Range Image Segmentation Based on Decomposition of Surface Normals

Kari Pulli and Matti Pietikäinen
Computer Laboratory
Department of Electrical Engineering
University of Oulu
90570 OULU, Finland

Abstract

A method for segmenting range images into homogeneous regions is presented. The resulting regions are either planar or smooth curved surfaces. Surface continuity is estimated by inspecting the values of normal vector components, which are linearized in respect to angular changes, and scaled depth values. The x- *and* y-*components of the normal vector, and the depth map are treated as three color bands in a color image. The image thus formed is segmented using a hierarchical connected component analysis method, which also allows curved regions to be segmented into one region.*

1 Introduction

The large amount of data present in a range image can be more readily utilized by high-level scene analysis processes if transformed into a more compact intermediate representation. Such symbolic scene description can be obtained by segmenting the image into homogeneous surface patches.

The literature abounds with different approaches to the range image segmentation problem. Besl locates seeds for variable order surface fitting and area growing, e.g, by estimating local curvatures [1], and others extract edges which give information about surface boundaries [2]. There are also methods that combine these methods, gaining robustness from independent redundant information [3][4]. Methods exist that utilize surface normals for segmenting planar surface regions [5] or also curved surfaces [6].

The segmentation method of Taylor *et al.* [5] is a split-and-merge method, where the homogeneity criterion is based on the comparison of two angles describing the normal orientation and the original range value. Merging is based on simple minimum and maximum value comparison of neighboring regions.

Sabata *et al.* [6] use the homogeneity of normal vectors and their three projections onto the xy-plane, the yz-plane, and the zx-plane. After the initial clustering, the method proceeds to refine the clustering iteratively using a pyramidal algorithm. Four independent segmentations are made to form an over-segmented image. These are then merged by higher-level routines (e.g., variable order bivariate polynomial fitting).

We propose here a simple but powerful segmentation method, based on local approximations of normal vectors, that segments a range image into homogeneous surface regions. The image is first split into many regions which are subsequently combined into larger regions. The homogeneity criterion is a comparison of two orthogonal components of normal vectors combined with a depth map. The regions thus formed need not be planar—they can also be curved. The algorithm converts a depth map to local normal vectors, and uses their x- and y-components to detect orientation discontinuities, and the depth map to detect depth discontinuities. These three components are treated as three color bands, and the image is segmented using a connected component analysis [7] method developed for color images [8].

2 Normal decomposition

There are several methods for obtaining local surface normals from range data. The basic approach would be to fit a continuous differentiable function to data and compute its derivatives analytically. If data is clean enough, computationally more efficient methods such as the local quadratic surface least squares (LSQ) approach presented in [1] suffice. Another possibility is to use the slightly more complex local LSQ planar fitting method of Taylor *et al.* [5]. The problem in these methods is that the orientation disconti-

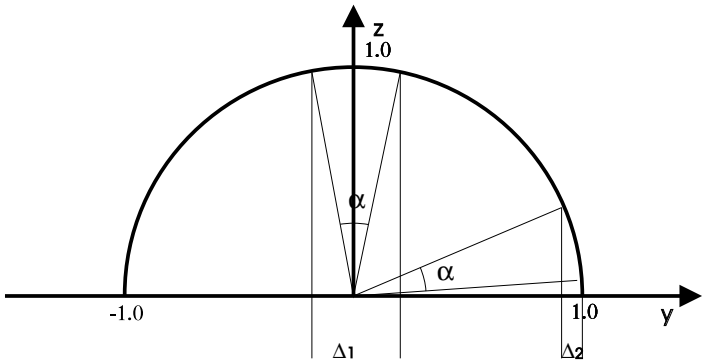

Figure 1: A constant change in the angle does not map to a constant change in the y-component's value.

nuities get blurred. Further, if there is noise present in the measurement data, the data must first be filtered. Gaussian filtering (or approximately Gaussian such as binomial filtering [1]) attenuates Gaussian noise well but blurs sharp edges. Median filtering removes shot noise without blurring edges, but it also makes roof edges flat.

More reliable alternatives to pre-filtering are the so-called robust methods, such as M-estimators [9], iterative reweighting least squares [10], least median squares [11] and least trimmed squares (LTS) [11][4]. Because LSQ methods try to fit a function to all of the data, the outliers that deviate a lot from the bulk of the data pull the fit towards them. Robust methods, on the other hand, fit a function to the majority of the data disregarding outliers.

We chose to use LTS as our principal method for obtaining normal vectors. When estimating normal vectors, the idea is to choose, within a local neighborhood, sets of three measurements (either all the possible combinations or a subset) and determine the planes that fit to those point sets. For each plane we determine the squared residuals or fitting errors (within that same neighborhood) and h of them are summed together ($h = n/2$, n is the size of the neighborhood). The normal of the plane with the smallest trimmed sum of squared residuals is chosen. The normal vectors are normalized, and if a normal vector points away from the view point, it is flipped.

Having calculated the local normal vectors, we deal with the individual components separately. Physically, this corresponds to lighting a scene of Lambertian surfaces with five light sources that are aligned along the coordinate axes — one shines from the positive z, and the rest from positive and negative x and y. You can see this by examining, e.g., the z light: the reflected light is calculated by taking a dot product of the light source direction vector ($\vec{l} = [0, 0, 1]$) with the normal vector. The result is the z-component of the normal vector. With the x and y components the light from the positive direction illuminates only the surfaces where also the corresponding normal component is positive, while the opposing light source illuminates the surfaces with a negative corresponding normal component. The illumination information of both the light sources is combined in one scalar if the normal components (both x and y) are taken as they are.

Because the change of the Lambertian reflectance is unlinear within a curved surface of constant curvature, we deviated from the artificial lighting idea to facilitate the segmentation process. In Fig. 1, the behaviour of the y-component is shown when the x-component equals zero. The horizontal axis represents the y-component, whereas the vertical axis represents the z-component. Now if a normal vector is rotated by a constant angle α, the resulting change in the y-component's value is not constant ($\Delta_1 \neq \Delta_2$). However, a linear mapping can be obtained by applying the *arcus cosine*-function to each component. This results in a smooth transition of the normal components' values for surfaces with a constant curvature.

3 Depth component

We also want to have robust estimates for the depth values. These are obtained using the plane equations calculated by the LTS method, and inserting x and y values corresponding to the measurements into the equation. We also need robust estimates for the depth values' dynamic range. To this end, we tesselate the original data in a number of non-overlapping windows, calculate the median depth in each window, and use

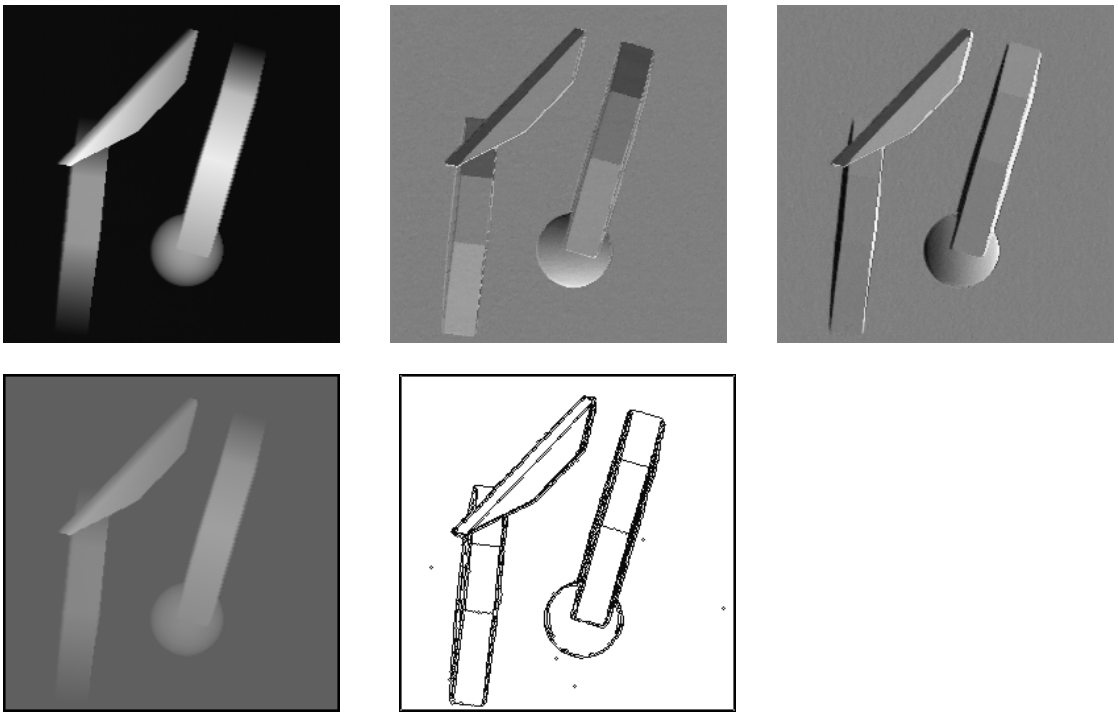

Figure 2: The segmentation of a few blocks using 5×5 operator size (LTS method). From left to right, top to bottom: original depth map, normal x- and y-components, scaled robust depth map, and segmentation result.

the maximum and minimum of these values to approximate the total range of depth values.

The segmentation can proceed once all the components have been obtained. Homogeneous regions are separated by roof edges and step edges. Roof edges can be found where there is a large enough difference between neighboring normal vectors. Instead of comparing each vector to its neighbors, we chose to compare x- and y-components of the normalized normal vectors to the neighbors' respective components separately. The z-component is not needed as it is redundant information (it can be obtained from the x- and y-components). For dealing with step edges, a scaled depth map is included in the comparison as the third component. This causes a difference in the neighboring pixel values over a step edge so that those pixels will be classified as belonging to different regions. The values of the three components are taken as the pixels' three-dimensional coordinates, and the connected component analysis is based on the Euclidean distance between neighboring components. In the case of a large difference between the components, the connected component analysis deduces that those measurements belong to different regions.

Both the x- and y components will be linearized and are then scaled to an integer value $[0, 255]$ and treated as separate color bands. The third band is obtained by scaling also the depth map to $[0, 255]$. The image thus formed can be segmented, for example, using the hierarchical connected component segmentation method developed for color images by Westman *et al.* [8].

4 Segmentation of three-band images

The segmentation procedure in [8] is a robust and iterative connected component analysis method which works by first merging pixels and then regions, depending on their average boundary contrast in the color space. Each successive stage recomputes connected components by a transitive closure of connectivity among adjacent regions, uniquely identifying the resulting maximal connected components. Usually two stages suffice to produce good segmentation results.

The first stage computes the initial image segmentation based on the connectivity of the adjacent pixels, which is determined by testing the differences in their colors. Here, a conventional two-pass algorithm is used. In the first pass, the image is scanned, row by row from top to bottom, and pixels are assigned labels by comparing them with adjacent left- and upper-neighbors using either 4- or 8-connectivity. Adjacent pixels are said to be connected if their color difference

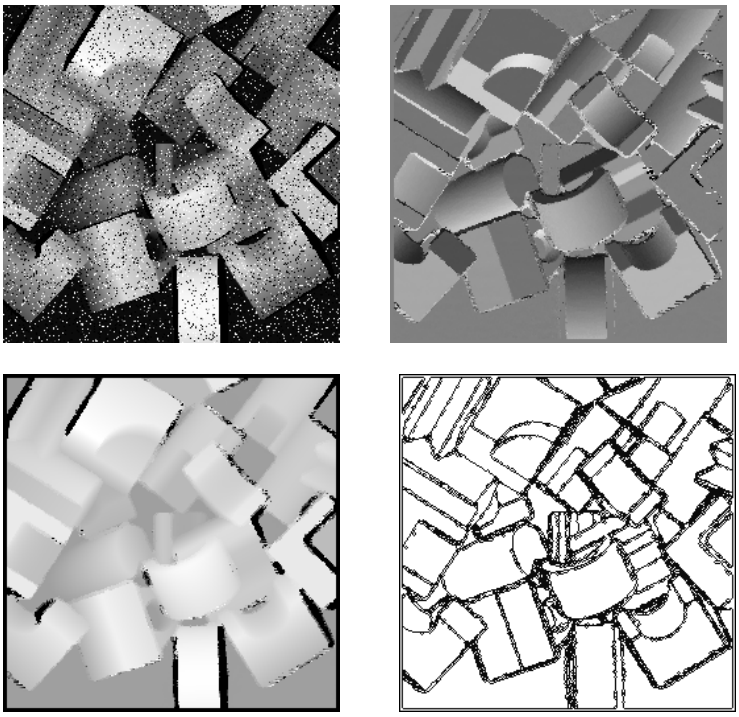

Figure 3: The segmentation of a pile of planar and curved blocks using 7×7 operator size (LTS method). 15% of the data is contaminated by impulse noise. From left to right, top to bottom: original depth map, normal x- and y-components, scaled robust depth map, and segmentation result.

is lower than a pre-determined threshold value, δ. In the second pass, the pixels with component-equivalent labels are re-labelled uniquely.

After obtaining the basic connected components the procedure is iterated in a second stage, where two components are merged if their average boundary contrast falls below the threshold ϵ ($\epsilon > \delta$). The information that is needed for the merging process, i.e., region labels, average edge contrasts, and lengths of the borders between adjacent regions, is stored in a region adjacency graph.

Because the merging criterion is based on the average edge contrasts rather than the maximum and minimum contrasts of the neighboring regions, the method combines into one segment regions where the contrast changes smoothly. This is desirable as this results in curved but locally smooth regions to be segmented correctly into a single region. The method results in a reliable and robust segmentation which can also be efficiently implemented in parallel hardware.

5 Results and discussion

We have tested our segmentation method on several range images containing both planar and curved objects. The images were obtained from the NRCC (National Research Council of Canada) range image library [12]. The images are of good quality, but we have tested the effect of adding salt-and-pepper noise to the images.

Normal vectors are approximated both by using the LTS method [11] and by using a local quadratic surface LSQ method [1]. We have implemented the methods for 3×3, 5×5, and 7×7 neighborhoods. In order to accelerate the processing of the LTS method, not all the possible triplets are used for estimating the plane equation, but only 5, 10, and 20 preselected triplets for 3×3, 5×5, and 7×7 neighborhoods, respectively. The triplets were preselected and not randomly selected for two reasons: to avoid selecting same triplets again and to avoid degenerized triangles that do not span a triangle (i.e., the measurements form a straight line). A third reason is speed: it is faster to consult a look-up table for triangle vertices than to generate them randomly. The LSQ method is a very fast mask-based method.

When using the LTS method, we obtain a robust estimate for the current range measurement once the plane that best fits the data for the current neighborhood is determined. It is obtained from the plane equation by using the x and y values of the cur-

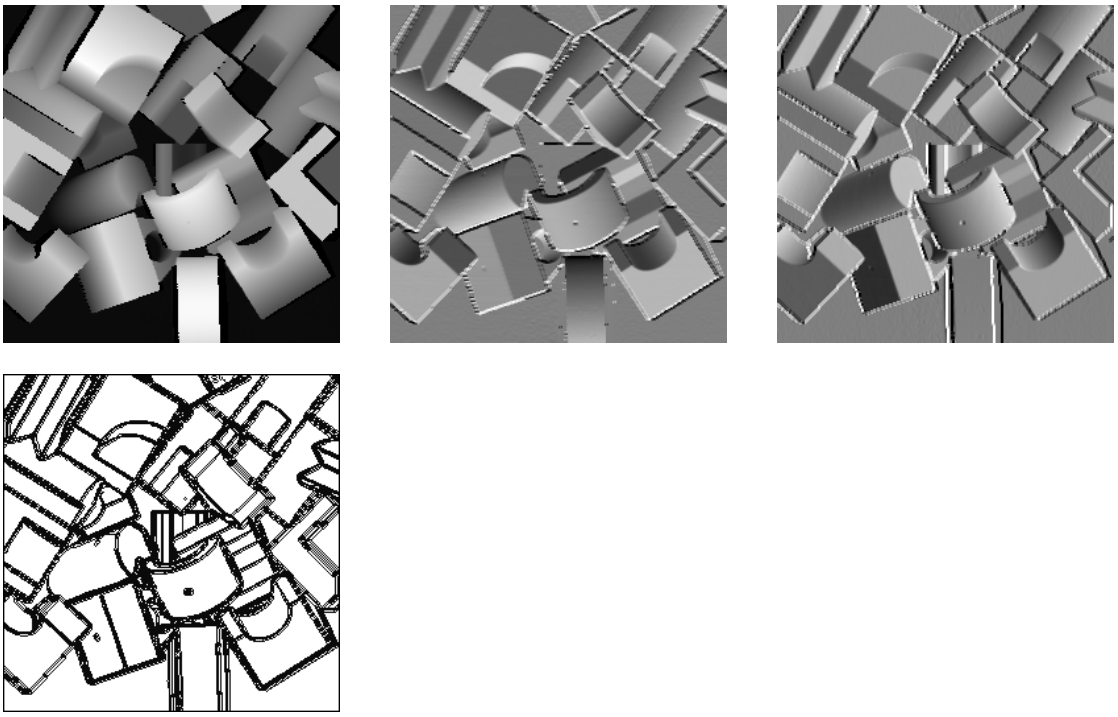

Figure 4: The segmentation of a pile of planar and curved blocks using 3×3 operator size and LSQ method. Depth information is not used in segmentation. From left to right, top to bottom: original depth map, normal x- and y-components, and segmentation result.

rent measurement. With the LSQ method, we take the measured depth value as it is; another possibility would be to take the median of a small neighborhood.

In order to be able to scale the depth values to $[0, 255]$ we obtain a robust estimate of their dynamic range by tesselating the original data into 5×5 pixel non-overlapping windows, and calculating the median depth in each window. The minimum and maximum values are used for linear scaling of the depth values.

Some examples of the results, using LTS normal extraction, are presented in Figs. 2 and 3. The images from left to right, top to bottom contain the original range image, the linearized x- and y-components of the normals, the depth image, and finally the segmentation. Fig. 2 is the same as in [4], but the segmentation is achieved without surface fitting (other than for normal approximation). Fig. 3 is a more complicated image of a collection of curved and planar blocks lying on a table. Notice that even curved surfaces have been correctly segmented.

In the current implementation, our segmentation method calculates the contrast of two pixels by using the Euclidean distance of the three "color" components. Nearly as good results can be obtained by using city-block distance, or even the maximum difference.

The domain of the x- and y-components is a real scalar $[-1, 1]$, i.e., the angles vary from $0.0°$ to $180.0°$. This means that the resolution of the x- and y-components is $180.0°/256 = 0.7°$. The normalization of the normal components was found to be very important; if they were left unnormalized, curved surfaces were often fragmented into several regions. The resolution of the depth components varies depending on the dynamic range of the depth values in a scene, but we have put limits for the accepted depth range. The upper limit is due to the depth of field or useful operating range of the range finding device, and the lower limit is set in order to avoid too swift change of contrasts within very inclined surfaces.

All the images were obtained after one basic segmentation and one merging phase. We used 4-connectivity, and the thresholds used with the LTS method were 7 (δ) and 15 (ϵ) which correspond to $5°$ and $10.5°$, respectively, in the x- and y-components. With the LSQ method (Fig. 4) we chose slightly lower thresholds, 5 and 10. The LSQ method tends to blur images and smoothen the transition from one surface to another, which makes it more difficult for the connected component analysis part to separate inclined surfaces. This blurring can be observed by thicker region borders (many small regions) between large homogeneous regions in Fig. 4. If larger operator sizes

are used, the blurring effect is also obvious to the eye in the normal component images.

When using the robust LTS method for normal extraction, the segmentation does not seem to be sensitive to the choice of thresholds. Too low thresholds result in several tiny isolated regions, which can be easily detected from the region graph and be merged into larger regions. Setting thresholds too high leads to undersegmentation. A safe way is to use smaller thresholds and more merging iterations. There are also ways of choosing thresholds based on image complexity using a cumulative difference histogram [8].

The depth information is often not needed if LSQ methods are used because the estimated surface normals are perturbed by step edges. These perturbed normals keep two regions apart, even if their orientation is the same. Also in the LTS method the normals tend to sway when close to step edges, but the barrier thus formed is narrow and may contain holes, so the depth information is needed to keep the regions apart. Figure 4 is segmented without using the depth information.

The LTS method tolerates impulse noise very well, but LSQ based methods do not. On the other hand, LSQ methods tolerate small Gaussian noise better than LTS. In order to get the best of both worlds, one would first have to apply LTS, determine the outliers (very noisy measurements or measurements belonging to another surface), remove them and use some LSQ method to determine the surface orientation. Of course, this would have large repercussions on the processing costs.

The segmentation results in large homogeneous regions. At the boundaries, there are some small segments, most of which belong to one of the neighboring large regions. Higher level processes could, depending on the application, choose to use only the segmented regions or possibly to fit some surface models to large regions and check from the original data which of the large regions the small regions belong to (or whether they form a surface of their own).

Since our segmentation method concentrates on finding surface or surface orientation discontinuities, it is not the method of choice for modeling smoothly varying surfaces, e.g., a sculpture of a person's face. More suitable for that purpose are methods that try to find areas with different types of curvatures (paraboloids, saddle surfaces, etc.) [1][4]. However, our system is well suited for situations where the whole scene has to be analysed, as in robot navigation or with intelligent industrial manipulators.

6 Conclusions

A simple but powerful method for range image segmentation has been presented. Range images are segmented into homogeneous regions consisting of planar and curved surfaces. The comparison of surface normal vectors is decomposed into the comparison of normal vector x and y components, which are normalized in respect of angular changes. Also the depth component is included in the segmentation process. The decomposition is treated as a three-band color image which is segmented using a hierarchical connected component method. The decisions of this method about merging neighboring regions are based on the average contrast between those regions, and result in more robust segmentation than if the merging decision was based on their maximal contrast differences. The merging criterion also connects curved surfaces instead of splitting them into planar patches.

The method yields a robust segmentation without the need for applying resource consuming variable order surface fitting. Thus, it is suited for scene analysis processes used by, e.g., intelligent robots: it is rather simple and can thus be made to function fast and it reliably separates different objects. Should the higher-level processes need the information about explicit surfaces, then most of the segmentation is already done, and the region growing phase of variable order surface fitting is reduced to fewer iterations.

Acknowledgements

The financial support provided by the Technology Development Center of Finland is gratefully acknowledged. Special thanks are due to David Harwood who first gave the idea of studying surface normals and how they could be used in segmentation. We would also like to thank Visa Koivunen for his comments on the paper.

References

[1] P. Besl, *Surfaces in Range Image Understanding*, Springer-Verlag, New York, 1988.

[2] T. Fan, G. Medioni, R. Nevatia, "Segmented Descriptions of 3-D Surfaces", *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 6, pp. 527-538, December 1987.

[3] N. Yokoya, M. Levine, "Range Image Segmentation Based on Differential Geometry: A Hybrid

Approach", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, no. 6, 1989.

[4] V. Koivunen, M. Pietikäinen, "Experiments with Combined Edge and Region-Based Range Image Segmentation", *Theory & Applications of Image Analysis—Selected Papers from the 7th Scandinavian Conference on Image Analysis*, World Scientific, 1992.

[5] R. Taylor, M. Savini, A. Reeves, "Fast Segmentation of Range Imagery into Planar Regions", *Computer Vision, Graphics, and Image Processing*, vol. 45, pp. 42-60, 1989.

[6] B. Sabata, F. Arman, J. Aggarwal, "Segmentation of 3-D Range Images Using Pyramidal Data Structures", *Proc. 3rd International Conference in Computer Vision*, Japan, pp. 662-666, 1990.

[7] R. Haralick, L. Shapiro, *Computer and Robot Vision*, Vol. 1, Addison-Wesley, USA, 1992.

[8] T. Westman, D. Harwood, T. Laitinen, M. Pietikäinen, "Color Segmentation by Hierarchical Connected Component Analysis with Image Enhancement by Symmetric Neighborhood Filters", *Proc. 10th International Conference on Computer Vision and Pattern Recognition Systems and Applications*, Atlantic City, pp. 769-802, 1990.

[9] P. Huber, *Robust Statistics*, John Wiley & Sons, 1981.

[10] P. Besl, J. Birch, L. Watson, "Robust Window Operators", *Machine Vision and Applications*, vol. 2, pp. 179-191, 1989.

[11] P. Rousseeuw, A. Leroy, *Robust Regression & Outlier Detection*, John Wiley & Sons, 1987.

[12] M. Rioux, L. Cournoyer, *The NRCC Three Dimensional Image Data Files*, CNRC 29077, National Research Council Canada, 1989.