Graphical Models 73 (2011) 335–345

Contents lists available at ScienceDirect

Graphical Models

journal homepage: www.elsevier.com/locate/gmod

Thinning combined with iteration-by-iteration smoothing for 3D binary images

Gábor Németh, Péter Kardos, Kálmán Palágyi *

Department of Image Processing and Computer Graphics, University of Szeged, Hungary

ARTICLE INFO

Article history:
Received 19 August 2010
Received in revised form 9 February 2011
Accepted 13 February 2011
Available online 20 February 2011

Keywords:
Thinning
Contour smoothing
Parallel reduction operators
Topology preservation

ABSTRACT

In this work we present a new thinning scheme for reducing the noise sensitivity of 3D thinning algorithms. It uses iteration-by-iteration smoothing that removes some border points that are considered as extremities. The proposed smoothing algorithm is composed of two parallel topology preserving reduction operators. An efficient implementation of our algorithm is sketched and its topological correctness for (26, 6) pictures is proved.

1. Introduction

Skeletons are frequently applied shape features in image processing, pattern recognition, and visualization, hence fast skeletonization is extremely important for large 3D objects [1–6]. Unfortunately, skeletonization methods are rather sensitive to coarse object boundaries, hence the produced skeletons generally contain some false segments. In order to overcome this problem, unwanted skeletal parts are usually removed by a pruning process as a post-processing step [7–11].

Thinning algorithms [12] are capable of extracting skeleton-like shape descriptors in a topology preserving way [13]. In 3D, surface-thinning algorithms are to extract *medial surfaces* by preserving *surface-endpoints* and curve-thinning algorithms produce *centerlines* by preserving *curve-endpoints* [14]. Due to the topological constraint, each arisen endpoint is to be connected with the medial surface or the centerline of the given elongated object. Hence the number of unwanted skeletal parts can be reduced by removing some "unimportant" endpoints during the thinning process. In this paper we propose a new thinning scheme that uses iteration-by-iteration contour smoothing. Since unwanted endpoints are salient object points, the proposed topology preserving smoothing algorithm is to remove additive contour noise elements.

There exist numerous approaches for smoothing binary objects in 2D and 3D [15–18]. Yu and Yan developed a 2D sequential boundary smoothing algorithm that uses operations on chain codes [15]. It removes some noisy pixels along a contour, decomposes the contour into a set of straight lines, and detects structural feature points which correspond to convex and concave segments along the contour. Based on this work, Hu and Yan proposed an improved algorithm [16]. The method that is introduced by Taubin is suitable for smoothing piecewise linear shapes of arbitrary dimensions [17]. This method is a linear low-pass filter that removes high curvature variations. These three approaches mentioned above cannot smooth 3D binary objects. In [18], Couprie and Bertrand introduced the homotopic alternating sequence filter (HASF), a topology preserving operator which is controlled by a constraint set. Their HASF is a composition of homotopic cuttings and fillings by spheres of various radii. Unfortunately, the efficient implementation scheme for parallel thinning

* Corresponding author. Fax: +36 62 546397.
E-mail addresses: gnemeth@inf.u-szeged.hu (G. Németh), pkardos@inf.u-szeged.hu (P. Kardos), palagyi@inf.u-szeged.hu (K. Palágyi).

1524-0703/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.gmod.2011.02.001

[19,20] cannot be adopted to the HASF, hence we have not combined it with 3D parallel thinning algorithms.

That is why we proposed a parallel 3D smoothing algorithm for 3D binary images [21]. Our first algorithm removes some border points that are considered as extremities. It is composed of two topology preserving parallel reduction operators, hence the entire algorithm is topology preserving too.

In this work we present the advanced version of that smoothing algorithm that is capable of removing much more salient border points than the previously proposed one. Deletable points (i.e., object points to be deleted simultaneously in the two-pass process) are given by $3 \times 3 \times 3$ matching templates.

The rest of this paper is organized as follows. Section 2 gives an outline of 3D digital topology. In Section 3 we propose our new 3D parallel smoothing algorithm. Section 4 gives the new thinning scheme that uses iteration-by-iteration smoothing for reducing the noise sensitivity of 3D thinning algorithms. Section 5 presents an efficient implementation of the proposed smoothing algorithm. The topology preservation of the advanced smoothing algorithm for (26, 6) binary pictures is proven in Section 6. After, we round off the paper with a few brief concluding remarks.

2. Basic notions and results

In this paper, we use the fundamental concepts of digital topology as reviewed by Kong and Rosenfeld [13].

Let p be a point in the 3D digital space denoted by $\mathbb{Z}^3$. Let us denote $N_j(p)$ (for $j = 6, 18, 26$) the set of points that are *j-adjacent* to point p (see Fig. 1a).

The sequence of distinct points $\langle x_0, x_1, \ldots, x_n \rangle$ is called a *j-path* (for $j = 6, 18, 26$) of length n from point x_0 to point x_n in a non-empty set of points X if each point of the sequence is in X and x_i is j-adjacent to x_{i-1} for each $1 \leqslant i \leqslant n$ (see Fig. 1a). Note that a single point is a j-path of length 0. Two points are said to be *j-connected* in the set X if there is a j-path in X between them.

The *3D binary (26,6) digital picture* $\mathcal{P}$ is a quadruple $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ [13]. Each element of $\mathbb{Z}^3$ is called a *point* of $\mathcal{P}$. Each point in $B \subseteq \mathbb{Z}^3$ is called a *black point* and has a value of 1. Each point in $\mathbb{Z}^3 \setminus B$ is called a *white point* and has a value of 0. 26-adjacency is associated with the black points and 6-adjacency is assigned to the white ones. A *black component* is a maximal 26-connected set of points in B, while a *white component* is a maximal 6-connected set of points in $\mathbb{Z}^3 \setminus B$. A black point is called a *border point* in a (26, 6) picture if it is 6-adjacent to at least one white point.

A *reduction operator* transforms a binary picture only by changing some black points to white ones (which is referred to as the deletion of 1's). A *parallel reduction operator* deletes all points satisfying its condition simultaneously. A 3D reduction operator does *not* preserve topology [22] if any black component is split or is completely deleted, any white component is merged with another white component, a new white component is created, or a hole (that donuts have) is eliminated or created.

A *simple point* is a black point whose deletion is a topology preserving reduction [13]. Now we will make use the following result:

Theorem 1. [23] *A black point p is simple in picture $(\mathbb{Z}^3, 26, 6, B)$ if and only if all of the following conditions hold:*

1. *The set $(B\backslash\{p\}) \cap N_{26}(p)$ contains exactly one 26–component.*
2. *The set $(\mathbb{Z}^3 \setminus B) \cap N_6(p)$ is not empty.*
3. *Any two points in $(\mathbb{Z}^3 \setminus B) \cap N_6(p)$ are 6-connected in the set $(\mathbb{Z}^3 \setminus B) \cap N_{18}(p)$.*

Based on Theorem 1, simple points can be locally characterized; the support of an operator which deletes (26,6)–simple points is $3 \times 3 \times 3$.

Parallel reduction operators delete a set of black points and not just a single simple point. Hence we need to consider what is meant by topology preservation when a number of black points are deleted simultaneously. The following theorem provides *sufficient conditions* for 3D parallel reduction operators to preserve topology.

Theorem 2. [14] *Let $\mathcal{O}$ be a parallel reduction operator. Let p be any black point in any picture $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ such that p is deleted by $\mathcal{O}$. Let $\mathcal{Q}$ be the family of all the sets of*

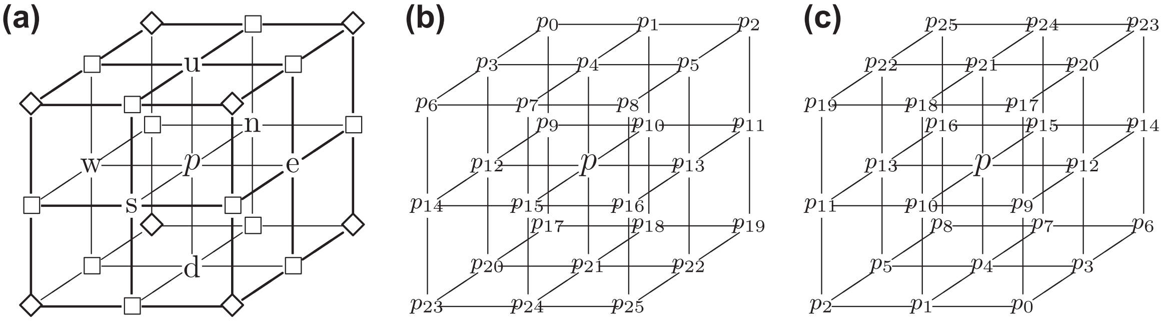

Fig. 1. Frequently used adjacencies in $\mathbb{Z}^3$ (a). The set $N_6(p)$ of the central point $p \in \mathbb{Z}^3$ contains p and the 6 points marked $U = u(p)$, $N = n(p)$, $E = e(p)$, $S = s(p)$, $W = w(p)$, and $D = d(p)$. The set $N_{18}(p)$ contains the set $N_6(p)$ and the 12 points marked "□". The set $N_{26}(p)$ contains the set $N_{18}(p)$ and the 8 points marked "◇". Indexing schemes to encode all possible $3 \times 3 \times 3$ configurations (b and c). They are assigned to the first (b) and the second (c) parallel reduction operators of the proposed method.

$Q \subseteq (N_{18}(p)\setminus\{p\}) \cap B$ contained in a $2 \times 2 \times 1$, a $2 \times 1 \times 2$, or a $1 \times 2 \times 2$ subset of $\mathbb{Z}^3$. The operator $\mathcal{O}$ is topology preserving if all of the following conditions hold:

1. *p is simple in the picture $(\mathbb{Z}^3, 26, 6, B \setminus Q)$ for any Q in $\mathcal{Q}$.*
2. *No black component contained in a $2 \times 2 \times 2$ cube can be deleted completely by $\mathcal{O}$.*

3. The new smoothing algorithm

In this section, we present an advanced parallel algorithm for smoothing 3D binary pictures.

The proposed algorithm is composed of two parallel reduction operators denoted by R_1 and R_2. Deletable points in these reduction operators are given by sets of $3 \times 3 \times 3$ matching templates. Templates are usually composed of three kinds of elements: *black*, *white*, and *don't care*. A *black* element matches a black point, a *white* one matches a white point, and a *don't care* template position matches either a black or a white point. A black point p of a picture is deletable if at least one template in the corresponding set of templates matches the neighborhood configuration of p. (Note that a template with k ($k = 0, 1, \ldots$) *don't care* elements matches exactly 2^k binary configurations.)

A point is deletable by R_1 if at least one template in the set of 37 templates

$$\mathcal{T}_{R_1} = \{U_0, \ldots, U_8,\ N_0, \ldots, N_8,\ W_0, \ldots, W_8,\\ UN, \ldots, NE,\ UNW, \ldots, USW\}$$

shown in Figs. 2–6 matches it. In these figures, we use the following notations: each element marked "c" (that is the central element of a template), "•", or "■" matches a black point, each white template element is denoted by a "○", and positions masked "·" correspond to the *don't care* template elements. (Note that using different symbols for black template positions helps us to prove the topological correctness of the algorithm.)

Deletable points by operator R_2 are defined by matching templates too. Templates in Figs. 2–6 reflected to the point p are taken into consideration by reduction operator R_2. Note that template positions marked "•" in templates assigned to operator R_1 (see Figs. 2–6) coincide with the 13 elements marked $p_0, p_1, \ldots, p_{12}$ in Fig. 1b. Template positions marked "•" in templates assigned to operator R_2 correspond to the remaining 13 elements marked $p_{13}, p_{14}, \ldots, p_{25}$.

Our smoothing algorithm consists of two steps. First, points are deleted according to the rules of operator R_1. Then, in a basically identical step, all points deletable by R_2 are removed simultaneously.

Deletable points of our first two-pass smoothing algorithm [21] were given by 13–13 matching templates. The set of templates assigned to its first phase was

$$\{U_0, N_0, W_0,\ UN, UE, US, UW, NW, NE,\ UNW, UNE, USE, USW\}$$

(see Figs. 2–6). Since the set $\mathcal{T}_{R_1}$ contains 24 additional templates ($U_1, \ldots, U_8,\ N_1, \ldots, N_8,\ W_1, \ldots, W_8$), the new algorithm can remove much more salient border points. Figs. 7–9 are to compare the proposed algorithm with our first attempt [21]. Numbers in parentheses mean the count of object points. Notice that both of them are proper

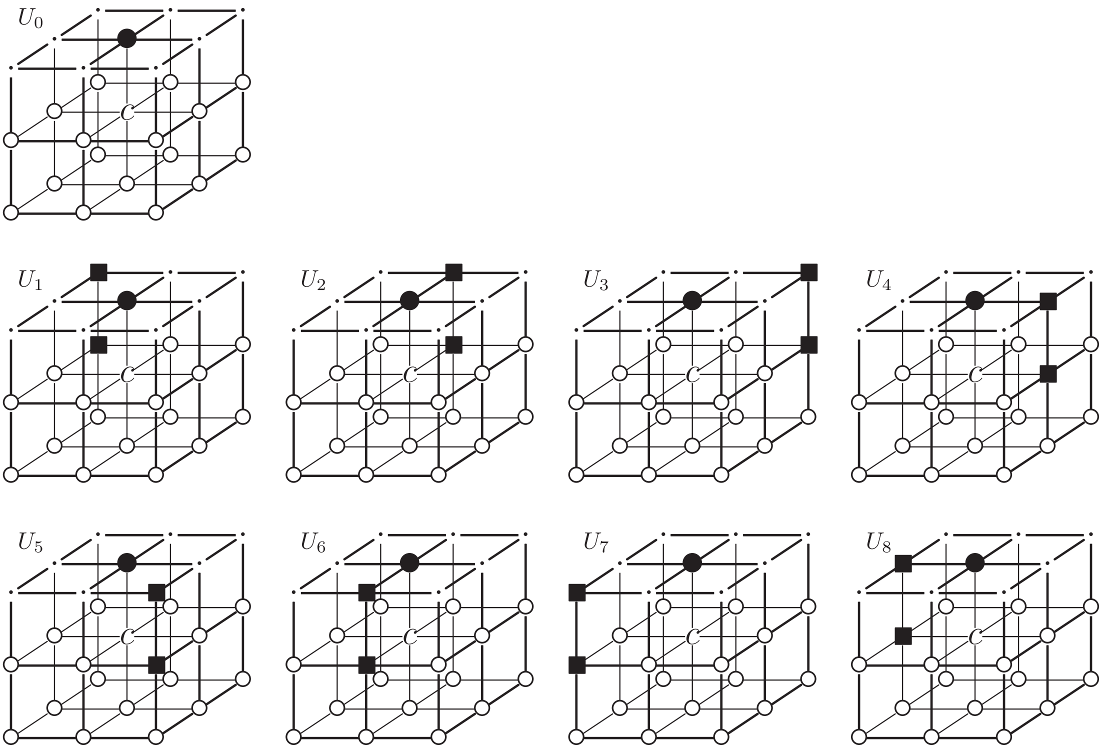

Fig. 2. The nine templates U_i ($i = 0, 1, \ldots, 8$) assigned to the U-face.

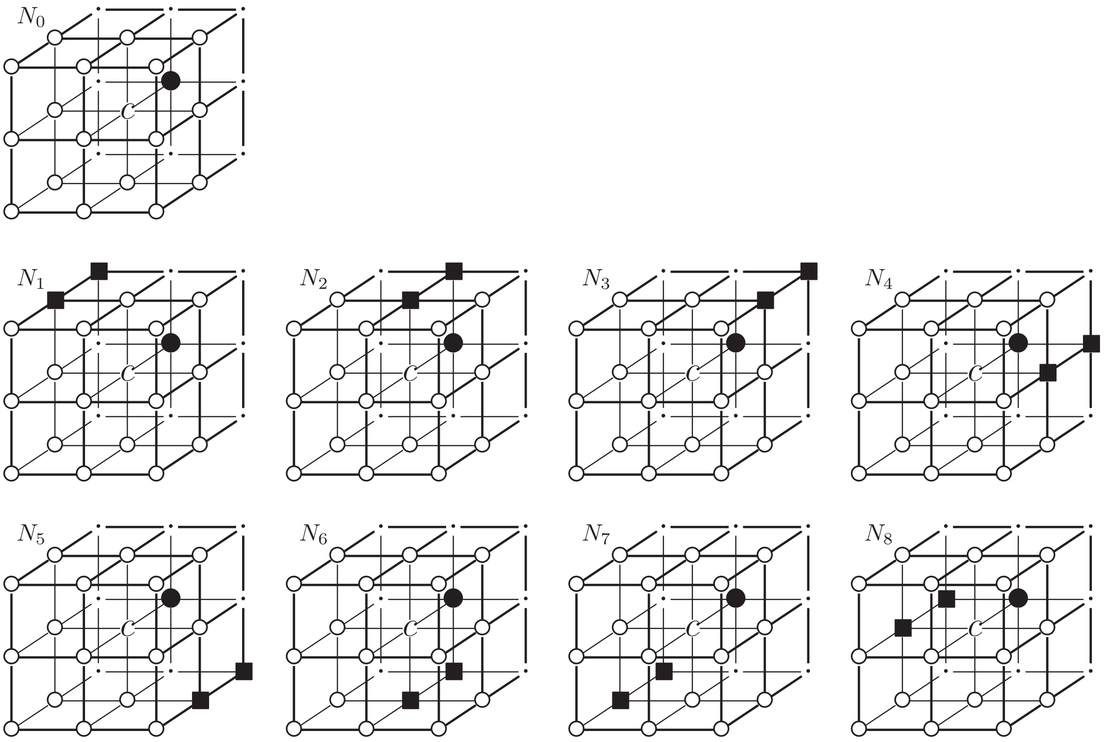

Fig. 3. The nine templates N_i ($i = 0, 1, \dots, 8$) assigned to the N-face.

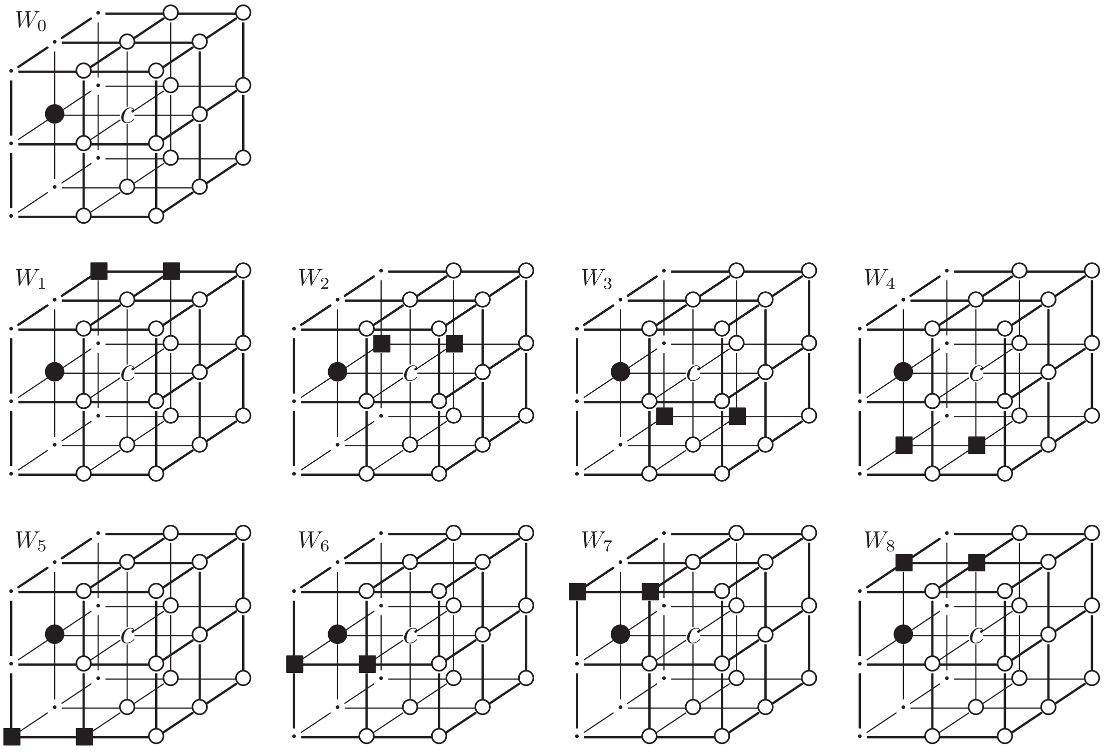

Fig. 4. The nine templates W_i ($i = 0, 1, \dots, 8$) assigned to the W-face.

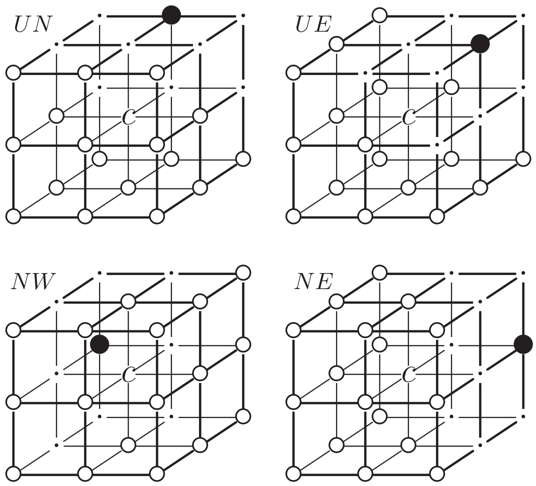

Fig. 5. Templates assigned to the first six edges.

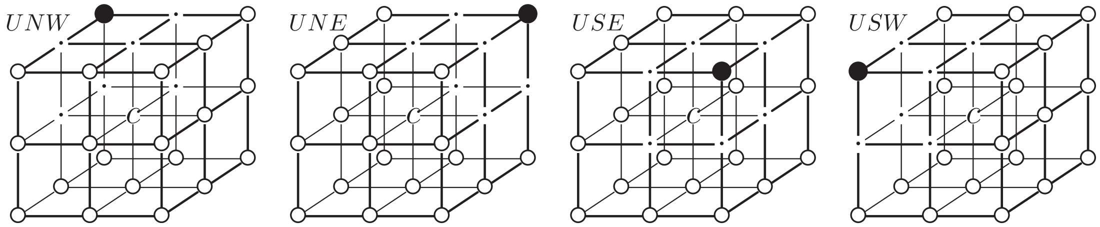

Fig. 6. Templates assigned to the first four nodes.

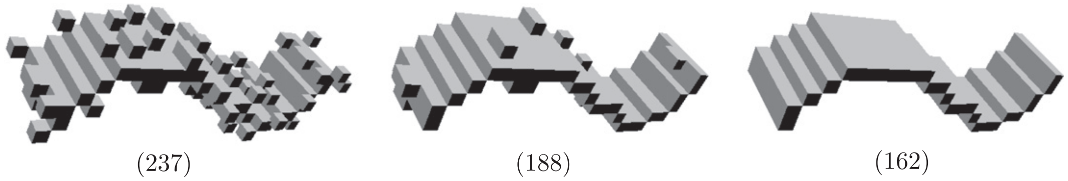

Fig. 7. A 20 × 30 × 10 3D image of a noisy ribbon (left), the smoothed image produced by our first algorithm [21] (middle), and the result of the advanced algorithm (right).

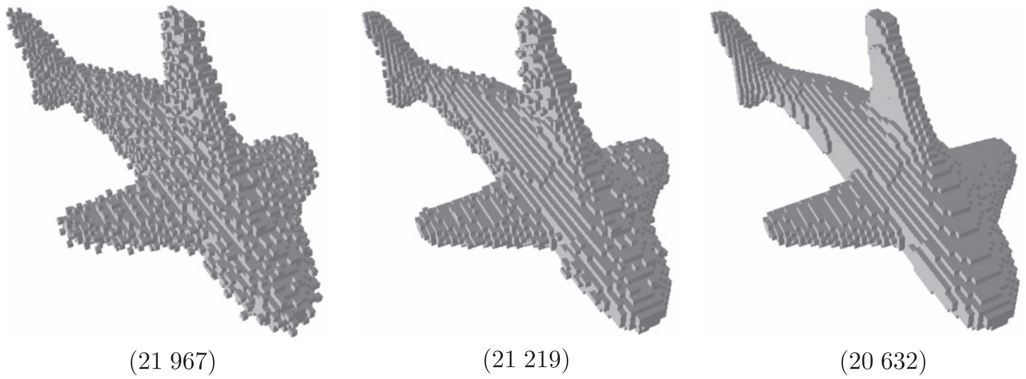

Fig. 8. A 103 × 42 × 60 3D image of a noisy shark (left), the smoothed image produced by our first algorithm [21] (middle), and the result of the advanced algorithm (right).

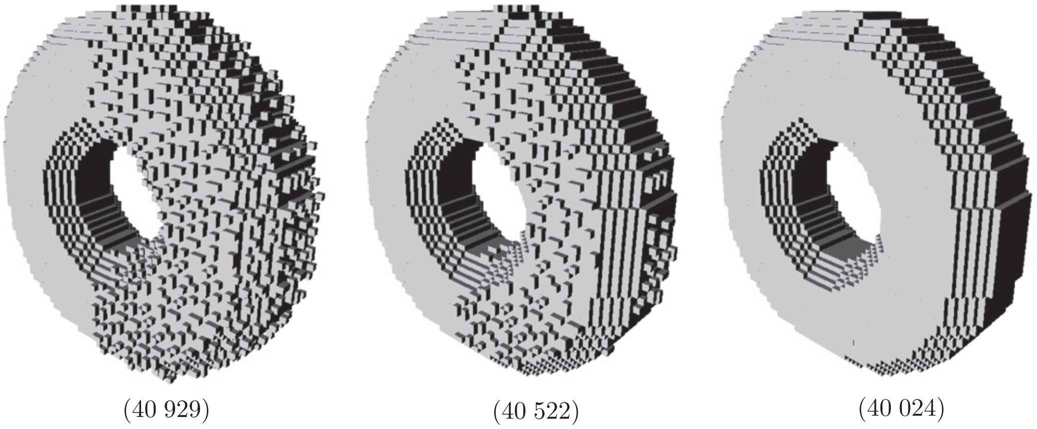

Fig. 9. A 64 × 64 × 19 3D image of a noisy torus (left), the smoothed image produced by our first algorithm [21] (middle), and the result of the advanced algorithm (right). Notice that the smooth boundary segments are not altered by the proposed algorithm.

smoothing algorithms, since they do not alter the smooth boundary segments of the original image (see Fig. 9).

4. The new thinning scheme

We are to apply our smoothing algorithm for reducing the noise sensitivity of 3D parallel thinning algorithms. Consider an arbitrary thinning algorithm called $\mathcal{T}$. The proposed thinning scheme combined with iteration-by-iteration smoothing is sketched by the following program:

```
Input:  picture($\mathbb{Z}^3, 26, 6, X$)
Output:  picture($\mathbb{Z}^3, 26, 6, Y$)
begin
    $Y = X$ ;
    repeat
      //smoothing
        $Y = Y \setminus \{p | p$ is deletable by $R_1$ in $(\mathbb{Z}^3, 26, 6, Y)\}$ ;
        $Y = Y \setminus \{p | p$ is deletable by $R_2$ in $(\mathbb{Z}^3, 26, 6, Y)\}$ ;
      //one thinning iteration
        $D = \{p | p$ is deletable by $\mathcal{T}$ in $(\mathbb{Z}^3, 26, 6, Y)\}$ ;
        $Y = Y \setminus D$ ;
    until $D = \emptyset$ ;
end
```

In experiments the proposed thinning scheme was tested on objects of various images. Here we present six examples, where six kinds of 3D parallel thinning algorithms were applied (Figs. 10–15). Numbers in parentheses mean the count of object points.

Note that a modified version of the proposed smoothing algorithm is to be combined with curve-thinning algorithms. That is why the 13 templates

$$U_0, N_0, W_0,\ UN, UE, US, UW, NW, NE,\ UNW, UNE, USE, USW$$

in $\mathcal{T}_{R_1}$ (see Figs. 2–6) can truncate 1-point thin curves. It is easy to overcome this problem by modifying these 13 masks in the following way: at least one element marked "·" matches a black point. The topology preservation of the proposed smoothing algorithm for $(26, 6)$ binary pictures is proven in Section 6. Since the modification suggested above yields a more restrictive algorithm, the modified smoothing process for reducing the noise sensitivity of 3D curve-thinning algorithms is topology preserving as well.

5. Implementation

If the 37+37 templates of operators R_1 and R_2 are considered, then one may think that the proposed algorithm is time consuming and it is rather difficult to implement it on conventional sequential computers. Thus we sketch here an efficient and fairly general implementation method. It can be used for various reduction operators (e.g., parallel thinning algorithms) as well [19,20].

The proposed implementation uses just one pre-calculated look-up-table (LUT) to encode deletable points. Since the 3 × 3 × 3 support of our operators contains 26 points with the exception of the central point in question (see Figs. 2–6), the LUT has 2^{26} entries of 1 bit in size. It is not hard to see that it requires just 8 MB of storage space in memory.

An integer in $[0, 2^{26})$ can be assigned to each 3 × 3 × 3 configuration. This index is calculated as $\sum_{k=0}^{25} 2^k p_k$, where $p_k \in \{0, 1\}$ $(k = 0, \ldots, 25,$ see Fig. 1b and c). We applied the indexing scheme depicted in Fig. 1b when the LUT assigned to operator R_1 was built. The ith bit of that LUT has the value of 1 if the central point of the ith configuration is deletable by R_1, otherwise a value of 0 is assigned to the ith bit of the LUT $(i = 0, \ldots, 2^{26})$. If a matching template in the set of 37 templates of operator R_1 contains n $(n = 0, 1, \ldots)$ *don't care* elements, then the central points of the matched 2^n configurations are deletable by R_1.

Note that operator R_2 does not need an additional LUT. Operator R_2 can be executed by the LUT assigned to R_1, but it is to be addressed by the reflected indexing scheme depicted in Fig. 1c.

In addition, two lists are used to speed up the process: one for storing the border points in the current picture (since operators R_1 and R_2 can only delete border points, thus the repeated scans of the entire image array are avoided); the second list is to store all deletable points in the current phase of the process.

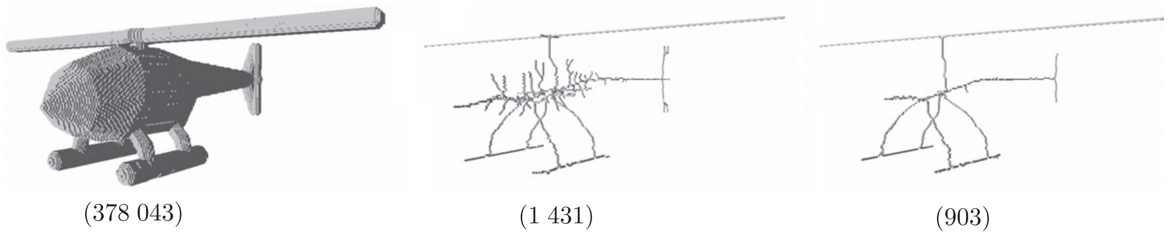

Fig. 10. A 304 × 96 × 261 3D image of a helicopter (left), its centerlines produced by the 6-subiteration curve-thinning algorithm proposed by Palágyi and Kuba [24] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

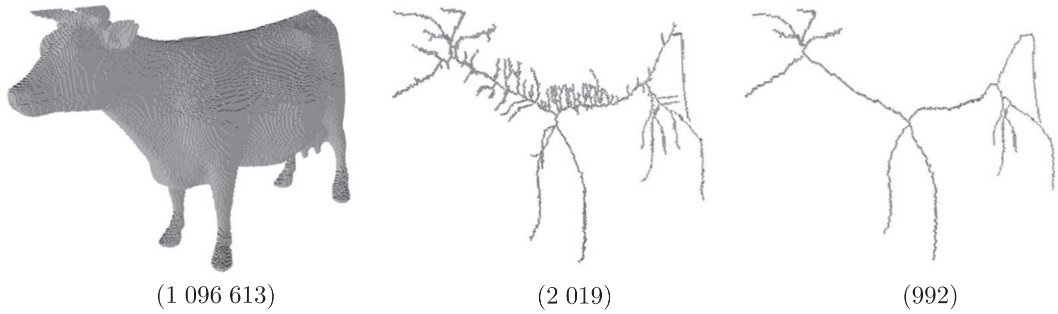

Fig. 11. A 175 × 93 × 285 3D image of a cow (left), its centerlines produced by the 8-subfield curve-thinning algorithm proposed by Németh et al. [27] with the endpoint characterization introduced by Bertrand and Aktouf [25] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

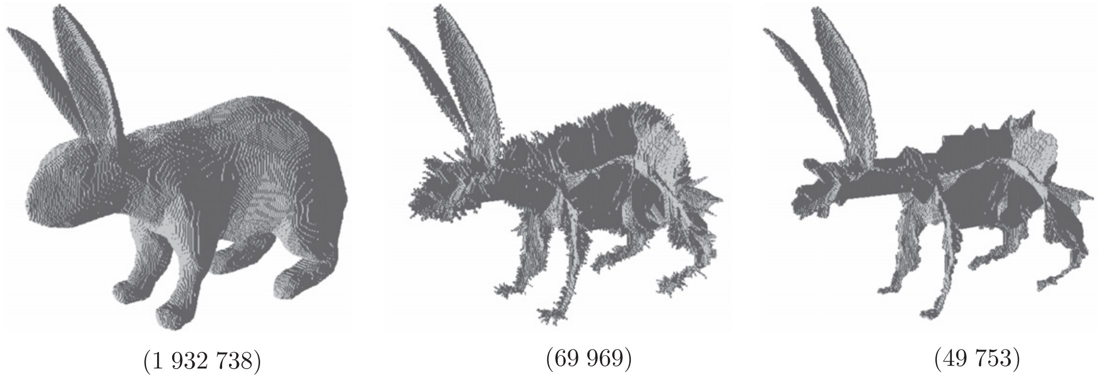

Fig. 12. A 124 × 207 × 300 3D image of a rabbit (left), its centerlines produced by the 6-subiteration surface-thinning algorithm proposed by Gong and Bertrand [26] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

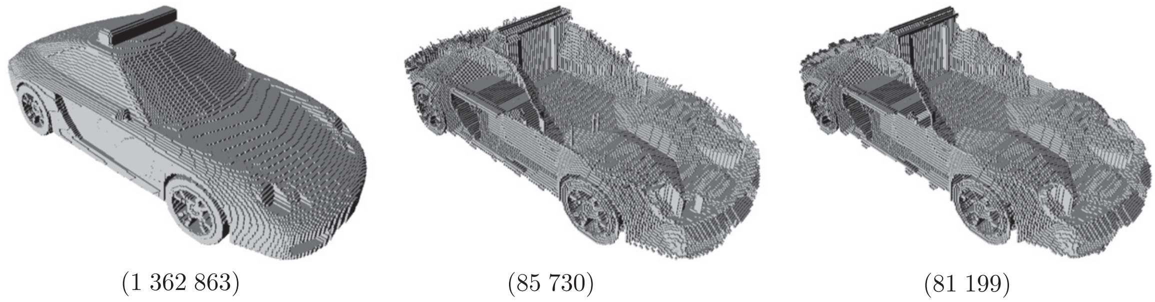

Fig. 13. A 139 × 90 × 285 3D image of a car (left), its medial surface produced by the 2-subfield surface-thinning algorithm proposed by Németh et al. [27] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

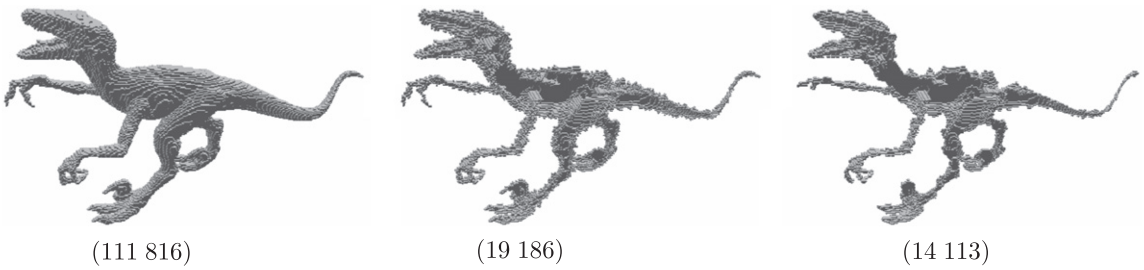

Fig. 14. A 59 × 285 × 139 3D image of a raptor (left), its medial surface produced by the fully parallel surface-thinning algorithm proposed by Manzanera et al. [28] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

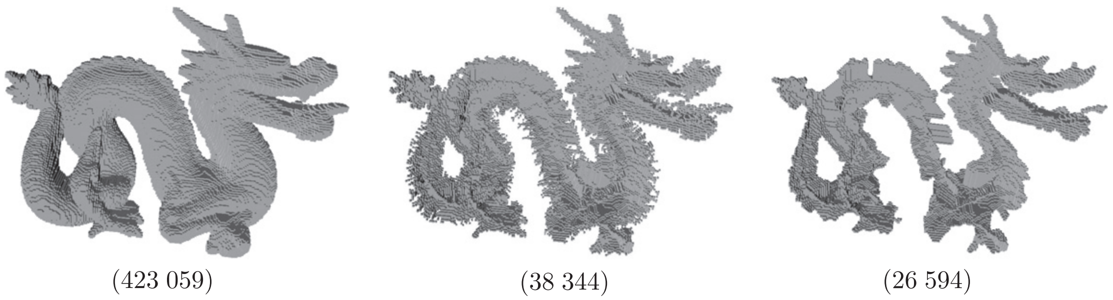

Fig. 15. A 135 × 86 × 191 3D image of a dragon (left), its medial surface produced by the 8-subfield surface-thinning algorithm proposed by Németh et al. [29] (middle), and the result of that algorithm combined with iteration-by-iteration smoothing.

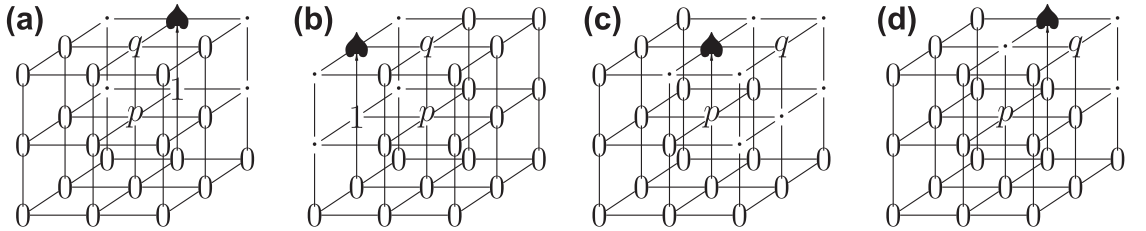

Fig. 16. Possible configurations in which point p is deleted by template U_i ($i = 0, 1, \ldots, 8$) and q is to be deleted by templates N_6 (a) or W_4 (b). Possible configurations in which point p is deleted by template UE and q is to be deleted by templates W_i ($i = 0, 1, \ldots, 8$) (c) and NW (d). Each black point marked "♠" is not deletable by R_1.

6. Verification

Now we will show that the proposed smoothing algorithm is topology preserving for (26, 6) pictures. We are to prove that the first operator R_1 given by the set of matching templates $\mathcal{T}_{R_1}$ fulfills both conditions of Theorem 2. It can be proved for the second operator R_2 in the same way. Hence the entire smoothing algorithm is topology preserving, since it is composed of topology preserving reductions.

Let us classify the elements of the templates in the set of templates $\mathcal{T}_{R_1}$ (see Figs. 2–6). The element in the centre of a template (marked "p") is called *central*. A noncentral template element is called *black* if it is marked "•" or "■". A noncentral template element is called *white* if it is marked "∘". Any other noncentral template element which is neither white nor black, is called *potentially black* (marked "·"). A black or a potentially black noncentral template element is called *nonwhite*.

A black point p is *deletable* if at least one template in the set of 37 templates in $\mathcal{T}_{R_1}$ matches it (i.e., if it is deletable by R_1).

Lemma 1. *Each deletable point is simple.*

Proof. The first thing we need to verify is that there exists a 26-path between any two potentially black positions (Condition 1 of Theorem 1). Here it is sufficient to show that any potentially black position is 26-adjacent to a black position and any black position is 26-adjacent to another black position. This is really apparent from a careful examination of the templates in $\mathcal{T}_{R_1}$.

To prove that Conditions 2 and 3 of Theorem 1 hold, it is sufficient to show that, for each template,

- there exists a white position 6–adjacent to the central position,
- for any potentially black or white position 6-adjacent to the central position p, there exists a 6-adjacent white 18-neighbour which is 6-adjacent to a white position 6-adjacent to p.

These two points are obvious by a careful examination of the set of templates $\mathcal{T}_{R_1}$. □

Lemma 2. *The simplicity of a deletable point does not depend on any point coinciding with a template position marked "■". (In other words, a deletable point remains simple after the deletion of any (sub)set of points coinciding with potentially black or "■" template positions.)*

It can be seen similarly as Lemma 1.

Lemma 3. *Let p and q be any two black points in a picture $(\mathbb{Z}^3, 26, 6, B)$ such that $q \in N_{18}(p)$. If both points p and q are deletable, then p is simple in picture $(\mathbb{Z}^3, 26, 6, B \setminus \{q\})$.*

Proof. Since point p is deletable, by Lemma 1 it is simple. To prove this lemma, we must show that p remains simple after the deletion of q.

If q coincides with a potentially black template element, then this lemma holds by Lemma 2. Hence it is sufficient to deal with the deletable points coinciding with template elements marked "•" in templates U_i, N_i, W_i, *UN*, *UE*, *US*, *UW*, *NW*, and *NE* ($i = 0, 1, \ldots, 8$, see Figs. 2–5). We do not have to take templates *UNW*, *UNE*, *USE*, and *USW* into consideration since elements marked "•" in these four templates are not 18-adjacent to their central elements marked "p" (see Fig. 6).

Let us see the 33 templates in question:

- If p is deleted by U_i ($i = 0, 1, \ldots, 8$), then $q = u(p)$ may be deleted by templates N_6 or W_4. The two possible configurations are depicted in Fig. 16a and b.
- If p is deleted by N_i ($i = 0, 1, \ldots, 8$), then $q = n(p)$ may be deleted by templates U_6, W_6, *US*, *USE*, or *USW*. The four possible configurations are depicted in Fig. 17.
- If p is deleted by W_i ($i = 0, 1, \ldots, 8$), then $q = w(p)$ may be deleted by templates U_4, N_4, *UE*, *NE*, *UNE*, or *USE*. The four possible configurations are depicted in Fig. 18.

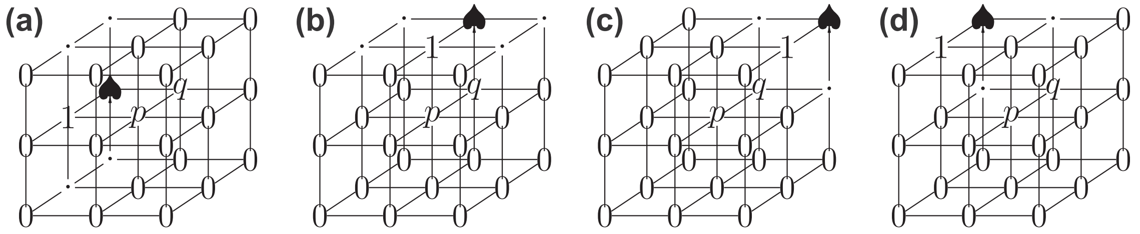

Fig. 17. Possible configurations in which point p is deleted by template N_i ($i = 0, 1, \ldots, 8$) and q is to be deleted by templates W_6 (a), U_6 or *US* (b), *USE* (c), and *USW* (d). Each black point marked "♠" is not deletable by R_1.

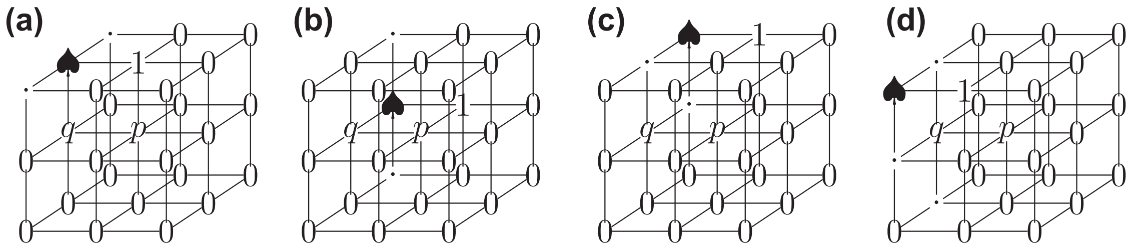

Fig. 18. Possible configurations in which point p is deleted by template W_i ($i = 0, 1, \ldots, 8$) and q is to be deleted by templates U_4 or *UE* (a), N_4 or *NE* (b), *UNE* (c), and *USE* (d). Each black point marked "♠" is not deletable by R_1.

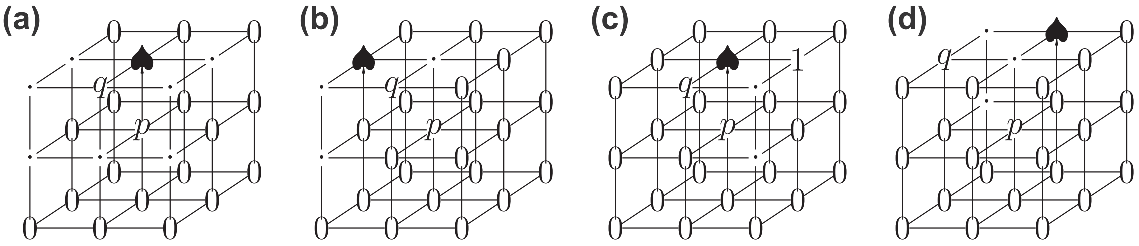

Fig. 19. Possible configurations in which point p is deleted by template *US* and q is to be deleted by templates N_i ($i = 0, 1, \ldots, 8$) (a), *NW* (b), and *NE* (c). Possible configuration in which point p is deleted by template *UW* and q is to be deleted by template *NE* (d). Each black point marked "♠" is not deletable by R_1.

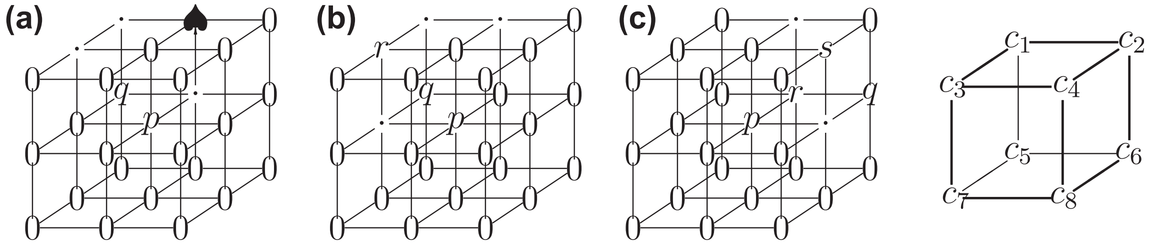

Fig. 20. Possible configurations in which point p is deleted by template *NW* and q is to be deleted by templates *UE* (a) and *US* (b). Possible configuration in which point p is deleted by template *NE* and q is to be deleted by templates W_i ($i = 0, 1, \ldots, 8$) (c). Black point marked "♠" is not deletable by R_1. The sets of black points $\{p, q, r\}$ in (b) and $\{p, q, r, s\}$ in (c) are not contained in a $2 \times 2 \times 1$, a $2 \times 1 \times 2$, or a $1 \times 2 \times 2$ subset of $\mathbb{Z}^3$. The $2 \times 2 \times 2$ cube that contains a black component C (right).

- If p is deleted by *UN*, then $q = n(u(p))$ is not deletable by R_1.
- If p is deleted by *UE*, then $q = e(u(p))$ may be deleted by templates W_i ($i = 0, 1, \ldots, 8$) or *NW*. The two possible configurations are depicted in Fig. 16c and d.
- If p is deleted by *US*, then point $q = s(u(p))$ may be deleted by templates N_i ($i = 0, 1, \ldots, 8$), *NW*, or *NE*. The three possible configurations are depicted in Fig. 19a,b, and c.
- If p is deleted by *UW*, then point $q = w(u(p))$ may only be deleted by template *NE*. The possible configuration is depicted in Fig. 19d. It is not hard to see that p remains simple after the deletion of q.
- If p is deleted by *NW*, then point $q = w(n(p))$ may be deleted by templates *UE* or *US*. The two possible configurations are depicted in Fig. 20a and b.
- If p is deleted by *NE*, then $q = e(n(p))$ may be deleted by templates W_i ($i = 0, 1, \ldots, 8$). The possible configuration is depicted in Fig. 20c.

It is easy to see that p remains simple after the deletion of q in all cases. □

Lemma 4. *No black component C contained in a $2 \times 2 \times 2$ cube can be deleted completely by the operator R_1.*

Proof. Let us examine the $2 \times 2 \times 2$ cube depicted in Fig. 20.

It is easy to check that if $c_1 \in C$, then c_1 is not deletable by R_1, and if $c_k \in C$ ($k = 2, \ldots, 8$), then there exists a $c_j \in C$ ($j = 1, \ldots, k - 1$) that is not deletable by R_1. Thus C cannot be deleted completely. □

We are now ready to state our main theorem.

Theorem 3. *Operator R_1 is topology preserving for (26, 6) pictures.*

Proof. We need to show that both conditions of Theorem 2 are satisfied:

1. Let us examine the simplicity of a deletable point p in $(\mathbb{Z}^3, 26, 6, B \setminus Q)$, where the set of deletable points $Q \subseteq (N_{18}(p) \setminus \{p\}) \cap B$ is contained in a $2 \times 2 \times 1$, a $2 \times 1 \times 2$, or a $1 \times 2 \times 2$ subset of $\mathbb{Z}^3$. It is clear that the number of elements in Q (denoted by $\#(Q)$) is less than or equal to 3.
   The following points have to be checked:
   - $\#(Q) = 0$ ($Q = \emptyset$):Condition 1 of Theorem 2 is satisfied by Lemma 1.
   - $\#(Q) = 1$ ($Q = \{q\}$): Condition 1 of Theorem 2 is satisfied by Lemma 3.
   - $\#(Q) = 2,3$: If elements of Q coincide with template elements marked "·" or "■", then point p is simple after Q is deleted by Lemmas 1 and 2. If an element of Q coincides with a template element marked "•", then all possible configurations are depicted in Figs. 16–20. It is easy to check that point p is simple after the deletion of Q. Thus Condition 1 of Theorem 2 is satisfied.
2. Condition 2 of Theorem 2 (i.e., no black component contained in a $2 \times 2 \times 2$ cube can be deleted completely) is satisfied by Lemma 4. □

7. Conclusions

In this paper we presented an advanced contour smoothing algorithm for reducing the noise sensitivity of 3D thinning algorithms and the associated new thinning scheme with iteration-by-iteration smoothing. An efficient and fairly general implementation method was also sketched. We proved that the proposed smoothing algorithm is topology preserving for (26, 6) pictures, hence it cannot alter the topological correctness of the applied thinning algorithms. We gave some examples to illustrate that the proposed thinning scheme can produce skeletons with less unwanted parts.

Acknowledgments

This research was supported by the TÁMOP-4.2.2/08/1/2008-0008 program of the Hungarian National Development Agency, the European Union and the European Regional Development Fund under the grant agreement TÁMOP-4.2.1/B-09/1/KONV-2010-0005, and the Grant CNK80370 of the National Office for Research and Technology (NKTH) & the Hungarian Scientific Research Fund (OTKA).

References

[1] B.R. Gomberg, P.K. Saha, H.K. Song, S.N. Hwang, F.W. Wehrli, Topological analysis of trabecular bone MR images, IEEE Transactions on Medical Imaging 19 (2000) 166–174.

[2] T. Itoh, Y. Yamaguchi, K. Koyamada, Fast isosurface generation using the volume thinning algorithm, IEEE Transactions on Visualization and Computer Graphics 7 (2001) 32–46.

[3] K. Palágyi, J. Tschirren, E.A. Hoffman, M. Sonka, Quantitative analysis of pulmonary airway tree structures, Computers in Biology and Medicine 36 (2006) 974–996.

[4] K. Siddiqi, S. Pizer (Eds.), Medial Representations – Mathematics, Algorithms and Applications, Computational Imaging and Vision, vol. 37, Springer, 2008.

[5] H. Sundar, D. Silver, N. Gagvani, S. Dickinson, Skeleton based shape matching and retrieval, in: Proc. Int. Conf. Shape Modeling and Applications, IEEE, 2003, pp. 130–139.

[6] M. Wan, Z. Liang, Q. Ke, L. Hong, I. Bitter, A. Kaufman, Automatic centerline extraction for virtual colonoscopy, IEEE Transactions on Medical Imaging 21 (2002) 1450–1460.

[7] D. Shaked, A. Bruckstein, Pruning medial axes, Computer Vision Image Understanding 69 (1998) 156–169.

[8] S. Svensson, G. Sanniti di Baja, Simplifying curve skeletons in volume images, Computer Vision and Image Understanding 90 (2003) 242–257.

[9] T. Ju, M.L. Baker, W. Chiu, Computing a family of skeletons of volumetric models for shape description, Computer-aided Design 39 (2007) 352–360.

[10] X. Bai, L.J. Latecki, L. Wen-Yu Liu, Skeleton pruning by contour partitioning with discrete curve evolution, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 449–462.

[11] W. Shen, X. Bai, R. Hu, H. Wang, L.J. Latecki, Skeleton growing and pruning with bending potential ratio, Pattern Recognition 44 (2011) 196–209.

[12] R.W. Hall, Parallel connectivity–preserving thinning algorithms, in: T.Y. Kong, A. Rosenfeld (Eds.), Topological Algorithms for Digital Image Processing, Machine Intelligence and Pattern Recognition, vol. 19, Elsevier Science, 1996, pp. 145–179.

[13] T.Y. Kong, A. Rosenfeld, Digital topology: introduction and survey, Computer Vision, Graphics, and Image Processing 48 (1989) 357–393.

[14] K. Palágyi, A. Kuba, A parallel 3D 12-subiteration thinning algorithm, Graphical Models and Image Processing 61 (1999) 199–221.

[15] D. Yu, H. Yan, An efficient algorithm for smoothing, linearization and detection of structural feature points of binary image contours, Pattern Recognition 30 (1997) 57–69.

[16] J. Hu, D. Yu, H. Yan, A multiple point boundary smoothing algorithm, Pattern Recognition Letters 19 (1998) 657–668.

[17] G. Taubin, Curve and surface smoothing without shrinkage, in: Proc. 5th Int. Conf. Computer Vision, ICCV'95 1995, pp. 852–857.

[18] M. Couprie, G. Bertrand, Topology preserving alternating sequential filter for smoothing two-dimensional and three-dimensional objects, Journal of Electronic Imaging 13 (2004) 720–730.

[19] K. Palágyi, A 3D fully parallel surface-thinning algorithm, Theoretical Computer Science 406 (2008) 119–135.

[20] K. Palágyi, G. Németh, Fully parallel 3D thinning algorithms based on sufficient conditions for topology preservation, in: Proc. 15th Int. Conf. Discrete Geometry for Computer Imagery, DGCI 2009, Lecture Notes in Computer Science, vol. 5810, Springer, Heidelberg, 2009, pp. 481–492.

[21] G. Németh, P. Kardos, K. Palágyi, Topology preserving parallel smoothing for 3D binary images, in: Proc. Int. Symposium of Computational Modeling of Objects Presented in Images: Fundamentals, Methods, and Applications, CompIMAGE'10, Lecture Notes in Computer Science, vol. 6026, Springer, Heidelberg, 2010, pp. 287–298.

[22] T.Y. Kong, On topology preservation in 2-d and 3-d thinning, International Journal of Pattern Recognition and Artificial Intelligence 9 (1995) 813–844.

[23] G. Malandain, G. Bertrand, Fast characterization of 3D simple points, in: Proc. 11th IEEE Internat. Conf. on Pattern Recognition, ICPR'92, 1992, pp. 232–235.

[24] K. Palágyi, A. Kuba, A 3D 6-subiteration thinning algorithm for extracting medial lines, Pattern Recognition Letters 19 (1998) 613–627.

[25] G. Bertrand, Z. Aktouf, A 3D thinning algorithm using subfields, in: Proc. SPIE Conf. on Vision Geometry III 2356, 1994, pp. 113–124.

[26] W.X. Gong, G. Bertrand, A simple parallel 3D thinning algorithm, in: Proc. 10th IEEE Internat. Conf. on Pattern Recognition, ICPR'90, 1990, pp. 188–190.

[27] G. Németh, P. Kardos, K. Palágyi, Topology preserving 2-subfield 3D thinning algorithms, in: Proc. 7th IASTED Int. Conf. Signal Processing, Pattern Recognition and Applications, SPPRA 2010, 2010, pp. 310–316.

[28] A. Manzanera, T.M. Bernard, F. Pretêux, B. Longuet, Medial faces from a concise 3D thinning algorithm, in: Proc. 7th IEEE Internat. Conf. Computer Vision, ICCV'99, vol. 1, 1999, pp. 337–343.

[29] G. Németh, P. Kardos, K. Palágyi, Topology preserving 3D thinning algorithms using four and eight subfields, in: Proc. Int. Conf. on Image Analysis and Recognition, ICIAR 2010, Lecture Notes in Computer Science, vol. 6111, Springer, Heidelberg, 2010, pp. 316–325.