ELSEVIER

Pattern Recognition Letters 19 (1998) 279–286

Pattern Recognition Letters

A modified morphological corner detector [1]

Rey-Sern Lin [a], Chyi-Hwa Chu [b], Yuang-Cheh Hsueh [a,*]

[a] *Department of Computer and Information Science, National Chiao Tung University, Hsinchu 30050, Taiwan, ROC*

[b] *Taiwan Provincial Institute for Elementary School Teachers, Inservice Education, 30. Sec. 1, Ta Kwan Road, Pan-Chiao City, Taipei County, Taiwan, ROC*

Received 2 May 1997

Abstract

In this paper we propose a modified morphological corner detection method which finds convex and concave significant points using simple integer computation. We use the morphological peak extractor to detect convex corners and use a modified valley extractor to detect concave corners. Moreover, information of the distances between pixels on the boundary segment determined by two contiguous significant points and the chord connecting these two contiguous significant points can be applied to remedy the loosing corner due to the shape of the chosen structuring element.

Keywords: Corner detection; Chain code; Morphology; Structuring element

1. Introduction

Corners are very useful features in image matching, image representation and shape analysis (Gupta and Malakapalli, 1990). Traditionally, a point is declared as a corner point if, at this point, the object boundary makes discontinuous changes in direction or the curvature of the boundary is above some threshold. Many approaches (Rosenfeld and Johnston, 1973; Rosenfeld and Weszka, 1974; Freeman and Davis, 1977; Sankar and Sharma, 1978; Teh and Chin, 1989; Liu et al., 1990; Pei and Horng, 1994, 1996; Sarkar, 1993) have been developed for corner detection. They are all based on the analysis of chain-coded boundaries of objects. Most of them involve complex floating-point computation and selection of region support. Especially, for tremendous objects, the floating-point computation is time consuming and the selection of improper region support will cause false corners. Sarkar (1993) proposes a simple algorithm based on the difference of boundary direction to detect significance vertices, however his method is sensitive to boundary noise or boundary distortions. In summary, the above boundary-based corner detection methods suffer the dependence either on the correctness of region segmentation or the susceptivity of noise.

With computational simplicity and effectiveness, mathematical morphology has been applied successfully to image and signal processing (Serra, 1982; Giardina and Dougherty, 1988). The existent morphological corner detectors (Noble, 1992; Shapiro et al., 1987; Zhang and Zhao, 1997) are derived from (Meyer and Beucher, 1990). It should be noted that

* Corresponding author. E-mail: ychsueh@cis.nctu.edu.tw.

[1] This work was supported by the National Science Council, Republic of China, under grant NSC 87-2213-E-009-007.

0167-8655/98/$19.00
PII S0167-8655(97)00177-3

the result of peak or valley extractor of an object consists of areas around corner points. Zhang and Zhao (1997) shrink corner portion to a single corner according boundary information, recently. Therefore, it is our goal in this paper to modify the existent morphological corner detectors such that they can find the actual corner points.

The paper is organized as follows. In Section 2, we will introduce the elementary theorem of mathematical morphology. In Section 3, we will describe the method we proposed for corner detection. In Section 4, we will discuss the effects of different types and sizes of structuring elements and compare with some pioneer works for minute and tremendous objects. In our experiment, the results show that our method is effective in detecting corners for tremendous objects. On the other hand, it also has a little error with the measurement of distance difference for minute objects. Finally, the conclusions will be given in Section 5.

2. Mathematical morphology

Mathematical morphology is a well-known tool in image processing (Serra, 1982; Giardina and Dougherty, 1988). The basic operations in the algebraic framework of mathematical morphology are dilations and erosions, each associated with a structuring element. These operations can be defined on Euclidean space with arbitrary dimension. In this paper, we confine ourselves to define them on the discrete Euclidean plane $\mathbb{Z}^2$.

Definition 2.1. Let A and B be two subsets of $\mathbb{Z}^2$. The *dilation* of A by B, written as $A \oplus B$, is given by $A \oplus B = \bigcup_{b \in B} A_b$. The *erosion* of A by B, written as $A \ominus B$, is given by $A \ominus B = \bigcap_{b \in B} A_{-b}$. Then, the *closing* of A by B, written as $A \cdot B$, is given by $A \cdot B = (A \ominus B) \oplus B$. The *opening* of A by B, written as $A \circ B$, is given by $A \circ B = (A \ominus B) \oplus B$.

The subset B used in the above definition is called a *structuring element*. In most applications, a structuring element is chosen to have small size and simple shape. Many useful properties of dilations, erosions, closings and openings can be found in the pioneer works of Matheron (1975) and Serra (1982). To our application, we notice that, if B is a disk-shaped structuring element, the effect of $A \circ B$ is to smooth away some convex portions of A. While the effect of $A \cdot B$ is to fill in some concave portions of A. Therefore, as in the works of Noble (1992) and Shapiro et al. (1987), it is intuitively to use the set difference $A - A \circ B$ (this is called the peak extractor when A is a numerical function (Serra, 1982)) to locate convex corners and use $A \cdot B - A$ (this is called a valley extractor when A is a numerical function) to locate concave corner points. However, as we pointed out before, the results of $A - A \circ B$ and $A \cdot B - A$ consist of areas around corner points not only corner points themselves. Furthermore, since $A \cap (A \cdot B - A) = \varnothing$, if we are required that all corner points of A must lie in A, then $A \cdot B - A$ is obviously not the desired corner detector. In a word, the localization ability of the existent morphological corner detection must be improved in order to find the actual positions of corner points.

3. Proposed method

Let A be a binary image, i.e., a subset of $\mathbb{Z}^2$, and B be a disk-shaped structuring element. In this section, we will first modify the morphological operator $A - A \circ B$ to find convex corner points. Then, we will modify the operator $A \cdot B - A$ to find concave corner points.

3.1. Convex corner detection

Before describing the proposed convex corner detection, we give the following definition first.

Definition 3.1. Let A be a binary image, N a neighborhood of the origin. For each point p in A, the *N-hit number* of p, written as $H_{N \uparrow A}(p)$, is defined by

$$H_{N \uparrow A}(p) = |N_p \cap A|.$$

Definition 3.2. A pixel p_i is said to have local maximum *N-hit number* around its neighbor p_{i-1} and p_{i+1}, if $H_{N \uparrow A}(p_{i-1}) \leqslant H_{N \uparrow A}(p_i) > H_{N \uparrow A}(p_{i+1})$ or $H_{N \uparrow A}(p_{i-1}) < H_{N \uparrow A}(p_i) \geqslant H_{N \uparrow A}(p_{i+1})$.

Definition 3.3. A pixel p_i is said to have local minimum *N-hit number* around its neighbor p_{i-1} and p_{i+1}, if $H_{N\uparrow A}(p_{i-1}) \geqslant H_{N\uparrow A}(p_i) < H_{N\uparrow A}(p_{i+1})$ or $H_{N\uparrow A}(p_{i-1}) > H_{N\uparrow A}(p_i) \leqslant H_{N\uparrow A}(p_{i+1})$.

Keeping this definition in mind, we observe that a boundary point p in a binary image A is a convex corner point then the number $H_{N\uparrow A}(p)$ should be a local minimum around its neighbor for a suitable neighborhood N. But, it is not true that a boundary point with a local minimal hit number is a convex corner point. To overcome this cumbersome, in the first step of our algorithm, we will use the morphological operator $A - A \circ B$ to locate the approximate positions of convex corner points. In the second step of our algorithm, we will search the boundary of each connected component of $A - A \circ B$ points with local minimal hit numbers then choose one of these points as the convex corner point.

Algorithm 1

1. *Input*:
   - Binary image A;
   - Neighborhood of the origin N;
   - Disk-shaped structuring element B;
2. *Output*:
   - Convex corner points of A;
3. *Step 1*. Form the set difference $A - A \circ B$.
4. *Step 2*. For each connected component R of $A - A \circ B$ do
   - Find the boundary curve C which is the intersection of the boundary of A and that of R;
   - Chain-coded the curve C by $p_1, p_2, \ldots, p_n$, for some n;
   - Compute the N-hit number $H_{N\uparrow A}(p_i)$ for each i;
   - Search for points with local minimal hit numbers;
   - Choose the point whose index is as close as to $n/2$ as the convex corner point.

Fig. 1 shows an acute part detected by $A - A \circ B$ with a large disk-shaped structuring element with size 99 and Table 1 shows the corresponding N-hit numbers of points on curve C using a neighborhood of 33 square centered at origin. Positions with local minimal of N-hit numbers are in the indices of 1, 3, and 6 that are labeled with circle, determined by algorithm Step 2. In our experiment, the index 6 is as close as to 5 $(n/2)$. Thus, we choose p_6 as the convex corner point.

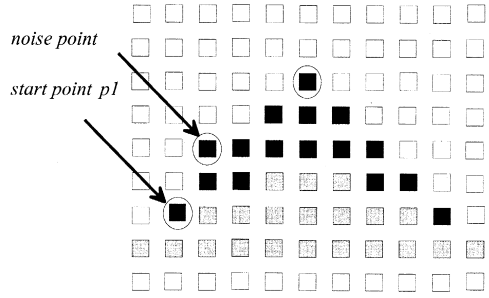


Fig. 1. A connected region of $A - A \circ B$.

This corner detector uses simple integer operations instead of complex floating point computations to find convex significant corner points, it is simple and fast.

3.2. Concave corner detection

Similar to the observation in Section 3.1, a boundary point p in a binary image A is a concave corner point then the number $H_{N\uparrow A}(p)$ should be a local maximum around its neighbor for a suitable neighborhood N. But, the converse is not true. Moreover, as we pointed out in Section 2, $A \cap (A \cdot B - A) = \varnothing$. That is, the boundary of A does not belong to the set $A \cdot B - A$. This situation is illustrated in Fig. 2. Thus, we modify the operator $A \cdot B - A$ to be $(A \cdot B - A) \oplus E$, where E is the rhombus structuring element. Then the true concave corner can be enforced to locate on the chain code of the boundary curve C that is the intersection of the boundary of A and that of a connected region R obtained from $(A \cdot B - A) \oplus E$. In the second step of our algorithm, we will search the boundary of each connected component of $A \cdot B - A$ to find

Table 1
N-hit number string of Fig. 1

Index of p_i	1	2	3	4	5	6	7	8	9	10
$H_{N\uparrow A}(p_i)$	6	7	4	6	6	4	6	6	6	6

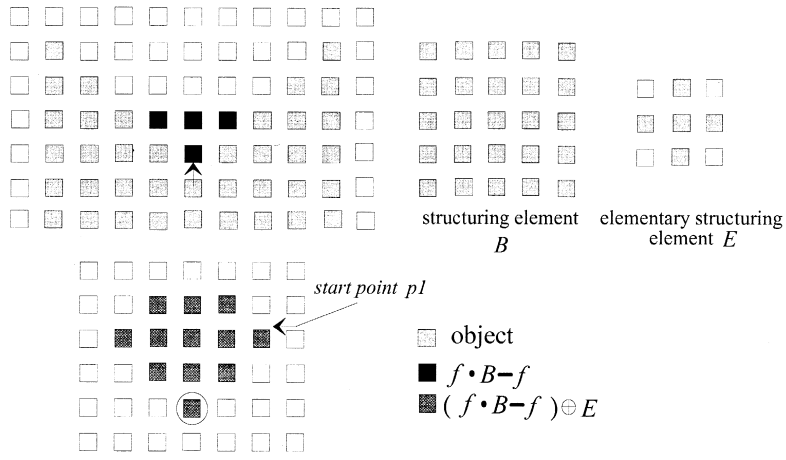


Fig. 2. An example of concave corner detection phase.

points with local maximal hit numbers then choose one of these points as the concave corner point.

Algorithm 2

1. *Input*:
   - Binary image A;
   - Neighborhood of the origin N;
   - Disk-shaped structuring element B;
   - Rhombus structuring element E;
2. *Output*:
   - Concave corner points of A;
3. *Step 1.* Form the image $(A \cdot B - A) \oplus E$.
4. *Step 2.* For each connected component R of $(A \cdot B - A) \oplus E$ do
   - Find the boundary curve C which is the intersection of the boundary of A and that of R;
   - Chain-coded the curve C by $p_1, p_2, \ldots, p_n$, for some n;
   - Compute the N-hit number $H_{N \uparrow A}(p_i)$ for each i;
   - Search for points with local maximal hit numbers;
   - Choose the point whose index is as close as to $n/2$ as the concave corner point.

Table 2
The concave number string of a local concave region in Fig. 3

Index of p_i	1	2	3	4	5
$H_{N \uparrow A}(p_i)$	6	6	8	6	6

Table 2 shows the corresponding N-hit number string of the chain code of boundary curve in Fig. 2. Position with the maximal hit number and as close as to $n/2$ is in the index of 3 which is labeled with circle. By our algorithm, the detected concave corner point is p_3.

Moreover, the detection phase of concave corners can be parallel computed with the detection phase of convex corners. The complete architecture of our corner detection method is shown in Fig. 3. In our algorithm, the procession only involved the morphological operators and the simple integer computation. As we know, morphological operator can be implanted to parallel machine. Then the computation time will decrease respectively.

4. Experimental results and discussion

In the using of morphological operators as a tool for digital image processing, it is a difficult skill to choose a suitable structuring element to fit working

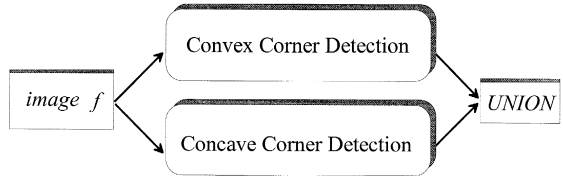


Fig. 3. The architecture of our proposed method.

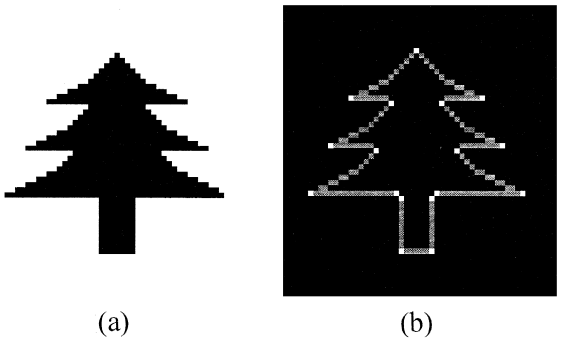


Fig. 4. (a) A binary artificial tree image. (b) The corner image of (a).

purpose. By our observation, we choose different structuring elements such as square and rhomboid for preserving convex and concave portions of an object. Square can preserve rhombus convex angles and rhombus concave angles with morphological operators $A - A \circ B$ and $A \cdot B - A$, but it will lose right angles. On the other hand, rhomboid can maintain right convex angles and right concave angles, but lose rhombus angles. The first test of our experiments took above two types structuring elements and applied to the architecture of Fig. 3 for detection. Then, the result is the union of each corner map. Fig. 4(a) is the original object image, Fig. 4(b) shows the result of taking the union of each corners map using the square and the rhombus structuring elements with the size of 7×7. Secondarily, our algorithm is applied to two digital binary images chromosome and eight images. The corresponding corner images shown in Fig. 5(a) and Fig. 5(b) resulted from the union of two corner images in which one using square and another using rhombus structuring element. Obviously, our experimental results show that our method can detect significant corner points of the stereoscope image, such as Fig. 5(a) and Fig. 5(b).

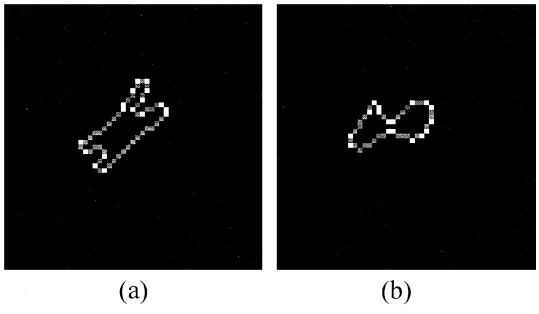


Fig. 5. (a) The corner map of chromosome image. (b) The corner map of eight image.

Unfortunately, morphological operators $A - A \circ B$ and $A \cdot B - A$ miss connected convex or concave component when the corner point has the angle of 135°. For the purpose of resolving the weakness of square or rhomboid structuring element, we choose circular structuring element to preserve local convex or concave parts. Fig. 6(a) is the result detected by applying a circular structuring element to a civil aircraft image with size of 56×57.

We also observe that as the size of structuring element increases, our method will increase the precision of corners detected, but will increase computation as well. It is because that the corner parts detected from morphological operators with larger structuring element contain more information about corners.

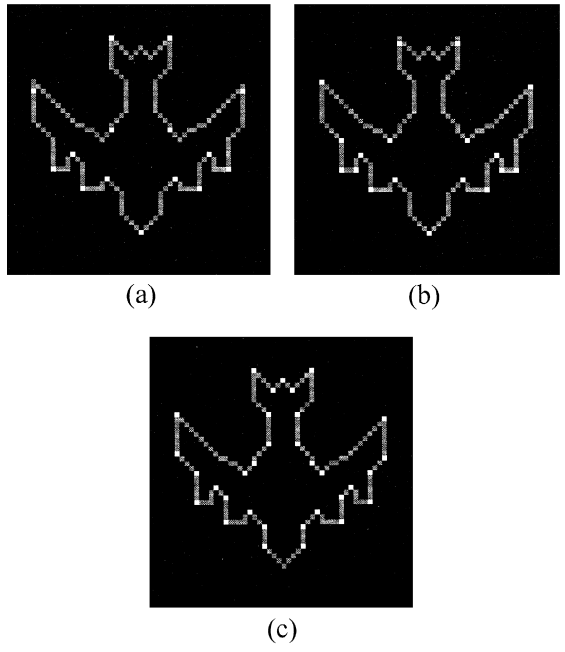


Fig. 6. The comparisons of our method with other methods for a minute object. (a) The result of the Rosenfeld–Johnston method. (b) The result of the Rosenfeld–Weszka method. (c) The result of our proposed method.

Finally, we will compare our method with previous works when being applied in different situations. As reminded before, boundary based algorithms such as the Rosenfeld–Johnston algorithm (Rosenfeld and Johnston, 1973) and the Rosenfeld–Weszka method (Rosenfeld and Weszka, 1974) suffer from the dependence of the correctness of region support. It is inappropriate that they determine the smoothing parameter fixedly. In 1989, Teh and Chin determined the region support automatically. But it is time consuming for tremendous objects since it needs to compute the curvature for each pixel firstly. Here, we consider two kinds of maps, the minute and tremendous object maps. For a minute object, Fig. 6(b) shows the result of the Rosenfeld–Johnston method, Fig. 6(c) shows the result of the Rosenfeld–Weszka method, where each smoothing parameter of theirs is ten. On the other hand, Fig. 7(a), Fig. 7(b) and Fig. 7(c) show the results of Rosenfeld–Johnston, Rosenfeld–Weszka, and our method on a tremendous object image segmented from another real image with size of 253×208, respectively. Moreover, we also apply our method on multiple objects image, Fig. 7(d) show the result. All of above our experiments, the size of structuring element is 7×7.

It is not so fortunate that circular structuring element perform well on both type of objects. There still exist defects as the structuring element can not fit the shape of object, especially, for minute objects.

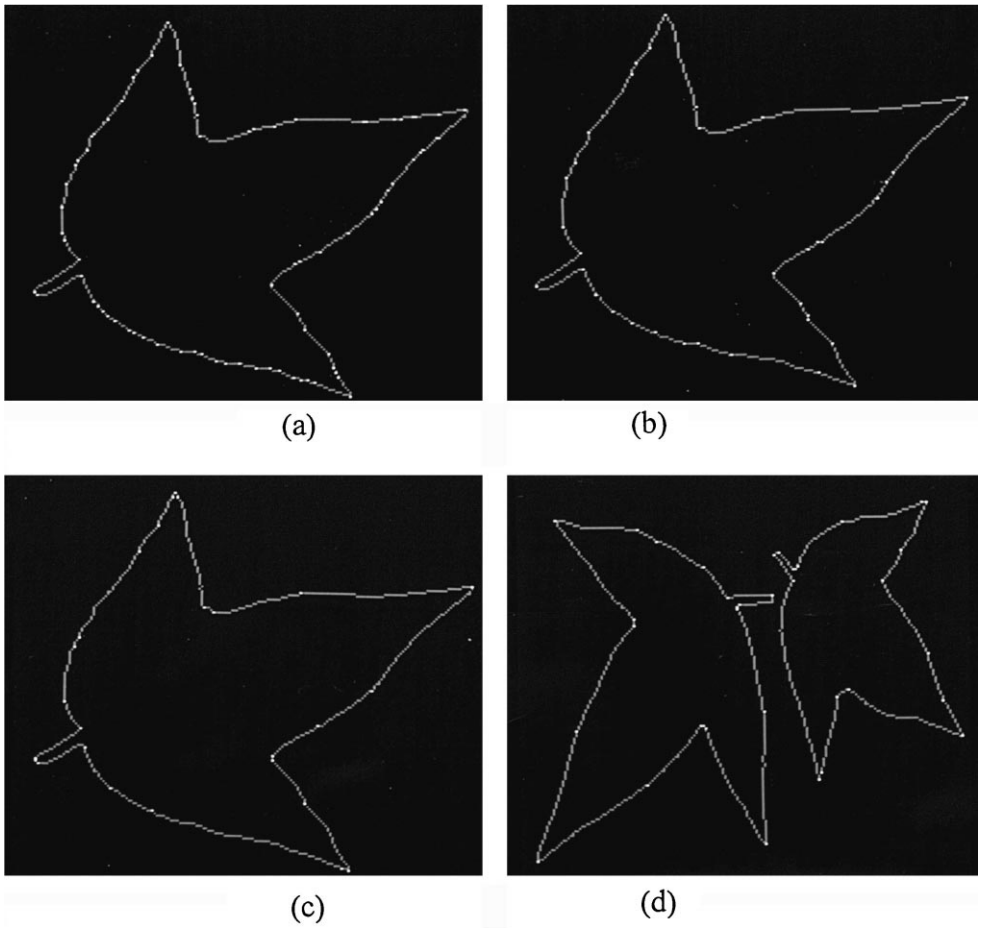
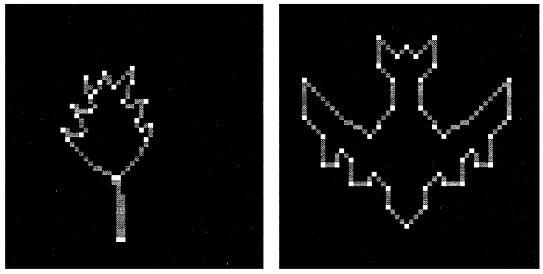


Fig. 7. The comparisons of our method with other methods for a tremendous object. (a) The result of the Rosenfeld–Johnston method. (b) The result of the Rosenfeld–Weszka method. (c,d) The result of our proposed method for single and multiple tremendous objects.



(a) (b)

Fig. 8. The results of our method with distance measurement. (a) The corner map of leaf image with threshold $t = 2.0$. (b) The corner map of civil aircraft image with threshold $t = 1.0$.

Hence, we join a secondary representative point between two contiguous significant corners with shape representation for minute objects. We use the maximum Euclidean distance as the measurement of secondary representative points. The intuitive concept is, for each pixel belong to a boundary segment, we compute the perpendicular Euclidean distance between the pixel and the chord connecting two contiguous significant corner points of the boundary segment. Let c_1 and c_2 be two contiguous significant corners detected from the above method, and $p_1, p_2, \ldots, p_{l-1}, p_l$ be the boundary points from c_1 to c_2, where $p_1 = c_1$ and $p_1 = c_2$, then the Euclidean distance from p_i to the chord connecting c_1 and c_2 is denoted as $d(p_i, \overline{c_1 c_2})$. If $d(p_{i-1}, \overline{c_1 c_2}) < d(p_i, \overline{c_1 c_2}) > d(p_{i+1}, \overline{c_1 c_2})$ and $d(p_i, \overline{c_1 c_2}) > t$, where t is a threshold reflecting the degree of the importance of the secondary representative corner. In other words, for polygonal approximation in shape analysis, the threshold t can be selected with smaller number to minimize the integral square $E_2 = \sum_i e_i^2$, where e_i denotes the error between a boundary segment and a line approximation. It is reasonable to select a lower threshold as the object is small. But, how could we select proper threshold t. Initially, a larger threshold is a default value. Then decrease the threshold step by step until $E_2 = \sum_i e_i^2$ did not decrease and then add the secondary representative points to actual corner point set. Fig. 8 shows the result for shape representation. In our experiment, we measure the merit E_2 of some algorithms on chromosome image in Table 3. The result shows that our method has a smaller error than other methods.

Table 3
The distance error of some algorithms with chromosome image

Algorithms	Total error $E_2 = \sum_i e_1^2$
Rosenfeld–Johnston, $m = 6$	14.131202
Rosenfeld–Weszka, $m = 6$	15.473706
Freeman–Davis, $s = 3$, $m = 2$	17.329360
Sankar–Sharma	24.873161
Teh–Chin (k-cosine)	10.081549
Teh–Chin (k-curvature)	7.481549
Teh–Chin (1-curvature)	7.481549
M. M. corner detector $t = 1$	7.286041

5. Conclusions

In this paper, we have presented a corner detection method with simple integer computation by mathematical morphological operators. As we know, corners are usually divided into convex and concave corners. First, we use morphological operators to extract connected regions containing convex and concave corners. Then we locate convex and concave corners as those points on the boundaries of the extracted regions with maximal N-hit numbers. In our experiment, we exhibit satisfied experiment results in detecting significant points and providing good approximation on shape description.

References

Freeman, H., Davis, L.S., 1977. A corner-finding algorithm for chain-coded curves. IEEE Trans. Comput. 26, 297–303.

Giardina, C.R., Dougherty, E.R., 1988. Morphological Method in Image and Signal Processing. Prentice-Hall, Englewood Cliffs, NJ.

Gupta, L., Malakapalli, K., 1990. Robust partial shape classification using invariant breakpoints and dynamic alignment. Pattern Recognition 23 (10), 1103–1111.

Liu, H.-C., Srinath, M.D., 1990. Corner detection from chain-code. Pattern Recognition 23 (2), 51–68.

Matheron, G., 1975. Random Sets and Integral Geometry. Wiley, New York.

Meyer, F., Beucher, S., 1990. Morphological segmentation. J. Visual Comm. and Image Representation 1 (1), 21–46.

Noble, J.A., 1992. Images as functions and sets. Image and Vision Comput. 10 (1), 19–29.

Pei, S.C., Horng, J.H., 1994. Corner point detection using nest moving average. Pattern Recognition 27 (11), 1533–1537.

Pei, S.C., Horng, J.H., 1996. Optimum approximation of digital

planar curves using circular arcs. Pattern Recognition 29 (3), 383–388.

Rosenfeld, A., Johnston, E., 1973. Angle detection on digital curves. IEEE Trans. Comput. 22, 875–878.

Rosenfeld, A., Weszka, J.S., 1974. An improved method of angle detection on digital curves. IEEE Trans. Comput. 24, 940–941.

Sankar, P.V., Sharma, C.V., 1978. A parallel procedure for the detection of dominant points on a digital curve. Comput. Graphics Image Process. 7, 403–412.

Sarkar, D., 1993. A simple algorithm for detection of significant vertices for polygonal approximation of chain-coded curves. Pattern Recognition Letters 14 (12), 959–964.

Serra, J., 1982. Image Analysis and Mathematical Morphology. Academic Press, London.

Shapiro, L.G., Macdonald, R.S., Sternberg, S.R., 1987. Ordered structural shape matching with primitive extraction by mathematical morphology. Pattern Recognition 20.

Teh, C.H., Chin, R.T., 1989. On the detection of dominant points on digital curves. IEEE Trans. Pattern Anal. Machine Intell. 11 (8), 859–880.

Zhang, X., Zhao, D., 1997. A parallel algorithm for detecting dominant points on multiple digital curves. Pattern Recognition 30 (2), 239–244.