CEUR-WS.org/Vol-2210/paper18.pdf

Wave skeletonization algorithm for raster images

M A Kudrina[1] **and V S Mishenev**[1]

[1]Samara National Research University, Moskovskoe Shosse 34, Samara, Russia, 443086

Abstract. This article covers a wave skeletonization algorithm for raster images. It contains diagrams of algorithms for primary graph building and separating wave generation into wavelets. It is proposed to store the graph's structure in nested form to save memory and make it easier to modify. Deming orthogonal regression is proposed for primary graph optimization as part of successive approximations method.

1. Introduction

There is a number of tasks which require building raster image skeletons, e.g. handwritten text recognition, fingerprint comparison, processing medical or cartographical images, technical drawings [1-5], etc.

Various skeletonization algorithms exist, including Zhang-Suen algorithm [4], template method, wave algorithm [1, 6, 7], etc. The wave algorithm is covered in this article.

Essence of the wave algorithm for raster image skeleton calculation has already been described and illustrated in detail particularly in [1, 4, 6]. However, when implementing programs with this method, a number of questions, subtleties and nuances may arise, which are the subject of this article. The article contains diagrams of algorithms for primary graph building and separating wave generation into wavelets. Scientific novelty of this publication is that the graph describing object's skeleton is stored in a nested structure instead of connectivity and incident matrices which saves memory and makes it easier to change graph's structure, and also that Deming orthogonal regression is used for primary graph optimization as part of successive approximations method.

Image skeleton refers to a set of points equidistant from image borders. Several preparatory steps are usually taken before object's skeleton is built. These steps include image preprocessing (eliminating minor noise, etc.) and binarization. Binarization refers to the process of converting color and halftone images into dual-color, i.e. binary, images. Binarization is done using thresholding, point transformation, folding and border enhancement methods, extracting low-frequency and high-frequency image components, etc.

For our purpose, wave skeletization algorithm can be divided in two stages: primary graph building, including wave initiation and tracking covered distance, separation and attenuation points, followed by graph optimization where excessive information is dropped. Workflow for these stages is illustrated in figure 1.

At the first stage, spherical wave is initiated within an object. Wave generation is the wave front. Points that belong to certain wave generations are marked by different colors in the image. Central pixels of each odd generation included in the primary graph as its nodes are tracked.

However, primary graph usually contains a lot of excessive information, so redundant point are eliminated at the optimization stage.

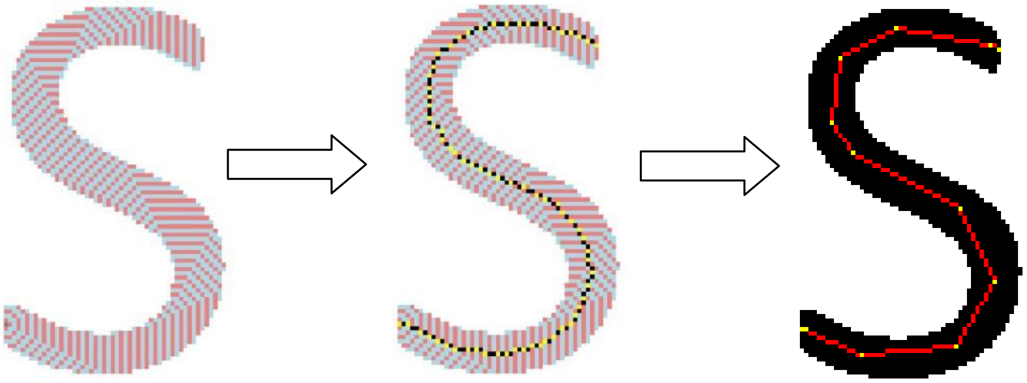


Figure 1. Key skeletonization stages - primary graph building and graph optimization.

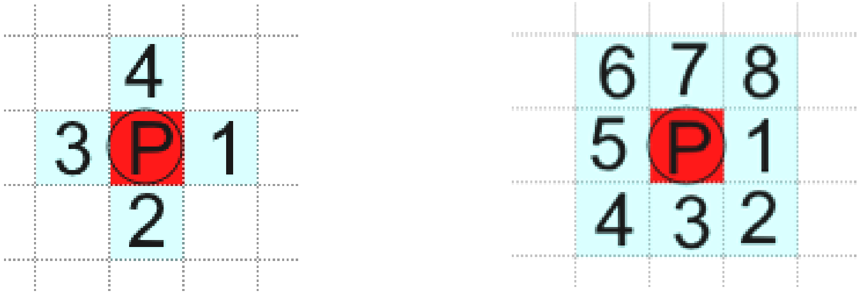
2. Primary graph building stage

Graph is built by tracking spherical wave propagation path over the image.

2.1. Spherical wave propagation

Spherical wave is initiated at any pixel inside the object.

Spherical wave generation is obtained by using 4- and 8-connected propagation alternatively. Connectivity refers to number of pixels adjacent to the current pixel. For 4-connected wave propagation, the pixels above, below, to the left and to the right of the current one are considered its neighbor pixels (see figure 2(a)). For 8-connected propagation, accordingly, all 8 pixels around the current one are considered its neighbor pixels (see figure 2(b)). In figure 2, the current pixel is designated as P, and its neighbors for 4-connected and 8-connected representations are numbered.



(a) (b)

Figure 2. Neighbor pixels and their winding order for 4-connected (a) and 8-connected (b) representations.

When a wave is initiated, the 4-connected propagation is applied first; then 8-connected propagation is applied to each pixel of the previous 4-connected generation, and so on alternately. This way, the wave propagates in octagonal shape. This approach is based on Huygens-Fresnel wave theory principle positing that each element of wave front can be considered as secondary disturbance center producing secondary spherical waves.

There are several special aspects to spherical waves:

- wave propagation becomes steady regardless of its initial point after at most 2N steps, where N is the line width in pixels (see figure 3);
- such wave is “able to” turn and effectively go round various obstacles. Small one- or two-pixel obstacles have little impact on wave propagation. However, it is recommended to eliminate such obstacles at the binary image generation stage for wave stability.

Zero generation consists of one initial pixel that is marked as first in the image. It is located at the center in figure 4.

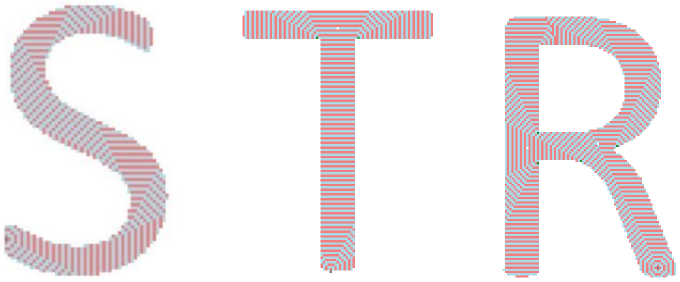
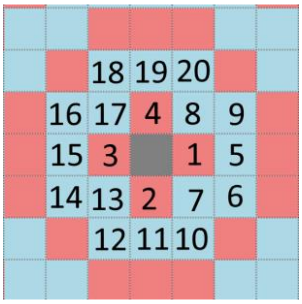


Figure 3. Spherical wave propagation.



		18	19	20		
	16	17	4	8	9	
	15	3		1	5	
	14	13	2	7	6	
		12	11	10		

Figure 4. Numbering of first and second generation pixels.

First wave generation consists of unmarked neighbors of the initial pixel at 4-connected propagation. First generation pixels are marked red on the image. In figure 4, they have numbers 1, 2, 3, 4. Secondary generation is unmarked neighbors of the first generation pixels at 8-connected propagation (in figure 3, they have numbers 5-20), etc.

For each point with (i, j) coordinates, checking whether its neighbor pixels have been marked is performed in fixed order - clockwise, always starting from the right middle pixel. This way, the check is done in order shown in figure 2(a) at 4-connected propagation or in order shown in figure 2(b) at 8-connected propagation. You may notice from figure 4 that the winding order for the second generation pixels in the example differs from order of pixels in the image. This is due to fixed order of marking neighbor pixels; however, this simplifies the algorithm.

2.2. Graph's structure

Object’s graph is stored in memory as a nested structure [8]. When this method of storing graph structure is used, each of its nodes is determined by a nest consisting of (x, y) coordinates of graph node and a list of addresses for transitions to next (child) nests (graph nodes). Advantages to this method are that it allows to easily modify graph’s structure and does not require much RAM as opposed to connectivity or incident matrices.

An example of graph and corresponding nested structure are shown in figure 5. Nests contain graph node coordinates (x_i,y_i) and A_i – a reference to the list of references to neighbor nests (graph nodes).

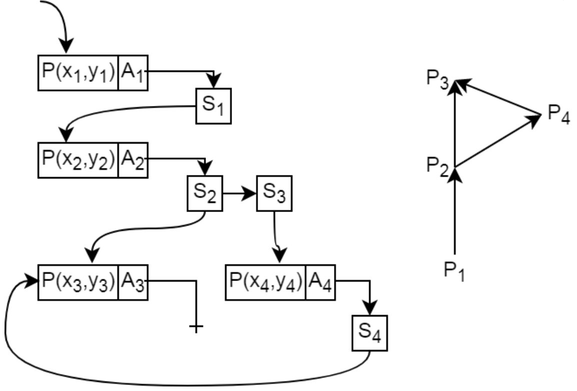


Figure 5. Nested structure for storing an object’s graph.

2.3. Key primary graph building algorithms

A diagram of the primary graph building algorithm is shown in figure 6.

Figure’s skeleton is built by tracking centers of each odd wave generation. Current wave generation is stored in memory as a list of pixel coordinates of this front. Let us designate the list of points of any odd generation as S_4, because its points are 4-connected neighbors of previous generation’s points.

Accordingly, we shall designate lists of even generations as S_8, because they are 8-connected neighbors of previous generation’s points.

Each generation of a wave is located between extreme pixels that have coordinates (x_1, y_1) and (x_2, y_2). Extreme pixels of a wave are pixels that have at most one neighbor in their generation. For the figure skeleton point we shall take the midpoint of segment that connects extreme points of wave generation; thus, the point's coordinates are $((x_1+x_2)/2; (y_1+y_2)/2)$.

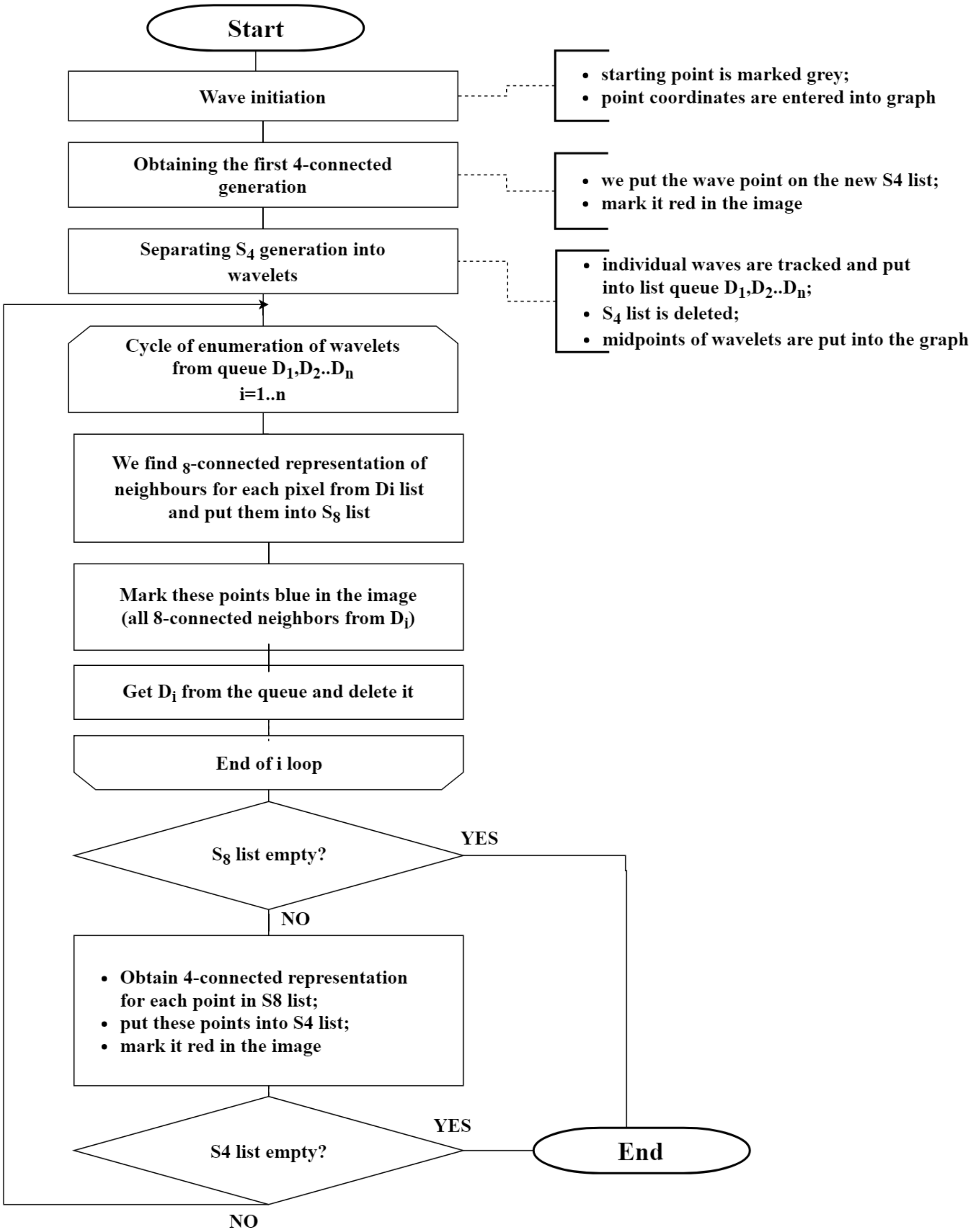

Start
Wave initiation
• starting point is marked grey;
• point coordinates are entered into graph
Obtaining the first 4-connected generation
• we put the wave point on the new S4 list;
• mark it red in the image
Separating S_4 generation into wavelets
• individual waves are tracked and put into list queue $D_1, D_2..D_n$;
• S_4 list is deleted;
• midpoints of wavelets are put into the graph
Cycle of enumeration of wavelets from queue $D_1, D_2..D_n$ i=1..n
We find 8-connected representation of neighbours for each pixel from Di list and put them into S_8 list
Mark these points blue in the image (all 8-connected neighbors from D_i)
Get D_i from the queue and delete it
End of i loop
S_8 list empty?
YES
NO
• Obtain 4-connected representation for each point in S8 list;
• put these points into S4 list;
• mark it red in the image
S4 list empty?
YES
NO
End

Figure 6. Diagram of the primary graph building algorithm.

Besides, wave generation may include several separate fronts. This happens when wave generation is separated into wavelets.

This situation can be identified using the number of extreme points in S_4 list of generation points. If there are more than two extreme points, separation has taken place and several wavelets are in the list (an example of a wave divided into three fronts is shown in figure 7). In this case, separate lists of wavelets (D_1, D_2,.., D_n) are queued. Separation of S_4 wave generation list into D_1, D_2,.., $D_n$wavelets is shown in figure 8.

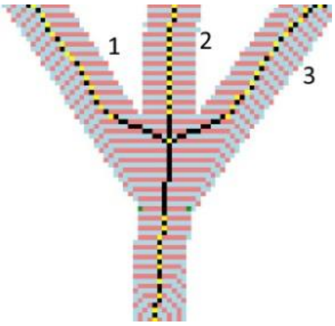


Figure 7. Wave generation separated into wavelets.

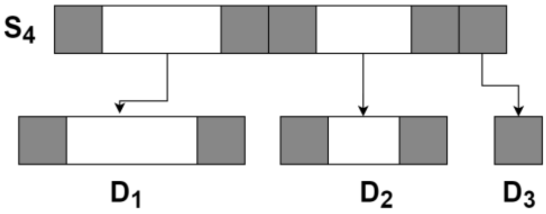


Figure 8. List of wave generation points separated into three fronts.

Algorithm diagram for separating wave generation (S_4) into wavelets (D_1, D_2,.., D_n) is shown in figure 9.

Before separation, wave width steadily grows and the number of wave generation extreme points increases. By tracking this, it is possible to identify junction point of image skeleton segments. Point of junction of separate branches is calculated as barycenter of the polygon formed by extreme points of the parent wave and wavelets [5, 6, 9].

Whenever a wave turns or an image element becomes thicker, the number of extreme points remains constant, thus the wave width increase can be neglected.

A wave attenuates when encountering another wave. In this case, the graph closes.

3. Graph optimization stage

The first method used to optimize the graph (i.e. eliminate redundant information) is successive approximations method. It works as follows: a straight line is drawn through the first two nodes of the graph, and a check of deviation of each successive point from that line is done. If the deviation is within ε limit (which is specified comparable to thickness of lines in the image), the point belongs to this line; otherwise, it forms the next line.

In successive approximations method, equation coefficients of the line $Ax + By + C = 0$ going through points (x_1, y_1) and (x_2, y_2) are calculated using the following formulas:

$$A = y_1 - y_2, B = x_2 - x_1, C = x_1 y_2 - y_1 x_2.$$

Then, for each successive point (x_i, y_i) of graph, its distance from the line $Ax + By + C = 0$ is calculated using the formula $r = |Ax_i + By_i + C| \cdot (A^2 + B^2)^{-1/2}$. If $r < \varepsilon$, the point belongs to the current line.

The obtained set of points belonging to the same line is approximated using orthogonal regression. Orthogonal regression differs from simple linear regression in that it makes allowances for observation errors along both x and y directions (see figure 10).

If points are approximated by the line that is determined by equation $y = \beta_0 + \beta_1 x$, regression coefficients are calculated using the following formulas [10]:

$$\beta_1 = \left(s_{yy} - s_{xx} + ((s_{yy} - s_{xx})^2 + 4{s_{xy}}^2)^{1/2}\right) \cdot \left(2s_{xy}\right)^{-1},$$
$$\beta_0 = \bar{y} - \beta_1 \bar{x},$$

where

$$\bar{x} = \frac{1}{n}\sum x_i,\ \bar{y} = \frac{1}{n}\sum y_i,$$
$$s_{xx} = \frac{1}{n-1}\sum(x_i - \bar{x})^2,$$
$$s_{yy} = \frac{1}{n-1}\sum(y_i - \bar{y})^2,$$
$$s_{xy} = \frac{1}{n-1}\sum(x_i - \bar{x})(y_i - \bar{y}).$$

Since the equation y$= \beta_0 + \beta_1 x$ does not determine vertical lines, let us put it to the form $Ax + By + C = 0$. For that purpose, we shall calculate the following coefficients A, B, C: $A = \beta_1, B = -1, C = \beta_0$.

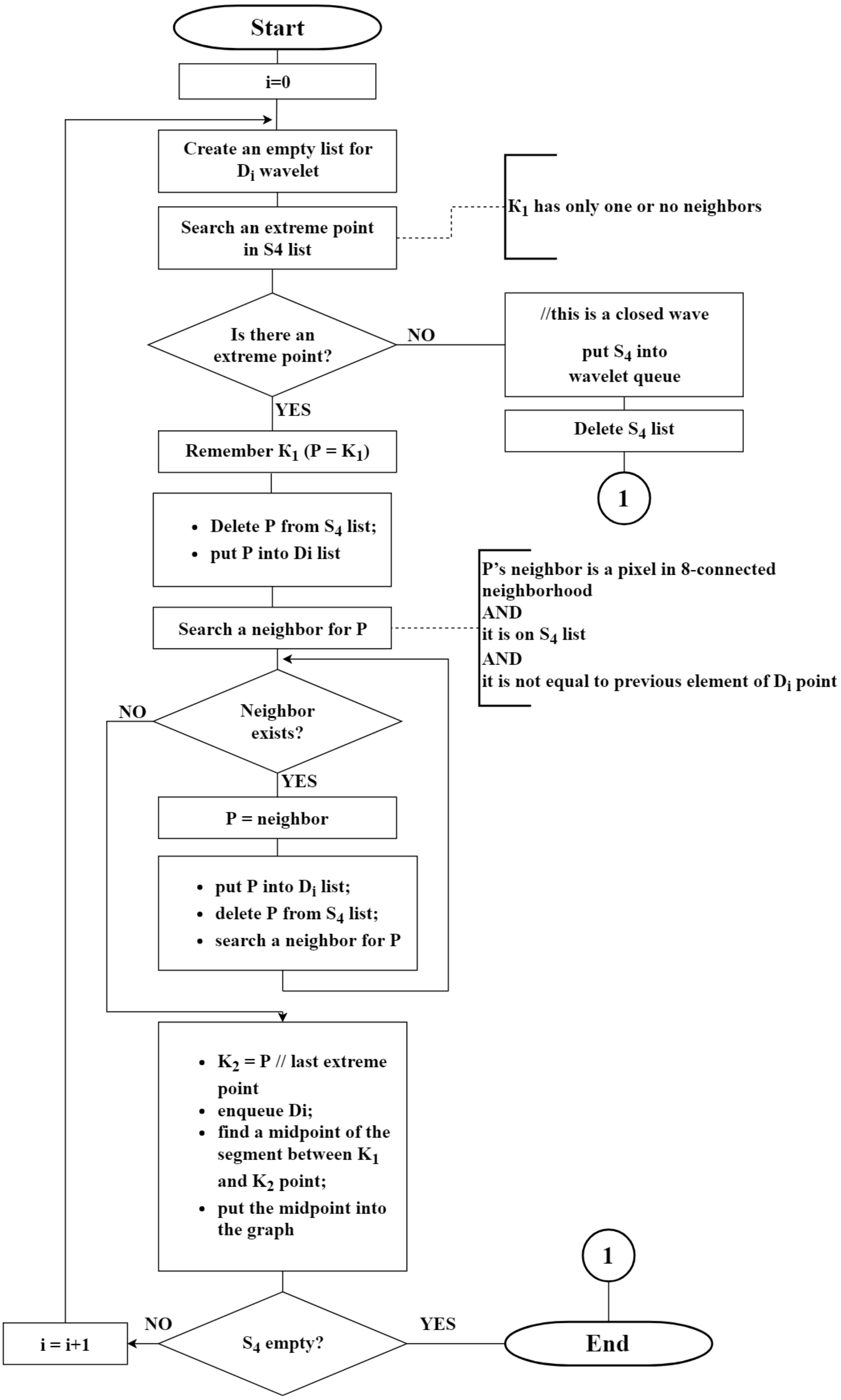

Start
i=0
Create an empty list for D_i wavelet
Search an extreme point in S4 list
K_1 has only one or no neighbors
Is there an extreme point?
NO
//this is a closed wave
put S_4 into wavelet queue
Delete S_4 list
1
YES
Remember K_1 ($P = K_1$)
Delete P from S_4 list;
put P into Di list
Search a neighbor for P
P's neighbor is a pixel in 8-connected neighborhood
AND
it is on S_4 list
AND
it is not equal to previous element of D_i point
NO
Neighbor exists?
YES
P = neighbor
put P into D_i list;
delete P from S_4 list;
search a neighbor for P
K_2 = P // last extreme point
enqueue Di;
find a midpoint of the segment between K_1 and K_2 point;
put the midpoint into the graph
1
NO
YES
i = i+1
S_4 empty?
End

Figure 9. Algorithm diagram for separating S_4 generation into wavelets D_1, D_2,.., D_n.

If $s_{xx} = 0$, the line is parallel to Y axis, and the equation looks like $x = \bar{x}$. In this case, the line has the following coefficients: $A = -1, B = 0, C = \bar{x}$.

If, on the other hand, $s_{yy} = 0$, the line is parallel to X axis, and the equation looks like $y = \bar{y}$. In this case, the line has the following coefficients: $A = 0, B = -1, C = \bar{y}$.

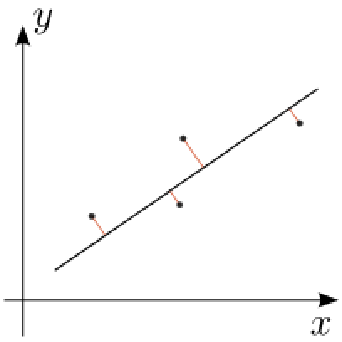


Figure 10. Approximation of graph node set using Deming orthogonal regression.

After the coefficients of approximating line have been found, extreme points are adjusted, i.e. projected onto approximating line, whereas other intermediary points are eliminated from the graph.

Coordinates (x_{proec}, y_{proec}) of projection of extreme point (x_0, y_0) onto line $Ax + By + C = 0$ obtained with orthogonal regression method are calculated using the following formulas:

$$x_{proec} = (B(Bx_0 - Ay_0) - AC)/(A^2 + B^2),$$
$$y_{proec} = (A(Ay_0 - Bx_0) - BC)/(A^2 + B^2).$$

Because, in general, the graph may be closed, all processed nodes are marked so the program does not get caught in an endless loop.

4. Results and conclusions

In this article, diagrams of algorithms for primary graph building and separating wave generation into wavelets are proposed. A suggestion is made to store the structure of the graph representing object's skeleton in nested form instead of connectivity or incident matrices to save memory and make the graph's structure easier to modify. Scientific novelty of this publication is also that Deming orthogonal regression is used in successive approximations method for primary graph optimization.

A wave skeletonization program has been developed using *C#* language. The program proactively converts color raster image into binary image using thresholding where the threshold is an average between maximum and minimum brightness. Then the primary graph is built and optimized.

Wave skeletization algorithm does not require much memory because memory is only allocated to store graph's skeleton and lists of pixels of the current and next wave generations. Spherical wave marking is done in the source image which is easily restorable. The algorithm is easily parallelized when several starting points of wave initiation are used. For wave initiation, it is preferable to choose pixels on lower or upper borders of symbol images; this makes skeletons smoother.

Today, wave skeletization is used successfully to build skeletons of symbols (letters, digits, etc.) for further processing as part of recognition process. However, this algorithm is not as effective for building fingerprint skeleton, because finger papillary picture has more complex structure, thus wave propagation and further analysis have poor accuracy [1].

References

[1] Gudkov V Yu and Klyuev D 2015 Skeletonization of Binary Images and Finding of Singular Points for Fingerprint Recognition *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics* **15(3)** 11-17

[2] Khaustov P A 2017 Algorithms for handwritten character recognition based on constructing structural models *Computer Optics* **41(1)** 67-78 DOI:10.18287/2412-6179-2017-41-1-67-78

[3] Glaister P 2001 Least squares revisited *The Mathematical Gazette* **85** 104-107

[4] Zhang T Y and Suen C Y 1984 A fast parallel algorithm for thinning digital patterns *Comm. of ACM* **27(3)** 236-239 DOI:10.1145/357994.358023

[5] Saeed K, Tabędzki M, Rybnik M and Adamski M 2010 K3M: A universal algorithm for image skeletonization and a review of thinning techniques *International Journal of Applied Mathematics and Computer Science* **20(2)** 317-335 DOI:10.2478/v10006-010-0024-4

[6] Klubkov I 2001 Primeneniye volnovogo algoritma dlya nakhozhdeniya skeleta rastrovogo izobrazheniya *Vestnik DGTU* **1(7)** 9

[7] Fisher R, Perkins S, Walker A and Wolfart E 2000 Skeletonization *Medial Axis Transform* (Access mode: http://www.dai.ed.ac.uk/HIPR2/skeleton.htm) (31 May 2018)

[8] Poulovassilis A and Levene M 1994 A Nested-Graph Model for the Representation and Manipulation of Complex Objects *ACM Transactions on Information Systems* **12(1)** 35-68

[9] Mishenev V S and Kudrina M A 2017 Building Raster Image Skeleton Using Wave Algorithm *Trudy Mezhdunarodnogo simpoziuma Nadezhnost' i kachestvo* **1** 378-381 (in Russian)

[10] Haeckel R, Wosniok W and Klauke R 2013 Comparison of ordinary linear regression, orthogonal regression, standardized principal component analysis, Deming and Passing-Bablok approach for method validation in laboratory medicine *Laboratoriumsmedizin* **37(3)** 147-163 DOI:10.1515/labmed-2013-0003