Manuscript accepted in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010.

Automatic correction of Ma and Sonka's thinning algorithm using P-simple points

Christophe LOHOU[1] and Julien DEHOS[2]

[1] Laboratoire d'Algorithmique et Image de Clermont, Université d'Auvergne, France
[2]Laboratoire d'Informatique du Littoral, Université du Littoral Côte d'Opale, France

Abstract

Ma and Sonka proposed a fully parallel 3D thinning algorithm which does not always preserve topology. We propose an algorithm based on P-simple points which automatically corrects Ma and Sonka's Algorithm. As far as we know, our algorithm is the only fully parallel curve thinning algorithm which preserves topology.

Keywords: 3D thinning algorithm, curve skeleton, digital topology, topology preservation.

1 Introduction

In the image processing field, thinning is a technique which extracts skeletons from objects. Such a representation has the advantage of being compact while conserving important features of objects like topology and geometry. Thinning has many applications such as compression, analysis and recognition [1].

Thinning can be achieved by different approaches. For instance, in the field of digital topology, we find thinning algorithms based on parallel deletion of voxels [2–11], or of surfels [12], or guided by the notion of distance [13–15]. In the field of algorithmical geometry, we find thinning algorithms based on Delaunay triangulation and Voronoï diagram notions [16]. In the field of mathematical morphology, we find thinning algorithms based on erosion or skeleton by influence zones notions [17]. In the field of algebraical topology, we find works based on cellular complexes [18], homotopy chains [19], or critical kernels [20]. In this paper, we only deal with thinning algorithms for 3D binary images by parallel deletions of voxels, in the case of digital topology.

In a binary image, objects are represented by black points. The complement is represented by white points. Thinning consists in deleting black points i.e. changing black points to white. It is generally done in successive iterations of deletions. Removing a point should preserve certain properties such as topology. This leads to the essential notion of a simple point: a simple point is a point which can be removed without changing topology [21]. We may notice that a $3 \times 3 \times 3$ neighborhood of a point x is necessary and sufficient to decide whether x is simple or not. Since we consider parallel algorithms, we face a major problem: the simultaneous removal of simple points may change the topology of an object. This is the case, for example, of an object constituted of two adjacent voxels.

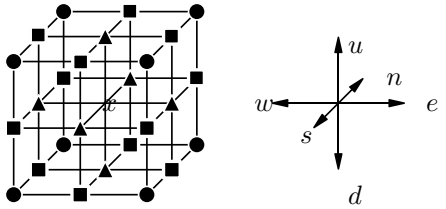
To solve this problem, three different solutions may be considered:

- giving a set of $3 \times 3 \times 3$ masks or templates (a $3 \times 3 \times 3$ neighborhood centered around a considered point) to determine whether a point is deletable. In other words, all points matching at least one of the templates are removed. Note that the templates should be conceived in such a way that the algorithm preserves topology. This leads to algorithms performing deletion iterations. Each iteration is divided into several subiterations based, for example, on directions [4–6, 22–24] or on subgrids [5, 25, 26];

- accessing an extended neighborhood (i.e. a neighborhood which strictly includes the $3 \times 3 \times 3$ neighborhood) to obtain more information. Such a strategy may lead to fully parallel thinning algorithms [2, 3] (which use specific directions) or to symmetrical thinning algorithms [7] (which are isometrically invariant);

- using another class of points that ensures that the deletion of all points of the class does not affect topology. This is what P-simple points accomplish [27]. In fact, this notion is very general and leads to different thinning schemes according to different strategies [28], for example, directional [8, 9, 12, 27, 29] or symmetrical [10] (in this case, since the deleting process is symmetrical, skeleton centering is automatic therefore the notion of distance is not required).

In [3], Ma and Sonka propose a fully parallel thinning algorithm based on 38 templates. In [30], Lohou gives an object whose topology is not preserved by Ma and Sonka's algorithm. In [31], Chaturvedi and Lee found that Ma and Sonka's algorithm disconnects some objects. In [32], Wang and Basu propose a correction of Ma and Sonka's algorithm. Their correction consists in changing 12 templates of the original algorithm by



(a) (b)

Figure 1: (a) The 6-, 18-, and 26-neighbors of x are respectively represented by black triangles, black squares, and black discs. (b) The six major directions: up (u), down (d), north (n), south (s), west (w) and east (e).

36 new templates. However, this correction makes the algorithm more difficult to implement and to analyse. Furthermore, the topology of the object given in [30] is still not preserved by the algorithm.

Topology preservation of the previous parallel thinning algorithms is not automatic and has to be proven, which is difficult. On the contrary, parallel thinning algorithms removing P-simple points have the property of preserving topology. In this paper, we propose a correction of a templates-based parallel thinning algorithm using P-simple points, which automatically ensures topology preservation.

This paper is organized as follows. Section 2 gives some basic notions of digital topology and presents P-simple points. Section 3 recalls Ma and Sonka's thinning algorithm. Section 4 shows how to use P-simple points in a thinning algorithm. In Section 5, we propose such an algorithm correcting Ma and Sonka's algorithm. Section 6 gives some results and Section 7 concludes.

2 Basic notions of digital topology

2.1 Neighborhoods, connected components and holes

A point $x \in \mathbb{Z}^3$ is defined by (x_1, x_2, x_3) with $x_i \in \mathbb{Z}$. We consider the three neighborhoods: $N_{26}(x) = \{x' \in \mathbb{Z}^3 : Max[|x_1 - x'_1|, |x_2 - x'_2|, |x_3 - x'_3|] \leq 1\}$, $N_6(x) = \{x' \in \mathbb{Z}^3 : |x_1 - x'_1| + |x_2 - x'_2| + |x_3 - x'_3| \leq 1\}$, and $N_{18}(x) = \{x' \in \mathbb{Z}^3 : |x_1 - x'_1| + |x_2 - x'_2| + |x_3 - x'_3| \leq 2\} \cap N_{26}(x)$. We define $N^*_n(x) = N_n(x) \setminus \{x\}$. We call respectively 6-, 18-, 26-*neighbors* of x the points of $N^*_6(x)$, $N^*_{18}(x) \setminus N^*_6(x)$, $N^*_{26}(x) \setminus N^*_{18}(x)$, such points are represented in Fig. 1 (a). The 6-neighbors of x determine six major directions (Fig. 1 (b)). The points belonging to X (resp. $\overline{X}$) are called *black points* (resp. *white points*).

Two points x and y are said to be *n-adjacent* if $y \in N^*_n(x)$ ($n = 6, 18, 26$). An *n-path* is a sequence of points $x_0, \ldots, x_k$, with x_i n-adjacent to x_{i-1} for any $1 \leq i \leq k$. If $x_0 = x_k$, the path is *closed*. Let $X \subseteq \mathbb{Z}^3$. Two points $x \in X$ and $y \in X$ are *n-connected in* X if they can be linked by an n-path included in X. The equivalence classes relative to this relation are the *n-connected components of* X. In order to have a correspondence between the topology of X and that of $\overline{X}$, we have to consider two different kinds of adjacency for X and for $\overline{X}$ [21]: if we use an n-adjacency for X, we have to use another $\overline{n}$-adjacency for $\overline{X}$. In this paper, we only consider $(n, \overline{n}) = (26, 6)$.

Let $X \subset \mathbb{Z}^3$. A *hole* (sometimes called a *tunnel* [21,33]) in X is detected when there is a closed path in X which cannot be deformed in X into a single point (see [21, 34, 35] for further details).

2.2 Simple points and topological numbers

Let $X \subseteq \mathbb{Z}^3$. A point $x \in X$ is said to be *simple* if its removal does not "change the topology" of the image, in the sense that there is a one-to-one correspondence between the components, the holes of X (resp. $\overline{X}$) and the components, the holes of $X \setminus \{x\}$ (resp. $\overline{X} \cup \{x\}$), see [34] for a precise definition.

The set composed of all n-connected components of X which are n-adjacent to a point x is denoted by $\mathcal{C}^x_n(X)$. Let $\#X$ denote the number of elements which belong to X. The *topological numbers* relative to X and x are the two numbers [35]: $T_6(x, X) = \#\mathcal{C}^x_6[N^*_{18}(x) \cap X]$ and $T_{26}(x, X) = \#\mathcal{C}^x_{26}[N^*_{26}(x) \cap X]$. These numbers lead to a very concise characterization of 3D simple points [36]: $x \in X$ is simple for X if and only if $T_{26}(x, X) = 1$ and $T_6(x, \overline{X}) = 1$.

For example, in Fig. 2 (a), the points y and z both belong to $N^*_{18}(x) \cap \overline{X}$ but there is no 6-path of white points included in $N^*_{18}(x) \cap \overline{X}$ which joins them. Thus, $T_{26}(x, X) = 1$ and $T_6(x, \overline{X}) = 2$ therefore x is not simple. In Fig. 2 (b), there is a single black 26-connected component in $N^*_{26}(x) \cap X$ therefore $T_{26}(x, X) = 1$; w does not belong to $N^*_{18}(x) \cap \overline{X}$ therefore $T_6(x, \overline{X}) = 1$; thus x is simple.

2.3 P-simple points

In the following, we consider a subset X of $\mathbb{Z}^3$, a subset P of X, and a point x of P. The point x is *P-simple* for X if for each subset S of $P \setminus \{x\}$, x is simple for $X \setminus S$. A subset of X is *P-simple* if it is composed of solely P-simple points. We have the property that any algorithm removing only P-simple subsets is guaranteed to keep the topology unchanged [27].

We give a local characterization of a P-simple point [37]. Let R denote the set $X \setminus P$. The point x is P-simple for X if and only if:

$$\begin{cases} T_{26}(x, R) = 1; \\ T_6(x, \overline{X}) = 1; \\ \forall y \in N^*_{26}(x) \cap P, \exists z \in R \\ \quad \text{such that } z \text{ is 26-adjacent to } x \text{ and to } y; \\ \forall y \in N^*_6(x) \cap P, \exists z \in \overline{X} \text{ and } \exists t \in \overline{X} \\ \quad \text{such that } \{x, y, z, t\} \text{ is a unit square.} \end{cases}$$

For example, in Fig. 2 (c), $N^*_{26}(x) \cap P = \{y_1, y_2\}$, there is no point of R which is 26-adjacent to both x and y_1 therefore the third condition is not verified and x is not P-simple. In Fig. 2 (d), all conditions are verified for x therefore x is P-simple.

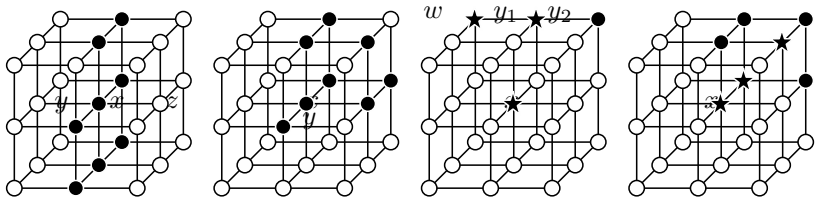


Figure 2: In (a) and (b), points belonging to X and $\overline{X}$ are respectively represented by black discs and white discs. In (c) and (d), points belonging to R, P and $\overline{X}$ are respectively represented by black discs, black stars and white discs.

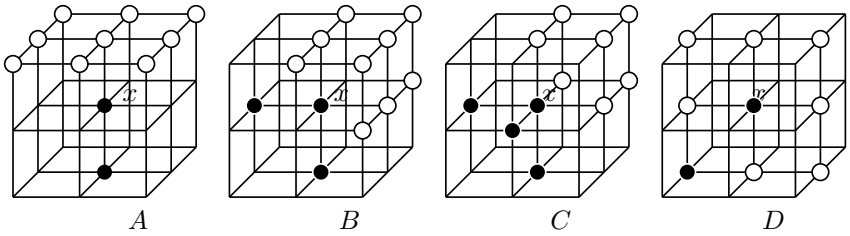


Figure 3: The four basic template cores of MS96. Each unmarked point is either a black point or a white point. For D, x must be simple.

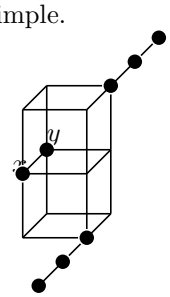


Figure 4: MS96 changes the topology of this object. For convenience sake, points of $\overline{X}$ are not represented.

3 Ma and Sonka's algorithm

Ma and Sonka's algorithm (denoted by MS96 in the following) is a fully parallel curve thinning algorithm, based on templates requiring access to up to 50 points to decide whether a point is deletable or not [3] (see below). The templates can be obtained from four given subsets of template cores (classes A, B, C and D) and additional directional conditions. The template cores are depicted in Fig. 3. The deleting templates are the set of all reflections and all 90° rotations around any one of the three major axes of the four template cores. Thus, a point satisfies a template if it matches at least one of the deleting templates T_i and verifies all the conditions (see Fig. 1 (b) for directions):

1. if T_i is isomorphic to any template core A, B or C:
   (a) $s(x)$ is black in $T_i \Rightarrow s(s(x))$ is black;
   (b) $w(x)$ is black in $T_i \Rightarrow w(w(x))$ is black;
   (c) $d(x)$ is black in $T_i \Rightarrow d(d(x))$ is black.
2. if T_i is isomorphic to the template core D:
   (a) x is a simple point;
   (b) $y = s(u(x))$ is black in $T_i \Rightarrow$ at least one of $s(s(x))$, $s(y)$, $s(u(y))$, $u(y)$ and $u(u(x))$ is black;
   (c) $y = s(d(x))$ is black in $T_i \Rightarrow$ at least one of $s(s(x))$, $s(y)$, $s(d(y))$, $d(y)$ and $d(d(x))$ is black;
   (d) $y = s(e(x))$ is black in $T_i \Rightarrow$ at least one of $s(s(x))$, $s(y)$, $s(e(y))$, $e(y)$ and $e(e(x))$ is black;
   (e) $y = s(w(x))$ is black in $T_i \Rightarrow$ at least one of $s(s(x))$, $s(y)$, $s(w(y))$, $w(y)$ and $w(w(x))$ is black;
   (f) $y = w(u(x))$ is black in $T_i \Rightarrow$ at least one of $w(w(x))$, $w(y)$, $w(u(y))$, $u(y)$ and $u(u(x))$ is black;
   (g) $y = w(d(x))$ is black in $T_i \Rightarrow$ at least one of $w(w(x))$, $w(y)$, $w(d(y))$, $d(y)$ and $d(d(x))$ is black.

Now, we give additional definitions and describe MS96. Let x be a black point of a 3D image, then

- x is a *line-end point* if x is 26-adjacent to exactly a single point of the object;
- x is a *near-line-end point* if x is 26-adjacent to exactly two points of the object which are:
  - either $s(x)$ and $e(x)$, or $s(x)$ and $u(x)$ but not both;
  - either $n(x)$ and $w(x)$, or $u(x)$ and $w(x)$ but not both;
  - or $n(x)$ and $d(x)$, or $e(x)$ and $d(x)$ but not both.
- x is called a *tail point* if it is either a line-end point or a near-line-end point; otherwise it is

called a *nontail point*.

MS96

1: **repeat**
2: parallel delete every nontail object point which satisfies at least one deleting template of classes A, B, C or D
3: **until** no points are deleted

It has been showed that MS96 does not always preserve topology. Indeed, let us consider the object studied in [30] and depicted in Fig. 4. Applying MS96 to this object only removes x and y (they match templates of class D), which disconnects the object i.e. changes its topology.

As mentioned before, in [32], Wang and Basu proposed a correction of MS96 by changing several templates. However, applying their algorithm (denoted by WB07 in the following) to the previous object also removes x and y (x satisfies template $D11.3$ and y satisfies template $D12.2$, notations given in [32]) therefore changes the topology too.

4 Thinning algorithms based on P-simple points

Now, let us consider an algorithm which removes P-simple points in parallel, in successive iterations. A skeleton is obtained when no more points can be removed. Since removing P-simple sets is guaranteed to keep the topology unchanged [27], this algorithm is a topology-preserving thinning algorithm. Such an algorithm is given below:

THIN(X: object, C: condition) $\rightarrow X$: object

1: **repeat**
2: {parallel labelling of points which belong to P}
3: $P \leftarrow \emptyset$
4: **for** each point x in X, in parallel, **do**
5: **if** $C(x, X)$ **then**
6: put x in P
7: **end if**
8: **end for**
9: {parallel deletion of P-simple points}
10: $X' \leftarrow X$
11: **for** each point x in P, in parallel, **do**
12: **if** x is P-simple for X' **then**
13: delete x in X
14: **end if**
15: **end for**
16: **until** no points are deleted

where C denotes the condition a point of an object should satisfy to belong to P (intuitively, the set of candidate points to deletion). Each iteration is composed of two passes. Thanks to P-simple points, the algorithm can examine points in parallel. Furthermore, using the local characterization of P-simple points given in Section 2.3, the algorithm can consider limited neighborhoods. It is thus well-suited for parallel machines.

The major difficulty of such an algorithm is to provide an appropriate condition C to determine P, the set of candidate points to deletion. On the one hand, if P contains a few points, only a few points may be P-simple therefore deleted. On the other hand, if P contains a lot of points, each point of P has very little chance to be P-simple therefore deleted.

P-simple points have already been used by Lohou and Bertrand to conceive thinning algorithms. In [8], they take Palagyi and Kuba's 12-subiteration surface thinning algorithm (PAKU12) [6] to derive a 12-subiteration surface thinning algorithm based on P-simple points. Their algorithm deletes at least the same points as PAKU12 therefore their algorithm is more powerful than PAKU12 (in the sense that, for a given object, it deletes at least all points deleted by PAKU12 during the first deleting iteration). Moreover, this automatically proves that PAKU12 preserves topology well. In [9], Lohou and Bertrand proposed a 6-subiteration curve thinning algorithm based on Palagyi and Kuba's 6-subiteration curve thinning algorithm [4]. Here, the aim was to show that their method can be applied to different kinds of algorithms, for instance, a thinning algorithm which preserves curves. Finally, in [10], they proposed a curve thinning algorithm and a surface thinning algorithm by using two different conditions of end points. This study did not aim to enhance a given algorithm but to propose isometric-invariant thinning algorithms, also classified as *symmetrical*.

In the current study, the objective is completely different and is new in the literature: using P-simple points to automatically correct a failing algorithm. Thus, the obtained algorithm is ensured to preserve topology. No deep examination of templates is required.

5 A correction of Ma and Sonka's algorithm

To correct Ma and Sonka's algorithm, we propose to parallely remove P-simple points in successive iterations where P is the set of points which would be deleted by Ma and Sonka's algorithm. This algorithm (denoted by LD_MS96 in the following) can be described as:

LD_MS96(X: object) $\rightarrow X$: object

1: $X \leftarrow$ THIN(X,C_MS96)

where:

C_MS96(x: point, X: object) $\rightarrow b$: boolean

1: **if** x is a nontail point which satisfies at least one template of classes A, B, C or D in X (Fig. 3) **then**
2: $b \leftarrow$ true
3: **else**
4: $b \leftarrow$ false

5: **end if**

Thus, during an iteration, LD_MS96 deletes the points which are removed by MS96 such that the parallel deletion of any subset of these points does not affect topology. Any point deleted by LD_MS96 is also deleted by MS96. Since the characterization of P-simple points is linear [27], any iteration is bounded by $O(N)$ where N is the size of the image (number of points).

By the definition of P-simple points, there is no ambiguity: once P is defined, any algorithm which deletes in parallel P-simple points is ensured to preserve topology. The main interest of our present work is to precisely define P as the set of points deleted by Ma and Sonka's algorithm. Thus, since LD_MS96 deletes only P-simple points where P is defined as mentioned previously, it automatically preserves topology. Neither additional proof nor deep template examination is required.

Let us consider again the object depicted in Fig. 4. This time, LD_MS96 states that $P = \{x, y\}$ therefore neither x nor y is P-simple (the first condition is not verified for x and y). Therefore, no points are deleted and the topology is well preserved.

We may notice that LD_MS96 brings us many questions. For example, does the use of P-simple points make the simplicity condition of the class D templates unnecessary? If we change the place of the tail point evaluation, can we obtain a more efficient or powerful algorithm? What can we expect of using other tail point conditions? All these questions are beyond the scope of this paper and are part of our current work [38].

6 Results and analysis

We compared MS96, WB07 and LD_MS96 using various objects. Some of these objects and their skeletons (obtained with MS96, WB07 and LD_MS96) are depicted Fig. 5. Note that the LETTERD object is used by Ma and Sonka in [3] to illustrate MS96. We applied LD_MS96 (and WB07) to this object and found exactly the same results as with MS96. Among our 7 objects, MS96 disconnects all the objects but LETTERD. In the same way, WB07 disconnects three objects (BULL , KNOT and VERTEBRA). On the contrary, LD_MS96 preserves the topology of all the 7 objects, which agrees with the topology preservation property of any thinning algorithms based on P-simple point deletion and illustrates the relevance of our method. Note that LD_MS96 produces thin and centered skeletons and therefore is a pertinent thinning algorithm.

Some statistics concerning our results are given in Table 1. We found that during the first iteration, MS96 and LD_MS96 delete the same number of points. In fact, by its conception, LD_MS96 cannot delete, during the first iteration, more points of a given object than MS96. Indeed, P is the set of points deleted by MS96 and the set of P-simple points (which is deleted by LD_MS96) is included into P. However, we can prove nothing else from this because even if both algorithms are applied to the same initial object, an iteration may come where LD_MS96 removes fewer points than MS96. Thus, the next iteration happens on different objects which makes final results (deleted points, iterations) difficult to predict. For example, the number of iterations performed by MS96 can be less (BULL and VERTEBRA), equal (DOLPHIN, KNOT, LETTERD and VENUS) or greater (3L) than the one performed by LD_MS96.

Since the templates of WB07 are more restrictive than the templates of MS96, we verify that, during the first iteration, the number of points deleted by WB07 is less or equal than the number of points deleted by MS96. On the other hand, the number of iterations performed by WB07 is greater or equal to the number of iterations performed by MS96.

7 Conclusion

Using P-simple points, a thinning algorithm is guaranteed to preserve topology. Ma and Sonka proposed a templates-based curve thinning algorithm which fails to preserve topology. In this paper, we have proposed an automatic correction of this algorithm, using P-simple points. We tested and compared our algorithm with the original algorithm of Ma and Sonka and with the correction proposed by Wang and Basu. Our algorithm also extracts curve skeletons therefore it can be used for the same applications as the original algorithm. Since our algorithm preserves topology, it can make these applications more reliable. As far as we know, our algorithm is the only fully parallel curve thinning algorithm which preserves topology.

Symmetrical thinning algorithms using distance yield well centered but thick skeletons. On the contrary, other approaches results in thin but non-centered skeletons. The initial idea of Ma and Sonka is to use directions without performing subiteration to give a proof of topology preservation. In fact, their algorithm is a good compromise to obtain well centered as well as thin curve skeletons therefore was worth correction.

Acknowledgments

We are grateful to Professor Gilles BERTRAND (ESIEE, Paris, France) for our numerous discussions about P-simple points applied to thinning algorithms, to Romain JACQUINOT (INSA Lyon, France) for rendering the images of Fig. 5 and to Susan ARBON (ISIMA, Clermont, France) for help in preparing this paper.

References

[1] V. Sauret, K. Goatman, J. Fleming, and A. Bailey, "Semi-automated tabulation of the 3D topology and morphology of branching networks using CT: application to the airway tree," *Physics in*

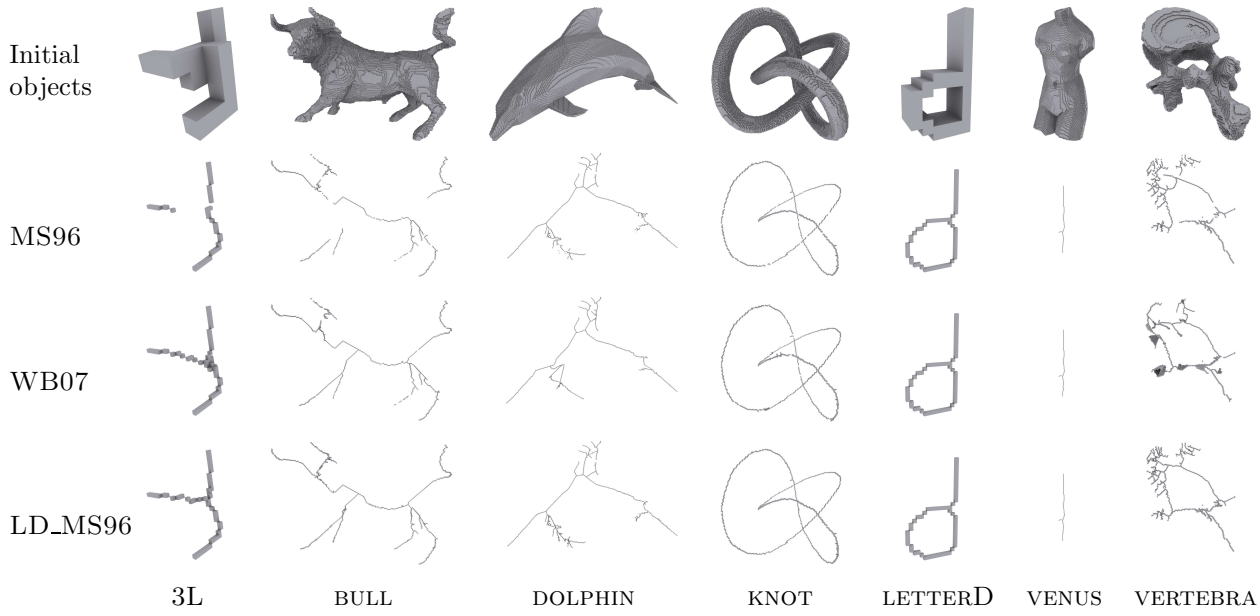


Figure 5: Some of the objects used for our tests and the obtained skeletons.

Object	Initial black points	Deleted points during the first iteration		Deleted points			Iterations		
		MS96/LD_MS96	WB07	MS96	WB07	LD_MS96	MS96	WB07	LD_MS96
3L	2 050	1 131	1 131	2 009	1 996	2 000	11	15	10
BULL	424 346	24 231	24 230	423 821	423 739	423 645	70	97	82
DOLPHIN	1 202 772	43 699	43 696	1 201 818	1 201 946	1 201 758	102	129	102
KNOT	231 239	26 114	26 114	230 493	230 456	230 443	20	22	20
LETTERD	2 079	1 281	1 281	2 020	2 020	2 020	4	4	4
VENUS	973 861	32 667	32 667	973 727	973 727	973 725	58	58	58
VERTEBRA	180 161	19 036	19 036	179 513	178 133	179 423	56	115	61

Table 1: Comparison between MS96, WB07 and LD_MS96 applied to the 7 objects depicted in Fig. 5.

Medicine and Biology, vol. 44, no. 7, pp. 1625–1638, 1999.

[2] C. Ma, "A 3D fully parallel thinning algorithm for generating medial faces," *Pattern Recognition Letters*, vol. 16, pp. 83–87, 1995.

[3] C. Ma and M. Sonka, "A fully parallel 3D thinning algorithm and its applications," *Computer Vision and Image Understanding*, vol. 64, no. 3, pp. 420–433, 1996.

[4] K. Palágyi and A. Kuba, "A 3D 6-subiteration thinning algorithm for extracting medial lines," *Pattern Recognition Letters*, vol. 19, no. 7, pp. 613–627, 1998.

[5] ——, "A hybrid thinning algorithm for 3D medical images," *Journal of Computing and Information Technology*, vol. 6, no. 2, pp. 149–164, 1998.

[6] ——, "A parallel 3D 12-subiteration thinning algorithm," *Graphical Models and Image Processing*, vol. 61, pp. 199–221, 1999.

[7] A. Manzanera, T. Bernard, F. Prêteux, and B. Longuet, "A unified mathematical framework for a compact and fully parallel n-D skeletonization procedure," in *Conference on Vision Geometry VIII*, vol. 3811, 1999, pp. 57–68.

[8] C. Lohou and G. Bertrand, "A 3D 12-subiteration thinning algorithm based on P-simple points," *Discrete Applied Mathematics*, vol. 139, pp. 171–195, 2004.

[9] ——, "A 3D 6-subiteration curve thinning algorithm based on P-simple points," *Discrete Applied Mathematics*, vol. 151, pp. 198–228, 2005.

[10] ——, "Two symmetrical thinning algorithms for 3D binary images, based on P-simple points," *Pattern Recognition*, vol. 40, pp. 2301–2314, 2007.

[11] T. Wang and A. Basu, "A note on 'a fully parallel 3D thinning algorithm and its applications'," *Pattern Recognition Letters*, vol. 28, no. 4, pp. 501–506, 2007.

[12] J. Burguet and R. Malgouyres, "Strong thinning and polyhedric approximation of the surface of a voxel object," *Discrete Applied Mathematics*, vol. 125, no. 1, pp. 93–114, 2003.

[13] C. Pudney, "Distance-ordered homotopic thinning: A skeletonization algorithm for 3D digital images," *Computer Vision and Image Understanding*, vol. 72, no. 3, pp. 404–413, 1998.

[14] G. Borgefors, I. Nyström, and G. S. D. Baja, "Computing skeletons in three dimensions," *Pattern Recognition*, vol. 32, pp. 1225–1236, 1999.

[15] S. Svensson, I. Nyström, and G. S. di Baja, "Curve skeletonization of surface-like objects in 3D images guided by voxel classification," *Pattern Recognition Letters*, vol. 23, pp. 1419–1426, 2002.

[16] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry - Algorithms and Applications*. Springer Verlag, 2008.

[17] J. Serra, *Image Analysis and mathematical morphology*. Academic Press, 1982.

[18] V. Kovalevsky, "Finite topology as applied to image analysis," *Computer Vision, Graphics and Image Processing*, vol. 46, pp. 141–161, 1989.

[19] P. R. Rocío González-Díaz, "On the cohomology of 3D digital images," *Discrete Applied Mathematics*, vol. 147, pp. 245–263, 2005.

[20] G. Bertrand and M. Couprie, "Two-dimensional thinning algorithms based on critical kernels," *Journal of Mathematical Imaging and Vision*, vol. 31, no. 1, pp. 35–56, 2008.

[21] T. Kong and A. Rosenfeld, "Digital topology: Introduction and survey," *Computer Vision, Graphics and Image Processing*, vol. 48, pp. 357–393, 1989.

[22] Y. Tsao and K. Fu, "A parallel thinning algorithm for 3D pictures," *Computer Graphics and Image Processing*, vol. 17, pp. 315–331, 1981.

[23] W. Gong and G. Bertrand, "A simple parallel 3D thinning algorithm," in *International Conference on Pattern Recognition*, 1990, pp. 188–190.

[24] F. Rolland, J.-M. Chassery, and A. Montanvert, "3D medial surfaces and 3D skeletons," in *International Workshop on Visual Form*, 1991, pp. 443–450.

[25] G. Bertrand and Z. Aktouf, "A three-dimensional thinning algorithm using subfields," in *Conference on Vision Geometry III*, vol. 2356, 1994, pp. 113–124.

[26] C. Ma, S. Wan, and J. Lee, "Three-dimensional topology preserving reduction on the 4-subfields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, 2002.

[27] G. Bertrand, "On P-simple points," *Comptes-rendus de l'Académie des Sciences*, vol. 321, no. 8, pp. 1077–1084, 1995.

[28] ——, "P-simple points: A solution for parallel thinning," in *International Conference on Discrete Geometry for Computer Imagery*, 1995, pp. 233–242.

[29] R. Malgouyres and S. Fourey, "Strong surfaces, surface skeletons and image superposition," in *Conference on Vision Geometry VII*, vol. 3454, 1998, pp. 16–27.

[30] C. Lohou, "Contribution à l'analyse topologique des images : étude d'algorithmes de squelettisation pour images 2D et 3D selon une approche topologie digitale ou topologie discrète," Ph.D. dissertation, Université de Marne-la-Vallée, France, 2001.

[31] A. Chaturvedi and Z. Lee, "Three-dimensional segmentation and skeletonization to build an airway tree data structure for small animals," *Physics in Medicine and Biology*, vol. 50, no. 7, pp. 1405–1419, 2005.

[32] T. Wang and A. Basu, "A note on ´ A fully parallel 3D thinning algorithm and its applications´ ," *Pattern Recognition Letters*, vol. 28, pp. 501–506, 2007.

[33] C. Ma, "On topology preservation in 3D thinning," *Computer Vision, Graphics, and Image Processing: Image Understanding*, vol. 59, no. 3, pp. 328–339, 1994.

[34] T. Kong, "A digital fundamental group," *Computer and Graphics*, vol. 13, no. 2, pp. 159–166, 1989.

[35] G. Bertrand, "Simple points, topological numbers and geodesic neighborhoods in cubic grids," *Pattern Recognition Letters*, vol. 15, pp. 1003–1011, 1994.

[36] G. Bertrand and G. Malandain, "A new characterization of three-dimensional simple points," *Pattern Recognition Letters*, vol. 15, pp. 169–175, 1994.

[37] G. Bertrand, "Sufficient conditions for 3D parallel thinning algorithms," in *Conference on Vision Geometry IV*, vol. 2573, 1995, pp. 52–60.

[38] J. Dehos and C. Lohou, "Analysis of algorithms derived from Ma and Sonka's thinning algorithm," in preparation.