Isthmus-Based 6-Directional Parallel Thinning Algorithms

Benjamin Raynal and Michel Couprie

Université Paris-Est
Laboratoire d'Informatique Gaspard Monge, Equipe A3SI

Abstract. Skeletons are widely used in pattern recognition and image analysis. A way to obtain skeletons is the thinning approach, consisting in iteratively removing points from the object without changing the topology. In order to preserve geometric information, it is usual to preserve curve end points (for curve skeletons) or surface end points (for surface skeletons). In this paper we propose a new fast directional parallel thinning scheme, preserving isthmuses (a generalization of curve/surface interior points), and providing skeletons with low amount of noise. We also prove the topology preservation of our approach.

Keywords: Thinning Algorithm, Surface Skeleton, Curvilinear Skeleton, Topology Preservation, Directional Thinning, Isthmus.

1 Introduction

Skeletons were originally defined by Blum [1] based on a "grass fire" analogy. Imagine a shape as a field covered by dry grass; if you set on fire the contour of the field, then the meeting points of the flame fronts would constitute the skeleton of the shape.

Various methods can be used to obtain skeletons: Voronoi-based transformations [2,3], distance-based transformations [4,5], general-field methods [6,7] and thinning.

Thinning consists of iteratively removing points of the object (called *simple points*) without changing its topology, until stability. Without constraints, a thinning process results in a *ultimate skeleton*, which has the same topology as the original object, and no simple point. However, ultimate skeletons are not efficient as shape descriptors, due to the fact that they may fail to preserve important geometric information. In 3D, two other kinds of skeletons can be obtained, which are closer to the original Blum's analogy: *curvilinear skeletons* (1D) and *surface skeletons* (2D). Figure 1 shows examples of different skeletons. In the literature, in order to obtain such skeletons, it is usual to constrain the thinning to preserve the skeleton extremity points (i.e. curve end points or surface end points).

A common problem encountered in applications is linked to the sensibility of the skeletonization process to small irregularities of the contour: the skeleton may contain "spurious branches" that would make it difficult to exploit.

Fig. 1. From the left to the right (in red):ultimate, curve and surface skeletons

Thinning algorithms have usually either low computational cost, or good quality results (i.e. few spurious branches), but rarely both. In this paper, we propose a new fast parallel directional thinning algorithm constrained by 1D-isthmuses (a generalization of curve interior points) instead of curve end points, resulting in curve skeletons with very few spurious branches. We also propose variations to provide surface skeletons and ultimate skeletons.

This paper is organized as follows: In the sequel of this section, we recall basic notions of digital topology necessary to understand thinning theory, then we present the different strategies of thinning. In Sect. 2, we present the works on which our method is based. In Sect. 3, we introduce our new curve-thinning algorithm, and prove the preservation of topology. We also introduce variations for surface skeletons and ultimate skeletons. Finally, in Sect. 4, we show some results of our algorithms and compare them to those of classical algorithms of the same kind.

1.1   Basic Notions of Digital Topology

In 3D digital topology, the framework is the *discrete grid* $\mathbb{Z}^3$. The *object* is represented by $X \subset \mathbb{Z}^3$, and its *complementary* $\mathbb{Z}^3 \setminus X$ is denoted by $\bar{X}$. A *point* $p \in \mathbb{Z}^3$ is defined by a triplet of integers (p_1, p_2, p_3).

The notion of *neighborhood* is central for digital topology. In 3D, several neighborhoods are considered: the *6-neighborhood* of p is the set $N_6(p) = \{q \in \mathbb{Z}^3; |q_1 - p_1| + |q_2 - p_2| + |q_3 - p_3| \leq 1\}$; the *26-neighborhood* of p is the set $N_{26}(p) = \{q \in \mathbb{Z}^3; max(|q_1 - p_1|, |q_2 - p_2|, |q_3 - p_3|) \leq 1\}$. For a k-neighborhood, we define $N_k^*(p) = N_k(p) \setminus \{p\}$.

Based on the notion of neighborhood we can define the notion of connectivity. Let p be an element of X, we define the *k-connected component of X containing* p, denoted by $C_k(p, X)$, as the maximal subset of X containing p, such that for all points $a \in C_k(p, X)$, there exists a sequence of points of X $\langle p_0, ..., p_n \rangle$ such that:

- $p_0 = p$ and $p_n = a$,
- $\forall i \in \{1, ..., n\}, p_{i-1}$ is in the k-neighborhood of p_i.

The set of all k-connected components of X is denoted by $C_k(X)$. A subset Y of $\mathbb{Z}^3$ is *k-adjacent* to a point $p \in \mathbb{Z}^3$ if $Y \cap N_k^*(p) \neq \emptyset$. The set of all k-connected

components of X which are k-adjacent to a point p is denoted by $C_k^p(X)$. When we consider the k-connectivity of X, it is mandatory to consider a different $\bar{k}$-connectivity for $\bar{X}$ [8]. In 3D, if $k = 6$, then $\bar{k} = 26$ and inversely. This rule is necessary in order to retrieve some important topological properties such as the Jordan theorem.

Connectivity Numbers. In order to provide a local description of simple points and other characteristic points, Bertrand and Malandain[9] introduced the notion of connectivity numbers. In 3D, the definition of connectivity numbers lies on the notion of *geodesic neighborhood*. Let $X \subseteq \mathbb{Z}^3$ and $p \in \mathbb{Z}^3$. The *t-order k-geodesic neighborhood of p in X* is the set $N_k^t(p, X)$ recursively defined by:

- $N_k^1(p, X) = N_k^*(p) \cap X$
- $N_k^t(p, X) = \bigcup\{N_k(q) \cap N_{26}^*(p) \cap X, q \in N_k^{t-1}(p, X)\}$

The geodesic neighborhoods $G_k(p, X)$ are defined by: $G_6(p, X) = N_6^2(p, X)$ and $G_{26}(p, X) = N_{26}^1(p, X)$. We can now define the connectivity numbers in 3D, for $k \in \{6, 26\}$ as: $T_k(p, X) = |C_k(G_k(p, X))|$, where $|S|$ denotes the number of elements of the set S.

Simple points. Intuitively, a point is simple for X if it can be removed from X without changing its topology. In the digital topology framework, the topology of an object depends on the chosen connectivity; for this reason, when considering a k-connected object, we will talk about k-simple points. The notion of simple point is central for homotopic thinning in the digital framework: a skeleton is obtained by removing iteratively simple points from an object. In 3D, the removal of a point may not only change the number of connected components of the object or its complementary, but may also change the tunnels of the object. Bertrand and Malandain propose a local characterization of simple points using connectivity numbers T_6 and T_{26}:

Proposition 1. *[9] Let $X \subseteq \mathbb{Z}^3$ and $x \in X$. The point x is k-simple for X iff $T_k(x, X) = 1$ and $T_{\bar{k}}(x, \bar{X}) = 1$.*

1.2 Thinning Algorithms

Thinning in the digital framework consists of removing simple points from an object, until either no more simple point can be found (resulting in a ultimate skeleton), or a satisfactory subset of voxels has been reached (resulting in a curvilinear of surface skeleton). Two main strategies are possible for removing simple points: sequential removal and parallel removal.

Sequential algorithms. Sequential removal of simple points can be achieved by detecting simple points in an object, and removing them one after the other, until no more simple point can be found. After removing a simple point, the new set of simple points of the object must be computed. Such basic strategy does not guarantee the result, which is an ultimate skeleton, to be centered in the original

object. It is important, when designing a sequential thinning algorithm, to decide of a removal order of simple points, and of a strategy for preserving interesting visual features of the object. A widely used strategy to obtain a centered skeleton with a sequential thinning process consists of computing a priority function (e.g. distance to the background) on the object and removing the simple points of X according to the value of this function [10] : at each step, the simple point that is removed is one with the lowest possible value.

Parallel algorithms. Whereas iterative algorithms remove only one simple point at a time, parallel algorithms consists in removing a set of simple points simultaneously. The main problem of parallel thinning is that removing simple points simultaneously from an object usually "breaks" the topology. Thus, additional conditions must be introduced in order to solve this problem.

For a given algorithm, the *deletion conditions* are the conditions that a point has to satisfy to be removable. The *support* is the minimal set of points whose values must be examined to verify the deletion conditions for a point $p \in X$. According to Hall [11], parallel thinning algorithms can be divided into three categories:

- In *directional algorithms*, the main loop is divided into sub-iterations, and the deletion conditions are changed from one sub-iteration to another. The most usual kind of directional algorithm use 6 sub-iterations [12,13,15,14]. Directional algorithms with 3 sub-iterations [16], 8 sub-iterations [17] and 12 sub-iterations [18] have also been proposed.
- In *subfield-based algorithms*, the points of the object are decomposed into subfields, and at at a given iteration of the algorithm, only points in a given subfield are studied. Algorithms using 2 subfields [19], 4 subfields [20] and 8 subfields [20] have been proposed.
- In *fully parallel algorithms*, no sub-iteration takes place : the same thinning operator is used at each iteration of the main loop [21,22,23,24].

Parallel algorithms are generally more robust to contour noise than sequential ones. Subfield-based algorithms usually produce shaky skeletons, and fully parallel algorithms require an extended support (usually, $5 \times 5 \times 5$), while for other strategies the usual support is a $3 \times 3 \times 3$ neighborhood (i.e. N_{26}).

For our aims, a parallel directional algorithm is the best choice.

2   Background

Our method is based on two different works. We use constraints that were originally introduced by Bertrand and Couprie [25] in the context of sequential thinning, and the design of the masks used in our parallel directional algorithm is based on those defined by Palágyi and Kuba [12].

2.1   Bertrand and Couprie Isthmus Based Thinning

In 2007, Bertrand and Couprie [25] proposed a sequential thinning based on isthmuses.

Definition 1. *Let $X \subseteq \mathbb{Z}^3, p \in X$,*

- *p is a 1D-isthmus iff $T_k(p, X) \geq 2$.*
- *p is a 2D-isthmus iff $T_{\bar{k}}(p, \bar{X}) \geq 2$.*

The strategy used by Bertrand and Couprie consists in dynamically detecting the considered isthmuses (1D or 2D) and accumulating them in a constraint set. A point can be deleted if it is simple and not in the constraint set. By this way, a point detected as an isthmus in a given iteration cannot be removed later. Depending on the considered isthmuses, the method provides curvilinear skeletons (in case of 1D isthmuses) or surface skeletons (in case of 2D isthmuses).

The consideration of isthmuses instead of extremities points is interesting for two reasons: isthmuses can be easily locally defined and detected (it is not the case of surface end points), and appear less often than extremities during thinning process, leading to less noisy skeletons. However, the sequential approach is less adapted than parallel ones for this purpose, and provides skeletons of lower quality.

2.2   Palágyi and Kuba 6-Directional Thinning

In 1998, Palágyi and Kuba [12] proposed a 6-directional thinning algorithm producing curvilinear skeletons, which we denote by PKD6.

Each iteration of PKD6 consists of 6 sub-iterations in which points are removed from the shape if and only if the configuration of their 26-neighborhood matches at least one mask of the mask set $\mathcal{M}_d$ for the given direction d. The mask set $\mathcal{M}_U$ used for the *UP* direction is presented in Fig. 2. Mask sets $\mathcal{M}_D, \mathcal{M}_N, \mathcal{M}_S, \mathcal{M}_E, \mathcal{M}_W$ for the other directions (*DOWN*, *NORTH*, *SOUTH*, *EAST* and *WEST*) are obtained by appropriate rotations and reflections. The algorithm is summarized in Alg. 1.   Palágyi and Kuba proved that their algo-

---

Algorithm 1. PKD6 [12]

Data: $X \subseteq \mathbb{Z}^3$
Result: A skeleton of X
1 $Y = X$
2 **repeat**
3 | **foreach** *direction d in* (U, D, N, S, E, W) **do**
4 | | $Y = Y \setminus \{p \in Y; p$ matching one mask of $\mathcal{M}_d\}$
5 **until** *stability*;
6 **return** Y

---

rithm preserves topology, using the following theorem (Theorem 1) proved by Ma in [26].

Definition 2.   *[26] Let $X \subset \mathbb{Z}^3$ be an object. The set $D = \{d_1, ..., d_k\} \subseteq X$ is called a simple set of X if D can be arranged in a sequence $\langle d_{i_1}, ..., d_{i_k} \rangle$ in which each d_{i_j} is simple for $X \setminus \{d_{i_1}, ..., d_{i_{j-1}}\}$ for $j = 1, ..., k$.*
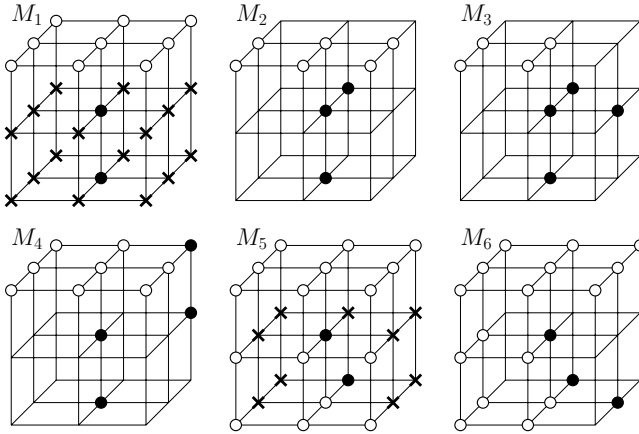
Fig. 2. Base masks M_1-M_6 and their rotations around the vertical axis form the set of masks $\mathcal{M}_U$ for direction *UP*. Points marked by a black disk have to belong to the object, points marked by a white disk have to be out of the object, and at least one point marked by a cross in masks M_1 and M_5 has to belong to the object.

Definition 3. *[26] A 3D parallel reduction operation is said to* preserve topology *if, for all possible 3D pictures, the set of all points that are deleted simultaneously is simple.*

Theorem 1. *[26] A 3D parallel reduction operation preserves topology if all of the following conditions hold:*

1. *Only simple points are deleted.*
2. *If two points belonging to the object, p and q, of a 2×2 square in $\mathbb{Z}^3$ are deleted, then the set $\{p, q\}$ is simple.*
3. *If three points belonging to the object, p, q and r, of a 2×2 square in $\mathbb{Z}^3$ are deleted, then the set $\{p, q, r\}$ is simple.*
4. *If four points belonging to the object, p,q,r and s, of a 2×2 square in $\mathbb{Z}^3$ are deleted, then the set $\{p, q, r, s\}$ is simple.*
5. *No component of the object contained in a $2 \times 2 \times 2$ cube in $\mathbb{Z}^3$ can be deleted completely.*

Theorem 2. *[12] The thinning algorithm PKD6 preserves topology.*

3   New Method

We propose a 6-directional thinning algorithm based on isthmuses. Our algorithm is designed in two steps, separating the thinning process (using a 6-directional approach) and the isthmuses detection. Each iteration consists of:

1. updating the constraint set, by adding points of the object detected as isthmuses,

---

Algorithm 2. D6I1D

Data: $X \subseteq \mathbb{Z}^3$, $K \subseteq X$
Result: A skeleton of X
1 **repeat**
2 $\quad$ $K = K \cup \Psi_{I1D}(X)$
3 $\quad$ **foreach** *direction d in* (U, D, N, S, E, W) **do**
4 $\quad\quad$ $X = \tau_d(X, K)$
5 **until** *stability*;
6 **return** X

---

2. removing deletable points of the object which are not in the constraint set (see Algo.2).

This design allows for the preservation of some points defined a priori by the initial set K (which may be empty). The function $\Psi_{I1D}(X)$ returns the set of all the 1D-isthmuses of X (see Def. 1). The function $\tau_d(X, K)$ returns the set $X \setminus S$, S being the set of all points from $X \setminus K$ respecting deletion conditions in X for direction d. This function is defined in the following of this section.

3.1 Design of Deletion Condition Masks

Contrary to deletion conditions used in directional algorithms found in the literature, those of τ_d permit the deletion of curve end points, the constraint points being considered separately.

Deletion conditions can be represented using a set of masks: a point respects deletion conditions if and only if it matches at least one of the masks. In this section, we present the masks used by τ_d, obtained by modifying those used by PKD6.

In order to separate geometrical and topological conditions, we have to allow in τ_d the deletion of ending points (i.e. points with only one neighbor in the object). For this purpose, we add three masks to $\mathcal{M}_d$, representing ending points which may be removed for the direction d. These masks are shown in Fig. 3 for the case $d = U$.

Proving that our algorithms preserve topology (Prop.2 and Prop.3) can be done using the framework of critical kernels, introduced by Bertrand [27]. This framework is indeed the most powerful one for both proving and designing n-dimensional homotopic thinning algorithms. However, in this particular case it is simpler to build on the proof that was given by Palágyi and Kuba for PKD6, as our algorithm only slightly differs from the latter.

Proposition 2. *Algorithm PKD6 with mask sets $\mathcal{M}_d$ extended by adding convenient rotations of M_α, M_β and M_γ preserves topology.*

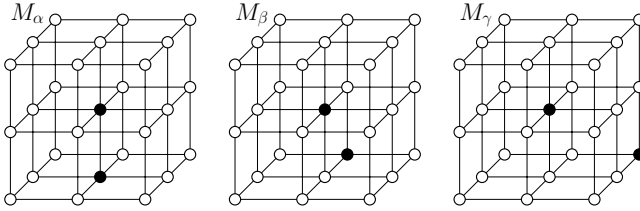*Proof.* We prove that all conditions of theorem 1 hold.

Fig. 3. Masks M_α-M_γ and their rotations around vertical axis are added to the set of masks $\mathcal{M}_U$ for direction UP in order to remove ending points

Let $X \subseteq \mathbb{Z}^3$ and $p \in X$. If p matches a mask in $\mathcal{M}_U$, then we know that it is simple from theorem 2. If p matches a mask in $\{M_\alpha, M_\beta, M_\gamma\}$, then p is obviously simple.

Let Z be a set of object points contained in a 2×2 square in $\mathbb{Z}^3$, or being an object component contained in a $2 \times 2 \times 2$ cube in $\mathbb{Z}^3$, with $|Z| \geq 2$.

If all points of Z match masks of $\mathcal{M}_U$, then we know that all conditions of theorem 1 hold, from theorem 2.

Otherwise, if one point p of Z matches a mask in $\{M_\alpha, M_\beta, M_\gamma\}$, then clearly $|Z| = 2$. Let q be the other point in Z. By examination of all configurations, it can be seen that any mask in $\mathcal{M}_U \cup \{M_\alpha, M_\beta, M_\gamma\}$ positioned on q does not match, thus Z cannot be deleted. $\qquad\square$

3.2   Mask Set Reduction

The set of masks $\mathcal{M}_U \cup \{M_\alpha, M_\beta, M_\gamma\}$ can be compacted as proposed in Fig.4. We can observe that mask M_1' matches exactly the same configurations as those matched by masks M_1 and M_α. In the same way, mask M_5' matches exactly the same configurations as those matched by masks M_5 and M_β. Masks M_2', M_3', M_4', M_6' and M_7' are respectively the same as M_2, M_3, M_4, M_6 and M_γ.

Using the sets $\mathcal{M}_d'$ with $d \in \{U, D, N, S, E, W\}$, we can now define τ_d, as proposed in algorithm 3.

Algorithm 3. τ_d

Data: $X \subseteq \mathbb{Z}^3$, $K \subseteq X$
Result: A subset of X
1   $R =$ set of points matching a mask belonging to $\mathcal{M}_d'$
2   $R = R \setminus K$
3   **return** $X \setminus R$

Proposition 3. *Algorithm D6I1D preserves topology.*

Proof. From proposition 2, algorithm PKD6 with mask sets $\mathcal{M}_d$ extended by adding convenient rotations of M_α, M_β and M_γ preserves topology. We observe that:
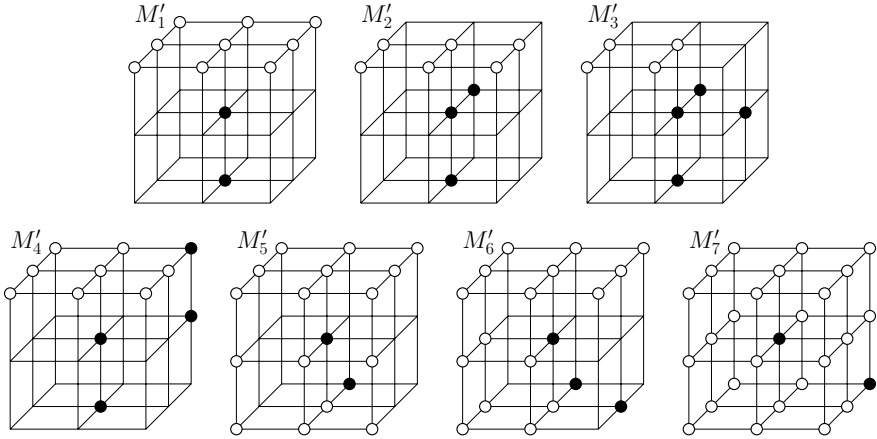
Fig. 4. Final set of masks $\mathcal{M}'_U$ for direction *UP*

- D6I1D differs only from this algorithm by constraining some points to be preserved;
- if theorem 1 holds for an operation A, then it also holds for an operation B that removes only a subset of the points removed by A.

These observations lead to the conclusion that D6I1D preserves topology.   □

3.3   Other Kinds of Skeletons

In fact, our algorithm offers great flexibility, due to the fact that constraint set detection (function Ψ) can be changed to implement other conditions. We provide two examples of such variations.

Ultimate Skeletons. In order to obtain ultimate skeletons instead of curvilinear ones, we propose a very simple variation of D6I1D, consisting in replacing Ψ_{I1D} by the function returning empty set (K is unchanged). We call this variation D6U.

Surface Skeletons. In order to obtain surface skeletons instead of curvilinear ones, we propose a very simple variation of D6I1D, consisting in replacing Ψ_{I1D} by a function Ψ_{I2D} returning the set of 2D-isthmuses (see Def. 1), which can be locally detected. We call this variation D6I2D.

4   Comparative Results

We implemented our method in C++, using two usual optimizations: border lists [24], in order to check only border points in each iteration, and look-up
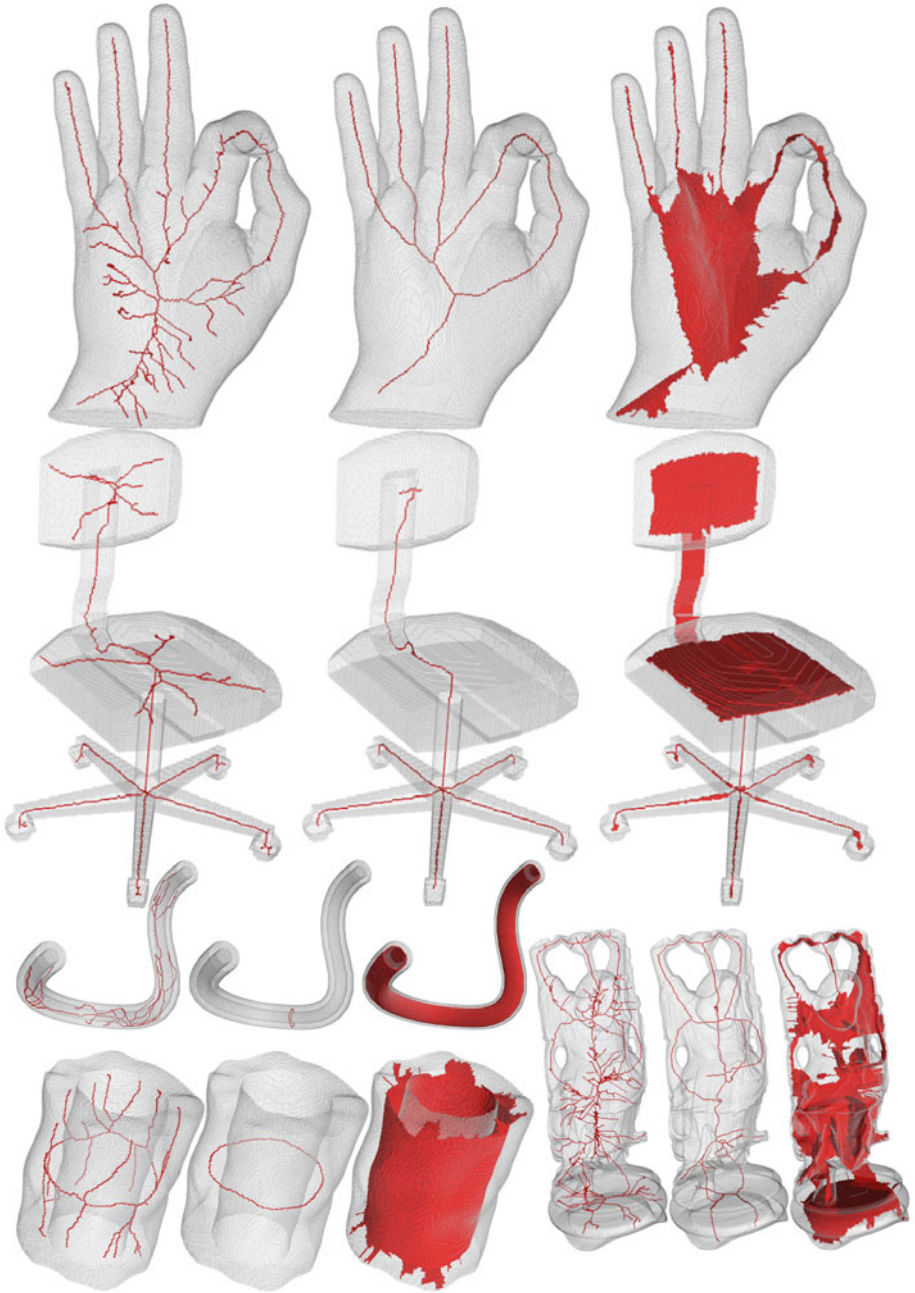
Fig. 5. Skeletons obtained with different algorithms. For each object, from the left to the right: PKD6,D6I1D, D6I2D. The used objects, from the left to the right, then from the top to the bottom: *hand*, *chair*, *pipe*, *cylinder* and *Buddha*.

Table 1. For each object, the size of the image, the numbers of object points (**OP**), and for both *D6I1D* and *D6I2D*, the number of skeleton points (**SP**), the number of iterations (**It**) and the computation time in seconds (**CT(s)**)

Object	Size	OP	PKD6			D6I1D			D6I2D		
			SP	It	CT(s)	SP	It	CT(s)	SP	It	CT(s)
Buddha	$180 \times 400 \times 180$	2 514 223	1 953	90	1.06	1 636	90	1.06	66 492	56	0.98
chair	$300 \times 215 \times 215$	907 601	855	81	0.35	814	82	0.35	20 238	28	0.33
pipe	$300 \times 300 \times 300$	663 586	224	386	1.00	78	385	1.03	54 562	10	0.36
cylinder	$200 \times 200 \times 200$	1 077 425	455	112	0.56	229	112	0.57	31 452	26	0.47
hand	$185 \times 350 \times 220$	2 281 776	899	94	0.80	863	94	0.81	13 756	55	0.79

tables in order to use pre-computation, for each possible configuration of 26-neighborhood, of the deletion tests and isthmuses detection. The results, obtained on a PC under Linux, with a processor Intel(R) Core(TM) 2 Quad Q8200 at 2.33 GHz, are presented in Table 1.

Concerning the quality of the resulting skeleton, as no general quality measurement exists (to our knowledge), we propose to visually compare the results of PKD6 and D6I1D (see Fig. 5). It can be observed that our algorithm provides results with less spurious branches than PKD6.

5   Conclusion

In this paper, we have proposed a new scheme based on isthmus, highly flexible, fast to compute, and providing different kinds of good quality skeletons.

References

1. Blum, H.: An associative machine for dealing with the visual field and some of its biological implications. Biological Prototypes and Synthetic Systems 1, 244–260 (1962)
2. Brandt, J.W., Algazi, V.R.: Continuous skeleton computation by voronoi diagram. CVGIP: Image Understanding 55(3), 329–338 (1992)
3. Näf, M., Székely, G., Kikinis, R., Shenton, M.E., Kübler, O.: 3d voronoi skeletons and their usage for the characterization and recognition of 3d organ shape. CVGIP: Image Understanding 66(2), 147–161 (1997)
4. Borgefors, G., Nyström, I., Sanniti Di Baja, G.: Computing skeletons in three dimensions. Pattern Recognition 32(7), 1225–1236 (1999)
5. Toriwaki, J.I., Mori, K.: Distance transformation and skeletonization of 3d pictures and their applications to medical images. In: Digital and Image Geometry, Advanced Lectures, pp. 412–428. Springer, London (2001)
6. Ahuja, N., Chuang, J.-H.: Shape representation using a generalized potential field model. IEEE Trans. Pattern Anal. Mach. Intell. 19(2), 169–176 (1997)
7. Rumpf, M., Telea, A.: A continuous skeletonization method based on level sets. In: VISSYM 2002, p. 151. Eurographics Association, Aire-la-Ville (2002)
8. Kong, T.Y., Rosenfeld, A.: Digital topology: introduction and survey. Comp. Vision, Graphics and Image Proc. 48, 357–393 (1989)

9. Bertrand, G., Malandain, G.: A new characterization of three-dimensional simple points. Pattern Recognition Letters 15(2), 169–175 (1994)
10. Davies, E.R., Plummer, A.P.N.: Thinning algorithms: A critique and a new methodology. Pattern Recognition 14(1-6), 53–63 (1981)
11. Hall, R.W.: Parallel connectivity-preserving thinning algorithms. Machine Intelligence and Pattern Recognition 19, 145–179 (1996)
12. Palágyi, K., Kuba, A.: A 3D 6-subiteration thinning algorithm for extracting medial lines. Pattern Recognition Letters 19(7), 613–627 (1998)
13. Lohou, C., Bertrand, G.: A 3d 6-subiteration curve thinning algorithm based on p-simple points. Discrete Appl. Math. 151(1-3), 198–228 (2005)
14. Gong, W., Bertrand, G.: A simple parallel 3D thinning algorithm. In: 10th International Conference on Pattern Recognition 1990 Proceedings, pp. 188–190 (1990)
15. Ma, C.M., Wan, S.Y.: Parallel thinning algorithms on 3D (18, 6) binary images. Computer Vision and Image Understanding 80(3), 364–378 (2000)
16. Palágyi, K.: A 3–Subiteration Surface–Thinning Algorithm. In: Computer Analysis of Images and Patterns, pp. 628–635. Springer, Heidelberg (2007)
17. Palágyi, K., Kuba, A.: Directional 3D thinning using 8 subiterations. In: Discrete Geometry for Computer Imagery, pp. 325–336. Springer, Heidelberg (1999)
18. Lohou, C., Bertrand, G.: A 3d 12-subiteration thinning algorithm based on p-simple points. Discrete Appl. Math. 139(1-3), 171–195 (2004)
19. Kardos, P., Németh, G., Palágyi, K.: Topology preserving 2-subfield 3d thinning algorithms. In: Proceedings of the IASTED International Conference on Signal Processing, Pattern Recognition and Applications, pp. 310–316. IASTED, Innsbruck (2010)
20. Németh, G., Kardos, P., Palágyi, K.: Topology Preserving 3D Thinning Algorithms Using Four and Eight Subfields. Image Analysis and Recognition, 316–325 (2010)
21. Manzanera, A., Bernard, T.M., Preteux, F., Longuet, B.: Medial faces from a concise 3D thinning algorithm. In: Proc. 7th IEEE Internat. Conf. Computer Vision, ICCV 1999, vol. 1, pp. 337–343 (1999)
22. Wang, T., Basu, A.: A note on 'A fully parallel 3D thinning algorithm and its applications'. Pattern Recognition Letters 28(4), 501–506 (2007)
23. Bertrand, G., Couprie, M.: A new 3D parallel thinning scheme based on critical kernels. In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) DGCI 2006. LNCS, vol. 4245, pp. 580–591. Springer, Heidelberg (2006)
24. Palágyi, K.: A 3D fully parallel surface-thinning algorithm. Theoretical Computer Science 406(1-2), 119–135 (2008)
25. Bertrand, G., Couprie, M.: Transformations topologiques discrètes. In: Coeurjolly, D., Montanvert, A., Chassery, J. (eds.) Géométrie discrète et images numériques, Hermès, pp. 187–209 (2007)
26. Ma, C.M.: On topology preservation in 3d thinning. CVGIP: Image Understanding 59(3), 328–339 (1994)
27. Bertrand, G.: On critical kernels. Comptes Rendus de l'Académie des Sciences, Série Math. 1(345), 363–367 (2007)