Skeleton-Based Algorithm for Increasing Spectral Resolution in Digital Elevation Model

Rolando Quintero, Serguei Levachkine, Miguel Torres, and Marco Moreno

Centre for Computer Research, National Polytechnical Institute, Mexico City, Mexico
{quintero,sergei,mtorres,mmoreno}@cic.ipn.mx
http://geo.cic.ipn.mx, http://geopro.cic.ipn.mx

Abstract. In this work, we present an algorithm for increasing spectral resolution in DEM. The algorithm is based on the 8-connected skeleton of polygons formed by the contour lines and to prune this skeleton by translating it into a graph. This is an alternative to the processes of vector interpolation. With this approach, it is possible to find new elevation data from the information contained in DEM and generate new data with the same spatial resolution.

1 Introduction

Nowadays, Digital Elevation Models (DEM) have gained popularity in applications for simulating natural disasters. Nevertheless, these applications require a huge amount of data. In many cases, the available data do not present enough quality for simulation processes. The Statistics, Geography and Informatics National Institute of Mexico (INEGI) produces DEM with 50 meters of resolution [1][2], but some simulation processes require a better level of detail (1 meter is the standard). In all cases, DEM are generated by means of contours. These have different representations and thresholds of separation. For instance, in the topological maps of INEGI, the contours are separated by 10 meters and 5 meters near the coast. In Simulation processes like flooding simulations we need more detailed information (less than 1m. of resolution).

In this work, we propose an algorithm to increase spectral resolution of DEM for simulation processes; the algorithm is based on a skeletonization. In [3], we present an application to handle the huge quantity of data contained in DEM for real-time rendering by discriminating the less significant elevation data, without changing the *semantics* of the raster data. However, we cannot improve the quality of the more relevant data to obtain additional information. In [4], we extended the algorithm to increase the spatial resolution of DEM. In this paper, we introduce a method for increasing spectral resolution of DEM. It is important to lineout that the algorithms mentioned above were developed to be used with raster data sets, although similar considerations can be taken for vector data sets.

By using the algorithm presented in this work and the previous jointly, we can solve the problem of 3D data representation and generate virtual scenes, which are ready to navigate, either by simulations or by defined trajectories [5].

In the next section, we present some frameworks of the underlying theory of the skeletonization making different approaches. In Section 3, we give the pertinent considerations for the application of skeletonization in DEM and outline the proposed

approach. In Section 4, some results and tests are presented and analyzed. Finally, Section 5 describes our conclusions.

2 Skeletonization

The skeleton of a region is defined by means of the Medial Axes Transform (MAT) proposed by Blum [6]. The MAT of a region R with edge B is defined as follows: Find the *nearest* neighbor for each point p of R, if p has more than one neighbor then p belongs to the medial axis (skeleton) of R. The concept *"nearest"* depends on a distance, while the results of MAT depend on the selection of a distance measure. Fig. 1 shows some examples that use Euclidean distance as measure [7].
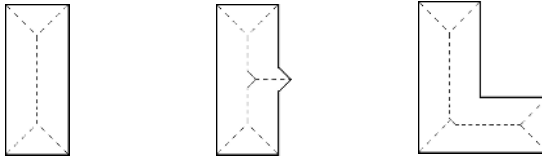

Fig. 1. Medial Axis of some simple geometrical shapes

Strictly speaking, the skeleton of a region should fulfill the following conditions:
- Be a connected subset of points from the original region.
- Represent the geometric characteristics of the original region.
- Preserve some topological characteristics of the original region, such as connectivity and holes.

In general, all skeletonization methods of binary images can be classified in two groups: *pixel-based* methods and *non-pixel-based* ones. In the first case, every pixel belonging to an object in the image is used for computation of the skeleton (thinning and distance transformations).

On the other hand, non-pixel-based methods attempt to determine analytically the symmetric points of a shape from opposite points in the shape edge. The thinning techniques provide different types of skeletons, depending on the connectivity desired for the skeleton. In general, there are four types of skeleton connectivity (Eqn. 1) They are: 8-connected, 4-connected, 6-right-connected and 6-left-connected.

$$d_8(p,q) = \max(|x-u|,|y-v|),$$
$$d_4(p,q) = |x-u|+|y-v|,$$
$$d_{6R}(p,q) = \max(|x-u|,|y-v|,|x-u+y-v|),$$
$$d_{6L}(p,q) = \max(|x-u|,|y-v|,|x-u-y+v|),$$

$$(1)$$

For a pixel p, from an image I, exists four types of neighborhoods, depending on the type of connectivity, they are denoted by v_8, v_4, v_{6R} y v_{6L}. In Fig. 2 the neighborhoods mentioned are depicted.

Thinning is the process that consists of deleting all pixels of a set of objects with region R and edge B, which fulfill the following conditions:
- Pixel p is not an extreme, i.e. it has just one neighbor belonging to the object.
- If p is deleted, the connectivity must not be affected.

X	X	X
X	p	X
X	X	X

(a)

	X	
X	p	X
	X	

(b)

	X	X
X	p	X
X	X	

(c)

X	X	
X	p	X
	X	X

(d)
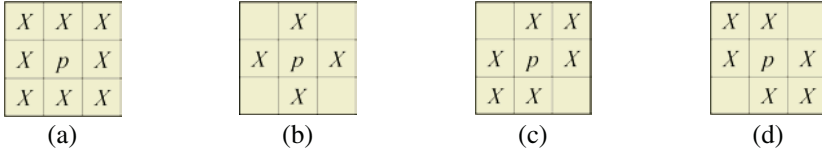
Fig. 2. Set of neighbors for a pixel p using different connectivity criteria. (a) 8-connected. (b) 4-connected. (c) 6R-connected. (d) 6L-connected

A classic thinning algorithm consists of the following stages: First, determining the edge pixels. To determine these pixels, it is necessary to use a dual connectivity criteria (if we want a 8-connected skeleton, then we must use 4-connected criteria). Once determined the edge pixels, we must delete the pixels that are not necessary. The conditions for determining if a pixel p can be deleted are defined in Eqn. 2 (the rotated versions of masks are conditions too).

$$\sum p_i = 0 \wedge \sum p_j = 1 \wedge \sum p_k = 1, i \in \{3,7\}, j \in \{8,1,2\}, k \in \{6,5,4\},$$
$$\sum p_i = 0 \wedge \sum p_j = 1 \wedge \sum p_k = 1, i \in \{1,3\}, j \in \{2\}, k \in \{4,5,6,7,8\},$$

(2)

Eqn. 2 can be explained by the masks that result from all possible combinations of the equations (see Fig. 3). If an edge pixel gives a true result by applying any of the masks of Fig. 3, then this pixel cannot be deleted. Additionally, it is necessary to test if the pixel is an extreme one (Eqn. 3). If it is true, then the pixel cannot be deleted.

$$\sum p_i = 1, i \in \{1,3,5,7\},$$

(3)

This process is iteratively repeated, while we can delete pixels. The result of the process is the 8-connected skeleton of the original object.

a_1	a_2	a_3
0	p	0
b_1	b_2	b_3

b_1	0	a_1
b_2	p	a_2
b_3	0	a_3

a_1	0	1
a_2	p	0
a_3	a_4	a_5

a_3	a_2	a_1
a_4	p	0
a_5	0	1

a_5	a_4	a_3
0	p	a_2
1	0	a_1

| a_5	a_4	a_3
0	p	a_2
1	0	a_1

Where:
$$\sum a_i = 1$$
$$\sum b_i = 1$$

Fig. 3. Masks used to obtain the 8-connected skeleton

3 Applying Skeletonization for Increasing DEM Spectral Resolution

In this paper, we propose the use of a skeletonization to obtain a new contour between two known ones. In previous section, we present the process to compute the skeleton of a binary image. In this section we define how to generate the new contour from the skeleton of the region between the two known contours. We will call this region *Equi-Heigth* region or EH regions.

Generally, we do not have the DEM of a whole world, i.e. we should work with DEM that describe a region. Due to this, many times the contours are interrupted in the edges of the image. Hence, the EH regions can be incomplete; in this case we have identified three cases (See Fig. 4):

- *Case A.* The contours that define EH region are completely inside of the image, forming a blob with at least one hole.
- *Case B.* The contours begin and end outside of the image, forming a strip across the image.
- *Case C.* The same as case B, but there are holes in the strip.

(a)   (b)   (c)
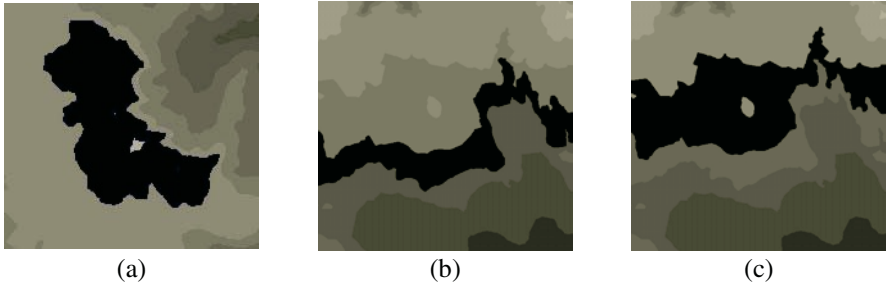
**Fig. 4.** Cases considered in the raster analysis

In Fig. 5 we present the skeletons obtained for EH regions of Fig. 4. As we can see, each skeleton has a lot of branches that are not suitable for being part of the contour. We can discriminate the noise branches (prune the skeleton). In same Fig. 5 the noise lines are in light gray, while the contour is in black.

(a)   (b)   (c)
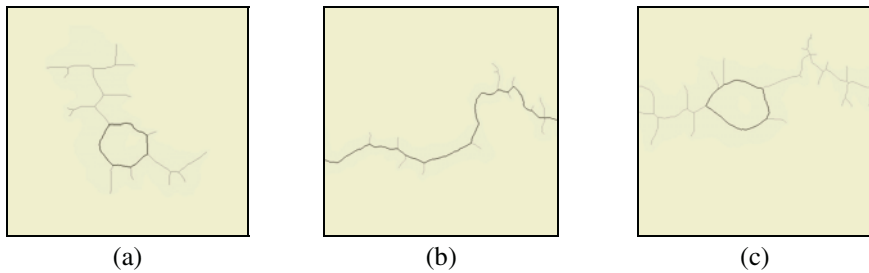
Fig. 5. The skeletons for the examples depicted in Fig. 4

Once having the skeleton, it is necessary to obtain the contour from it by pruning the skeleton. To perform this, we propose to generate a graph that describes the morphology of the skeleton. Thus, it is possible to find the contour of EH region by using a graph that describes the skeleton (see Fig. 6). In other words; it is possible to prune the skeleton using its graph; see in the following.

To generate the graph from the skeleton, we use on the fact that the skeleton is 8-conncted. Let I be a binary image containing a skeleton and p_{ij} the value of the image matrix $I(i,j)$. The first element of the image f is defined by Eqn. 4.
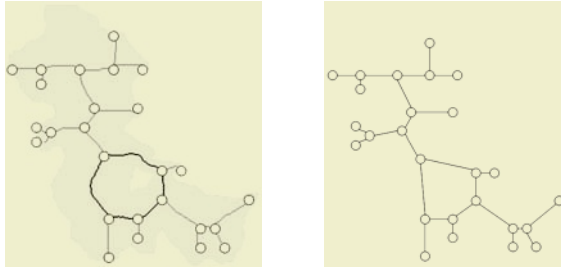
Fig. 6. Transformation of skeleton into graph

$$f = p_{ij} \ni p_{ij} \neq 0, p_{kl} = 0, k < i, l < j, \tag{4}$$

Let p be a pixel of the image and $n(p)$ the number of 8-connected neighbors. Then we define T as the set of terminal pixels (Eqn. 5). Similarly, we define A as the set of edge pixels (Eqn. 5) and R as the set of triad pixels (Eqn. 5). While, the set of vertices V is given by Eqn. 6.

$$T = \{p : n(p) = 1\},$$
$$A = \{p : n(p) = 2\}, \tag{5}$$
$$R = \{p : n(p) \geq 3\},$$
$$V = \{f\} \cup T \cup R. \tag{6}$$

Let p be an image pixel and $N(p)$ the set of the 8-connected neighbors of p, then we define a branch s by Eqn. 7. Also, $l(s)$ denotes the length of a branch s, and $first(s)$ and $last(s)$ defines the extreme elements of a branch s.

$$s = \{p_1, p_2, ..., p_n\} \ni p_i \in N(p_{i+1}), p_1 \in V, p_n \in V,$$
$$l(s) = card(s) = n,$$
$$first(s) = p_1, \tag{7}$$
$$last(s) = p_n.$$

We now define the graph representing the skeleton of the image as $G(V, E)$ where $E = \{s : s \text{ is a branch}\}$. Also, we define a path w on the graph G by Eqn. 8. Moreover, we define the length of a path $\lambda(w)$, the set of all paths in G as $W(G)$, and $\omega(G)$ is the longest path in G.

$$w = \{s_1, s_2, ..., s_n\} \ni s_i \in E, last(s_i) = first(s_{i+1}), i = 1, ..., n-1,$$
$$\lambda(w) = \sum_i l(s_i) \ni s_i \in w \tag{8}$$
$$\omega(G) = w_i \ni \lambda(w_i) \geq \lambda(w_j), w_i \in W(G), w_j \in W(G)$$

Once we have obtained the graph from skeleton, we simply take the following criteria for discriminating noise branches:

- If there are loops in graph, then all branches that are not in one loop are eliminated.

- If a branch is in more than one loop, then that branch is eliminated.
- If there are not loops in graph, then all branches outside of the longest path in graph are eliminated.

The next definitions are used to discriminate branches. A path b is a loop if $last(s_n)=first(s_1)$. So, let $B(G)$ be the set of all loops in graph G and $S_R(G)$ the set of redundant branches in G (Eqn. 9). Then we define the candidate contour C as is denoted in Eqn. 9.

$$S_R(G) = \{s : s \in b_i, s \in b_j, i \neq j, b_i \in B(G), b_j \in B(G)\},$$
$$C(G) = \{s : s \in b, b \in B(G), s \notin S_R(G)\}, \tag{9}$$

Finally, let C_N be the resulting contour within the skeleton, defined by Eqn. 10.

$$C_N = \begin{cases} C(G) \Leftrightarrow C \neq \phi \\ w(G) \Leftrightarrow C = \phi \end{cases}. \tag{10}$$

4   Tests and Results

We have developed an application implementing the described process. The contours that have been obtained compose DEM. Nevertheless, we have found some pathological cases in the skeletonization as in well as the pruning process. Fig. 7 is depicted a pathological case in the skeletonization process: parasite branches (circled in Fig. 7.a). This case has been solved by applying some morphological operations (erosion, dilatation and aperture) to the image before obtaining the skeleton. The result is shown in Fig. 7.b.

(a)   (b)
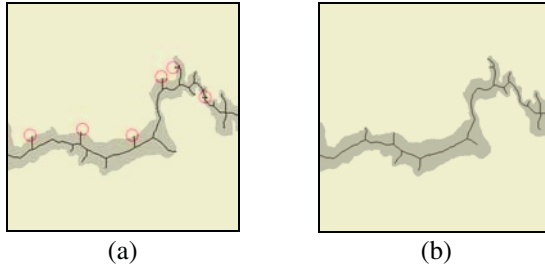
Fig. 7. Pathological case in the skeletonization

A pathological case of the pruning is illustrated in Fig. 8: there are no loops in the graph and there is a triad pixel near the edge of the image, Also the larger branch does not accomplish the edge of the image. Hence, the larger branch becomes parasite, so it does not belong to the contour. The problem can be easily solved by defining the set of edge pixels and requested the larger path in the graph begins and ends at edge pixel.

Fig. 9 shows the final step of processing of our algorithm. Moreover, Fig. 9.a depicts the found contour in the context of the DEM. This contour is integrated into DEM (Fig. 9.b).

Fig. 8. Pathological case in the pruning

| (a) | (b) |

Fig. 9. Final result. (a) Contour discovered into original DEM. (b) Modified DEM according to the source

5   Conclusions

In this work, we have presented an algorithm for increasing spectral resolution in DEM. The algorithm is based on the 8-connected skeleton of polygons formed by the contour lines; and to prune this skeleton by transforming it into a graph.

The skeletonization algorithm provides 8-connected skeleton. We identify some pathological cases in which the skeleton has parasite branches. These branches do not affect the pruning algorithm. Nevertheless we reduce the number of parasites by applying morphological operations (erosion, dilatation and aperture) on the EH regions.

For the definition of the pruning algorithm, we describe the cases that can be DEM. This algorithm consists of the transformation of the skeleton into a graph. In the graph, the singular pixels (extreme, terminal and triad pixels) are the vertices and all other pixels form the edges of the graph. Once obtained the graph, criteria for deleting edges and vertices are defined. This conversion of the skeleton into graph can be used in other pruning algorithms, by changing the deletion criteria.

The algorithm is an alternative to the processes of vector interpolation. With this approach, it is possible to find new elevation data from the information contained in DEM, and generate new data with the same spatial resolution. The use of this approach jointly with the previous works [3][4] allow processing huge quantity of data contained in DEM for in simulation and visualization processes, optimizing its performance.

Acknowledgments

References

1. Modelos Digitales de Elevación, Generalidades y Especificaciones, Instituto Nacional de Estadística, Geografía e Informática de México, Aguascalientes, México, 1999 (ISBN: 970-13-2511-7).
2. Normas Técnicas para la Elaboración de Ortofotos Digitales, Instituto Nacional de Estadística, Geografía e Informática de México, Aguascalientes, México, 1999 (ISBN: 970-13-2510-9).
3. Quintero, R., Guzman, G., Torres, M. and Moreno, M.: Spatial Analysis Based on 3D Rendering Techniques, *Proc. of 6th Association of Geographic Information Laboratories Europe conference on Geographic Information Science (AGILE 2003)*, Lyon, France, april 24 – 26, 2003.
4. Quintero, R., Moreno, M., Torres, M. and Levachkine, S.: Scale Changing of Digital Elevation Models using Bicubic Parametric Patches for Real-time Simulations, *Proc. of 8th Iberoamerican Congress on Pattern Recognition (CIARP 2003)*, La Habana, Cuba, 2003.
5. Reddy, M. and Leclerc, Y.: Enabling Geographic Support in Virtual Reality Modeling with Geo-VRML, Journal of Cartography and Geographic Information Science, 26(3), 1999.
6. Blum, H.: A transformation for extraction new descriptors of shape. *Proc. of symp. models for the perception of speech and visual form*, W. W. Dunn (Ed), MIT Press: Cambridge MA, 1967, pp. 362-380.
7. González, R. and Woods, R.: Digital image processing, Second Edition, Ed. Prentice Hall, 2002.