

SKELETONIZATION BY BLOCKS FOR LARGE 3D DATASETS: APPLICATION TO BRAIN MICROCIRCULATION

C. Fouard^{1,2,5}
F. Cassot²

G. Malandain¹
C. Mazel⁵

S. Prohaska³
D. Asselot⁵

M. Westerhoff⁴
J.P. Marc-Vergnes²

¹ EPIDAURE, INRIA, 2004 route des lucioles, BP 93
06 902 Sophia-Antipolis Cedex, France

⁴ INDEED Visual Concept Berlin, Allemagne

² U455, INSERM Toulouse, France

³ Zuse Institute Berlin (ZIB), Allemagne

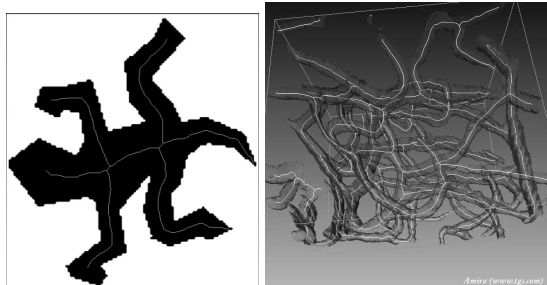
⁵ TGS Europe Bordeaux, France

ABSTRACT

Skeletons are compact representations that allow mathematical analysis of objects. A skeleton must be *homotopic*, *thin* and *medial* in relation to the object it represents. Numerous approaches already exist which focus on computational efficiency. However, when dealing with data too large to be loaded into the main memory of a personal computer, such approaches can no longer be used. We present in this article a skeletonization algorithm that processes the data locally (in sub-images) while preserving global properties (medial localization). Our privileged application is the study of the cerebral micro-vascularisation, and we show some results obtained on a mosaic of 3-D images acquired by confocal microscopy.

1. INTRODUCTION

Skeletonization gives a compact representation of shapes, even in the case of complex topology. In dimension 2, skeletons are connected center lines (figure 1 (a)). In dimension 3, skeletons can be center lines as in dimension 2 (figure 1 (b)) or center surfaces. They are commonly used in image



(a) 2D centerlines

(b) 3D centerlines

Fig. 1. Samples of center lines

analysis and pattern recognition, as they can synthetically describe shapes and mathematical properties of objects, for example length or surface area. Many algorithms have been proposed to compute skeletons [1], but all of them implicitly

suppose that the image can be processed at once in the computer memory. However, as image resolution keeps increasing, image size keeps growing, which leads to a huge increase of the amount of data to study. For example, vascular network analysis in medical imaging or oil network analysis in petrol imaging can lead to images of several gigabytes (see section 2). These kind of images cannot be loaded at once in a standard computer memory. We should therefore propose adapted methods to process such images. For many image processing tools, processing in sub-images does not involve much difficulty (for example algebraic operations, mathematical morphologic operators, filtering, etc). This is not the case for skeletonization, because we must ensure that the *global* properties of a skeleton (as being located at the center of the global object for example) are preserved by *local* operations. In the following section, we present our data, and the pre-processing we apply on them. Then we present the general skeleton computation algorithm followed by its adaptation for a block by block implementation. At last, we present the results on our images.

2. MICRO-VASCULAR DATA

Our application aims at analyzing the brain micro-vascular network morphology to help anatomists to have a better knowledge of micro-vascular anatomy (for example, measures on the network fractal dimension can give information on tissue [2]) and physiologists to better understand the mechanism of several image modalities depending on vascularity. Indeed, fMRI, may be biased by micro-vascularisation structures [3]. To better understand what exactly happens, a quantification of micro-vascular features, as for example vessel density, is needed.

To perform such an analysis, we study wide cortex areas with a confocal microscope. We then extract the vessel center lines and modelize vessels as a set of cylinders centered at the center lines points and with radii corresponding to the distance map value at these points.

Our data come from Duvernoy's collection [4]. A human

brain has been injected with Indian ink and then cut in thin sections to be observed with a traditional microscope. These sections can also be observed with a confocal microscope. The mean size of a confocal microscope image is about $600 \times 600 \times 300 \mu m$, which is much smaller than the original sections. To study wider areas, we build an *image mosaic*: the section is located on a table which can be translated with a micro-metric screw. Once an image is taken, we translate the section and take another image, and so on. Figure 2 shows such a mosaic. This method allows to ob-

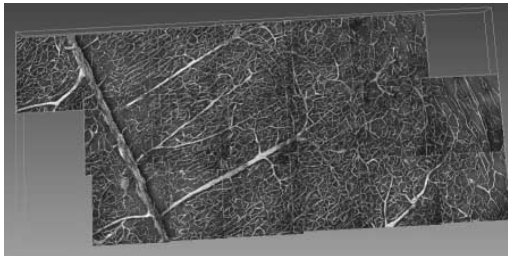


Fig. 2. image mosaic (18 images of 25 Mb each)

tain a mosaic with about a hundred images, representing several mm^3 of the brain. Each image is composed with about $512 \times 512 \times 100$ voxels (25Mb).

The acquisition protocol provides an overlap of about 50 voxels between the mosaic images. In addition the micro-metric screw may lead to imprecise relative image positions. We re-estimate the image positions by optimizing a similarity measure on the overlapping area. We then perform filtering and morphological operations on the mosaic before binarizing it by a simple user-defined threshold. At last, we compute a block by block chamfer map with anisotropic chamfer mask coefficients [5].

3. SKELETONIZATION

3.1. Distance Ordered Homotopic Thinning

A **skeleton** is a subset of the foreground object that verifies the following properties:

homotopy: the skeleton is topologically equivalent to the original image. It has the same number of connected components, holes and cavities as the original image.

thinness: the skeleton is one point wide except at junction points, where the connectivity can require several points.

medialness: the skeleton is centrally located within the foreground object.

Skeletons can be built in the continuous space using Voronoï diagrams. Discrete points are chosen on the object contour. The skeleton is the subgraph of the Voronoï diagram of these

points, which is entirely included within the object. The obtained skeleton is connected, topologically equivalent to the object, centered and thin. But this method rises difficulties when applied on the discrete space. Moreover the algorithm complexity and the computational time are crippling for huge images. On the other hand, we can be interested in discrete methods, generally fast and easy to use. The skeletonization process can be based on:

thinning: the skeleton is computed by iteratively peeling off the boundary of the object, layer-by-layer. The *deletable* points (simple points¹ that are also border points and non-end points²) are removed either sequentially [7] or in parallel [8, 9, 10, 11, 12], or with morphological operations [13]. These methods lead to a skeleton *homotopic* to the object, by construction, *thin* and geometrically representative (if the end-points have been correctly characterized), but not necessarily centered.

distance maps: the skeleton is defined as the locus of the local maxima of the distance map [14, 15, 16]. The principle of these methods is to calculate the distance map of the object, to find local maxima and to reconnect these maxima. The resulting skeleton is *centered* by construction, *thin*, depending on the local maxima threshold, but not necessarily homotopic, depending on the path reconstruction.

Hybrid methods have been recently introduced to take advantage of both of these approaches [17, 18]. These methods, called Distance Ordered Homotopic Thinning (DOHT), use a homotopic thinning, this means an iterative deletion of *simple points*, leading to a homotopic skeleton, but in the increasing distance map order leading to a centered skeleton. In order to better preserve rotation invariance, we add to this method a directional strategy inherited from the parallel algorithms. For each distance, we first consider border points, located on the east of the object, then the border points located on the bottom, then those located on the west, then on the south, and at last those located on the top of the object.

3.2. Block by block skeletonization

Here, we adapt the previous skeletonization algorithm to huge images, proceeding block by block. This adaptation is guided by the skeleton properties : homotopy, medialness and thinness.

¹a point is said *simple* if its deletion preserves the object topology (see [6] for simple points characterization).

²If all simple points are removed iteratively the result object is topologically equivalent to the original one, but far too simple: a connected component without hole nor cavity will be shrunk to a single point. Conditions of *end point* are defined for points located on the border of a line or surface to keep it a line or a surface.

3.2.1. Homotopy

Homotopy is a local property which can be reached by looking at the neighborhood of points to delete. Problems of homotopy may appear at the borders of blocks. Indeed, neighborhoods of points located at block borders are unknown. If we consider these neighborhoods as background points as well as foreground points, disconnection will appear at block junctions (see figure 3 (a)). To solve this problem, we

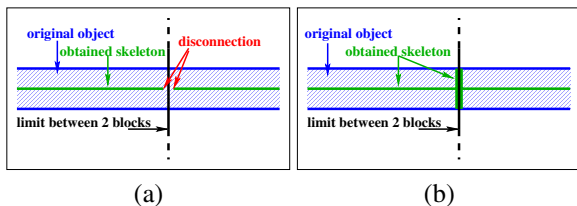


Fig. 3. disconnections between 2 blocks.

“freeze” points located at the border of a block, i.e. we consider as *deletable* only points whose whole neighborhood is included within the block. This condition guarantees a homotopic skeleton since only simple points will be deleted. We can then notice additional branches appearing. These elements correspond to low value of the distance map and will be deleted if we process the algorithm once again on the block border.

3.2.2. Medialness

As opposed to local property as homotopy, medialness is a regional property which is more difficult to ensure. Indeed, if we delete every points except the border ones in a block, the skeleton may be relocated. As shown in figure 4 (a), some points expected to be in the skeleton can be deleted. The connected component kept from the object becomes the “frozen” points of the border. The skeleton is “stucked”

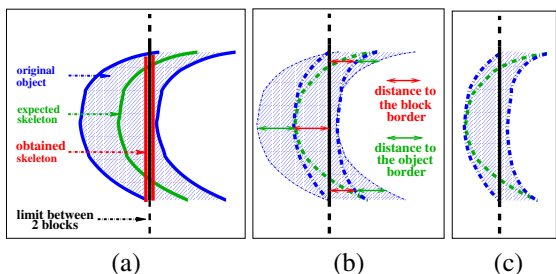


Fig. 4. skeleton relocation

to the border of the first thinned block and not located at the object center. To overcome this problem, we consider as *deletable* only points whose distance to the block border is larger than its distance to the object border (see figure 4 (b)). This means that a point can be deleted only if its associated maximal ball is entirely included within the block. We can notice that this condition also allows to “freeze” the

points located at the block border as wanted in the previous section. Figure 4 (c) shows the object component which is kept after this skeletonization phase. The expected skeleton is located within this component, but is not really centered. Anyway, if we apply another skeletonization phase on this area, the simple points deletion will be ordered by the distance map of the original object. The correct skeleton will be located on this distance map maxima, and the skeletonization algorithm will delete every point located around these maxima before reaching the expected skeleton points. This will lead to a medial skeleton.

3.2.3. Thinness

The two previous conditions lead to a *homotopic* and *medial* skeleton. But it may remain thick. Indeed, object areas located at the borders of blocks have not entirely been thinned. To obtain a thin skeleton, we re-apply the skeletonization algorithm with the same conditions, but on the area remained thick, that is to say the block borders areas. This is done by shifting image blocks. Figure 5 shows a 2D example of the

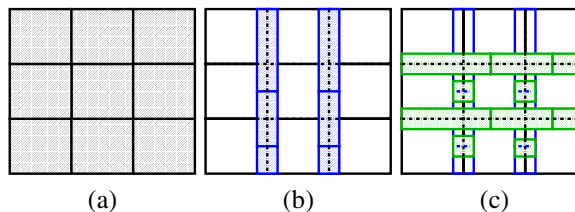


Fig. 5. skeleton thinness

different blocs:

first pass: we first divide the original image into several adjacent blocks (figure 5 (a)).

pass on x boundary: border block areas have not been entirely thinned. We cut up the image into stripes overlapping the previous boundaries (figure 5 (b)). The stripes are twice as wide as the largest radius found in the distance map to ensure that every point remaining in the stripes will be thinned.

pass on y and z boundary: we process the same way on y boundaries (figure 5 (c)), and then on z boundaries.

The obtained skeleton is *homotopic* because the neighborhood of any deleted points was entirely known, *medial* because the thinning is performed using the distance map order and we take care of not relocating the skeleton, and *thin* because we process several passes on the block boundaries in a way that every part of the object is located at least once at the center of a block.

4. RESULTS

Figure 6 shows the center lines obtained when we apply the skeletonization algorithm presented here on the image mosaic of figure 2. This result allows to see vessel network

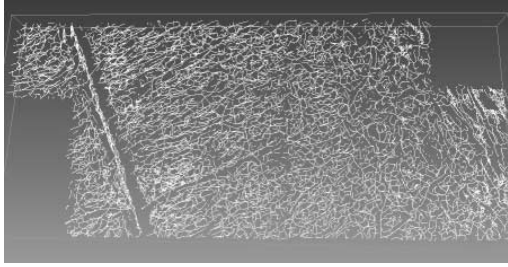


Fig. 6. vessel center lines

connections. Moreover, to extract morphometric features of the vessel network, we add diameters to the center lines. To do so, we model a vessel as a set of cylinders centered in the center lines points and which radii correspond to the distance map value at these points (see figure 7).

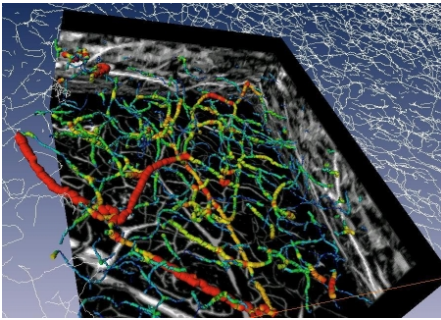


Fig. 7. center lines added with vessel diameters

5. CONCLUSION

We have presented a pattern recognition tool adapted to huge images through a particular application (study of the brain micro-vascularisation network). Indeed as huge images cannot be loaded at once in a standard computer memory, they need adapted algorithms. Our block by block skeletonization method preserves the global properties as well as the local properties of a skeleton by avoiding border effects. Indeed in a first pass, we process 3D blocks without overlapping, then, we process sub-images covering boundaries. The size of these sub-images depends on the size of the object to thin. Doing so, inner block areas are processed only once. The algorithm overhead due to the block by block process appears only on boundaries. We have also shown that for our application, our tools allow to extract quantitative information precious for neuro-anatomists and neuro-physiologists to describe the brain micro-vascular network.

6. ACKNOWLEDGMENTS

This work has been done in MicroVisu3D project, and was partially founded by a “Fond de la Recherche Technologique” grant of the French government. The presented tools have been transferred to the user team (INSERM U455, IMFT) through an integration into the Amira software [19].

7. REFERENCES

- [1] G. Borgefors, I. Nystrom, and G. Sanniti di Baja, “Computing skeletons in three dimensions,” *PR*, vol. 32, no. 7, pp. 1225–1236, July 1999.
- [2] O. Craciunescu, S.K Das, and Dewhirst M. W., “Three-dimensional microvascular networks fractal structure: potential for tissue characterization?,” in *Advances in Heat and Mass Transfer in Biotechnology*, 1999.
- [3] R. Turner, “How much cortex can a vein drain? downstream dilution of activation-related cerebral blood oxygenation changes,” *NeurImage*, 2001.
- [4] Henri M Duvernoy, “Cortical blood vessels of the human brain,” *Brain Research Bulletin*, 1981.
- [5] C. Fouard and G. Malandain, “Systematized calculation of optimal coefficients of 3-D chamfer norms,” in *Proceedings of DGCI*. 2003, LNCS, Springer.
- [6] G. Bertrand and G. Malandain, “A new characterization of three-dimensional simple points,” *PRL*, vol. 15, no. 2, pp. 169–175, February 1994.
- [7] K. Palágyi, E. Sorantin, E. Balogh, and A. Kuba, “A sequential 3D thinning algorithm and its medical applications,” in *IPMI 2001*, june 2001, vol. 2082, pp. 409–415.
- [8] Y.F. Tsao and K.S. Fu, “A parallel thinning algorithm for 3-d pictures,” *CGIP*, vol. 17, pp. 315–331, 1981.
- [9] W. Gong and G. Bertrand, “A simple parallel 3D thinning algorithm,” in *ICPR*, 1990, pp. 188–190.
- [10] K. Palágyi and Kuba A., “Directional 3D thinning using 8 subiterations,” in *DGCI*, 1999, vol. 1568, pp. 325–336.
- [11] C.M. Ma and S.Y. Wan, “Parallel thinning algorithms on 3D (18, 6) binary images,” *Computer Vision and Image Understanding*, vol. 80, pp. 364–378, 2000.
- [12] C. Lohou and G. Bertrand, “A new 3D 6-subiteration thinning algorithm based on p-simple points,” in *DGCI*, 2002, vol. 2301, pp. 102–113.
- [13] P.P. Jonker, “Morphological operations on 3D and 4D images: From shape primitive detection to skeletonization,” in *DGCI*, 2000, vol. 1953, pp. 371–391.
- [14] L. Calabi and W.E. Harnett, “Shape recognition, prairie fires, convex deficiencies and skeletons,” *A.M.M.*, pp. 335–342, 1968.
- [15] G. Malandain and S. Fernández-Vidal, “Euclidean skeletons,” *IVC*, vol. 16, no. 5, pp. 317–327, Apr. 1998.
- [16] Y. Zhou, A. Kaufman, and Toga A. W., “3D skeleton and centerline generation based on an approximate minimum distance field,” *IJVC*, vol. 14, no. 7, pp. 303–314, 1998.
- [17] T. Saito and J.i. Toriwaki, “A sequential thinning algorithm for three dimensional digital pictures using the euclidean distance transformation,” in *Proceedings of SCIA*. IAPR, 1995, pp. 507–516.
- [18] C.J. Pudney, “Distance-ordered homotopic thinning: A skeletonization algorithm for 3D digital images,” *Proceedings of CVIU*, vol. 72, no. 3, pp. 404–413, 1998.
- [19] Zuse Institute Berlin (ZIB) and Indeed - Visual Concepts, Berlin, *Amira 3.0 – Programmer’s Guide*, Nov. 2002, <http://amira.zib.de>.