Computation of curvatures from 2.5D raster data

X. Laboureux, S. Seeger, G. Häusler

Curvatures are shift and rotation invariant local features of an object surface, well suited for object localization and recognition. Since they are very sensitive to noise (2^{nd} order derivatives), a locally parametric surface description is a convenient way to smooth the noise. In this report we present how to calculate the extremal curvatures at each point of a parametric surface using partial derivatives with respect to the surface parameters. In a second step we show that the matrix used to determine the partial derivatives on **2.5D raster** data is independent from the location within the image, which facilitates and speeds up the calculation of the curvatures.

Given a parametrical surface description $\vec{P}(u,v)=[x(u,v),y(u,v),z(u,v)]^T$, the minimal and maximal curvatures $\kappa_i\ (i=1,2)$ at $\vec{P}(u_0,v_0)$ can be calculated as the eigenvalues of the so called Weingarten map:

$$\begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1}\begin{pmatrix} L & M \\ M & N \end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix}=\kappa_i\begin{pmatrix}\alpha\\ \beta\end{pmatrix}$$ where

E, F, G, L, M and N directly depend on the first and second order partial derivatives to u und v [1].

As 3D-sensors provide no surface description but only point clouds, in order to calculate partial derivatives we locally approximate at each data point $\vec{P}_0$ a polynomial surface through the points in the neighborhood Ω of $\vec{P}_0$

$$\vec{P}(u,v)=\begin{bmatrix} x(u,v)=a_0+a_1u+a_2v+a_3u^2+a_4uv+... \\ y(u,v)=b_0+b_1u+b_2v+b_3u^2+b_4uv+... \\ z(u,v)=c_0+c_1u+c_2v+c_3u^2+c_4uv+... \end{bmatrix}$$

with $\vec{P}(0,0)=\vec{P}_0$.

In this way the partial derivatives at $\vec{P}_0$ are always proportional to one of the polynomial coefficients $\left(\partial\vec{P}/\partial u=[a_1,b_1,c_1]^T,\dots\right)$.

As our data have a raster structure, i.e. each point $\vec{P}_k$ can be written as $\vec{P}_{i,j}$ where (i,j) are the raster indices, an efficient way to determine the coefficients a_L, b_L, c_L of these polynomials defined at the point $\vec{P}_{i_0,j_0}$ is to choose (u,v) at the positions of the approximated points $\vec{P}_{i,j}$ in the neighborhood as $(i-i_0, j-j_0)$: therefore for each component of $\vec{P}(u,v)$ we have in a neighborhood of $N_P=\Delta I\times\Delta J$ points the N_P equations (for example for the x-component)[2].

$$x(i-i_o,j-j_o)=a_0+a_1u+a_2v+a_3u^2+a_4uv+a_5v^2+...\Big|_{\substack{u=i-i_0\\ v=j-j_0}}\overset{!}{=}x_{i,j}$$

for $i_{0\min}\le i\le i_{0\max}, j_{0\min}\le j\le j_{0\max}$ with:

$$i_{0\min}=i_0-\frac{\Delta I}{2}, i_{0\max}=i_0+\frac{\Delta I}{2}, j_{0\min}=j_0-\frac{\Delta J}{2}, j_{0\max}=j_0+\frac{\Delta J}{2}$$

Since this equation is linear with respect to the coefficients, it can also be written as

$\mathbf{M}\,\mathbf{a}=\mathbf{x}$ with $\mathbf{a}=[a_0,a_1,a_2,...]^T$,

$\mathbf{x}=[x_{i_{0\min},j_{0\min}},x_{i_{0\min+1},j_{0\min}},...,x_{i_{0\max-1},j_{0\max}},x_{i_{0\max},j_{0\max}}]^T$

and the matrix M following in a straight forward way from the last equation.

For a given polynomial order and a given neighborhood size the matrix M is a constant for all $\vec{P}_{i_0,j_0}$ and has to be calculated only once.

If there are less polynomial coefficients than equations (depending on both the order of the polynomials and the size of the neighborhood) we get an overdetermined system of equations and therefore search for the corresponding least-squares solution by minimizing

$\Phi:=\sum_{i,j\in\Omega}\left\|x(i-i_o,j-j_o)-x_{i,j}\right\|^2$ with respect to the polynomial coefficients a_L. Therefore the quadratic system of equations $\frac{\partial\Phi}{\partial a_L}=0$ has to be solved, which can be written as $\mathbf{M}^T\mathbf{M}\,\mathbf{a}=\mathbf{M}^T\,\mathbf{x}$ so that the solution for **a** is: $\mathbf{a}=\left(\mathbf{M}^T\mathbf{M}\right)^{-1}\mathbf{M}^T\,\mathbf{x}$.

[1] M.P. do Carmo, "Differentialgeometrie von Kurven und Flächen", Vieweg Studium, 1993.

[2] Wu Wang, S.S. Iyengar, "Efficient Data Structures for Model-Based 3-D Object Recognition and Localization from Range Images", IEEE Trans. PAMI Vol 14, No 10, Oct 1992, pp1035-1045.