Viewpoint Invariant Computation of Surface Curvatures in Range Images

Pierre Boulanger
National Research Council of Canada
Institute for Information Technology
Ottawa, Canada
K1A-0R6
e-mail: Boulanger@iit.nrc.ca
and
Paul Cohen
Department of Electrical Engineering
École Polytechnique of Montréal
Montréal, Canada
H3C-3A7
e-mail: cohen@ai.polymtl.ca

Abstract

This paper presents a new method of computing Gaussian and mean curvatures in range images, which produces stable (i.e. viewpoint invariant) segmentation map. The first and second partial derivatives are estimated by the weighted least square fit of a parametric biquadratic polynomial within a local moving window in the direction normal to the surface. The weights are determined on the basis of surface distance and normal angular distance from the center point of the window. An intrinsic parametrization method is used to ensure viewpoint invariance of the fitted model. This method can be used to perform a robust estimation of the surface curvatures. Experimental results are presented.

1 Introduction

Range image processing and representation has recently received increased attention because of the improvements in active and passive range sensing techniques (for example, [12]). Range data provide explicit geometrical information about the shape of visible surfaces. This basic information makes many problems in 3-D object description and recognition easier to solve.

Besl and Jain [1] have proposed an attractive idea for surface characterization from the point-of-view of differential geometry. Mean and Gaussian curvatures (surface curvatures) are invariant under viewpoint transformations. Smooth surfaces are then locally classified into one of eight surface types according to their curvature signs from which range images can be segmented into scene-specific viewpoint invariant regions [9]. The stability of this representation however, crucially depends upon the ability to accurately compute local surface curvatures.

Since differential geometry is a theory for smooth differentiable surfaces, one must take into account the fact that real range images contain discontinuities (depth and orientation) which will effect the computation of the curvatures. Most of the methods found in the literature estimate the derivatives necessary to compute the curvatures by using a local surface model of at least second order. The problem with these methods is two-fold. First, since in most cases the norm used to compute the local model is not truly invariant, there is a residual dependence of the Gaussian and mean curvatures on viewpoint and parametrization. Secondly, since in most methods there is no test performed to determine the geometrical compatibility of the centerpoint and its neighbors, there is a strong possibility that the estimate of the local curvature will be biased by the presence of geometrical discontinuities in the neighborhood.

To address these issues, Fan [6] proposed that discontinuities be detected first and then local curvatures measured. Yokoya and Levine [13] proposed a method which employs a selective surface fit. The local window which provides the best minimum fitting error among the covering windows is selected and is used for curvature estimation.

In order to solve these problems, we present in this paper a new method to compute Gaussian and mean curvatures which is invariant to parameterization and viewpoint. The first and second partial derivatives are determined by the weighted least-square fit of a parametric biquadratic polynomial within a local moving window. The present weighting method assigns small weights for the points which are not geometrically compatible with the centerpoint of the window. Thus, the derivatives are estimated based primarily on the points which are compatible with the centerpoint. Since the distances across a discontinuity are large, small weights are assigned to points which are on the other side of that discontinuity. Surface curvatures are then computed from these derivatives. The viewpoint invariance of the estimate arises from the fact that the surface estimation process minimizes the distance between the surface and the data points in the direction perpendicular to the tangent plane of the surface at this point. Taubin [14] suggests solving this problem by minimizing this distance using an implicit form of the quadric surfaces. Such a constraint necessitates the solution of a non-linear problem. This paper shows that the parametric form of this problem is linear. It will also demonstrate that one can modify the original surface parametrization, which is extrinsic to the surface, into an intrinsic one corresponding to the minimal norm between the data points and the local surface model. This new parameterization technique is a generalization of the method introduced by Hoschek [10].

In Section 2, the concept of intrinsic surface distance and normal angular distance are reviewed. Then an algorithm to compute these distances within a window is presented. In Section 3, the use of these distances to do a weighted least-square fit using a parametric polynomial, where the parametrization is intrinsic to the fitting problem and the surface geometry, is demonstrated. In Section 4, experimental results of mean and Gaussian curvature computations using our method are presented and then compared with the results produced by a non-weighted least-square method. The robustness of the method to Gaussian and non-Gaussian noise is also analyzed.

2 Surface and Normal Angular Distance

In this section, concepts of differential geometry [4] will be briefly reviewed, and then the notion of intrinsic surface distance and normal angular distance will be defined [3]. An algorithm to compute these distances within a window is also presented.

2.1 Differential Geometry of Range Images

Consider a parametric surface $\vec{\eta}(u,v) = (h_x(u,v), h_y(u,v), h_z(u,v))$ where (u,v) belong to an open set $U \subset R^2$. Using this parametric form, one can define uniquely the local property of the surface by using the Gaussian $K(u,v)$ and mean curvature $H(u,v)$ defined as:

$$K(u,v) = \frac{[\vec{\eta}_{uu}\ \vec{\eta}_u\ \vec{\eta}_v][\vec{\eta}_{vv}\ \vec{\eta}_u\ \vec{\eta}_v] - [\vec{\eta}_{uv}\ \vec{\eta}_u\ \vec{\eta}_v]^2}{|\vec{\eta}_u \times \vec{\eta}_v|^4} \tag{1}$$

and

$$H(u,v) = ((\vec{\eta}_v \cdot \vec{\eta}_v)[\vec{\eta}_{uu}\ \vec{\eta}_u\ \vec{\eta}_v](\vec{\eta}_u \cdot \vec{\eta}_u)[\vec{\eta}_{vv}\ \vec{\eta}_u\ \vec{\eta}_v] \\ -2(\vec{\eta}_u \cdot \vec{\eta}_v)[\vec{\eta}_{uv}\ \vec{\eta}_u\ \vec{\eta}_v]^2)/2|\vec{\eta}_u \times \vec{\eta}_v|^3 \tag{2}$$

where $[\vec{a}\ \vec{b}\ \vec{c}] = \vec{a} \cdot (\vec{b} \times \vec{c})$. The Gaussian and mean curvatures are two natural algebraic invariants of the surface and are independent of surface parametrization and viewpoint. In addition, Gaussian curvature uniquely determines the shape of convex surfaces, and mean curvature uniquely determines the shape of graph surfaces under various auxiliary conditions [2], [7],[8].

The second most important geometrical property necessary to compute is the notion of surface distance. To define a distance on a surface, it is convenient to express the first fundamental form of the surface at a point $\vec{p} = \vec{\eta}(u_p, v_p)$ in the basis $\{\vec{\eta}_u, \vec{\eta}_v\}$ associated with the parametrization u, v. Let $\vec{\alpha}'(t)$ be the tangent vector to a parametrized curve $\vec{\alpha}(t) = \vec{\eta}(u(t), v(t)), t \in (-\epsilon, \epsilon)$ embedded in the surface and going through the point $\vec{p}$. Then the first fundamental form at $\vec{p}$ is given by:

$$I_p(\vec{\alpha}'(0)) = E(u')^2 + 2Fu'v' + G(v')^2, \tag{3}$$

where $E =< \vec{\eta}_u, \vec{\eta}_u >_p$, $F =< \vec{\eta}_u, \vec{\eta}_v >_p$, and $G =< \vec{\eta}_v, \vec{\eta}_v >_p$ are the coefficients of the first fundamental form in the basis of the tangent plane at the point $\vec{p}$. The operator $< \cdot, \cdot >$ corresponds to the inner product. u' and v' are the derivatives with respect to the parameter t.

By using the first fundamental form, one can treat metric questions on a regular surface without reference to the ambient space. For example, the arc length s_α between two points $\vec{p} = \vec{\alpha}(t_p)$ and $\vec{q} = \vec{\alpha}(t_q)$ along a parametrized curve $\vec{\alpha}(t)$ is given by:

$$s_\alpha(\vec{p},\vec{q}) = \int_{t_p}^{t_q} |\vec{\alpha}'(t)|dt = \int_{t_p}^{t_q} \sqrt{I_p(\vec{\alpha}'(t))}dt. \quad (4)$$

Since only discrete surface measurements are available on discrete parametric coordinates (u_i, v_i), one must have a discrete form of equation (4) to compute the arc length between two points along the curve.

Let us consider a partition of a curve $\vec{\alpha}(t)$ defined as $\vec{\alpha}(t_i), t_p = t_0 < t_1 < \ldots < t_k < t_{k+1} = t_q$. If the steps of the piecewise approximation are sufficiently small, one can approximate the curve by a linear equation of the form:

$$\vec{\alpha}(t) = (x(t_i)+a_i(t-t_i), y(t_i)+b_i(t-t_i), z(t_i)+c_i(t-t_i)), \quad (5)$$

where $a_i = [x(t_{i+1}) - x(t_i)]/(t_{i+1} - t_i)$, b_i, and c_i have similar expressions based on y and z. Then the derivative $\vec{\alpha}'(t)$ of $\vec{\alpha}$ with respect to t is given by:

$$\vec{\alpha}'(t) = (a_i, b_i, c_i) \text{ for } t_i < t < t_{i+1}.$$

Thus a discrete approximation of the arc length between $\vec{p}$ and $\vec{q}$ is given by:

$$s_\alpha(\vec{p},\vec{q}) = \sum_{i=0}^{k} \sqrt{a_i^2 + b_i^2 + c_i^2}. \quad (6)$$

This is equivalent to a polygonal approximation of the surface.

Then the surface distance d_S, namely the minimum distance among all trajectories joining the two points, is defined by:

$$d_S(\vec{p},\vec{q}) = \min_\alpha s_\alpha(\vec{p},\vec{q}). \quad (7)$$

Note that the value of d_S will be large if the minimum trajectory goes across a depth discontinuity.

However, this distance is not very sensitive to orientation discontinuity. Therefore, it is necessary to consider another distance which is sensitive to the change of surface orientation. The normal angular distance is defined as the average angular variation of the normal vector along the trajectory joining the points $\vec{p}$ and $\vec{q}$ as expressed by:

$$d_A(\vec{p},\vec{q}) = \frac{1}{(t_q - t_p)} \int_{t_p}^{t_q} \cos^{-1}(<\vec{n}(t_p), \vec{n}(t)>)dt, \quad (8)$$

where $\vec{n}(t_p)$ is the normal vector at t_p.

From the definition (8), it is obvious that $d_A(\vec{p},\vec{q})$ is equal to zero if the normal vector along the trajectory is constant. If, however, the trajectory goes across an orientation discontinuity, the value of d_A will be large. Moreover, this measure is also independent of viewpoint. The discrete form of equation (8) is given by:

$$d_A(\vec{p},\vec{q}) = \frac{1}{k} \sum_{i=1}^{k} \cos^{-1}(<\vec{n}(t_0), \vec{n}(t_i)>). \quad (9)$$

2.2 Algorithm to Find Minimum Trajectory

To obtain surface distances from the center point to the other points in a moving window, one needs to find minimum trajectories from the center point to all other points in the window. From the definition (7), one can design an efficient algorithm by using the Single-Source Shortest Paths Algorithm for weighted graphs (for example, see [11]).

The vertices of the graph correspond to the points in the window and the edges represent neighboring connections of points. From the equation (6), the arc length of the edge between points (u_{t_i}, v_{t_i}) and $(u_{t_{i+1}}, v_{t_{i+1}})$ is given by:

$$\sqrt{a_i^2 + b_i^2 + c_i^2}.$$

The algorithm to find minimum distances is given as follows.

Input: $G = (V, E)$ (a weighed graph), and v (the source vertex corresponding to the center point)

Output: for each vertex w, w.SP is the length of the shortest path from v to w and corresponding minimum trajectories w.TR.

```
begin
for all vertices w do
    /* w.mark indicate if the vertex distance is determined*/
    w.mark := false;
    w.SP := ∞;
v.SP := 0;
v.TR := 0;
while there exists an unmarked vertex do
    let w be an unmarked vertex such that w.SP is minimum;
    w.mark := true;
    for all neighboring edges (w, z) such that z is unmarked do
        if w.SP + length(w, z) < z.SP then
            z.SP := w.SP + length(w, z);
            z.TR := w;
end
```

In this algorithm, one needs to find the minimum distances among a set of path lengths and to update the path lengths frequently. One can implement this efficiently by using a heap. All unmarked vertices are kept in a heap with their current known shortest path lengths from the center point v as their keys. To find an unmarked vertex w such that the path length w.SP is minimum, one can simply take it from the top of the heap. All the edges connected to the vertex w can be checked and the path lengths can be updated without difficulty. Since the elements of the heap are the vertices of the graph, the space requirement is only $O(|V|)$, where $|V|$ denotes the number of vertices. The running time of this algorithm is $O((|E| + |V|)\log|V|)$, where $|E|$ denotes the number of edges.

To compute the normal angular distance, one needs to estimate the surface normal at each point. In the following experiments, the estimates used were obtained by a least-square fitting of a plane within local 3 by 3 windows using a method described by Faugeras and Hebert [5].

Once estimates of the surface normal have been obtained, the normal angular distance is easily computed by tracing the minimum trajectory obtained by the previous algorithm.

3 Computation of Local Surface Model

This section presents a new local parametric modeling scheme based on an intrinsic parametrization of the local neighborhood which will allow computation of a local second-order model which is invariant to viewpoint and parametrization. From this local model, one could compute estimates of the following partial derivatives: $\frac{\partial \vec{\eta}}{\partial u} = \vec{\eta}_u$, $\frac{\partial \vec{\eta}}{\partial v} = \vec{\eta}_v$, $\frac{\partial^2 \vec{\eta}}{\partial u^2} = \vec{\eta}_{uu}$, $\frac{\partial^2 \vec{\eta}}{\partial v^2} = \vec{\eta}_{vv}$, $\frac{\partial^2 \vec{\eta}}{\partial u \partial v} = \vec{\eta}_{uv}$ that can be used to compute Gaussian and mean curvatures. An invariant estimates of these derivatives is essential in order for the Gaussian and mean curvatures estimates to be truly invariant. Let Ω be a neighborhood of size $m = N \times N$ where $\vec{r}(u,v) = (x(u,v), y(u,v), z(u,v))$ are the measurements produced by the range sensor. Let u, v be the initial parametrization of the surface where $u, v \in U \times U$, and N are assumed to be odd. The set U is defined as:

$$U = \{-(N-1)/2, \ldots, -1, 0, 1, \ldots, (N-1)/2\}. \quad (10)$$

Let

$$\vec{\eta}(u,v) = \sum_{i=0}^{2} \sum_{j=0}^{2} \vec{a}_{ij} u^i v^j = (h_x(u,v) \;\; h_y(u,v) \;\; h_z(u,v)) \quad (11)$$

where $\vec{a}_{ij}$ is the coefficient vector of each component of $\vec{\eta}(u,v)$ and is equal to zero for $i + j > 2$, and $h_x(u,v)$, $h_y(u,v)$, and $h_z(u,v)$ are the respective parametric functions in each direction. Using a matrix notation $\vec{\eta}(u,v) = M^t A$ one can express equation (11) as:

$$\vec{\eta}(u,v) = \begin{bmatrix} 1 & u & v & u^2 & uv & v^2 \end{bmatrix} \begin{bmatrix} \vec{a}_{00} \\ \vec{a}_{10} \\ \vec{a}_{01} \\ \vec{a}_{20} \\ \vec{a}_{11} \\ \vec{a}_{02} \end{bmatrix} \quad (12)$$

With these polynomials, the partial derivatives are equal to $\vec{\eta}_u = \vec{a}_{10} + \vec{a}_{11} v_o + 2\vec{a}_{20} u_o$, $\vec{\eta}_v = \vec{a}_{01} + \vec{a}_{11} u_o + 2\vec{a}_{02} v_o$, $\vec{\eta}_{uv} = \vec{a}_{11}$, $\vec{\eta}_{uu} = 2\vec{a}_{20}$, and $\vec{\eta}_{vv} = 2\vec{a}_{02}$ where (u_o, v_o) is the parametric coordinate of the center window. One of the method to compute these coefficients is to find the one which minimize the following equation:

$$\sum_{(u,v)\in\Omega} w(u,v) \parallel \vec{r}(u,v) - \vec{\eta}(u,v) \parallel^2, \quad (13)$$

where $w(u,v)$ is an intrinsic weighting function which measures the geometric distance between the central pixel and its neighborhood. This weighting function, first introduced by Boulanger and Cohen [3], is defined as:

$$w(u,v) = \frac{\exp(-(d_S^2(u,v)/2\sigma^2) + \beta d_A^2(u,v))}{\sum_{(u,v)\in\Omega} \exp(-(d_S^2(u,v)/2\sigma^2) + \beta d_A^2(u,v))}, \quad (14)$$

The parameter σ controls the scope of the function and the parameter β controls the ratio between surface distance and angular distance. The present weights have the following attractive properties:

- Larger weights are assigned for the points which are geometrically more compatible with the center point.
- The weights are sensitive to both depth and orientation discontinuities. Thus, small weights are assigned to points which are in the opposite side of a discontinuity.
- The weights are independent of viewpoint since there are based on an intrinsic surface metric.

The minimization problem is the following:

$$\min_{\vec{a}_{ij}} \sum_{(u,v)\in\Omega} w(u,v) \parallel \vec{r}(u,v) - \vec{\eta}(u,v) \parallel^2 \quad (15)$$

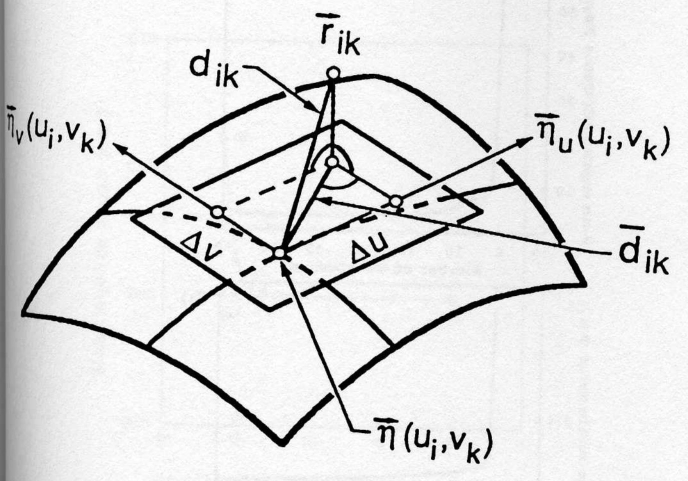

Figure 1: Norm used in the minimization problem.

which is equivalent to three independent minimization. The solution to this problem is given by the Euler condition, i.e.:

$$\frac{\partial}{\partial \bar{a}_{ij}} \sum_{(u,v)\in\Omega} w(u,v)(\vec{r}(u,v) - M^t A)^2 = \vec{0} \quad (16)$$

and is equal to:

$$A = (\sum_{(u,v)\in\Omega} (w(u,v)MM^T))^{-1} \sum_{(u,v)\in\Omega} w(u,v)\vec{r}(u,v)M. \quad (17)$$

where A correspond to the optimal coefficients of the parametric function.

3.1 Intrinsic Parametrization of the Local Model

One of the problems of fitting the data points using the previous method is the fact that one does not minimize the error in the direction perpendicular to the tangent plane. In fact this minimization depends on the initial parametrization which is extrinsic to the problem and viewpoint dependent. In this section a new intrinsic parametrization that minimizes such a distance will be defined.

One could avoid this problem of parametrization by using an implicit surface primitive such as a quadric. But, as demonstrated by Taubin [14], the constraint of orthogonal minimization necessitates the solution of a non-linear problem. It will be demonstrate that in the parametric case the problem can remain linear. The minimization problem is the following:

$$\min_{\hat{u},\hat{v}} ||\vec{r}(u,v) - \vec{\eta}(\hat{u},\hat{v};A)||^2 \text{ where } \vec{r}(u,v) = \vec{r}(\hat{u},\hat{v}), \quad (18)$$

i.e., find a new parametrization $(\hat{u},\hat{v})$ and a new model parameter A such that the distance between $\vec{r}(u,v)$ and $\vec{\eta}(\hat{u},\hat{v})$ is minimum. If one performs a Taylor expansion of the surface one obtains:

$$\vec{\eta}(\hat{u},\hat{v}) = \vec{\eta}(u,v) + \frac{\partial\vec{\eta}}{\partial u}(\hat{u}-u) + \frac{\partial\vec{\eta}}{\partial v}(\hat{v}-v). \quad (19)$$

Then the minimization problem is the following:

$$\min_{\hat{u},\hat{v}} ||\vec{r}(u,v) - \vec{\eta}(u,v) - \frac{\partial\vec{\eta}}{\partial u}(\hat{u}-u) - \frac{\partial\vec{\eta}}{\partial v}(\hat{v}-v)||^2 = \min_{\hat{u},\hat{v}} ||J\vec{w} - \vec{d}||^2. \quad (20)$$

where J is the Jacobian matrix of $\vec{\eta}(u,v)$, and $\vec{w}$ is a two components vector corresponding to the variation of the parametrization, i.e.:

$$\vec{w} = (\hat{u}-u, \hat{v}-v)^t. \quad (21)$$

The vector $\vec{d}(u,v)$ is the error vector of the original parameterization (u,v), i.e.:

$$\vec{d}(u,v) = \vec{r}(u,v) - \vec{\eta}(u,v). \quad (22)$$

The solution of equation (20) corresponds to the Euler condition:

$$\frac{\partial}{\partial\vec{w}} ||J\vec{w} - \vec{d}||^2 = 0 = 2J^TJ\vec{w} - 2J^T\vec{d} \quad (23)$$

from which the new optimal parametrization is computed:

$$\vec{w} = (J^TJ)^{-1}J^T\vec{d}. \quad (24)$$

where $J^T\vec{d}$ corresponds to the projection of the error vector on to the tangent plane. This new parametrization is not optimal for A since the model A is optimal for the parametrization (u,v). Using the new parametrization found, $(\hat{u},\hat{v})$, one must compute the new optimal model A^* using equation (17). This iterative process is carried out until the orthogonality condition has reached the desired precision ϵ, i.e.

$$|\frac{\pi}{2} - \arccos\frac{\vec{d}\cdot\frac{\partial\vec{\eta}}{\partial u}}{|\vec{d}||\frac{\partial\vec{\eta}}{\partial u}|}| \le \epsilon, \quad |\frac{\pi}{2} - \arccos\frac{\vec{d}\cdot\frac{\partial\vec{\eta}}{\partial v}}{|\vec{d}||\frac{\partial\vec{\eta}}{\partial v}|}| \le \epsilon. \quad (25)$$

The procedure is the following:

1. Set the initial conditions equal to: $L = L_{\max}$, and $\epsilon = \epsilon_{thr}$.

2. Compute the model A for the parametrization u, v.

3. Compute the Jacobian matrix J

4. For each parametric coordinates (u_i, v_i) compute the following:
   - $\vec{d}(u_i, v_i) = \vec{r}(u_i, v_i) - \vec{\eta}(u_i, v_i)$.
   - $\vec{w}_i = (J^T J)^{-1} J^T \vec{d}(u_i, v_i)$
   - $(u_i, v_i) = (u_i, v_i) + \vec{w}_i$
   - $\gamma_i = \arccos \frac{\vec{d}_i \cdot \frac{\partial \vec{\eta}(u_i, v_i)}{\partial u}}{|\vec{d}_i|| \frac{\partial \vec{\eta}(u_i, v_i)}{\partial u}|}$
   - $\xi_i = \arccos \frac{\vec{d}_i \cdot \frac{\partial \vec{\eta}(u_i, v_i)}{\partial v}}{|\vec{d}_i|| \frac{\partial \vec{\eta}(u_i, v_i)}{\partial v}|}$

5. Compute $\gamma = \max_{i=0,\bar{m}}(\gamma_i)$ and $\xi = \max_{i=0,\bar{m}}(\xi_i)$

6. **If** $|\frac{\pi}{2} - \gamma| \leq \epsilon$ **and** $|\frac{\pi}{2} - \xi| \leq \epsilon$ **or** maximum number of iterations L is reached **then** stop

7. **else goto step 2**

The convergence rate of such a procedure is quite fast. One can see in Figures 2a the variation of average angle between the surface normal and the error vector as a function of the number of iterations n for an initial surface defined as a sphere of radius equal to 4 mm where a Gaussian noise of variance equal to 0.3 mm and zero average was added to the data. The corresponding signal to noise ration is equal to 33 db.

One can also see in Figure 2b the variation of the corresponding mean square error. It is noticeable that after only four iterations the average approximation error is reduced by half. In order to illustrate the invariance of the method to viewpoint one have generated and artificial surface corresponding to a cylinder of radius equal to 4.0 mm. The surface was sampled using fixed x and y which produce a non-uniform parametrization of the surface, i.e. the arc length of two consecutive points is not the same every where. One can see in Figures 3a and 3b the variation of the approximation error as a function of the position of the center of an approximation window of size 7×7. Figure 3a corresponds to the intrinsic method and Figure 3b to the uniform parametrization method. Since the surface distance between each consecutive points is not the same, the approximation error of the uniform parametrization is increased as this distance get larger, especially near the side of the cylinder. Contrary to the uniform parametrization case, the intrinsic parametrization is truly invariant to this distance. Since in most realistic cases, the distance

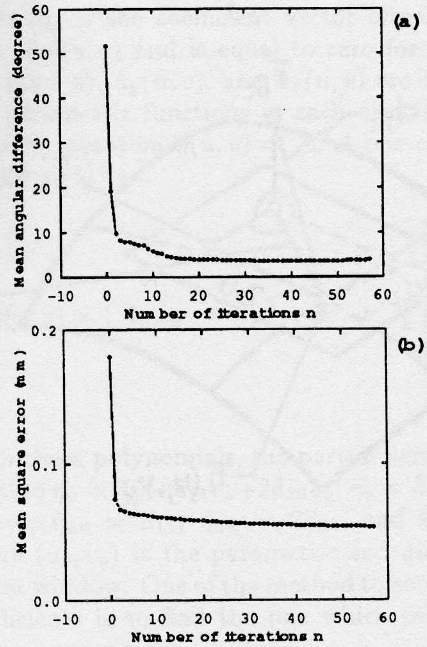

Figure 2: Variation of fitting parameters as a function of the number of iterations n: (a) variation of the mean angular difference between the error vectors and the surface normals, and (b) variation of the corresponding mean square error.

between two consecutive points on a surface is viewpoint dependent, one can conclude that the intrinsic parametrization method is truly viewpoint invariant.

The parametrization produced by this algorithm is the optimal parametrization, defined by Hoschek [10] as the intrinsic parametrization.

4 Experimental Results

In this section, experimental results of mean and Gaussian curvature computation are presented and compared with those obtained by the usual method based on non-weighted least square. One of the best ways to illustrate globally the results is to generate a curvature label sign map as described in [1], where each image pixel is classified into eight categories determined by the sign of the Gaussian and mean curvatures.

One can see in Figure 4a a range image used where the range data are coded in grey levels. Figure 4b illustrates an artificially shaded version of this image. Figure 4c illustrates the curvature signed map produced by an operator with a window of size 11×11 and weight parameters equal to

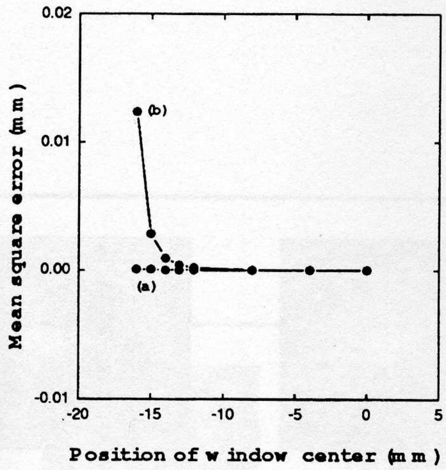

Figure 3: Variation of the approximation error of a local model as a function of surface distance: (a) intrinsic method (b) uniform parametrization method.

$\sigma = 1.5$ and $\beta = 20.0$. Figure 4d illustrates the same map using a standard least-square method. The curvature computation was improved by using the geometrical weights, especially near discontinuities. In fact, one can see in Figure 4e and 4f the average mean-square error of the local surface model for each pixel. Figure 4e corresponds to the intrinsic method and Figure 4f to the uniform parameterization method and are coded in grey levels corresponding to an error range between 0 and 1.0 mm.

To illustrate further the properties of this algorithm a Gaussian noise of variable variance and zero average was added to the test image. One can see in Figure 5a the curvature map for a Gaussian noise with a variance of 1.0 mm and at Figure 5b with a variance of 4.0 mm. The corresponding signal to noise ratio are respectively 22.5db and 10.5db. One can see in Figures 5c and 5d the same map produced by the nonintrinsic method. It can be concluded that for an operator of equal size, the results are approximatively the same in both cases. But, contrary to the uniform parametrization method, the intrinsic method can improve this map by the fact that one can use a larger neighborhood to improve the local fit without the problem of misclassification produced by geometrical discontinuities.

The algorithm has also been tested with impulse noise. One can see in Figure 6a to 6b the curvature map, as computed by the intrinsic method, where 1% and 20% of the pixels were corrupted by the noise. One can see in Figure 6c and 6d the same results using the standard least-square method. In this experiment, the advantages of the method used here over the standard least-square method are clearly demonstrated. One have also performed a series of experiments on more complex range images such as the one illustrated in Figures 7a, 7c and 7e. The Figures 7b, 7d, and 7f corresponds to the curvature label map.

5 Conclusion

In this paper, a new method to compute mean and Gaussian curvatures from range images was proposed. The first and second partial derivatives are estimated by the weighted least-square fit of a biquadratic parametric polynomial within a local moving window. The weights are determined on the basis of surface intrinsic distances and normal angular distances from the center point of the window and the parametrization is computed in order to minimize the error in the direction perpendicular to the tangent plane at this point. It was also demonstrated that a better estimate of the derivatives was obtained by this weighted least- square fit and that the new weighting function allows the method to be very robust to geometrical discontinuities.

Although, an efficient algorithm to find minimum trajectory in a local window was presented, the present algorithm is still computationally more expensive than the usual non-weighted least-square fit. On the other hand, this algorithm allows computation of surface curvatures on a larger neighborhood without prior knowledge of the position of depth and orientation discontinuities that would normally bias these estimates.

Acknowledgments

The authors would like to thank M. Rioux, L. Cournoyer, and J. Domey of the National Research Council of Canada who kindly provided access to the range image database. Publication no. NRCC 37120

References

[1] P.J. Besl and R.C. Jain, "Invariant Surface Charac-

teristics for 3D Object Recognition in Range Images", *Comp. Vision, Graphics, and Image Processing*, 33, 33–80, 1986.

[2] P.J. Besl,"Surface in Range Image Understanding", *Springer-Verlag*, New York, 1988.

[3] P. Boulanger and P. Cohen,"Adaptive smoothing of range images based on intrinsic surface properties," *Proc. of the IEEE Technical Symposium on Optical Engineering and Photonics in Aerospace Sensing*, Orlando, FL, April 16–20,1990.

[4] M.P. do Carmo,"Differential Geometry of Curves and Surfaces," *Prentice-Hall*, 1976.

[5] O.D. Faugeras and M. Hebert,"The Representation, Recognition, and Locating of 3-D Objects", *The International Journal of Robotics Research*, Vol. 5, No. 3, Fall 1986.

[6] T.J. Fan. "Describing and recognizing 3-D objects using surface properties",*Ph.D. thesis, Institute for Robotics and Intelligent Systems*, School of Engineering, University of Southern California, Los Angeles, California, August 1988.

[7] D. Gilbarg and N. Trudinger,"Elliptic Partial Differential Equations of Second Order". 2nd Ed., *Springer-Verlag*, New York, 1983.

[8] E. Guisti,"On the equation of surfaces of prescribed mean curvature: existence and uniqueness without boundary conditions". *Inventiones Mathematicae*, 46, 111–137, 1978.

[9] R.M. Haralick, L.T. Watson, and T.J. Laffey. "The topographic primal sketch", *Int. J. Robotics Res*, 2(1): 50–72; 1983.

[10] J. Hoschek, " Intrinsic parametrization for approximation". *Computer Aided Geometric Design*, 5, 27–31, 1988.

[11] U. Manber, "Introduction to Algorithms, A Creative Approach," *Addison-Wesley*, 1989.

[12] M. Rioux, F. Blais, J.A. Beraldin and P. Boulanger, "Range image sensors development at NRC laboratories", *Proc. of the Workshop on Interpretation of 3D Scenes*, Austin, TX, November 27–29, 1989.

[13] N. Yokoya and M.D. Levine,"Range Image Segmentation Based on Differential Geometry: A Hybrid Approach," *IEEE Trans. on Pattern Anal. Machine Intell.*, Vol. PAMI-11, No. 6, pp. 643–649, 1989.

[14] G. Taubin," Estimation of Planar Curves, Surfaces, and Nonplanar Space Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation", em IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 13, No. 11, November 1991.

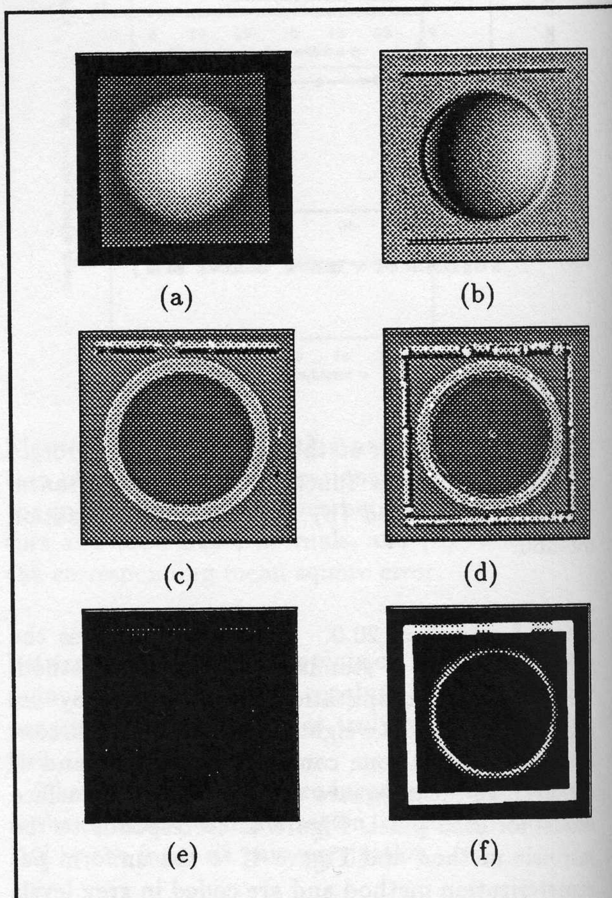

Figure 4: (a) Range image coded in grey levels, (b) artificially shaded image, (c) curvature label map using intrinsic method, and (d) curvature label map using standard mean square method. Approximation error coded in grey levels for a range between zero and 1.0 mm (values higher than 1.0 mm are thresholded): (e) intrinsic method and (f) non-weighted mean-square method.

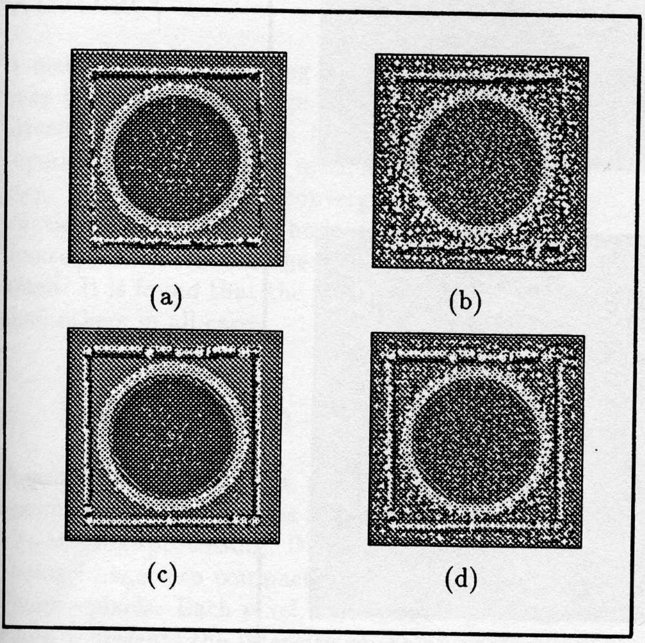

Figure 5: Variation of the curvature label map as a function of noise for each method: (a) map with a Gaussian noise with a variance of 1.0 mm using intrinsic method, (b) map with a Gaussian noise with a variance of 4.0 mm using intrinsic method, (c) map with a Gaussian noise with a variance of 1.0 mm using least square method, (d) map with a Gaussian noise with a variance of 4.0 mm using least-square method.

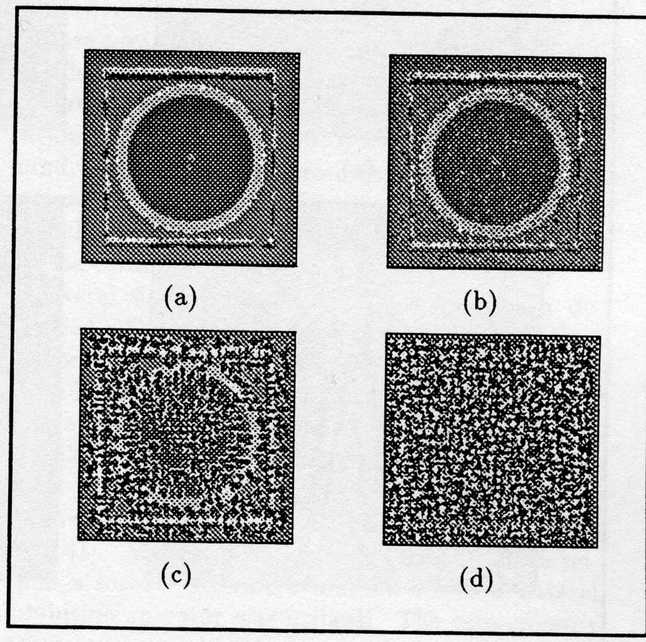

Figure 6: Variation of the curvature label map as a function of noise for each method: (a) map with an impulse noise of 5% using intrinsic method, (b) map with as impulse noise of 20% using intrinsic method, (c) map with an impulse noise of 5% using least-square method, and (d) map with an impulse noise of 20% using least-square method.

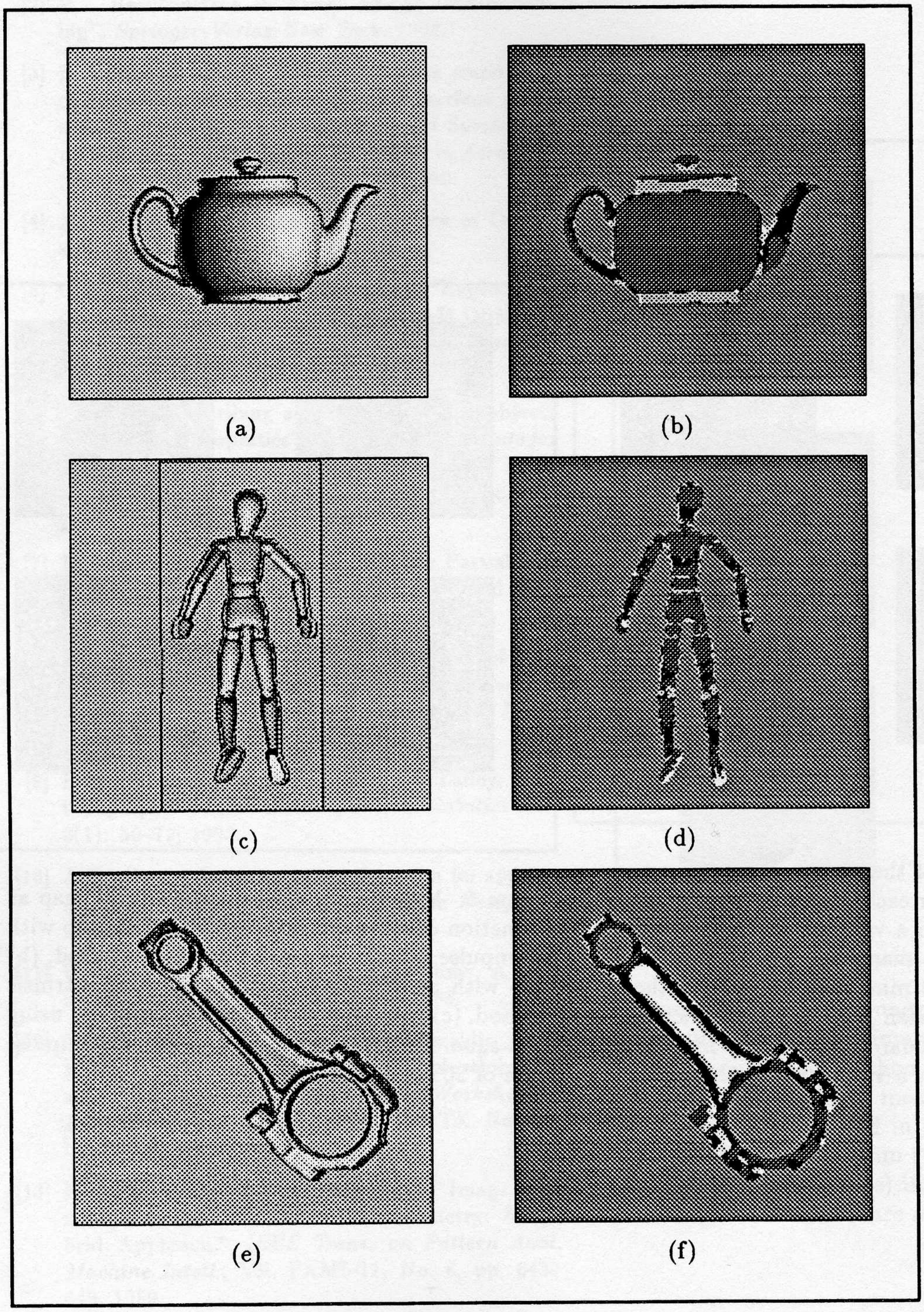

Figure 7: Label map of complex range images. Left side: shaded range images, Right side: label map.