Boundary Segmentation by Detection of Corner, Inflection and Transition Points

Kazuhide SUGIMOTO

Real World Computing Partnership
Tsukuba Mitsui Bldg., 1-6-1 Takezono,
Tsukuba, Ibaraki 305, Japan
sugimoto@trc.rwcp.or.jp

Fumiaki TOMITA

Electrotechnical Laboratory
1-1-4 Umezono,
Tsukuba, Ibaraki 305, Japan
tomita@etl.go.jp

Abstract

For future intelligent man-machine system with vision, it is necessary to visualize the results of shape and motion analysis of observed objects in the images. As for object recognition, there are at least three steps. The first is to detect edges which correspond to the boundaries of objects (image segmentation). The second is to segment each boundary into simple line or curve segments (feature extraction). The third is to match those features between the data and the model. This paper presents a new method for the second step: boundary segmentation. It can detect not only corners but inflection points on which the sign of the curvature changes and transitional points on which a line and a curve connect smoothly without any delicate threshold. It also calculates the curvature and the normal vector at each point on the boundary with good accuracy. The features extracted by the proposed method are useful for both machine vision and visualization.

1 Introduction

Segmentation of boundaries in image is one of the crucial steps in shape recognition and description. The goal of a boundary segmentation algorithm is to partition a boundary of the regions into the line or curve edge segments. These segments can be used as units for correspondence in edge based stereo[1] and units for feature matching process between known physical properties and observed object features[2]. Furthermore, line and curve segments are good unit for qualitative and hierarchical definition and symbolic description and classification of objects[3]. However, many of the popular techniques for boundary segmentation make use of local k-curvature[4] which is sensitive to noise, so they need smoothing process using such as Gaussian filtering which is commonly used also to describe the hierarchical structure of images[5][6][7][8], and zero-crossing of this smoothed curvature to detect only corners[9][10][11]. Almost of them are impossible to detect various feature points on boundaries, such as inflection points on which the sign of the curvature changes and transitional points on which line and curve connect smoothly, since zero-crossing cannot be applied to the points where the change of the curvature is small. Even if possible, they need delicate threshold value for segmentation[12]. Methods for polygonal approximation are not good enough for analysis and interpretation of planar curved figures[13][14]. In this paper, we propose a method for segmentation of boundaries which does not need delicate threshold value for segmentation by adapting it to the details of a feature automatically. Furthermore, this method can detect all kinds of feature points listed above and calculate the curvature and the normal vector at each point on the boundary with good accuracy.

2 Feature points for segmentation

Since line sections on the boundary are key feature for shape recognition and description, it is important to detect transitional points with good accuracy. As for curve sections it is difficult to determine the best suited section for quadratic curve approximation. If the error value for curve approximation is used for segmentation, a boundary is often segmented into too many short sections at the point even where the sign of the curvature does not change and curvature changes smoothly. This *over segmentation* should be avoided because it increases the number of segments on a boundary and also becomes the cause of the difficulty in finding key features. Since it is easy to detect end points and branching points using the connectivity number, the other feature points of which the connectivity number is two should be talk about and they are defined as follows.

corner point: a point at which the direction of the normal vector changes abruptly

inflection point: a point at which the sign of the curvature changes from positive(negative) to negative(positive).

transition point: a point at which the curvature changes from zero to non-zero (a point at which two curves that have the same convexity property and different curvature connect smoothly (so called *smooth join*) is not treated as transition point.)

3 Calculation of curvature and normal vector

3.1 Directional boundaries

There are two types of directional boundaries which surround the region on the right, one is the peripheral boundary the other is the hole boundary as shown in figure 1. After this, the order of the edge points on the boundary for tracking depends on this direction.

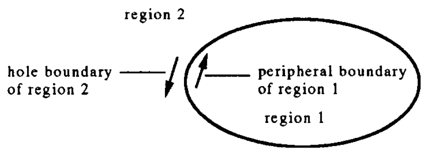


Figure 1: **Directional boundaries**

3.2 Approximation of circle and straight line

First, to every point, both straight line l_{p_i} and circle C_{p_i} are approximated using a set of neighbor points within the distance K from the noticed point P_i by the least squares method to calculate the local *temporal* curvature and the *temporal* normal vector as shown in figure 2. Here, K is the threshold value.

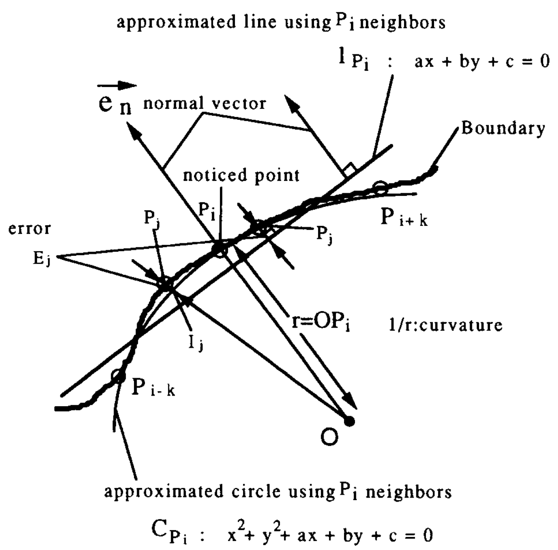


Figure 2: **Approximation of circle and straight line, and calculation of curvature and normal vector**

3.3 Error calculation

Then, calculate the error value Le_i for line approximation and Ce_i for circle approximation at each point P_i. How to calculate Le_i is as follows.

- Let E_j be the distance between P_j and the foot of the perpendicular dropped from point P_j to l_{p_i}.
- Calculate E_j to every neighbor point $P_j(i-k \leq j \leq i+k)$ which is used for straight line approximation.
- Let Le_i be the maximum error value of $E_j(i-k \leq j \leq i+k)$.

How to calculate Ce_i is as follows.

- Let O be the center of the approximated circle C_{p_i}.
- Let I_j be the intersection of C_{p_i} and the line which passes through both points P_j and O.
- Let E_j be the distance between P_j and I_j.
- Calculate E_j to every neighbor point $P_j(i-k \leq j \leq i+k)$ which is used for the circle approximation.
- Let Ce_i be the maximum error value of $E_j(i-k \leq j \leq i+k)$.

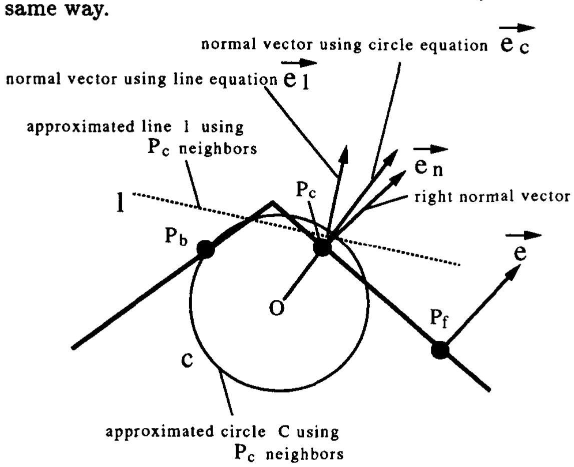
Let E_i be the smaller error value Le_i or Ce_i at the noticed point P_i.

3.4 Temporal curvature and normal vector

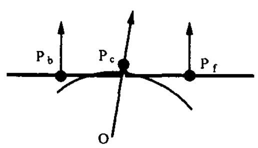
Next, the local *temporal* curvature and the *temporal* normal vector are calculated at every point P_i. If $Le_i \leq Ce_i$, then the curvature is zero and the direction of the normal vector differs $\frac{\pi}{2}$(radian) from the one of the boundary and outward from the region. If $Ce_i < Le_i$, then the curvature is $\frac{1}{r}$. Here, r is the radius of the circle C_{p_i}. The direction of the normal vector is equal to the one of the vector from O to P_i. The sign of the curvature depends on which side of region O is. If O is inside of the region then the sign of curvature is plus (plus means convex), if outside, minus (concave). These processes are applied to every point P_i on the same boundary and calculate the local *temporal* curvatures and the *temporal* normal vectors.

3.5 Renewal of curvature and normal vector

However, E_i becomes big near the corner and the noisy point. Let the line and the circle which are approximated to the noticed point P_c using a set of neighbor points be l and c respectively as shown in figure 3(a). Both of the *temporal* normal vector and the *temporal* curvature sometimes can be unexpected value in this case, whether the line is selected or not. To solve this problem, the lines and the circles on the backward point P_b and forward point P_f which are the distance of K away from the noticed point P_c in the same figure are used for comparison. Then select the line or the circle which has the minimum error value E_{min} from lines or circles at P_b and P_f to calculate the accurate curvature and the normal vector, and let E_{min} be the error value E_c at the noticed point P_c. Furthermore, when the point P_b is selected, E_{min} is compared with E_i of all points between P_c and P_b. At every point where E_i is bigger than E_{min}, the direction of the normal vector and the curvature is calculated again by using the line or circle equation on P_b. When the point P_f is selected, this renewal process should be applied to every point between P_c and P_f in the same way.



(a):Error near the corner point



(b):Error near the noisy point

Figure 3: **Renewal of curvature and normal vector**

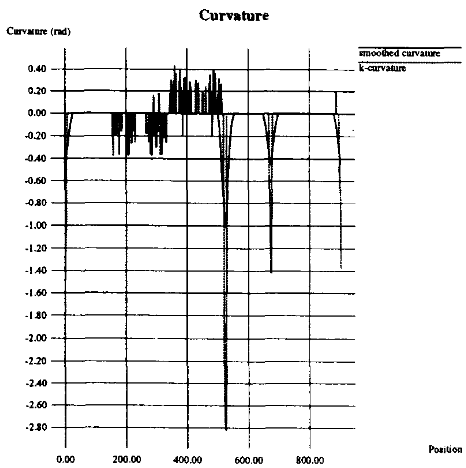


Figure 4: **k-curvature**

Figure 4 shows the usual local k-curvatures on the boundary in figure 7(a), original (with dotted line) and smoothed curvature by Gaussian filtering ($\sigma = 10$) (with thick line). There are some problems that the former is sensitive to noise and the latter calculate curvature as nearly equal to zero at the points where the change of the curvature is small (see position 150 to 500 in the same figure). And zero-crossing of this smoothed curvature can be used only for detection of the corner points.

Figure 5 shows the curvatures on the boundary in figure 7(a) by our new method, temporal (with thick line) and after renewal (with dotted line).

Figure 6 shows the direction of the normal vectors on the boundary in the same figure, before (with thick line) and after renewal (with dotted line).

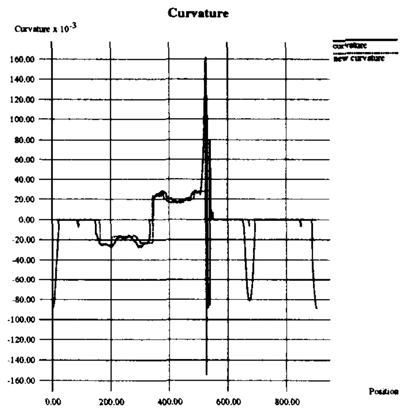


Figure 5: **Renewal of curvature**

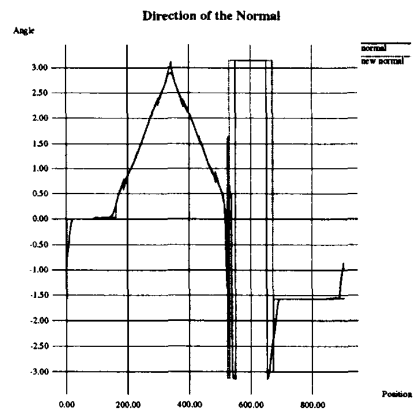


Figure 6: **Renewal of normal vector**

4 Segmentation point detection

4.1 Corner point

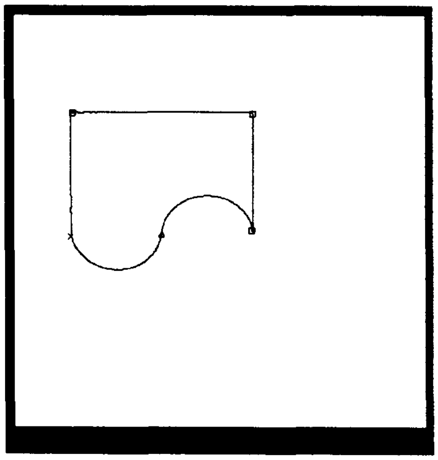
Corner points are detected as the point where the direction of the normal vector changes abruptly. When the difference is bigger than the threshold, the point can be detected as the corner point.

4.2 Transition point

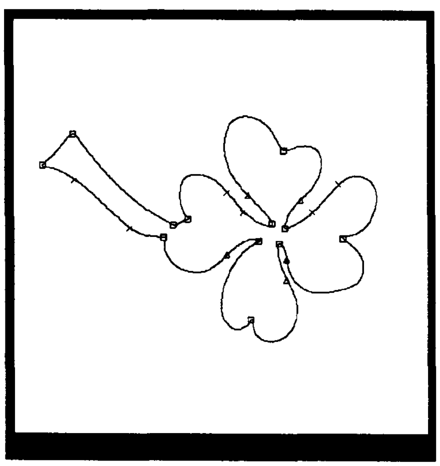
Candidates of transition points are detected at the point where the curvature changes from zero to non-zero. However these points can be observed near the noisy points. So it is necessary to select only the point where the line section and the curve section smoothly join from these candidates. If both sections have enough length (longer than the threshold K) and the point is not the corner point, it can be detected as the transition point.

4.3 Inflection point

Inflection points are detected at the point where sign of the curvature changes from positive(negative) to negative(positive). The point where two curve sections, one is convex and the other is concave, smoothly join can be detected as the inflection point. However, there can be the case that there exists short line section which is shorter than the threshold K between these two curves. In such case, the midpoint of this short line section becomes the inflection point.



(a)



(b)

□:Corner point, △:Inflection point, ×:Transition point

Figure 7: **Result of segmentation**

Figure 7 shows the result of segmentation of boundaries. □ , △ and × denote the corner point, the inflection point and the transition point respectively.

4.4 Noisy point

Points where the error value for line or circle approximation becomes big should be treated as noisy point. However, when the default value K is longer than the length of the line or curve section that should be segmented, error value for approximation becomes big also at each point in such section and desirable result cannot be obtained.

5 Adaptation of threshold value K

The bigger the threshold value K becomes, the robuster to noise the calculation accuracy of the curvature and the normal vector is. So the initial value of K should be set as big as possible to the length of the section that should be segmented. However, big K cannot be applied to the short section. When K is big, poits in short section are treated as noisy point. It is necessary only for such points to change the threshold value K smaller (ex. half size) and calculate the curvature and the normal vector again using renewed K to detect the feature points. While there is noisy point and K is bigger than the minimum length of the section that should be segmented, this process is repeated.

Figure 8 shows the result of segmentation of boundaries (the size of image is 512x512). □ , △, × and ⊙ denote corner point, inflection point, transition point and noisy point respectively. Figure 8(a) shows the result of detection of these feature points using 20 as the threshold value K without adaptaion. Figure 8(b) shows the one with adaptation of K using 20 as the initial value and changed to 5 ($\frac{1}{4}$ of the initial value) automatically so that the hopeful result is acquired.

6 Conclusions

In this paper, we presented a method that can detect the various feature points (corner, inflection and transition) and calculate the curvature and the normal vector at each point on the boundary with good accuracy, and does not require the delicate threshold values for segmentation and interactive renewal of them. It takes only five threshold values, the first one is for the difference of the angle at the corner point(default value is fixed to 0.3 radian), the second one is the threshold of radius for the linearity(default value is fixed to 256), the third one is the error value for approximation (default value is fixed to 1 pixel), the fourth one is the minimum length of the section that should be segmented and the rest is adaptive K (the initial value is fixed to 20). These threshold values except K need no delicate adjastment for input images. In this way, adaptive boundary segmentation which needs no delicate threshold value for segmentation is realized.

This method is very important for many kinds of higher-level post-processing and may be useful to visualize many characteristics (types of feature points, curvature and direction of normal at each point, etc) and can be the one connection between machine vision and visualization.

References

[1] G.G.Medioni , R.Nevatia, "Segment-based stereo matching," *Proc. of Image Understanding Workshop*, pp. 128-136, 1983.

[2] R.B.Fishler, "Model invocation for three dimensional scene Understanding," *Proc. of 10th International Joint Conference on Artificial Intelligence*, pp. 805-807, 1987.

[3] F.Tomita , M.Koizumi, "A step toward generic object recognition," *Proc. of 11th International Conference on Pattern Recognition*, pp. 632-636, 1992.

[4] Rosenfeld.A. and A.C.KAK, *Digital picture processing,* New York : Academic Press, 1976.

[5] H.Asada , M.Brady, "The Curvature Primal Sketch," *PAMI-8*, NO.1 pp. 2-13, 1986.

[6] A.P.Witkin, "Scale space filtering," *Proc. of 8th International Joint Conference on Artificial Intelligence*, pp. 1019-1022, 1983.

[7] A.L.Yuille , T.Poggio, "Scaling theorems for zero crossings," *PAMI-8*, NO.1 pp. 15-25, 1986.

[8] J.Babaud , A.P.Witkin , M.Baudin and R.O.Duda, "Uniqueness of the Gaussian kernel for scale space filtering," *PAMI-8*, NO.1 pp. 26-33, 1986.

[9] J.Q.Fang , T.S.Huang, "A corner finding algorithm for image analysis and registration," *Proc. of AAAI Conf.*, pp. 46-49, 1982.

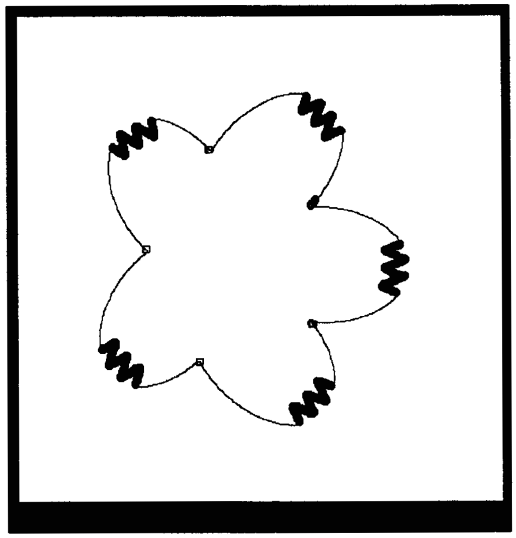
[10] O.A.Zuniga , R.Haralick, "Corner detection using the facet model," *Proc. of IEEE CVPR.*, pp. 30-37, 1983.

[11] R.Mehrotra , S.Nichani, "Corner Detection," *Pattern Recognition*, Vol.23, No.11 pp. 1223-1233, 1990.

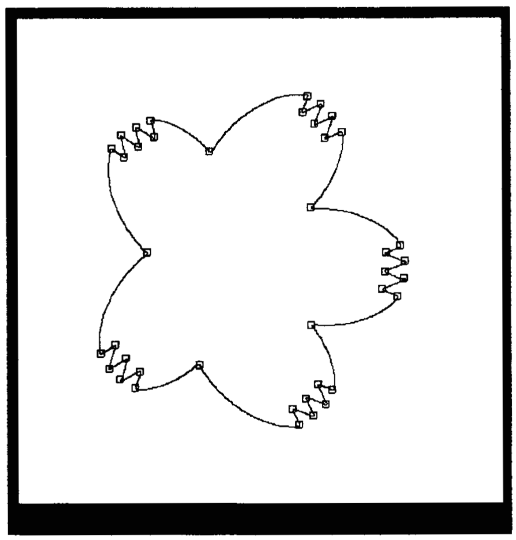
[12] C.H.Teh , R.T.Chin, "On the detection of dominant points on digital curves," *PAMI-11*, NO.8 pp. 859-872, 1990.

[13] T.Pavlidis, "Algorithms for shape analysis and waveforms," *PAMI-2*, NO.7 pp. 301-312, 1980.

[14] J.L.Dunham, "Optimum uniform piecewise linear approximation of planar curves," *PAMI-8*, NO.1 pp. 66-75, 1986.



(a) K**=20**



(b) K **from 20 to 5**

□:Corner point, △:Inflection point, ×:Transition point, ⊙:Noisy point

Figure 8: **Adaptation of** K