

A New Sequential Thinning Algorithm to Preserve Topology and Geometry of the Image

Ashwin.M.¹, Dinesh Babu.G.²

^{1,2}Dept of Computer Science and Engg, Adhiyamaan College of Engineering, Anna University

Abstract - A thinning algorithm is used to reduce unnecessary information by peeling objects layer by layer so that the result is sufficient to allow topological analysis. It has several applications, but is particularly useful for skeletonization. A new sequential thinning algorithm is introduced to preserve both the topology and geometry of the object. In sequential thinning, only a single point may be deleted at a time and it always guarantees the preservation of the topology of the original image. It is based on removing the central pixel in the 3x3 neighborhood of the candidate pixel which preserves the topology and geometry. The algorithm is based on computing the local Euler number before and after removing the candidate pixel and then checking whether there is a difference in the computed values. Furthermore, in order to preserve the geometric criteria (preserve end points of the object), we consider change of the boundary length when removing the central pixel.

Index Terms: skeletonization techniques, sequential thinning algorithm

I. Introduction

Topology and geometry are the attributes that uniquely define a shape. Two objects are said to have the same topological structure if one can be morphed into the other without tearing and gluing, whereas geometry describes the relative position of points on a surface. Skeletonization is the process of reducing foreground regions in a binary image to skeletal remnant that preserves the connectivity of the original image while throwing away most of the original foreground pixels/voxels. In other words, given an input binary image, skeletonization changes non-skeletal object pixels/voxels into background pixels/voxels. The results of skeletonization processes are skeletons. Definition of the skeleton is given by the gross-fire or wave fronts propagation analogy by Soille (1999): the body of an object is set on fire and is propagating at uniform speed within the body; the skeleton is the set of points where the fire or wave fronts meet. The purpose of skeletonization for the 2D case is to reduce 2D discrete objects to 1D linear representations preserving topological and geometrical information.

The skeleton generated by a skeletonization process is expected to have the following desirable properties (as given by Bernard and Manzanera (1999)): Homotopy: the skeleton must preserve the topology of the original image. In other words, the

Euler number should be the same before and after skeletonization, One-pixel-thickness: the skeleton should be made of curves or surfaces, i.e. one-pixel-thick objects. In other words, the skeleton should be as thin as possible, Mediality: the skeleton should lie in the middle of the shape, with every point of the skeleton having the same distance from the two closest borders of the object, Rotation invariance: skeletonization and object rotation should commute, Noise immunity: the skeleton should be fairly insensitive to noise on the image (boundary pixels added or removed), Reconstructibility: it should be possible to reconstruct the original image from the skeleton.

In real world there is a need for skeletonization of images due to following reasons: To reduce the amount of data required to be processed, To reduce the time required to be processed, Extraction of critical features such as end-points, junction-points, and connection among the components, The vectorization algorithms often used in pattern recognition tasks also require one-pixel-wide lines as input, Shape analysis can be more easily made on line like patterns.

II. Related work

To get a skeleton, different techniques can be used. These techniques are classified into three major categories each of which is discussed below.

- a. Distance transformation
- b. Voronoi methods
- c. Thinning technique

a. Distance transformation

The distance transform at a point within an object is defined as the minimum distance to a boundary point. Since the skeleton is required to be centered with respect to the object boundary, the distance transform gives useful clues for point removal. Points closest to the center of the object would have the maximum distance transform value. Several distance metrics can be used to compute the distance transform, for instance, the Euclidean distance, manhattan and chessboard distance metrics (Lohman 1998).

Skeletonization based on distance transformation is performed using the following three steps: 1.The original (binary) image is converted into feature and non-feature elements. The feature elements belong to the boundary of the

object, 2.The distance map is generated where each element gives the distance to the nearest feature element, 3.The ridges (local extremes) are detected as skeletal points. The distance transformation can be executed in linear ($O(n)$) time in arbitrary dimensions (where n is the number of the image elements, e.g. pixels or voxels). This method fulfills the geometric requirement but the topological correctness is not guaranteed.

b. Voronoi methods

The Voronoi diagram of a discrete set of points (called generating points) is the partition of the given space into cells so that each cell contains exactly one generating point and the locus of all points which are nearer to this generating point than to other generating points. If the density of the boundary points (as generating points) goes to infinity then the corresponding Voronoi diagram converges to the skeleton. Voronoi methods have the advantage of being very well defined and theoretically sound since they operate on a well known and powerful concept - the Voronoi diagram. Both the topological and geometrical requirements of a skeleton can be fulfilled by the skeletonization based on Voronoi diagrams, but it is an expensive process, especially for large and complex objects

c. Thinning technique

A thinning algorithm is used to reduce unnecessary information by peeling objects layer by layer so that the result is sufficient to allow topological analysis. A thinning algorithm reserves the topology which implies, for example that an object cannot vanish completely or be split into two or more objects, and no background components can be created or merged. It also should preserve the geometry of the object. Thinning can be done in two ways: by using morphological operators and deleting simple points.

Morphological operators employ a method that deletes simple points. A pixel or voxel p is considered to be simple if it is a boundary point and its neighbor set with p and with p removed has exactly the same number of connected components. Deleting a simple point does not change the local topology or connectedness properties of the set. This method needs additional criteria for deleting simple points which prevent excessive shrinking. This involves identifying end-points of branches and disallows their deletion even if these points happen to be simple points.

The second class of thinning algorithms allows the simultaneous deletion of many points at a time. These algorithms are called parallel thinning algorithms as voxels/pixels may be deleted in parallel. Simultaneous deletion of points may destroy the topology of the image.

Euler number

Topology preserving can also be defined in terms of the Euler number. The Euler number describes the connectivity of components and is a topologically invariant property of an object. It is calculated by using the Euler Poincare formula:

- Euler number = number of connected components - number of tunnels + number of cavities in 3D and
- Euler number = number of connected components - number of holes in 2D. It can also define in terms of the number of vertices, edges and faces as:
- Euler Number = number of vertices - number of edges + number of faces.

Therefore, two objects are topologically equivalent if their Euler number is the same. In the subsequent sections we will use the Euler number to study topology preserving skeletonization.

III. A New Sequential Thinning Algorithm

In this paper, with the aim of finding a topology preserving 2D thinning algorithm, we first employ an algorithm based on removing the central pixel in the 3x3 neighborhood of the candidate pixel which preserves the topology and geometry and, then try to adapt the algorithm to the 3D case. The 3x3 neighborhood of the point to be removed is coded as in Table 1 with the Candidate pixel to be removed at the center.

1	2	4
8	16	32
64	128	256

Table 1: Encoding of 3x3 neighborhood configuration of the central pixel.

The algorithm is based on computing the local Euler number before and after removing the candidate pixel and then checking whether there is a difference in the computed values. Furthermore, in order to preserve the geometric criteria (preserve end points of the object), we consider change of the boundary length when removing the central pixel. This involves finding the optimal values for the difference of the boundary length so that the resulting skeleton maintains its end points in addition to having the same Euler number as the original image. This depends on the type of neighborhood configuration chosen. For all the possible neighborhood configurations ((4,8), (6.1,6.1),(6.2,6.2),(8,4)), the Euler number before and after skeletonization is calculated and a check is made whether there is a difference in the Euler number after and before skeletonization and whether the central pixel is a foreground pixel. For the geometric criteria, we consider the difference in the boundary length before and after removing the candidate pixel. Optimal acceptable values for the boundary length difference as well as the type of skeleton generated based on these values have been found.

An outline of the algorithm that we used for 2D skeletonization is given below.

1. A look up table is generated which corresponds to the 8 neighbor pixels with different cases for each of the 4 neighborhood configurations. This lookup table stores a Boolean value depending on the Euler number difference before and after removing the candidate pixel and the extent of the boundary length difference.

2. We scan the image sequentially and for each pixel we consider its 3x3 neighborhood. We calculate the values for the current configuration of the candidate pixel and compare these values with the values stored in the lookup table. If the two values match, then the candidate pixel is turned into background as it is not part of the skeleton. Otherwise, it is retained and will be part of the skeleton. The fact that the current pixel is removed will be taken into account when processing subsequent candidate pixels. In the algorithm, consideration is made to treat boundary points.

IV. Experimental Results

The test bodies considered are rectangles. The optimal values for the difference in the boundary length are found to be different under different neighborhood configurations. The values for a pair of neighborhood configurations coincide. In other words, the values for (4, 8) and (6.1, 6.1) neighborhood configurations are identical and the values for (8, 4) and (6.2, 6.2) are also the same. These values along with the type of skeleton they can produce are given in Table 2 and Table 3.

The following results were obtained during the experiment:

- Euler number is preserved for the (4, 8) neighborhood configuration for all types of skeletons for the test bodies used.
- Euler number for (6.2, 6.2) neighborhood configuration was preserved for all types of skeleton while (6.1, 6.1) neighborhood configuration was not tested for preservation of Euler number.
- Euler number is preserved for the (8, 4) neighborhood for all types of skeletons for the test bodies used.

For complex images the algorithm also showed a slight difference in the Euler number of the original image and its skeleton which is due to the boundary effects. But when the image is padded with zeros along its boundaries, the Euler number for the (4, 8) and (6.2, 6.2) is preserved. But Euler number is not preserved for the (8, 4) neighborhood. (4, 8)

Boundary length difference	Resulting skeleton
≤ -0.95	a point
between -0.68 and -0.94	a vertical or horizontal line
between -0.28 and -0.67	in between horizontal/vertical line and a skeleton with End points preserved
between -0.27 and 0.39	a skeleton with its end points preserved
≥ 0.40	no change in the original image

Table 2: Optimal values for boundary length difference for (4, 8) and (6.1, 6.1) Adjacency pairs

between -0.28 and -0.94	a vertical or horizontal line
between -0.27 and 0.39	a skeleton with its end points preserved
≥ 0.40	no change in the original image

Table 3: Optimal values for boundary length difference for (8, 4) and (6.2, 6.2) Adjacency pairs

Application to sample 2D image data

Since the original image shown in Figure 4 is not suitable for skeletonization, we performed a preprocessing step. That is, we first applied closure by using the structuring element $f_{15,15,15,15,15,15,15,15,15}g$ and then opening by the structuring element $f_{3,3,3,3,3,3,3,3,3}g$ to remove the noise. Finally it is padded with zeros along its boundaries to take account of boundary effects. Therefore, after these preprocessing steps, the filtered image shown in Figure 5 is obtained. By applying our skeletonization algorithm to this filtered image, we obtained a skeleton which preserves both topology and geometry for (4, 8) and (6.2, 6.2) adjacency pairs while topology is not preserved for the (8, 4) adjacency pair. But our opinion is that there should be an error in the algorithm for the Euler number calculation for the (8,4) adjacency as the number of connected components for the resulting skeleton are easily counted to be nine which is equal to the number of connected components of the original image. However, the number of connected components for the (8, 4) adjacency pair using the program gives an Euler number of 2836 for the skeleton which is not the correct result.

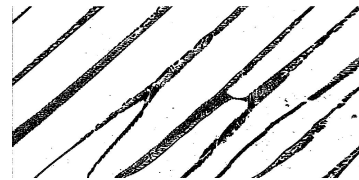


Figure 4: original image

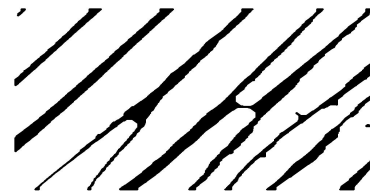


Figure 5: The original image closed by the structuring element: $f_{15,15,15,15,15,15,15,15,15}g$ and then opened with the structuring element $f_{3,3,3,3,3,3,3,3,3}g$ to remove the noise and then padded with zeros along its borders

Boundary length difference	Resulting skeleton
≤ -0.95	a point

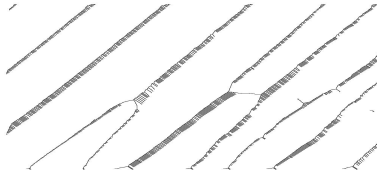


Figure 6: Skeleton for (4, 8) adjacency and boundary length difference ≥ 0.20 . Euler number: Original: 9, Skeleton: 9

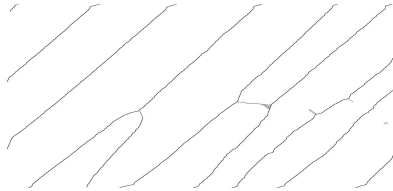


Figure 7: Skeleton for (6.2, 6.2) adjacency and boundary length difference ≥ 0.20 . Euler number: Original: 9, Skeleton: 9

V. Conclusion:

We presented a new sequential thinning algorithm for binary images which preserves the topology and geometry of the image. In the case of 2D binary images, we considered the 3x3 neighborhood of the candidate pixel. We used the difference in the local Euler number before and after removing the candidate pixel and as well as the difference in the boundary length as a criteria for determining whether a given pixel belongs to the skeleton or not. While the Euler number is used for determining the topological criteria, the difference in the boundary length is used to determine the geometric criteria. Using these criteria, we got the expected results for our test bodies (rectangles and squares) and also for a sample 2D image. In other words, we arrived at a topology preserving sequential thinning algorithm which preserves the topology and geometry of the original image and gives different skeletons depending on the values in the boundary length difference. All the possible adjacency pairs (4, 8), (8, 4), (6.1, 6.1), (6.2, 6.2) were considered and expected results obtained. Further work can be done with the 3D binary images and testing the adjacency systems for the three dimensional case.

Reference:

1. Sajjad Baloch, and Hamid Krim, *Object Recognition through Topo-Geometric Shape Models Using Error-Tolerant Sub graph Isomorphisms*, IEEE transactions on image processing, vol. 19, no. 5, May 2010
2. Michielsen, K. and De Raedt, H. 2000. *Morphological Image Analysis*. Computer Physics Communication 132, 94-103.
3. Borgefors, G., Nyström, L., and Di Bija, G.S. 1999. *Computing Skeletons in three dimensions*. Pattern Recognition 32, 1225-1236.
4. Toriwaki, J. and Yonekura, T. 2002. *Euler Number and Connectivity Indexes of a 3D digital Picture*. Review Forma, 17, 183-209.
5. Ohser, J., Nagel, W. and Schladitz, K. 2002. *The Euler number of discretized sets - on the choice of adjacency in homogenous lattices*. In: Mecke KR, Stoyan D, eds. *Morphology of Condensed Matter*. Heidelberg: Springer.
6. Ohser, J., Nagel, W. and Schladitz, K. 2003. *The Euler number of discretized sets - surprising results in three dimensions*. *Image Analysis and Stereology* 22, 11-19.
7. Serra, J. 1982. *Image Analysis and Mathematical Morphology*, Vol.1. London: Academic press.
8. Soille, P. 1999. *Morphological Image Analysis: Principles and Application*. <http://www.inf.u-szeged.hu/lagyi/skel/skel.html>
9. *Object recognition from sub-sampled image processing submitted by Alex North in September 2005*.
10. *Skeletons in Digital Image Processing published by Gisela Klette in July 2002*.
11. *Skeletons in Digital Image Processing published by Gisela Klette in July 2002*.
12. *Algorithms for 2-Dimensional Object Recognition published by A.Ashbrook and N.A.Thacker in 1998*.
13. *Digital Image Processing using MATLAB by Gonzalez, Woods & Eddins, Prentice Hall, 2004*.