www.cadanda.com Computer-Aided Design AND APPLICATIONS™

Type-Constrained Direct Fitting of Quadric Surfaces

James Andrews[1] and Carlo H. Séquin[2]

[1]University of California, Berkeley, jima@eecs.berkeley.edu
[2]University of California, Berkeley, sequin@cs.berkeley.edu

ABSTRACT

We present a catalog of type-specific, direct quadric fitting methods: Given a selection of a point cloud or triangle mesh, and a desired quadric type (e.g. cone, ellipsoid, paraboloid, etc), our methods recover a best-fit surface of the given type to the given data. Type-specific quadric fitting methods are scattered throughout the literature; here we present a thorough, practical collection in one place. We add new methods to handle neglected quadric types, such as non-circular cones and general rotationally symmetric quadrics. We improve upon existing methods for ellipsoid- and hyperboloid-specific fitting. Our catalog handles a wide range of quadric types with just two high-level fitting strategies, making it simpler to understand and implement.

Keywords: reverse engineering, surface fitting, quadrics, rotational symmetry.

1. INTRODUCTION

Efficient fitting of quadric surfaces to unstructured point clouds or triangle meshes is an important component of many reverse engineering systems [17,24]. Users may want a given surface to be fit by a specific quadric type: For example, they may want the quadric to be a cone, ellipsoid, or a rotationally-symmetric subtype (spheroid, circular cone, etc). Without constraints on the quadric type, even small amounts of noise can cause the result to have an undesired quadric type – for example, a small portion of an ellipsoid sampled with some noise can often be best fit by a hyperboloid of two sheets (Fig. 1). Likewise, general quadric fitting tends to always return the most general quadric types – ellipsoids and hyperboloids – and almost never returns cylinders, cones, or rotationally symmetric shapes.

Methods for type-specific quadric fitting are scattered throughout the literature: Some papers handle spheres, circular cones and cylinders [15]; a few others handle ellipsoids [14] or hyperboloids [1]. Non-circular cones and general rotationally symmetric quadrics are not typically discussed. In this paper, we present a thorough catalog of type-specific quadric fitting methods, including new methods that handle neglected quadric types, and improvements to previously proposed methods for ellipsoid- and hyperboloid-specific fitting methods.

Reverse engineering with quadric surfaces includes several problems: first *segmentation*, to find subsets of the data that can be fit by single quadric surface patches, and then *fitting*, to find the best quadric surface parameters fitting that data. Because fitting quadric surfaces is a non-linear problem, efficient quadric-fitting methods typically work in two steps: (1) a linear, direct method (typically an “algebraic” fitting method) generates an initial guess; and (2) a non-linear, iterative method is used to refine that guess [7,15]. Chernov and Ma presented efficient, non-linear optimization techniques to handle the non-linear optimization step for any quadric type [7]. The remaining challenge, and thus focus of this paper, is the first step: generating an ‘initial guess’ that matches the desired quadric type, and is as close as possible to the error-minimizing result.

Our direct fitting methods handle a wide range of quadric types with just two high-level strategies. In the first part of our paper, we show how a closed-form line search in parameter space allows us to more-effectively fit hyperboloids, ellipsoids, and paraboloids. In the second part, we show that transforming the problems to more convenient spaces and reducing the parameters used in fitting allows us to handle all remaining cases. In all cases we ensure good results by minimizing a nearly-unbiased linear error metric introduced by Taubin [23].

Computer-Aided Design & Applications, 11(1), 2013, 107–119, http://dx.doi.org/10.1080/16864360.2013.834155

Taylor & Francis
Taylor & Francis Group

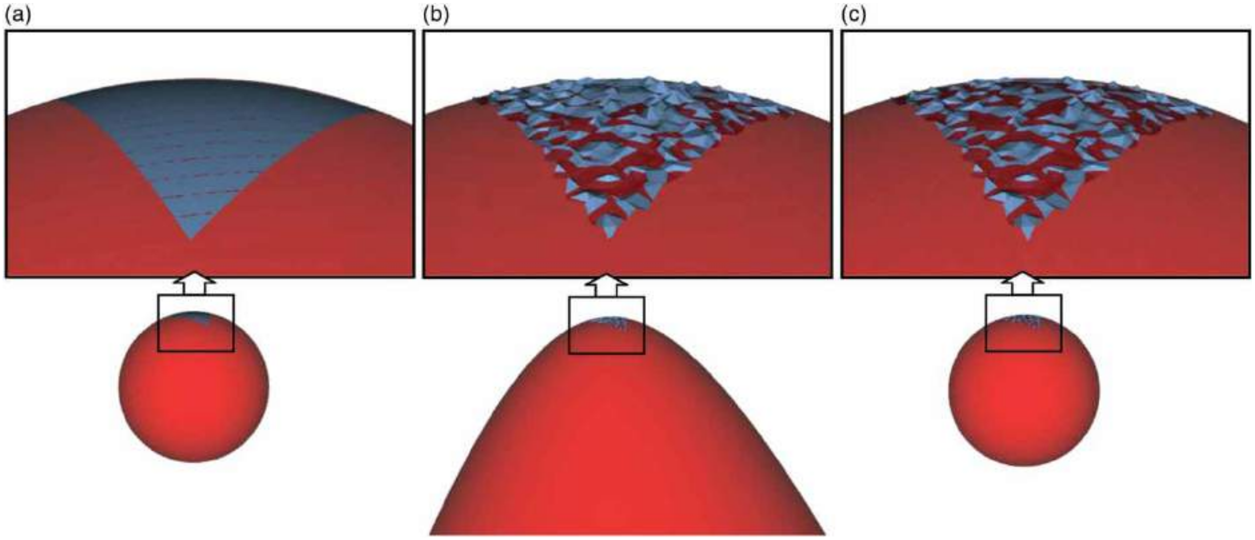

Fig. 1: The difference between an ellipsoid and a hyperboloid can be very small in terms of local surface behavior, leading to ambiguity in the best-fitting quadric type. We show a small section of a sphere (blue) with no noise (a) and with small Gaussian noise ($\sigma = 1\%$ of bounding box size) (b and c) fit by a quadric (red). The top row shows a zoomed in view, while the bottom row is zoomed out; note the shapes are almost indistinguishable in the zoomed in view. For (a) and (b) we show the result of general quadric fitting (Sec. 2.2); for (c) we show the result of sphere-specific fitting (Sec 4.5).

2. BACKGROUND

2.1. A Catalog of Quadric Types

For completeness, we give a brief overview of the different quadric types. Quadrics in their most general form are represented by a 10-parameter implicit function of the form:

$$f(\mathbf{c},\mathbf{p}) = c_0 + c_1 p_x + c_2 p_y + c_3 p_z + c_4 p_x^2 + c_5 p_y^2 + c_6 p_z^2 + c_7 p_x p_y + c_8 p_x p_z + c_9 p_y p_z = 0. \tag{2.1}$$

When categorizing quadric types, it is more convenient to rotate to a canonical, axis-aligned form. In that form, the quadratic cross terms are eliminated, and we need only consider the signs of the remaining pure squared terms. To perform this rotation, we re-write the implicit quadric equation in matrix form:

$$\mathbf{p}^\mathbf{T}\mathbf{A}\mathbf{p} + \mathbf{b}^\mathbf{T}\mathbf{p} + c_0 = 0 \text{ with } \mathbf{A} = \begin{pmatrix} c_4 & c_7/2 & c_8/2 \\ c_7/2 & c_5 & c_9/2 \\ c_8/2 & c_9/2 & c_6 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}. \tag{2.2}$$

We can then find an axis-aligning rotation of the quadric by way of an eigendecomposition: $\mathbf{A} = \mathbf{R}\mathbf{D}\mathbf{R}^\mathbf{T}$. In a rotated space $\mathbf{p_r} = \mathbf{R}^\mathbf{T}\mathbf{p}$, the new quadric becomes $\mathbf{p_r^T}\mathbf{D}\mathbf{p_r} + (\mathbf{R}\mathbf{b})^\mathbf{T}\mathbf{p_r} + c_0$. For any non-zero squared term, we can also translate the quadric to center it at the origin along the corresponding axis, which eliminates the linear term and thus results in a further-simplified canonical equation.

We list the equations of different quadratic surface types in Tab. 1. Note that all quadric types besides ellipsoids and hyperboloids exist right on the boundary of other, more general quadric types, typically right at the point where a term of the equation in canonical form becomes zero. These quadric types are then all an infinitesimal parameter change ϵ away from a more general type. When fitting a quadric of a more general type, we allow the fitting result to include bordering more-constrained types: i.e. an elliptical cylinder is a valid fitting result of ellipsoid-specific fitting. This simplifies our discussion, and avoids exclusion of, or bias against, quadrics near the boundary of their quadric type. Enforcing the “true” quadric type can always be done subsequently by perturbing the result by ϵ as needed. We illustrate these boundary relationships in Fig. 2.

Rotationally-symmetric quadric types (circular cylinders, circular cones, spheroids, and circular hyperboloids) have the same equations, with the added constraint that two squared terms have the same coefficient. In the case of a sphere, all the squared terms are equal.

2.2. Algebraic Direct Fitting Methods and Taubin’s Method

Algebraic direct fitting methods are a standard class of methods commonly used for fitting quadric surfaces [19]. In this work, we focus primarily on Taubin’s direct fitting method [23], which has been shown to be an effective, nearly-unbiased direct method for quadric fitting [7,9,21].

Equation (Canonical Form)	Quadric Type (Canonical Position)	Distinguishing Property
$c_0 + c_4p_x^2 + c_5p_y^2 + c_6p_z^2 = 0$	Ellipsoid or Hyperboloid (Centered)	Ellipsoid if all squared terms have same sign
$c_4p_x^2 + c_5p_y^2 + c_6p_z^2 = 0$	Cone (Centered)	The constant term is zero
$c_0 + c_3p_z + c_4p_x^2 + c_5p_y^2 = 0$	Paraboloid (Aligned with z-axis)	One squared term is zero
$c_0 + c_4p_x^2 + c_5p_y^2 = 0$	Cylinder (Aligned with z-axis)	Squared and corresponding linear term are both zero
$c_0 + c_4p_z^2 = 0$	Double Plane (Plane normals aligned with z-axis)	Two pairs of squared and linear terms are zero
$c_0 + c_1p_x + c_2p_y + c_3p_z = 0$	Plane	Quadratic terms all zero

Tab. 1: Equations of common quadratic surface types, in a canonical form (centered, axis aligned). In each case we indicate the key property of the equation, indicating the quadric type.

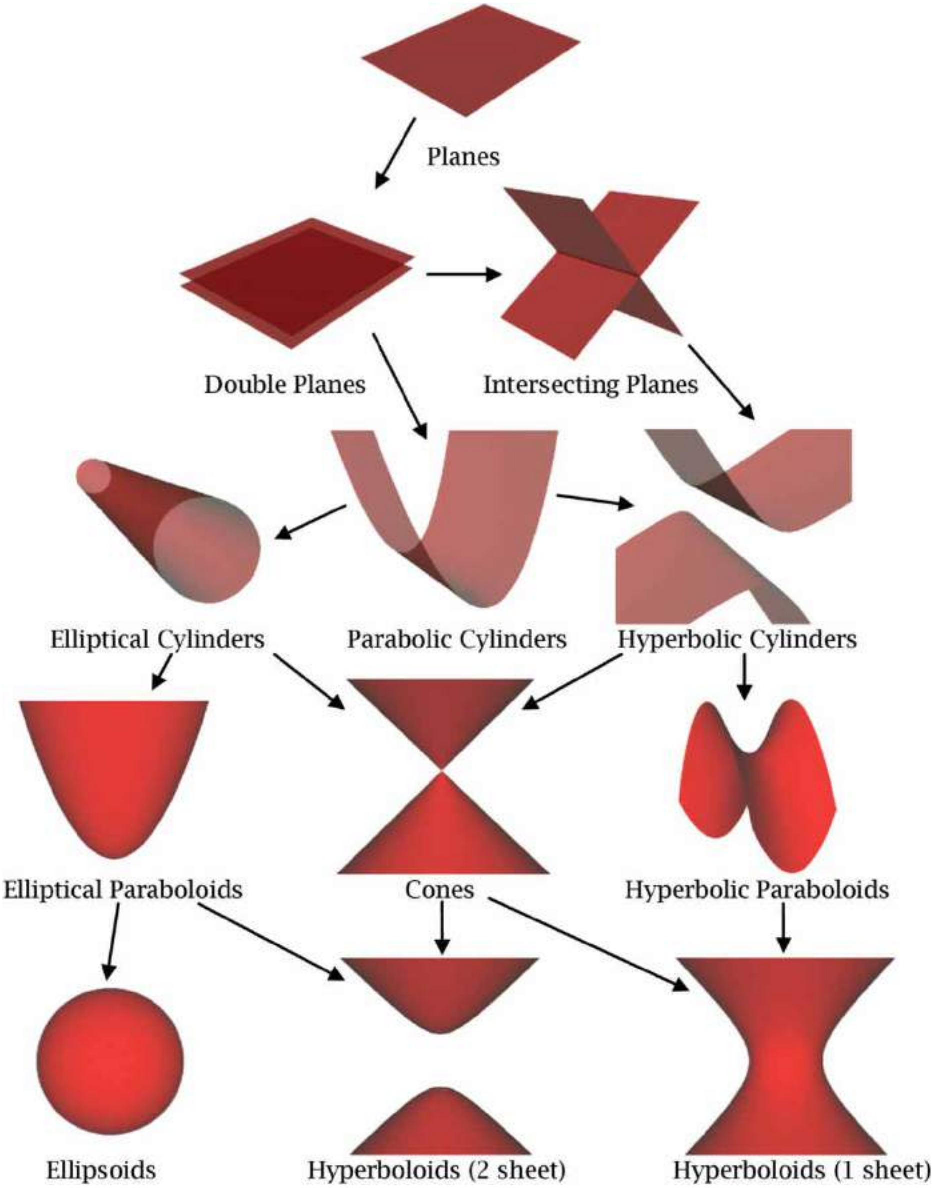

Fig. 2: A chart of quadric types, not including rotationally symmetric subtypes. Arrows indicate "is on the boundary of" relationships: One quadric type is on the boundary of another if any quadric of the first type can be perturbed by ϵ to create a quadric of the second type. These relationships are transitive, so, for example, planes lie on the boundary of all other quadric types.

All algebraic fitting methods are based on the same simple idea: We can approximate a difficult, non-linear error (like the true orthogonal distance from data points to a quadric surface) by a simple, linear approximation: $\sum_{i=0}^{n} f(\mathbf{c}, \mathbf{p}_i)^2$ where $f(\mathbf{c}, \mathbf{p})$ is an error metric that is linear in terms of the parameter vector $\mathbf{c}$. For general quadrics, we use the algebraic function that implicitly defines the quadric surface, Eqn. (2.1), as our "algebraic" error metric, $f(\mathbf{c}, \mathbf{p})$.

The scale of that linear error metric is arbitrary: Scaling the vector $\mathbf{c}$ scales the error without changing the surface. All algebraic methods therefore *normalize* the error by some quadratic normalization function, $q(\mathbf{c})$, arriving at the error metric, $\left(\sum_{i=0}^{n} f(\mathbf{c}, \mathbf{p}_i)^2\right)/q(\mathbf{c})$. Because the error and normalization are both quadratic functions, this permits an efficient solution: We define symmetric matrices $\mathbf{M}$ and $\mathbf{N}$ such that $\sum_{i=0}^{n} f(\mathbf{c}, \mathbf{p}_i)^2 \equiv \mathbf{c}^{\mathbf{T}}\mathbf{M}\mathbf{c}$ and $q(\mathbf{c}) \equiv \mathbf{c}^{\mathbf{T}}\mathbf{N}\mathbf{c}$, and take the eigenvector $\mathbf{c}$ of the generalized eigenvalue problem $(\mathbf{M} - \lambda\mathbf{N})\mathbf{c} = 0$ with smallest eigenvalue λ as our solution.

Algebraic fitting methods are distinguished by their choice of quadratic normalization function $q(\mathbf{c})$. The choice of $q(\mathbf{c})$ dramatically affects the quality of the results: A good choice can result in a fit that is nearly as good as the best non-linear fit, while a poor one will exhibit clear biases against large parts of the solutions space. For example, an ellipse-specific normalization used for 2D conic fitting [10] biases against eccentric ellipses.

The best choices of $q(\mathbf{c})$ in theory [21] and in practice [9] are Taubin's method [23], and the more recent HyperLS method [21]. HyperLS is marginally more accurate, but also more complex and less intuitive. For this paper we explain and use Taubin's method.

Taubin's method is to choose $q(\mathbf{c}) = \sum_{i=0}^{n} ||\nabla_{\mathbf{p}} f(\mathbf{c}, \mathbf{p}_i)||^2$. This is based on Sampson's error, which is a first-order approximation of the squared distance $d(\mathbf{c}, \mathbf{p})^2$ from a point $\mathbf{p}$ to a quadric $\mathbf{c}$: $d(\mathbf{c}, \mathbf{p})^2 \approx s(\mathbf{c}, \mathbf{p})^2 = \frac{f(\mathbf{c},\mathbf{p})^2}{||\nabla_{\mathbf{p}} f(\mathbf{c},\mathbf{p})||^2}$ [22]. We can view the algebraic error $f(\mathbf{c}, \mathbf{p})$ as a product, $f(\mathbf{c}, \mathbf{p}) = s(\mathbf{c}, \mathbf{p})||\nabla_{\mathbf{p}} f(\mathbf{c}, \mathbf{p})||$, of Sampson's error approximation multiplied by the (arbitrary) magnitude of the gradient of $f(\mathbf{c}, \mathbf{p})$. This arbitrary gradient magnitude can be seen as a bias term, weighting the algebraic error at each data point. If these bias weights at the data points are smaller overall for some shapes, then the algebraic error will be smaller for those shapes, leading to a bias toward those shapes. Taubin's idea – normalizing by the squared magnitude of these bias weights – counters the aggregate effect of this bias: For example, when the bias weights are smaller overall, Taubin's normalization $q(\mathbf{c})$ is likewise also smaller, countering the algebraic error's bias towards the shape. For a rigorous analysis of how well this approach removes bias, see [21].

For any given data set, we compute matrices $\mathbf{M}$ and $\mathbf{N}$ such that $\sum_{i=0}^{n} f(\mathbf{c}, \mathbf{p}_i)^2 \equiv \mathbf{c}^{\mathbf{T}}\mathbf{M}\mathbf{c}$ and $q(\mathbf{c}) = \sum_{i=0}^{n} ||\nabla_{\mathbf{p}} f(\mathbf{c}, \mathbf{p}_i)||^2 \equiv \mathbf{c}^{\mathbf{T}}\mathbf{N}\mathbf{c}$. To do so, we define $\mathbf{l(p)}$ such that $\mathbf{l(p)} \cdot \mathbf{c} = f(\mathbf{c,p})$ (we can always do so because $f(\mathbf{c,p})$ is linear in $\mathbf{c}$), and define $\mathbf{l}_i(\mathbf{p})$ as the partial derivative of $\mathbf{l(p)}$ with respect to the i^{th} dimension. Then $\mathbf{M} = \sum_{i=0}^{n} \mathbf{l}(\mathbf{p}_i)\mathbf{l}(\mathbf{p}_i)^{\mathbf{T}}$ and $\mathbf{N} = \sum_{i=0}^{n} \sum_{j=1}^{3} \mathbf{l}_j(\mathbf{p}_i)\mathbf{l}_j(\mathbf{p}_i)^{\mathbf{T}}$. We compute Taubin's error metric as: $\frac{\mathbf{c}^{\mathbf{T}}\mathbf{M}\mathbf{c}}{\mathbf{c}^{\mathbf{T}}\mathbf{N}\mathbf{c}}$. Taubin's method applies generally to many least squares fitting problems. We will use it to fit customized quadric functions and also kinematic fields in Sec. 4.

3. FITTING METHOD FOR HYPERBOLOIDS, ELLIPSOIDS AND PARABOLOIDS

Previous direct fitting methods for ellipsoids and hyperboloids have used algebraic fitting with a custom normalization function $q(\mathbf{c})$ to ensure that at least one of the resulting eigenvectors has the desired quadric type [1,14]. These methods guarantee a hyperboloid or ellipsoid, but the result may not be a good fit: The type-constraining normalizations introduce more bias than Taubin's method, leading to poorer fitting results in the presence of noise, as shown in Fig. 3(a,c).

Others have proposed exploring the space of solutions returned by the fitting method: While the best fit may not have the desired type, algebraic fitting uses a generalized eigenvalue method that returns a basis of solutions, and one of the other eigenvectors *might* have the desired type [1]. However, these eigenvectors are not optimal in terms of fitting error – in fact, even the second-best eigenvector tends to correspond to a very poor fit, as we illustrate in Figs. 4 and 5.

Our approach is to combine these ideas with previous work in the domain of 2D conic fitting: Harker et al. [12] showed that an effective approach for ellipse- and hyperbola-specific fitting is to rely on the biased ellipse- or hyperbola-specific fitting methods to ensure the correct conic type is found, but then compensate for the bias by searching a linear subspace of conics for a better-fitting result.

In this section, we generalize the search method of Harker et al. to quadrics. We improve the method by more carefully considering which subspace we should search, and we simplify the search by introducing the observation that, when Taubin's method does not return the desired type, then the best fit is right on the boundary of that type; i.e. it is within ϵ of being a paraboloid. This result also leads us naturally to a method to fit paraboloids, and to constrain the number of sheets in the hyperboloid fit.

3.1. How to Define a Good Linear Subspace of Ellipsoids and Hyperboloids

Our goal is to search a linear subspace spanning both hyperboloids and ellipsoids for the best quadric

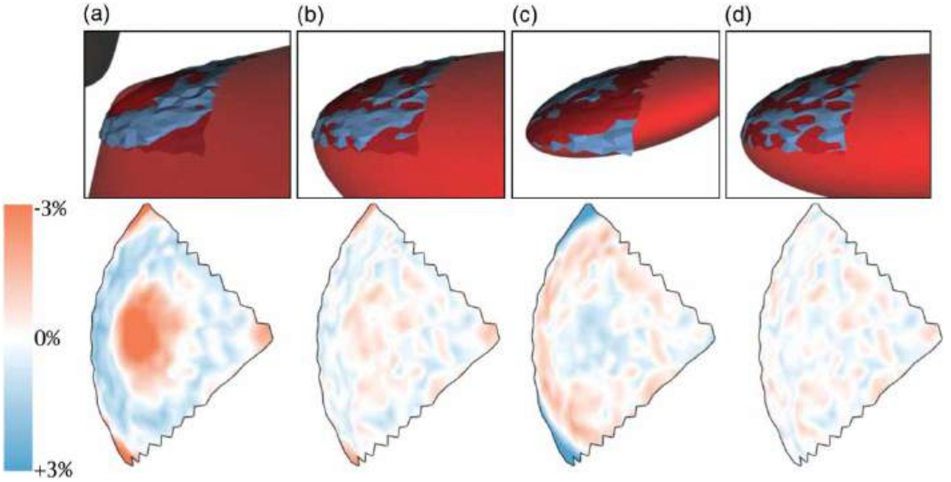

Fig. 3: Ellipsoid- and hyperboloid-specific fitting results for an 8th of an ellipsoid with vertices perturbed by Gaussian noise ($\sigma = 0.5\%$ of bounding box size). We show the results of Allaire et al.'s hyperboloid- and ellipse-specific fitting methods (a) and (c) respectively, and our improved methods (b) and (d). Top row: Side view of mesh to be fit (blue) and quadric fitting result (red). Bottom row: Heat maps of error of fit over mesh surface, with error key on left. Errors are orthogonal distances from mesh surface to quadric, relative to bounding box size, ranging from 3% outside (blue, bottom of key) to 3% inside (red, top of key). Note the high-error regions in the center of (a) and corners of (c).

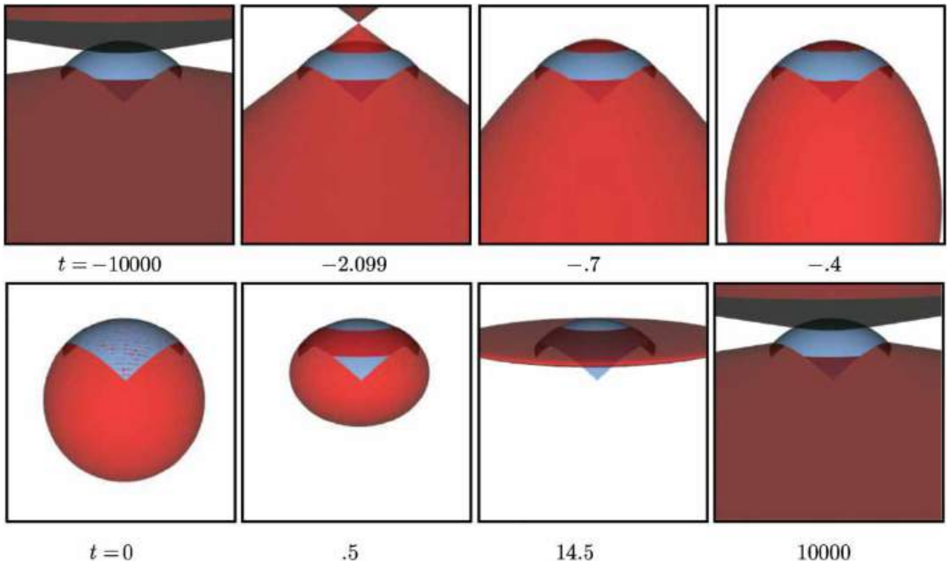

Fig. 4: A sampling of the quadrics in a linear subspace $\mathbf{c}_a + t\mathbf{c}_b$, where $\mathbf{c}_a$ and $\mathbf{c}_b$ are the best and second-best eigenvectors of Taubin's fitting method applied to a section of a sphere. In each image the quadric is shown in red, and the data (a section of a sphere) is shown in blue. As parameter t goes from -10000 to 10000, the quadric shape goes through hyperboloids of one and two sheets, ellipsoids, cones, and paraboloids. When t is near zero, the quadrics approximate the data well.

of the desired type. First, we need to identify a good subspace to search: That is, we need to identify two parameter vectors $\mathbf{c}_a, \mathbf{c}_b$, such that $\mathbf{c}_a + t\mathbf{c}_b$ includes both ellipsoids and hyperboloids – ideally, well-fitting ones. For example, if we have one parameter vector $\mathbf{c}_e$ that defines an ellipsoid, and another parameter vector $\mathbf{c}_h$ that defines a hyperboloid, then $\mathbf{c}_e + t\mathbf{c}_h$ is an acceptable subspace spanning both hyperboloids and ellipsoids; at $t = 0$, the original ellipsoid is reproduced, and as $t \to \infty$, the original hyperboloid is reproduced (because the scale of the quadric parameters has no effect on the shape).

We always let the first parameter vector $\mathbf{c}_a$ be the best fitting quadric under Taubin's method, without any type-specific normalization; it will be within ϵ of being either an ellipsoid or a hyperboloid (because

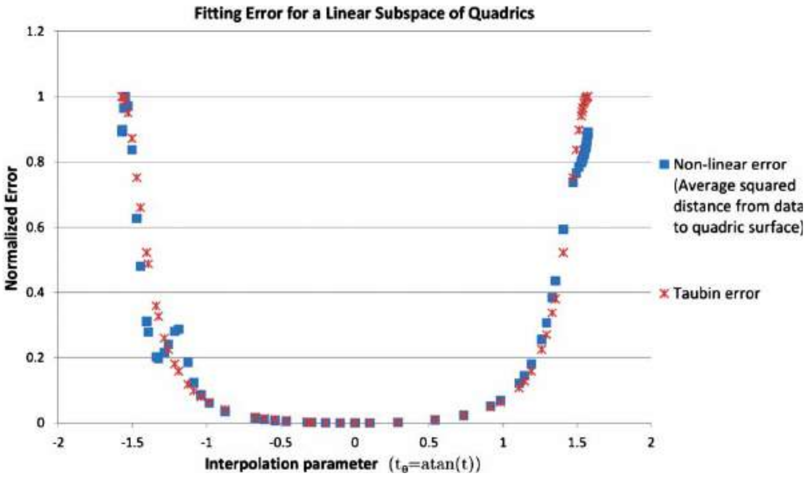

Fig. 5: Fitting errors at samples of a linear subspace of quadrics formed from the best two eigenvalues of Taubin's method (visualized in Fig. 4 above). Interpolation parameter t has been transformed by $t_\theta = \operatorname{atan}(t)$ to compress its range, and errors have been normalized by the error at the worst-fitting quadric. Taubin error (red crosses) approximates the true non-linear error (blue squares).

every quadric is, as shown in Sec. 2.1). For the second parameter vector, $\mathbf{c}_\mathrm{b}$, there are two natural strategies: We could use one of the remaining (not optimal) eigenvectors from Taubin's fit, or we could use one of the biased, constrained fitting methods. The subspace including the first- and second-best eigenvectors has the lowest maximum error of all subspaces, because the Taubin error is bounded between the corresponding eigenvalues. We find that when the data could be reasonably approximated by both ellipsoids and hyperboloids, this subspace tends to include both types and to span reasonable fitting results, as shown in Fig. 4. If the data is very far from an ellipsoid, however, the subspace may only include hyperboloids. The biased fitting methods may give poorer fitting results, but *guarantee* the resulting quadric type. We use a hybrid strategy: First we search the space of the two best eigenvectors, and if this fails to include the desired quadric type we fall back to using the biased fitting method to find the second vector $\mathbf{c}_\mathrm{b}$ by constraining it to have the type that $\mathbf{c}_\mathrm{a}$ does not have.

In the fall back case, where a biased fitting method is needed, we follow the method of Allaire et al. [1]. Allaire et al. build basis-invariant normalization functions that constrain the solution to have the desired type, using a weighted combination of basis-invariant quadratic functions of matrix $\mathbf{A}$ from the matrix form of the quadric (Eqn. (2.2)): $q_{\mathrm{Allaire}}(\mathbf{A}) = \alpha \sum \det_2^P(\mathbf{A}) + \eta \operatorname{tr}(\mathbf{A})^2$, where $\sum \det_2^P(\mathbf{A})$ is the sum of the three principal second-order minors of $\mathbf{A}$, and $\operatorname{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$. This normalization, used with the algebraic fitting framework (Sec. 2.2), guarantees the fitting solution will include an ellipsoid if we choose $\alpha = 4, \eta = -1$, and hyperboloids if we choose $\alpha = 0, \eta = -1$. Solving with this normalization will result in a set of eigenvectors; we evaluate each eigenvector and take the best under Taubin's error metric with the desired quadric type. To efficiently determine if a quadric is an ellipsoid, we check that the second leading principal minor of $\mathbf{A}$ is positive, and that the first and third leading principal minors of $\mathbf{A}$ have the same sign. Note that previous work has suggested instead using the sign of the eigenvalue corresponding to each eigenvector to determine the quadric type [14], but (as noted by that work) doing so makes the fit unable to handle some shapes such as flattened ellipsoids. Our approach of directly testing the quadric type of each eigenvector avoids this limitation.

3.2. The Best Ellipsoid or Hyperboloid is Always a Paraboloid (if it isn't the Best Quadric Overall)

Assume for this discussion that the best quadric under Taubin's metric does *not* have the desired type: Otherwise, we would have simply returned the result of Taubin's method. When the best quadric does not have the desired type, then the best quadric of the desired type must be right at the transition point between quadrics: at some exact point t where the interpolated quadric becomes the quadric of the desired type. This transitional quadric is a paraboloid (or on the boundary of a paraboloid). To prove this, we first note that Harker et al. showed that the maxima and minima of Taubin's error for *any* linear subspace $\mathbf{c}_\mathrm{a} + t\mathbf{c}_\mathrm{b}$ can be found by solving the roots of a quadratic equation in the interpolation parameter t [12]. Because this error function is always non-negative, and quadratic equations have at most two roots, it can only have one minima and one maxima. Let $\mathbf{c}'_\mathrm{a}$ and $\mathbf{c}'_\mathrm{b}$ be the minima and maxima respectively, and consider the subspace $\mathbf{c}'_\mathrm{a} + t\mathbf{c}'_\mathrm{b}$. This is equivalent to searching the original subspace $\mathbf{c}_\mathrm{a} + t\mathbf{c}_\mathrm{b}$:

Due to scale invariance, $\mathbf{c}_\mathrm{a} + t\mathbf{c}_\mathrm{b}$ spans the full planar subspace $s(\mathbf{c}_\mathrm{a} + t\mathbf{c}_\mathrm{b})$, and any two different quadrics from this planar subspace will span the same set of quadrics. In the limit as $t \to \infty$ or $t \to -\infty$ the interpolated quadric $\mathbf{c}'_\mathrm{a} + t\mathbf{c}'_\mathrm{b}$ becomes $\mathbf{c}'_\mathrm{b}$, again due to scale invariance. Therefore the maximum error is at the limits $t \to \pm\infty$, and the error is monotonic in the ranges $t \in [0, +\infty)$ and $t \in [0, -\infty)$. When the minima does not have the desired type, it follows that within each monotonic range the best quadric of the desired type must be on the boundary of that type: If it were not, we could move t closer to 0 without changing the type and arrive at a quadric of the desired type with lower error, thanks to monotonicity. Therefore, the best quadric of the desired type overall is also on the boundary of that type – i.e., it is a paraboloid.

This result is not an artifact of our choice to restrict our search to a specific linear subspace of quadrics, since the reasoning applies to any linear subspace – including an optimal subspace spanning the true best ellipsoid and hyperboloid. It is therefore a fundamental property of Taubin's metric on quadrics that the best quadric of a specific type is either the best quadric of general type, or it is right on the boundary of the desired quadric type. (It is also a property of other metrics, including HyperLS.)

Note that the result is not specific to quadrics: Any linear subspace of parameters has at most one maxima and one minima under Taubin's error, regardless of the application, so the logic of this section also applies to any other cases where Taubin's error is used. For example, it follows for 2D conics that the best ellipse or hyperbola under Taubin's metric is a parabola (if it isn't the best conic overall).

3.3. Fitting Ellipsoids or Hyperboloids

If the desired type is returned by unconstrained Taubin fitting, then we just use that result. Otherwise, we know that the best quadric of the desired type is at a transition (Sec. 3.2), and we have selected a reasonable subspace of quadrics to search for that best quadric (Sec. 3.1). We now find all the transitional quadrics in our chosen subspace, and return one with the desired type and lowest error.

To search the subspace $\mathbf{c}_t = \mathbf{c}_\mathrm{a} + t\mathbf{c}_\mathrm{b}$ for transitions between ellipsoids and hyperboloids, we refer to the matrix form of the quadric expression (2.2) to express the subspace of the quadratic coefficient matrices as an interpolated matrix $\mathbf{A}_t = \mathbf{A}_\mathrm{a} + t\mathbf{A}_\mathrm{b}$. We search for the points t where $\mathbf{A}_t$ could change its positive- or negative-definiteness. Because eigenvalues of a symmetric matrix are continuous with respect to the entries of the matrix [11], these are the points where one or more of the eigenvalues is 0; i.e., where the matrix determinant is 0. To compute the determinant of $\mathbf{A}_t$ as a function of t, we express the determinant of $\mathbf{A}_t$ with column vectors $|\mathbf{A}_t| = |\mathbf{a}_1 + t\mathbf{b}_1 \quad \mathbf{a}_2 + t\mathbf{b}_2 \quad \mathbf{a}_3 + t\mathbf{b}_3|$ where $\mathbf{a}_i$ and $\mathbf{b}_i$ are the i^{th} column of $\mathbf{A}_\mathrm{a}$ and $\mathbf{A}_\mathrm{b}$, respectively. We use the linear independence of the columns to expand the determinant to a third degree polynomial:

$$\begin{aligned}|\mathbf{A}_t| = {} & |\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_3| + (|\mathbf{b}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_3| + |\mathbf{a}_1 \quad \mathbf{b}_2 \quad \mathbf{a}_3| \\ & + |\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{b}_3|)\, t + (|\mathbf{a}_1 \quad \mathbf{b}_2 \quad \mathbf{b}_3| + |\mathbf{b}_1 \quad \mathbf{a}_2 \quad \mathbf{b}_3| \\ & + |\mathbf{b}_1 \quad \mathbf{b}_2 \quad \mathbf{a}_3|)\, t^2 + |\mathbf{b}_1 \quad \mathbf{b}_2 \quad \mathbf{b}_3|\, t^3. \qquad (3.1)\end{aligned}$$

The best quadric of the desired type is then (within ϵ of) one of the roots of this cubic polynomial. If we want a hyperboloid, then it is simply the root with the lowest error under Taubin's metric. For an ellipsoid, we also require all non-zero eigenvalues of $\mathbf{A}_t$ to have the same sign, to ensure it can be perturbed by ϵ to generate an ellipsoid.

When fitting hyperboloids, we may also wish to constrain the number of sheets in the hyperboloid we fit. If the result of hyperboloid-specific fitting has the wrong number of sheets, then Sec. 3.2 shows that the best hyperboloid with the desired number of sheets must be at the boundary of the desired type. In this case, that means it must be (within ϵ of) either a paraboloid or a cone. Therefore, if the initial fit does not have the desired number of sheets, we separately fit a paraboloid (elliptical for two sheets, hyperbolic for one sheet; see Sec. 3.4) and cone (Sec.4.3) to the data, and take whichever has the lowest Taubin error.

3.4. Fitting Paraboloids

Because the hyperboloid and ellipsoid fitting method above already finds well-fitting paraboloids, it seems natural to use the same approach for paraboloid-specific fitting. Any quadric is within ϵ of a paraboloid if its matrix of quadratic coefficients $\mathbf{A}$ is singular; i.e., if $|\mathbf{A}| = 0$. Therefore, the roots of (3.1) are all the paraboloids in a linear subspace of quadrics. We use the linear subspace of quadrics formed by the best two eigenvectors of Taubin's method, and take the quadric corresponding to the root of (3.1) with the lowest Taubin error as our paraboloid-specific fit.

To fit elliptical paraboloids specifically, we search the subspace including both ellipsoids and hyperboloids found in Sec. 3.2; this subspace must also include elliptical paraboloids because the quadric type on the boundary between hyperboloids and ellipsoids is the elliptical paraboloid (Fig. 2).

To fit hyperbolic paraboloids specifically, we first search the space of the best two eigenvectors, which generally includes hyperbolic paraboloids. However, since this is not guaranteed, in the event that no hyperbolic paraboloid is found, we suggest fitting a hyperbolic cylinder (Sec. 4.2) – the most general quadric type on the boundary of hyperbolic paraboloids (Fig. 2) – based on the intuition of Sec. 3.2.

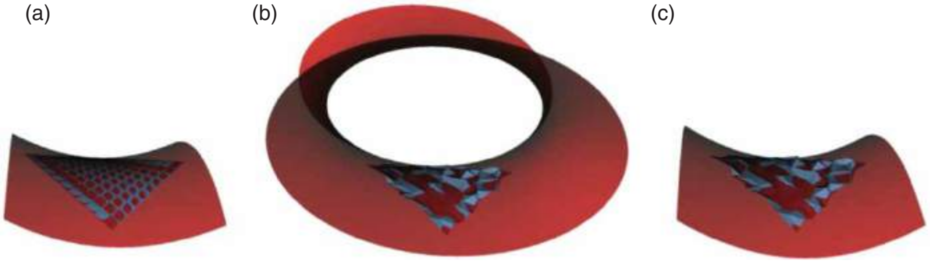

Fig. 6: Data from a hyperbolic paraboloid (blue) fit with a quadric (red). (a) Noise-free data. (b and c) Data with Gaussian noise ($\sigma = 2\%$ of bounding box size). (b) General quadric fitting finds a hyperboloid. (c) Paraboloid-specific fitting finds a hyperbolic paraboloid.

An example of hyperbolic paraboloid fitting is shown in Fig. 6.

4. FITTING METHODS FOR LOWER-DIMENSIONAL QUADRIC TYPES

Ellipsoids and hyperboloids exist in a high-dimensional parameter space with 9 degrees of freedom (having 10 unconstrained, but scale-invariant, parameters). The remaining quadric types all exist in simpler sub-spaces with fewer degrees of freedom [17]. The key to efficiently fitting these quadric types is to express the quadric with fewer parameters, such that only quadrics of the specified type can be generated, and then apply the standard algebraic fitting procedures of Sec. 2.2 on that parametrically-reduced form. For example, for planes and spheres we can simply drop and combine terms from the standard implicit quadric function to arrive at a plane- or sphere-specific function (e.g. $c_0 + c_1 p_x + c_2 p_y + c_3 p_z + c_4(p_x^2 + p_y^2 + p_z^2) = 0$ for spheres).

For most other lower-dimensional quadrics, the required low-dimensional space is more complicated: For example, there is no known linear least-squares method to fit circular cones and cylinders to a point cloud using just point positions [17]. However, there are linear least-squares methods for fitting such shapes to point clouds with normals [15]. For dense point clouds and polygonal meshes we can estimate normals (e.g. by local plane fitting, or averaging triangle normals), and then use a two-step process to fit the quadric. First, we estimate key parameters of the quadric using a direct “kinematic surface fitting” procedure that can determine properties such as a rotation symmetry axis (for a rotationally symmetric quadric), the direction in which the shape does not change (the axis of a cylindrical shape), or the central point of scaling (for a general cone) [5]. Second, we transform the data to a more convenient space and perform the standard algebraic fit in that space. In the transformed space, it is possible to reduce the quadric parameters as we did for planes and spheres. For example, to fit general cones we translate the data so that the cone apex is at the origin, and then fit a quadric with the linear and constant parameter terms dropped: $c_4 p_x^2 + c_5 p_y^2 + c_6 p_z^2 + c_7 p_x p_y + c_8 p_x p_z + c_9 p_y p_z = 0$ (Sec. 4.3).

4.1. Kinematic Field Fitting

Kinematic surface fitting is a type of fitting procedure aimed at reconstructing shapes that can be generated by sweeping a general profile curve along a simple, linear velocity field, such as surfaces of revolution [5]. These methods work by reconstructing velocity fields, and can be adapted for quadric fitting to find properties of the desired quadric: the axis of invariance for a cylinder, or the axis of rotation for a spheroid, or the central point of scaling for a general cone. We can use that property to transform the data to a space where the type-specific quadric fitting problem is easier to specify.

Kinematic velocity fields are linear functions $\mathbf{v(p)}$ that define a velocity everywhere in space. For our quadric fitting catalog we use three specific field types (Fig. 7): First, the simple general cylinder field due to [20], which defines a constant velocity field: $\mathbf{v(p)} = \mathbf{a}$, where $\mathbf{a}$ is a parameter vector defining the direction of the field. Second, we use the rotational field due to [18], which defines a rotational (or helical) motion: $\mathbf{v(p)} = \mathbf{r} \times \mathbf{p} + \mathbf{a}$, where $\mathbf{r}$ and $\mathbf{a}$ are parameter vectors encoding the rotation axis and position of the axis (plus helical motion, if any). And finally, we use a scaling field, which defines a scaling motion: $\mathbf{v(p)} = \gamma \mathbf{p} + \mathbf{a}$, where γ is a parameter defining the scaling and $\mathbf{a}$ is a parameter vector encoding the center of the scaling.

To fit a field, we need data points that include both position and normal information. (Normals, if unavailable, can be approximated for dense point clouds [6]; for meshes we use area-weighted vertex normals.) Data points fit the field if they define a surface that is tangent to the field, which we characterize by the error function $f(\mathbf{p}, \mathbf{n}) = \mathbf{v}(\mathbf{p}) \cdot \mathbf{n}$. Andrews and Séquin show [5] that an effective direct solution to this fitting problem for any field is to apply Taubin’s method (Sec. 2.2), with $\sum_{i=0}^{n} (\mathbf{v}(\mathbf{p}_i) \cdot \mathbf{n}_i)^2 \equiv \mathbf{c^T M c}$ and $q(\mathbf{c}) = \sum_{i=0}^{n} ||\mathbf{v}(\mathbf{p}_i)||^2 \equiv \mathbf{c^T N c}$, where $\mathbf{c}$ is a vector concatenating all parameters defining the velocity field (e.g. for scaling, $\mathbf{c} = [\gamma, a_x, a_y, a_z]^T$).

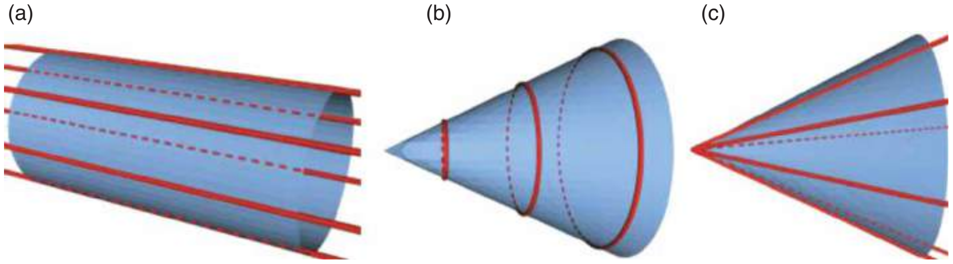

Fig. 7: Red streamlines showing velocity fields fit to mesh data: (a) constant, (b) rotational, (c) scaling.

4.2. Fitting Cylinders

Cylinders of any subtype can be fit in two steps: First, we fit a general cylinder field to find the invariant axis of the cylinder, along which the cross section does not change. Next, we rotate all points so that this axis is aligned with the z-axis. We then fit a conic of the desired type to the points: i.e., for a circular cylinder, we fit a circle; for an elliptical cylinder, we fit an ellipse. For circle fitting, we fit the equation $c_0 + c_1 p_x + c_2 p_y + c_4(p_x^2 + p_y^2) = 0$ directly using Taubin's method. For other types, we use the methods from Sec. 3 (adapted from Harker et al. [12]) for type-specific conic fitting, as explained below. The resulting conic equation, evaluated in 3D, is a cylinder along the z-axis. We finally rotate the cylinder back to the original basis.

In detail, the type-specific conic fitting method proceeds as follows: First, we fit using Taubin's method, and check the type of the best fit result. If it is the correct type, we return that and stop. Otherwise, we use biased fitting to find some (sub-optimal) conic of the desired type; specifically, use the normalization $q_{\text{Fitzgibbon}}(\mathbf{c}) = 4c_4c_5 - c_7^2$ [10]. We search the subspace between the best fit conic using Taubin's method, $\mathbf{c}_\text{a}$, and the biased fit of the desired type, $\mathbf{c}_\text{b}$, for an improved fit of the desired type. We define $\mathbf{A}_\text{a}$ and $\mathbf{A}_\text{b}$ as 2×2 matrices of quadratic terms terms, of the form $\mathbf{A} = \begin{pmatrix} c_4 & c_7/2 \\ c_7/2 & c_5 \end{pmatrix}$ (analogous to the 3×3 of Eqn. (2.2)), and define $\mathbf{A}_t \equiv \mathbf{A}_\text{a} + t\mathbf{A}_\text{b}$. Sec. 3.2 shows that the best conic of the desired type must be the transitional parabola where the conic changes type: i.e., where $|\mathbf{A}_t| = 0$. With $\mathbf{a}_i$ and $\mathbf{b}_i$ defining the i^th columns of $\mathbf{A}_\text{a}$ and $\mathbf{A}_\text{b}$ respectively, we solve for the roots of $|\mathbf{A}_t| = |\mathbf{a}_1 \quad \mathbf{a}_2| + (|\mathbf{b}_1 \quad \mathbf{a}_2| + |\mathbf{a}_1 \quad \mathbf{b}_2|)\, t + |\mathbf{b}_1 \quad \mathbf{b}_2|\, t^2 = 0$, and take the root with lowest Taubin error.

4.3. Fitting General Cones

To fit a cone, we first find the center point of scaling by fitting the kinematic scaling field $\mathbf{v(p)} = \gamma\mathbf{p} + \mathbf{a}$. The center of scaling is $-\mathbf{a}/\gamma$. Note that if $|\gamma|$ is very small, then division by γ becomes unstable. In the limit, as $\gamma \to 0$, the center of scaling moves infinitely far from the origin, so the cone becomes a cylinder. We therefore define a threshold $t_\text{cyl} = 10^{-6}$, and if $|\gamma| < t_\text{cyl}$, we consider the best cone to be a cylinder, so we can fit an elliptical cylinder instead of a cone (see Sec. 4.2). Alternatively, if a cylinder is not desired, we can take the best eigenvector returned by algebraic fitting with $|\gamma| > t_\text{cyl}$.

Once some (non-infinite) cone center is chosen, we can translate the points so that the cone center is at the origin, and fit a centered cone quadric equation $c_4p_x^2 + c_5p_y^2 + c_6p_z^2 + c_7p_xp_y + c_8p_xp_z + c_9p_yp_z = 0$ using Taubin's method. Finally, we translate the resulting quadric back to the original space.

4.4. Fitting Rotationally Symmetric Quadrics

To fit a rotationally symmetric quadric, we first find the axis of rotational symmetry by fitting the rotational field $\mathbf{v(p)} = \mathbf{r} \times \mathbf{p} + \mathbf{a}$. We normalize the resulting parameters by dividing both $\mathbf{a}$ and $\mathbf{r}$ by $\|\mathbf{r}\|$, and then find a point on the axis of the rotational field $\mathbf{p}_r = \mathbf{r} \times (\mathbf{a} - (\mathbf{a} \cdot \mathbf{r})\mathbf{r})$. If $\|\mathbf{r}\|$ is very small, then normalizing by $\|\mathbf{r}\|$ is unstable, so we use a threshold $t_\text{cyl} = 10^{-6}$: If $\|\mathbf{r}\| < t_\text{cyl}$ then we assume $\|\mathbf{r}\| \approx 0$ so the best motion is approximately a pure translation, i.e. a cylinder; we then fit a circular cylinder (see Sec. 4.2). Otherwise, we transform the data so that the z-axis is aligned with $\mathbf{r}$, and the point $\mathbf{p}_r$ is at the origin. We then fit the equation $c_0 + c_3p_z + c_4(p_x^2 + p_y^2) + c_6p_z^2 = 0$ for a general quadric with rotational symmetry, and finally transform the result back to the original basis. We can directly apply the methods of Section 3 to further constrain the quadric type.

For circular cones we either find the center of scaling and make that $\mathbf{p}_r$, then fit the simpler equation $c_4(p_x^2 + p_y^2) + c_6p_z^2 = 0$, or we can alternatively rotate the data points to a shared plane and fit a 2D line to find the angle of the cone. Note that quadric cones are in fact two cones connected at the tip, and this second method ignores the possibility that the data points may come from both cones; but it may be desirable in the common case where only one of the two cones is desired in the fit.

For spheres we directly fit the equation $c_0 + c_1p_x + c_2p_y + c_3p_z + c_4(p_x^2 + p_y^2 + p_z^2) = 0$ [2]. For planes, we translate the data to its centroid and fit the equation $c_1p_x + c_2p_y + c_3p_z = 0$ [16]. We handle double planes and intersecting planes with multiple plane fits, because we prefer single-plane solutions.

5. APPLICATIONS

Because noise in the data can easily alter the best-fitting quadric type detected by general quadric fitting (Figs. 1, 6, 8), type-specific fitting is useful in any quadric fitting application where the correct or desired quadric type is known a-priori. There are many such applications: Allaire et al. give the example of fitting a bone joint that is known to be a hyperboloid [1]; many CAD parts are known to be cylinders or cones by construction; hyperbolic paraboloids show up in architecture, art, and Pringles potato chips. In some cases it is important that the quadric surface be bounded – for example, if the whole quadric surface is expected to correspond to some real surface – and in this case, the type must be an ellipsoid and not a hyperboloid. (The best ellipsoid is within ϵ of a paraboloid (Sec. 3.2), which is effectively not bounded, but the linear subspace of Sec. 3.1 generally also contains much less eccentric, well-fitting ellipsoids, so this subspace is a good place to start a non-linear search for an ellipsoid of bounded eccentricity.) There are other cases where one knows the surface must have one of several types: Minimal surfaces ("soap film") have negative Gaussian curvature everywhere, so they should be fit by hyperboloids of one sheet or related boundary types such as hyperbolic paraboloids. Developable surfaces must have zero Gaussian curvature everywhere, so they should be fit by general cylinders, cones, and planes.

Type-specific fitting is also useful when a designer has some preference regarding the quadric type they wish to fit to some surface: They may not know a-priori what the best fit is, but still prefer to work with simpler or rotationally-symmetric quadric types. To demonstrate this kind of user-oriented application, we extend a simple region growing method from Andrews et al. [4] for selecting a quadric on a mesh surface: Starting from a user stroke on a mesh surface, the original algorithm selects the triangles under the stroke and then iteratively fits a quadric to the selection, and grows the selection to the neighboring triangles that are close to the fitted quadric (in position and orientation). This algorithm is convenient, but faces inherent ambiguity both in the best directions to grow the selection (Fig. 9, top row) and how much error should be accepted in order to fit a larger region (Fig. 9, bottom row). A user's preference for a quadric of a specific type or with a specific property, like rotational symmetry, can reduce that ambiguity (Fig. 9c). If the user's type-preference is absolute, we simply replace general quadric fitting with the appropriate type-specific fitting method. Otherwise, the user can specify how much additional error they will accept in exchange for a simpler quadric type; the system will fit both the simpler type and the general type, and use the simpler fit only if its error is within the user's additional error threshold of the general fit.

6. IMPLEMENTATION

Each direct fitting method we describe involves simply solving generalized eigenvalue problems, and in some cases finding the roots of a cubic or quadratic equation. Generalized eigenvalue problems can be solved with standard linear algebra packages (we use LAPACK [3]), but some care must be taken to avoid numerical instability and ensure good results. We observed some cases in which floating point error corrupted the ordering of the eigenvalues of a generalized eigenvector problem $\mathbf{c}^T\mathbf{M}\mathbf{c} = \lambda\mathbf{c}^T\mathbf{N}\mathbf{c}$ for clean data with low noise. To ensure that the best eigenvector is chosen, we disregard the computed eigenvalues

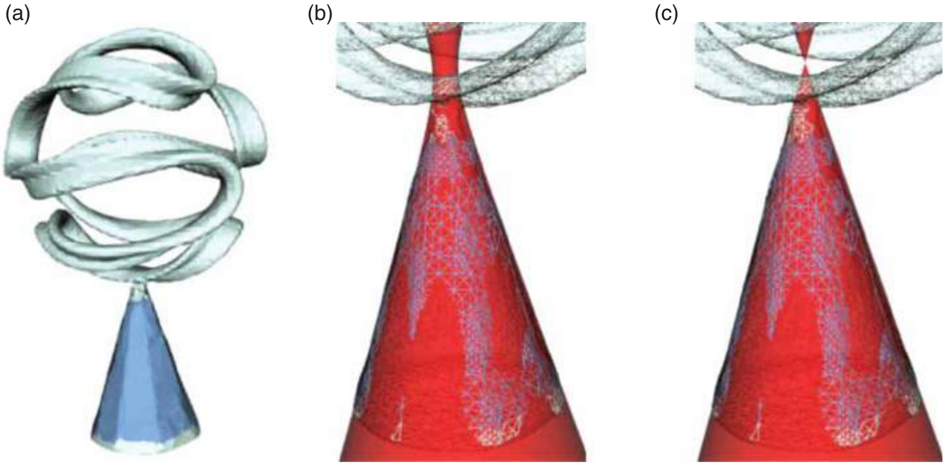

Fig. 8: An image-based reconstruction of a sculpture has a noisy cone as its base (a). We select the base (blue) and fit quadrics (red). (b) Without type specific-fitting, the result is a hyperboloid with a non-zero neck at the top. (c) With cone-specific fitting, the result is an exact cone.

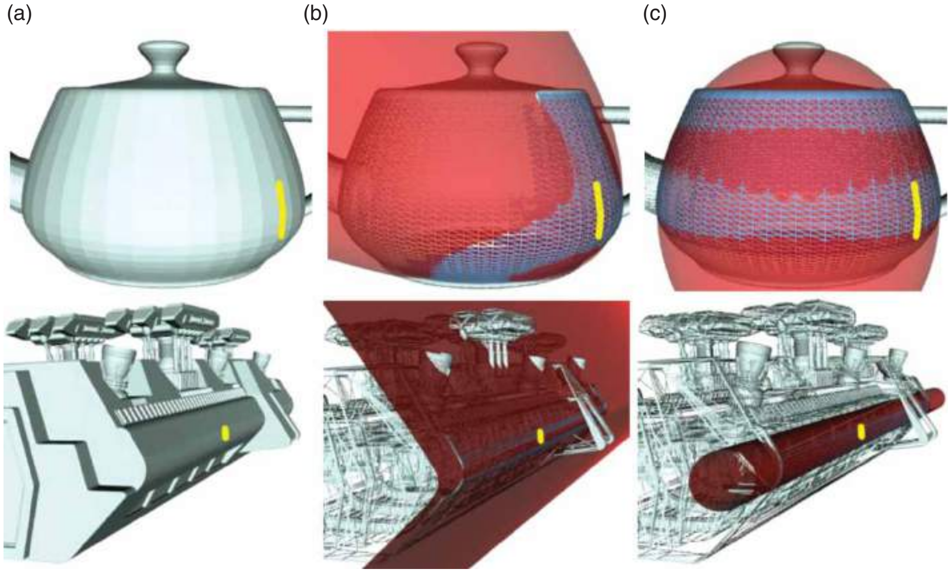

Fig. 9: (a) A user has made a small stroke (yellow) on a mesh surface (top row: teapot; bottom row: moonbus), to select a quadric surface under the stroke using a region growing method [4]. (b,c) We show the fit quadric (red) and wireframe views of the mesh (selection in blue): (b) Shows the result of general quadric fitting. (c) Shows the result of rotationally-symmetric quadric fitting.

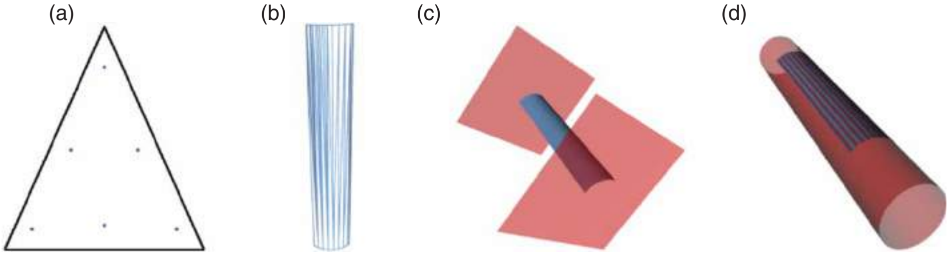

Fig. 10: (a) Dunavant's integration points. (b) A wireframe view of a mesh with sparse, non-uniformly distributed vertices. (c) Fitting result when error is measured only at vertices: A double-plane quadric that interpolates the vertices. (d) Fitting result when error is integrated over mesh surface: A cylinder.

and instead directly compute Taubin's error metric for each eigenvector. We then choose the eigenvector with lowest error (ignoring eigenvectors of the wrong quadric type, if the matrix **N** was intended to constrain the quadric type). We also use double precision floating point numbers, and center and re-scale the data points (to a unit-sized bounding box) before fitting.

Another important implementation detail is the method of sampling of the original surface. When fitting segments of a polyhedral mesh, we have two reasonable options: We could sample the error at vertices, or we could integrate the error over the surface of the mesh. Sampling error at vertices makes it easier to smooth noise implicitly, for example by using averaged vertex normals. It also leads to more intuitive results if the vertices represent true samples from an original surface, which is often the case. However, integrating the error over the surface of the mesh can greatly reduce ambiguity in the case of sparse, non-uniform sampling, as shown in Fig. 10. In our tests, we found that the reduced ambiguity can greatly improve robustness when fitting implicit quadric equations.

For ease of implementation, we perform integration over a polyhedral mesh by triangulating the mesh and using Dunavant's Gaussian quadrature rules [8]. This is simple to implement because it is exactly like fitting with discrete points: We just generate six sample points per triangle (at barycentric coordinates given by Dunavant) and weight each point according to Dunavant's rules. Because the error function (a squared quadratic) is a quartic polynomial in terms of position, Dunavant's six-point rule correctly integrates the error without approximation.

7. LIMITATIONS AND FUTURE WORK

All of the methods presented in this paper are least squares fitting methods, and therefore are sensitive to outliers. Small noise and discontinuities are not a problem, but large outliers may require robust statistical methods such as M-estimators to achieve good results [13]: i.e., iteratively fit with least squares, and then re-weighting the data based on the distance to the last fitted surface.

Some issues arise if one fits a quadric type that *cannot* match the data well. For example, if one fits a hyperbolic paraboloid to bowl-shaped data, even though hyperbolic paraboloids cannot be bowl-shaped, then the solution may look like a double-plane slicing through the bowl twice to better minimize squared error. This approximates a single patch of data with two disconnected patches of quadric surface, which is a valid error-minimizing result, but typically not desirable by the user. New constraints may need to be invented to avoid this kind of solution. Fortunately, this is only a problem in the uncommon case that one wishes to fit a quadric type to data that it cannot fit well.

The linear subspace search we perform for fitting ellipsoids, hyperboloids and paraboloids is something that could be improved in future work: Our current choice of subspace is heuristic; it would be nice to find a more rigorous choice of subspace, and a direct method to search a larger subspace (e.g., a 2D or 3D subspace instead of a linear subspace).

For simplicity, we used Taubin's method instead of the more-advanced HyperLS method [21]; for slightly better results one can use the HyperLS method instead. All analysis and methods we presented in the context of Taubin's method apply directly to the HyperLS method as well.

ACKNOWLEDGEMENTS

Thanks to Rahul Narain and Sridhar Ramesh for mathematical advice. This work was supported in part by the National Science Foundation (NSF award #CMMI-1029662 (EDI)).

REFERENCES

[1] Allaire, S.; Jacq, J.-J.; Burdin, V.; Roux, C.; Couture, C.: Type-Constrained Robust Fitting of Quadrics with Application to the 3D Morphological Characterization of Saddle-Shaped Articular Surfaces, ICCV 2007, 1–8. DOI: 10.1109/ICCV.2007.4409163.

[2] Al-Sharadqah, A.; Chernov, N.: Error analysis for circle fitting algorithms, Electronic Journal of Statistics. 3, 2009, 886–911. DOI: 10.1214/09-EJS419.

[3] Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D.: LAPACK Users' Guide, Society for Industrial and Applied Mathematics, 1999. DOI: 10.1137/1.9780898719604.

[4] Andrews, J.; Jin, H.; Séquin, C.H.: Interactive Inverse 3D Modeling, Computer-Aided Design and Applications, 9(6), 2012, 881–900. DOI: 10.3722/cadaps.2012.881-900.

[5] Andrews, J.; Séquin, C.H.: Generalized, basis-independent kinematic surface fitting, Computer-Aided Design. 45(3), 2013, 615–620. DOI: 10.1016/j.cad.2012.10.047.

[6] Boulch, A.; Marlet, R.: Fast and Robust Normal Estimation for Point Clouds with Sharp Features, Computer Graphics Forum, 31(5), 2012, 1765–1774. DOI: 10.1111/j.1467-8659.2012.03181.x.

[7] Chernov, N.; Ma, H.: Least squares fitting of quadratic curves and surfaces, Computer Vision, 285–302. 2011.

[8] Dunavant, D.: High Degree Efficient Symmetrical Gaussian Quadrature Rules for the Triangle, International Journal for Numerical Methods in Engineering. 21, 1985, 1129–1148. DOI: 10.1002/nme.1620210612.

[9] Fitzgibbon, A.; Fisher, R.B.: A Buyer's Guide to Conic Fitting, British Machine Vision Conference 1995, 513–522.

[10] Fitzgibbon, A.; Pilu, M.; Fisher, R.B.: Direct least square fitting of ellipses, IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(5), 1999, 476–480. DOI: 10.1109/34.765658.

[11] Golub, G.H.; Van Loan, C.F.: Matrix computations (3rd ed.), Johns Hopkins Univ. Press. 1996.

[12] Harker, M.; O'Leary, P.; Zsombor-Murray, P.: Direct type-specific conic fitting and eigenvalue bias correction, Image and Vision Computing, 26(3), 2008, 372–381. DOI: 10.1016/j.imavis.2006.12.006.

[13] Huber, P.J.: Robust Statistics, John Wiley and Sons, 1981. DOI: 10.1002/0471725250.

[14] Li, Q.; Griffiths, J.G.: Least squares ellipsoid specific fitting, Geometric Modeling and Processing, 2004, 335–340. DOI: 10.1109/GMAP.2004.1290055.

[15] Lukács, G.; Martin, R.; Marshall, D.: Faithful Least-Squares Fitting of Spheres, Cylinders, Cones and Tori for Reliable Segmentation, ECCV 1998, 671–686.

[16] Pearson, K.: On lines and planes of closest fit to systems of points in space, Philosophical Magazine, 2(6), 1901, 559–572. DOI: 10.1080/14786440109462720.

[17] Petitjean, S.: A survey of methods for recovering quadrics in triangle meshes, ACM Computing Surveys. 34(2), 2002, 211–262. DOI: 10.1145/508352.508354.

[18] Pottmann, H.; Lee, I.-K.; Randrup, T.: Reconstruction of kinematic surface from scattered data, Proc. of Symposium on Geodesy for Geotechnical and Structural Engineering, 1998, 483–488.

[19] Pratt, V.: Direct least-squares fitting of algebraic surfaces, SIGGRAPH Comput. Graph.,

21(4), 1987, 145–152. DOI: 10.1145/37402.37420.

[20] Randrup, T.: Approximation by cylinder surfaces, Computer Aided Design, 30, 1998, 807–812. DOI: 10.1016/S0010-4485(98)00038-4.

[21] Rangarajan, P.; Kanatani, K.; Niitsuma, H.; Sugaya, Y.: Hyper Least Squares and Its Applications, International Conference on Pattern Recognition, 2010, 5–8. DOI: 10.1109/ICPR.2010.10.

[22] Sampson, P.D.: Fitting conic sections to “very scattered” data: An iterative refinement of the Bookstein algorithm, Computer Graphics and Image Processing, 18(1), 1982, 97–108. DOI: 10.1016/0146-664X(82)90101-0.

[23] Taubin, G.: Estimation Of Planar Curves, Surfaces And Nonplanar Space Curves Defined By Implicit Equations, With Applications To Edge And Range Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 13, 1991, 1115–1138. DOI: 10.1109/34.103273.

[24] Várady, T.; Martin, R.; Cox, J.: Reverse Engineering of Geometric Models - An Introduction, Computer Aided Design, 29, 1997, 255–268. DOI: 10.1016/S0010-4485(96)00054-1.