MVA '92 IAPR Workshop on Machine Vision Applications Dec. 7-9,1992, Tokyo

THREE DIMENSIONAL BOUNDARY DETECTION USING HIGHER-ORDER SURFACE FITTING AND DIRECTIONAL SMOOTHING

Tomas Gustavsson and Quan Liang
Department of Applied Electronics, Chalmers University of Technology

S-412 96 Gothenburg, Sweden

ABSTRACT

The authors propose an algorithm for detection of three-dimensional boundaries in noisy images based on higher-order polynomial surface fitting and directional smoothing. Fitting a polynomial to the local intensities gives the intensity hypersurface. An isointensity surface is derived from the hyperplane and directional smoothing is defined as smoothing along this isointensity surface. The developed boundary detection algorithm exploits this smoothing technique and gives good edge preservation, localization and noise reduction. Experiments were performed on synthetic noisy data sets. Results indicate that the algorithm is superior to other edge-preserving smoothing approaches reported previously.

1. INTRODUCTION

Boundary detection algorithms applied to data which is severely contaminated by noise, e.g. many medical images, need a pre-smoothing step. Classical smoothing has a tendency to blur edges, so edge-preserving smoothing techniques have attracted many investigators [1, 2]. The basic idea behind most of the non-linear adaptive edge-preserving filters is to carry out smoothing using pixels belonging to the same homogeneous area [2]. A similar idea prompted by the template matching method [3] is described in [4]. By estimating the orientation of local intensity planes, smoothing along the plane but not across was applied. Good results could be obtained matching small neighborhoods and when surface curvature was low. Using larger neighborhoods for greater smoothing, or when boundary curvature was high, edges became degraded. This paper extends the idea of surface match and proposes a new method for edge preserving smoothing and 3D boundary detection based on higher-order surface fitting and directional smoothing. The main point is that smoothing is carried out along an estimated curved surface. The principle of directional smoothing is explained in section 2. Section 3 discusses the influence of noise on directional smoothing. Section 4 presents a boundary detection algorithm based on higher-order surface fitting and directional smoothing. Section 5 describes an experiment for evaluating this technique and also contains a comparison with two other well-known edge preserving smoothing techniques. Section 6 gives experimental results and a discussion.

2. SURFACE FITTING AND DIRECTIONAL SMOOTHING

Surface fitting is first described in two dimensions. Suppose we have digitized image intensities in a symmetric neighborhood

$$I'(x,y),$$

where

$$x \in \left[-x_0, x_0\right],\ y \in \left[-y_0, y_0\right],$$

and a step or ramp edge passes through this area. It is assumed that the gray level structure of $I'(x,y)$ can be modelled by a hypersurface denoted by a polynomial

$$I'(x,y) = \sum_{0 \le i+j \le m} a_{ij} x^i y^j \tag{1}$$

where m can be 1, 2, 3, ... , M. This model surface is fitted to the local intensities in least squares sense [5]. Suppose that at the center of this area, the value of the fitting polynomial is I_0, i.e. $I(0,0) = I_0$, then (1) can be written as

$$I(x,y) = I_0 + I_1(x,y), \tag{2}$$

where

$$I_1(x,y) = \sum_{1 \le i+j \le m} a_{ij} x^i y^j. \tag{3}$$

Using the boundary condition

$$I(x,y) = I_0, \tag{4}$$

gives

$$I_1(x,y) = \sum_{1 \le i+j \le m} a_{ij} x^i y^j = 0. \tag{5}$$

This means that the curved line defined by (5) passes through the neighborhood center pixel and has the property that pixels on this line have the same intensity value as the center pixel, in other words a model-based isointensity line passing through the center pixel. This can be geometrically represented as in Fig.1. Ideally, if the model fits the intensity value well, this line should overlap the real edge when the center

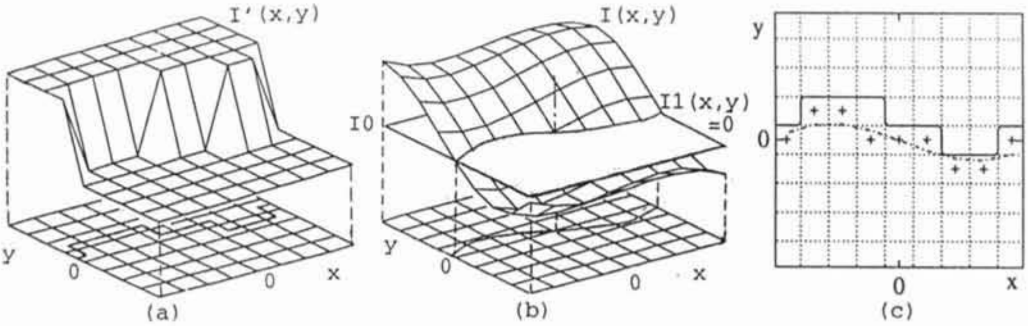

Fig. 1. x-y-I space representation of directional smoothing.

(a) Local intensity $I'(x,y)$.

(b) Fitting surface $I(x,y)$ and the plane $I_0 = I(0,0)$ which have an intersection line $I_1(x,y) = 0$. The projection of this line on the x-y plane is an isointensity line.

(c) Real edge (solid line) and isointensity line (dashed line) in the x-y plane. '+' denotes pixels located on the isointensity line. Smoothing is carried out with these pixels.

pixel lies on the real edge, and is otherwise 'parallel' to the real edge. The amount of 'parallellism' depends on how well the model used fits the real data.

If the noise added to all of the neighborhood pixels has a normal distribution, then noise added on pixels located on the isointensity line is also normally distributed with zero mean. An average (or weighted average) of neighborhood pixel values should therefore recover the original value of the homogeneous area.

The extension of this principle to 3D case is straightforward. Consider the local image intensities

$$I'(x,y,z),$$

where

$$x \in \left[-x_0, x_0\right],\ y \in \left[-y_0, y_0\right],\ z \in \left[-z_0, z_0\right].$$

These data are modelled by the hypersurface polynomial

$$I(x,y,z) = \sum_{0 \le i+j+k \le m} a_{ijk}\, x^i y^j z^k. \qquad (6)$$

The equation

$$I_1(x,y,z) = \sum_{1 \le i+j+k \le m} a_{ijk}\, x^i y^j z^k = 0 \qquad (7)$$

defines an isointensity surface. Directional smoothing is carried out using an average value of pixels located on this surface.

3. INFLUENCE OF NOISE

It should be noted that the noise itself influences surface fitting, and hence the trajectory of the isointensity surface. If the noise is excessive, the trajectory of the surface may go across the boundary causing edge blurring instead of being parallel to the boundary. This problem can be reduced both by using larger neighborhoods for surface fit and lower order fitting which is less sensitive to noise. Linear fitting, the lowest order fitting, is relatively resistant to noise but causes blurring if the boundary curvature is high. So when there is a lot of noise, it is recommended to carry out a linear fit first using a relatively large neighborhood in order to get a stable plane orientation. Then smoothing is applied using only pixels in a small segment near the center of the isoplane to reduce the possible blurring caused by smoothing along a large plane. The next step involves higher order surface fitting using the plane-smoothed data to extract the isointensity surface. Smoothing is now carried out along this curved surface using the original data. The initial linear fit and smoothing is used only as a preprocessing step to reduce the effect of noise before higher order fitting is applied.

4. 3D BOUNDARY DETECTION

An algorithm for 3D boundary detection is presented below. It is based on directional smoothing as described in Section 2 and the noise considerations presented in Section 3. There are three points to note about the algorithm. First, Gaussian filters are used in all the directional smoothing operations. Second, directional smoothing is only applied to pixels in regions where a possible edge is detected; otherwise standard isotropic smoothing is applied both to reduce computing time and also to get a better result. Third, a gradient threshold is used to determine if a particular pixel belongs to an edge region or not. This threshold is not used directly for edge detection and therefore can be estimated by a simple calculation using a threshold determined from the statistics of the gradient image.

The boundary detection algorithm is

(1) For each pixel, apply hyperplane fitting in the neighborhood. If the slope (or gradient) value is higher than the automatically preset threshold, the pixel is assumed to belong to an edge region and directional smoothing is applied, otherwise ordinary

isotropic smoothing is applied.

(2) Use data generated from step 1 for carrying out quadratic or cubic hypersurface fitting for edge pixels only. Then apply directional smoothing using the original data, i.e. the data before step 1.

(3) Detect the boundary by searching for zero-crossing points of the second directional derivatives of the fitting surface with pixels extracted by step 2.

(4) Apply a boundary thinning procedure.

5. EXPERIMENT

The presented boundary detection algorithm was applied to a synthetic 128x128x25 image which had step edge boundaries with high curvature, corners and twisted surfaces as shown in Fig.2. Gaussian noise from SNR 6db to 0db were added. The neighborhood size used was 5x5x5, plane pre-smoothing size was 3x3, order of surface fit was 3, i.e. cubic.

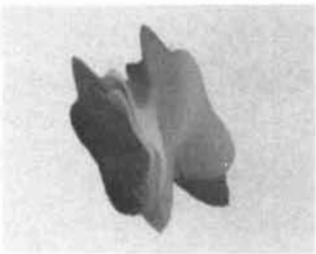

Fig. 2. Test image boundary.

For comparison, two published 2D edge-preserving smoothing techniques were used. These were SNN (Symmetric Nearest Neighborhood) [6], which is an improved version of the well known KNN filter [7], and MAXH (MAXimum Homogeneity) [8]. In our experiment these techniques were extended to 3D and replaced directional smoothing in the given boundary detecting algorithm. The automatic threshold estimation was replaced by manually threshold setting to get the best result.

The detected edges were quantitatively evaluated using the first four criteria below. The fifth criterion was used to evaluate the image error after smoothing.

COVERAGE — Percentage of the ideal edge covered by detected edge.(0-100%)

MATCH — Percentage of the detected edge that match the ideal edge.(0-100%)

MSD — Mean square distance between the detected pixels and their corresponding nearest ideal edge pixels. (in pixel unit)

SHAPE — A local edge coherence score as suggested by Kitchen and Rosenfeld in [9], with the parameter r = 0.8. (0.0-1.0)

RMS — Residual mean square error at edge region after smoothing.

6. RESULTS AND DISCUSSIONS

The edge preserving, localization and noise reduction performance of the boundary detection algorithm with different smoothing approaches is reflected in the four criteria COVERAGE, MATCH, MSD and SHAPE. RMS reflects the edge-preserving quality to some degree.

By comparing the results shown in Fig.3, it can be seen that directional smoothing performs better than SNN and MAXH in terms of all four boundary evaluation criteria. The relatively smaller slopes in most parts of all the solid lines in Fig.3 indicate that directional smoothing is less sensitive to noise.

When the noise level is low, RMS values from SNN and MAXH are less than that from directional smoothing. However the scores which evaluate the following boundary detection show that directional smoothing still gives the more accurate boundary. These results indicate that RMS values for edge regions reflect the edge-preserving smoothing effects to some degree, but are not adequate as sole performance criteria. Furthermore, the relatively poor RMS scores of directional smoothing in cases with little noise is mainly due to the plane smoothing in step 1, which is redundant in a low-noise situation where higher-order surface fit can be applied directly without pre-smoothing.

Further experiments were carried out using iterations of SNN and MAXH. However the results were inconclusive in terms of the set of performance criteria.

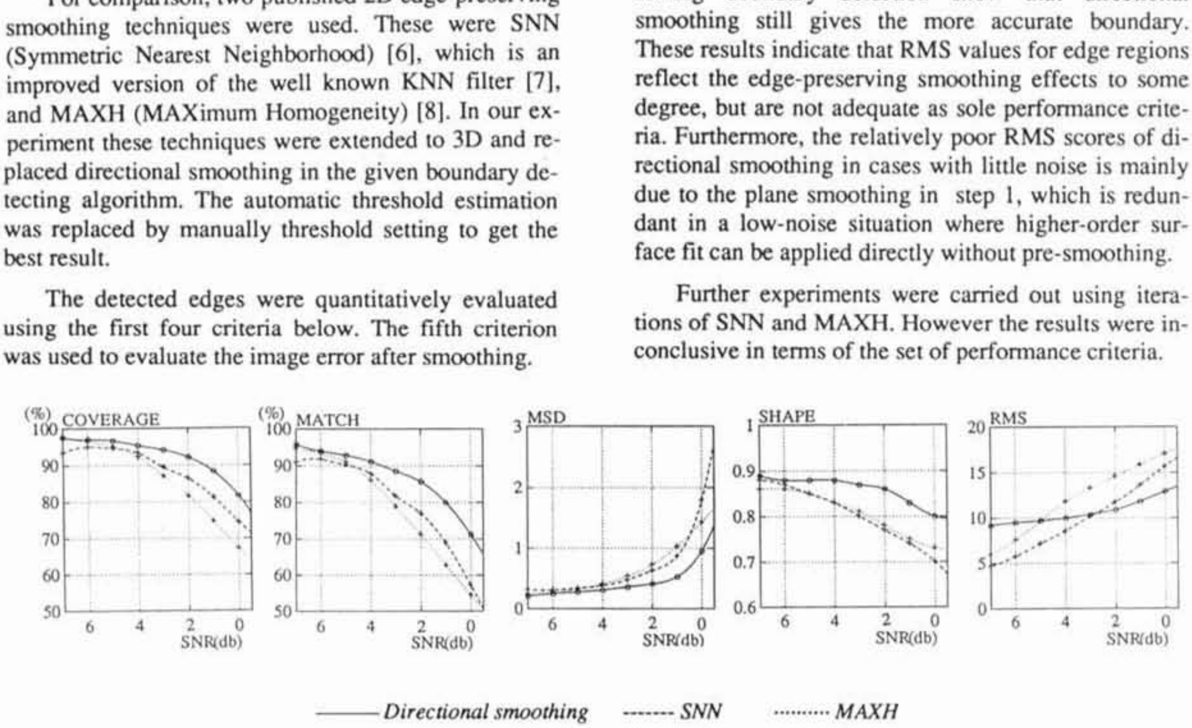

Fig. 3. Directional smooting vs. SNN & MAXH.

7. CONCLUSION

The new boundary detection algorithm based on higher order surface fitting and directional smoothing detects boundaries in noisy images with good noise reduction, edge preservation and edge localization. It should serve as an important initial stage of an object recognition process for noisy 3D images.

8. REFERENCES

[1] R.T.Chin and C.L.Yeh, Quantitative evaluation of some edge-preserving noise-smoothing techniques, CVGIP, Vol.23, 1983, pp.67-91.

[2] J.M.H. Du Buf and T.G.Campbell, A quantitative comparison of edge-preserving techniques, Signal Processing, Vol.21, No.4, 1990, pp.289-301.

[3] L.C.Topa and R.J.Schalkoff, Edge detection and thinning in time-varying image sequences using spatio-temporal templates, Pattern Recognition, Vol. 22, No.2, 1989, pp.143-154.

[4] T. Gustavsson, On the acquisition, analysis, and display of echocardiographic image sequences, Technical Report No.209, School of Electrical and Computer Engineering, Chalmers University of Technology, Sweden.

[5] P.J.Besl and R.Jain, Segmentation through various order surface fitting, IEEE-PAMI. Vol.10, No. 2, 1988, pp.167-192.

[6] T.Lonnestad, Connected Filters for Noise Removal, Proc. 9th ICPR, Rome, Italy, 14-17 Nov. 1988, pp.848-850.

[7] L.S.Davis and A.Rosenfeld, Noise cleaning by iterated local average, IEEE Trans. on Syst. Man. and Cybern., Vol. SMC-8, No.8, 1978, pp.705-710.

[8] M.Nagao and T.Matsuyama, Edge preserving smoothing, CGIP, Vol.9,1979, pp.394-407.

[9] L.Kitchen and A.Rosenfeld, Edge Evaluation Using Local Edge Coherence, IEEE Trans. on Syst., Man., and Cybern., Vol. SMC-11, No.9.Sept., 1981, pp.597-605.

ACKNOWLEDGEMENT

This project was supported in part by the Swedish National Board for Industrial and Technical Development (NUTEK) under Grant No. 90-01693P.