Segmentation of 3D Surface Data

Dr Alan M. McIvor

Machine Vision Team, Industrial Research Limited
P.O. Box 2225, Auckland
New Zealand
Ph: +64 9 3034116, Fax: +64 9 3028106
E-mail: a.mcivor@irl.cri.nz
URL: http://kauri.auck.irl.cri.nz/

Abstract

Methods are presented for segmenting dense 3D surface point data into smooth surface patches. The smooth patches are further subdivided into patches of particular surface types. These highly structured patches provide useful cues for identifying objects or locating features of an object.

1 Introduction

3D sensors such as laser stripe scanners and temporally encoded structured light systems generate dense surface position samples [1]. This paper considers a fundamental step in processing this data, that of partitioning the points into sets which correspond to smooth surface patches. The boundaries of a smooth surface patch is formed from three different types of boundary segments: visibility boundaries; occlusion boundaries; and, surface folds. Visibility boundaries mark the limits of the sensed surface regions. Occlusion boundaries occur where part of an object obscures the sensor's view of another. Surface folds are discontinuities in the surface normal, such as at the sharp edges of a manufactured object.

Once smooth surface patches are available, it is useful to further segment them into sub-patches of particular type, e.g, cylindrical surfaces, flat surfaces, etc. Such surface patches form useful features for cues to recognising the object, determining it location, or identifying its size (or other free parameters in an object model).

The rest of paper is organised as follows. Visibility boundary detection is described in Section 2. Occlusion boundary detection is discussed in Section 3, and surface fold detection is discussed in Section 4. Section 5 presents several ways in which the smooth patches can be further subdivided. Finally, a conclusion and discussion of future work is presented in Section 6.

2 Visibility boundaries

Most 3D sensors generate data indexed on a 2D array. Data is not always obtained for each index value, because the surface in that direction is too far away, is a poor signal reflector, or is shadowed by another (when there is a finite baseline separation between

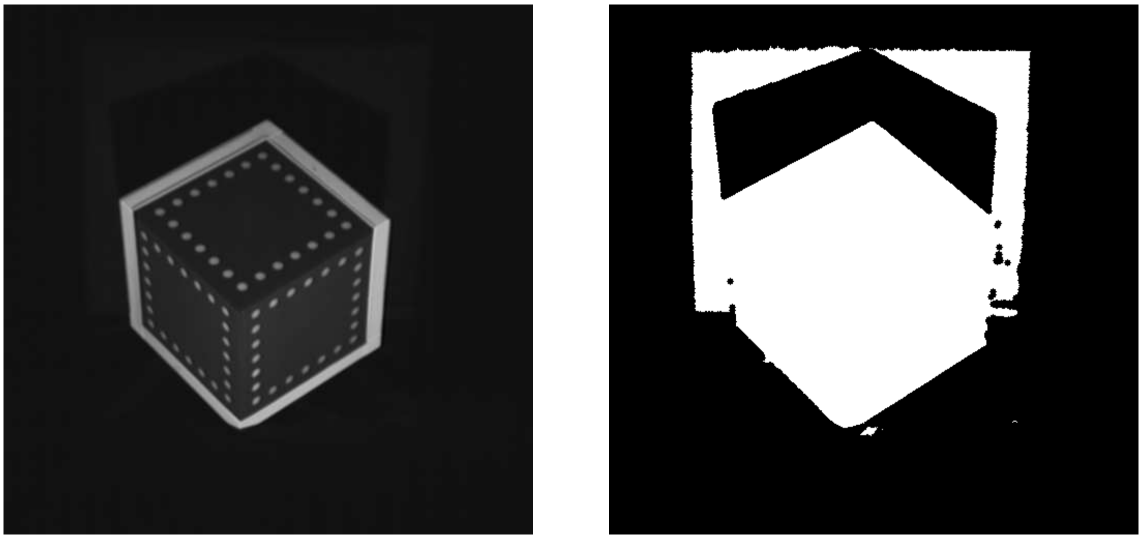

Figure 1: Intensity image of a test cube, and associated mask of valid data points.

scanner components), etc. Thus, associated with the data array is a binary image denoting at which locations valid data was obtained. The borders of the regions consisting of valid-data locations can be extracted using boundary following algorithms [2, Chapter 7]. The corresponding 3D points form the visibility boundaries, as shown in Figure 2.

Since visibility boundaries can pass over occlusion boundaries, the 3D data points corresponding to neighbouring indices on a boundary may be from different surfaces and hence be widely separated in 3D space. It is possible to detect such discontinuities and not include the link between the data points. However, if they are kept, they match up with occlusion boundaries to form complete contours around a virtual surface formed by the occlusion. It is known that the true surface must lie on one side of this virtual surface. The virtual surfaces are useful in building up closed surface models [3], and for planning further viewpoints for building up complete surface models.

3 Occlusion boundaries

An occlusion boundary occurs when a surface patch occludes another patch, i.e., the 3D data points corresponding to neighbouring index locations are from separate surfaces at different distances from the viewpoint.

Occlusion boundaries are detected as follows. First the surface normal at each data point is estimated [4]. The estimator uses a finite patch of data around each data point. When such a patch crosses an occlusion boundary, the data will be dominated by the virtual surface formed by the segments of the lines of sight from the front surface at the occlusion boundary to the back surface, as in Figure 2. Consequently, the surface normal estimate will be almost perpendicular to the line of sight (at the data points on each side of the occlusion boundary). Thresholding this angle at a value close to $\pi/2$ radians generates "fat line" like regions which contain the occlusion boundaries. These could be thinned using a skeletonization algorithm to localise the occlusion boundary. However, in practice, the 3D data obtained close to an occlusion boundary is of poor quality. Hence, it was found that a better approach than trying to localise the occlusion boundary precisely was to use the pair of lines formed by the borders of the "fat lines" as the occlusion boundary, and discard the information in-between.

Several other approaches to detecting occlusion boundaries have been investigated. One approach is to form a "depth map", which is the distance from the camera focal point to

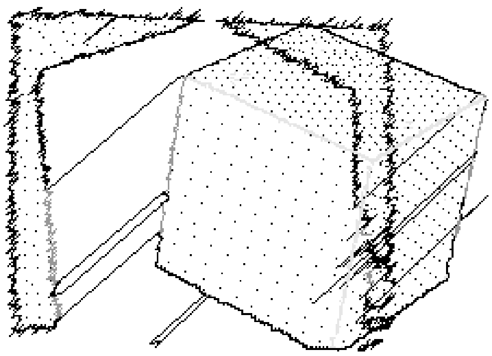

Figure 2: 3D data segmented into smooth regions. Visibility boundaries are drawn in black, occlusion boundaries in mid grey, and folds in light gray. The black dots are a subsampling of the 3D data points. Note the virtual surfaces bounded by visibility boundaries and occlusion boundaries.

the surface point in the case of a structured light system. Some other 3D sensors produce data directly in this form [5]. Then occlusion boundaries can be extracted by applying an edge detector to this single valued image [6, 7]. Looking at the incident angle, as described above, was found to be more robust than depth map discontinuity detection.

4 Surface folds

All surface folds, including sharp edges, will generate data regions of high curvature due to the sampling process, and the finite support operators used to estimate curvature. Thus, the approach taken to detecting surface folds is to look for regions where the magnitude of at least one principal curvature is high, i.e., looking at the value of $\max(|k_1|, |k_2|)$, where the k_i are the principal curvatures. This is illustrated in Figure 3, in which the surface folds are clearly detected. The index locations of the surface folds can be extracted from this image by applying a ridge detector. A simple thresholding scheme was used to extract the surface folds shown in Figure 2. Another approach to refining the location of the fold is to determine the osculating cubic to the surface in its vicinity [8, Page 288].

5 Smooth patch subdivision

The previous sections have described how the 3D point data can be segmented into smooth surface patches. It is useful to further segment these surface patches into sub-patches of specific types. This is considered in this section. The first segmentation uses the classification of points on the basis of the signs of the Gaussian and mean curvature there. The second segmentation is the detection of umbilic regions (i.e., planar and spherical patches).

5.1 Mean and Gaussian curvature sign

Surface points can be classified on the basis of the sign of the mean (H) and Gaussian (K) curvatures into 6 basic types: hyperbolic ($K < 0$); convex parabolic ($K = 0$, $H < 0$); planar umbilic ($K = 0$, $H = 0$); concave parabolic ($K = 0$, $H > 0$); convex elliptic ($K > 0$, $H < 0$); and, concave elliptic ($K > 0$, $H > 0$) [8]. The classification can be determined

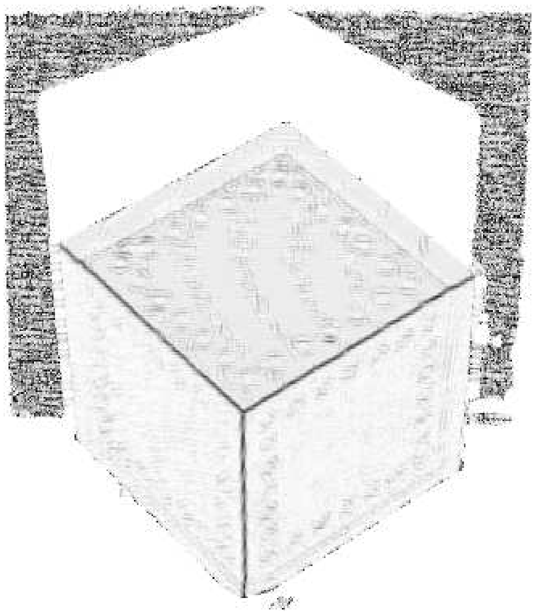

Figure 3: The maximum principal curvature magnitude at each data point.

by comparing the principal curvatures, and their difference, to constant thresholds as in Figure 4. Alternatively, the Gaussian and mean curvatures can be used. In the latter case, the thresholds used must be data dependent so that the equivalent thresholds for each principal curvature are the same [9].

5.2 Umbilic regions

Umbilic points are points at which the principal curvatures are equal. We can define equality by

$$k_1 - k_2 < \epsilon \tag{1}$$

(since $k_1 >= k_2$ by definition). This is equivalent to

$$H^2 - K < \frac{\epsilon^2}{4} \tag{2}$$

The values of $H^2 - K$ at all data points can be thresholded to generate an image of all umbilic points. Such points can be divided into regions of constant curvature. For example, Figure 5 shows the regions labelled by using a Euclidian distance metric over 8-connected neighbourhoods. More sophisticated segmentation algorithms such as the K-Means method or Basic Isodata algorithm [10, Page 201] could be used.

Regions containing umbilic points with all the same value of principal curvature are spherical surface patches. If the curvature is zero, then the region is a planar patch.

6 Conclusion

This paper has described methods for segmenting 3D data into smooth surface patches. These can be formed into a region adjacency graph describing the observed surfaces. The incorporation of the virtual surfaces bounded by visibility and occlusion boundaries is useful because they form a constraint on the unobserved part of the surface, and this can be used in planning further views so as to generate a complete surface model. The smooth surface patches can be further subdivided into sub-patches with particular properties. Two

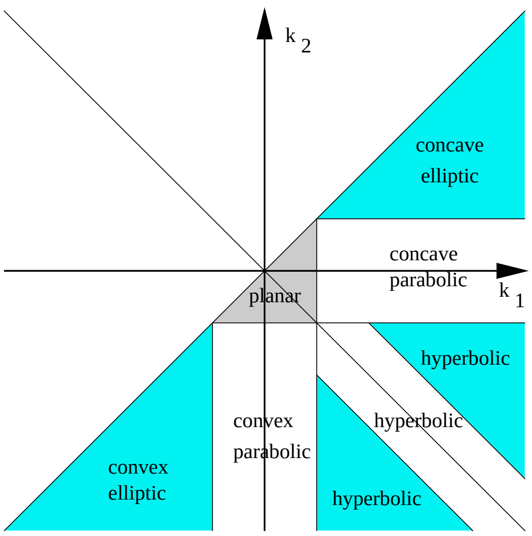

Figure 4: The implementation of the classification by thresholding the principal curvatures. It is assumed that $k_1 > k_2$.

examples were discussed: sub-division on the basis of the sign of the mean and Gaussian curvature; and, the extraction of umbilic (i.e., spherical and planar) regions. Both of these examples use point properties only.

The described segmentation of 3D data consists of 4 layers:

- segmentation of visible surfaces from no-data regions;
- segmentation of visible surfaces into connected patches;
- segmentation of connected patches into smooth patches;
- segmentation of smooth patches into spatially uniform regions.

This provides a basic framework for generating local surface models. Future work will look at other sorts of smooth patch segmentations. In particular, regions defined not only in terms of point properties, but by their distribution as well will be considered. For example, cylindrical regions consist of points with zero Gaussian curvature, with the directions of non-zero principal curvature all aligned.

Visibility and occlusion boundaries are artifacts of the sensing process and sensor position. They do not necessarily correspond to surface features. Only when occlusion boundaries are formed by sharp edges (such as on polyhedra) do they can correspond to object features. In all other cases, the apparent contour forming the occlusion boundary corresponds to a smooth line (a contour generator) on the surface which changes with sensor position. Thus, it is useful to divide occlusion boundaries into two classes, according to whether they were generated by surface edges or contour generators. This will be investigated in future work.

References

1. Ray Jarvis. *Range sensing for computer vision.* In Anil K. Jain and Patrick J. Flynn, editors, Three-Dimensional Object Recognition Systems, volume 1 of *Advances in Image Communications*, pages 17–56. Elsevier Science Publishers, Amsterdam, 1993.

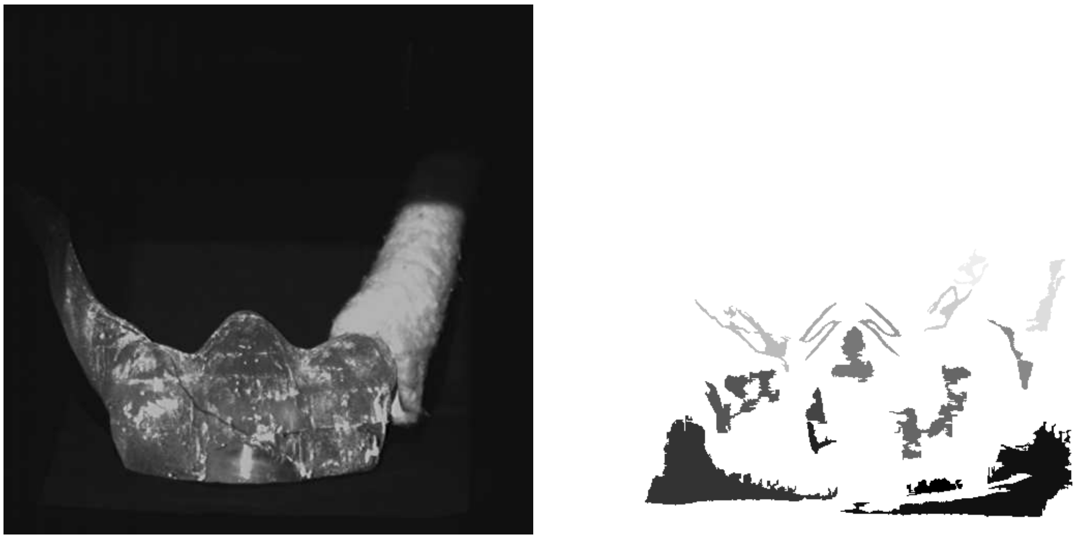

Figure 5: Detected umbilic points segmented into constant curvature regions shown on right from 3D data taken from scene on left.

2. T. Pavlidis. *Algorithms for graphics and image processing.* Computer Science Press, Springer-Verlag, Berlin – Heidelberg, 1982.

3. Adam Hoover, Dmitry Goldgof, and Kevin W. Bowyer. *Extracting a valid boundary representation from a segmented range image. IEEE Transaction on Pattern Analysis and Machine Intelligence*, 17(9):920–924, September 1995.

4. Alan M. McIvor and Robert J. Valkenburg. *Principal frame and principal quadric estimation.* In Image & Vision Computing New Zealand, pages 55–60, Lower Hutt, August 1996.

5. Fred W. DePiero and Mohan M. Trivedi. *3-D computer vision using structured light: Design, calibration and implementation issues. Advances in Computers*, 43:243–278, 1995.

6. Raghu Krishnapuram and Sundeep Gupta. *Edge detection in range images through morphological residue analysis.* In CVPR, pages 630–632, 1992.

7. Satish Kaveti, Eam Khwang Teoh, and Han Wang. *A novel approach for detection of edges in range images using splines.* In MVA'94 IAPR Workshop on Machine Vision Applications, pages 486–489, Kawasaki, Dec 13–15 1994.

8. Jan J. Koenderink. *Solid Shape.* MIT Press, Cambridge, Mass., 1990.

9. Emanuele Trucco and Robert B. Fisher. *Computing surface-based representations from range images.* In IEEE International Symposium on Intelligent Control, pages 275–280, Glasgow, Scotland, 1992.

10. R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* Wiley, New York, 1973.