A Sequential 3D Curve-Thinning Algorithm Based on Isthmuses

Kálmán Palágyi

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

Abstract. Curve-thinning is a frequently applied technique to obtain centerlines from volumetric binary objects. Conventional curve-thinning algorithms preserve endpoints to provide important geometric information relative to the objects. An alternative strategy is also proposed that accumulates isthmuses (i.e., generalization of curve interior points as elements of the centerlines). This paper presents a computationally efficient sequential isthmus-based 3D curve-thinning algorithm.

1 Introduction

Thinning [3,6] is a layer-by-layer erosion: some border points that satisfy certain topological and geometric constraints are deleted in iteration step. The entire process is repeated until stability is reached. Thinning is a frequently used approach to obtain skeleton-like shape features. 3D skeleton-like shape features (i.e., centerlines, medial surfaces, and topological kernels) play important role in various applications in image processing and pattern recognition [14,15].

Curve-thinning algorithms are used to extract centerlines, surface-thinning algorithms produce medial surfaces, while kernel-thinning algorithms are capable of extracting topological kernels (i.e., minimal sets of points that are topologically equivalent [6] to the original objects). Medial surfaces are usually extracted from general shapes, tubular structures can be represented by their centerlines, and topological kernels are useful in topological description. Tubular structures (e.g., arterial and venous systems, intrathoracic airways, and gastrointestinal tract) are frequently found in living organisms. Centerlines as 1D structures can serve as viewpoint trajectory for navigation purposes in virtual angioscopy, bronchoscopy, or colonoscopy, and help us to generate formal structures for the forthcoming analysis and measurements [9,16,17].

In an iteration step, sequential thinning algorithms traverse the border points in a binary picture, and consider a single point for possible deletion, while parallel algorithms can delete a set of border points simultaneously [3]. Conventional 3D curve-thinning algorithms preserve some curve-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 interior points that are called isthmuses [2]. These isthmuses were characterized

G. Bebis et al. (Eds.): ISVC 2014, Part II, LNCS 8888, pp. 406–415, 2014.

first by Bertrand and Aktouf [1]. There are numerous endpoint-based 3D curve-thinning algorithms, but only a few ones use the isthmus-based thinning scheme [1,2,5,8,11,12]. Note that Kardos and Palágyi [5] proposed a sequential isthmus-based 3D curve-thinning algorithm [5]. That algorithm is time consuming, since in each iteration step begins a labeling phase, then the deletion rule is evaluated in a labeled (non-binary) picture.

In this paper presents a computationally efficient 3D curve-thinning algorithm. The new algorithm accumulates isthmuses in each thinning phase as elements of the final centerline. It uses subiteration-based 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 [3,10]. The new algorithm considers six subiterations associated with the six main directions in 3D. It is illustrated that the proposed isthmus-based algorithm produces "more reliable" results with fewer skeletal points than the existing endpoint-based 3D curve-thinning algorithm proposed by Palágyi et al. [9].

2 Basic Notions

In this paper, we use the fundamental concepts of digital topology as reviewed by Kong and Rosenfeld [6] and Palágyi et al. [10].

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. 1.

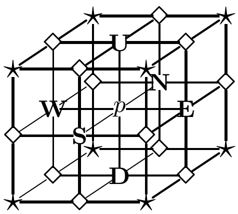


Fig. 1. The considered adjacency relations on $\mathbb{Z}^3$. The set $N_6(p)$ contains point p and the six points marked **U**, **D**, **N**, **E**, **S**, and **W**. The set $N_{18}(p)$ contains $N_6(p)$ and the twelve points marked "$\diamond$". The set $N_{26}(p)$ contains $N_{18}(p)$ and the eight points marked "$\bigstar$".

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* is a quadruple $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$. 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 a value of 1 is assigned to it. Each point in $\mathbb{Z}^3 \backslash B$ is said to be a *white point* and has a value of 0. A picture $(\mathbb{Z}^3, 26, 6, B)$ is called *finite* if 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.

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. 1 is white. We can define **D**-, **N**-, **E**-, **S**-, and **W**-border points in the same way. A black point is called an *interior point* if it is not a border point.

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 topology-preserving [6] if any object of the input picture contains exactly one object of the output picture, and each white component of the output picture contains exactly one white component of the input picture. There is an additional concept called *tunnel* (which doughnuts have) in 3D pictures [6]. Topology preservation implies that eliminating or creating any tunnel is not allowed.

A black point is *simple* in a $(26, 6)$ picture if and only if its deletion is a topology-preserving reduction [6]. A useful characterization of simple points on $(26, 6)$ pictures is stated by Malandain and Bertrand [7] as follows:

Theorem 1. *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 $N^*_{26}(p) \cap B$ 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 $N^*_{26}(p)$. We can state that simple points are border points by Condition 2 of Theorem 1.

Endpoint-based 3D curve-thinning algorithms preserve curve-endpoints. The following characterization of curve-endpoints is generally considered:

Definition 1. *A black point p in picture $(\mathbb{Z}^3, 26, 6, B)$ is a* curve-endpoint *if the set $N^*_{26}(p) \cap B$ contains exactly one point (i.e., p is 26-adjacent to exactly one further black point).*

Note that each curve-endpoint is simple.

Bertrand and Couprie proposed an alternative approach for curve-thinning by accumulating some curve interior points that are called isthmuses [2]. Curve-isthmuses were characterized first by Bertrand and Aktouf [1]:

Definition 2. *A border point p in a picture $(\mathbb{Z}^3, 26, 6, B)$ is an* curve-isthmus *if the set $N^*_{26}(p) \cap B$ contains more than one 26–component.*

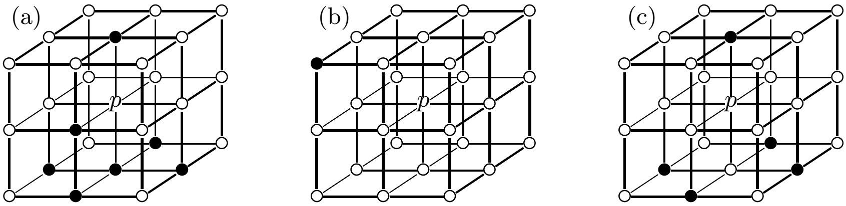


Fig. 2. Sets $N_{26}^*(p)$ in which the central point p is simple (a), curve-end (b), and curve-isthmus (c). Note that each curve-endpoint is simple, and each curve-isthmus point is not simple.

These curve-isthmuses are not simple points since Condition 1 of Theorem 1 is violated. Note that the considered characterization of curve-isthmuses depends on the set $N_{26}^*(p)$ for a point p in question.

Figure 2 presents examples of simple, curve-end, and curve-isthmus points.

3 An Isthmus-Based 3D Curve-Thinning Algorithm

In this section the new isthmus-based 3D sequential 6-subiteration curve-thinning algorithm is presented, and its efficient implementation is outlined. The scheme of the proposed algorithm **I-3D-C-T** is sketched in Algorithm 1.

The kernel of the **repeat** cycle corresponds to one iteration step of the thinning process. Each iteration step is decomposed into six successive subiterations according to the six main directions in 3D, and each subiteration consists of two phases. At first the border points of the actual type that are simple points are marked as potential deletable points, and the new curve-isthmus points are added to the previously detected isthmuses. During the second phase, a marked point is deleted if it remains simple after the deletion of some previously visited marked points.

Since the sequential algorithm **I-3D-C-T** may delete just one simple point at a time, it is topology preserving for $(26, 6)$ pictures [6].

One may think that the proposed algorithm is time-consuming and it is rather difficult to implement it. That is why Algorithm 2 outlines the efficient implementation of algorithm **I-3D-C-T**. Note that similar implementation was proposed by Palágyi et al. [10] for parallel thinning algorithms.

The input of Algorithm 2 is array A which stores the $(26, 6)$ picture to be thinned. In input array A, the value "1" corresponds to black points and the value "0" is assigned to white ones. According to the proposed scheme, the input and the output pictures can be stored in the same array, hence array A will contain the produced centerline.

Algorithm 2 uses two lists to speed up the process: *border_list* stores the border points in the current picture (hence the repeated scans of the entire array A are avoided); *potentially_deletable_list* is to collect all potentially deletable points in the current subiteration. (Note that *potentially_deletable_list* is a sublist of

Algorithm 1. Algorithm **I-3D-C-T**

```
Input: picture (ℤ³, 26, 6, X)
Output: picture (ℤ³, 26, 6, Y)
Y = X
I = ∅   // Initialize the set of isthmuses
repeat
    // One iteration step
    foreach direction d ∈ {U,N,E,S,W,D} do
        // Subiteration according to the deletion direction d
        // Phase 1
        Z = ∅   // Initialize the set of potential deletable points
        foreach point p ∈ Y \ I do
            if point p is d-border and simple in (ℤ³, 26, 6, Y) then
                Z = Z ∪ {p}   // Candidate found
            if point p is curve-isthmus in (ℤ³, 26, 6, Y) then
                I = I ∪ {p}   // Isthmus found
        // Phase 2
        foreach point p ∈ Z do
            if point p is simple (ℤ³, 26, 6, Y) then
                Y = Y \ {p}   // Deletion
until no changes occur;
```

border_list.) In order to avoid storing more than one copy of a border point in *border_list*, array A represents a four-colour picture:

- a value of "0" corresponds to white points,
- a value of "1" is assigned to (black) interior points,
- a value of "2" corresponds to (black) border points in the actual picture (i.e., elements of *border_list*), and
- a value of "3" is assigned to the detected and accumulated curve-isthmus points.

First, the original picture is scanned and all the border points are inserted into the list *border_list*. Then the thinning process itself is performed. The number of deleted points withing an iteration step is stored in the variable *number_of_deleted_points*. If a point p is deleted, then *border_list* is updated since all interior points that are 6-adjacent to p become border points. The algorithm terminates when stability is reached (i.e., *number_of_deleted_points*$= 0$). Then all points having a nonzero value belong to the produced centerline.

We can use two pre–calculated look-up-tables to encode simple and curve-isthmus points. Simple points in $(26, 6)$ pictures and the considered curve-isthmus points (see Definition 2) can be locally characterized; both properties for a point p can be decided by examining the set $N^*_{26}(p)$ that contains 26 points. Hence 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 8–8 megabytes of storage space in memory.

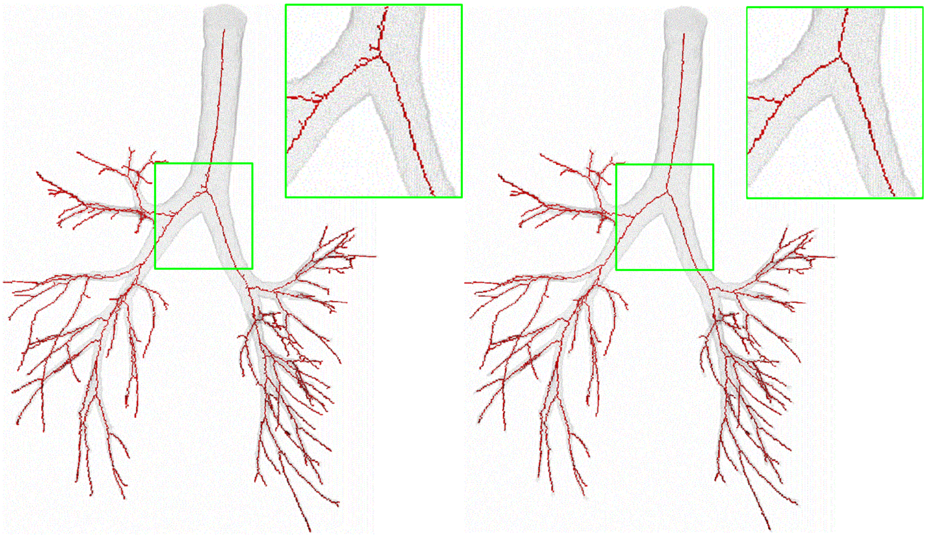
Algorithm 2. Efficient Implementation of Algorithm **I-3D-C-T**

```
Input: array A storing the 3D binary picture to be thinned
Output: array A containing the picture with the produced centerline
// Collect border points by a single scan of array A
border_list = < empty list >
foreach element p = (x, y, z) in array A do
    if p is a border point then
        border_list = border_list + < p >
        A[x, y, z] = 2
// Thinning process
repeat
    // One iteration step
    number_of_deleted_points = 0
    foreach direction d ∈ {U,N,E,S,W,D} do
        // Subiteration according to the deletion direction d
        // Phase 1
        potentially_deletable_list = < empty list >
        foreach point p = (x, y, z) in border_list do
            if point p is d-border and simple then
                // Candidate found
                potentially_deletable_list = potentially_deletable_list + < p >
            if point p is curve-isthmus in (ℤ^3, 26, 6, Y) then
                // Isthmus found
                A[x, y, z] = 3
                border_list = border_list − < p >
        // Phase 2
        foreach point p = (x, y, z) in potentially_deletable_list do
            if point p is simple then
                // Deletion
                A[x, y, z] = 0
                border_list = border_list − < p >
                number_of_deleted_points = number_of_deleted_points +1
                // Update border_list
                foreach point q = (x', y', z') that is 6-adjacent to p do
                    if A[x', y', z'] = 1 then
                        A[x', y', z'] = 2
                        border_list = border_list + < q >
until number_of_deleted_points = 0;
```

Thanks to the use of lists and look-up-tables, the proposed implementation of algorithm **I-3D-C-T** is is very efficient computationally: it is capable of producing centerlines from large 3D pictures containing 1 000 000 object points within half a second on a standard PC.

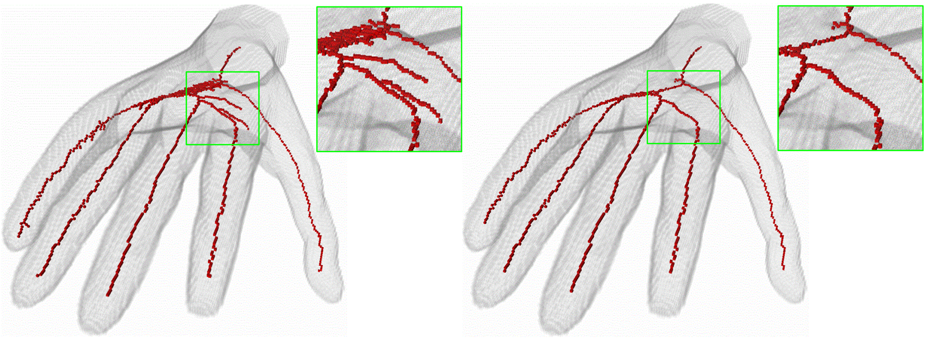
4 Results

In experiments the existing endpoint-based algorithm **E-3D-C-T** proposed by Palágyi et al. [9] and the new isthmus-based algorithm **I-3D-C-T** were tested



E-3D-C-T (4 960) **I-3D-C-T** (4 622)

Fig. 3. Centerlines produced by the existing endpoint-preserving curve-thinning algorithm and the proposed isthmus-based curve-thinning algorithm superimposed on a $512 \times 512 \times 591$ image of a segmented human airway tree containing 385 423 object points



E-3D-C-T (1 021) **I-3D-C-T** (781)

Fig. 4. Centerlines produced by the existing endpoint-preserving curve-thinning algorithm and the proposed isthmus-based curve-thinning algorithm superimposed on a $174 \times 103 \times 300$ image of a hand containing 865 941 object points

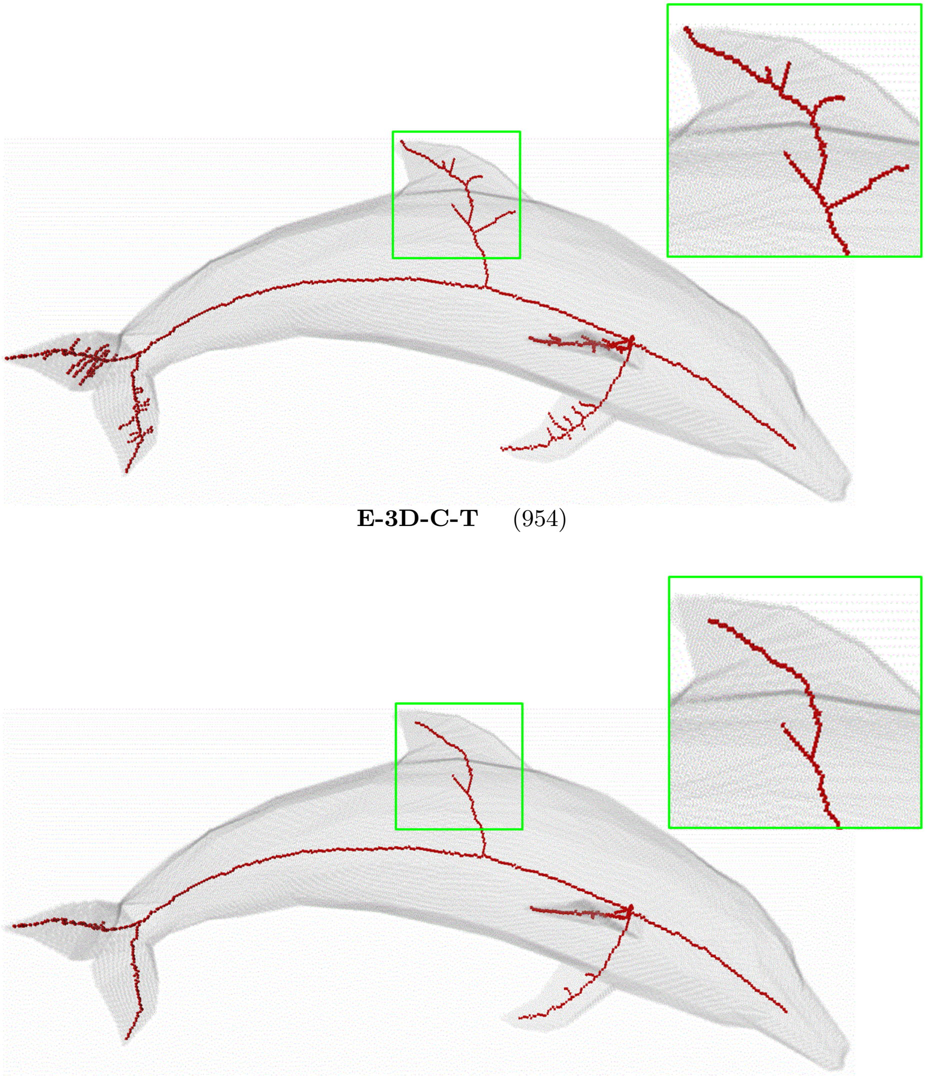


Fig. 5. Centerlines produced by the existing endpoint-preserving curve-thinning algorithm and the proposed isthmus-based curve-thinning algorithm superimposed on a $348 \times 130 \times 215$ image of a dolphin containing 1 202 772 object points

on various synthetic and natural objects. Note that we cannot compare the new algorithm **I-3D-C-T** with the sequential 3D thinning algorithm proposed by [4], since that is a surface-thinning algorithm (i.e., it cannot produce centerlines). Due to the lack of space, here we can present just three illustrative examples, see Figs. 3-5. The numbers in parentheses are the counts of object points in the produced centerlines.

Thanks to the isthmus-based approach, the proposed algorithm **I-3D-C-T** can produce less unwanted side branches than the conventional endpoint-based algorithm **E-3D-C-T** do. Note that each skeletonization technique (including thinning) is rather sensitive to coarse object boundaries. The false segments included by the produced centerlines can be removed by a pruning process (i.e., a post-processing step) [13].

5 Conclusions

In this paper we present a new sequential isthmus-based 3D curve-thinning algorithm named **I-3D-C-T**. It is guaranteed that the proposed algorithm preserves topology for all possible pictures. Due to the described implementation scheme (which uses a list to store the border points in the actual picture and two look-up-tables to encode simple points and curve-isthmuses) the new algorithm is computationally efficient. It is demonstrated that the isthmus-based algorithm **I-3D-C-T** can produce less unwanted side branches than the conventional existing endpoint-based algorithm **E-3D-C-T** proposed by Palágyi et al. [9] do.

Acknowledgements. The author thanks the anonymous reviewers for their valuable suggestions and remarks about this work.

This work was supported by the European Union and co-funded by the European Social Fund. Project title: "Telemedicine-focused research activities on the field of Mathematics, Informatics and Medical sciences." Project number: TÁMOP-4.2.2.A-11/1/KONV-2012-0073.

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. Kardos, P., Palágyi, K.: Order-independent sequential thinning in arbitrary dimensions. In: Proc. Int. Conf. Signal and Image Processing and Applications, SIPA 2011, pp. 129–134 (2011)

5. Kardos, P., Palágyi, K.: Isthmus-based order-independent sequential thinning. In: Proc. Int. Conf. 9th Signal Processing, Pattern Recognition and Applications, SP-PRA 2012, pp. 28–34 (2012)
6. Kong, T.Y., Rosenfeld, A.: Digital topology: Introduction and survey. Computer Vision, Graphics, and Image Processing 48, 357–393 (1989)
7. 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)
8. 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)
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., 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, Heidelberg (2012)
11. Palágyi, K.: Parallel 3D 12-subiteration thinning algorithms based on isthmuses. In: Bebis, G., et al. (eds.) ISVC 2013, Part I. LNCS, vol. 8033, pp. 87–98. Springer, Heidelberg (2013)
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)
15. Sundar, H., Silver, D., Gagvani, N., Dickinson, S.: Skeleton based shape matching and retrieval. In: Proc. Int. Conf. Shape Modeling and Applications, pp. 130–139. IEEE (2003)
16. Wan, M., Liang, Z., Ke, Q., Hong, L., Bitter, I., Kaufman, A.: Automatic centerline extraction for virtual colonoscopy. IEEE Transactions on Medical Imaging 21, 1450–1460 (2002)
17. Wong, W.C.K., So, R.W.K., Chung, A.C.S.: Principal curves for lumen center extraction and flow channel width estimation in 3-D arterial networks: Theory, algorithm, and validation. IEEE Transactions on Image Processing 21, 1847–1862 (2012)