Parallel 3D 12-Subiteration Thinning Algorithms Based on Isthmuses

Kálmán Palágyi

Department of Image Processing and Computer Graphics,
University of Szeged, Hungary
palagyi@inf.u-szeged.hu

Abstract. Thinning is an iterative object reduction to obtain skeleton-like shape features of volumetric binary objects. Conventional thinning algorithms preserve endpoints to provide important geometric information relative to the object to be represented. An alternative strategy is also proposed that accumulates isthmuses (i.e., generalization of curve and surface interior points as skeletal elements). This paper presents two parallel isthmus-based 3D thinning algorithms that are capable of producing centerlines and medial surfaces. The strategy which is used is called subiteration-based or directional: each iteration step is composed of 12 subiterations each of which are executed in parallel. The proposed algorithms make efficient implementation possible and their topological correctness is guaranteed.

Keywords: Object Recognition, Shape Representation, Discrete Geometry, Digital Topology, Thinning.

1 Introduction

Skeleton is a region-based shape descriptor which represents the general shape of objects. 3D skeleton-like shape features (i.e., centerlines and medial surfaces) play important role in various applications in image processing, computer vision, and pattern recognition [14].

A fairly illustrative definition of the skeleton is given using the prairie-fire analogy: the object boundary is set on fire, and the skeleton is formed by the loci where the fire fronts meet and extinguish each other. Thinning is a digital simulation of the fire front propagation [5]: the border points that satisfy certain topological and geometric constraints are deleted in iteration steps. The entire process is repeated until stability is reached.

Most of the existing thinning algorithms are parallel, since the fire front propagation is by nature parallel. Those algorithms delete some object points in a binary image simultaneously [3]. Thinning has a major advantage over the alternative 3D skeletonization methods: it can produce both skeleton-like shape features. Surface-thinning algorithms can extract medial surfaces and curve-thinning algorithms can produce centerlines. General 3D objects can be represented by their medial surfaces, and centerlines are usually extracted from tubular structures.

G. Bebis et al. (Eds.): ISVC 2013, Part I, LNCS 8033, pp. 87–98, 2013.

Conventional 3D thinning algorithms preserve some curve-endpoints or surface-endpoints that provide relevant geometrical information with respect to the shape of the object. Bertrand and Couprie proposed an alternative approach by accumulating some curve/surface interior points that are called isthmuses [2]. Characterizations of these isthmuses (for curve-thinning and surface-thinning) were defined first by Bertrand and Aktouf [1]. There are dozens of endpoint-based 3D thinning algorithms, but just a few ones use the isthmus-based thinning scheme [1,2,7,12].

In this paper a curve-thinning algorithm and a surface-thinning algorithm are presented. Both 3D parallel thinning algorithms accumulate isthmuses in each thinning phase as elements of the final shape features. They use subiteration-based (or directional) strategy: each iteration step is composed of a number of subiterations where only border points of a certain kind can be deleted in each subiteration [3,11]. The new algorithms are derived from endpoint-preserving 3D parallel 12-subiteration thinning algorithms proposed by Palágyi and Kuba [8]. It is illustrated that the isthmus-based algorithms produce "more reliable" results with fewer skeletal points than the original endpoint-based algorithms do. The topological correctness of the new algorithms is proved.

2 Basic Notions and Results

Some concepts of digital topology and their key results will be given below as they will be needed later on. The basic concepts of digital topology are applied as reviewed in [5].

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

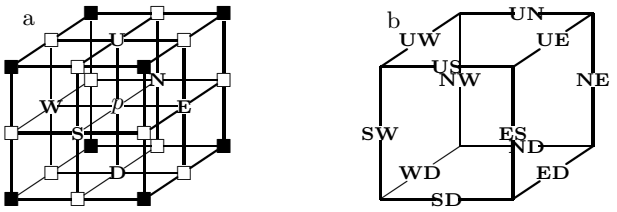


Fig. 1. Frequently used adjacency relations in $\mathbb{Z}^3$ (a). The set $N_6(p)$ contains point p and the six points marked **U**, **D**, **N**, **E**, **S**, and **W**. The set $N_{18}(p)$ contains $N_6(p)$ and the twelve points marked "□". The set $N_{26}(p)$ contains $N_{18}(p)$ and the eight points marked "■". The 12 possible non-opposite pairs of points in $N_6^*(p)$ (b).

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

$i = 1, \ldots, n$. 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. A set of points X is *j-connected* in the set of points $Y \supseteq X$ if any two points in X are j-connected in Y.

A *3D binary* $(26, 6)$ *digital picture* $\mathcal{P}$ is a quadruple $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ [5]. Each element of $\mathbb{Z}^3$ is said to be a *point* of $\mathcal{P}$. Each point in $B \subseteq \mathbb{Z}^3$ is called a *black point* and has a value of 1 assigned to it. Each point in $\mathbb{Z}^3 \backslash B$ is known as a *white point* and has a value of 0. A picture $(\mathbb{Z}^3, 26, 6, B)$ is called *finite* if the set B contains finitely many points. An *object* is a maximal 26-connected set of black points, while a *white component* is a maximal 6-connected set of white points. In a finite picture there is a unique infinite white component, which is called the *background*. A finite white component is said to be a *cavity*.

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 border point is said to be a **U**-*border point* if the point marked **U** in Fig. 1a is white. We can define **D**-, **N**-, **E**-, **S**-, and **W**-border points in the same way.

There are three kinds of *opposite* (unordered) pair of points in $N_6^*(p)$ denoted by **UD**, **NS**, and **EW**. The twelve possible *non–opposite* pairs of points in $N_6^*(p)$ are denoted by **US**, **NE**, **DW**, **SE**, **UW**, **DN**, **SW**, **UN**, **DE**, **NW**, **UE**, and **DS**. These can be associated with the twelve edges of a cube, see Fig. 1b. A border point is called a **US**-*border point* if it is a **U**-border point or an **S**-border point. The remaining 11 kinds of border points corresponding to the other non–opposite pairs can be defined in the same way.

A *reduction* transforms a binary picture only by changing some black points to white ones (which is referred to as the deletion of black points). A reduction is *not* topology-preserving [4] if any object in the input picture is split (into several ones) or is completely deleted, any cavity in the input picture is merged with the background or another cavity, or a cavity is created where there was none in the input picture. There is an additional concept called *hole* (which doughnuts have) in 3D pictures [5]. Topology preservation implies that eliminating or creating any hole is not allowed.

A black point is *simple* in a $(26, 6)$ picture if and only if its deletion is a topology-preserving reduction [5]. A useful characterization of simple points on $(26, 6)$ pictures is stated by Malandain and Bertrand as follows:

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

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

Based on Theorem 1, the simplicity of a point p can be decided by examining the set $N_{26}^*(p)$.

Reductions delete a set of black points and not just a single simple point. Palágyi and Kuba proposed the following sufficient conditions for 3D reductions to preserve topology.

Theorem 2. [8] *Let $\mathcal{R}$ be a reduction. Let p be any black point in any picture $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ such that p is deleted by $\mathcal{R}$. Let $\mathcal{Q}$ be the family of all the sets of $Q \subseteq (N_{18}(p) \backslash \{p\}) \cap B$ such that $q_1 \in N_{18}(q_2)$, for any $q_1 \in Q$ and $q_2 \in Q$. $\mathcal{R}$ is topology-preserving for $(26, 6)$ pictures if all of the following conditions hold:*

1. *p is a simple point in $(\mathbb{Z}^3, 26, 6, B \backslash Q)$ for any Q in $\mathcal{Q}$.*
2. *No object contained in a $2 \times 2 \times 2$ cube can be deleted completely by $\mathcal{R}$.*

3 Endpoint-Preserving 12-Subiteration Thinning Algorithms

Palágyi and Kuba developed two endpoint-preserving parallel 3D 12-subiteration thinning algorithms [8]. Let us denote these algorithms as **3D-12S-**E_C and **3D-12S-**E_S. Algorithm **3D-12S-**E_C is capable of producing centerlines by preserving curve-endpoints of type E_C, and **3D-12S-**E_S is a surface-thinning algorithm that does not delete surface-endpoints of type E_S. The considered types of endpoints are defined as follows:

Definition 1. [8] *A black point p in picture $(\mathbb{Z}^3, 26, 6, B)$ is a* curve-endpoint of type E_C *if the set $N^*_{26}(p) \cap B$ contains exactly one point.*

Definition 2. [8] *A black point p in picture $(\mathbb{Z}^3, 26, 6, B)$ is a* surface-endpoint of type E_S *if the set $N^*_6(p)$ contains at least one opposite pair of white points.*

Note that each curve-endpoint of type E_C is simple and a surface-endpoint of type E_S.

Existing algorithms **3D-12S-**E_C and **3D-12S-**E_S are described by Algorithm 1.

Algorithm 1. Algorithm **3D-12S-**$\mathcal{E}$ $\quad (\mathcal{E} \in \{E_C, E_S\})$

```
1: Input: picture (Z^3, 26, 6, X)
2: Output: picture (Z^3, 26, 6, Y)
3: Y = X
4: repeat
5:    // one iteration step
6:    for each d ∈ {US, ..., DS} do
7:       // subiteration for deleting some d-border points
8:       D(d) = { p | p is d-E-deletable in Y}
9:       Y  = Y \ D(d)
10:   end for
11: until  D(US) ∪ ... ∪ D(DS) = ∅
```

Note that choosing another order of the 12 types of border points yields another algorithm. Palágyi and Kuba proposed the following ordered list of

the border points associated with the 12 subiterations of Algorithm 1 [8]: $\langle \mathbf{US}, \mathbf{NE}, \mathbf{DW},\ \mathbf{SE}, \mathbf{UW}, \mathbf{ND},\ \mathbf{SW}, \mathbf{UN}, \mathbf{DE},\ \mathbf{NW}, \mathbf{UE}, \mathbf{DS} \rangle$.

In the first subiteration, all **US**-$\mathcal{E}$-*deletable* points are deleted simultaneously, and all **DS**-$\mathcal{E}$-*deletable* points are deleted in the last (i.e., the 12th) subiteration at a time ($\mathcal{E} \in \{E_C, E_S\}$). Deletable points are given by a set of $3 \times 3 \times 3$ matching templates. A black point is deletable if at least one template in the corresponding set of templates matches it. Templates are usually described by three kinds of elements, "●" (black), "○" (white), and "." ("don't care"), where "don't care" matches either black or white point in a given picture. In order to reduce the number of templates Palágyi and Kuba use additional notations [8].

Deletable points in the first subiteration of the curve-thinning algorithm **3D-12S-**E_C (i.e., **US-**E_C-*deletable* points) are given by the set of 14 matching templates $\mathcal{T}_C = \{\mathbf{C1}, \ldots, \mathbf{C14}\}$ depicted in Fig. 2. Similarly, deletable points in the first subiteration of the surface-thinning algorithm **3D-12S-**E_S (i.e., **US-**E_S-*deletable* points) are given by the set of 6 matching templates $\mathcal{T}_S = \{\mathbf{S1}, \mathbf{S2}, \mathbf{C7}, \mathbf{C8}, \mathbf{C9}, \mathbf{C10}\}$, see Figs. 2 and 3. Deletable points of the remaining 11 subiterations can be obtained by proper rotations and/or reflections of the templates associated with the first subiteration.

It is easy to see that all curve-endpoints of type E_C (see Def. 1) are preserved by curve-thinning algorithm **3D-12S-**E_C and surface-thinning algorithm **3D-12S-**E_S never deletes any surface-endpoint of type E_S (see Def. 2).

4 Isthmus-Based 12-Subiteration Thinning Algorithms

In this section two isthmus-based 3D parallel 12-subiteration thinning algorithms are presented. The new curve-thinning and surface-thinning algorithms use the following characterizations of isthmuses.

Definition 3. [1] *A border point p in a picture $(\mathbb{Z}^3, 26, 6, B)$ is an* I_C-isthmus *(for curve-thinning) if the set $N^*_{26}(p) \cap B$ contains more than one* 26*–component (i.e., Condition 1 of Theorem 1 is violated).*

Definition 4. [1] *A border point p in a picture $(\mathbb{Z}^3, 26, 6, B)$ is an* I_S-isthmus *(for surface-thinning) if p is not a simple point (i.e., Condition 1 of Theorem 1 or Condition 3 of Theorem 1 is violated).*

It can be stated that no isthmus point is simple and the considered characterizations of isthmuses depend on the set $N^*_{26}(p)$ for a point p in question.

The scheme of the proposed two isthmus-based thinning algorithms **3D-12S-**I_C and **3D-12S-**I_S is sketched in Algorithm 2.

In each subiteration of the new algorithms, isthmuses (i.e., some border points that are not simple ones) are dynamically detected and accumulated in a constraint set I. In the first subiteration of both thinning algorithms, all **US**-*deletable* points are deleted simultaneously, and all **SD**-*deletable* points are deleted in the last (i.e., the 12th) subiteration. **US**-deletable points are given by the set of 16 matching templates $\mathcal{T}_I = \{\mathbf{I1}, \mathbf{I2}, \mathbf{I3}, \mathbf{C4}, \ldots, \mathbf{C14}, \mathbf{I15}, \mathbf{I16}\}$

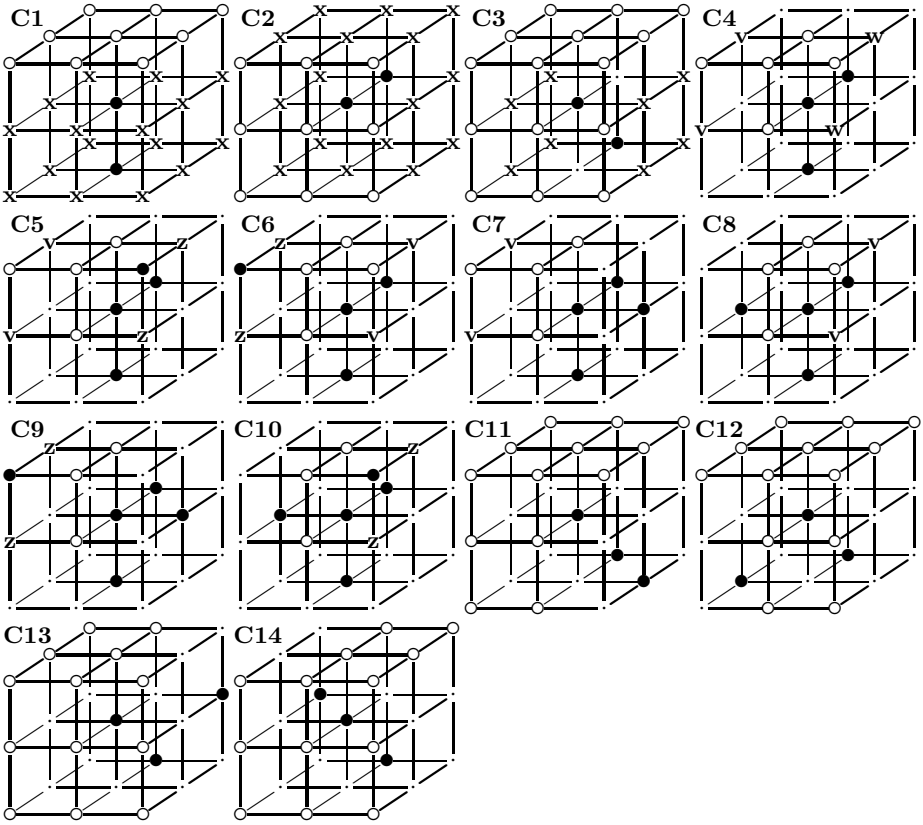


Fig. 2. The set of templates $\mathcal{T}_C$ assigned to the first subiteration of the curve-thinning algorithm **3D-12S-**E_C. Notations: at least one position marked "**x**" matches a black point; at least one position marked "**v**" matches a white point; at least one position marked "**w**" matches a white point; two positions marked "**z**" match different points (one of them matches a black point and the other one matches a white one).

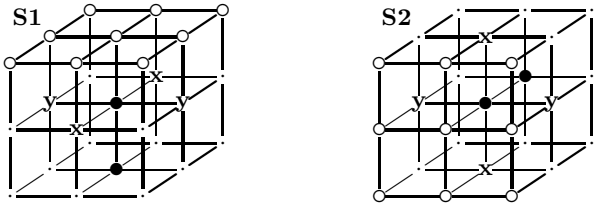


Fig. 3. Two special templates in the set $\mathcal{T}_S$ that are assigned to the first subiteration of the surface-thinning algorithm **3D-12S-**E_S. Notations: at least one position marked "**x**" matches a black point; at least one position marked "**y**" matches a black point.

Algorithm 2. Algorithm**3D-12S-$\mathcal{I}$** $(\mathcal{I} \in \{I_C, I_S\})$

Input: picture $(\mathbb{Z}^3, 26, 6, X)$
Output: picture $(\mathbb{Z}^3, 26, 6, Y)$
$Y = X$
$I = \emptyset$
repeat
  // one iteration step
  for each $d \in \{\mathbf{US}, \ldots, \mathbf{DS}\}$ **do**
    // subiteration for deleting some d-border points
    $I = I \cup \{\ p \mid p \in Y \setminus I$ and p is an $\mathcal{I}$-isthmus $\}$
    $D(d) = \{\ p \mid p \in Y \setminus I$ and p is *d-deletable* in $Y\}$
    $Y\ = Y \setminus D(d)$
  end for
until $D(\mathbf{US}) \cup \ldots \cup D(\mathbf{DS}) = \emptyset$

depicted in Figs. 2 and 4. Deletable points of the other 11 subiterations can be obtained by proper rotations and/or reflections of the templates assigned to the first subiteration.

It can be readily seen that – due to the new templates **I1**, **I2**, **I3**, **I15**, and **I16**, (see Fig. 4) and their rotated and reflected versions associated with the other 11 subiterations – all curve-endpoints of type E_C (with the exception of objects that are formed by two endpoints) and some (simple) surface-endpoints of type E_S are deleted by an iteration step of the proposed algorithms **3D-12S-I_C** and **3D-12S-I_S**.

The topological correctness of the new isthmus-based algorithms is proved in Section 6.

5 Results and Implementation

In experiments the existing endpoint-based and the proposed isthmus-based algorithms were tested on various synthetic and natural objects. Due to the lack of space, here we can present just three illustrative examples, see Figs. 5-7. The numbers in parentheses are the counts of object points in the produced skeleton-like shape features.

Thanks to the isthmus-based approach, the proposed algorithms (**3D-12S-I_C** and **3D-12S-I_S**) can produce less unwanted side branches and surface patches than the conventional endpoint-ones (**3D-12S-E_C** and **3D-12S-E_S**) do. Note that each skeletonization technique (including thinning) is rather sensitive to coarse object boundaries. The false segments included by the produced skeleton-like shape features can be removed by a pruning process (i.e., a post-processing step) [13].

One may think that the proposed algorithms are time consuming and it is rather difficult to implement them. In [10], Palágyi proposed a fairly general framework that can be used for parallel 3D thinning algorithms [11] and some sequential ones as well [9]. That efficient method uses pre-calculated look-up-tables

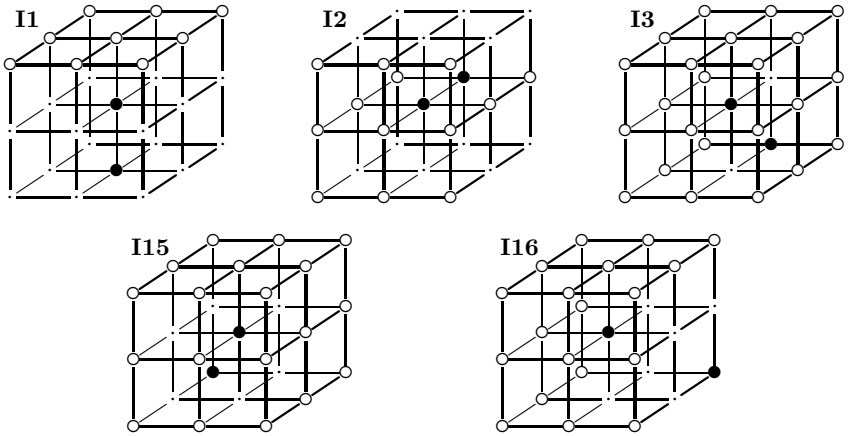


Fig. 4. The three modified and the two new templates in the set $\mathcal{T}_I$ that is assigned to the first subiteration of the proposed isthmus-based thinning algorithms **3D-12S-**I_C and **3D-12S-**I_S. The modified templates **I1**, **I2**, and **I3** are derived from templates **C1**, **C2**, and **C3** (see Fig. 2), respectively. All these five templates match some curve-endpoints of type E_C.

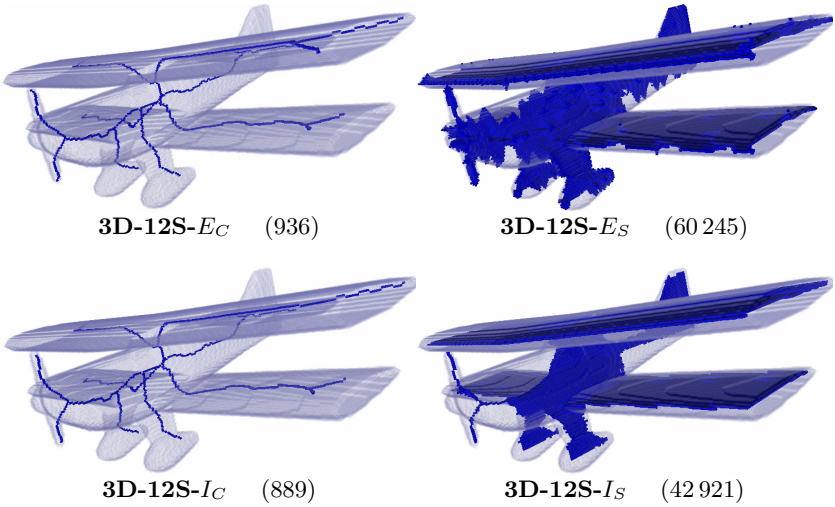


Fig. 5. The skeleton-like shape features produced by the existing endpoint-preserving algorithms and the proposed isthmus-based algorithms superimposed on a 217×304×98 3D image of a biplane containing 656 424 object points

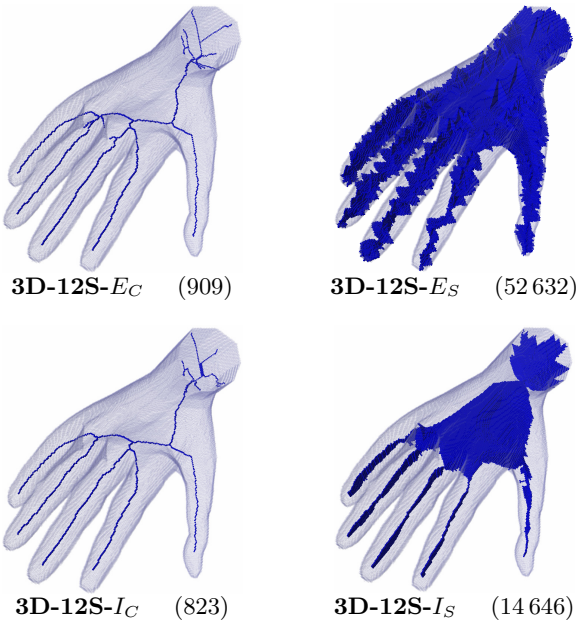


Fig. 6. Skeleton-like shape features produced by the existing endpoint-preserving algorithms and the proposed isthmus-based algorithms superimposed on a $174 \times 103 \times 300$ 3D image of a hand containing 865 941 object points

to encode the deletion rules of the thinning algorithm to be implemented. In addition, two lists are used to speed up the process: one for storing the border-points in the current picture (since thinning can only delete border-points, thus the repeated scans/traverses of the entire array storing the picture can be avoided); the other list is to store all deletable points in the current phase of the process. At each iteration, the deletable points are found and deleted, and the list of border points is updated accordingly. The algorithm terminates when no further update is required. To implement the proposed isthmus-based 3D 12-subiteration thinning algorithms we use three look-up-tables, one for detecting the **US**-*deletable* points (see Algorithm 2) and two additional ones to encode I_C-isthmus and I_S-isthmus points, respectively. Note that deletable points for the other 11 subiterations can be identified by the look-up-table associated with the first subiteration by using the proper permutations of the elements in the set $N_{26}^*(p)$ for a point p in question. Since **US**-*deletable* points are given by a set of $3 \times 3 \times 3$ matching templates (see Fig. 4) and the considered characterizations of isthmuses can be decided by investigating the $3 \times 3 \times 3$ neighborhood of the point is question, each pre-calculated look-up-table has 2^{26} entries of 1 bit in size. It is not hard to see that both look-up-tables require just 8 megabytes of storage space in memory.

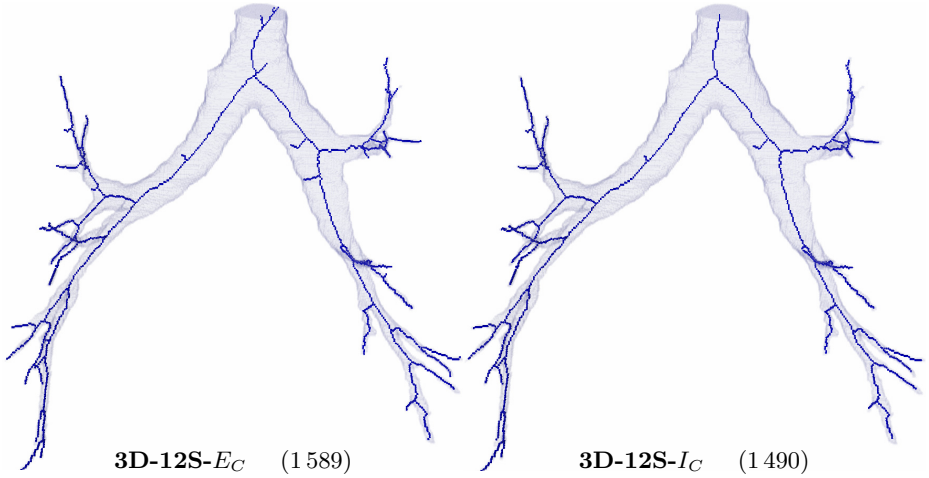


Fig. 7. Centerlines produced by the existing endpoint-preserving curve-thinning algorithm and the proposed isthmus-based curve-thinning algorithm superimposed on a $512 \times 512 \times 333$ 3D image of a segmented human airway tree containing 272 901 object points.

By adapting the efficient implementation method, our algorithms can be well applied in practice: they are capable of producing skeleton-like shape features from large 3D pictures containing 1 000 000 object points within half a second on a standard PC.

6 Verification

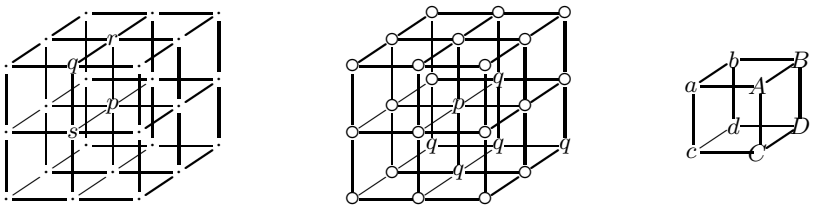
Now we will show that the proposed algorithms are topologically correct. It is sufficient to prove that reduction given by the set of matching templates $\mathcal{T}_I$ (see Figs. 2 and 4). Let us state some properties of the **US**-*deletable* points (see Algorithm 2).

Proposition 1. *Let us consider the configuration depicted in Fig. 8a. If black point p is* **US**-*deletable, then*

- *point q is white,*
- *at least one point in $\{r, s\}$ is white,*
- *if point r is black, then p can be deleted only by template* **I2** *in set $\mathcal{T}_I$, and*
- *if point s is black, then p can be deleted only by template* **I1** *in set $\mathcal{T}_I$.*

Proposition 2. *Let us consider the configuration depicted in Fig. 8b. If black point p is* **US**-*deletable, then at least one point marked q is black.*

These properties follow from an examination of the templates in $\mathcal{T}_I$.



a b c

Fig. 8. Configurations associated with Proposition 1 (a) and Proposition 2 (b). A $2 \times 2 \times 2$ cube corresponding to Condition 2 of Theorem 2 (c).

Lemma 1. *Each* **US**-*deletable point is simple of* $(26, 6)$ *pictures.*

We need to show that all conditions of Theorem 1 are satisfied. It is obvious by careful examination of the templates in $\mathcal{T}_I$.

We are now ready to state the main theorem.

Theorem 3. *Both algorithms* **3D-12S-**I_C *and* **3D-12S-**I_S *are topology-preserving for* $(26, 6)$ *pictures.*

Proof. (Sketch) It is sufficient to prove that the reduction that deletes all **US**-*deletable* points from any $(26, 6)$ pictures is topology-preserving. We need to show that both conditions of Theorem 2 are satisfied.

1. Let us consider Condition 1 of Theorem 2. If $Q = \emptyset$, then it holds by Lemma 1. If $Q \neq \emptyset$, then the given point p in question is simple after the deletion of Q by Proposition 1.
2. Let us consider an object $\mathcal{O}$ that is contained in the $2 \times 2 \times 2$ cube as it is illustrated in Fig. 8c. Then the following statements hold by Proposition 2.
   - If $a \in \mathcal{O}$ and it is **US**-*deletable*, then b, c, d, or D is in $\mathcal{O}$.
   - If $A \in \mathcal{O}$ and it is **US**-*deletable*, then B, C, D, or d is in $\mathcal{O}$.
   - If $b \in \mathcal{O}$ and it is **US**-*deletable*, then $d \in \mathcal{O}$.
   - If $B \in \mathcal{O}$ and it is **US**-*deletable*, then $D \in \mathcal{O}$.
   - If $c \in \mathcal{O}$ and it is **US**-*deletable*, then $d \in \mathcal{O}$.
   - If $C \in \mathcal{O}$ and it is **US**-*deletable*, then $D \in \mathcal{O}$.
   - If $d \in \mathcal{O}$, then it is not **US**-*deletable*.
   - If $D \in \mathcal{O}$, then it is not **US**-*deletable*.

   Since $\mathcal{O}$ contains at least one point that is not **US**-*deletable*, no object contained in a $2 \times 2 \times 2$ cube can be deleted completely.

The reduction given by the set of templates $\mathcal{T}_I$ fulfills both conditions of Theorem 2. Hence, the first subiteration of the proposed thinning algorithms is topology-preserving. It can be proved for the other 11 subiterations in a similar way. The entire algorithms are topology-preserving, since they are composed of topology-preserving reductions. □

Acknowledgements. This research was supported by the grant CNK80370 of the National Office for Research and Technology (NKTH) & the Hungarian Scientific Research Fund (OTKA).

References

1. Bertrand, G., Aktouf, Z.: A 3D thinning algorithm using subfields. In: SPIE Proc. of Conf. on Vision Geometry, pp. 113–124 (1994)
2. Bertrand, G., Couprie, M.: Transformations topologiques discrètes. In: Coeurjolly, D., Montanvert, A., Chassery, J. (eds.) Géométrie Discrète et Images Numériques, pp. 187–209. Hermès Science Publications (2007)
3. Hall, R.W.: Parallel connectivity-preserving thinning algorithms. In: Kong, T.Y., Rosenfeld, A. (eds.) Topological Algorithms for Digital Image Processing, pp. 145–179. Elsevier Science B. V.(1996)
4. Kong, T.Y.: On topology preservation in 2–d and 3–d thinning. International Journal of Pattern Recognition and Artificial Intelligence 9, 813–844 (1995)
5. Kong, T.Y., Rosenfeld, A.: Digital topology: Introduction and survey. Computer Vision, Graphics, and Image Processing 48, 357–393 (1989)
6. Malandain, G., Bertrand, G.: Fast characterization of 3D simple points. In: Proc. 11th IEEE Internat. Conf. on Pattern Recognition, ICPR 1992, pp. 232–235 (1992)
7. Németh, G., Palágyi, K.: 3D parallel thinning algorithms based on isthmuses. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P., Zemčík, P. (eds.) ACIVS 2012. LNCS, vol. 7517, pp. 325–335. Springer, Heidelberg (2012)
8. Palágyi, K., Kuba, A.: A parallel 3D 12-subiteration thinning algorithm. Graphical Models and Image Processing 61, 199–221 (1999)
9. Palágyi, K., Tschirren, J., Hoffman, E.A., Sonka, M.: Quantitative analysis of pulmonary airway tree structures. Computers in Biology and Medicine 36, 974–996 (2006)
10. Palágyi, K.: A 3D fully parallel surface-thinning algorithm. Theoretical Computer Science 406, 119–135 (2008)
11. Palágyi, K., Németh, G., Kardos, P.: Topology preserving parallel 3D thinning algorithms. In: Brimkov, V.E., Barneva, R.P. (eds.) Digital Geometry Algorithms. Theoretical Foundations and Applications to Computational Imaging, pp. 165–188. Springer (2012)
12. Raynal, B., Couprie, M.: Isthmus-based 6-directional parallel thinning algorithms. In: Debled-Rennesson, I., Domenjoud, E., Kerautret, B., Even, P. (eds.) DGCI 2011. LNCS, vol. 6607, pp. 175–186. Springer, Heidelberg (2011)
13. Shaked, D., Bruckstein, A.: Pruning medial axes. Computer Vision Image Understanding 69, 156–169 (1998)
14. Siddiqi, K., Pizer, S. (eds.): Medial representations – Mathematics, algorithms and applications. Computational Imaging and Vision, vol. 37. Springer (2008)