Range Image Segmentation Based on Differential Geometry and Refined by Relaxation Labeling

Gerhard Roth ‡ † Martin D. Levine ‡

† On leave from the Division of Electrical Engineering of the National Research Council of Canada

‡ Computer Vision and Robotics Laboratory
McGill Research Center for Intelligent Machines
McGill University, Montréal, Québec

Abstract

Concepts from the field of differential geometry are used to categorize each pixel in a range image as being a member of one of eight classes depending on the sign of the Gaussian and Mean curvature of the pixel's local neighbourhood. Since the local curvature estimates used to provide a class label for each pixel are very noise sensitive, these labels are made consistent with each other by a relaxation labeling process that assumes the range image is a discrete sampling of a piecewise smooth surface. The local neighbourhoods used during the curvature estimation and relaxation process are shifted away from discontinuities in order to produce more accurate results. Connected pixels with the same label can be grouped together to produce a segmentation of the range image into regions of homogeneous curvature. The result is a segmentation of the range image into a small number of regions which have the same curvature class. This segmentation can then be used as input to a matching processes in an object recognition system or to a viewpoint integration process in an object reconstruction system.

1. Introduction

One of the key problems in computer vision is that of segmentation. To segment an image is to extract from it a symbolic description that is useful for later processing. Segmentation has long been attempted on intensity images [13]. The difficulty with these approaches is that the contents of such images vary considerably with the ambient lighting and the surface reflectivity of the objects in the scene. These dependencies are removed if the direct depth from the sensor to the objects in the scene is available since this information is a function only of the scene geometry. Such depth data is called range data and can be obtained by active sensors, such as lasers and structured light [8], or by passive sensors such as stereo vision [7, 10]. The passive methods have not lived up to expectations, and as a result interest has increased in active range finding methods. These produce a dense map of the range of the objects in the scene to the sensor, which is appropriately called a range image.

We segment range images obtained from a laser rangefinder [12] by classifying each pixel into one of eight classes depending on the sign of the local Gaussian and Mean curvature [4, 9]. These curvatures are differential geometric quantities that completely characterize the surface locally in the sense that knowing them at each point enables the surface to be reconstructed. They are also invariant to viewpoint and have been used by a number of researchers as a criterion for grouping pixels together into regions [1, 2, 5]. However, there are two difficulties with this approach. The first problem is that since the Gaussian and Mean curvatures are based on second order derivatives computed with discrete data, they are very noise sensitive. If the assumption is made that a range image is a discrete sampling of an underlying piecewise smooth surface, then the curvature should not change dramatically over a small area. This simple principle enables relaxation compatibilities to be defined between pixels of different curvature classes in a neighbourhood [14, 3]. These compatibilities are then used to make initial curvature estimates more consistent with their neighbours by a relaxation labeling process. The second problem is that the curvatures are not correct if calculated across discontinuities; that is, the surface is not smooth but piecewise smooth. In order to avoid such errors, the windows from which these initial local curvatures are calculated must be shifted away from discontinuities with the degree of the shift depending on the strength of the discontinuity. This same shifting process also occurs for the window that defines the neighbours of a pixel during the relaxation process so that regions separated by a discontinuity do not affect each other. The result is a simple and efficient method of producing a coarse segmentation of the range image into regions of homogeneous curvature which can then be used as input to higher level processes, such as matching in an object recognition system [16], or viewpoint integration in an object reconstruction system.

2. Computing H and K For Range Images

Since a range image is taken from a single viewpoint it is not a general surface in $\mathbf{R}^3$. Instead it is a graph, sometimes called a Monge patch. This means that the surface S defined by this graph can be written as follows:

$$S = \left\{(x,y,z) : \ x, \ y, \ z = f(x,y) \ where \ (x,y) \in D \subseteq \mathbf{R}^2\right\} \tag{1}$$

From this definition of the surface the following formulae for the Gaussian (K) and Mean (H) curvature can be computed

[4, 9]:

$$H = \frac{(1+f_x^2)\,f_{yy} - 2\,f_x\,f_y\,f_{xy} + (1+f_y^2)\,f_{xx}}{2\,(1+f_x^2+f_y^2)^{3/2}} \quad (2)$$

$$K = \frac{f_{xx}f_{yy} - f_{xy}^2}{(1+f_x^2+f_y^2)^2} \quad (3)$$

These relationships hold as long as the surface S is a graph and $f(x,y)$ has at least C^2 continuity.

The difficulty with the above equations is that the actual range data consist of a set of discrete points. This means that the function $f(x,y)$ is not given explicitly and the derivatives $f_x, f_y, f_{xy}, f_{xx}, f_{yy}$ must be calculated numerically. If this is done directly by using finite differences the effects of noise reduce accuracy. For this reason the methods used to compute the derivatives generally involve the following two steps:

(1) Consider a local window centered around each pixel. Fit some function $f(x,y)$ to the data that has at least C^2 continuity in this window by minimizing some error norm.

(2) Compute the closed form expressions of the required partial derivatives of the function $f(x,y)$. Apply these formulae to the particular $f(x,y)$ found in the previous step in order to compute the estimate of the partial derivative at each pixel.

This has the effect of smoothing the data to a degree dependent on the size of the window which makes the results more accurate by decreasing the effects of noise. Once the partial derivatives are available, equations (2) and (3) can be used directly to compute H and K.

This basic method is widely used [1, 2, 3, 15, 17]. The latter may differ in the approximating function and error norm for which there are many possible options. If the approximating function is a quadric polynomial then we have the required C^2 continuity. If the error norm is L^2, then the entire fitting process can be done efficiently by convolving the image with the appropriate masks. We use this approach for our range images. For each pixel we find the qaudric with the least mean square error E^2 on a window of size 5 by 5 centered at that pixel.

The most serious problem with the previous approach is that the partial derivative estimates are meaningless if done across a discontinuity. If all the discontinuities could be easily found this problem would disappear. However, finding them reliably is not a simple task. Since we are not seeking a perfect segmentation we should be able to deal with this issue without having to perfectly isolate the discontinuities. One possible solution comes from the observation that the mean square error of the quadric fit at each pixel increases in the neighbourhood of a discontinuity [11]. Therefore, if the mean square error (E^2) for all the windows that overlap a given pixel were computed, the window with the smallest E^2 would be the least likely to contain a discontinuity and the most likely to give the best estimates of the Gaussian and Mean curvatures. The quadric associated with this shifted window is then used to compute the Gaussian and Mean curvatures for the current pixel, which is now offset from the origin of this quadric. More formally, if this offset is denoted as (u,v), then for each pixel the best window is the one which provides a minimum fitting error as determined by:

$$E^2(x-u, y-v) = Min\left\{E^2(k,l) : \ (k,l) \in W(x,y)\right\} \quad (4)$$

In this notation $W(x,y)$ is the set of all the windows that overlap the pixel at location x, y of the image. This approach prevents curvature calculations from crossing a discontinuity by shifting the window at a pixel away from any discontinuities. However, while this shifting process avoids abrupt discontinuities such as the edges of polyhedral objects it tends to find spurious discontinuities in objects with curved surfaces.

The problem with this idea, which is described in [17], is that in its current form this shifting process moves away from discontinuities which are not significant. This is because the Min operation in equation (4) forces a shift of the same distance regardless of the magnitude of the difference in E^2 (which we shall call ΔE^2). Since this magnitude is an indication of the significance of the discontinuity, we may counteract this tendency by shifting each pixel by an amount u^*, v^* where $u^* = u \times (\Delta E^2/\Delta E^2_{max})$ and $v^* = v \times (\Delta E^2/\Delta E^2_{max})$. We refer to this as the Compensated Shift Method. What this achieves is to scale the window shift by the ratio of ΔE^2 to the ΔE^2_{max} where ΔE^2_{max} is a threshold above which this pixel is truly a discontinuity. The question is how should ΔE^2_{max} be selected. There is no perfect answer to this perennial problem in computer vision. For now we have set ΔE^2_{max} to 1000 since this gives good results for the images we are processing.

If the actual ΔE^2 reaches the ΔE^2_{max}, then the maximum shift (the original u and v) is allowed. If ΔE^2 is insignificant at a pixel, then the discontinuity is small and the shift away from it will be small; if ΔE^2 is significant, the shift away from it will be large. The results of applying this rule during the fitting of the quadric at each pixel is shown in Figure 1. Applying equations (2) and (3) with no shift, we observe considerable distortion at edges (the black areas). However, with the shifting windows in effect, curvatures are not calculated across significant discontinuities as seen by the removal of blurring on the edges of the polygonal objects. Also, there are few spurious shifts on the surface of the smoothly varying mask.

3. Refining H and K by Relaxation

It is generally conceded that local curvature estimates are noisy since they are based on second order derivatives. A larger window size for the quadric fit produces a better signal to noise ratio for K and H but also reduces the locality of K and H [3]. In order to produce a reasonable segmentation, these estimates must be made more consistent with each other and for this purpose we use a relaxation labeling scheme [6]. Relaxation is an iterative process that updates the degree of belief that a node is a member of a particular class by the support given to this assertion by the neighbours of the node.

For the problem at hand, a node is a pixel of the range image and a neighbourhood is a square window W of a size N_{win} centered on this pixel. Note that this window size is not equal to that used for the curvature calculations. In our experiments N_{win} is set to 3 for the relaxation and 5 for the

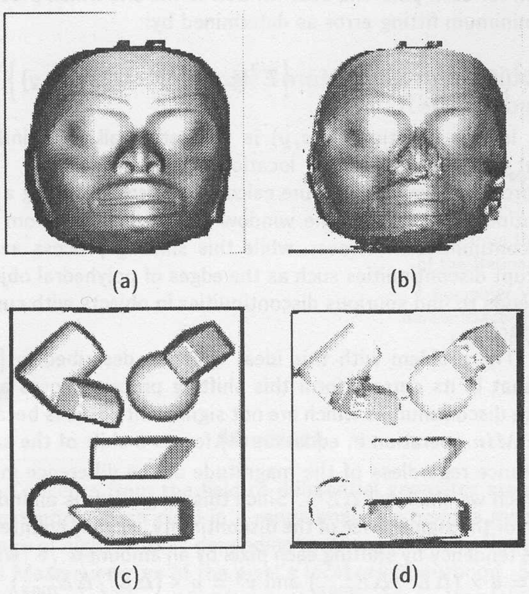

Figure 1 Effect of shifting windows away from a discontinuity during curvature calculations :(a) and (c) No Shift;(b) and (d) Compensated Shift:

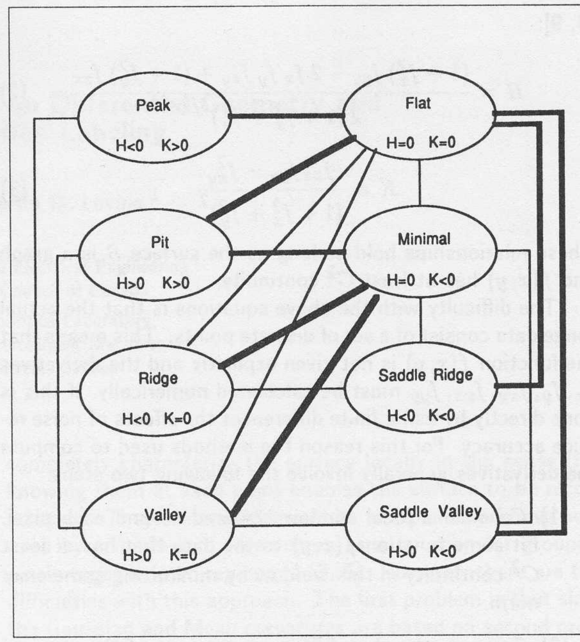

Figure 2 Allowable transitions in H and K. Light solid lines are sign changes in one of H or K: Heavy Solid Lines are Transitions in both H and K

curvature calculations. The possible labels for a node are the eight curvature classes defined by the signs of K and H [1]. More precisely, the Gaussian and Mean curvature values are classified into one of $(+,0,-)$ depending on whether the values are $(>,=,<)$ some thresholds K_{zero} and H_{zero}. Since the combination $K>0, H=0$ is impossible, this produces eight possible curvature classes instead of nine. In the notation of a relaxation labeling scheme, $p_k(\lambda_i)$ is the degree of belief that pixel k adopts a curvature class λ_i, where λ_i is one of these eight classes. During the relaxation it must be true that $\sum_{i=1}^{n} p_k(\lambda_i) = 1$ and $0 \leq p_k(\lambda_i) \leq 1$ for all classes λ_i and pixels k.

The question is what should the compatibility matrix be for this application? The answer comes from the underlying assumption that we are dealing with a smooth, almost everywhere continuous surface. This means that the surface curvature should change gradually in the local window W centered at each pixel. One possible interpretation of this statement is to require that K and H pass through zero when varying from $(-)$ to $(+)$ or from $(+)$ to $(-)$. In this way, any two pixels in the relaxation window which are of opposite sign in K and H are not compatible with each other. The allowable transitions in curvature of neighbouring pixels are shown in Figure 2 and from these transitions two possible compatibility matrices can be derived. The first allows differences in *both* H and K between two pixels. This compatibility matrix is identical to the one described by Boulanger [3] who used the standard continuous relaxation approach [6] to solve the same problem, that of refining the initial estimates of the curvatures. The second is a subset of the first matrix and allows differences in one of H or K between two pixels, but disallows differences in both H and K. The second compatibility matrix does not favour the flat curvature class as much as the first matrix and is the one that we have used.

There is a number of difficulties with using a strictly continuous relaxation method to achieve local curvature consistency. The first is the question of how to find the initial estimates of $p_k(\lambda)$. One idea is to set them in proportion to the difference between each pixel's H and K value and H_{zero} and K_{zero}. This method was used by Boulanger and seems to give good results, but the choice seems rather arbitrary. The second and more serious problem with purely continuous relaxation is that the entire process is very slow because a $p_k(\lambda)$ must be updated for each of the eight possible λ's (curvature classes) at every iteration for each pixel in the range image.

We have simplified the relaxation process by restricting $p_{k(\lambda)} \in \{0,1\}$ at all times. This means that only the label of the most consistent curvature class is retained for each pixel. With this approach $p_k(\lambda)$ is initially set equal to 1 where λ is the curvature class computed by the local quadric fit as decided by the K_{zero} and H_{zero} thresholds, and all the other $p_k(\lambda)$ are set to zero This relaxation is more efficient than a purely continuous one since only one piece of information must be stored for each pixel, but as we shall see it still retains the flavour of a continuous relaxation.

This type of relaxation can be understood by describing the support function and updating rule in a more formal fashion. Consider a vector **n** defined as $\mathbf{n} = (n_k(\lambda_1), \ldots, n_k(\lambda_8))$ where $n_k(\lambda_k)$ is the number of neighbouring pixels in the window W centered on pixel k that have curvature class λ. The computation of the support at pixel k for curvature class λ is given by $S_k(\lambda)$ which is computed by

$$S_k(\lambda) = \sum_{\lambda'=1}^{8} r(\lambda, \lambda') n_k(\lambda') \qquad (5)$$

where $r(\lambda, \lambda')$ is an element of the second compatibility matrix that we have described. This is a simpler form of the standard support function and takes into account the fact that the compatibility depends only on the curvature classes of the neighbouring pixels and does not vary with their position in the window. $S_k(\lambda)$ is still the support given by all the pixels in window W to the belief that pixel k has curvature class λ, but it is always an integer quantity. The standard updating rule for relaxation labeling is then modified in order to guarantee that $p_k \in \{0, 1\}$ for all λ. That is, we take as the new curvature class of the current pixel the one with the highest support from its neighbours:

$$p_k(\lambda)^{(K+1)} = \begin{cases} 1 & \text{when } S_k(\lambda) = max[S_k(1), \ldots, S_k(8)] \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Note that unlike the standard updating rule, this one does not use the value of $p_k(\lambda)^{(K)}$ to compute $p_k(\lambda)^{(K+1)}$ since the current curvature class of the pixel is ignored. This relaxation process will be called Boulanger's relaxation since the first compatibility matrix we described was used by Boulanger [3] for a continuous relaxation algorithm.

The approach described by Sander [14] to refine the surface trace points was actually the first to use this modified form of relaxation algorithm. However, Sander uses rules for computing $S_k(\lambda)$ and $p_k(\lambda)$ that do not require a compatibility matrix, but these rules are still based on the principle that locally the curvature changes very slowly. The original version of Sander's algorithm was designed for Gaussian curvature only, and for comparison purposes we have modified it to deal with both Gaussian and Mean curvature. In this relaxation method, which we will call Sander's relaxation, one can think of the support function $S_k(\lambda)$ as being set to $n_k(\lambda)$, the number of pixels in the window W at pixel k that belong to curvature class λ. Thus:

$$S_k(\lambda) = n_k(\lambda) \text{ for all } \lambda \quad (7)$$

The resulting updating rule is dramatically different from the standard one and contains the information that would normally be in the compatibility matrix. It is defined as follows:

$$p_k(\lambda)^{(K+1)} = \begin{cases} 1 & \text{if } S_k(\lambda) \geq N_{win}^2/2 \\ & \text{or } \lambda = Parabolic \text{ and} \\ & S_k(Elliptic) + S_k(Hyperbolic) > \frac{N_{win}^2}{2} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

In simple terms, it sets the pixel curvature class to the same as the majority of its neighbours, if such a majority exists. If there is no majority, it forces the pixel curvature to be parabolic if the majority of the neighbours classes are split between elliptic and hyperbolic. Note that as in Boulanger's method, the current curvature class of the pixel is ignored when computing the new curvature class. This tends to enlarge parabolic regions at the expense of planar regions. The justification that Sander provides for this is that because of their particular perceptual importance, parabolic regions should be emphasized.

Both of these relaxation schemes are based on the underlying assumption that the range data are obtained from a discrete sampling of an almost continuous surface, and this assumption clearly fails at discontinuities. For this reason we do not center the window W used to define the neighbours during the relaxation process at the pixel. Instead we offset this window from the pixel by the same (u, v) offset (scaled appropriately for the different N_{win} sizes) computed during the curvature calculation process. This has the effect of stopping the relaxation process from propagating across discontinuities. Since we have already computed these (u, v) offsets, there is no extra cost in using them in the relaxation algorithm.

4. Experimental Results

A number of range images were processed using this algorithm and the results for two of these images are shown in Figures 3 and 4.

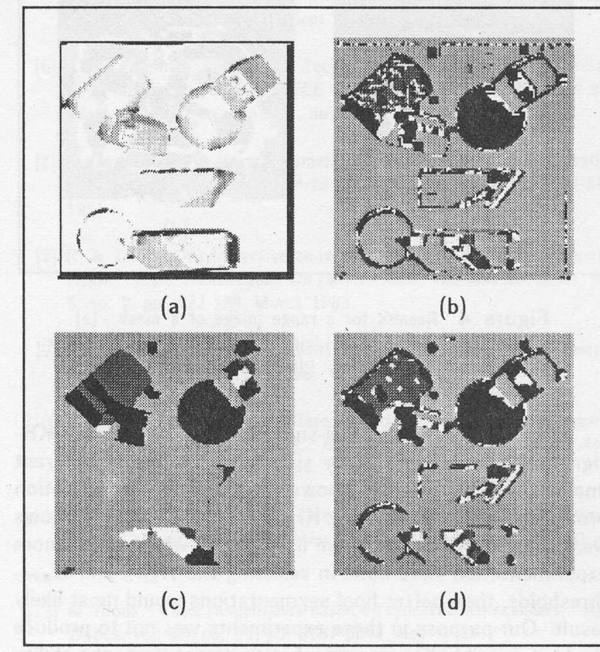

Figure 3 Results for a range image of simple objects: (a) Shaded range image; (b) Initial KH-sign map; (c) Boulanger relaxation KH-sign map; (d) Sander relaxation KH-sign map.

For both images, the K_{zero} and H_{zero} thresholds were selected empirically by choosing a small portion of the flat background and computing the minimum K and H values necessary to classify this area as flat. Since these images have very little noise, this method produces reasonable results, but would fail on noisier images. The first image is of a mixed scene containing polyhedra, cylinders and spheres, while the second image is of a wooden mask. The examples demonstrate that this segmentation method maintains the sharp edges of polyhedral objects since they are avoided by the shifting window process. The areas that are flat have no (or little) indication of their edges in the final KH-sign map. This is because even thought the flat regions contain discontinuities, all the parts necessarily have the same curvature sign Thus additional processing must be done to extract this edge information from these regions (see [17]). Both images show the considerable improvements obtained for the KH-sign map by the relaxation process which was arbitrarily stopped after four iterations.

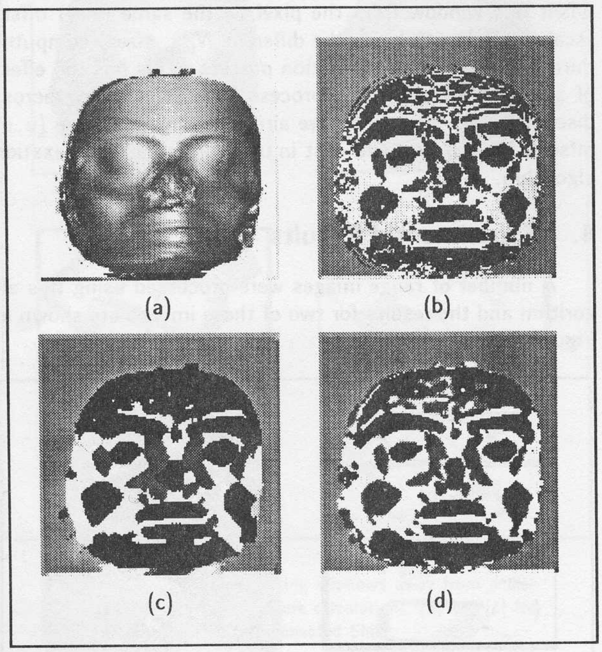

Figure 4 Results for a range image of a mask: (a) Shaded range image; (b) Initial KH-sign map; (c) Boulanger relaxation KH-sign map; (d) Sander relaxation KH-sign map.

In figure 5 the initial KH-sign map, and the relaxed KH-sign map are shown side by side for a number of different images. The result again shows the ability of the relaxation process to clean up the noisy KH-sign map. If larger windows were used in the initial fit (we used 5 by 5 windows) or more experimentation were done in selecting the K_{zero} and H_{zero} thresholds, then better final segmentations would most likely result. Our purpose in these experiments was not to produce the best possible KH-sign map but to demonstrate the ability of the relaxation process to produce reasonable segmentations from noisy input.

Both relaxation methods, Sander's and Boulanger's, do a good job in cleaning up the initial curvature estimates. The method of Boulanger tends to converge faster, while the method of Sander tends to be slower and creates more parabolic regions. In fact, both methods produce very similar results which is not surprising considering the fact that they are based on the same principles. As we have discussed in Section 3, during the relaxation process we offset the window which defines the neighbours of a pixel in the direction away from a discontinuity. This has the desirable effect of keeping the regions from spreading across a discontinuity, which can be observed in Figure 6. If this shifting process is not invoked, the regions produced by the relaxation process tend to bleed across discontinuities and become distorted.

5. Discussion and Conclusions

The method which uses only the signs of K and H has been shown to produce useful segmentations if the initial KH-

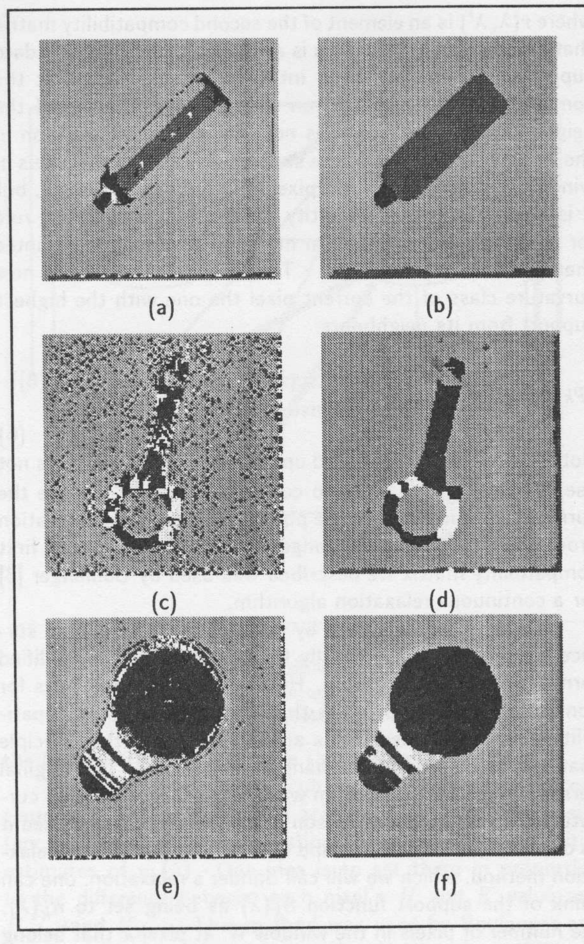

Figure 5 Results for a number of different parts: (a) , (c) and (e) Initial KH-sign map; (b) , (d) and (f) Boulanger relaxation KH-sign map.

sign map is further refined using a relaxation labeling scheme. The main problem with this approach is selecting the K_{zero} and H_{zero} thresholds. The reason this is particularly important is that the boundaries between regions of different curvature classes are defined by these thresholds. Thus, these boundaries are not actually discontinuities, but are lines on the surface of the object with a specific Gaussian and Mean curvature. This makes them very difficult to localize and very sensitive to small changes in these thresholds. This should be compared to discontinuities such as jumps or creases which are easier to localize and are present over a wider range of thresholds because they correspond to physical events on the surface. There is no obvious way of automatically selecting an optimum threshold for K_{zero} and H_{zero}. However, when we use a relaxation process to further process the curvatures we still obtain a useful segmentation.

One thing that could be done with the segmentation is to represent it by a graph. The nodes in such a graph would be the regions and their properties, while the arcs would be the adjacencies between regions. Such a graph could be used as an object model so that new graphs obtained from range images of unknown objects could be matched against these models in order to perform object recognition [16]. Of course,

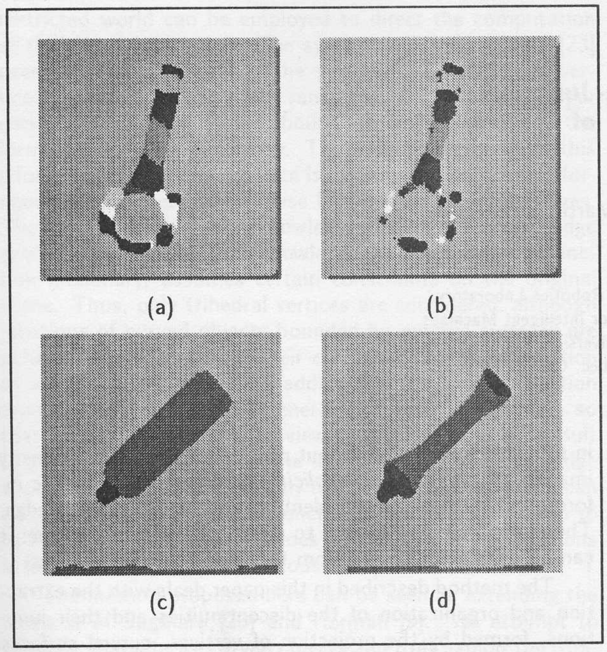

Figure 6 Effect of shifting windows away from a discontinuity during the relaxation: (a) and (c) with shifting; (b) and (d) without shifting

the matching program must be able to deal with noisy data because of the approximate nature of the segmentation, but this will always be the case regardless of the segmentation method used.

To conclude, we have used the KH-sign map to classify each pixel in a range image into one of eight curvature classes. The initial KH-sign map is refined by a relaxation labeling process which makes these curvatures consistent with each other. During the initial curvature calculations and during the relaxation process the local neighbourhoods of each pixel are shifted away from discontinuities to produce more accurate results. The segmentation can be used to create a set of regions, where each region consists of pixels of the same curvature class. As pointed out by Boulanger, this segmentation algorithm can be implemented efficiently on an array processor and is thus potentially useful for situations where a number of range images must be processed quickly. Other relaxation methods [15] might achieve more accurate results but at the cost of taking much more time. The segmentation into regions of homogeneous curvature is not perfect, but is good enough to provide useful information to a matching algorithm. Our experiments show that this is indeed the case.

Acknowledgements

The authors wish to acknowledge the National Research Council of Canada for providing the range images. M.D. Levine would like to thank the Canadian Institute for Advanced Research for its support and G. Roth would like to thank the the National Research Council of Canada for providing education leave to pursue graduate studies. This work was partially supported by the National Sciences and Engineering Research Council of Canada under Grant G1840.

References

[1] P. J. Besl, and R. C. Jain, "Invariant surface characteristics for 3d object recognition in range images," *Comput. Vision, Graphics, and Image Processing*, vol. 33, no. 1, pp. 33-80, Jan. 1986.

[2] P. J. Besl, and R. C. Jain, "Segmentation through variable order surface fitting," *IEEE Transactions On Pattern Anal. and Machine Intell.*, vol. 10, no. 2, pp. 167-192, March 1988.

[3] P. Boulanger, "Label relaxation technique applied to the topographic primal sketch," *Vision Interface 88*, pp. 426-431, June 1988.

[4] M. P. DoCarmo, *Differential Geometry of Curves and Surfaces*. Englewood Cliffs: Prentice-Hall, 1976.

[5] R. Hoffman, and A. K. Jain, "Segmentation and classification of range images," *IEEE Transactions On Pattern Anal. and Machine Intell.*, vol. 9, no. 5, pp. 608-620, Sept. 1987.

[6] R. A. Hummel, and S. W. Zucker, "On the foundations of relaxation labeling processes," *IEEE Transactions On Pattern Anal. and Machine Intell.*, vol. PAMI-5, no. 3, pp. 267-287, May 1983.

[7] K. Ikeuchi, and B. Horn, "Numerical shape from shading information and occluding boundaries," *Artif. Intell.*, vol. 17, no. 2, pp. 141-184, 1981.

[8] R. A. Jarvis, "A perspective on range finding techniques for computer vision," *IEEE Transactions On Pattern Anal. and Machine Intell.*, vol. 5, no. 2, pp. 122-139, March 1983.

[9] M. M. Lipschutz, *Theory and Problems of Differential Geometry*. New-York: McGraw-Hill, 1969.

[10] Y. Ohta, and T. Kanade, "Stereo by intra- and inter-scanline search using dynamic programming," *IEEE Transactions On Pattern Anal. and Machine Intell.*, vol. 7, no. 2, pp. 139-154, March 1985.

[11] T. Pavlidis, "Discontinuity detection for visual surface reconstruction," *Comput. Vision, Graphics, and Image Processing*, vol. 30, no. 3, pp. 316-330, 1986.

[12] M. Rioux, "Laser rangefinders based on synchronized scanners," *Appl. Optics*, vol. 23, no. 21, pp. 3837-3844, Nov. 1984.

[13] A. Rosenfeld, and A. Kak, *Digital Picture Processing, Vols. 1 and 2*. New-York: Academic Press, 1981.

[14] P. T. Sander, and S. W. Zucker, " Tracing Surfaces and Surfacing Traces, I: Stable local estimates and coarse segmentation by curvature," TR-CIM-86-7, Montréal, Québec, Canada: McGill Research Center for Intelligent Machines, October 1986.

[15] P. T. Sander, " On reliably inferring differential structure from three-dimensional images," Ph.D Thesis, Montréal, Canada: Department of Electrical Engineering, McGill University, 1988.

[16] A. Sokalski, " Range-Based object recognition using maximal cliques," Masters Thesis, Montréal, Canada: Department of Electrical Engineering, McGill University, Nov. 1988.

[17] N. Yokoya, and M. D. Levine, "Range image segmentation based on differential geometry:A hybrid approach," *IEEE Transactions On Pattern Anal. and Machine Intell.*, to appear.