2. Shape Recognition

Given a data set \( P \), we obtain a rough guess of its coefficients using the eigen-decomposition method [2].

The general function can be written as,

\[
Q(x) = x'Kx + [G \ H \ I]x + J = 0
\]

with \( x = [x, y, z]^T \).

Perform eigen-decomposition onto the quadric form,

\[
K = \begin{bmatrix}
A & D/2 & E/2 \\
D/2 & B & F/2 \\
E/2 & F/2 & C
\end{bmatrix} = USU^T
\]

\( S \) is a diagonal matrix with its diagonal entries \( \sigma_1 \geq \sigma_2 \geq \sigma_3 \).

\( U \) is a 3\( \times \)3 rotation matrix. Assuming \( x = U^T x' \), then

\[
Q(x) = x'^T S x' + [G \ H \ I] U x' + J = 0
\]

\[
= \sigma_1 x'^2 + \sigma_2 y'^2 + \sigma_3 z'^2 + \hat{G} \hat{y} + \hat{L} z + J = 0
\]

If \( \sigma_1 \sigma_2 \sigma_3 \neq 0 \),

\[
Q(x) = \sigma_1 (x-a_1)^2 + \sigma_2 (y-a_2)^2 + \sigma_3 (z-a_3)^2 + a_4 = 0
\]

To guarantee the surface representation’s uniqueness, the coefficients are scaled by a positive factor.

If \( \sigma_1 = 0 \),

\[
Q(x) = \sigma_2 (x-a_1)^2 + \sigma_3 (y-a_2)^2 + \hat{L} z + a_4 = 0
\]

If \( \sigma_2 = 0 \), the function can be processed similarly.

If \( \sigma_3 = \sigma_2 = 0 \),

\[
Q(x) = \sigma_1 (x-a_1)^2 + \hat{G} \hat{y} + a_4 = 0
\]

3. Orthogonal Distance Fitting

In refinement, transformations are always performed onto the data and the quadric surface is represented in a standard implicit form \( f(x, y, z) = 0 \).

Fitting is carried out in a nested approach \( \min \sum_{k=1}^{N} |q_k - p_k|^2 \), where \( q_k \) is the projection point from \( p_k \) to the surface, and \( a \) is the shape and motion parameters.

The Jacobian matrix at the outer iteration is [5],

\[
J = \frac{\partial \Phi}{\partial b} = \frac{\partial \Phi}{\partial q} \left[ \frac{\partial q}{\partial b} + \frac{\partial f}{\partial b} \right]
\]

The motion and shape parameters are updated using the Levenberg-Marquardt algorithm,

\[
(J'J + \lambda I)b = -J'd
\]

4. Numerical Experiments

We compared linear least squares, implicit ODF and specific ODF using a cylinder \( x^2 + y^2 = R^2 \) (\( R = 1.5 \) mm) and a cone \( \text{cot} \theta \sqrt{x^2 + y^2} = z \ (\theta = 45') \). The two surfaces were randomly moved to an arbitrary position. The fractal Brownian motion [5] was employed to simulate noise with mean 0 and \( \sigma = 0.5 \) \( \mu \). The programs were run 150 times.

Table 2. Fitting results of cylinder

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Linear</th>
<th>Implicit ODF</th>
<th>Specific ODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha_1 )</td>
<td>Bias (%)</td>
<td>Uncertainty (%)</td>
<td>R2</td>
</tr>
<tr>
<td>( \alpha_2 )</td>
<td>4.77</td>
<td>0.145</td>
<td>0.256</td>
</tr>
<tr>
<td>( \alpha_3 )</td>
<td>-29.75</td>
<td>-0.45</td>
<td>0.246</td>
</tr>
</tbody>
</table>

Table 3. Fitting results of cone

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Linear</th>
<th>Implicit ODF</th>
<th>Specific ODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha_1 )</td>
<td>Bias (%)</td>
<td>Uncertainty (%)</td>
<td>R2</td>
</tr>
<tr>
<td>( \alpha_2 )</td>
<td>0.941</td>
<td>0.136</td>
<td>0.255</td>
</tr>
<tr>
<td>( \alpha_3 )</td>
<td>-1.003</td>
<td>-0.013</td>
<td>0.245</td>
</tr>
</tbody>
</table>

Another three quadric surfaces were tested, an ellipsoid \( a_1x^2 + a_2y^2 + a_3z^2 = 1 \) with \( a_1 = 0.5, a_2 = 0.25, a_3 = 0.25 \), a hyperbolic paraboloid \( x^2 + a_2y^2 + b_2 = 0 \) with \( a_1 = -1, b_2 = -2 \) mm and a parabolic cylinder \( x^2 + b_2 = 0 \) with \( b = 3 \) mm.

The programs were run 150 times and the fitting results of the implicit ODF method are given below,

Table 4. Implicit ODF results of the three quadric surfaces

<table>
<thead>
<tr>
<th>Shape</th>
<th>Ellipsoid</th>
<th>Hyperbolic Paraboloid</th>
<th>Parabolic Cylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>( \alpha_1 )</td>
<td>( \alpha_2 )</td>
<td>( \alpha_3 )</td>
</tr>
<tr>
<td>Bias (%)</td>
<td>-0.012</td>
<td>-0.022</td>
<td>-0.067</td>
</tr>
<tr>
<td>Uncertainty (%)</td>
<td>0.188</td>
<td>0.162</td>
<td>0.468</td>
</tr>
</tbody>
</table>

References