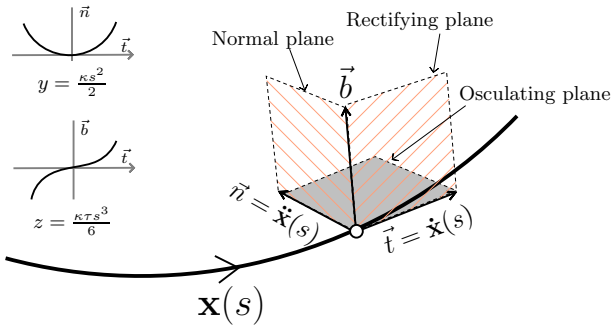
We exploit the property of local linearity in the curve through local principal component analysis (PCA) using an *adaptive* neighborhood size. Our estimate of the tangent at a point is the principal eigenvector of the scatter matrix computed in its local neighborhood. We choose this estimator as it is simple in form and has been used by other researchers for related tasks [17, 32, 33].

We propose that, for spatial curves, the neighborhood size should be chosen such that the principal eigenvalue of the scatter matrix is most closely aligned with the true tangent to the curve. To make this choice, we derive an upper bound on the expected angular error induced by finite sampling and sample noise as a function of neighborhood radius. The optimal radius is then chosen as the value that minimizes this upper bound on angular error. The ability to bound the accuracy of the estimate for a given neighborhood radius is the contribution of the locally semi-parametric approach, which we detail below.

4.1. *Curve model*

Our available data is a set of n unordered points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$. Each such point $\mathbf{x}_i = \{x_i, y_i, z_i\}$ may be thought of as a noisy observation of a true point lying on a smooth curve Γ at an (unknown) distance s_i along the curve.

Without loss of generality, we assume a Frenet reference frame (Figure 2) with origin located at the point of interest such that the tangent to the curve is aligned with the x-axis, the curvature vector in the plane of the osculating circle containing the point of interest is aligned with the y-axis and the normal to the osculating plane



is aligned with the z-axis. The neighborhood considered around the origin is defined as all points lying within the distance r along the curve from the origin. We may then adopt the semi-parametric generative model with the samples s_i assumed to be generated from a uniform distribution $S \sim \text{Uniform}(-r, r)$ with additive Gaussian noise $\eta \sim N(0, \sigma_0^2)$ as:

$$\begin{aligned} x_i &= s_i + \eta_{x,i} \\ y_i &= \frac{\kappa}{2} s_i^2 + \eta_{y,i} \\ z_i &= \frac{\kappa\tau}{6} s_i^3 + \eta_{z,i}, \end{aligned} \tag{8}$$

which is valid for moderate slowly changing values of curvature (κ) and torsion (τ). i.e. in a local neighborhood around the point of interest, curvature κ and torsion τ are assumed bounded and near constant, i.e. $\dot{\kappa}(s), \dot{\tau}(s) \approx 0$. We also assume Independent and Identically Distributed (i.i.d.) sensor noise that is zero-mean normally distributed with variance σ_0^2 affecting all three coordinates. In practice, this allows the value of σ_0 to differ across the scene to account for variation in noise level with distance from the laser sensor.

4.1.1. *The covariance matrix for curves*

One technique to estimate the direction of the local tangent at a given sample point on a curve is to look at the shape of a scatter matrix computed using points in its neighborhood [17, 32, 33]. If the curve is smooth, it is reasonable to expect that the scatter matrix will be elongated and that its major axis, or principal eigenvector, will approximate the direction of the local tangent for some appropriate (and unknown) range of neighborhood sizes. In this and the following subsection, we will derive and analyze the conditions under which this assumption will hold for both 2D and 3D curves.

The random variables X, Y and Z (denoted in capitals to distinguish them from the data) are noisy functions of the random variable S whose distribution is

assumed to be locally uniform. Hence the distribution of X, Y and Z, as well as estimators of their 1st and 2nd order statistics will depend on the coefficients (κ, τ) and order of the functions (given in (8)) as well as properties of the uniform (for S) and Gaussian (for η) distributions.

We start by computing the mean and variance of the estimators used to construct the sample covariance matrix $\hat{M}_n$. We will denote the true means of random variables by μ (e.g. μ_X for X) and standard deviation by σ (e.g. σ_X^2 for variance of X). Then

$$\hat{M}_n = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{12} & M_{22} & M_{23} \\ M_{13} & M_{23} & M_{33} \end{bmatrix}, \tag{9}$$

where

$$M_{11} = \frac{\sum_i (x_i - \bar{X}_n)^2}{n-1} \qquad M_{12} = \frac{\sum_i (x_i - \bar{X}_n)(y_i - \bar{Y}_n)}{n-1} \tag{10}$$

$$M_{22} = \frac{\sum_i (y_i - \bar{Y}_n)^2}{n-1} \qquad M_{13} = \frac{\sum_i (x_i - \bar{X}_n)(z_i - \bar{Z}_n)}{n-1} \tag{11}$$

$$M_{33} = \frac{\sum_i (z_i - \bar{Z}_n)^2}{n-1} \qquad M_{23} = \frac{\sum_i (y_i - \bar{Y}_n)(z_i - \bar{Z}_n)}{n-1}, \tag{12}$$

and $\bar{X}_n = \frac{1}{n}\sum_i x_i$ is the sample mean estimator for X, and similarly for $\bar{Y}_n$ and $\bar{Z}_n$.

Note that the diagonal elements are unbiased estimators for variance (e.g. M_{11} is the estimator for variance σ_X^2 of X) and the off-diagonal elements are unbiased estimators of covariance (e.g. M_{13} is the estimator for covariance $\mathrm{cov}(X, Z)$ of X and Z).

From the distribution of $S \sim \mathrm{Uniform}(-r, r)$, we can then compute the expected values of each of the above quantities.

For example, using $X = S + \eta_X$

$$\begin{aligned} \mathbb{E}(M_{11}) &= \mathbb{V}(X_i) = \mathbb{V}(S + \eta) = \mathbb{V}(S_i) + \sigma_0^2 \\ &= \sigma_X^2 + \sigma_0^2 = \int_{-r}^{r} s^2 \frac{1}{2r} ds + \sigma_0^2 = \frac{r^2}{3} + \sigma_0^2. \end{aligned} \tag{13}$$

Using a similar procedure, we can derive the following identities under the model

defined in (8).

$$\mathbb{E}(M_{12}) = \mathrm{cov}(X, Y) = \frac{\kappa}{2}\mathbb{E}(S^3) = 0 \tag{14}$$

$$\mathbb{E}(M_{13}) = \mathrm{cov}(X, Z) = \frac{\kappa\tau}{6}\mathbb{E}(S^4) = \frac{\kappa\tau}{30}r^4 \tag{15}$$

$$\mathbb{E}(M_{22}) = \mathbb{V}(Y) = \frac{\kappa^2}{4}\mathbb{V}(S^2) + \sigma_0^2 = \frac{\kappa^2}{45}r^4 + \sigma_0^2 \tag{16}$$

$$\mathbb{E}(M_{23}) = \mathrm{cov}(Y, Z) = \frac{\kappa^2\tau}{18}(\mathbb{E}(S^5) - \mathbb{E}(S^2)\mathbb{E}(S^3)) = 0 \tag{17}$$

$$\mathbb{E}(M_{33}) = \mathbb{V}(Z) = \left(\frac{\kappa\tau}{6}\right)^2 \frac{r^6}{7} + \sigma_0^2 \tag{18}$$

To proceed from here, we must use results on the variance of the sample variance and sample covariance estimators. We state them below, and their proofs may be found in [37].

Identity 1 (Variance of the *sample variance* estimator).

$$\mathbb{V}(\hat{\sigma}_X^2) = \frac{d_4(X)}{n} - \frac{(n-3)}{n(n-1)}\sigma_X^4 \tag{19}$$

for a random variable X, where

$$d_n(X) \triangleq \mathbb{E}(X - \mu_X)^n. \tag{20}$$

Identity 2 (Variance of the *sample covariance* estimator).

$$\mathbb{V}(\hat{S}_{XY}) = \frac{c_2(X,Y)}{n} + \frac{\sigma_X^2\sigma_Y^2}{n(n-1)} - \frac{(n-2)}{n(n-1)}c_1^2(X,Y) \tag{21}$$

for random variables X and Y, where

$$c_m(X,Y) \triangleq \mathbb{E}\left[(X - \mu_X)(Y - \mu_Y)\right]^m. \tag{22}$$

Note that we use the hat symbol ($\hat{}$) to distinguish the estimator from the true quantity.

Under the curve model defined in (8), we can derive the expressions for $d_4(X)$, $d_4(Y)$ and $d_4(Z)$ in a manner similar to that used for (14)–(18), using the identity:

$$d_4(X + \eta) = d_4(X) + 6\sigma_0^2 d_2(X) + 3\sigma_0^4 \tag{23}$$

for any random variable X affected by normally distributed independent noise $\eta \sim N(0, \sigma_0^2)$. Note that the simplification is because the odd moments of η vanish and $\mathbb{E}(\eta^4) = 3\sigma_0^4$. We may also similarly derive the expressions for c_1 and c_2 for all pairs of X,Y and Z.

Once we have the required values for c_i and d_i, we can then substitute them back in (19) and (21) to get the variance of the individual estimators, which we denote as $\mathbb{V}(M_{ij})$ with $i, j = \{1, 2\}$. The final expressions for $\mathbb{V}(M_{ij})$ were obtained using Mathematica™ and are listed in [37] due to space limitations.

Observe that the estimator for sample covariance matrix may be expressed as the sum of the matrix of its expected value and a matrix of random variables as

$$\hat{M}_n = \bar{M} + Q. \tag{24}$$

Here $\bar{M} = \mathbb{E}(M)$ is a symmetric matrix with elements given by (14)–(18), and Q is a symmetric *perturbation matrix* of random variables each with mean 0 and variance given by the expressions listed in [37].

4.1.2. *Perturbation model*

In the previous section, we were able to express the scatter matrix ($\hat{M}_n$) computed in a local neighborhood as a sum of an uncorrupted intrinsic quantity ($\bar{M}$) and a random matrix (Q) existing due to finite sampling and noise. In this section we compute the effect of the perturbation Q on the principal eigenvector of $\hat{M}_n$.

We denote the eigenvalues of $\bar{M} = \mathbb{E}(M)$ by $\lambda_1 \geq \lambda_2 \geq \lambda_3$. Let the eigenvector corresponding to λ_1 be e_1. Let $\hat{e}_1$ be the eigenvector corresponding to the largest eigenvalue of the estimated $\hat{M}_n$. If Q is the symmetric perturbation to the positive semidefinite matrix $\bar{M}$, then the application of the matrix perturbation theorem V.3.4 from [31] yields

$$\sin\left(\angle(\hat{e}_1, e_1)\right) \leq \frac{||Q||_F}{\delta}, \tag{25}$$

where $\angle(\hat{e}_1, e_1)$ denotes the angle between the estimated $\hat{\mathbf{e}}_1$ and the true $\mathbf{e}_1$. The quantity $\delta = \lambda_1 - \lambda_2$ is the *spectral gap* of the matrix $\mathbb{E}(M)$, and $||Q||_F$ represents Frobenius norm of matrix Q.[a]

Since the matrix Q consists of random variables, we are confined to making probabilistic statements about $||Q||_F$. Using Chebyshev's inequality, the square of the value attained by each element Q_{ij} can be upper bounded by

$$Q_{ij}^2 \leq \frac{\mathbb{V}(M_{ij})}{n\epsilon}$$

with probability $1 - \epsilon$, where $\mathbb{V}(M_{ij})$ is the variance of corresponding finite sample estimator of covariance (or variance if $i = j$). Note that minimizing the RHS of (25) is equivalent to minimizing the ratio

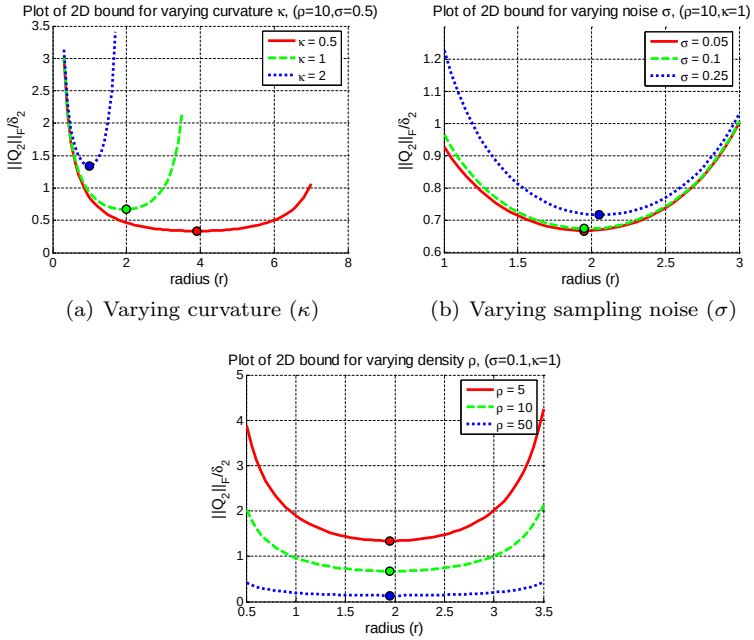
$$B \triangleq ||Q||_F/\delta. \tag{26}$$

We will analyze the function B for both 2D and 3D curves in the next section.

4.1.3. *Angular bounds and their behavior*

We first analyze the behavior of the perturbation bound to variation in sampling density, noise and curvature by looking at the slightly simpler case of 2D curves.

[a]A similar result was used in [27] where the authors invoked theorem V.2.8 from [31] to bound $||\hat{\mathbf{e}}_1 - \mathbf{e}_1||$ and analyzed the stability of document-link matrices constructed for ranking web pages.



(a) Varying curvature (κ) (b) Varying sampling noise (σ)

(c) Varying sample density (ρ)

Fig. 3. Plot of analytic 2D bound for varying sampling and geometry parameters

2D curves

We analyze the 2D case by working with the same assumptions as stated earlier except that we discard the z coordinate (or equivalently nullify torsion). The scatter matrix in this case is obtained as the top left 2×2 sub-matrix of Q, which we will refer to as Q_2. From our perturbation model in Section 4.1.2, we know that Frobenius norm of Q_2 is upper bounded with probability $1 - \epsilon$ by

$$\begin{aligned} ||Q_2||_F^2 &\leq \frac{1}{n\epsilon} \sum_{i=1}^{2} \sum_{j=1}^{2} \mathbb{V}(M_{ij}) \\ &= \frac{1}{n\epsilon} \left[\mathbb{V}(M_{11}) + \mathbb{V}(M_{22}) + 2\mathbb{V}(M_{12}) \right]. \end{aligned} \tag{27}$$

The spectral gap δ_2 of the corresponding top-left 2×2 sub-matrix $\bar{M}_2$ of $\mathbb{E}(M)$ given by

$$\bar{M}_2 = \begin{bmatrix} \frac{r^2}{3} + \sigma_0^2 & 0 \\ 0 & \frac{\kappa^2}{45} r^4 + \sigma_0^2 \end{bmatrix} \tag{28}$$

is obtained easily by inspection as

$$\delta_2 = \frac{r^2}{3} - \frac{\kappa^2 r^4}{45}. \tag{29}$$

This implies that for the dominant eigenvector of $\bar{M}$ to be the vector $\begin{bmatrix}1 & 0\end{bmatrix}$, the value of radius r must satisfy

$$0 < r < \sqrt{15}/\kappa. \tag{30}$$

The bound to be minimized then is

$$B_2(r) \triangleq \frac{||Q_2||_F}{\delta_2}. \tag{31}$$

To study the analytical behavior of this bound, we need to replace the discrete parameter n by a continuous function of radius r, and explicitly express their dependency. To do this, we use the assumption of minimum local point density ρ and substitute $n = 2\rho r$ to form the analytical plots that follow.

Note, however, that in the implementation of the proposed algorithm we directly set n in (31) to equal the number of points observed in the neighborhood of candidate radius r and do not ever need to estimate ρ. The assumption of an underlying ρ is used *only* for studying the expected behavior of the analytic bound in synthetic data and is *not* used at runtime.

Before proceeding, we point out that there are two expected limitations in the functional analysis of the derived expressions that will be relevant in their experimental validation. Firstly, although the bounds are discontinuous functions of high order polynomials in r, our analysis is restricted to the regime where the constraints (30) required for eigenvector dominance are satisfied. In this regime, the bound is convex with a unique minimum.

Secondly, and as also observed experimentally in [22, 26], the predicted error tends to 0 as $r \to 0$ for noise-free data. But for $\sigma_0 > 0$, the error tends to sharply increase for the same condition. This behavior is not reflected in our model as our continuous relaxation of n as $2\rho r$ is invalid for small r. Hence, we advocate the interpretation of the function only in terms of the behavior of its minima in the meaningful regimes of interest, rather than throughout the domain of the function.

Based on the analytical plots of $B_2(r)$ in Figures 3(a)–3(c), we make the following qualitative observations:

(1) Complexity: The closed-form expression in (31) unfortunately does not have a simple form. However, it can be easily shown that the terms with coefficients $(n(n-1))^{-1}$ in the numerator of $B_2(r)$ are dominated by the others for integer values of $n \geq 2$, reducing the expression to the ratio of the root of a 6th degree polynomial and a 4th degree polynomial of r, both only containing even powers of r.
(2) Variation with curvature κ: Figure 3(a) plots the function B_2 for multiple values of κ and fixed values of noise and sampling density. As one would expect, the optimal radius r tends to increase with decreasing curvature in order to compensate for noise and sparsity, without exceeding the bounds in (30) when the eigenvector more closely aligned to the x-axis is no longer dominant.

(3) Variation with sampling noise σ_0 : Figure 3(b) plots the function B_2 for multiple values of κ and fixed values of noise and sampling density. It can be seen that as the noise increases, the point of minima of B_2 increases but only approaching the required bounds for eigenvector dominance in (30).
(4) Variation with sampling density ρ : Figure 3(c) plots the function B_2 for multiple values of sampling density and fixed values of noise and curvature. It is interesting to note that although the value of the bound decreases as expected with increased number of points, the location of the extremum hardly changes. This is in contrast with the observations in [26] for surfaces which varies r with $\rho^{-0.5}$. We validate this later in Section 6.2.

4.1.4. *3D curves*

The derivation and behavior of the angular bound for 3D curves is fairly similar to the 2D case. From Section 4.1.2, the $||Q||_F$ is upper bounded with probability $1-\epsilon$ by

$$\begin{aligned} ||Q||_F^2 &\leq \frac{1}{n\epsilon}\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbb{V}(M_{ij}) \\ &= \frac{1}{n\epsilon}\Big[\mathbb{V}(M_{11})+\mathbb{V}(M_{22})+\mathbb{V}(M_{33})+2\left(\mathbb{V}(M_{12})+\mathbb{V}(M_{13})+\mathbb{V}(M_{23})\right)\Big]. \end{aligned} \tag{32}$$

Substituting the results from Section 4.1.1 gives the required final expression [37].

The matrix of expected values can be written as

$$\bar{M} = \mathbb{E}(M) = \begin{bmatrix} \frac{r^2}{3}+\sigma_0^2 & 0 & \frac{\kappa\tau}{30}r^4 \\ 0 & \frac{\kappa^2}{45}r^4+\sigma_0^2 & 0 \\ \frac{\kappa\tau}{30}r^4 & 0 & \left(\frac{\kappa\tau}{6}\right)^2\frac{r^6}{7}+\sigma_0^2 \end{bmatrix}. \tag{33}$$

We denote the eigenvalues of $\bar{M}$ as $\lambda_1 \geq \lambda_2 \geq \lambda_3$. The spectral gap of $\bar{M}$ is not as straightforward due to its off-diagonal terms. However, we can lower bound the spectral gap using the Gershgorin circle theorem (GCT) [8]. This gives the system of inequalities:

$$|\lambda_1 - \frac{r^2}{3}+\sigma_0^2| \leq \frac{\kappa\tau}{30}r^4 \tag{34}$$

$$\lambda_2 = \frac{\kappa^2}{45}r^4+\sigma_0^2 \tag{35}$$

$$|\lambda_3 - \left(\frac{\kappa\tau}{6}\right)^2\frac{r^6}{7}+\sigma_0^2| \leq \frac{\kappa\tau}{30}r^4 \tag{36}$$

Under the conditions of (30), this gives the additional constraint on radius as

$$r \leq \sqrt{28/5\tau}, \tag{37}$$

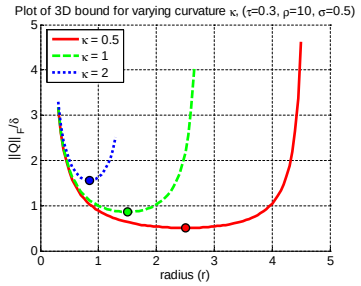


Fig. 4. Plot of analytic 3D bound for varying curvature

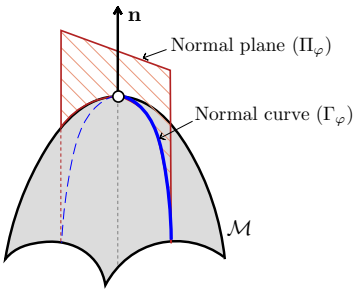


Fig. 5. Model of local surface geometry

and a bound on spectral gap as

$$\delta_3 \geq \frac{r^2}{3} - \frac{\kappa^2 r^4}{45} - \kappa\tau\frac{r^4}{15}. \tag{38}$$

Combining (32) and (38) with the continuous relaxation $n = 2\rho r$ in (26) gives the desired result.

The observations we make on the analytic behavior of $B(r)$ are analogous to those in the 2D case. The main effect of torsion is that due to its presence as an off-diagonal term in $\mathbb{E}(M)$, it always induces a finite angular offset of the dominant eigenvector in the rectifying plane (see Figure 2).

However as the radius is decreased, the off-diagonal term tends to 0 with r^4 while the leading eigenvector decays with r^2. Thus in moving from the 2D to 3D analysis, the overall effect of torsion is to decrease the optimal scale of analysis with increasing τ. This shift can be verified in Figure 4 which has the same parameters as the 2D curve of Figure 3(a) but with a non-zero torsion $\tau = 0.3$.

To summarize this section, we have shown how to relate the error in estimating the shape (specifically the tangent) of a curve locally from few observations to both its geometry related parameters, namely the curvature, torsion and sampling noise, as well as to the choice of support radius size. What this allows us to do in practice is to vary the free parameter of support radius size so that we get the best possible estimate of the model parameters. Furthermore, because an upper bound of the allowable search radius can be computed from the data, the search for the optimal radius that minimizes the error bound is easy to perform. From the plots showing the variation of the error bound to changes in geometry parameters, it can also be seen that the strategy of choosing a radius that minimizes the error bound has the behavior of automatically adapting the locality of the geometric fitting procedure to the unknown underlying shape.

5. From Curves to Surfaces

We now extend the analysis in the previous section from 2D curves to 3D surfaces by following elementary concepts from differential geometry.

Adhering to the notation from the previous section, our available data is a set of n unordered points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where each point $\mathbf{x}_i = \{x_i, y_i, z_i\}$ is now considered as a noisy observation of a point on a 3D surface $\mathcal{M}$. Without loss of generality, we assume a Darboux reference frame with origin located at the point of interest such that the surface normal is aligned with the positive z-axis, and the principal curvatures κ_1 and κ_2 are aligned with the positive x-axis and y-axis respectively.

There exists a family Π of planes that contain the origin and its normal vector. Each such plane Π_α, lying at an angle α to some reference vector in the tangent plane, intersects the surface $\mathcal{M}$ at a curve Γ_α termed the *normal curve*. (See Figure 5.)

The neighborhood considered around the origin on the surface is defined as all points lying within the distance r along *any* normal curve Γ_α from the origin. For each curve Γ_α, we may then adopt the semi-parametric generative model with the samples s_i assumed to be generated from a uniform distribution $S \sim \text{Uniform}(-r, r)$ with additive Gaussian noise $\eta \sim N(0, \sigma_0^2)$ as

$$\begin{aligned} x_i &= s_i \cos(\alpha) + \eta_{x,i} \\ y_i &= s_i \sin(\alpha) + \eta_{y,i} \\ z_i &= \frac{\kappa}{2} s_i^2 + \eta_{z,i}, \end{aligned} \tag{39}$$

where the sectional curvature [10, 41] is given by

$$\kappa = \kappa_1 \cos^2(\alpha) + \kappa_2 \sin^2(\alpha). \tag{40}$$

As before, sensor noise is assumed i.i.d. and zero-mean normally distributed with variance σ_0^2 affecting all three coordinates.

Following this generative model, each sampled point is assumed to be generated by randomly picking an angle $\alpha \in [0, \pi]$ and then picking a point from its normal curve Γ_α at distance $s \in [-r, r]$ following a uniform distribution in that interval.

Thus, the expected value of any function $g(X, Y, Z)$ of the associated random variables in a neighborhood $\mathcal{N}(0, r)$ under the above generative model may be obtained by integrating over each normal curve over all angles $\alpha \in [0, \pi]$ as

$$\mathbb{E}(g(X, Y, Z)) = \int_{-\infty}^{\infty} \int_{-r}^{r} \int_{0}^{2\pi} \frac{1}{2r} \frac{1}{\pi} g(X, Y, Z) p(\eta) d\alpha \, ds \, d\eta, \tag{41}$$

where $p(\eta)$ denotes the Gaussian distribution on the error variables.

The estimator for surface normal may be chosen in a similar manner as for tangents to curves, as the eigenvector corresponding to the minimum eigenvalue of the 3×3 scatter matrix $\hat{M}_n$ computed as per the expression (9) with terms given by (10) through (12).

The locally semi-parametric analysis of the defined surface normal estimator then proceeds similarly to the case of curves in the previous section. First, we decompose the scatter matrix estimator into two components – an expected matrix

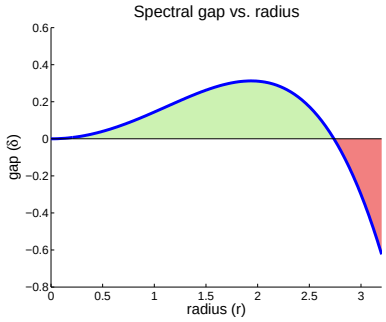


Fig. 6. Plot of spectral gap of the scatter matrix associated with points sampled on a quadric surface with $\kappa_1 = \kappa_2 = 1$. The region of the curve lying above x-axis (shaded green) marks the radius interval where the minimal eigenvector is aligned with the true surface normal in the limit. When the spectral gap changes sign beyond a critical value of radius (about 2.74 in the above scenario), the minimal eigenvector is no longer a suitable estimator for surface normal.

component $\bar{M} = \mathbb{E}(\hat{M}_n)$ and a perturbation matrix component Q that vanishes for infinite number of data points. The expected matrix may be computed analytically using (39) and (41) as

$$\begin{aligned}\bar{M} = \mathbb{E}(\hat{M}_n) &= \begin{bmatrix} d_2(X) & c_2(X,Y) & c_2(X,Z) \\ c_2(X,Y) & d_2(Y) & c_2(Y,Z) \\ c_2(X,Z) & c_2(Y,Z) & d_2(Z) \end{bmatrix} \\ &= \begin{bmatrix} \frac{r^2}{6} + \sigma^2 & 0 & 0 \\ 0 & \frac{r^2}{6} + \sigma^2 & 0 \\ 0 & 0 & \frac{r^4\left(17\kappa 1^2 - 2\kappa 1\kappa 2 + 17\kappa 2^2\right)}{1440} + \sigma^2 \end{bmatrix}.\end{aligned} \tag{42}$$

Thus the spectral gap δ is given simply by inspection as

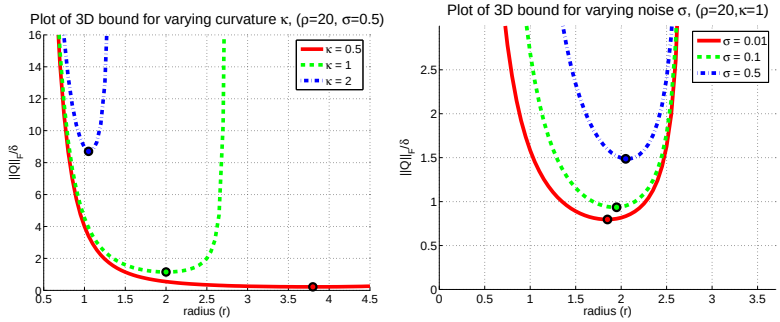
$$\delta = \frac{r^2}{6} - \frac{r^4\left(17\kappa_1^2 - 2\kappa_1\kappa_2 + 17\kappa_2^2\right)}{1440}. \tag{43}$$

Thus, for the minimal eigenvector of the computed scatter matrix $\hat{M}_n$ to be aligned with the true surface normal, the condition $\delta > 0$ must be satisfied. This translates to

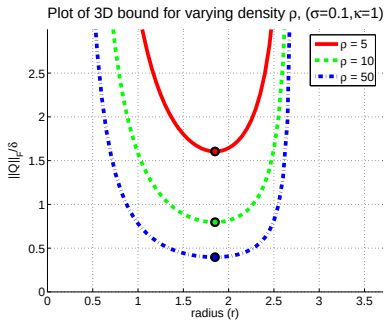
$$0 < r < \sqrt{\frac{240}{17\kappa_1^2 - 2\kappa_1\kappa_2 + 17\kappa_2^2}} \tag{44}$$

when $\kappa_1, \kappa_2 \geq 0$. Figure 6 plots the variation in spectral gap for $\kappa_1, \kappa_2 = 1$. It may be seen that beyond a certain critical radius, the spectral gap changes sign. Hence this critical radius may be used to bound the search for the radius r that minimizes the estimation error derived below.

From Section 4.1.4 the perturbation $||Q||_F$ is upper bounded with probability



(a) Varying curvature (κ) (b) Varying sampling noise (σ)



(c) Varying sample density (ρ)

Fig. 7. Plots of analytic error bound for computing normals to surfaces of varying geometry parameters

$1-\epsilon$ by

$$\begin{aligned} ||Q||_F^2 &\leq \frac{1}{n\epsilon}\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbb{V}(M_{ij}) \\ &= \frac{1}{n\epsilon}\Big[\mathbb{V}(M_{11})+\mathbb{V}(M_{22})+\mathbb{V}(M_{33})+2\left(\mathbb{V}(M_{12})+\mathbb{V}(M_{13})+\mathbb{V}(M_{23})\right)\Big], \end{aligned} \tag{45}$$

where the individual variance terms may be computed using the Identities 1 and 2 from Section 4.1.1.

Thus error in the estimate of surface normal $B(r)$ from the true normal is bounded by

$$B(r) \triangleq \frac{||Q||_F}{\delta}. \tag{46}$$

From the plots of $B(r)$ in Figures 7(a)–7(c), we can make the following qualitative observations:

Algorithm 1 Estimation of tangents (normals) from unorganized points

Data: Points $X = \{\mathbf{x}_i\} \in \mathbb{R}^3$ with $i = 1 \dots n$.

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** $x \in \{\mathbf{x}_i\}$ **do**
3: Choose starting neighborhood radius $r_{t=0}$, say using distance to the k-th nearest neighbor for a small k.
4: **repeat**
5: Estimate curvature $\kappa_t(r)$ (and torsion $\tau_t(r)$) for points in geodesic radius $r = r_t$ from $\mathbf{x}_i$.
6: Perform a 1D line search to solve $r_{t+1} = \arg\min B(r_t, \kappa_t, \tau_t)$ subject to boundary conditions on r to enforce positive spectral gap $\delta > 0$.
7: **until** convergence of r_t, else $t = t + 1$
8: **end for**

- Variation with curvature κ: Figure 7(a) plots the function $B(r)$ for multiple values of $\kappa_1 = \kappa_2 = \kappa$ and fixed values of noise and sampling density. As one would expect, the optimal radius r tends to increase with decreasing curvature in order to compensate for noise and sparsity upto the point where the eigenvector more closely aligned to the z-axis is no longer dominant.
- Variation with sampling noise σ_0 : Figure 7(b) plots the function $B(r)$ for multiple values of $\kappa_1 = \kappa_2 = \kappa$ and fixed values of noise and sampling density. As with the case of curves, with increasing noise, the point of minima of B increases but only approaching the required bounds for eigenvector dominance.
- Variation with sampling density ρ : Figure 7(c) plots the function $B(r)$ for multiple values of sampling density and fixed values of noise and curvature. As expected the value of the bound decreases with increased number of points, but the location of the extremum hardly changes.

6. Experiments

In this section we present experimental results to validate the theoretical behavior predicted by the models built in the previous section for curves and surfaces, and also study the numerical accuracy and stability of the resulting algorithms. We start with an outline of the algorithm procedure and draw attention to a few implementation details below.

6.1. *Algorithm and Implementation*

Algorithm 1 outlines the procedure used to estimate tangents (normals) from points sampled from curves (surfaces). We give a verbal description for the case of curves and draw attention to some implementation details below.

At $t = 0$, for a starting neighborhood size $r^{(t)}$, we estimate the curvature ($\kappa^{(t)}$) and torsion ($\tau^{(t)}$) using [22] and use a sensor model to obtain the value of sample noise. Then we perform line-estimation on r to obtain the $r^{(t+1)}$ minimizing (31), subject to (30) using values at time t. We then re-estimate $\kappa^{(t+1)}$ and $\tau^{(t+1)}$ corresponding to the new value of radius r and iterate till convergence. To prevent large changes in estimates of r between iterations, we use a damping factor $\alpha = 0.5$, although no significant difference in results was observed without it.

To estimate κ and τ for curves at each iteration, we use the procedure from [22] setting its scale parameter to the current estimate of r. Both the technique in [22] and our method for scale selection approximates distances between points along the underlying curve by a sum of edge distances in a graph constructed on the points. For the case of surfaces, we use the curvature estimation procedure suggested in [26].

We chose to construct the graph as the sum of disjoint minimum spanning trees (DMST) as suggested in [5], followed by a post-processing step of rejecting edges with length greater than that determined by our assumed minimum global density (ρ_0). Figure 8 shows an example of a construction for points acquired from a concertina wire. The range sensor used is a SICK LMS-291 attached to a custom made scanning mount. The angular separation between laser beams is $\frac{1}{4}^\circ$ over a 100° field of view. The angular separation between laser sweeps is $\frac{2}{3}^\circ$ over a range of 115°.

The construction using DMSTs has some desirable properties over traditional k-nearest neighbor or ϵ-ball schemes. In practice, it produces connected graphs without undesirable gaps and does not induce edges to clump together in noisy regions having relatively higher point density. The only parameter to be chosen is the number of spanning trees (in our case, = 2 for curves and = 4 for surfaces) and we have observed it to be robust to changes in the dataset for our choice.

6.2. *Validation with synthetic data*

In this section we use synthetic data to validate the model presented in sections 4 and 5.

6.2.1. *Curves in 2D and 3D*

As a first step, we test our model by attempting to validate the behavior predicted by the analytical bounds of Section 4.1.3 for the 2D case. The test curve is a 2D parabola and the error in tangent direction is evaluated at the apex for various values of curvature and point density. The estimation is done using PCA for various values of neighborhood radius. The reader is encouraged to compare Figures 9(a)-9(c) with the analytic curves of Figures 3(a)-3(c).

Figure 9(a) shows the observed angular error with varying curvature κ of the parabola. It can be seen to show the predicted systematic decrease in scale for increased curvature. The variation of estimation error with sample noise σ_0^2 in Figure 9(b) shows the increase in optimal scale for increased noise. Figure 9(c) shows

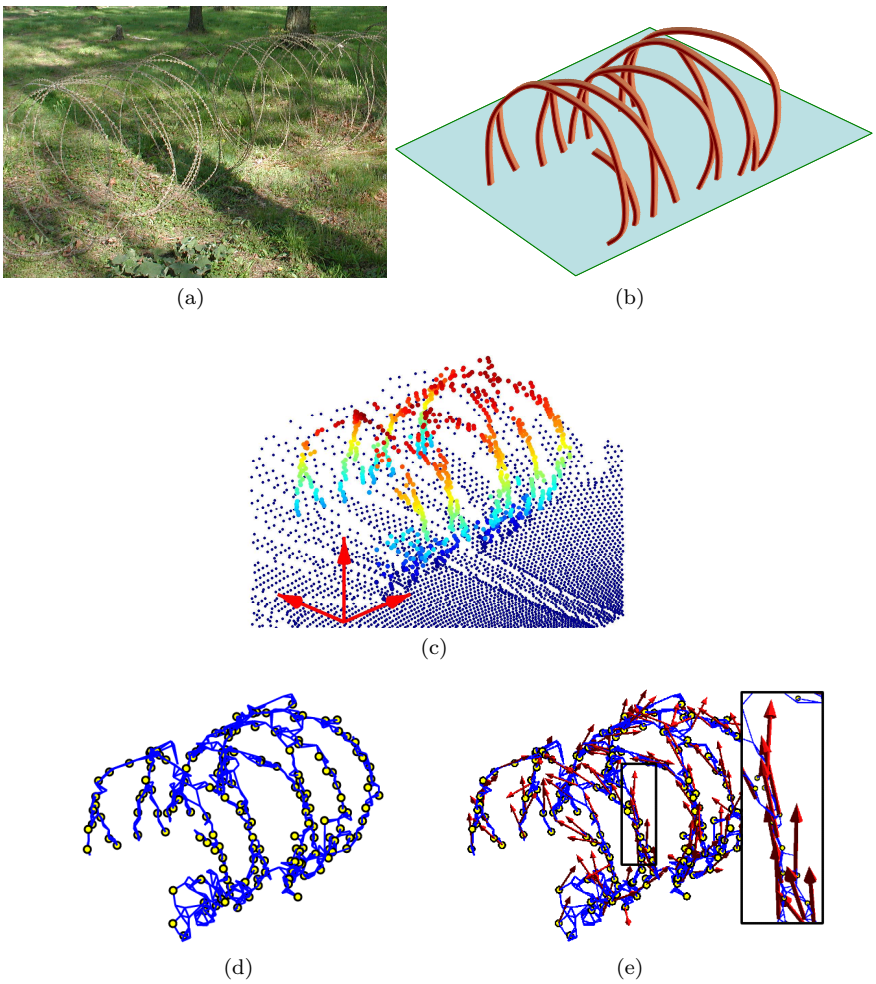
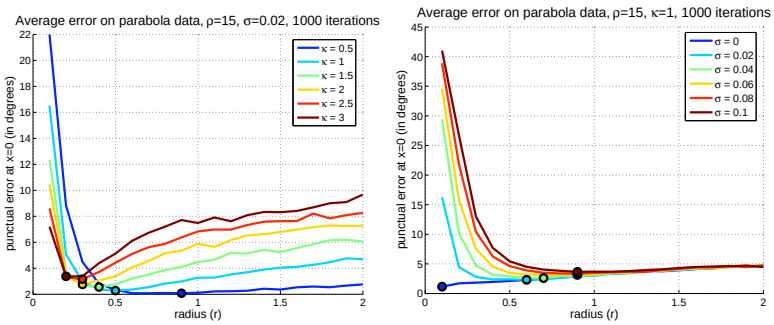


Fig. 8. Laser scan of a concertina wire having the geometry of two oppositely wound helices of equal diameter: (a) Illustration of a concertina wire (b) Scene outline showing the concertina wire structure and the ground plane (c) Raw 3-D points color-coded by elevation [axis length = 0.5m], (d) DMST graph constructed on manually extracted non-ground points, (e) Estimated tangents using scale-adaptive PCA.

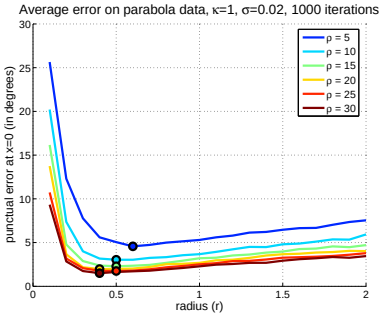
the relatively small change in choice of optimal scale except at a low density. It also shows the expected decrease in error with increasing sample density.

6.2.2. *Surfaces*

We perform a validation experiment similar to that done for curves but with a paraboloid surface. The estimate of surface normal at a fixed point is evaluated for varying values of geometric and sampling parameters. Figures 10(a), 10(b) and 10(c)



(a) Varying curvature (κ) (b) Varying sample noise (σ)



(c) Varying sample density (ρ)

Fig. 9. Plot of angular error observed when computing tangents from points sampled from 2D parabolas for varying sampling and geometric parameters. (Figure is best seen in color.)

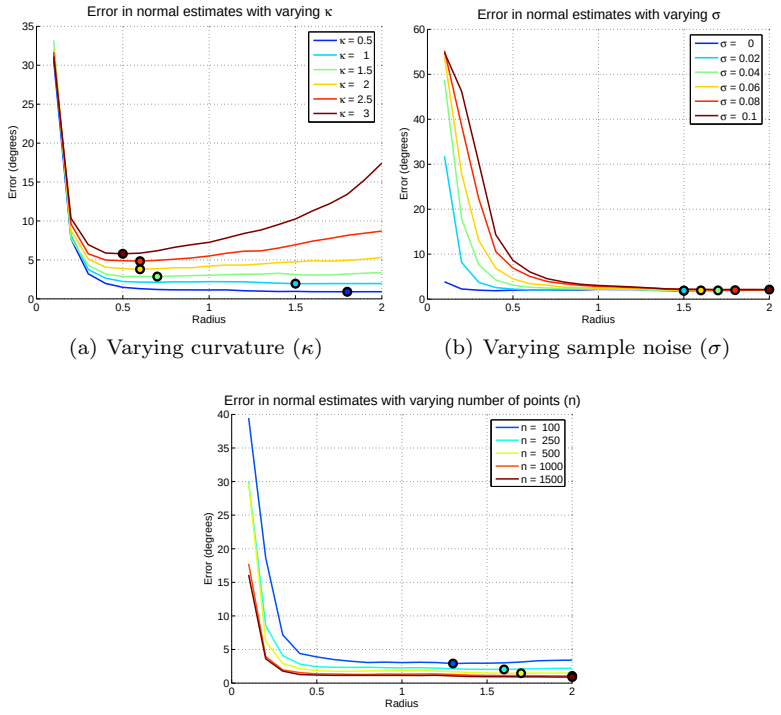
show variation in estimation error the radius of curvature, sampling noise level and number of points. The conclusion obtained on comparing these figures with the corresponding figures in Section 5 are similar to those for curves, and are omitted here for brevity.

6.3. *Stability and Accuracy*

6.3.1. *Synthetic curves*

We choose to compare the proposed method with the polynomial fitting algorithm of [22], as the latter performed nearly uniformly better experimentally on a variety of synthetic curves against a large family of other fitting approaches based on Gaussian smoothing, Fourier transforms and others.

Figure 11 presents results on 100 samples from two synthetic curves, a 2D hypocycloid and a 3D conical helix (as also used in [22]). The hypocycloid has the parametric form $\big(4\cos(t) - \cos(2t),\ 4\sin(t) + \sin(2t)\big)$ with $t \in [0, 2\pi]$ and the helix has the form $(t\cos(t), t\sin(t), t)$ with $t \in [\pi/2, 5\pi/2]$. These two are presented as their constantly varying curvature violates the assumptions made in both al-



(a) Varying curvature (κ) (b) Varying sample noise (σ)

(c) Varying no. of points (n)

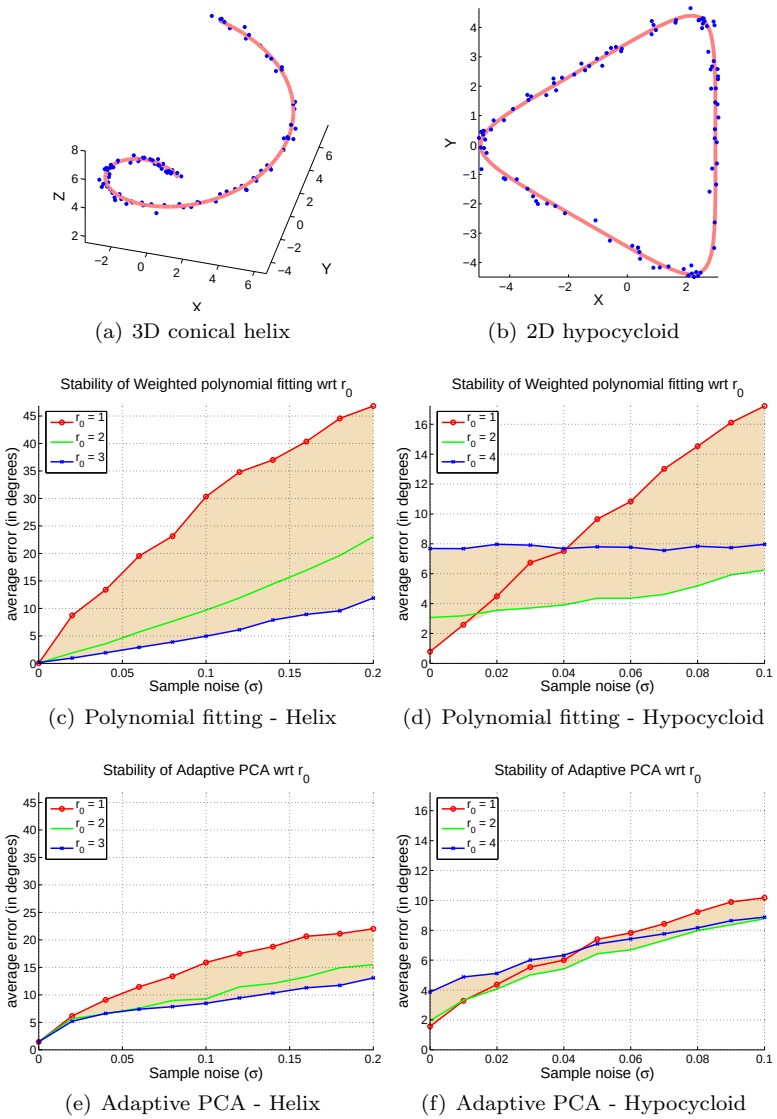
Fig. 10. Plots of angular error observed when computing normals from points sampled from paraboloid surfaces for varying geometry and sampling parameters. (Figure is best seen in color.)

gorithms, and PCA is intuitively not expected to perform well on them under its simplistic assumption of local linearity. The algorithms were run for 30 datasets each for varied sample noise (σ). A range of values for radius r_0 were used to fix the scale for polynomial fitting and correspondingly serve as the starting point of the proposed PCA algorithm.

As seen in Figure 11, the scale-adaptive PCA performs surprisingly well in terms of error rate, and is much more stable to varying values of r_0. Similar results were observed on comparison with other 2D and 3D curves from [22].

6.3.2. *Actual surfaces*

The accuracy of the adaptive PCA algorithm for surface normal estimation was tested using data from an open space natural environment containing a 1.5 m high pile of gravel surrounded by short cut and non cut grass. We collected high resolution, high density data with the Z+F laser and also collected low-resolution aerial data for the same scene with the CMU autonomous helicopter [18, 25]. The two data sets are co-registered. The Z+F data was used to produce the ground truth



(a) 3D conical helix (b) 2D hypocycloid

(c) Polynomial fitting - Helix (d) Polynomial fitting - Hypocycloid

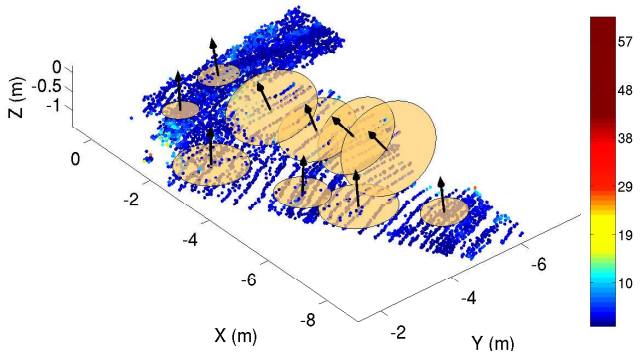
(e) Adaptive PCA - Helix (f) Adaptive PCA - Hypocycloid

Fig. 11. Plot of observed error on (a) 3D conical helix and (b) 2D hypocycloid dataset. The top row plots the true curves along with a typical example of points sampled from them. The middle row (c-d) plots error obtained with the method of [22] and the bottom row (e-f) plots error with proposed scale-adaptive PCA. Error plots in the same column have the same axis limits. The lower variation in the more stable PCA method as indicated by its thinner shaded region can be clearly seen.

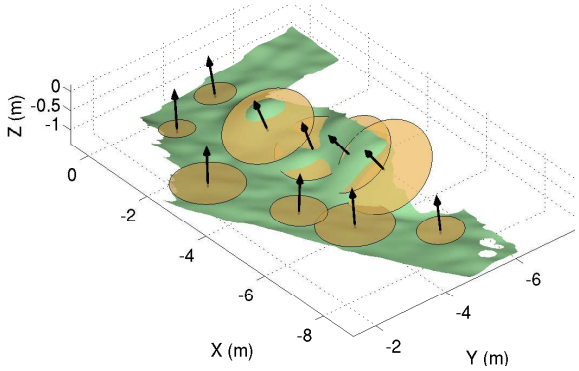
used to estimate the normal reconstruction error in the aerial data.

Figure 12 shows the results obtained. Figure 12-(a) shows the computed normals and the support regions for selected points in the aerial data. Figure 12-(b) shows

(a) Illustration of the terrain



(b) Low-resolution raw data



(c) High-resolution data used as ground-truth

Fig. 12. Normal estimation from the Zoller+Fröhlich (Z+F) laser dataset. (a) Estimated surface normals and corresponding support region for selected points are overlaid on top of the low-resolution data. Points are colored by the angular error (in degrees) of the estimated normal. Error is computed using ground-truth obtained from a high-resolution mesh (b) that also have the normals and support regions overlaid. (See text for details.)

the normal and support regions for the same points but overlaid on top of the high-resolution ground data. Points in Figure 12-(a) are color-coded by the difference between the error in estimated normals and the lowest possible error obtainable for any choice of support region in the aerial data. The algorithm was run on 6604 points and the median error was only 3.74° with an interquartile range of 5.42°.

7. Discussion

As elaborated in Section 3, the problem of fitting lines or surfaces to unorganized point clouds is one that does not fit easily into the realm of classical statistical methods. Furthermore, in trying to make formal guarantees in the accuracy of traditional methods, one is forced to make do with asymptotic guarantees which do not necessarily translate to real world performance. What this work shows is that it is possible to relate surface (or curve) geometry to finite sample accuracy of a statistical estimator, and exploit the available free parameter of support radius to minimize this estimation error. Thus, with principled scale selection, the error in tangent estimation using a simple estimator such as naïve local PCA can be made comparable, somewhat counter-intuitively, to the best fixed-scale alternative based on local polynomial fitting.

It is important to realize that the method of analysis proposed in this work is not restricted to the PCA-based estimators presented in Section 4 and Section 5. These particular choices were largely motivated by the need to study existing approaches that are popularly used [14, 26, 32, 33, 42] for the same kinds of tasks. There is potential for modifying many other algorithms that may be recast as local model fitting methods so that the chosen support radius is automatically adapted to the fitted surface. We may also expect some of these alternatives to have better numerical accuracy and stability than the algorithms presented in this paper. For instance, by choosing to fit a local quadric instead of a local plane for the surface fitting problem of Section 5, we can eliminate the need for a separate procedure to estimate the mean curvature κ and explicitly reparameterize $\mathbf{x}$ as a function of only the coefficients of the fitted surface.

One of the assumptions made about the surfaces (and curves) reconstructed with the proposed method is that they vary smoothly. This can pose challenges in several domains, particularly those involving man-made objects, where the underlying geometry consists of surfaces that are only piece-wise smooth. Such objects possess sharp features such as corners and edges which are created when these smooth surfaces intersect. One way to overcome this is may be to use a modified local regression technique, such as presented in [35], that models sharp intersections as an implicit product of lower dimensional subspaces. The geometric fitting of these sharp features may be done by iteratively first solving a generalized eigenvector problem to fit a smooth surface and then subjecting the solution to some non-linear constraints required for the model parameters to represent a degenerate surface. By using a local parameterization, the component of the model fitting problem involving the generalized eigenvector solution may then be analyzed in the manner presented in this paper.

For future work, it would be interesting to study the effect of different methods for neighborhood graph construction on the result of both algorithms. The same theoretical analysis could also be performed for the more robust variant of weighted PCA for some fixed family of weighting functions (e.g. Gaussian). This would make

the proposed algorithm more robust overall to outliers as well as to poor graph construction.

Acknowledgments

This document was prepared through collaborative participation in the Robotics Consortium, and we gratefully acknowledge sponsorship by the U.S Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-209912.

References

1. M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Point set surfaces. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 21–28, Washington, DC, USA, 2001. IEEE Computer Society.
2. N. Amenta and M. Bern. Surface reconstruction by voronoi filtering. *Discrete Computational Geometry*, 22(4):481–504, 1999.
3. N. Amenta, M. Bern, and M. Kamvysselis. A new voronoi-based surface reconstruction algorithm. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1998.
4. N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomeorphic surface reconstruction. In *Proc. 16th ACM Annual Symposium on Computational Geometry*, pages 213–222, 2000.
5. 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.
6. F. Cazals, J. Giesen, and M. Yvinec. Delaunay triangulation based surface reconstruction: a short survey. Research Report 5394, INRIA, 2004.
7. F. Cazals and M. Pouget. Estimating differential quantities using polynomial fitting of osculating jets. In *Proc. of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, pages 177–187, 2003.
8. F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
9. T. K. Dey and S. Goswami. Provable surface reconstruction from noisy samples. In *Proc. 20th ACM Annual Symposium on Computational Geometry*, pages 330–339, 2004.
10. M. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.
11. H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29:551–559, 1983.
12. H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. In *ACM Transactions on Graphics*, volume 13, pages 43–72, 1994.
13. J. Goodman and J. O. Rourke, editors. *Handbook of Discrete and Computational Geometry*, chapter 30. CRC press, 2004.
14. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *Computer Graphics*, 26(2):71–78, 1992.
15. K. Kanatani. Statistical optimization for geometric fitting: Theoretical accuracy analysis and high order error analysis. In *21st Intl. Conf. on Image and Vision Computing New Zealand (IVCNZ)*, 2006.
16. K. Kanatani. Statistical optimization for geometric fitting: Theoretical accuracy bound and high order error analysis. *International Journal of Computer Vision*, 80(2):167–188, Nov. 2008.

17. B. Kégl, A. Kryzak, T. Linder, and K. Zeger. Learning and design of principal curves. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 22(3):281–297, 2000.
18. D. Langer, M. Mettenleiter, F. Härtl, and C. Fröhlich. Imaging ladar for 3-d surveying and cad modeling of real world environments. *International Journal of Robotics Research*, 19(11), 2000.
19. I.-K. Lee. Curve reconstruction from unorganized points. *Computer Aided Geometric Design*, 17(2):161–177, 2000.
20. D. Levin. The approximation power of moving least-squares. *Mathematics of Computation*, 67(224):1517 – 31, 1998.
21. D. Levin. Mesh-independent surface interpolation. In *Geometric Modeling for Scientific Visualization*. Springer-Verlag, 2004.
22. T. Lewiner, J. D. Gomez, H. Lopes, and M. Craizer. Curvature and torsion estimators based on parametric curve fitting. *Computers and Graphics*, 29(5):641–655, 2005.
23. D. J. C. MacKay. *Introduction To Gaussian Processes*, volume 168 of *NATO ASI*, pages 133–165. Springer, 1998.
24. B. Matei and P. Meer. Estimation of nonlinear errors-in-variables models for computer vision applications. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 28:1537–1552, 2006.
25. J. Miller. *A 3D Color Terrain Modeling System for Small Autonomous Helicopters*. PhD thesis, Carnegie Mellon University, 2002.
26. N. J. Mitra, A. Nguyen, and L. Guibas. Estimating surface normals in noisy point cloud data. *Special issue of Int. Journal of Computational Geometry and Applications*, 14(4):261–276, 2004.
27. A. Y. Ng, A. X. Zheng, and M. Jordan. Link analysis, eigenvectors, and stability. In *Proc. of the Intl. Joint Conference on Artificial Intelligence (IJCAI)*, pages 903–910, 2001.
28. A. C. Oztireli, G. Guennebaud, and M. Gross. Feature preserving point set surfaces based on Non-Linear kernel regression. *Computer Graphics Forum*, 28:493–501, Apr. 2009.
29. C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.
30. V. C. Raykar and R. Duraiswami. Fast large scale gaussian process regression using approximate matrix-vector products. In *Learning Workshop, Peurto Rico*, 2007.
31. G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.
32. C. K. Tang and G. Medioni. Inference of integrated surface, curve, and junction descriptions from sparse 3-D data. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 20(11):1206–1223, 1998.
33. C. K. Tang, G. Medioni, P. Mordohai, and W. S. Tong. First order augmentations to tensor voting for boundary inference and multiscale analysis in 3-D. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 26(5):594–611, 2004.
34. B. Triggs. A new approach to geometric fitting. In *International Conference on Computer Vision*, 1998.
35. R. Unnikrishnan and M. Hebert. Denoising manifold and non-manifold point clouds. In *British Machine Vision Conference*, 2007.
36. R. Unnikrishnan, J.-F. Lalonde, N. Vandapel, and M. Hebert. Scale selection for the analysis of point-sampled curves. In *Third International Symposium on 3D Processing, Visualization and Transmission (3DPVT 2006)*, June 2006.
37. R. Unnikrishnan, J.-F. Lalonde, N. Vandapel, and M. Hebert. Scale selection for the analysis of point sampled curves. Technical report, Robotics Institute, CMU, 2006.
38. C. Walder, B. C. Lovell, and P. J. Kootsookos. Algebraic curve fitting support vector

machines. In *Digital Image Computing Techniques and Applications*, pages 693–702, 2003.

39. C. Walder, B. Schölkopf, and O. Chapelle. Implicit surfaces with globally regularised and compactly supported basis functions. In *Advances in Neural Information Processing Systems*, 2007.
40. H. Wang, C. Scheidegger, and C. Silva. Bandwidth selection and reconstruction quality in point-based surfaces. *Visualization and Computer Graphics, IEEE Transactions on*, 15(4):572–582, July-Aug. 2009.
41. T. Willmore. *Riemannian Geometry*. Clarendon Press, 1993.
42. M. Zwicker, M. Pauly, O. Knoll, and M. Gross. Pointshop 3D: An interactive system for point-based surface editing. In *Conf. on Computer Graphics and Interactive Techniques*, pages 322–329, 2002.