

# Line Geometry for 3D Shape Understanding and Reconstruction

Helmut Pottmann, Michael Hofer, Boris Odehnal, and Johannes Wallner

Technische Universität Wien, A 1040 Wien, Austria.

{pottmann,hofer,odehnal,wallner}@geometrie.tuwien.ac.at

**Abstract.** We understand and reconstruct special surfaces from 3D data with line geometry methods. Based on estimated surface normals we use approximation techniques in line space to recognize and reconstruct rotational, helical, developable and other surfaces, which are characterized by the configuration of locally intersecting surface normals. For the computational solution we use a modified version of the Klein model of line space. Obvious applications of these methods lie in Reverse Engineering. We have tested our algorithms on real world data obtained from objects as antique pottery, gear wheels, and a surface of the ankle joint.

**Introduction.** The geometric viewpoint turned out to be highly successful in dealing with a variety of problems in Computer Vision (see, e.g., [3,6,9,15]). So far mainly methods of analytic geometry (projective, affine and Euclidean) and differential geometry have been used. The present paper suggests to employ *line geometry* as a tool which is both interesting and applicable to a number of problems in Computer Vision. Relations between vision and line geometry are not entirely new. Recent research on generalized cameras involves sets of projection rays which are more general than just bundles [1,7,18,22]. A beautiful exposition of the close connections of this research area with line geometry has recently been given by T. Pajdla [17].

The present paper deals with the problem of *understanding and reconstructing 3D shapes from 3D data*. The data are assumed to be of a surface-like nature — either a cloud of measurement points, or another 3D shape representation such as a triangular mesh or a surface representation in parametric or implicit form — and we assume that we are able to obtain a discrete number of points on the surface and to estimate surface normals there. We are interested in classes of surfaces with special properties: planar, spherical and cylindrical surfaces, surfaces of revolution and helical surfaces; and the more general surface classes of *canal*, *pipe*, and *developable* surfaces. For applications in CAD/CAM it is essential that such special shapes are not represented by freeform surfaces without regard to their special properties, but treated in a way more appropriate to their ‘simple’ nature.

Line geometry enters the problem of object reconstruction from point clouds via the surface normals estimated at the data points. In fact, modern 3D photography and the corresponding software delivers such normals together with the data points. It turns out that the surface classes mentioned above can be

characterized in terms of their surface normals in an elegant, simple and computationally efficient way. Appropriate coordinates for lines (which yield a *model of line space* as a certain 4-dimensional manifold embedded in  $\mathbb{R}^6$ ) will allow to classify point clouds and their normals (i.e., the so-called ‘normal congruence’) by means of tools such as principal component analysis.

**Previous work.** There is a vast body of literature on surface reconstruction. Since we are interested in the reconstruction of special surfaces, we do not review the part of literature which deals with the reconstruction of triangular meshes or general freeform representations. In Computer Vision, recognition and reconstruction of *special shapes* is often performed by methods related to the *Hough transform*. Originally designed for the detection of straight lines in 2D, it received much attention and has been generalized so as to be able to detect and reconstruct many other shapes (see, e.g., [11,14]). Pure Hough transform methods work in ‘spaces of shapes’ and quickly lead to high dimensions and reduced efficiency. In order to avoid these problems, such tools are sometimes augmented by methods from constructive geometry. This approach is already close to techniques invented by the CAD community, which use geometric characterizations of surfaces (e.g. by means of the Gaussian image) for data segmentation and the extraction of special shapes (see the survey [24]). Many papers deal with axis estimation of rotational surfaces, like [25]. See [8] for an overview on the Hough transform, the RANSAC principle, and the least squares approach. In the present paper, however, rotational surfaces occur only as a special case.

The use of line geometry for surface reconstruction has been introduced by [20]. There cylinders, surfaces of revolution and helical surfaces are recognized by the fact that their surface normals are contained in a so-called linear line complex. In particular surfaces which can be moved within themselves in more than one way (right circular cylinders, spheres, planes) are detected. The technique is extendable to surfaces which may be locally well approximated by the surface types mentioned above [2,13,21].

**Contributions of the present paper.** Inspired by the line geometric work on reverse engineering of special shapes, our paper presents a broader line geometric framework for the solution of problems in 3D shape understanding, segmentation and reconstruction:

- We discuss a point model for line space, namely a certain 4-dimensional algebraic manifold  $M^4$  of order 4 in  $\mathbb{R}^6$ . It is better suited for line geometric approximation problems than the classical Klein model or the model used in [20], which is limited to linear line complexes.
- This point model makes it possible to perform the basic shape recognition tasks via principal component analysis (PCA) of a point cloud (contained in  $M^4$ ), which represents the estimated surface normals of the input shape. This procedure is further improved here and unlike [20] is stable in all special cases.
- The idea of looking for surface normals which *intersect* makes it possible to apply line-geometric methods to the problem of recognition and reconstruction

of *canal surfaces* (which are the envelope of a one-parameter family of spheres), and of *moulding surfaces*. The latter denotes a certain class of sweep surfaces which contains the developable surfaces and the pipe surfaces.

- The segmentation of composite surfaces into their ‘simple’ parts is addressed here in so far as recognition of surface type is essential for segmentation. Our algorithms may be included as a ‘black box’ into a segmentation algorithm.

## 1 The 4-Dimensional Manifold of Lines in $\mathbb{R}^6$

This paragraph discusses a computationally attractive point model of (i.e., coordinates in) the 4-dimensional manifold of straight lines in space. It is closely related to the classical Klein quadric (i.e., Plücker coordinates). We think of an *oriented* line  $L$  as one equipped with a unit vector  $\mathbf{l}$  indicating the direction of the line — so that there are two oriented lines for each line of space. Then  $L$  is determined by  $\mathbf{l}$  and the moment vector  $\bar{\mathbf{l}}$ , which is computed by means of an arbitrary point  $\mathbf{x}$  on  $L$  as  $\bar{\mathbf{l}} = \mathbf{x} \times \mathbf{l}$ .  $\bar{\mathbf{l}}$  is independent of the choice of  $\mathbf{x}$  on  $L$ . The six numbers  $(\mathbf{l}, \bar{\mathbf{l}})$  are called *normalized Plücker coordinates* of  $L$ . ‘Normalized’ means that  $\|\mathbf{l}\| = 1$ ; further, they satisfy the orthogonality condition  $\mathbf{l} \cdot \bar{\mathbf{l}} = 0$ . Conversely, any two vectors  $\mathbf{l}, \bar{\mathbf{l}} \in \mathbb{R}^3$  which satisfy these two conditions determine a unique oriented straight line  $L$  in  $\mathbb{R}^3$ , which has  $(\mathbf{l}, \bar{\mathbf{l}})$  as its normalized Plücker coordinates.

If we do not distinguish between the two opposite orientations of the same line, we may use all multiples of the pair  $(\mathbf{l}, \bar{\mathbf{l}})$  as coordinates of a line. Of course, we still have the condition  $\mathbf{l} \cdot \bar{\mathbf{l}} = 0$ . Such homogeneous coordinate vectors of lines represent those points of five-dimensional projective space which are contained in the *Klein quadric*  $M_2^4$  given by the equation  $(\mathbf{x}, \bar{\mathbf{x}}) \in M_2^4 \Leftrightarrow \mathbf{x} \cdot \bar{\mathbf{x}} = 0$ . This interpretation of lines is well studied in classical geometry, see [21].

The present paper pursues the following approach, which is closely related to the Klein quadric. We use only normalized coordinate vectors, and so we identify an oriented line with the point  $(\mathbf{l}, \bar{\mathbf{l}})$  in six-dimensional Euclidean space  $\mathbb{R}^6$ . In this way we obtain a mapping  $\alpha$  of oriented lines to points of a 4-dimensional manifold  $M^4 \subset \mathbb{R}^6$ .  $M^4$  is algebraic of degree 4, and is the intersection of the cylinder  $Z^5$  and the cone  $\Gamma^5$  defined by

$$Z^5 : \mathbf{x}^2 = 1, \quad \Gamma^5 : \mathbf{x} \cdot \bar{\mathbf{x}} = 0.$$

We use the Euclidean distance of points in  $\mathbb{R}^6$  in order to measure distances between oriented lines  $G, H$ : If  $G\alpha = (\mathbf{g}, \bar{\mathbf{g}})$  and  $H\alpha = (\mathbf{h}, \bar{\mathbf{h}})$ , then

$$d(G, H)^2 = (\mathbf{g} - \mathbf{h})^2 + (\bar{\mathbf{g}} - \bar{\mathbf{h}})^2. \tag{1}$$

The reasons why we prefer this distance function are the following: On the one hand, it is quadratic and thus lends itself to minimization. On the other hand, in a neighbourhood of the origin of the coordinate system, (1) models a distance of lines which is in accordance with visualization. The slight drawback that this is no longer the case in regions far away from the origin is in fact not important in most applications, as such applications very often define a natural *region of interest*, where we can put the origin of our coordinate system.

*Remark 1.* Another method for introducing coordinates and measuring distances for lines, which has been used in the past, is to fix two parallel planes and describe a line by the two intersection points with that plane (cf. [21]). This leads to simpler formulae and a region of interest which is bounded by the two fixed planes (in our case, this region is a sphere centered in the origin).

## Classification of Surfaces by Normal Congruences

The set of normals of a surface is called its *normal congruence*. Some surface types are easily recognized from properties of their normal congruence: The normals of a sphere pass through its center (they constitute a bundle with a finite vertex), and the normals of a plane are parallel (they constitute a parallel bundle, with vertex at infinity). These are the simplest examples; there are however other interesting and practically important classes of surfaces which are nicely characterized by their normal congruence. These classical results are basic to shape understanding and reconstruction algorithms and thus we summarize them in this section.

A uniform motion in 3-space, composed of a uniform rotation of unit angular velocity about an axis and a translation of constant speed  $p$  along this axis is called a *helical motion of pitch  $p$* . If we choose a Cartesian coordinate system with the  $x_3$ -axis being the axis of rotation, then the point  $(x_1, x_2, x_3)$  will move according to

$$x_1(t) = x_1 \cos t - x_2 \sin t, x_2(t) = x_1 \sin t + x_2 \cos t, \quad x_3(t) = x_3 + pt. \quad (2)$$

In the case  $p = 0$  we have a uniform *rotation*. For  $p \rightarrow \infty$  we get, in the limit, a uniform *translation*. A surface swept by a curve under a helical motion is called *helical surface*. As special and limit cases for  $p = 0$  and  $p = \infty$  we get *surfaces of revolution* and *cylinder surfaces*, respectively. We say that these surfaces are *kinematically generated*. However if we speak of a helical surface we always mean *part* of a complete helical surface as defined above, and analogously for other adjectives, like rotational/cylindrical/spherical/planar. Closely related to helical motions are *linear complexes*, which are certain three-parameter sets of lines defined by linear equations, and which are discussed in more detail in [21]. A line  $L$  with Plücker coordinates  $(\mathbf{l}, \bar{\mathbf{l}})$  is contained in the complex  $\mathcal{C}$  with coordinates  $(\mathbf{c}, \bar{\mathbf{c}})$  if and only if

$$L \in \mathcal{C} \iff \bar{\mathbf{c}} \cdot \mathbf{l} + \mathbf{c} \cdot \bar{\mathbf{l}} = 0. \quad (3)$$

Obviously the lines of the complex  $\mathcal{C}$  defined by (3) correspond to those points of  $\mathbb{R}^6$  which are both contained in  $M^4$  and fulfill (3), i.e., they lie in  $M^4$  and in the hyperplane  $H^5 : \bar{\mathbf{c}} \cdot \mathbf{x} + \mathbf{c} \cdot \bar{\mathbf{x}} = 0$ . Note that  $H^5$  passes through the origin. The set  $\mathcal{C}\alpha$  is a certain 3-dimensional manifold. A linear complex  $\mathcal{C}$  is called *singular* if  $\mathbf{c} \cdot \bar{\mathbf{c}} = 0$  (then there is a line  $A$  with  $A\alpha = (\mathbf{c}, \bar{\mathbf{c}})$ , and  $\mathcal{C}$  consists of all lines which intersect  $A$ ).

The sets of lines which correspond to the intersections of  $M^4$  with other  $d$ -dimensional subspaces  $H^d$  also play important roles and have special names.<sup>1</sup> These line sets are employed by the following classification result (see e.g. [21])

**Proposition 1.** *The normals of a given surface are contained in a linear complex  $\mathcal{C} = (\mathbf{c}, \bar{\mathbf{c}})$  if and only if the surface is helical or rotational or cylindrical. The complex  $\mathcal{C}$  is regular for a helical surface and singular otherwise. The axis of  $\mathcal{C}$  is at infinity (i.e.,  $\mathbf{c} = 0$ ) for cylindrical surfaces. Parts of surfaces which are kinematically generated in more than one way are characterized as follows: The normals of a right circular cylinder are contained in a linear congruence; the normals of spherical and planar surfaces are contained in a bundle.*

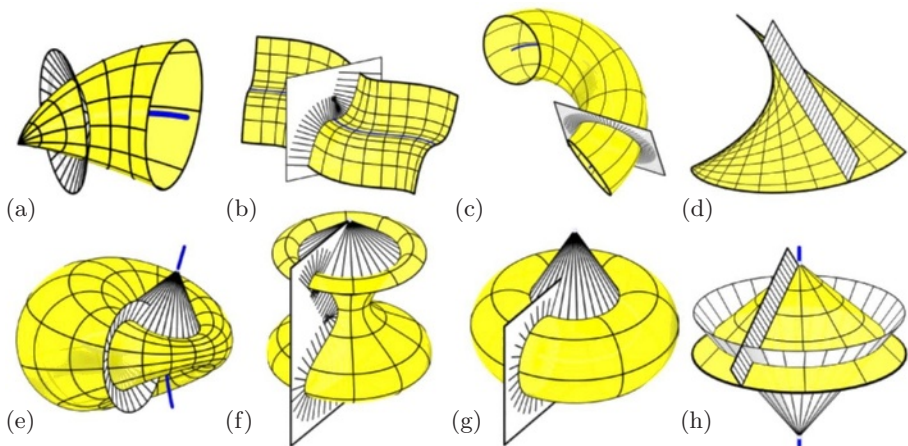
**A list of surface classes and their normals.** Surfaces of revolution (Fig. 1f) are envelopes of a one-parameter family of spheres, the centers of which lie on the axis of rotation. More generally the envelope of a smooth one-parameter family of spheres is called a *canal surface* (see Fig. 1a). The midpoints of these spheres form the surface's *spine curve*. If a sphere touches the envelope surface it does so along a circle.

If the spheres are of constant radius, one obtains a *pipe surface* (see Fig. 1c). Pipe and canal surfaces appear in CAD for example as blending surfaces. Obviously pipe surfaces play an important role in places like oil platforms and refineries. As-built reconstructions of these shapes from 3D data have a number of applications and thus this topic already received some attention (see e.g. [16]). Viewpoint-invariant recognition of canal surfaces from images has been addressed e.g. by Pillow et al. [19].

The envelope of a sphere under rotation about an axis (not through the sphere's center) is a *torus* (Fig. 1g), which may be generated as a canal surface also in another way. In general, surfaces, which are canal surfaces in two ways, are called *Dupin cyclides* (see Fig. 1e). They are well known algebraic surfaces of degree 3 or 4, and may be described as images of tori under inversion. Their use in geometric modelling is described e.g. in a survey article by W. Degen [4].

Pipe surfaces are also traced out by a circle  $\mathbf{p}$  which moves such that its center runs along the spine curve  $\mathbf{s}(t)$ , and its plane  $U(t)$  is always orthogonal to that curve. Rotation about the spine curve has no influence on the resulting surface, but we would like to specify the movement in a more precise way: We assume that a coordinate frame attached to the circle moves such that its angular velocity is minimal (i.e., no unnecessary rotations about the spine curve occur; see e.g. [12]). This is the case of the *rotation-minimizing frame*, where trajectories

<sup>1</sup> For  $d = 4$  we get *linear line congruences* (e.g. a *hyperbolic* linear congruence consists of all lines which meet two given lines). The case  $d = 3$  consists of three subcases: (i) the *bundle* of lines incident with a given point; (ii) the *field* of lines contained in a given plane; and (iii) a *regulus*, which is one of the two one-parameter families of lines contained in a ruled quadric. In the cases (i) and (ii)  $H^3$  is contained in  $\Gamma^5$ ; whereas in case (iii) the set  $\Gamma^5 \cap H^3$  is a quadratic cone. For  $d = 2$  it may happen that  $H^2$  is contained in  $\Gamma^5$ , in which case the corresponding line set is a *pencil*, consisting of lines in a fixed plane  $U$  which pass through a point  $p \in U$ .



**Fig. 1.** (a) Canal surface, (b) Moulding surface, (c) Pipe surface, (d) Developable surface, (e) Dupin cyclide, (f) Surface of revolution, (g) Torus, (h) Cone of revolution. Both the surface and the set  $I(\mathbf{x})$  are shown.

of points in the plane  $U(t)$  are orthogonal to  $U(t)$ . A surface generated by any profile curve  $\mathbf{p}$  (not necessarily a circle) during this motion is called a *moulding surface* (see Fig. 1b). It has the property that all positions  $\mathbf{p}(t)$  of the profile curve are principal curvature lines.

If the profile curve  $\mathbf{p}$  is a straight line, the moulding surface is *developable* and can be mapped isometrically into the Euclidean plane. All points of a *generator* or *ruling*  $\mathbf{p}(t)$  have the same tangent plane. Thus, such a surface is also the envelope of a one-parameter family of planes. Special cases are cones and cylinders, but in general the lines  $\mathbf{p}(t)$  are tangent to some space curve (the *line of regression*), which is a singular curve on the surface (see Fig. 1d).

Let us now describe how to characterize these surfaces via their normal congruences. It is well known that the normal congruence of any surface  $\Phi$ , which is not planar or spherical, can be decomposed into two families of developable surfaces. These are formed by the surface normals along the principal curvature lines. For canal surfaces one of these families consists of cones (or possibly cylinders) of revolution, and the vertices of these cones lie on the spine curve (see Fig. 1a). For pipe surfaces, these cones have an opening angle of 180 degrees (i.e., they are line pencils; see Fig. 1c). For developable surfaces the cones become cylinders of infinite radius (i.e., they are pencils of parallel lines; see Fig. 1d).

In order to exploit these facts computationally (cf. Sec. 3), we define the set  $I(\mathbf{x})$  of *locally intersecting normals* at a point  $\mathbf{x}$  of a surface  $\Phi$ : We pick a neighborhood  $N(\mathbf{x}) \subset \Phi$  of  $\mathbf{x}$  and look for points  $\mathbf{y}$  in  $N(\mathbf{x})$  whose normals intersect the normal at  $\mathbf{x}$ . These normals form the set  $I(\mathbf{x})$ . Interesting are those cases where we can identify two specially shaped components of  $I(\mathbf{x})$ .

**Proposition 2.** *The following table enumerates special surface classes and the shapes of the two components of  $I(\mathbf{x})$  for any surface point  $\mathbf{x}$  (cf. Fig. 1).*

	<i>cone</i>	<i>planar</i>	<i>pencil</i>	<i>parallel</i>
<i>arbitrary</i>	<i>canal surface</i>	<i>moulding surface</i>	<i>pipe surface</i>	<i>developable surface</i>
<i>cone</i>	<i>cyclide</i>	<i>rotational surface</i>	<i>torus</i>	<i>rotational cone</i>

Conversely, if for all  $\mathbf{x}$  we can identify two components of  $I(\mathbf{x})$  as listed, the surface is of the associated type.

Here, ‘arbitrary’ means that no condition is imposed on this component, ‘parallel’ means that the lines of the component are parallel, and ‘planar’ means that the lines are contained in some plane. The table can be used as follows. If, for example, we find that  $I(\mathbf{x})$  contains, for all  $\mathbf{x}$ , a parallel component, and a conical component, the surface is a cone of revolution (as a side-effect the conical component itself must be part of a cone of revolution; see Fig. 1h). Note that the second row of the table, except for the cyclides, contains only rotational surfaces already addressed by Prop. 1. This is the reason why adding new rows in the table above does not lead to anything new.

## 2 Principal Component Analysis (PCA) on the Surface Normals for Surface Recognition and Reconstruction

**Basic algorithm and PCA.** We would like to approximate a set of lines  $L_i$ ,  $i = 1, \dots, N$  by a linear line complex  $\mathcal{C}^*$  with coordinates  $(\mathbf{c}^*, \bar{\mathbf{c}}^*)$ . It will be sufficient to consider one orientation for each line, so we compute normalized Plücker coordinates  $L_i\alpha = (\mathbf{l}_i, \bar{\mathbf{l}}_i) \in \mathbb{R}^6$ .

A first approximation method is to compute the unique hyperplane  $H^5$  through the origin which minimizes the squared sum of distances from the points  $L_i\alpha$ . With  $H(\mathbf{x}, \bar{\mathbf{x}}) = \bar{\mathbf{c}} \cdot \mathbf{x} + \mathbf{c} \cdot \bar{\mathbf{x}}$ , the Euclidean distance of a point  $(\mathbf{p}, \bar{\mathbf{p}})$  from the hyperplane  $H^5 : H(\mathbf{x}, \bar{\mathbf{x}}) = 0$  simply equals  $H(\mathbf{p}, \bar{\mathbf{p}})$ , if

$$\mathbf{c}^2 + \bar{\mathbf{c}}^2 = 1. \tag{4}$$

Thus we find  $H^5$  by minimizing  $F(\mathbf{c}, \bar{\mathbf{c}}) = \sum_{i=1}^N [\bar{\mathbf{c}} \cdot \mathbf{l}_i + \mathbf{c} \cdot \bar{\mathbf{l}}_i]^2$  under the constraint (4). We consider both  $\mathbf{c}$  and  $\bar{\mathbf{c}}$  as column vectors and write  $F$  in the form

$F(\mathbf{c}, \bar{\mathbf{c}}) = [\mathbf{c}^T \ \bar{\mathbf{c}}^T] \cdot A \cdot \begin{bmatrix} \mathbf{c} \\ \bar{\mathbf{c}} \end{bmatrix}$ , with a certain symmetric  $6 \times 6$ -matrix  $A$ . It follows easily from the Lagrangian multiplier rule that the minimizer  $(\mathbf{c}^*, \bar{\mathbf{c}}^*)$  is given by an eigenvector of  $A$ , which belongs to  $A$ ’s smallest eigenvalue  $\lambda^*$ , and which is normalized according to (4) (in this case we also have  $F(\mathbf{c}^*, \bar{\mathbf{c}}^*) = \lambda^*$ ).

According to Prop. 1, to a linear complex  $\mathcal{C}$  belongs a helical motion. Its pitch  $p$  and the Plücker coordinates  $(\mathbf{a}, \bar{\mathbf{a}})$  of its axis can be computed by  $p = (\mathbf{c}^* \cdot \bar{\mathbf{c}}^*)/(\mathbf{c}^*)^2$ ,  $(\mathbf{a}, \bar{\mathbf{a}}) = (\mathbf{c}^*, \bar{\mathbf{c}}^* - p\mathbf{c})$  (cf. [21]). A small pitch  $p$  indicates that the lines  $L_i$  belong to a surface of revolution. We may then repeat the approximation process with the additional side condition that  $\mathbf{c} \cdot \bar{\mathbf{c}} = 0$  (according to Prop. 1). Large values of  $p$  indicate that the lines  $L_i$  belong to a cylinder. This circumstance is also detected by the fact that all  $L_i$  are orthogonal to a fixed line. If we decide we want the data to be approximated by a cylinder, we may repeat the minimization with the additional side condition that  $\mathbf{c} = 0$ .

So far we have shown how to find the *motion* which generates the surface. For the actual computation of a *profile curve* we use methods available in the literature [20,21,23,24].

*Two small eigenvalues* of  $A$  indicate that the points  $L_i\alpha \in \mathbb{R}^6$  are almost contained in a subspace of dimension 4. The normals  $L_i$  are then almost contained in a linear congruence. By Prop. 1 this means that the normals belong to a *right circular cylinder*. As a side-effect, all of them must now intersect the cylinder’s axis orthogonally.

*Three small eigenvalues* of  $A$  imply that the points  $L_i\alpha$  are almost contained in a 3-dimensional subspace, which indicates a spherical or planar surface. Note however that planes are also detected by a Gaussian image [24] or a Hough transform for planes in 3-space.

*Remark 2.* The procedure above is a principal component analysis of the set of image points  $L_i\alpha \in \mathbb{R}^6$ . Without going into details we remark that it may be refined in order to be more robust e.g. by incorporating a weighted least squares approximation which iteratively downweights outliers, or the RANSAC principle (see [5] and references therein).

**Refined algorithm.** So far we computed an approximating complex  $\mathcal{C}$  by finding a hyperplane  $H^5$  (carrying  $\mathcal{C}\alpha$ ) such that the sum of squares of distances of the points  $L_i\alpha$  from  $H^5$  is minimized. This is not the same as minimizing distances of the points  $L_i\alpha$  from the set  $\mathcal{C}\alpha = M^4 \cap H^5$ . Actually we should have minimized the *geodesic distance* of the points  $L_i\alpha$  from the set  $\mathcal{C}\alpha = H^5 \cap M^4$  within the manifold  $M^4$ . We will take this into account by considering the angle of intersection between  $H^5$  and  $M^4$ . For a point  $(\mathbf{l}, \bar{\mathbf{l}}) = L\alpha$  in  $M^4$  consider  $M^4$ ’s tangent 4-space  $T^4$  and let  $\phi = \angle(T^4, H)$ .  $d_H$  denotes the distance of  $L\alpha$  from  $H^5$ . Then the value  $d_{\mathcal{C}} = d_H / \sin \phi$  is an estimate for the geodesic distance of  $L\alpha$  from  $\mathcal{C}\alpha = M^4 \cap H^5$ . There is the following result:

**Lemma 1.** *The angle  $\phi$  used in the definition of  $d_{\mathcal{C}}$  is given by  $\cos^2 \phi = (\bar{\mathbf{c}} \cdot \mathbf{l})^2 + (\bar{\mathbf{c}} \cdot \bar{\mathbf{l}}^n + \mathbf{c} \cdot \mathbf{l}^n)^2$ , with  $(\mathbf{l}^n, \bar{\mathbf{l}}^n) = 1/(\mathbf{l}^2 + \bar{\mathbf{l}}^2) \cdot (\mathbf{l}, \bar{\mathbf{l}})$ .*

*Proof.* We note that the 2-dimensional normal space to  $M^4 = Z^5 \cap \Gamma^5$  in the point  $L\alpha = (\mathbf{l}, \bar{\mathbf{l}})$  is spanned by the normal vectors  $\mathbf{a}$  to  $Z^5$  and  $\mathbf{b}$  to  $\Gamma^5$ . These are given by  $\mathbf{a} = (\mathbf{l}, \mathbf{o})$  and  $\mathbf{b} = (\bar{\mathbf{l}}, \mathbf{l})$ , respectively. Note that  $\mathbf{a} \cdot \mathbf{b} = 0$ , and that  $\|\mathbf{a}\| = 1$ , whereas  $\mathbf{b}$  is not yet a unit vector in  $\mathbb{R}^6$ . In order to achieve this, we define  $\mathbf{l}^n$  and  $\bar{\mathbf{l}}^n$  as above, and replace  $\mathbf{b}$  by  $\mathbf{b}^n = (\bar{\mathbf{l}}^n, \mathbf{l}^n)$ . Now  $(\mathbf{c}, \bar{\mathbf{c}})$  is a normal vector of  $H^5$  and satisfies (4). It follows that  $\cos^2 \phi = (\mathbf{a} \cdot \mathbf{n})^2 + (\mathbf{b}^n \cdot \mathbf{n})^2$ , and the result follows by expanding the definitions.  $\square$

Minimizing the distances  $d_{\mathcal{C}}$  instead of orthogonal distances  $d_H$  means minimizing the function

$$F_1(\mathbf{c}, \bar{\mathbf{c}}) = \sum_{i=1}^N \frac{(\bar{\mathbf{c}} \cdot \mathbf{l}_i + \mathbf{c} \cdot \bar{\mathbf{l}}_i)^2}{1 - (\bar{\mathbf{c}} \cdot \mathbf{l}_i)^2 - (\bar{\mathbf{c}} \cdot \bar{\mathbf{l}}_i^n + \mathbf{c} \cdot \mathbf{l}_i^n)^2}$$

under the side-condition (4). This is a nonlinear optimization problem which we solve via a weight iteration: We initialize the algorithm with the minimizer

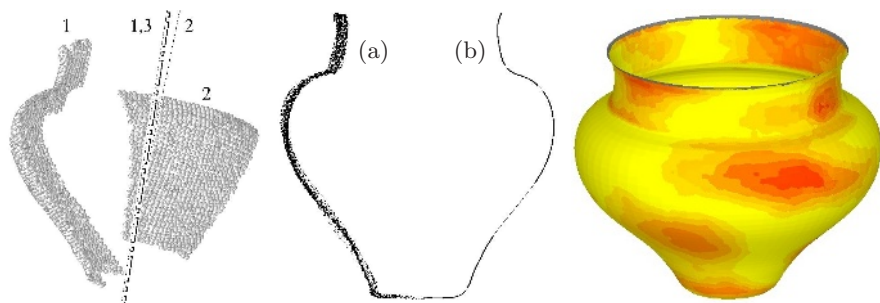


of  $F$ . Then, using the solution  $(\mathbf{c}^*, \bar{\mathbf{c}}^*)$  from the previous step, we minimize  $F_2(\mathbf{c}, \bar{\mathbf{c}}) = \sum_{i=1}^N w_i (\bar{\mathbf{c}} \cdot \mathbf{l}_i + \mathbf{c} \cdot \bar{\mathbf{l}}_i)^2$ , with  $1/w_i = 1 - (\bar{\mathbf{c}}^* \cdot \mathbf{l}_i)^2 - (\bar{\mathbf{c}}^* \cdot \bar{\mathbf{l}}_i^n + \mathbf{c}^* \cdot \mathbf{l}_i^n)^2$ . This procedure is motivated by the fact that different angles  $\phi_i$  at points  $L_i\alpha$  give different weights to distances  $d_{H,i}$  in the minimization of  $F$ . The unbalanced situation is corrected via the weight iteration described above. The iteration terminates if the change in  $F_1$  in one iteration step is below some given threshold.

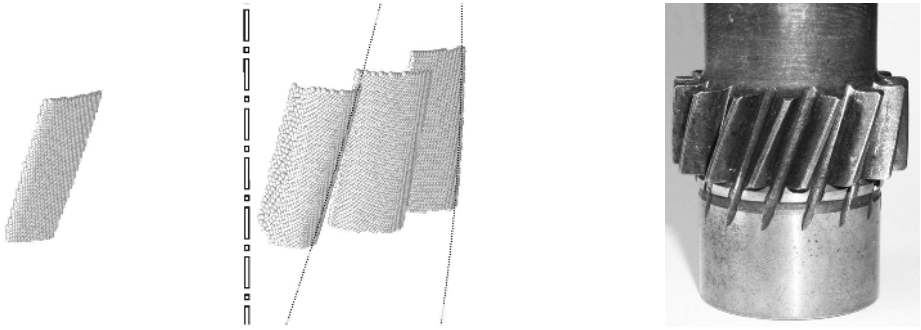
### 3 Examples

We used a Minolta VI-900 3D laser scanner to obtain point data and tested the effectiveness of our algorithms on them. Point clouds were thinned if necessary and triangulated. Surface normal vectors have been estimated by local regression planes. In Figures 2, 3, and 4, data points have been rendered as small balls.

**Surface type by PCA: Rotational surfaces.** Fig. 2 shows the procedure of reconstructing the axis and meridian curve of a near-rotational surface. Data have been obtained by scanning the outer surface of a late Hallstatt pottery object manufactured in approx. 550 B.C. without the use of a pottery wheel. The procedure of Sec. 2 has been applied to estimate a linear line complex which fits the surface normals best. With  $d$  as the diameter of the point cloud, we found  $|p| \ll d$ , so the data points in question may be approximated by a surface of revolution. The meridian curve (Fig. 2, center) was found by rotating the point cloud into a plane through the axis of rotation (a) and approximating (a) by a smooth curve (b). The deviations of the original cloud from the reconstructed ideal surface of revolution are shown by Fig. 2, right. Figure 2, left, shows how the estimated axis (3) changes if we use only parts of the data available (dotted lines 1,2). As is to be expected, accuracy decreases if we use small meridian strips.



**Fig. 2.** Reconstruction of axis and meridian curve of a pottery object. Left: Axes reconstructed from parts (1,2) and from entire object (3). Center: Reconstructed meridian curve. Right: Colour coded deviations of original from reconstructed ideal surface (maximum deviation: 2.6% of diameter).



**Fig. 3.** Left: Reconstruction of axis and two helical paths for helical gears. Right: Photo of helical gears.

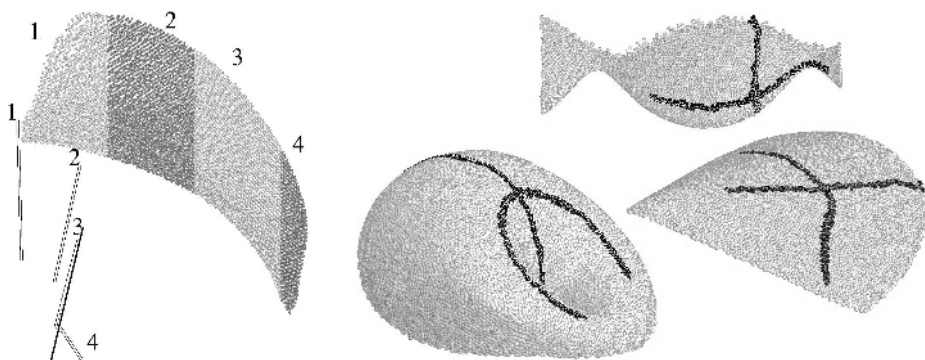
**Surface type by PCA: Helical surfaces.** Fig. 3, left shows a scan of four teeth of a helical gear wheel as an example of a surface which is known to be helical. The point data have been obtained in one pass of scanning. The underlying helical motion, defined by axis and parameter, has been reconstructed in the expected way.

**Surface type by PCA: Freeform surfaces.** The human body does not possess mathematically exact helical surfaces. However, we studied the following interesting example: Fig. 4, left, shows a scan of a *trochlea tali*, i.e., the distal interface of the ankle joint. The closest helical surface computed by the algorithm of Sec. 2 is not a surface of revolution. This piece of information is important when studying the relative motion of the talus (ankle bone) with respect to the tibiofibular (lower leg) system. We might ask whether the trochlea tali is close enough to a mathematical helical surface to be called helical. This turns out to be not the case, as can be seen from computing the closest helical surfaces to surface strips and comparing the results. The axes corresponding to four strips together with the axis corresponding to the entire data set are indicated in Fig. 4.

**Surface type by  $I(\mathbf{x})$ .** In order to recognize even more surface classes, we consider the sets  $I(\mathbf{x})$  of locally intersecting normals. We compute them as follows: The surface normal spanned by the point  $\mathbf{x}'$  and the vector  $\mathbf{n}'$  is contained in  $I(\mathbf{x})$ , if and only if  $\det(\mathbf{x}' - \mathbf{x}, \mathbf{n}, \mathbf{n}') = 0$  ( $\mathbf{n}$  being the normal vector at  $\mathbf{x}$ ). The discrete version of this is that the normals at the endpoints of an edge of a triangulation of our point cloud are in  $I(\mathbf{x})$ , if the determinant mentioned above changes its sign along that edge.

Our examples — scans of plaster and wood models — are shown by Fig. 4, where the data points whose normals contribute to  $I(\mathbf{x})$  are indicated in black. We can see that  $I(\mathbf{x})$  consists of two components, one of which may be an arc (in case of a canal surface) or a straight line (in case of a developable surface).

In order to apply Prop. 2, we have to test whether one or more of the two components of  $I(\mathbf{x})$  are conical or planar. If  $L_i = (\mathbf{l}_i, \bar{\mathbf{l}}_i)$  ( $i = 1, \dots, N$ ) are the



**Fig. 4.** Left: Computation of axes of nearest helical surfaces for strips 1–4 and also the entire data set (shown in bold) of a trochlea tali. Right: The sets  $I(\mathbf{x})$  for, counting clockwise from top, a surface of revolution, the developable *oid*, and for the plaster model of the Dupin cyclide shown in [10], Fig. 229a.

surface normals of such a component, we have to find either a point incident with all of them or a plane which contains them. There are several ways to do this. One would be to consider the lines  $L_i$  as surface normals and to try to reconstruct the corresponding surface type according to Sec. 2. This, however, would lead to numerical difficulties due to thinness of data. Therefore we recognized conical components with vertex  $\mathbf{s}$  in the following way: Incidence of the point  $\mathbf{s}$  with the line  $(\mathbf{l}_i, \bar{\mathbf{l}}_i)$  is characterized by  $\bar{\mathbf{l}}_i = \mathbf{s} \times \mathbf{l}_i$ . Thus finding a point  $\mathbf{s}$  ‘as incident as possible’ with the lines  $L_i$  means minimizing

$$G(\mathbf{x}) = \sum w_i (\bar{\mathbf{l}}_i - \mathbf{s} \times \mathbf{l}_i)^2, \quad (w_i \geq 0) \tag{5}$$

with weights  $w_i$  which determine the influence of the single lines. Minimizing (5) is standard, and iteratively downweighting outliers makes this method stable. Comparison of the minimal value of  $\sqrt{G}$  with the size of the point cloud determines conicality. *Planar* components of  $I(\mathbf{x})$  are detected in a similar way. Note that line pencils are both planar and conical.

The sets  $I(\mathbf{x})$  of Fig. 4 have, counting clockwise from the top, 1, 0, and 2 (non-planar) conical components, whereas the number of planar components equals 1, 1, and 0, respectively. It follows that these surfaces are a rotational surface, a moulding surface, and a Dupin cyclide. Additional information on the parallelity of surface normals along one component of  $I(\mathbf{x})$  shows that the right hand surface shown by Fig. 4 is a developable one.

**Conclusion and Future Research.** We have shown how techniques from classical line geometry can serve for recognizing and reconstructing special surfaces. Other applications, such as ruled surface approximation and the computation of approximating line congruences can benefit from the use of the embedding of line space into  $\mathbb{R}^6$ . Active B-spline curves and surfaces can be used efficiently.

This is a consequence of the fact that finding footpoints on  $M^4$  corresponds to finding the zeros of a fourth order polynomial, and is in fact equivalent to the footpoint problem for planar ellipses.

Future research will also address how to use the bisecting linear complex (cf. [21], p. 166) for checking and improving point correspondences in 3D registration problems, especially when looking for a good initial position.

**Acknowledgements.** This work was supported by the Austrian Science Fund (FWF) under grants P16002 and P15911; and the innovative project ‘3D technology’ of Vienna University of Technology. We want to express our thanks to Landesarchäologie Salzburg for the archaeological pottery artifact.

## References

1. Benosman R., Kang S. B. (editors): *Panoramic Vision: Sensors, Theory and Applications*, Springer Verlag, 2001.
2. Chen H.-Y., Lee I.-K., Leopoldseder S., Pottmann H., Randrup T., Wallner J.: On surface approximation using developable surfaces, *Graphical Models and Image Processing*, 61:110–124, 1999.
3. Cipolla R., Giblin P.: *Visual Motion of Curves and Surfaces*, Cambridge UP, 2000.
4. Degen W.: Cyclides, In *Handbook of Computer Aided Geometric Design*, G. Farin, J. Hoschek and M.-S. Kim, eds., Elsevier, pp. 575–601, 2002.
5. De la Torre, F., Black M. J.: Robust principal component analysis for Computer Vision, *Proc. 8th Int. Conf. on Computer Vision*, pp. 362–369, 2001.
6. Faugeras O.: *Three-dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, MA, 1993.
7. Gupta R., Hartley R.: Linear pushbroom cameras, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):963–975, 1997.
8. Halř R.: *Estimation of the axis of rotation of fragments of archaeological pottery*. Proc. 21st Workshop Austrian Assoc. for Pattern Recognition, Hallstatt 1997.
9. Hartley R., Zisserman A.: *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, Cambridge, UK, 2000.
10. Hilbert D., Cohn-Vossen S., *Anschauliche Geometrie*. Springer, 1932. Reprinted 1996. Translated as: *Geometry and the Imagination*, American Math. Soc. 1999.
11. Illingworth J., Kittler J.: A survey of the Hough transform, *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.
12. Jüttler B., Wagner M.: Kinematics and animation, In *Handbook of Computer Aided Geometric Design*, G. Farin et al. eds., Elsevier, pp. 723–748, 2002.
13. Kós G., Martin R., Várady T.: Recovery of blend surfaces in reverse engineering, *Computer Aided Geometric Design*, 17:127–160, 2000.
14. Leavers V.: Which Hough transform? *CVGIP:Image Understanding* 58:250-264, 1993.
15. Mundy J., Zissermann A.: *Geometric Invariance in Computer Vision*, MIT Press, '92.
16. Navab N.: Canonical representation and three view geometry of cylinders, *Intl. Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Commission III, Vol. XXXIV, Part 3A, pp. 218–224, 2002.
17. Pajdla T.: Stereo geometry of non-central cameras, PhD thesis, CVUT, 2002.
18. Peleg S., Ben-Ezra M., Pritch Y.: Omnistereo: panoramic stereo imaging, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):279-290, 2001.

19. Pillow N., Utcke S., Zisserman A.: Viewpoint-invariant representation of generalized cylinders using the symmetry set, *Image Vision Comput.*, 13:355–365, 1995.
20. Pottmann H., Randrup T., Rotational and helical surface reconstruction for reverse engineering, *Computing*, 60:307–322, 1998.
21. Pottmann H., Wallner J.: *Computational Line Geometry*, Springer 2001.
22. Seitz S. M.: The space of all stereo images, *Proceedings ICCV 2001*, pp. 26–33.
23. Várady T., Benkó T., G. Kós T.: Reverse engineering regular objects: simple segmentation and surface fitting procedures, *Int. J. Shape Modeling*, 4:127–141, 1998.
24. Várady T., Martin R.: Reverse Engineering, In *Handbook of Computer Aided Geometric Design*, G. Farin et al., eds., Elsevier, pp. 651–681, 2002.
25. Willis A., Orriols X., Cooper D.: Accurately Estimating Sherd 3D Surface Geometry with Application to Pot Reconstruction. CVPR Workshop, 2003.