Direct Solutions for Computing Cylinders from Minimal Sets of 3D Points

Christian Beder and Wolfgang Förstner

Institute for Photogrammetry
Bonn University, Germany
{ beder,wf } @ipb.uni-bonn.de

Abstract. Efficient direct solutions for the determination of a cylinder from points are presented. The solutions range from the well known direct solution of a quadric to the minimal solution of a cylinder with five points. In contrast to the approach of G. Roth and M. D. Levine (1990), who used polynomial bases for representing the geometric entities, we use algebraic constraints on the quadric representing the cylinder. The solutions for six to eight points directly determine all the cylinder parameters in one step: (1) The eight-point-solution, similar to the estimation of the fundamental matrix, requires to solve for the roots of a 3rd-order-polynomial. (2) The seven-point-solution, similar to the six-point-solution for the relative orientation by J. Philip (1996), yields a linear equation system. (3) The six-point-solution, similar to the five-point-solution for the relative orientation by D. Nister (2003), yields a ten-by-ten eigenvalue problem. The new minimal five-point-solution first determines the direction and then the position and the radius of the cylinder. The search for the zeros of the resulting 6th order polynomials is efficiently realized using 2D-Bernstein polynomials. Also direct solutions for the special cases with the axes of the cylinder parallel to a coordinate plane or axis are given. The method is used to find cylinders in range data of an industrial site.

1 Introduction

This paper presents direct solutions for estimating circular cylinders from range data both for unconstrained cylinders as well as for cylinders being parallel to a coordinate axis or a coordinate plane. Especially it provides an efficient direct solution for the estimation of a cylinder from the minimum number of five points.

1.1 Motivation

Cylinders play a central role in the representation of the geometry of man made structures such as industrial plants [2, 17], architectures or orthopedy [19]. As-built reconstruction as well as reverse engineering often rely on dense range data. Segmenting point clouds into basic geometric primitives such as planes, cylinders, cones and spheres often is a first step for object recognition.

Such segmentation may use different methods. Classical segmentation methods are based on local surface properties mainly depending on the local orientation and curvature thus address free form surfaces. These algorithms start from an initial surface description, mostly from triangular meshes, cf. the overview of [12] and of [8] where also the detection of breakline is addressed. Hence cylinders are not addressed explicitly. Tensor voting [16] may be used to achieve the transition from the raw 3D-point cloud to an initial surface description.

In case objects are known to consist of basic geometric primitives this knowledge may immediately be used for the segmentation. Random sample consensus (RANSAC) [4, 5] is a commonly applied technique due to its ease in implementation and efficiency to cope with large percentage of outliers. Basic prerequisite for RANSAC is a direct solution for the parameters of the geometric primitive. Roth and Levine [14] collect polynomial bases for extracting geometric primitives from range data. However, general cylinders do not have a simple basis, for which classical direct estimation schemes would work.

Most approaches to extract cylinders from range data use the information about the surface normal. The Gaussian image of the surface, i. e. the mapping of the surface normals to the unit sphere, is a great circle which may be found by RANSAC [2], clustering [19] or Hough-transform [17]. The so-called Blaschke-image of the surface, i. e. the mapping of the surfaces' tangent planes into the projective space $(\mathbf{n}, d)$ with unit normals and distances, eases the identification of multiple primitives [11].

Both analysis methods, surface segmentation as well as cylinder extraction using normals presume the neighborhood relations between the measured points are established. We want to provide direct methods for cylinder extraction which can work on the original 3D-point cloud. As a general cylinder has five degrees of freedom, four for the axis and one for the radius, one needs at least five points to determine the parameters. To our knowledge, no direct solution has been published hitherto in spite of various attempts to express the cylinder constraints on the quadric parameters [18]. As the solution is much more involving than the direct solutions for quadrics we also present solutions with more points, which allows to balance computing time and samples required in RANSAC. Moreover, as in many cases the 3D-data may easily be referred to the plumbline and horizontal and vertical cylinders are quite common we also present the solutions for cylinders with such special orientations.

1.2 General setup

A cylinder can be described by 5 parameters, the 4 parameters for the axis and one for the radius.

In case the cylinder axis is parallel to one coordinate plane, e. g. in case it is horizontal, the number of parameters reduces to 4, the 3 parameters for the axis and the radius.

In case the cylinder axis is parallel to a coordinate axis, we only need 3 parameters, 2 for the position of the axis and one for the radius. If we do not know the coordinate axis, we might check all three.

Each point on the surface yields one constraint. Therefore we have the cases collected in table 1. The number of solutions for the presented algorithms is also given, where we know it. Note, that this number is only an algebraic property of the algorithm and an unique solution is easily obtained for all of the non-minimal cases.

cylinder	# points + (# solutions)
general	**5** (?), 6 (10), 7 (1), 8 (3), 9 (1)
parallel to plane	**4** (3)
parallel to line	**3** (1)

Table 1. Number of parameters for a cylinder (boldface), presented algorithms with maximum number of solutions. The maximum number of solutions for the five point algorithm is not known.

The paper is organized as follows: In section 2 we present direct solutions for cylinders being parallel to an axis or a plane. These results will be be used for the solutions for cylinders with general orientation in section 3, where we present algorithms from 9 down to 5 points. Section 4 shows experiments and results for finding general cylinders in 3D-point-clouds.

2 Cylinders parallel to coordinate axes or planes

2.1 Cylinder parallel to an axis

Without restriction we may assume the axis is parallel to the Z-axis. Then the cylinder is given by

$$(X - s)^2 + (Y - t)^2 - r^2 = 0$$

The cylinder has 3 unknown parameters. The classical solution (cf. [1]) uses the substitution $u = s^2 + t^2 - r^2$. Then the the three parameters s, t and u can be determined from the following three equations

$$X_i^2 + Y_i^2 - 2X_i s - 2Y_i t + u = 0 \qquad i = 1, 2, 3 \tag{1}$$

linear in the parameters, which can be written as

$$\begin{bmatrix} 2X_1 & 2Y_1 & -1 \\ 2X_2 & 2Y_2 & -1 \\ 2X_3 & 2Y_3 & -1 \end{bmatrix} \begin{bmatrix} s \\ t \\ u \end{bmatrix} = \begin{bmatrix} X_1^2 + Y_1^2 \\ X_2^2 + Y_2^2 \\ X_3^2 + Y_3^2 \end{bmatrix} \tag{2}$$

The parameter r can be determined from $r = \sqrt{s^2 + t^2 - u}$.

2.2 Cylinder parallel to a plane

A cylinder parallel to a given plane is described by 4 parameters. Therefore we need four points $\boldsymbol{X}_i$.

Without restriction we may assume the cylinder is parallel to the XY-plane. Then we may describe the cylinder as a reference cylinder parallel to the X-axis

$$(Y' - s)^2 + (Z' - t)^2 - r^2 = 0$$

rotated around the Z axis by some angle κ. Then we first determine a direction $[\cos\kappa, \sin\kappa, 0] = [a, b, 0]$ such that the four points lie on a circle.

The four rotated points are $\boldsymbol{X}'_i = \boldsymbol{R}\boldsymbol{X}_i$ thus

$$\boldsymbol{X}'_i = \begin{bmatrix} aX_i + bY_i \\ -bX_i + aY_i \\ Z_i \end{bmatrix} \qquad a^2 + b^2 = 1$$

Similar to (1) we obtain the constraint $Y_i'^2 + Z_i'^2 - 2Y_i's - 2Z_i't + (s^2 + t^2 - r^2) = 0$ or expanding the rotation

$$(-bX_i + aY_i)^2 + Z_i^2 - 2(-bX_i + aY_i)s - 2Z_it + (s^2 + t^2 - r^2) = 0$$

For the four points we therefore get the linear system

$$\begin{bmatrix} (-bX_1 + aY_1)^2 + Z_1^2 & -2(-bX_1 + aY_1) & -2Z_1 & 1 \\ (-bX_2 + aY_2)^2 + Z_2^2 & -2(-bX_2 + aY_2) & -2Z_2 & 1 \\ (-bX_3 + aY_3)^2 + Z_3^2 & -2(-bX_3 + aY_3) & -2Z_3 & 1 \\ (-bX_4 + aY_4)^2 + Z_4^2 & -2(-bX_4 + aY_4) & -2Z_4 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ s \\ t \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The 4 × 4-matrix is singular if the four points are co-circular. The determinant is cubic in a and b, however only containing monomials $[a^3, a^2b, ab^2, b^3, a, b]$. Together with the constraint $a^2 + b^2 = 1$ we obtain 6 solutions for a and b, which pairwise differ by a factor -1, thus represent the same cylinder. Thus we may obtain up to 3 solutions.

An example would be three points in a horizontal triangle and a fourth point not in that height. Then we have three cylinders parallel to the three sides of that triangle.

3 General Cylinders

3.1 Representation of a cylinder

The cylinder is a special 3D-quadric, representable as symmetric and homogeneous matrix C for the surface points with homogeneous coordinates $\mathbf{X}$

$$\mathbf{X}^{\mathsf{T}}\mathsf{C}\mathbf{X} = 0 \tag{3}$$

which fulfills the constraint, that there exists a plane, so that all points $\mathbf{X}$ on the cylinder projected on that plane are co-circular. If this plane is without loss of generality the XY-plane, this condition can be expressed by

$$(X' - s)^2 + (Y' - t)^2 - r^2 = 0 \tag{4}$$

for some s, t and r, or in terms of the cylinder representation

$$\mathsf{C}' = \lambda \begin{bmatrix} \mathsf{D}' & \boldsymbol{d}' \\ \boldsymbol{d}'^{\mathsf{T}} & -r^2 \end{bmatrix}$$

with

$$\mathsf{D}' = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,0 \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{d}' = \begin{bmatrix} -s \\ -t \\ 0 \end{bmatrix}$$

Because the projection plane is in general unknown, one has to allow a spatial motion

$$\mathsf{M} = \begin{bmatrix} \mathsf{R} & \boldsymbol{t} \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix}$$

to be applied, so that one obtains the general cylinder as

$$\begin{aligned} \mathsf{C} &= \mathsf{M}^{-T}\mathsf{C}'\mathsf{M}^{-1} \\ &= \lambda \begin{bmatrix} \mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}} & -\mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}}\boldsymbol{t} + \mathsf{R}\boldsymbol{d}' \\ -\boldsymbol{t}^{\mathsf{T}}\mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}} + \boldsymbol{d}'^{\mathsf{T}}\mathsf{R}^{\mathsf{T}} & \boldsymbol{t}^{\mathsf{T}}\mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}}\boldsymbol{t} - 2\boldsymbol{t}^{\mathsf{T}}\mathsf{R}\boldsymbol{d}' - r^2 \end{bmatrix} = \begin{bmatrix} \mathsf{D} & \boldsymbol{d} \\ \boldsymbol{d}^{\mathsf{T}} & d \end{bmatrix} \end{aligned}$$

3.2 Constraints on the parameters of a cylinder

One immediately observes, that the matrix D is singular and has two identical eigenvalues, which can be expressed algebraically (cf. [3], p. 254) by the ten equations

$$|\mathsf{D}| = 0 \tag{5}$$

$$2\mathsf{D}\mathsf{D}^{\mathsf{T}}\mathsf{D} - \mathrm{tr}\mathsf{D}\mathsf{D}^{\mathsf{T}}\mathsf{D} = \underset{3\times3}{\mathsf{0}} \tag{6}$$

Note that the second equations yield only 6 independent constraints due to symmetry. Further one can see, that

$$\mathsf{D}\boldsymbol{d} = \lambda^2\mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}}(-\mathsf{R}\mathsf{D}'\mathsf{R}^{\mathsf{T}}\boldsymbol{t} + \mathsf{R}\boldsymbol{d}') = \lambda\boldsymbol{d}$$

thus $\boldsymbol{d}$ is an eigenvector of D yielding the additional three constraints

$$[\boldsymbol{d}]_{\times}\mathsf{D}\boldsymbol{d} = \mathbf{0} \tag{7}$$

and one finally arrives at ten linear independent algebraic constraints (5), (6) and (7).

We now exploit these constraints stepwise.

3.3 Solutions with 9, 8, 7, and 6 points

Solution with 9 points: If one has given 9 points $\mathbf{X}_i, i = 1, ..., 9$ on the cylinder, the constraint (3) is sufficient to solve the problem using a simple singular value decomposition of the homogeneous equation system

$$\mathsf{A}\ \mathrm{vech}\mathsf{C} = [\mathrm{vech}^{\mathsf{T}}(\mathbf{X}_i\mathbf{X}_i^{\mathsf{T}})]\ \mathrm{vech}\mathsf{C} = \mathbf{0}$$

in the ten unknown elements of C (cf. [7], p. 563).

Solution with 8 points: If only 8 points are given, the nullspace resulting from the singular value decomposition of the homogeneous matrix A imposed by constraint (3) is two-dimensional. The solution is thus known to be

$$\mathsf{C} = x\mathsf{C}_1 + \mathsf{C}_2$$

for some scalar x, where the two matrices C_i result from the nullspace of A. Analogous to the well-known 7-point-algorithm for computing the fundamental matrix (cf. [7], p. 264), one picks any of the ten constraints (e.g. (5)), which are all polynomials of degree three in x and solves for the roots yielding up to three solutions.

Solution with 7 points: Again constraint (3) is used to compute the now three-dimensional nullspace, in which the solution is found:

$$\mathsf{C} = x\mathsf{C}_1 + y\mathsf{C}_2 + \boldsymbol{C}_3$$

Following the approach of [13], x and y can be found using the ten constraints (5), (6) and (7), which are all polynomials of degree three in x and y. More specifically the ten polynomials are written as homogeneous equation system in the monomials

$$\mathsf{N}\begin{bmatrix}x^3 \; x^2y \; xy^2 \; y^3 \; x^2 \; xy \; y^2 \; x \; y \; 1\end{bmatrix}^{\mathsf{T}} = \mathbf{0}$$

The unknowns x and y are found uniquely as the 8th and 9th element of the right zero-eigenvector of N via singular value decomposition.

Solution with 6 points: Using only 6 points the nullspace of the homogeneous equation system imposed by (3) is four-dimensional:

$$\mathsf{C} = x\mathsf{C}_1 + y\mathsf{C}_2 + z\mathsf{C}_3 + \mathsf{C}_4 \tag{8}$$

The three coefficients are obtained similar to [15]. To do this, observe, that the ten constraints (5), (6) and (7) are cubic polynomials in x, y and z. Ordering the 20 monomials of up to 3rd degree in graded reverse lexicographic order and partitioning them into two vectors of size ten, one gets

$$\boldsymbol{q} = \begin{bmatrix}x^3 \; x^2y \; x^2z \; xy^2 \; xyz \; xz^2 \; y^3 \; y^2z \; yz^2 \; z^3\end{bmatrix}^{\mathsf{T}}$$

$$\boldsymbol{r} = \begin{bmatrix}x^2 \; xy \; xz \; y^2 \; yz \; z^2 \; x \; y \; z \; 1\end{bmatrix}^{\mathsf{T}}$$

The ten constraints are now expressible as

$$\mathsf{N}\begin{bmatrix}\boldsymbol{q}\\ \boldsymbol{r}\end{bmatrix} = \begin{bmatrix}\mathsf{N}_1 \; \mathsf{N}_2\end{bmatrix}\begin{bmatrix}\boldsymbol{q}\\ \boldsymbol{r}\end{bmatrix} = \mathsf{N}_1\boldsymbol{q} + \mathsf{N}_2\boldsymbol{r} = \mathbf{0}$$

and it follows, that

$$\boldsymbol{q} = -\mathsf{N}_1^{-1}\mathsf{N}_2\boldsymbol{r} = \mathsf{B}\boldsymbol{r}$$

Also observe, that the first six elements of $\boldsymbol{q}$ are a multiple of the first six elements of $\boldsymbol{r}$. Combining this and denoting with $\boldsymbol{B}_{1:6,:}$ the first six rows of B, one obtains the condition

$$\boldsymbol{q} = \begin{bmatrix}\boldsymbol{B}_{1:6,:}\\ \begin{bmatrix}\mathsf{I}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 2}\\ \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & 1 & \mathbf{0}_{1\times 3}\end{bmatrix}\end{bmatrix}\boldsymbol{r} = \mathsf{F}\boldsymbol{r} = x\boldsymbol{r}$$

Obviously $\boldsymbol{r}$ is an eigenvector of F and one obtains up to ten solutions with the 7th, 8th and 9th elements of these vectors being the unknown parameters to be fed into (8).

3.4 Solution with 5 points

To our knowledge the strategy taken thus far does not carry over to the minimal case of 5 given points. We were unable to find enough linear independent constraints. Therefore we chose a different path.

1. First, the direction of the cylinder axis is determined, leading to a 6-th degree polynomial in the direction parameters a and b
2. Second, the position of the cylinder axis across this direction and the radius are determined, leading to a linear equation system.

Determination of the direction of the cylinder axis: The direction of cylinder axis is determined by a rotation such that the cylinder axis is the Z-axis. Then all rotated points, when projected into the XY-plane are co-circular.

Using quaternions, this rotation can be represented as

$$\mathsf{R}(a,b) = \frac{1}{1+a^2+b^2}\begin{bmatrix} 1+a^2-b^2 & 2ab & 2b \\ 2ab & 1-a^2+b^2 & -2a \\ -2b & 2a & 1-a^2-b^2 \end{bmatrix}$$

as the quaternion $\mathbf{q} = (1,[a,b,0]) = (1, \boldsymbol{r}\tan\phi/2)$ represents a general rotation around a horizontal axis $\boldsymbol{r}$ with angle ϕ. Only angles $\phi \leq 90^o$ are relevant in our context, thus $a^2+b^2 \leq 1$.

All 5 points $\boldsymbol{X}_i$ are then transformed according to $\boldsymbol{X}'_i(a,b) = \mathsf{R}(a,b)\boldsymbol{X}_i$ leading to

$$\begin{bmatrix} X'_i(a,b) \\ Y'_i(a,b) \\ Z'_i(a,b) \end{bmatrix} = \frac{1}{1+a^2+b^2}\begin{bmatrix} X_i a^2 - X_i b^2 + 2Y_i ab + 2Z_i b + X_i \\ -Y_i a^2 + Y_i b^2 + 2X_i ab - 2Z_i a + Y_i \\ -Z_i a^2 - Z_i b^2 + 2Y_i a - 2X_i b + Z_i \end{bmatrix}$$

The projection of all 5 $\boldsymbol{X}'_i$ into the $X'Y'$-plane must be co-circular and therefore obey equation (4). Using the substitution $u = s^2+t^2-r^2$, this can be formulated as homogeneous equation system

$$\begin{bmatrix} X_1'^2(a,b)+Y_1'^2(a,b) & -2X'_1(a,b) & -2Y'_1(a,b) & 1 \\ X_2'^2(a,b)+Y_2'^2(a,b) & -2X'_2(a,b) & -2Y'_2(a,b) & 1 \\ X_3'^2(a,b)+Y_3'^2(a,b) & -2X'_3(a,b) & -2Y'_3(a,b) & 1 \\ X_4'^2(a,b)+Y_4'^2(a,b) & -2X'_4(a,b) & -2Y'_4(a,b) & 1 \\ X_5'^2(a,b)+Y_5'^2(a,b) & -2X'_5(a,b) & -2Y'_5(a,b) & 1 \end{bmatrix}\begin{bmatrix} 1 \\ s \\ t \\ u \end{bmatrix} = \mathsf{H}(a,b)\begin{bmatrix} 1 \\ s \\ t \\ u \end{bmatrix} = \mathbf{0} \tag{9}$$

Each of the five 4×4-submatrices of $\mathsf{H}(a,b)$ must therefore be singular, i.e. have a zero determinant. The numerators of this five determinants are bivariate polynomials of 6-th degree in the two variables a and b, hence are expressible as

$$p_l(a,b) = \begin{bmatrix} 1 \; a \; a^2 \; a^3 \; a^4 \; a^5 \; a^6 \end{bmatrix} \boldsymbol{G}_l \begin{bmatrix} 1 \; b \; b^2 \; b^3 \; b^4 \; b^5 \; b^6 \end{bmatrix}^{\mathsf{T}} = 0, \quad l = 1,...,5 \tag{10}$$

Their common roots need to be calculated in order to obtain the cylinder axis direction.

Determination of position and radius: Having computed a set of common roots, i.e. the cylinder axis directions, for each solution the translation and radius of the cylinder must be computed. Therefore one either solves the homogeneous equation system (9), or, more efficiently, selects three arbitrary rows and converts it into the linear equation system (2), however referring to the rotated points $\mathbf{X}'$ yielding the remaining cylinder parameters.

Finding the common roots of the 6-th order polynomials: For finding the common roots of the 6-th order polynomials (10) we use an interval method (cf. [9]) like [10] did in the univariate case. More specifically we use an approach using Bernstein polynomials (cf. [6]), to track down the roots of the bivariate polynomials. First the polynomials are transformed, so that all roots are inside the unit box $[0,1] \times [0,1]$. Since rotation of the cylinder axis by 180° does not change the cylinder, all roots are found inside the box $[-1,1] \times [-1,1]$ and therefore by a simple variable substitution the coefficients become

$$\overline{\boldsymbol{G}} = \boldsymbol{\Gamma} \boldsymbol{G} \boldsymbol{\Gamma}^{\mathsf{T}} \tag{11}$$

with

$$\boldsymbol{\Gamma}_{ij} = \begin{cases} \binom{j}{i}(-1)^{j-i}2^{i} & \text{if } i \leq j \\ 0 & \text{otherwise} \end{cases}$$

Next the polynomials are transformed into the Bernstein basis by

$$\boldsymbol{B} = \boldsymbol{\Phi}^{-1} \overline{\boldsymbol{G}} \boldsymbol{\Phi}^{-\mathsf{T}} \tag{12}$$

with

$$\boldsymbol{\Phi}_{ij} = \begin{cases} \binom{6}{j}\binom{6-j}{i-j}(-1)^{i-j} & \text{if } i \geq j \\ 0 & \text{otherwise} \end{cases}$$

One property of this Bernstein coefficients $\boldsymbol{B}$ is, that their minima and maxima yield a lower and upper bound on the polynomial in the unit box. Therefore bounds on equation (10) in the box $[-1,1] \times [-1,1]$ are given by

$$\min \boldsymbol{B} \leq p([-1,1],[-1,1]) \leq \max \boldsymbol{B}$$

so that one can easily decide for each polynomial, if there is any root in the interval of interest by checking, if there exists positive and negative coefficients.

To track down the roots, the intervals need to be bisected and the Bernstein coefficients of the polynomials, that have the roots of the bisected interval inside the unit box, must be computed. Fortunately there is a much more efficient method than applying equations (11) and (12). The two sets of Bernstein coefficients of the bisection are computable using the following dynamic programming algorithm: For the bisection along the x-axis the coefficients starting with ${}^{x_1}\boldsymbol{B}^{(0)} = \boldsymbol{B}$ are updated sequentially according to

$$ {}^{x_1}\boldsymbol{B}^{(k)}_{ij} = \begin{cases} \dfrac{{}^{x_1}\boldsymbol{B}^{(k-1)}_{i-1,j} + {}^{x_1}\boldsymbol{B}^{(k-1)}_{ij}}{2} & \text{if } i > k \\ {}^{x_1}\boldsymbol{B}^{(k-1)}_{ij} & \text{otherwise} \end{cases} \qquad k = 1, ..., 7$$

yielding the new set of coefficients ${}^{x_1}\boldsymbol{B} = {}^{x_1}\boldsymbol{B}^{(7)}$ representing the polynomial having the roots inside the left hand side subinterval put into the unit interval. The coefficients ${}^{x_2}\boldsymbol{B}$ of the right hand side subinterval are obtained during this computation using the fact, that

$$ {}^{x_2}\boldsymbol{B}_{ij} = \ {}^{x_1}\boldsymbol{B}^{(8-i)}_{7j} $$

The computation of the bisection along the y-axis is completely symmetric, i.e. starting with $\ {}^{y_1}\boldsymbol{B}^{(0)} = \boldsymbol{B}$ the coefficients are sequentially updated according to

$$ {}^{y_1}\boldsymbol{B}^{(k)}_{ij} = \begin{cases} \dfrac{{}^{y_1}\boldsymbol{B}^{(k-1)}_{i,j-1} + \ {}^{y_1}\boldsymbol{B}^{(k-1)}_{ij}}{2} & \text{if } j > k \\ {}^{y_1}\boldsymbol{B}^{(k-1)}_{ij} & \text{otherwise} \end{cases} \qquad k = 1, ..., 7 $$

$$ {}^{y_2}\boldsymbol{B}_{ij} = \ {}^{y_1}\boldsymbol{B}^{(8-j)}_{i7} $$

Putting everything together the roots of the five polynomials are found as follows: First the Bernstein coefficients for each polynomial are computed. Then the intervals are alternating bisected along the x- and the y-axis. By checking signs of the Bernstein coefficients it is decided, if each of the five polynomials has a possible root inside the subintervals. If this is the case, the search is continued inside this subinterval. Note, that the size of the subintervals and therefore the accuracy of the roots decreases exponentially. A final single Gauss-Newton update may be applied to further increase the accuracy of the roots.

4 Experiments

4.1 Finding Cylinders with RANSAC

The value of direct solutions for computing cylinders from minimal sets of 3D-points is, that the RANSAC-algorithm for robust estimation needs a direct solution from as few data as possible to be efficient. In [7], p. 104, the number of its iterations is given by $N = \log{(1-p)}/\log{(1-(1-\epsilon)^s)}$ where p is the error probability, ϵ is the proportion of outliers and s is the size of the sample. As discussed above, the complexity of the algorithm and thus the running time per sample increases with decreasing sample size s. Therefore the sample size must be carefully engineered with respect to the expected proportion of outliers in the data. If few outliers are expected, the 9-point-solution is fast and easy and the additional running time due to more RANSAC-iterations is negligible. If on the other hand many outliers are expected, the 5-point-solution will increase the overall running time. All intermediate solutions may be useful, too, depending on the speed of the implementations and the expected number of outliers in the data.

To find all cylinders contained in a 3D-point-cloud, we proceed as follows: Repeatedly a set of five points is sampled at random from the set of points and the cylinders going through this five points are computed. For each of this cylinders the points lying on its surface are counted and the one cylinder is retained, that has most supporting points on its surface. If the number of supporting points is to low, the process is stopped and the cylinder is removed. Otherwise the cylinder is kept, the supporting points are removed from the point-cloud and the whole process is iterated.

4.2 Results

The efficiency of the root-finder: The performance of the five-point-method mainly depends on the efficiency for finding the common roots of the five polynomial equations yielding the axis direction of the cylinder. In figure 1, left, the logarithm of the sum of the five squared polynomials is shown for a typical point configuration. The standard Gauss-Newton-Method for finding the four roots would search this cost function.

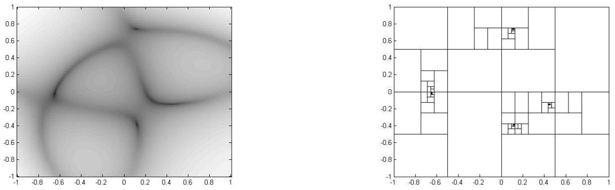

Fig. 1. Left: Logarithm of the sum of the five squared polynomials for a typical point configuration. The minima of this function would be searched with the standard Gauss-Newton-Method. Right: The bisections required with the Bernstein-Method for tracking down the roots of the same five polynomials as depicted in the figure left.

The approach using Bernstein-polynomials is much more efficient than this. The bisections required for the previous example polynomials are shown in figure 1, right. Obviously the quality of the bounds is essential for the efficiency of the approach. As seen in figure 1, right, the required bisection for that special example are very good. To quantify the quality, the area searched by the algorithm in each iteration is analyzed. For the method to be efficient, this area must decrease exponentially. As seen in figure 2, left, this is the case, as the logarithm of the average search area for random point configurations is shown to decrease linearly.

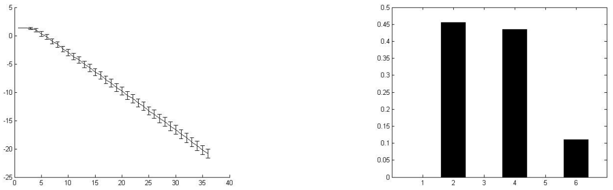

Fig. 2. Left: Logarithm of the average area considered by the root finder (with standard deviation) against the search depth for random point configurations. Right: Histogram of the number of solutions.

Number of solutions: Another crucial point for the efficiency of the RANSAC-procedure is the number of solutions, that are found by the algorithm. The maximum number of different solutions for this problem is not known. Due to the ambiguity of the rotation parameters (a, b) it must be less or equal 18. This is because two 6-th degree polynomials in general may have up to 36 real solutions and the two quaternions $(1, a, b, 0)$ and $(1, -a/(a^2 + b^2), -b/(a^2 + b^2, 0))$ rotate the same axis into the Z-axis. In our experiments the number of solutions was always 2, 4 or 6, though.

In figure 2, right, the histogram of the number of solutions for random point configurations is shown. The average number of solutions was 3.3.

Experiment with real data: Finally the performance of the algorithm on real data is shown. In figure 3, left, a 3D point cloud comprising of about 170.000 points is depicted. It was taken by a laser scanner at an industrial site containing several pipes. Figure 3, right, shows the cylinders, that were extracted from this point cloud.

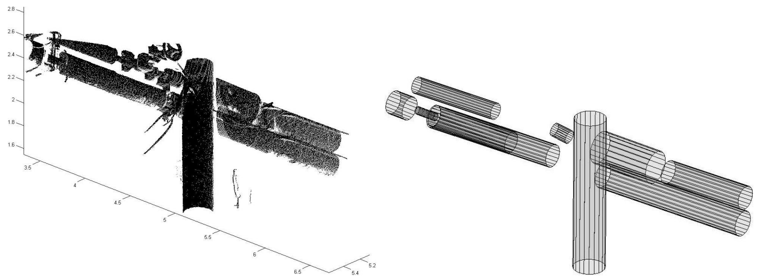

Fig. 3. Left: 3D point cloud obtained by a laser-scanner at an industrial site (courtesy of G.Vosselman and T.Rabbani). Right: Extracted cylinders.

5 Conclusion

We have presented direct solutions for determining the parameters of cylinders from surface points, which are to our knowledge new except for the 9-point-method. The five-point algorithm for circular straight cylinders has been efficiently realized using Bernstein polynomials and tested on synthetic and real range data. There are still some open problems:

- The maximum number of solutions is unknown.
- The critical configurations are unknown.
- It needs to be investigated under which constraints the other solutions, with 6 and more points, are more efficient.

References

1. F. L. Bookstein. Fitting conic sections to scattered data. *CGIP*, 9(1):56–71, 1979.
2. T. Chaperon and F. Goulette. Extracting cylinders in full 3d data using a random sampling method and the gaussian image. In *Proceedings of the Vision Modeling and Visualization Conference*, pages 35–42, 2001.
3. O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.
4. M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.
5. M. A. Fischler and R. C. Bolles. A RANSAC-based approach to model fitting and its application to finding cylinders in range data. In *IJCAI81*, pages 637–643, 1981.
6. J. Garloff and A. P. Smith. Solution of systems of polynomial equations by using bernstein expansion. In G. Alefeld, S. Rump, J. Rohn, and T. Yamamoto, editors, *Symbolic Algebraic Methods and Verification Methods*. Springer, 2001.
7. R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
8. H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer, and W. Stuetzle. Piecewise smooth surface reconstruction. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 295–302, New York, NY, USA, 1994. ACM Press.
9. R. Martin, H. Shou, I. Voiculescu, A. Bowyer, and Guojin Wang. Comparison of interval methods for plotting algebraic curves. *Comput. Aided Geom. Des.*, 19(7):553–587, 2002.
10. D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(6):756–777, 2004.
11. M. Peternell, H. Pottmann, and T. Steiner. Hough transform and Laguerre geometry for the recognition and reconstruction of special 3D shapes. Technical Report 100, Institute of Geometry, April 2003.
12. S. Petitjean. A survey of methods for recovering quadrics in triangle meshes. *ACM Comput. Surv.*, 34(2):211–262, 2002.
13. J. Philip. A non-iterative algorithm for determining all essential matrices corresponding to five point pairs. *Photogrammetric Record*, 15(88):589–599, 1996.
14. G. Roth and M. D. Levine. Segmentation of geometric signals using robust fitting. In *Int. Conference on Pattern Recognition*, pages 826–831, 1990.
15. H. Stewenius, C. Engels, and D. Nister. Recent developments on direct relative orientation. *ISPRS Journal*, 2006. to appear.
16. C.K. Tang and G. Medioni. Curvature-augmented tensor voting for shape inference from noisy 3d data. *PAMI*, 24(6):858–864, June 2002.
17. G. Vosselman, B. G. H. Gorte, G. Sithole, and T. Rabbani. Recognising structure in laser scanner point clouds. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 46, pages 33–38, 2004.
18. N. Werghi, R.B. Fisher, C. Robertson, and A.P. Ashbrook. Faithful recovering of quadric surfaces from 3d range data. In *Second International Conference on 3-D Imaging and Modeling3DIM99*, pages 280–289, 1999.
19. S. Winkelbach, R. Westphal, and T. Gösling. Pose estimation of cylindrical fragments for semi-automatic bone fracture reduction. In Bernd Michaelis and Gerald Krell, editors, *DAGM-Symposium*, volume 2781 of *Lecture Notes in Computer Science*, pages 566–573. Springer, 2003.