Rectifying Reverse Polygonization of Digital Curves for Dominant Point Detection

Hamid Abbasi[1],Mohammad Olyaee[2],Hamid Reza Ghafari[3]

[1]Department of Computer Science and Engineering, Damghan Branch, Islamic Azad University, Damghan, Iran

[2]Department of Computer Science and Engineering, Gonabad Branch, Islamic Azad University, Gonabad, Iran

[3]Department of Computer Science and Engineering, Ferdows Branch, Islamic Azad University, Ferdos, Iran

Abstract

A polygonal approximation technique using reverse polygonization is presented in this paper. This method rectifies the reverse polygonization in previous publications since it captures an efficient approximation independent to all initial points of curves. previous algorithms do not account for this, the algorithm presented in this article is more accurate than previous, specially when specified dominants for deleting are near to each other and follow a particular pattern. Our new approach starts from an initial set of dominant points i.e. break points and dominant points are deleted (one in each iteration) in a way that the maximal perpendicular distance of an approximating straight line from an original curve is minimized. The deletion is performed in a way that optimizes the results of approximation. Leftover points will be considered as dominant points that are not related to the starting point. Finally, a comparative study with former algorithms is provided which prove that under usage of this new technique better approximation results are obtainable.

Keywords: *Reverse polygonization; Dominant points; Break points; Polygonal approximation*

1- Introduction

One of the most important problems in searches and classifications is determining dominant points of an object and their properties as that by the use of these points, the object can be described. If these points are decreased, the use of memory and time of computing will be efficient, but we should consider that decrease of these points should not destroy the main properties of object.

Polygonal approximation captures the essence of boundary shapes with fewest possible straight line segments. The term dominant point (DP) is assigned to the end points of these segments. This is one of the popular approaches which can provide good representation of 2D shapes at different resolutions. One obvious advantage of using this approach is high data reduction and its immediate impact on efficiency of subsequent feature extraction and/or shape matching algorithms. This representation gained popularity due to its simplicity, locality, generality and compactness. This representation simplifies the analysis of images by providing a set of feature points containing almost complete information of a contour. It also causes high data reduction and efficiency of subsequent procedures. Such dominant points are used in shape understanding [3], matching and pattern recognition algorithms [2, 17].

Rest of the paper is organized as follow: Section 2 presents related works, Section 3 analyze the reverse polygonization approach, Section 4 presents the rectifying of reverse polygonization for determining dominant points, Section 5 provides a comparison and discussion of these results and finally Section 6 concludes this presentation and outline the directions for future research.

2- Related work

To obtain properties of an object, firstly its dominant points must be determined. To determine these points generally a segmentation of object is necessary. If the segmentation is stable against various conditions and environmental changes, then the object will have more stable properties. Many researchers have been carried out related to the determination of segmentation of gray or color images, i.e. edge detection method based on the maximizing objective function and fuzzy edge detection [4, 12, 13, 15, 18].

Some algorithms were also represented to determine properties and dominant points, these algorithms capture the essence of boundary shapes with fewest possible straight line segments. Such algorithms can be classified into three main groups namely, sequential approach, split-and-merge approach and heuristic-search approach. For sequential approaches, Sklansky and Gonzales [14] proposed a scan-along procedure which starts from a point and tries to find the longest line segments sequentially. Ray and Ray [20] proposed a method which determines the longest possible line segments with minimum possible error. Teh and Chin [5] determined region of support for each point based on its local properties and computed its relative significance (curvature) and finally detected dominant points by a process of non maxima suppression. Kurozumi and Davis [6] proposed a mini max method which derives the approximating segments by minimizing the maximum distance between a given set of points and the corresponding segment. Marji and Siy [18] determined region of support using integral square error (ISE) and selected endpoints of support arm (as dominant points) depending upon the frequency of their selection. Prasad, Quek and Leung presents a non-heuristic and control parameter independent dominant point detection method that is based on the suppression of break points. For split-and-merge approaches, Held et al. and Nagabhushan et al. [1,7] proposed a split-and-merge technique in which difference of slope was used to split segments. Also Prasad [8] compares the error bounds of two classes of dominant point detection methods, methods based on reducing a distance metric and methods based on digital straight segments. For heuristic-search approach, an exhaustive search for vertices of optimal polygon will result in an exponential complexity. Dunham and Sato [9] used dynamic programming to find the optimal approximating polygon.

Among the above three groups of algorithms, sequential approaches are simple and fast, but quality of their results dependent upon the selection of start point. Proposed polygonal approximation technique in [20](reverse polygonization) does not fall in any of the above groups. However, it is closely related to first group. The current method in this paper is based on reverse polygonization, however in accordance with next discussions, the reverse polygonization for some of curves doesn't have adequate accuracy and may be dependant to starting point, thus in this paper through rectifying reverse polygonization, the new method was proposed to determine dominant points.

3-Review of reverse polygonization

In this section was considered the basic proposed algorithm, therefore after determining initial dominant points, we consider the problems of algorithm in determining dominant points.

3.1. Break point detection

Break points are detected using Freeman's chain-coding [20]. These break points become the initial set of dominant points in this algorithm. Fig 1 shows a polygonal of break points for a chromosome. We can determine these break points easily, for that an integer value ci varying from 0 to 7 is assigned to each curve point (pi) according to the position/direction of next point. Fig. 2 shows value of Freeman's chain code for all possible directions, Any points (pi) is a break point if its chain code (ci) is not equal to the chain code of previous point (pi_1). Break points are taken as initial set of dominant points, these are called as dominant points from here on. To calculate the error associated with each dominant point (DPj), two neighboring dominant points (DPj-1 and DPj+1) are joined by a straight line. Maximum perpendicular (squared) distance of all boundary points between DPj-1 and DPj+1 from the straight line is called as associated error value (AEV) of dominant points DPj. For creating this table, at first we place all break points in table and calculate AEV for each dominant points, then one dominant point, with minimum AEV, is eliminated from table and for both of its neighbor points calculate AEV.

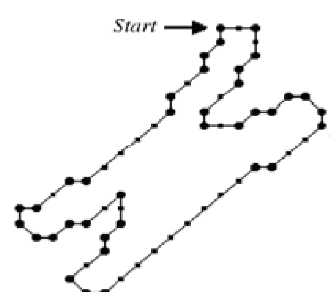

Fig. 1. Break points (in bold) for Chromosome shape connected by straight line.

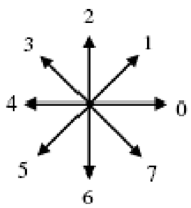

Fig. 2. Freeman's chain codes.

Reverse polygonization method, iteratively, removes the dominant points from the DP table until a polygonal approximation is up to required level. One dominant point, with minimum AEV, is eliminated in each iteration. If more than one dominant point with least AEV exists, any one of them may be removed. After deleting each point, we calculate AEV for two neighbor points. This algorithm functions as follows:

1- Find break points from given boundary points.

2- Make detected break points as initial set of dominant points.

3- Calculate AEV for all dominant points.

4-Do

a. Find dominant points having least AEV(DPj).

b. Remove DPj from the list of dominant points.

c. Recalculate the AEV of neighboring dominant points (DPj-1, DPj+1).

d-calculate Maxerr.

While Maxerr < threshold.

5-Leftover points make final set of dominant points.

3-2- Problems of algorithms

Represented simple algorithm correctly acts for many cures but has the problems when determined dominant points for deleting place near to each other, this algorithm states that if several points have minimal AEV, we can optionally delete one of them, and in next repetitions, leftover points are deleted, because in next repetitions, these points will have minimal AEV, thus this algorithm selects the first point having minimal AEV. This reasoning isn't correct on many positions, since deleting a point may change AEV of a point that in previous repetition had been a minimal value, and its new value may not be a minimum. At last it isn't selected at the next repetition, thus final leftover dominant points are related to start point and we don't have a unique set of final dominant points for different start points. For example in Fig. 3, points B and C may have minimum AEV, if point B is deleted, in accordance with algorithm we must calculate AEV for point C, that according to Fig. 5, that B is deleted, AEV of C may be different and not be minimum and doesn't delete at next iteration, it may imagine this problem will be produced when selected points to delete are neighbors, but this problem happened in other points too, for example at Fig. 4 points B and D have minimum AEV , with deleting B, AEV of C changes and probably it will be minimum AEV and delete at next iteration and D remains until the end of algorithm, thus above conditions may happen for other points after deleting some of these points, therefore this reasoning that if a set of points having least AEV and in each iteration one point is deleted, then all these points will be deleted, is incorrect. Therefore deleting a point may change AEV of other points and final result is related to starting point. On the other hand, if some of the selected points for deleting are neighbors, with deleting the first point, the AEV of it's neighbor was calculated again, this causes that AEV of two neighbors change and AEV of previous selected point might not be minimum, and this point remains until the end of algorithm and isn't deleted, while if the second is deleted, the algorithm may capture better approximation, thus selection of first point isn't correct, and if points are neighbors, we must select the point that through its deletion the best approximation is captured.

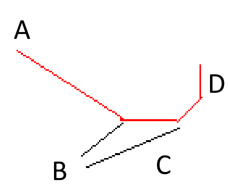

Fig 3. determined dominant points having minimal AEV

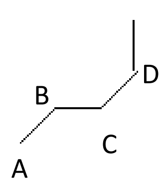

Fig 4 . points B and D having minimal AEV

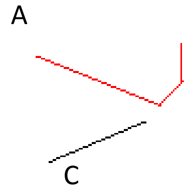

Fig 5. after deleting point B, AEV of point C changes

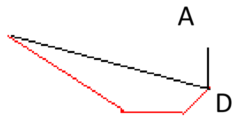

Fig 6. the captured approximation after deleting points B and C.

4.Rectified reverse polygonization algorithm

With respect to the represented problems, in this section an algorithm that captures efficient approximation of an object is presented. With application of this method final dominant points isn't related to start point. Break points become the initial set of dominant points, and algorithm deletes some of these points in accordance with the conditions. To clean up problems of the previous approach, the new method acts as such:

If several points have minimum AEV and these points have not influence on calculating AEV of each other, we can delete them all together, and calculate AEV for neighbors of deleted point, above process continues until a pre-determined threshold is achieved.

If two points having minimum AEV are neighbors, we must delete a point that better approximation is achieved by deletion of that point (with respected to Maxerr) .

In Fig.3, points B and C have minimum AEV, and they were selected to be deleted (point B is the first), if point B doesn't affect on calculating AEV of point C, we can delete points B and C all together, otherwise A and D are jointed with straight line, and Maximum perpendicular (squared) distance from points B and C to the straight line is calculated, we delete the point which has minimum AEV. Then to make a decision to delete other point, we should consider that Deletion of points B and C may produce inefficient approximation of curve and destroy important section of curve (Fig.6).

By applying above restriction, points are correctly deleted to obtain an efficient polygonal approximation of a curve and the set of leftover points are not related to the starting point. The proposed algorithm acts as follows:

1- Find break point from given boundary points;

2- Mark detected break points as initial set of dominant points;
3- Calculate AEV for all dominant points;
4- Do
   a. s=Find dominant points having least AEV;
   b. If points B and C from this set(points with least AEV) are neighbors, Find the best point from B and C and delete it;
   Else delete all points of this set.

   c. Recalculate the AEV for neighboring points of deleted dominant point
   d. Calculate MaxErr
   While MaxErr <threshold.

5- Leftover points make the final set of dominant points;

5- Simulation results

Presented algorithm includes several advantages to previous methods like flexibility and robustness. It is producing polygonal approximation with a desired number of dominant points with an easy implementation. Applied computation in the algorithm is effective because, first of all it was obtained break points applying

Fig 7: a picture of a tree's leaf

chain codes and AEV is calculated for each break points and during each repetition of operation, in exchange for every removing points, AEV is calculated just for two adjacent of this point. So it is favor as efficiency. To calculate accuracy of represented algorithm in extracting of dominant points and to compare it with previous methods, we are comparing this method with presented method in [8] in this part.

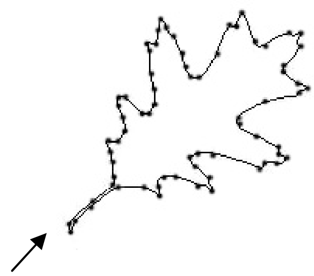

Fig 8: break points are obtained by chain code.

In order to compare, AEV took the same for both methods and MaxErr, ISE and nDP is calculated for each algorithms. Presented method, as we will see, indicate more desirable approximation results and remove the points to keep characteristics of principal picture. Compared to previous methods, obtained approximation was closer to main form. Removing the points in offering algorithm is not blindly and those points which cause inappropriate approximation is removed. Because of taking AEV the same in both method, number of remained dominant points are more than previous method but it is not noticeable. In Fig 7, there is a picture of leaf which break points in Fig. 8 are obtained by chain code (primary dominant points set) and then in Fig. 9 after applying the algorithm, all the dominant points with AEV fewer than two are removed. Finally, in Fig 10 the final picture is obtained with removing dominant points of Fig 9 (those with AEV fewer than 10).

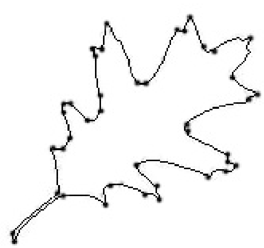

Fig 9: picture of the leaf after algorithm operation and primary removing points with AEV<2

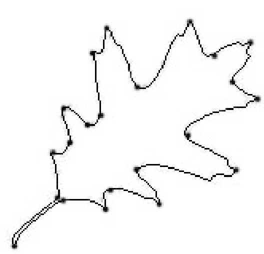

Fig 10: picture of the leaf after algorithm operation and removing the dominant points with AEV<10

Process of removing the dominant points and obtaining Fig 10 is shown in Table 1 to 3. Firstly, leaf boundary with 768 pixel and 70 primary dominant points are achieved using chain code algorithm, then all the dominant points in term of with AEV<2 are removed after applying the algorithm. After this step, DP table is updated and in next step all the dominant points are removed with AEV<10 that this step cause the creation of picture in Fig. 10.

Table 1: DP table, dominant points of the leaf showed with the few amount of AEV (CI=8) is marked in bold.

Dp	CI	AEV
1	1	9.1
2	5	4.3
3	**8**	**0.9**
4	26	2.5
5	31	1
6	41	0.9
7	46	1.2
8	53	9
9	60	6
10	67	2.8
11	74	1.3
12	81	2
13	85	5
14	90	2.2
-	-	-

Table 2: : DP table, picture of the leaf after updating AEV (dominant points with AEV<10 is marked in bold).

DP	CI	AEV
1	1	10.4
2	**5**	**8**
3	26	34
4	53	11
5	**60**	**9**
6	67	12
7	**81**	**2**
8	85	11.3
9	90	3.1
.	.	.

Table 3:comparing the results of proposed method for leaf

parameters	**Previous algoritm**			**new algoritm**		
N	745			745		
nDP	70	38	20	70	42	23
ISE	10.22	55.11	720	10.14	55.11	380.24
Maxrr	0.23	2.44	7.00	0.22	2.44	5.4

Fig 11: picture of Jellyfish

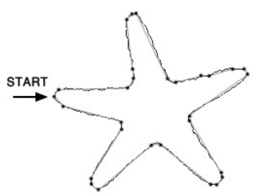

Fig 12: break points are obtained with freeman's chain code.

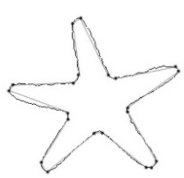

Fig 13: picture of Jellyfish after algorithm operation and removing the dominant points with AEV<2.

Fig 14: picture of Jellyfish after algorithm operation and removing the dominant points with AEV<10.

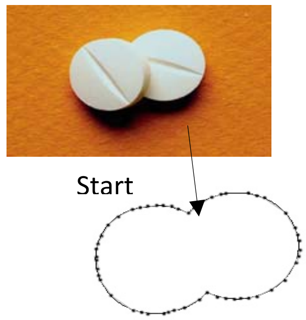

Fig 15: picture of a tab that break points are obtained using chain code.

Fig 16: picture of a tab after algorithm operation and removing the dominant points with AEV<10.

Fig 17: picture of a tab after algorithm operation and removing the dominant points with AEV<10.

Table 6 shows the results of applied algorithm in this picture. Referring to the above table, we can see that maximum error and total error decreased in new method than previous. It means that some dominant points in new method are not removed which played the role in structuring the picture and therefore can decrease the error. In fig 12 there is a picture of Jellyfish, this picture has a boundary with 901 pixel that there are 28 dominant points with AEV>2, so after algorithm operation in final step all of the dominant points are removed (with AEV<10).

(A)

(B)

Fig18-A: picture of map's boundary. B: break points are obtained by chain code.

Fig 19: the dominant points in map's boundary after algorithm operation and removing the dominant points with AEV<2.

Fig 20: the dominant points in map's boundary after algorithm operation and removing the dominant points with AEV<10.

As you see in picture 6 with the use of presented algorithm, the maximum error and total amount of error in Jellyfish picture are decreased. In some cases that a picture has an acute break in main boundary or a picture with a lot of differences in AEV of dominant points, amount

Table 4: dominant points are showed in bold with fewer amount of AEV. (AEV<2)

DP	CI	AEV
.	.	.
5	**108**	**1**
6	179	16
7	187	21
8	283	7
9	291	9
10	**333**	**2**
11	339	4
.	.	.

Table 5: Jellyfish's dominant points are showed with fewer amount of AEV. (AEV<10)

DP	CI	AEV
.	.	.
.	.	.
5	**179**	**9**
6	187	21
7	283	25
8	280	10
9	288	12
10	**375**	**9**
11	381	16
.	.	.

Table 6: comparing the results of previous and offered algorithm for the picture of Jellyfish.

parameters	**Previous algoritm**			**new algoritm**		
n	901			901		
nDP	28	12	11	28	22	10
ISE	59.22	447.11	770	57.56	410.25	704.35
MaxErr	0.85	10.5	25.3	0.22	7.3	25.1

of total error and maximum error in two algorithms are closer than previous ones. You can see a picture of tab in fig 15 that we want to extract the dominant points with the use of introduced algorithm. The break points are obtained by chain code and after algorithm operation, all of the dominant points with AEV<2 are removed. After this step, DP table is updated and in the next step all of the dominant points are removed with AEV<10. This step create the fig 17. It is observed that results of algorithm operation in this picture is the best and amount of both maximum error and total error are decreased. In tab's fig the best operation of algorithm observed rather than previous algorithm and the reason is crump form of the tab.

The extraction steps of dominant points of a map's boundary are indicated in figure 20 to 22. First, boundary of map with 1527 pixel and 76

Table7: comparing the results of previous and offered algorithm for the picture of tab.

parameters	Previous algoritm			new algoritm		
N	570			570		
nDP	49	18	12	49	26	17
ISE	4.22	17.77	354.5	4.22	11.25	22.35
MaxErr	0.25	2.4	10.59	0.25	1.01	5.21

dominant points is obtained by use of the chain code. After applying the algorithm, points with AEV<2 are removed, if two adjacent points are located in the same position, the one with a fewer AEV is removed. So after this step, the picture may have some dominant points with lower AEV than above step but they are not removed. This condition is applied in other pictures, too. In fig 18-A a picture of a map is indicated.

We want to extract the dominant points by use of presented algorithm. In fig 18-B the break points is obtained by use of chain code. After algorithm operation all of the dominant points are removed with AEV<2. So in this step, DP table is updated and in the next step all of the dominant points with AEV<10 are removed in which fig 20 is

Table8: comparing the results of previous and offered algorithm for the picture of map's boundary.

Parameters	Previous algoritm			new algoritm		
N	1527			1527		
nDP	32	21	9	76	24	20
ISE	173	3817.5	19078.5	162.55	2290.4	5344.5
MaxErr	8	9.3	24.73	8	9.1	11

achieved. In this picture as you see, the rate of error are decreased by use of presented algorithm. With studying the above pictures, it is seen that the algorithm have been successfully extracted the dominant points of various pictures independent to geometrical nature.

6- DISCUSSION

With respect to section 5 and obtained results on many images such as fish and flower that is considered in [8], we can say that the new method can obtain more efficient approximation with fewer iteration, few dominant points and low level of Maxerr, and we can say that obtained approximation is closer to original image. Also, according to comparison parameters, presented approach acts better than other algorithms.

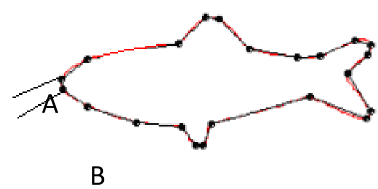

Fig 21. determined breakpoints have minimum AEV.

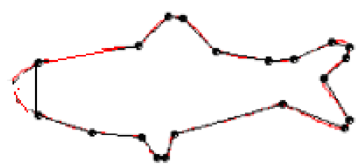

Fig 22. Captured approximation after deleting A and B

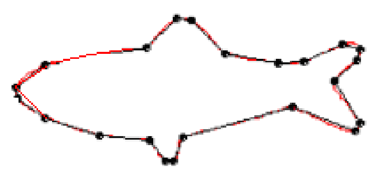

Fig 23. Captured approximation after deleting B.

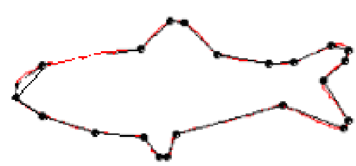

Fig 24. Captured approximation after deleting A.

For better understanding of the result, the previous method and the one presented in this article, is applied on the picture of a fish. In fig.21, if we only consider the head of fish, the existing dominant points may place in a way similar to fig.4 (for other sections, this situation can exist, too). The determined dominant points on the head of the fish have minimum AEV and so they are selected to delete, in reverse polygonization, if both points A and B are deleted, with 20 dominant points, approximation of fig.22 is captured, that unfortunately important section of the image was deleted and approximation is not accurate. It should be noted that the sequence of deletion of points A and B is important, and captured approximation is different. Because if the point A is deleted at first, the leftover point B may not have minimum AEV, and it remains until the end of algorithm, deletion of point B at first can cause the different situation. Deletion one of point A or B at first may present better approximation and we must make a decision to delete them. We delete a point that cause better approximation. At fig.23 and fig.24 it is illuminated that deletion of points A or B at first, present different approximation of object. These conditions may happen for other sections, and finally inefficient approximation is captured by reverse polygonization. On the other hand, if we act in accordance with new algorithm, final dominant points aren't related to starting point and they are a unique set. The leftover dominant points are not related to start point because we consider the points having minimum AEV in each iteration, if these points are not neighbor, we delete all of them and this deletion isn't related to start point, so the result is similar and independent to start point. If some of these points are neighbor, this algorithm selects the best point, so again in this situation this action is not related to start point and the result is similar and independent to start point.

This new method presents better approximation, because it removes previous problems of reverse polygonization algorithm and if two points that have minimum AEV are neighbors, this algorithm selects efficient approximation and acts similar to previous algorithm at other instances.

6- conclusion

This paper presents the method to determine dominant points. The new method removes problems that exist in former works, and presents an efficient approximation for all digital curves. This method is simple and flexible. This approach includes all robustness and advantages of reverse polygonization algorithm, and its calculation is similar to it. The difference between this method and the previous one is when several points have a minimum AEV, it performs a simple calculation and selects the point for deletion. This selection causes a more efficient approximation, moreover determined dominant points are not dependant to start point. The stable dominant points can be used for determining properties of an object and by these properties the problems of search and matching will be improved.

References :

[1] K. Abe, C. Arcelli, A. Held, Towards a hierarchical contour description via dominant point detection, IEEE Transactions on Systems, Man and Cybernetics 24 (6) (1994) 942–949.

[2] N. Ahuja, X. Hu, Matching point features with ordered geometric, rigidity, and disparity constraints, IEEE Transactions on Pattern Analysis Machinery Intelligence 16 (1994) 1041–1049.

[3] F. Attneave, Some informational aspects of visual perception,Psychological Review 61 (1954) 183–193.

[4] H.D.Cheng, Liming Hu, Ming Zhang. A high performance edge detector based on fuzzy inference rules. Information Sciences 177 (2007) 4768-4784.

[5] R. Chin , C. Teh, On the detection of dominant points on digitalcurves, IEEE Transactions on PAMI. 8 (1989) 859–873.

[6] W.A. Davis, Y. Kurozumi, Polygonal approximation by the minimax method , Computer Graphics and Image Processing 19 (1982) 248–264.

[7] R. Dinesh, D.S. Guru, P. Nagabhushan, Boundary based corner detection and localization using new 'cornerity' index: a robustapproach, Proceedings of the 1st Canadian Conference on Computer and Robot Vision (2004) 417–423.

[8] D.K. Prasad, Assessing Error Bound For Dominant Point Detection, International Journal of Image Processing (IJIP), Volume 6, Issue 5 (2012) 326-333

[9]J.G. Dunham, Optimum uniform piecewise linear approximation of planar curves, IEEE Transactions on Pattern Analysis Machinery Intelligence 8 (1986) 67–75.

[10] Mahmoud Fathy, Mohsen Sharifi, Maryam Tayefeh Mahmoudi. (n.d.). A Classified and Comparative Study of Edge Detection Algorithms. IEEE, Proceedings of the International Conference on Information Technology: Coding and Computing(2002)

[11] H. Freeman, On the encoding of arbitrary geometric configurations,IEEE Transactions on Electronic Computer 10 (1961) 260–268.

[12] Carl G. Looney, Lily Rui Liang. (n.d.). Competitive fuzzy edge detection. Applied Soft Computing 3 (2003) 123–137 .

[13] N. Sedaghat, H. Porreza, An Accurate Fuzzy Edge Detection Method Using Wavelet Details Subimages, Second International Conference on Digital Image Processing, Vol. 7546, 75461Q (2010).

[14] V. Gonzalez , J. Sklansky , Fast polygonal approximation of digitized curves, Pattern Recognition vol. 12 (1980) 327–331.

[15] G.Heidemann, Region saliency as a measure for colour segmentation stability, Image and Vision Computing 26 (2008) 211–227.

[16] R. Jain, I.K. Sethi, Finding trajectories of feature points in a monocular image sequence, IEEE Transactions on Pattern AnalysisMachinery Intelligence 9 (1987) 56–73.

[17] Chung-Chia Kang,Wen-June Wang. (n.d.). A novel edge detection method based on the maximizing objective function. Pattern Recognition 40 (2007) 609 – 618 .

[18] M. Marji, P. Siy, A new algorithm for dominant points detection and polygonization of digital curves, Pattern Recognition 36 (2003) 2239–2251.

[19]A. Masood, Dominant point detection by reverse polygonization of digital curves, Image and Vision Computing 26 (2008) 702–715

[20] B.K. Ray, K.S. Ray, Determination of optimal polygon from digital curve using L1 norm, Pattern Recognition 26 (1993) 505–509.714 A. Masood / Image and Vision Computing 26 (2008) 702–715

[21] W.H. Tsai, Attributed string matching with merging for shape recognition, IEEE Transactions on Pattern Analysis Machinery Intelligence 7 (1985) 453–462.

[22] D.K. Prasad, C. Quek, M. K. H. Leung, A Non-heuristic Dominant Point Detection Based on Suppression of Break Points, Image Analysis and Recognition, Lecture Notes in Computer Science Volume 7324,(2012), 269-276.

Hamid Abbasi received Master Degree of Artificial Intelligence from Mashhad Branch, Islamic Azad University in 2009. He is currently working towards his Ph.D. Degree at the Science and Research Branch, Islamic Azad University, Iran. His research interests include Chaotic System, Neuro fuzzy System and Image Processing.

Mohammad Olyaee received Master Degree of Artificial Intelligence from Mashhad Branch, Islamic Azad University in 2010. His research interests include Data mining, Bioinformatics, Neuro fuzzy System and Image Processing.

Hamid Reza Ghafari received the Ph.D. degree in Artificial Intelligence from Ferdowsi University, Mashhad, Iran in 2013. He is now a Professor in Ferdows Branch, Islamic Azad University, Ferdows, Iran