Topology preservation and skeletonization on the BCC and FCC grids

PhD Thesis

Gábor Karai

Supervisors: Péter Kardos, PhD and Kálmán Palágyi, DSc

Doctoral School of Computer Science
Department of Image Processing and Computer Graphics
Faculty of Science and Informatics
University of Szeged

Szeged
2025

Contents

List of Figures

Chapter 1

Introduction

Digital topology is the study of the topological properties of *binary digital pictures* (*pictures* in short) that assign a color of *black* or *white* to each element (i.e., *point*) of the given grid [43]. *Reductions* transform a picture by changing some *black* points to *white* ones that is referred to as *deletion*, while all white points remain unchanged [29]. *Sequential reductions* can delete only one point at a time, while *parallel reductions* may delete a set of points simultaneously [29].

Distance transform (*DT* in short) generates a non-binary array for the input binary picture consisting of feature and nonfeature elements, where the calculated value of each element gives the distance to the nearest feature element [12]. The produced *distance map* strongly depends on the applied distance. *Distance-based skeletonization* relies on a distance transform in which white points form the set of feature points. *Thinning* is another strategy for skeletonization, which is an iterative object reduction [44]. Distance-based methods are frequently combined with thinning, e.g., in [4, 13, 69, 78]. *Topology preservation* is a key requirement for skeletonization algorithms.

The cubic grid is mostly preferred for sampling 3D digital pictures due to its fairly simple and easy-to-implement structure. In this thesis, we put an emphasis on two of the non-standard 3D grids: the *body-centered cubic (BCC) grid* and the *face-centered cubic (FCC) grid* [10, 41, 43, 74]. These grids tessellate the 3D Euclidean space into truncated octahedra and rhombic dodecahedra, respectively. It is known that they have beneficial topological and geometrical properties compared to the cubic grid. First, both alternatives are proved to be more efficient in the sampling of 3D pictures, as their packing density (i.e., the ratio between the volume of the largest ball completely enclosed in a voxel and the volume of the voxel itself) is lower than for the conventional sampling scheme [31, 53, 81]. Another benefit of the alternate tessellation lies in its less ambiguous connectivity structure [41, 43]. Due to these advantages, both non-standard grids attract scientific interest, such as organising coordinate-systems [9, 20, 21, 22], repairing 3D binary images [17,

18], performing a reconstruction scheme [16], investigating Hamiltonian cycles [19], graph coloring [26] and distance transformation [25, 73, 75].

This thesis is organised as follows: in Chapter 2, the basic concepts and preliminaries of digital topology and skeletonization are reviewed. In Chapters 3–5, the Author presents his own results. Chapter 3, which is part of the **first thesis group**, reports sufficient conditions for topology-preserving parallel reductions on binary pictures sampled on the BCC and FCC grids. Chapter 4, which is part of the **second thesis group**, presents the derived thinning algorithms of the constructed conditions generated parallel reductions discussed in Chapter 3. In Chapter 5, as part of the **third thesis group**, the Author recalls Strand's distance-based surface-thinning methods [72], then describes the modified variants which have linear time complexity and are less sensitive to the visiting order of border points. In addition, two novel algorithms are reported, which are capable of producing curve skeletons on $(14, 14)$ pictures of the BCC grid and $(18, 12)$ pictures of the FCC grid. These chapters are followed by the bibliography, then a summary in English and Hungarian. Finally, the thesis ends with a list of the Author's publications.

Chapter 2

Preliminaries

In this chapter, we introduce the key concepts of digital topology as reviewed by Kong and Rosenfeld [43]. Furthermore, we also briefly review the existing results that we will rely on to present our contribution.

2.1 Fundamentals of digital topology

A $(k, \bar{k})$ *binary digital picture* (*picture* in short) on grid $\mathcal{V}$ is defined as a quadruple $(\mathcal{V}, k, \bar{k}, B)$, where $\mathcal{V}$ is the set of picture elements called *points* [43]. Each point in $B \subseteq \mathcal{V}$ is called a *black point* and has a value of 1 assigned to it. Each point in $\mathcal{V} \setminus B$ is called a *white point* and has a value of 0 assigned to it. A binary picture is *finite* if $|B|$ (i.e., the number of black points) is finite. Furthermore, k and $\bar{k}$ are associated with the numbers of points *adjacent* to black points and white points, respectively. In 3D, two *Voronoi cell* (i.e., *voxel*) generated by $\mathcal{V}$ are used to be called *adjacent* to each other if they are connected by a face, an edge, or a vertex. On the other hand, the *Voronoi neighborhood* of a point $p \in \mathcal{V}$ is the set of all points in the 3D Euclidean space that are at least as close to p as to any other point in the corresponding grid [43]. A common metric to specify the 'closeness' is the *Euclidean distance*, which measures the length of the straight line segment between two points.

Since the adjacency relations are symmetric, their transitive closures form equivalence relations, and their generated equivalence classes are called *components*. In a picture, a *black component* or an *object* is a k-component, while a *white component* is a $\bar{k}$-component of the set of white points $\mathcal{V} \setminus B$. A black point is called a *border point* in a picture if it is $\bar{k}$-adjacent to at least one white point. Furthermore, a border point p is called an *isolated point* if all points in its k-adjacency are white (i.e., $\{p\}$ is a singleton object). In a finite picture, there is a unique white component that is called the *background*, while each finite white component is called a *cavity*.

2.2 Basic notions and results on the BCC and FCC grids

This dissertation focuses on two 3D grids: the *body-centered cubic grid* (*BCC grid* in short) - denoted by $\mathbb{B}$ - and the *face-centered cubic grid* (*FCC grid* in short) - denoted by $\mathbb{F}$. Hence, $\mathcal{V} \in \{\mathbb{B}, \mathbb{F}\}$. Both grids can be formally defined as the following subsets of the cubic grid ($\mathbb{Z}^3$):

$$\begin{aligned} \mathbb{B} &= \{(x, y, z) \in \mathbb{Z}^3 \mid x \equiv y \equiv z \pmod 2\}, \\ \mathbb{F} &= \{(x, y, z) \in \mathbb{Z}^3 \mid x + y + z \equiv 0 \pmod 2\}. \end{aligned} \tag{2.1}$$

On the BCC grid, each voxel p has a shape of a *truncated octahedron* with 14 face-neighbors denoted by $N_{14}(p)$, without having further edge- or vertex-neighbors. However, 8 of them are closer to p, that is why we make a distinction between them. The set $N_8(p)$ includes these closer neighbors. Note that d_e denotes the Euclidean distance.

$$\begin{aligned} N_8(p) &= \{q \in \mathbb{B} \mid \big(d_e(p, q)\big)^2 = 3\} \\ N_{14}(p) &= N_8(p) \cup \{q \in \mathbb{B} \mid \big(d_e(p, q)\big)^2 = 4\} \end{aligned} \tag{2.2}$$

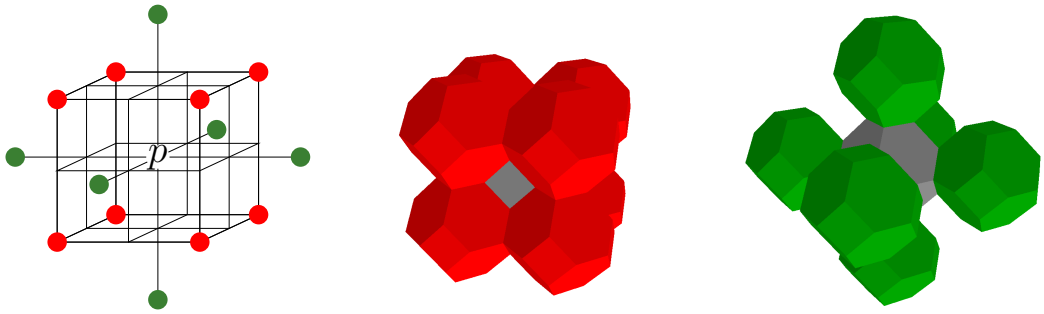
Due to the fact that, on the BCC grid, any two adjacent truncated octahedra share a face, the same adjacency relation can be applied for the black and white points. Hence, we assume $(14, 14)$ pictures on this type of grid. Additionally, a border point $p \in \mathbb{B}$ is called a *strong border point* if $N_8(p)$ contains at least one white point, otherwise p is called a *weak border point*.

On the FCC grid, each voxel p has a shape of a *rhombic dodecahedron*. For an FCC grid point p, its 12 face-neighbors are included by set $N_{12}(p)$. Additionally, it has 6 further vertex-neighbors, and the set $N_{18}(p)$ covers all of its 18 neighbors. Note that elements of $N_k(p)$ are *k-adjacent* to p.

$$\begin{aligned} N_{12}(p) &= \{q \in \mathbb{F} \mid \big(d_e(p, q)\big)^2 = 2\} \\ N_{18}(p) &= N_{12}(p) \cup \{q \in \mathbb{F} \mid \big(d_e(p, q)\big)^2 = 4\} \end{aligned} \tag{2.3}$$

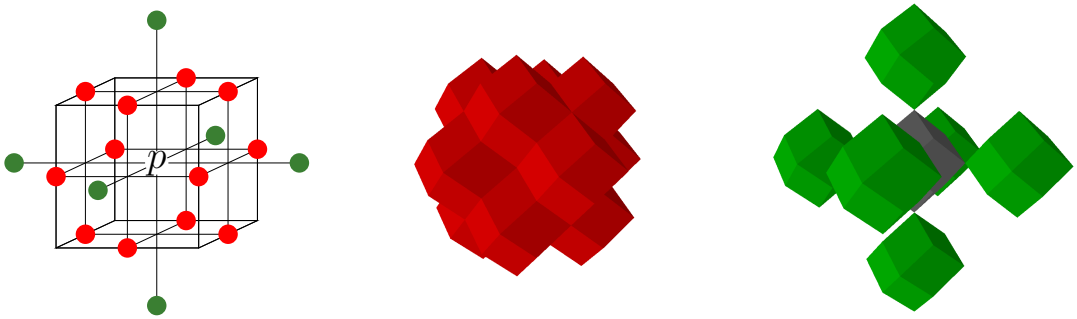
Gau and Kong examined three 'reasonable' types of pictures on the FCC grid, where $(k, \bar{k}) \in \{(18, 12), (12, 18), (12, 12)\}$ [27]. They showed that the $(18, 18)$ case is to be ignored since it generates 'connectivity paradoxes'. The problematic nature of $(18, 18)$ pictures is illustrated with the example of Fig. 2.3.

Let us focus on the addressing scheme shown in Figure 2.4, which maps every point in $\mathcal{V} \in \{\mathbb{B}, \mathbb{F}\}$ to a triplet of integer coordinates. The *lexicographical order*



(a) (b) (c)

Figure 2.1: *The two studied adjacency relations on the BCC grid shown in lattice view (a), where the eight points colored red form the set $N_8(p)$, and $N_{14}(p)$ contains six further points colored green. The voxel-representations of $N_8(p)$ in red (b) and $N_{14}(p) \setminus N_8(p)$ in green (c). (Note that unmarked elements of $\mathbb{Z}^3$ in (a) are not grid points in $\mathbb{B}$.)*



(a) (b) (c)

Figure 2.2: *The two studied adjacency relations on the FCC grid shown in lattice view (a), where the twelve points colored red form the set $N_{12}(p)$, and $N_{18}(p)$ contains six further points colored green. The voxel-representations of $N_{12}(p)$ in blue (b) and $N_{18}(p) \setminus N_{12}(p)$ in green (c). (Note that unmarked elements of $\mathbb{Z}^3$ in (a) are not grid points in $\mathbb{F}$.)*

relation "$\prec$" between two distinct points $p = (p_x, p_y, p_z)$ and $q = (q_x, q_y, q_z)$ is defined as follows:

$$p \prec q \quad \Leftrightarrow \quad (p_z < q_z) \vee (p_z = q_z \wedge p_y < q_y) \vee (p_z = q_z \wedge p_y = q_y \wedge p_x < q_x). \tag{2.4}$$

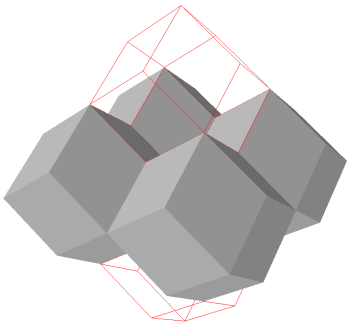


Figure 2.3: *If the* $(18, 18)$ *case is assumed, the four black voxels in this picture forms a single (18-connected) object and, surprisingly, a hole. That hole is illustrated by the two white voxels with red edges. We note that this contradiction can be formally derived by using the so-called Euler characteristic, a well-known topological invariant of binary objects [55].*

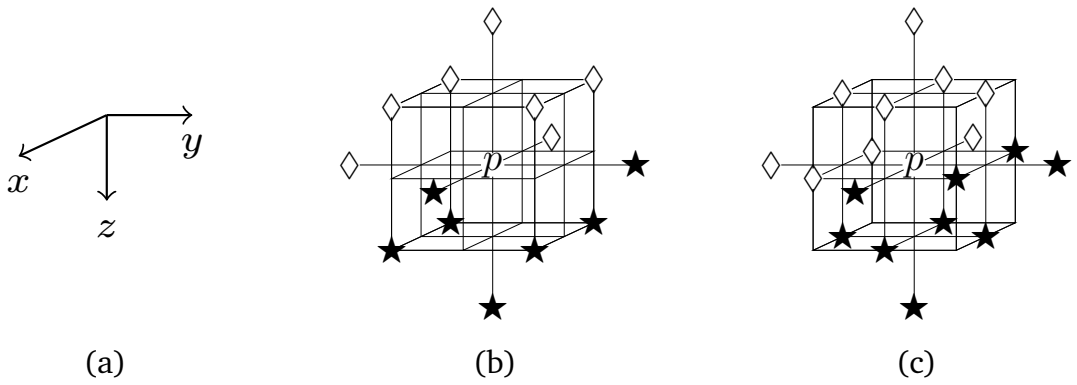


Figure 2.4: *The considered coordinate system (a) and the ordering scheme for the BCC grid (b) and the FCC grid (c). For center point* $p \in \mathbb{B}$ *the elements of the set of seven points* $\{q \in N_{14}(p) \mid p \prec q\}$ *are marked '★', and the remaining seven points in* $\{r \in N_{14}(p) \mid r \prec p\}$ *are marked '◊' (b). For center point* $p \in \mathbb{F}$ *the elements of the set of nine points* $\{q \in N_{18}(p) \mid p \prec q\}$ *are marked '★', and the remaining nine points in* $\{r \in N_{18}(p) \mid r \prec p\}$ *are marked '◊' (c).*

2.3 Topology-preserving reductions

Let $\mathcal{O}$ be an operation which takes the picture $(\mathcal{V}, k, \bar{k}, B)$ as input and results the picture $(\mathcal{V}, k, \bar{k}, \mathcal{O}(B))$. Operation $\mathcal{O}$ is a *reduction* if $\mathcal{O}(B) \subseteq B$ holds for any $B \subseteq \mathcal{V}$. In other words, it can change some black points to white ones, but cannot affect the white points from the input picture.

A reduction in a 2D picture is topology-preserving if each object in the input picture contains exactly one object in the output picture, and each white component

in the output picture contains exactly one white component in the input picture [43]. This definition can be rephrased as follows:

Definition 2.3.1. *A 2D reduction preserves the topology if*

- *it does not split any object (into several objects) in the input picture,*
- *it does not completely delete any object in the input picture,*
- *it does not merge any cavity with the background or another cavity in the input picture, and*
- *it does not create a cavity where there was none in the input picture.*

In 3D, there is an additional concept called *hole* or *tunnel*, which, e.g., a solid torus has. That is why Def. 2.3.1 must be extended for 3D pictures.

Definition 2.3.2. *A 3D reduction preserves the topology if it satisfies all conditions in Def. 2.3.1 and the following criteria:*

- *it does not eliminate or merge any existing hole in the input picture, and*
- *it does not create new holes where there was none in the input picture.*

2.3.1 Simple points

A black point is called *simple* if its deletion is a topology-preserving reduction [43]. Strand and Brunner characterized simple points on the BCC grid:

Theorem 2.3.1. [76] *A point $p \in B$ in picture $(\mathbb{B}, 14, 14, B)$ is simple if and only if the following conditions all hold:*

1. *$N_{14}(p) \cap B$ contains exactly one 14-component.*
2. *$N_{14}(p) \setminus B$ contains exactly one 14-component.*

Figure 2.5 presents examples of simple and non-simple black points on the BCC grid. Note that each set of points denoted with the same black symbol '●', or '▲' forms a black 14-component, and each set of points denoted with the same white symbol '○', or '◇' determines a white 14-component.

Gau and Kong gave the following characterizations of simple points in the considered three types of pictures on the FCC grid:

Theorem 2.3.2. [27] *A point $p \in B$ is simple in picture $(\mathbb{F}, k, \bar{k}, B)$ $(k, \bar{k}) \in \{(18, 12), (12, 18)\}$ if and only if the following conditions all hold:*

1. *The set of points $N_k(p) \cap B$ forms exactly one k-component.*

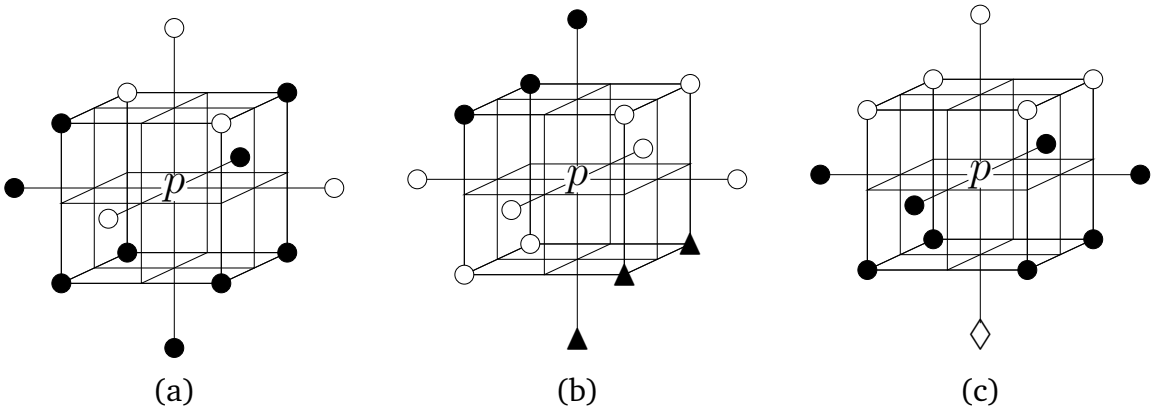


Figure 2.5: *Examples of simple (a) and non-simple black points (b–c) for* $(14, 14)$ *pictures on the BCC grid. In configuration (a),* p *is simple (since both conditions of Th. 2.3.1 hold); in configuration (b),* p *is not simple (since condition 1 of Th. 2.3.1 is violated); in configuration (c),* p *is not simple (since condition 2 of Th. 2.3.1 is violated).*

2. *The set of points* $N_{\bar{k}}(p) \setminus B$ *forms exactly one* $\bar{k}$*-component.*

Theorem 2.3.2 provides a unified characterization for $(18, 12)$ and $(12, 18)$ pictures, but the remaining case is an odd-one-out:

Theorem 2.3.3. [27] *A point* $p \in B$ *is simple in picture* $(\mathbb{F}, 12, 12, B)$ *if and only if the following conditions all hold:*

1. *The set of points* $N_{12}(p) \cap B$ *is nonempty and* 12*-connected in* $N_{18}(p) \cap B$*.*

2. *The set of points* $N_{12}(p) \setminus B$ *is nonempty and* 12*-connected in* $N_{18}(p) \setminus B$*.*

Theorems 2.3.1–2.3.3 imply that only non-isolated border points may be simple, and simple points can be locally characterized. The simpleness of a black point $p \in \{\mathbb{B}, \mathbb{F}\}$ in a picture can be decided by determining an integer code corresponding to the $N_{14}(p)$ or $N_{18}(p)$, and addressing a single pre-calculated (unit time access) look-up-table (LUT) containing the answers for all possible 2^{14} or 2^{14} configurations. The size of this look-up-table is 2 Kbytes and 32 Kbytes on the BCC and FCC grid, respectively.

Figures 2.6–2.8 give some illustrative examples of simple and non-simple black points on the FCC grid. Note that each set of points denoted with the same black symbol '●', '▲', or '♦' forms a black 18-component. Each set of points denoted with the same white symbol '○', '△', or '◊' determines a white 18-component. The 18-components of points marked '♦' and '◊' are not 12-components, but any other 18-components are also 12-compoments.

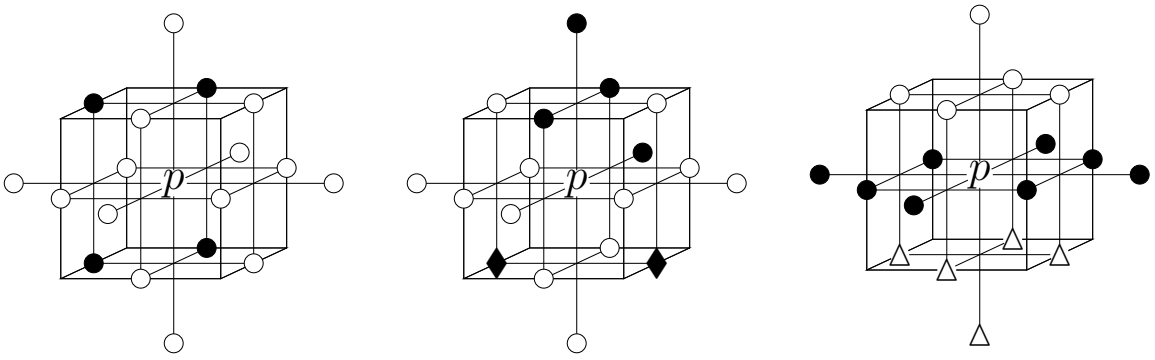


Figure 2.6: *Examples of simple (a) and non-simple black points (b–c) for* $(18, 12)$ *pictures on the FCC grid. In configuration (a),* p *is simple (since both conditions of Th. 2.3.2 hold); in configuration (b),* p *is not simple (since condition 1 of Th. 2.3.2 is violated); in configuration (c),* p *is not simple (since condition 2 of Th. 2.3.2 is violated).*

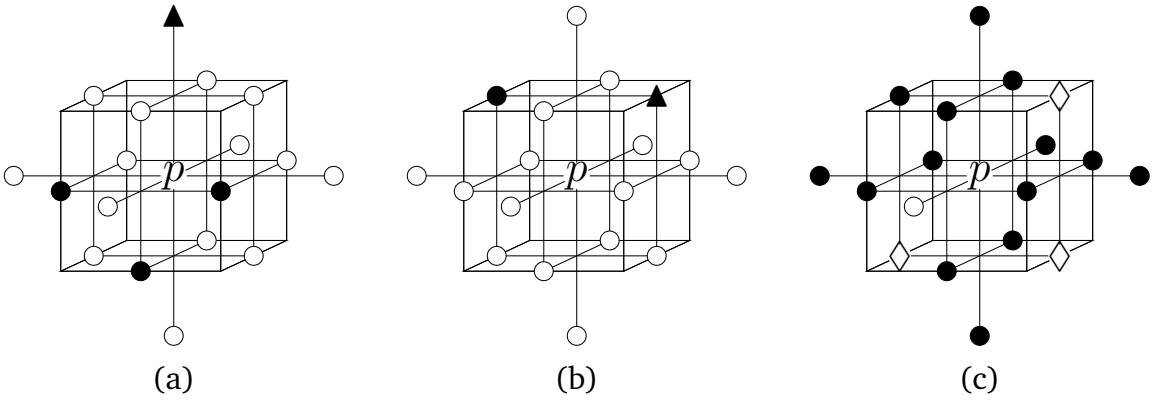
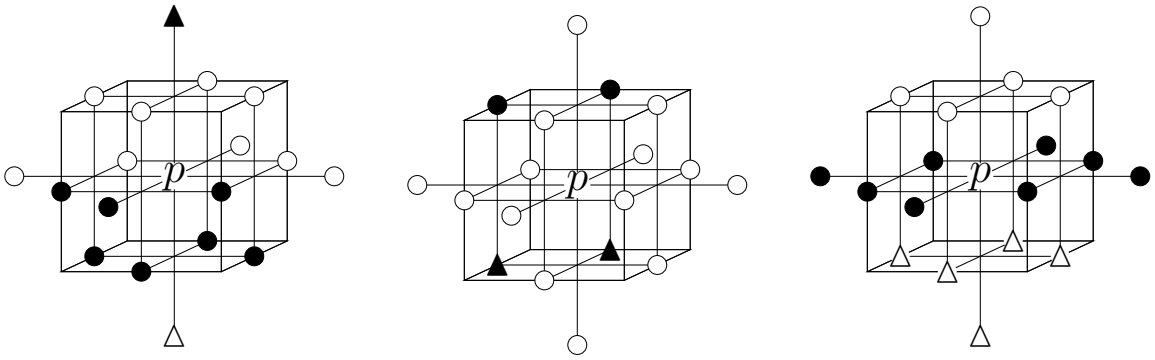


Figure 2.7: *Examples of simple (a) and non-simple black points (b–c) for* $(12, 18)$ *pictures on the FCC grid. In configuration (a),* p *is simple (since both conditions of Th. 2.3.2 hold); in configuration (b),* p *is not simple (since condition 1 of Th. 2.3.2 is violated); in configuration (c),* p *is not simple (since condition 2 of Th. 2.3.2 is violated).*

2.3.2 Sufficient conditions

Sequential reductions may delete only one black point at a time, hence for such operations, the deletion of only a simple point ensures topology preservation. In contrast, *parallel reductions* can delete a set of points simultaneously. Thus we need to consider what is meant by topology preservation when more than one point is deleted at a time.

Before describing the existing results on the considered grids, we need to recall some universal ones.



(a) (b) (c)

Figure 2.8: *Examples of simple (a) and non-simple black points (b–c) for* $(12, 12)$ *pictures on the FCC grid. In configuration (a),* p *is simple (since both conditions of Th. 2.3.3 hold); in configuration (b),* p *is not simple (since condition 1 of Th. 2.3.3 is violated); in configuration (c),* p *is not simple (since condition 2 of Th. 2.3.3 is violated).*

Universal sufficient conditions

Here, we define first the concepts of a *simple set*, and a *simple sequence*.

Definition 2.3.3. [42] *A set of* n *points* $Q \subset B$ *is a simple set in an arbitrary picture* $(\mathcal{V}, k, \bar{k}, B)$ *if it is possible to arrange the elements of* Q *in a sequence* $\langle q_1, \ldots, q_n \rangle$ *such that* q_1 *is simple for* B*, and each* q_i *is simple for* $B \setminus \{q_1, \ldots, q_{i-1}\}$ $(i = 2, \ldots, n)$*. Such a sequence is called a simple sequence. (And let the empty set be simple.)*

It is an axiom that a parallel reduction is topology-preserving if it only deletes simple sets from all possible pictures [42].

Ronse proposed a key concept called a *minimal non-simple (MNS)* set:

Definition 2.3.4. [66] *A set of black points in an arbitrary picture is a minimal non-simple (MNS) set if it is non-simple but all of its proper subsets are simple.*

The following theorem gives a universal sufficient condition for topology-preserving parallel reductions.

Theorem 2.3.4. [66] *A reduction preserves the topology for an arbitrary picture if it never deletes MNS sets.*

Bertrand introduced the notion of a *P-simple set*, whose simultaneous deletion preserves the topology:

Definition 2.3.5. [7] *Let* B *be the set of black points in an arbitrary picture. A set of points* $Q \subset B$ *is a P-simple set if for any point* $q \in Q$ *and any set of points* $R \subseteq Q \setminus \{q\}$*,* q *is simple for* $B \setminus R$*. Each element of a P-simple set is called a P-simple point.*

He also stated that simultaneous deletion of P-simple sets preserves the topology:

Theorem 2.3.5. [7] *A reduction that deletes a set composed solely of P-simple points is topology-preserving.*

Kong reported a further solution to the problem by introducing the notion of a *hereditarily simple set*, whose simultaneous deletion is proved to be topology-preserving:

Definition 2.3.6. [42] *Let B be the set of black points in an arbitrary picture. A set of points $Q \subset B$ is said to be hereditarily simple if all subsets of Q are simple sets.*

Theorem 2.3.6. [42] *A reduction that deletes only hereditarily simple sets is topology-preserving.*

The above mentioned universal conditions (see Theorems 2.3.4–2.3.6) examine some sets of deletable points or sets of points to be preserved. Palágyi proposed a unique sufficient condition that takes the *deletion rule*s of reductions into consideration [63]. (The deletion rule associated with a reduction specifies the individual points to be deleted.) This approach is also valid for arbitrary pictures.

Definition 2.3.7. [63] *A deletion rule is general-simple if the deletability of any point does not depend on the 'color' of any deletable point, and it deletes only simple points.*

Theorem 2.3.7. [63] *A (sequential or parallel) reduction is topology-preserving if its deletion rule is general-simple.*

Existing results on the BCC and FCC grids

Gau and Kong gave the definition of a small set on the FCC grid [27]: A set of mutually 18-adjacent points is called a *small set*. Kardos extended the concept of a small set for the BCC grid [39]: A set of mutually 14-adjacent points is called a *small set*. We even define the concept of a *small object* for the BCC and FCC grids: an object is said to be small if its elements form a small set.

It can be readily seen that a small set and a small object may consist of at most four and six points on the BCC and FCC grid, respectively, see Figs. 2.9 and 2.10.

Note that a small set on the FCC grid is an 18-component, but it does not need to be a 12-component, see Fig. 2.11. Furthermore, on the FCC grid, a set of four points form a *cross* if we delete two points from the maximal small set that are not 12-adjacent, see Fig. 2.12.

Figure 2.13 gives examples for simple sets and small sets on the BCC grid.

In their seminal work, Gau and Kong identified all the possible MNS sets for the studied types of pictures of the FCC grid:

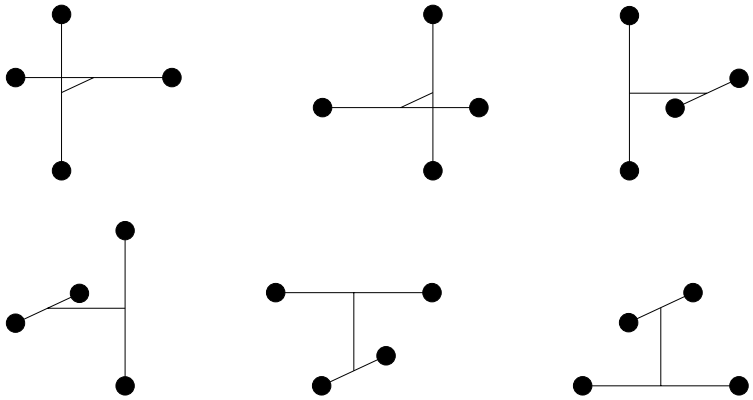


Figure 2.9: *The six possible maximal small sets on the BCC grid. All small sets are (nonempty) subsets of them.*

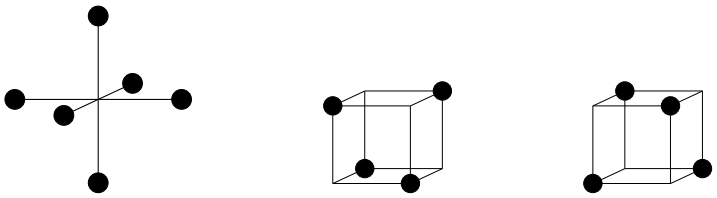


Figure 2.10: *The three possible maximal small sets on the FCC grid. All small sets are (nonempty) subsets of them. Note that all points are mutually 12-adjacent only in the middle and right configurations.*

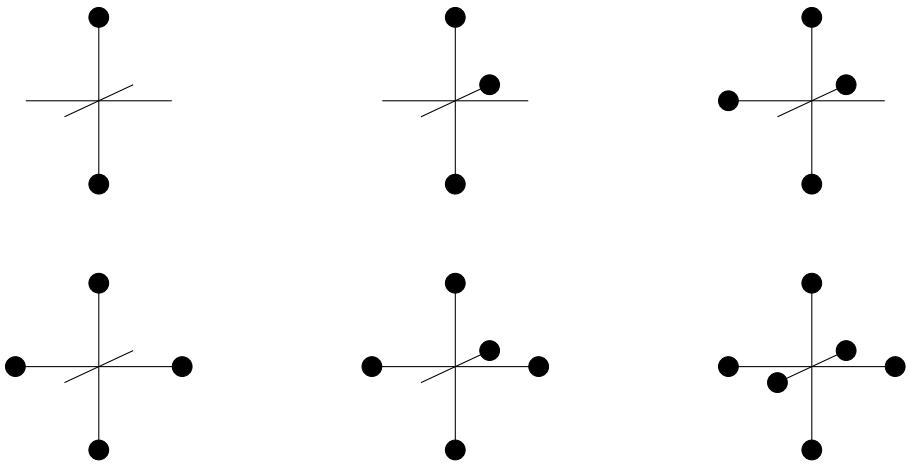


Figure 2.11: *Six of the possible 37 small objects in which the points are not mutually* 12*-adjacent. The remaining ones are rotated and reflected versions of these six base objects.*

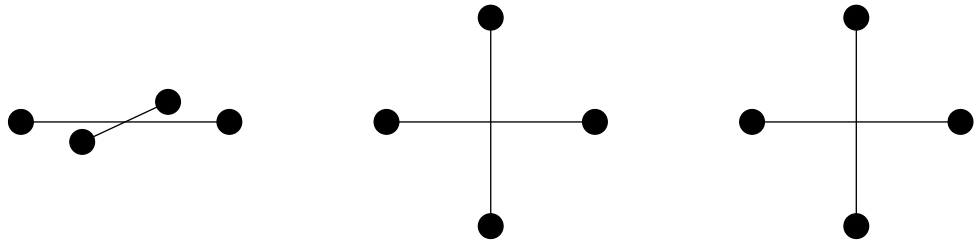


Figure 2.12: *The three possible crosses on the FCC grid.*

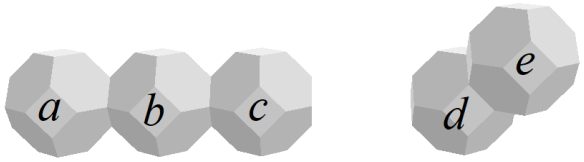


Figure 2.13: *Examples of simple sets and small sets on the BCC grid. The sets of black voxels* $\{a, b, c\}$ *and* $\{d, e\}$ *form two objects. The sequence* $\langle a, b\rangle$ *is a simple sequence, because point* a *is simple, and point* b *becomes simple after the deletion of* a*. Hence,* $\{a, b\}$ *is a simple set. For similar reasons,* $\{b, c\}$ *is also a simple set. Both* $\{a, b\}$ *and* $\{b, c\}$ *are small sets, but none of them is a small object. Since* $a \notin N_{14}(c)$*,* $\{a, b, c\}$ *is not a small set and it is not a small object. The set* $\{d, e\}$ *is a small set, and also a small object. We can state that each small object is a small set, but some small sets are not small objects.*

Theorem 2.3.8. [27] *Let* $Q \subseteq B$ *be a nonempty small set of black points in picture* $(\mathbb{F}, 18, 12, B)$.

1. *If* Q *is a subset of three mutually* 12*-adjacent points,* Q *may be an MNS set (without being an object).*

2. *Let* Q *be a set of four mutually* 12*-adjacent points. Then* Q *is an MNS set if and only if it is an object.*

3. *Let* Q *be a set of points in which its elements are not mutually 12-adjacent (i.e., there are two points* $p, q \in Q$ *such that* $q \in (N_{18}(p) \setminus N_{12}(p))$*). Then* Q *is an MNS set if and only if it is an object.*

Theorem 2.3.9. [27] *Let* $Q \subseteq B$ *be a nonempty small set in picture* $(\mathbb{F}, 12, 12, B)$.

1. *If* Q *is a set of three mutually* 12*-adjacent points,* Q *may be an MNS set (without being an object).*

2. *If* Q *is a subset of four points that form a cross,* Q *may be an MNS set (without being an object).*

3. *Let Q be a set of four mutually 12-adjacent points. Then Q is an MNS set if and only if it is an object.*

Theorem 2.3.10. [27] *Let $Q \subseteq B$ be a nonempty small set in picture $(\mathbb{F}, 12, 18, B)$.*

1. *If Q is a subset of three mutually 12-adjacent points, Q can be an MNS set (without being an object).*
2. *If Q is a set of two points $\{p, q\}$, where $q \in N_{18}(p) \setminus N_{12}(p)$, Q can be an MNS set (without being an object).*
3. *Let Q be a set of four mutually 12-adjacent points. Then Q is an MNS set if and only if it is an object.*

Kardos gave the following sufficient condition for topology-preserving parallel reductions on the BCC grid:

Theorem 2.3.11. [39] *A parallel reduction $\mathcal{R}$ is topology-preserving if it fulfills all the following conditions:*

1. *Any set of at most three mutually 14-adjacent black points deleted by $\mathcal{R}$ is simple.*
2. *$\mathcal{R}$ does not delete completely any object composed of at most four mutually 14-adjacent points.*

2.4 Skeletonization and its techniques

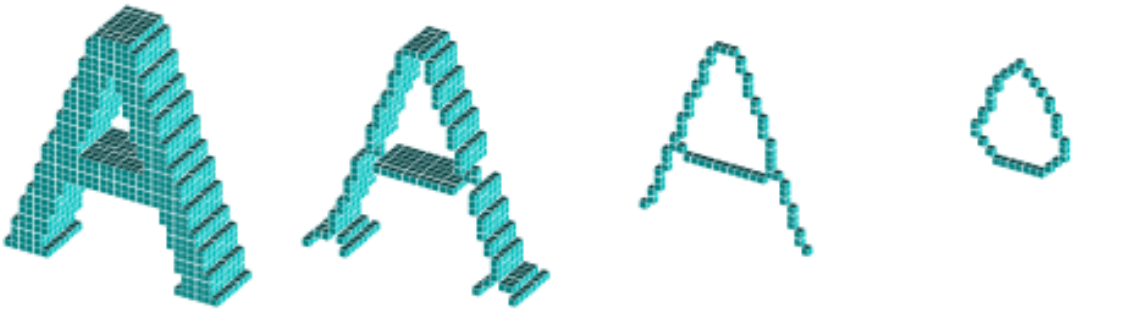
Skeleton is a shape feature which describes the topology and the geometry of a binary object [11]. Blum gave three equivalent approaches for this continuous descriptor.

- The skeleton of a continuous object consists of its points having at least two closest boundary points. (Result of the *Medial Axis Transform*).
- The *Prairie-fire propagation* model assumes that the object is a field of dry grass and its entire boundary is set on fire at a time. It is also supposed that the fire spreads in all directions with equal velocity. Then the skeleton is the set of points where the fire fronts meet and extinguish each other.
- The skeleton is the union of the centers of all maximal inscribed hyperspheres (i.e., disks in 2D, balls in 3D), where a hypersphere is inscribed if it is completely included by the object, and it is maximal if it is not covered by any other inscribed hypersphere

The skeleton is a widely used shape descriptor due to its advantageous properties:

- It represents local object symmetries [14, 23, 28].
- It reflects the topological structure of the object to be characterized [54].
- It can be used to object decomposition (i.e., to partition an object into a set of primitives) [11, 56].
- It reduces the dimensionality since a 2D object is reduced to 1D curves; the skeleton of a 3D object may contain just 2D and 1D structures (surfaces and curves) [23, 28, 56].
- It is 'thin' (i.e., the skeleton contains 1-point width manifolds and offers dramatically less information than the original object) [23, 28].

The above-defined skeleton assumes *continuous* approach. Since we deal with digital pictures, *skeletonization* means extraction of skeleton-like features from digital binary objects [68]. In 3D there are three kinds of *skeleton-like shape features*. The *medial surface* (or so-called *surface skeleton*) can both contain (2D) surface patches and (1D) line segments, while the *centerline* (or so-called *curve skeleton*) should have only line segments, and finally, the *topological kernel* is a minimal set of points that is topologically equivalent to the original object. Note that a line segment can be also called *branch*. Figure 2.14 illustrates the differences between these shape features.



original object | *medial surface (surface skeleton)* | *centerline (curve skeleton)* | *topological kernel*

Figure 2.14: *Examples for each 3D skeleton-like shape feature.*

Several approaches have been proposed for producing skeleton-like features from (segmented) binary objects. Some authors presented comprehensive and concise surveys [67, 68, 71, 80]. There are three major skeletonization techniques: geometric approach based on Voronoi diagrams, modeling the fire-front propagation (thinning), and generation from distance maps. In this dissertation, the latter two ones are discussed in more details.

2.4.1 Voronoi approach

The *Voronoi skeleton* is obtained by computing the Voronoi diagram of sampled boundary points [1, 79]. Schmitt has shown that if the density of the sampled boundary points uniformly goes to infinity, the 'internal' elements of the corresponding Voronoi diagram converges to the (exact continuous) skeleton [70]. Several authors proposed computationally efficient algorithms for computing the Voronoi diagram, e.g., in [15, 59, 65]. The raw Voronoi skeleton may contain a large number of spurious branches in 2D, and unwanted surface patches in the 3D case. Thus the Voronoi skeletonization is to be paired with a proper pruning method [5, 60].

Although the Voronoi skeleton is correct in both geometrical and topological senses, this method is rather time-consuming and 'hybrid' (i.e., its input is a discrete set of points, but the output is formed by continuous segments).

2.4.2 Thinning

Thinning [29, 44, 71] is an iterative object reduction, where in each iteration the black points classified as *deletable* are removed from the set of border points (i.e., the outmost object layer) from the actual picture [44]. The procedure terminates when stability is reached. In order to determine if a border point is deletable, its local environment called *support* must be examined [29]. If the applied criterion ensures only topology-preservation without leaving any simple point in the resulting picture, the object's topological kernel will be extracted. These methods are called reductive *shrinking* [30]. We also refer to them as *kernel-thinning* algorithms. If some geometrical constraints are also included in the deletion criterion, it can produce the medial surface or centerline of the input objects. For this purpose, a common way is to apply a *surface-* or *line-endpoint* definition. These endpoints are always simple, but considered as non-deletable. Due to the expectation of topology-preservation, these endpoints must be connected to the resulting skeleton. Thus, the appearance of each endpoint at some stage of thinning necessarily creates an extra surface patch or line segment.

A general and computationally efficient implementation scheme for thinning algorithms was proposed by Palágyi [62]. This scheme takes advantage of the fact that all thinning algorithms may delete only border points. Thus we do not have to examine the deletability of interior points, and the repeated scans of the entire array (that stores the actual picture) can be avoided by using a linked list that stores the set of border points to be evaluated for possible deletion in the actual thinning phase. The reported thinning algorithms in chapters 4, and 5 are implemented in this way.

Thinning algorithms can be classified as *sequential* or *parallel* depending on whether they delete one or more border points at a time [29, 44]. Additionally, parallel thinning strategies can be further distributed into three subcategories: *fully parallel*,

directional, and *subfield-based*.

Sequential thinning

Sequential thinning methods traverse the border points of a picture, and focus on the actually visited single point for possible deletion, see Alg. 2.1. To make sure only those black points are visited, which were part of the border at the beginning of the actual iteration, the border points should be collected into a set. Then, only the precollected black points will be investigated for possible deletion. The significant advantage of this approach is that if only simple points may be deleted, topology preservation is already guaranteed.

Algorithm 2.1: Sequential thinning

Input: picture $(\mathcal{V}, k, \bar{k}, X)$
Output: picture $(\mathcal{V}, k, \bar{k}, Y)$

```
1 Y ← X
2 repeat
      // Phase one: border-tracking
3     deleted ← false
4     S ← {p ∈ Y | p is a border point for Y}
      // Phase two: reduction
5     foreach p ∈ S do
6         if p is 'deletable' for Y then
7             Y ← Y \ {p}
8             deleted ← true
9 until deleted = false
```

Note that the visiting order may affect the resulting skeleton. Moreover, the examination order (within one iteration) is usually not predefined (e.g. in [13, 69, 72, 78]), which makes harder to reproduce these methods. In contrast, for the Author's sequential thinning algorithms presented in Chapter 5, the investigation of border points during all thinning stages is restricted to lexicographical order.

Fully parallel thinning

Among the parallel thinning strategies, the fully parallel one has the simplest general scheme, see Alg. 2.2. Each iteration step requires exactly one parallel reduction with a global deletion rule [45, 46, 47]. However, the deletion of simple points only is unsufficient for topology preservation by itself. For example, all border points belonging to a two-voxel thick line segment of the input object are simple, but their deletion may tear the object into multiple parts, which is a forbidden operation.

Chapter 3 presents the Author's sufficient conditions for topology-preserving parallel reductions acting on the BCC and FCC grids.

Algorithm 2.2: Fully parallel thinning

Input: picture $(\mathcal{V}, k, \bar{k}, X)$
Output: picture $(\mathcal{V}, k, \bar{k}, Y)$

$Y \leftarrow X$
repeat
  $D \leftarrow \{p \in Y \mid p \text{ is 'deletable' for } Y\}$
  $Y \leftarrow Y \setminus D$
until $D = \emptyset$

Directional parallel thinning

In directional (also referred to as subiteration-based) thinning algorithms [29], an iteration step is decomposed into k successive parallel reduction operations according to $k \geq 2$ deletion directions, e.g., in [6, 48, 61]. If the current deletion direction is d, then a set of d-border points can be deleted by the parallel reduction operation assigned to it, see Alg. 2.3.

Algorithm 2.3: Directional thinning

Input: picture $(\mathcal{V}, k, \bar{k}, X)$
Output: picture $(\mathcal{V}, k, \bar{k}, Y)$

$Y \leftarrow X$
repeat
  $D \leftarrow \emptyset$
  for $i \leftarrow 1$ **to** k **do**
    `// One subiteration`
    $D_i \leftarrow \{p \mid p \text{ is '}i\text{-deletable' for } Y\}$
    $Y \leftarrow Y \setminus D_i$
    $D \leftarrow D \cup D_i$
until $D = \emptyset$

Subfield-based parallel thinning

In this thinning strategy, the set of all picture elements $\mathcal{V}$ is partitioned into $t \geq 2$ subfields, and let $SF(i)$ denote the set of picture elements belonging to the i-th subfield ($i \in \{0, 1, \ldots, t-1\}$). In each subiteration, only one subfield is activated, and the set of picture elements corresponding to the currently activated subfield are

taken into consideration for possible deletion, see Alg. 2.4. In existing subfield-based parallel 3D thinning algorithms, the cubic grid is partitioned into two [49, 50, 57], four [51, 58], and eight [8, 58] subfields.

Algorithm 2.4: Subfield-based thinning

Input: picture $(\mathcal{V}, k, \bar{k}, X)$
Output: picture $(\mathcal{V}, k, \bar{k}, Y)$

```
1 Y ← X
2 repeat
3     D ← ∅
4     for i ← 0 to t − 1 do
          // One subiteration
5         D_i ← {p ∈ SF(i) | p is 'deletable' for Y}
6         Y ← Y \ D_i
7         D ← D ∪ D_i
8 until D = ∅
```

Subsection 4.2 presents the Author's subfield-based kernel-thinning algorithms, in which all elements of the BCC grid are partitioned into 4 and 8 subfields, respectively. Note that, in his related work, Kardos proposed a shrinking algorithm on the BCC grid with 2 subfields [39].

2.4.3 Distance-based methods

Distance-based skeletonization refers to the definition of the (continuous) skeleton by the centers of all maximal inscribed hyperspheres. It relies on the *distance map* which is computed by using a properly selected distance function d and a computationally efficient algorithm for distance transform. *Distance transform* (*DT* in short) generates a non-binary array for the input binary picture consisting of feature and nonfeature elements, where the calculated value of each element gives the distance to the nearest feature element [12]. For skeletonization, the set of white points $\{q \mid q \in \mathcal{V} \setminus B\}$ should be marked as feature points, and let Γ denote the set of feature points. For a point p, the value in the distance map is defined as follows:

$$DT(p) = \min_{q \in \Gamma} d(p, q). \tag{2.5}$$

To compute the distance, function d must be chosen. For geometrically accurate results, the *Euclidean distance* should be applied. Usually, EDT refers to the Euclidean distance transform. If $\mathcal{V} \subseteq \mathbb{Z}^3$ and we omit the square root operation, we get the square of the Euclidean distance transform – generally denoted by $SEDT$ –, in which the distance values are guaranteed to be integer numbers.

There are two frequently applied ways to approximate the Euclidean distance. One of them is the *neighbor-based distance*, where $d_i(p, q)$ denotes the length of the shortest i-path between p and q [52]. The sequence of distinct points $\langle x_0, x_1, \ldots, x_s \rangle$ in a non-empty set of points X is called an *i-path* from x_0 to x_s in X if x_k is i-adjacent to x_{k-1}, $(k = 1, \ldots, s)$. For better approximations to the exact Euclidean distance, the length of the 'valid moves' between two points can be weighted according to some criteria, and the path of the minimal sum of weights called as *Chamfer distances* can be used. Borgefors showed that Chamfer distance maps can be computed in linear time in arbitrary dimensions, which requires two raster scans through the entire picture [12].

The conventional way to apply distance information for skeletonization is the detection of *local maxima* or *center of maximal ball* (*CMB*, for short) as skeletal points. These 'ridges' have been identified by standard techniques of differential geometry and heuristic methods [2, 3, 13]. If a geometrically precise distance function is used (i.e., d_e or its 'good' approximation), this strategy can guarantee that the skeleton is placed in the center of the object. As a critical drawback, the set of detected skeleton points are usually not connected. Hence, these methods do not preserve the topology.

To ensure topology preservation and produce geometrically correct skeleton, distance information is often applied in thinning, e.g., in [4, 13, 69, 78]. These methods can be classified as follows:

- *Anchor-based thinning*: First, the set of CMBs is detected as safe skeletal (anchor) points. Then thinning is applied to connect CMBs and determining the set of further skeletal points.

- *Distance-driven thinning*: Distance mapping is followed by a thinning process. In the i-th iteration, only object points with distance value i are taken into consideration for possible deletion.

- *Hybrid approach*: It combines the anchor-based and distance-driven approaches.

Among the algorithms presented in Chapter 5, we can see examples for each of these categories.

Chapter 3

Sufficient conditions for topology-preserving parallel reductions

In this chapter, first, we propose configuration-based and point-based conditions for topology-preserving parallel reductions acting on the BCC (see Sect. 3.1) and FCC grids (see Sect. 3.2). Then, in Sect. 3.3, we show that our point-based sufficient conditions allow us to directly construct topology-preserving parallel reductions. This work is covered by publications [37, 38, 64].

3.1 Conditions on the BCC grid

In this section, the sufficient conditions for $(14, 14)$ pictures of the BCC grid are presented.

3.1.1 Configuration-based conditions

Theorem 2.3.11 takes configurations of at most four mutually 14-adjacent points into consideration. Thus that theorem states a *configuration-based* sufficient condition for topology-preserving parallel reductions.

Notice that, by Theorem 2.3.11, it is difficult to verify that a previously designed parallel reduction preserves the topology.

In order to simplify this very first condition, first, we state the following proposition that is a straightforward consequence of the definition of simple set (see Def. 2.3.3):

Proposition 3.1.1. *Let $\langle q_1, \ldots, q_n \rangle$ be a simple sequence of a set of $n \geq 0$ black points Q. If a black point $p \notin Q$ is simple after the deletion of Q, the set of $n + 1$ points*

$\{q_1, \ldots, q_n, p\}$ *is simple.*

We will also make use of the following proposition:

Proposition 3.1.2. *If a set of black points Q forms an object, Q is not a simple set.*

Here, we recall a general lemma stated by Kardos and Palágyi:

Lemma 3.1.1. [40] *Let p and q be two black simple points in an arbitrary picture. If p remains simple after the deletion of q, then q remains simple after the deletion of p.*

In other words, the simpleness of a set of two simple points can be decided by examining just one sequence of its elements.

With the help of the previous propositions and lemma, Theorem 2.3.11 can be rephrased as follows:

Theorem 3.1.1. [64] *A parallel reduction is topology-preserving for $B \subset \mathbb{B}$ if the following conditions all hold:*

1. *Only simple points for B are deleted.*
2. *If two 14-adjacent points p and q are deleted,*
   p is simple for $B \setminus \{q\}$.
3. *If three mutually 14-adjacent points p, q, and r are deleted,*
   p is simple for $B \setminus \{q, r\}$, or
   q is simple for $B \setminus \{p, r\}$, or
   r is simple for $B \setminus \{p, q\}$.
4. *No object consisting of exactly four mutually 14-adjacent points is deleted completely.*

We can state that (similarly to Theorem 2.3.11) Theorem 3.1.1 provides a *configuration-based* sufficient condition for topology-preserving parallel reductions on the BCC grid.

3.1.2 Point-based conditions

Theorems 2.3.11 and 3.1.1 (i.e., the two configuration-based sufficient conditions for topology-preserving parallel reductions) just provide methods of verifying that a previously designed parallel reduction preserves the topology, rather than a methodology, for constructing topology-preserving parallel reductions. That is why we propose *point-based* sufficient conditions that directly provide deletion rules of topology-preserving parallel reductions, and allow us to generate various topology-preserving parallel thinning algorithms.

The following theorem deals with the deletability of individual points:

Theorem 3.1.2. [64] *A parallel reduction is topology-preserving for $B \subset \mathbb{B}$ if each point p deleted by this reduction satisfies all the following conditions:*

1. *Point p is simple for B.*
2. *For any point $q \in N_{14}(p) \cap B$ that is simple for B, point p is simple for $B \setminus \{q\}$.*
3. *For any two points $q \in N_{14}(p) \cap B$ and $r \in N_{14}(p) \cap N_{14}(q) \cap B$ that are simple for B, and q is simple for $B \setminus \{r\}$, p is simple for $B \setminus \{q, r\}$.*
4. *Point p is not an element of an object consisting of four mutually 14-adjacent points.*

Conditions of Theorem 3.1.2 may be viewed as *symmetric* since elements in the examined sets of at most four mutually 14-adjacent points are not distinguished. To construct *asymmetric* conditions for both grids, we introduce the concept of *smallest element* of a finite set.

Definition 3.1.1. *Let $Q \subset \mathcal{V}$ be a finite set of points. Point $p \in Q$ is the smallest element of Q if for any $q \in Q \setminus \{p\}$, $p \prec q$.*

Now we can state the following *asymmetric point-based condition*, which says that only the smallest element is non-deletable of any MNS set:

Theorem 3.1.3. [64] *A parallel reduction is topology-preserving for $B \subset \mathbb{B}$ if each point p deleted by that reduction satisfies all the following conditions:*

1. *Point p is simple for B.*
2. *For any point $q \in N_{14}(p) \cap B$ that is simple for B, point p is simple for $B \setminus \{q\}$, or $q \prec p$.*
3. *For any two points $q \in N_{14}(p) \cap B$ and $r \in N_{14}(p) \cap N_{14}(q) \cap B$ that are simple for B, and q is simple for $B \setminus \{r\}$, point p is simple for $B \setminus \{q, r\}$, or p is not the smallest element of set $\{p, q, r\}$.*
4. *Point p is not the smallest element of an object formed by four mutually 14-adjacent points.*

Since Theorem 3.1.3 distinguishes the elements of each MNS set, this condition can be called *asymmetric*.

3.2 Conditions on the FCC grid

In this section, we present our sufficient conditions for all three types of pictures of the FCC grid.

3.2.1 Configuration-based conditions

Following the same method as for the BCC grid (see Subsect. 3.1.1), we construct configuration-based conditions for all three types of pictures of the FCC grid.

Configuration-based conditions for *(18,12)* pictures

As a consequence of Theorem 2.3.8, we derive the following sufficient condition for topology-preserving parallel reductions:

Theorem 3.2.1. [38, 37] *A parallel reduction $\mathcal{R}$ is topology-preserving for an arbitrary picture $(\mathbb{F}, 18, 12, B)$ if the following conditions all hold:*

1. *Any set of at most three mutually 12-adjacent points $Q \subseteq B$ deleted by $\mathcal{R}$ is simple.*

2. *$\mathcal{R}$ does not delete completely any object of B composed of four mutually 12-adjacent points.*

3. *$\mathcal{R}$ does not delete completely any small object in which the points are not mutually 12-adjacent.*

For the same reason as in the case of the BCC grid (see Subsect. 3.1.1), we propose a simplified version of the previous condition:

Theorem 3.2.2. [38, 37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 18, 12, B)$ if the following conditions all hold:*

1. *Only simple points for B are deleted by $\mathcal{R}$.*

2. *If two 12-adjacent points p and q are deleted by $\mathcal{R}$, p is simple for $B \setminus \{q\}$.*

3. *If three mutually 12-adjacent points p, q, and r are deleted by $\mathcal{R}$,*
   p is simple for $B \setminus \{q, r\}$, or
   q is simple for $B \setminus \{p, r\}$, or
   r is simple for $B \setminus \{p, q\}$.

4. *$\mathcal{R}$ does not delete all points of any object of B composed of four mutually 12-adjacent points.*

5. *$\mathcal{R}$ does not delete all points of any small object in which the points are not mutually 12-adjacent.*

Configuration-based conditions for *(12,18)* pictures

Similarly to the situation of $(18, 12)$ pictures, we derive the following sufficient condition for $(12, 18)$ pictures based on Theorem 2.3.10:

Theorem 3.2.3. [37] *A parallel reduction $\mathcal{R}$ acting on $(12, 18)$ pictures of the FCC grid is topology-preserving if the following conditions all hold:*

1. *Any set of at most three mutually 12-adjacent black points deleted by $\mathcal{R}$ is a simple set.*
2. *Any set of two black points p and $q \in N_{18}(p) \setminus N_{12}(p)$, deleted by $\mathcal{R}$ is a simple set.*
3. *$\mathcal{R}$ does not delete all points of any object composed of four mutually 12-adjacent points.*

For similar reasons as for the $(18, 12)$ case, we simplify the above condition as follows:

Theorem 3.2.4. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 18, B)$ if the following conditions all hold:*

1. *Only simple points for B are deleted by $\mathcal{R}$.*
2. *If two 18-adjacent points p and q in B are deleted by $\mathcal{R}$, p is simple for $B \setminus \{q\}$.*
3. *If three mutually 12-adjacent points p, q, and r in B are deleted by $\mathcal{R}$, then*
   p is simple for $B \setminus \{q, r\}$, or
   q is simple for $B \setminus \{p, r\}$, or
   r is simple for $B \setminus \{p, q\}$.
4. *None of the objects in B consisting of four mutually 12-adjacent points are deleted by $\mathcal{R}$.*

Configuration-based conditions for *(12,12)* pictures

We establish the following sufficient condition for the last picture type based on Theorem 2.3.9:

Theorem 3.2.5. [37] *A parallel reduction $\mathcal{R}$ acting on $(12, 12)$ pictures of the FCC grid is topology-preserving if the following conditions all hold:*

1. *Any set of at most three mutually 18-adjacent black points deleted by $\mathcal{R}$ is a simple set.*
2. *Any set of four black points that form a cross and is deleted by $\mathcal{R}$ is a simple set.*

3. *$\mathcal{R}$ does not delete all points of any object composed of four mutually 12-adjacent points.*

Similarly to the $(18, 12)$ and $(12, 18)$ cases, we propose a simplified configuration-based condition:

Theorem 3.2.6. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 12, B)$ if the following conditions all hold:*

1. *Only simple points for B are deleted by $\mathcal{R}$.*
2. *If two 18-adjacent points p and q in B are deleted by $\mathcal{R}$, p is simple for $B \setminus \{q\}$.*
3. *If three mutually 18-adjacent points p, q, and r in B are deleted by $\mathcal{R}$, then*
   p is simple for $B \setminus \{q, r\}$, or
   q is simple for $B \setminus \{p, r\}$, or
   r is simple for $B \setminus \{p, q\}$.
4. *If four points p, q, r, and s in B that form a cross and are deleted by $\mathcal{R}$, then*
   p is simple for $B \setminus \{q, r, s\}$, or
   q is simple for $B \setminus \{p, r, s\}$, or
   r is simple for $B \setminus \{p, q, s\}$, or
   s is simple for $B \setminus \{p, q, r\}$.
5. *None of the objects in B consisting of four mutually 12-adjacent points are deleted by $\mathcal{R}$.*

Unified configuration-based conditions

Theorems 3.2.2, 3.2.4, and 3.2.6 related to the three types of pictures can be unified in the following form:

Theorem 3.2.7. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, k, \bar{k}, B)$ ($(k, \bar{k}) \in \{(18, 12), (12, 18), (12, 12)\}$) if the following conditions all hold:*

1. *Only simple points for B are deleted by $\mathcal{R}$.*
2. *Let p and q be two points in B such that if $(k, \bar{k}) = (18, 12)$, then p and q are 12-adjacent, else they are 18-adjacent. If $\mathcal{R}$ deletes both p and q, then p is simple for $B \setminus \{q\}$.*
3. *Let p, q, and r be three points in B such that if $(k, \bar{k}) = (12, 12)$, then p, q, and r are mutually 18-adjacent, else they are mutually 12-adjacent. If $\mathcal{R}$ deletes each of the points p, q, and r, then*

p is simple for $B \setminus \{q, r\}$, or
q is simple for $B \setminus \{p, r\}$, or
r is simple for $B \setminus \{p, q\}$.

4. *If $(k, \bar{k}) = (18, 12)$, then $\mathcal{R}$ does not delete all points of any small object in which the points are not mutually 12-adjacent.*

5. *In the case $(k, \bar{k}) = (12, 12)$, if four points p, q, r, and s in B that form a cross and are deleted by $\mathcal{R}$, then*
p is simple for $B \setminus \{q, r, s\}$, or
q is simple for $B \setminus \{p, r, s\}$, or
r is simple for $B \setminus \{p, q, s\}$, or
s is simple for $B \setminus \{p, q, r\}$.

6. *$\mathcal{R}$ does not delete all points of an object of B composed of four mutually 12-adjacent points.*

3.2.2 Point-based conditions

Similarly to the BCC case, our configuration-based sufficient conditions stated in Subsect. 3.2.1 are not useable for designing topology-preserving parallel reductions. The presented theorems examine the deletability of individual points.

Point-based conditions for *(18,12)* pictures

Just like in the case of the BCC grid, we propose a symmetric and an asymmetric *point-based* sufficient condition based on Theorems 3.2.1 and 3.2.2:

Theorem 3.2.8. [38, 37] *A parallel reduction is topology-preserving for an arbitrary picture $(\mathbb{F}, 18, 12, B)$ if each point $p \in B$ deleted by this reduction satisfies all of the following conditions:*

1. *Point p is simple for B.*

2. *Let q be a point in B such that*

   - *$q \in N_{12}(p)$,*
   - *q is simple for B.*

   Then p is simple for $B \setminus \{q\}$.

3. *Let q and r be two points in B such that*

   - *p, q, and r are mutually 12-adjacent,*

- *both points q and r are simple for B,*
- *q is simple for $B \setminus \{r\}$.*

Then p is simple for $B \setminus \{q, r\}$.

4. *Point p is not an element of an object consisting of four mutually 12-adjacent points.*
5. *Point p is not an element of a small object in which the points are not mutually 12-adjacent.*

Theorem 3.2.9. [38, 37] *A parallel reduction is topology-preserving for an arbitrary picture $(\mathbb{F}, 18, 12, B)$ if each point $p \in B$ deleted by that reduction satisfies all of the following conditions:*

1. *Point p is simple for B.*
2. *Let q be a point in B such that*
   - *$q \in N_{12}(p)$,*
   - *q is simple for B.*

   Then p is simple for $B \setminus \{q\}$, or $q \prec p$.
3. *Let q and r be two points in B such that*
   - *p, q, and r are mutually 12-adjacent,*
   - *both points q and r are simple for B,*
   - *q is simple for $B \setminus \{r\}$.*

   Then p is simple for $B \setminus \{q, r\}$, or p is not the smallest element of $\{p, q, r\}$.
4. *Point p is not the smallest element of an object consisting of four mutually 12-adjacent points.*
5. *Point p is not the smallest element of a small object in which the points are not mutually 12-adjacent.*

Point-based conditions for *(12,18)* pictures

Similarly to the $(18, 12)$ case, we state a pair of (symmetric and asymmetric) point-based conditions for $(12, 18)$ pictures:

Theorem 3.2.10. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 18, B)$ if each point $p \in B$ deleted by $\mathcal{R}$ satisfies all of the following conditions:*

1. *Point p is simple for B.*
2. *Let q be a point in B such that*
   - $q \in N_{18}(p)$*,*
   - q *is simple for* B*.*

   Then p is simple for $B \setminus \{q\}$.
3. *Let q and r be two points in B such that*
   - p*,* q*, and* r *are mutually* 12*-adjacent,*
   - *both points* q *and* r *are simple for* B*,*
   - q *is simple for* $B \setminus \{r\}$*.*

   Then p is simple for $B \setminus \{q, r\}$.
4. *Point p is not an element of an object consisting of four mutually 12-adjacent points.*

Theorem 3.2.11. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 18, B)$ if each point $p \in B$ deleted by $\mathcal{R}$ satisfies all of the following conditions:*

1. *Point p is simple for B.*
2. *Let q be a point in B such that*
   - $q \in N_{18}(p)$*,*
   - q *is simple for* B*.*

   Then p is simple for $B \setminus \{q\}$, or $q \prec p$.
3. *Let q and r be two points in B such that*
   - p*,* q*, and* r *are mutually* 12*-adjacent,*
   - *both points* q *and* r *are simple for* B*,*
   - q *is simple for* $B \setminus \{r\}$*.*

   Then p is simple for $B \setminus \{q, r\}$, or p is not the smallest element of the set $\{p, q, r\}$.
4. *Point p is not the smallest element of an object consisting of four mutually 12-adjacent points.*

Point-based conditions for *(12,12)* pictures

Here we also state a pair of symmetric and asymmetric point-based conditions:

Theorem 3.2.12. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 12, B)$ if each point $p \in B$ deleted by $\mathcal{R}$ satisfies all of the following conditions:*

1. *Point p is simple for B.*
2. *Let q be a point in B such that*
   - *$q \in N_{18}(p)$,*
   - *q is simple for B.*

   Then p is simple for $B \setminus \{q\}$.
3. *Let q and r be two points in B such that*
   - *p, q, and r are mutually 18-adjacent,*
   - *both points q and r are simple for B,*
   - *q is simple for $B \setminus \{r\}$.*

   Then p is simple for $B \setminus \{q, r\}$.
4. *Let q, r, and s be three points in B such that*
   - *p, q, r, and s form a cross,*
   - *all the three points q, r, and s are simple for B,*
   - *point q is simple for $B \setminus \{r\}$, point q is simple for $B \setminus \{s\}$, point r is simple for $B \setminus \{s\}$,*
   - *point q is simple for $B \setminus \{r, s\}$, or point r is simple for $B \setminus \{q, s\}$, or point s is simple for $B \setminus \{q, r\}$.*

   Then p is simple for $B \setminus \{q, r, s\}$.
5. *Point p is not an element of an object consisting of four mutually 12-adjacent points.*

Theorem 3.2.13. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, 12, 12, B)$ if each point $p \in B$ deleted by $\mathcal{R}$ satisfies all of the following conditions:*

1. *Point p is simple for B.*

2. *Let q be a point in B such that*

   - $q \in N_{18}(p)$*,*
   - q *is simple for* B*.*

   Then p is simple for $B \setminus \{q\}$, or $q \prec p$.

3. *Let q and r be two points in B such that*

   - p*,* q*, and* r *are mutually* 18*-adjacent,*
   - *both points* q *and* r *are simple for* B*,*
   - q *is simple for* $B \setminus \{r\}$*.*

   Then p is simple for $B \setminus \{q, r\}$, or p is not the smallest element of the set $\{p, q, r\}$.

4. *Let q, r, and s be three points in B such that*

   - p*,* q*,* r*, and* s *form a cross,*
   - *all the three points* q*,* r*, and* s *are simple for* B*,*
   - *point* q *is simple for* $B \setminus \{r\}$*, point* q *is simple for* $B \setminus \{s\}$*, point* r *is simple for* $B \setminus \{s\}$*,*
   - *point* q *is simple for* $B \setminus \{r, s\}$*, or point* r *is simple for* $B \setminus \{q, s\}$*, or point* s *is simple for* $B \setminus \{q, r\}$*.*

   Then p is simple for $B \setminus \{q, r, s\}$, or p is not the smallest element of the set $\{p, q, r, s\}$.

5. *Point p is not the smallest element of an object consisting of four mutually 12-adjacent points.*

Unified point-based conditions

We are now ready to establish a pair of unified conditions for the symmetric and asymmetric cases presented in the previous subsections.

Theorem 3.2.14. [37] *A parallel reduction $\mathcal{R}$ preserves the topology for an arbitrary picture $(\mathbb{F}, k, \bar{k}, B)$ ($(k, \bar{k}) \in \{(18, 12), (12, 18), (12, 12)\}$) if each point $p \in B$ deleted by $\mathcal{R}$ satisfies all of the following conditions:*

1. *Point p is simple for B.*

2. *Let q be a point in B such that*

   - *if* $(k, \bar{k}) = (18, 12)$*, then* $q \in N_{12}(p)$*, else* $q \in N_{18}(p)$*,*

- q *is simple for* B.

Then p *is simple for* $B \setminus \{q\}$.

3. *Let* q *and* r *be two points in* B *such that*

   - *if* $(k, \bar{k}) = (12, 12)$, *then* p, q, *and* r *are mutually* 12*-adjacent, else* p, q, *and* r *are mutually* 18*-adjacent,*
   - *both points* q *and* r *are simple for* B,
   - q *is simple for* $B \setminus \{r\}$.

   Then p *is simple for* $B \setminus \{q, r\}$.

4. *If* $(k, \bar{k}) = (18, 12)$, *then point* p *is not an element of a small object in which the points are not mutually* 12*-adjacent.*

5. *In the case of* $(k, \bar{k}) = (12, 12)$, *let* q, r, *and* s *be three points in* B *such that*

   - p, q, r, *and* s *form a cross,*
   - *all the three points* q, r, *and* s *are simple for* B,
   - *point* q *is simple for* $B \setminus \{r\}$, *point* q *is simple for* $B \setminus \{s\}$, *point* r *is simple for* $B \setminus \{s\}$,
   - *point* q *is simple for* $B \setminus \{r, s\}$, *or point* r *is simple for* $B \setminus \{q, s\}$, *or point* s *is simple for* $B \setminus \{q, r\}$.

   Then p *is simple for* $B \setminus \{q, r, s\}$.

6. *Point* p *is not an element of an object consisting of four mutually* 12*-adjacent points.*

Theorem 3.2.15. [37] *A parallel reduction* $\mathcal{R}$ *preserves the topology for an arbitrary picture* $(\mathbb{F}, k, \bar{k}, B)$ ($(k, \bar{k}) \in \{(18, 12), (12, 18), (12, 12)\}$) *if each point* $p \in B$ *deleted by* $\mathcal{R}$ *satisfies all of the following conditions:*

1. *Point* p *is simple for* B.

2. *Let* q *be a point in* B *such that*

   - *if* $(k, \bar{k}) = (18, 12)$, *then* $q \in N_{12}(p)$, *else* $q \in N_{18}(p)$,
   - q *is simple for* B.

   Then p *is simple for* $B \setminus \{q\}$, *or* $q \prec p$.

3. *Let* q *and* r *be two points in* B *such that*

- *if* $(k, \bar{k}) = (12, 12)$*, then* p*,* q*, and* r *are mutually* 12*-adjacent, else* p*,* q*, and* r *are mutually* 18*-adjacent,*
- *both points* q *and* r *are simple for* B*,*
- q *is simple for* $B \setminus \{r\}$*.*

Then p *is simple for* $B \setminus \{q, r\}$*, or* p *is not the smallest element of the set* $\{p, q, r\}$*.*

4. *If* $(k, \bar{k}) = (18, 12)$*, then point* p *is not the smallest element of a small object in which the points are not mutually* 12*-adjacent.*

5. *In the case of* $(k, \bar{k}) = (12, 12)$*, let* q*,* r*, and* s *be three points in* B *such that*

   - p*,* q*,* r*, and* s *form a cross,*
   - *all the three points* q*,* r*, and* s *are simple for* B*,*
   - *point* q *is simple for* $B \setminus \{r\}$*, point* q *is simple for* $B \setminus \{s\}$*, point* r *is simple for* $B \setminus \{s\}$*,*
   - *point* q *is simple for* $B \setminus \{r, s\}$*, or point* r *is simple for* $B \setminus \{q, s\}$*, or point* s *is simple for* $B \setminus \{q, r\}$*.*

   Then p *is simple for* $B \setminus \{q, r, s\}$*, or* p *is not the smallest element of the set* $\{p, q, r, s\}$*.*

6. *Point* p *is not the smallest element of an object consisting of four mutually* 12*-adjacent points.*

3.3 Generating topology-preserving parallel reductions

In this section, we provide some deletion rules of topologically correct reductions from our point-based conditions detailed in Subsect. 3.1.2 and 3.2.2.

3.3.1 Derived reductions on the BCC grid

From our point-based conditions reported in Subsect. 3.1.2, we can directly yield deletion rules of topology-preserving parallel reductions for the BCC grid:

Definition 3.3.1. *A black point is deleted by the parallel reduction* $\mathcal{R}_{\text{symm\_}(14,14)}$ *if it satisfies all conditions of Theorem 3.1.2 (i.e., symmetric point-based condition for topology-preserving parallel reductions).*

Definition 3.3.2. *A black point is deleted by the parallel reduction* $\mathcal{R}_{\text{asymm\_(14,14)}}$ *if it satisfies all conditions of Theorem 3.1.3 (i.e., asymmetric point-based condition for topology-preserving parallel reductions).*

The *support* of an image operator $\mathcal{O}$ is the minimal set of points whose values determine whether a point is changed by $\mathcal{O}$ [29]. The support $\mathcal{S}^{\mathbb{B}}_{\text{symm}}$ of the symmetric parallel reduction contains 64 points, and the count of points is 47 in the support $\mathcal{S}^{\mathbb{B}}_{\text{asymm}}$ of the asymmetric parallel reduction (see Fig. 3.1). We can observe that these supports centered at a point p can be expressed as follows:

$$\begin{aligned} \mathcal{S}^{\mathbb{B}}_{\text{symm}} &= \bigcup_{q \in N_{14}(p)} N_{14}(q) \ , \\ \mathcal{S}^{\mathbb{B}}_{\text{asymm}} &= N_{14}(p) \cup \bigcup_{q \in N_{14}(p), p \prec q} N_{14}(q) \ . \end{aligned}$$

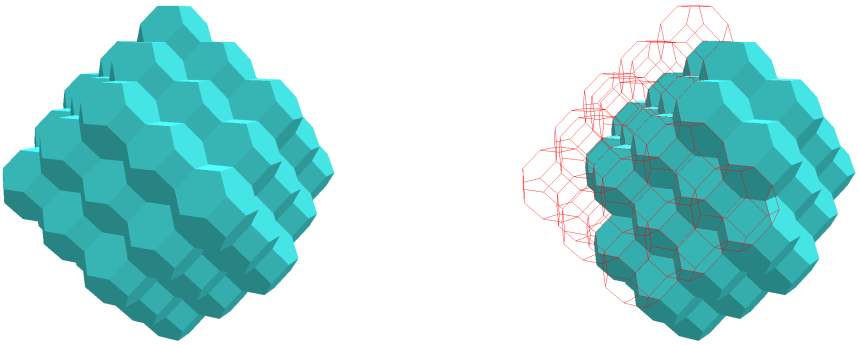


Figure 3.1: *Sets* $\mathcal{S}^{\mathbb{B}}_{\text{symm}}$ *(left) and* $\mathcal{S}^{\mathbb{B}}_{\text{asymm}}$ *(right) containing, respectively, 64 and 47 voxels.*

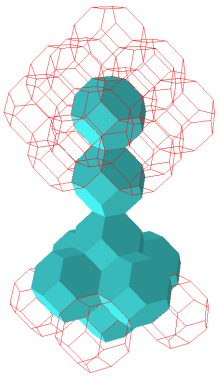
By Theorems 3.1.2 and 3.1.3, it is obvious that both derived parallel reductions $\mathcal{R}_{\text{symm\_(14,14)}}$ and $\mathcal{R}_{\text{asymm\_(14,14)}}$ are topology-preserving. Notice that reduction $\mathcal{R}_{\text{asymm\_(14,14)}}$ can delete more points from a picture than reduction $\mathcal{R}_{\text{symm\_(14,14)}}$ does. Figure 3.2 illustrates the difference between these two derived reductions.

3.3.2 Derived reductions on the FCC grid

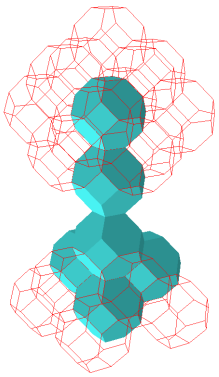
Similarly to the BCC case (see Sect. 3.3.1), we can derive deletion rules of topology-preserving parallel reductions from our point-based conditions described in Subsect. 3.2.2.

Definition 3.3.3.

- *The parallel reduction* $\mathcal{R}_{\text{symm\_(18,12)}}$ *deletes all black points from* $(18, 12)$ *pictures that satisfy all conditions of Theorem 3.2.8.*



$\mathcal{R}_{\text{symm}\_(14,14)}$ (8)



$\mathcal{R}_{\text{asymm}\_(14,14)}$ (6)

Figure 3.2: *Results of the two generated reductions for a synthetic object containing* 25 *voxels. Numbers in parentheses are the counts of the non-deleted voxels, and the resulting objects are superimposed on the original one.*

- *The parallel reduction* $\mathcal{R}_{\text{symm}\_(12,18)}$ *deletes all black points from* $(12, 18)$ *pictures that satisfy all conditions of Theorem 3.2.10.*
- *The parallel reduction* $\mathcal{R}_{\text{symm}\_(12,12)}$ *deletes all black points from* $(12, 12)$ *pictures that satisfy all conditions of Theorem 3.2.12.*
- *The parallel reduction* $\mathcal{R}_{\text{asymm}\_(18,12)}$ *deletes all black points from* $(18, 12)$ *pictures that satisfy all conditions of Theorem 3.2.9.*
- *The parallel reduction* $\mathcal{R}_{\text{asymm}\_(12,18)}$ *deletes all black points from* $(12, 18)$ *pictures that satisfy all conditions of Theorem 3.2.11.*
- *The parallel reduction* $\mathcal{R}_{\text{asymm}\_(12,12)}$ *deletes all black points from* $(12, 12)$ *pictures that satisfy all conditions of Theorem 3.2.13.*

Let $\mathcal{S}^{\mathbb{F}}_{\text{symm}}$ and $\mathcal{S}^{\mathbb{F}}_{\text{asymm}}$ denote the support of the symmetric and asymmetric reductions provided in Def. 3.3.3. We can observe that the support of the symmetric contains 84 points, and the count of points is 64 in the support of the asymmetric reductions (see Fig. 3.3). Similarly to the BCC case, these supports centered at a point p can be expressed as follows:

$$\begin{aligned} \mathcal{S}^{\mathbb{F}}_{\text{symm}} &= \bigcup_{q \in N_{18}(p)} N_{18}(q) \,, \\ \mathcal{S}^{\mathbb{F}}_{\text{asymm}} &= N_{18}(p) \cup \bigcup_{q \in N_{18}(p), p \prec q} N_{18}(q) \,. \end{aligned}$$

By Theorems 3.2.14, and 3.2.15, it is obvious that all the six derived parallel reductions are topology-preserving. It can be readily seen that reduction $\mathcal{R}_{\text{asymm}\_(k,\bar{k})}$ can delete more points from a picture than $\mathcal{R}_{\text{symm}\_(k,\bar{k})}$ does ($(k, \bar{k}) \in \{(18, 12),$

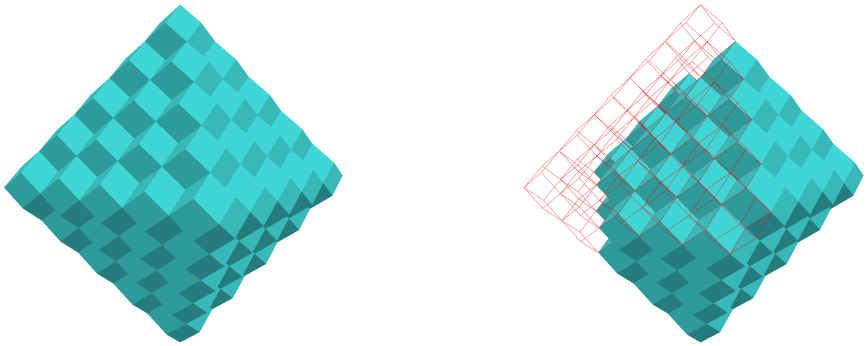


Figure 3.3: *Sets $\mathcal{S}^{\mathbb{F}}_{\text{symm}}$ (left) and $\mathcal{S}^{\mathbb{F}}_{\text{asymm}}$ (right) containing, respectively, 84 and 64 voxels.*

$(12, 18), (12, 12)\}$). Figure 3.4 illustrates the difference between these derived reductions. Notice that only $\mathcal{R}_{\text{symm\_(18,12)}}$ and $\mathcal{R}_{\text{asymm\_(18,12)}}$ does not affect the two middle voxels connected by a vertex, because they are non-simple only in $(18, 12)$ pictures.

3.4 Concluding remarks

In this chapter, the results belonging to the *first thesis group* are presented. We examined the issues of checking whether a parallel reduction preserves the topology and how to construct topologically correct algorithms on the BCC and FCC grids. We proposed configuration-based and point-based sufficient conditions for topology-preserving parallel reductions acting on, in total, four types of pictures of these unconventional 3D grids. Our point-based conditions directly provide deletion rules of topology-preserving parallel reductions, a safe technique for designing topology-preserving parallel thinning algorithms.

In future work, we are to construct sufficient conditions for topology-preserving parallel reductions on further non-conventional 3D grids, such as the diamond grid [77].

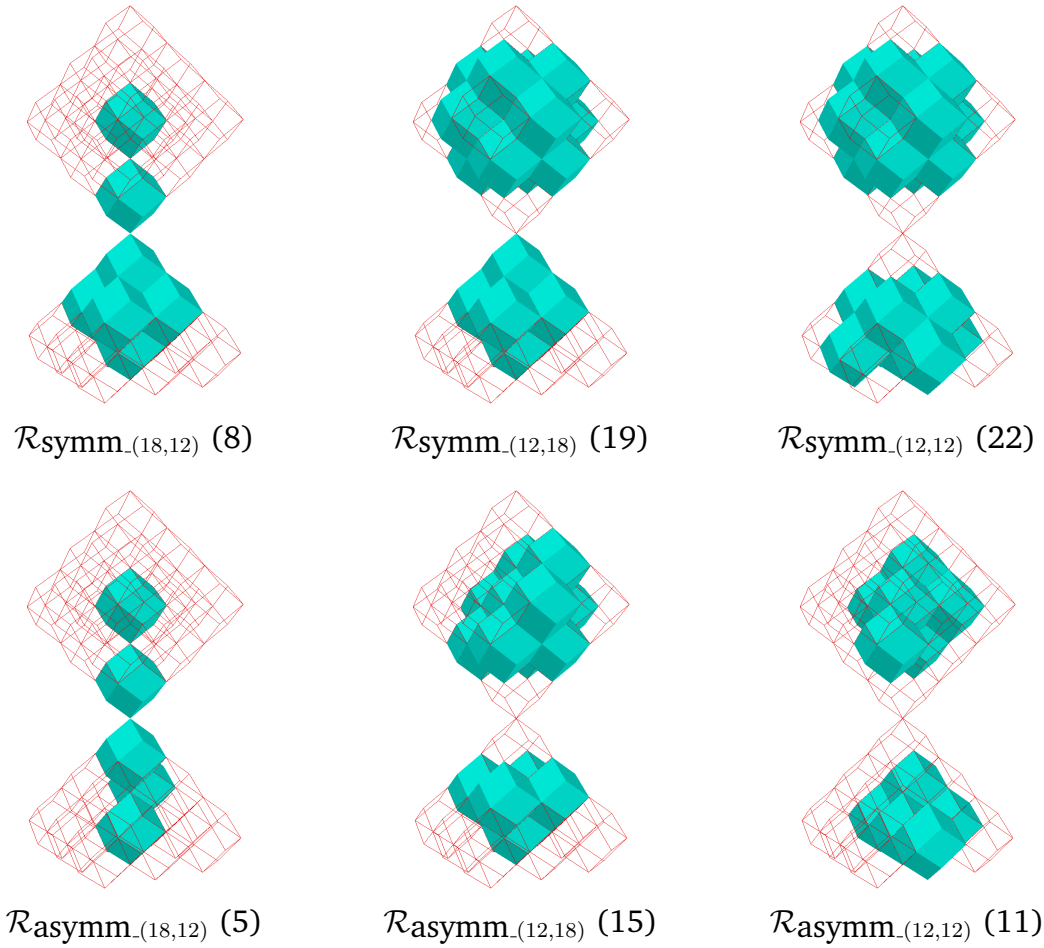


Figure 3.4: *Results of the six generated reductions for a synthetic object containing* 33 *voxels. Note that the original object is* 18*-connected, but it contains two* 12*-components. Numbers in parentheses are the counts of the non-deleted voxels, and the resulting objects are superimposed on the original one.*

Chapter 4

Parallel kernel-thinning algorithms

In this chapter, we derive thinning algorithms from some of our proposed sufficient conditions discussed in chapter 3. In these methods we do not use any geometric constraint. We prove that they are capable of producing the topological kernels of the input objects. Hence, these algorithms deserve to be called kernel-thinning methods. The publications associated with this work are [35, 36].

4.1 Fully parallel algorithms

In this section, we construct fully parallel kernel-thinning algorithms by iteratively applying the asymmetric reductions introduced in Sect. 3.3.

It is important to note that we could also establish similar algorithms from the symmetric reductions. However, the symmetric reductions are not suitable for kernel-thinning, since they cannot produce topological kernels. For example, if a small object is formed by more than one black point, then the latter type of reductions will leave it unchanged instead of shrinking it to an isolated point.

4.1.1 Algorithm on the BCC grid

In order to construct a kernel-thinning method, we define the following deletion criterion:

Definition 4.1.1. *A black point is said to be $\mathcal{R}_{asymm\_(14,14)}$-deletable if reduction $\mathcal{R}_{asymm\_(14,14)}$ (see Def. 3.3.2) classifies it as deletable.*

We are now ready to present our kernel-thinning algorithm, see Alg. 4.1.

Next, we show that the proposed algorithm (see Alg. 4.1) is capable of producing topological kernels.

Proposition 4.1.1. *If an object in a $(14, 14)$ picture contains four mutually 14-adjacent points then all of its points are simple.*

Algorithm 4.1: $FP\text{-}KT\_(14,14)$

Input: picture $(\mathbb{B}, 14, 14, X)$
Output: picture $(\mathbb{B}, 14, 14, Y)$
// initialize resulted black points
1 $Y \leftarrow X$
2 **repeat**
  // initialize deleted points
3 $D \leftarrow \emptyset$
  // collect deletable points
4 $D \leftarrow \{\, p \in Y \mid p$ is $\mathcal{R}_{\text{asymm}\_(14,14)}$-deletable $\}$
  // parallel deletion
5 $Y \leftarrow Y \setminus D$
6 **until** $D = \emptyset$

The above statement can be easily proved by verifying the conditions of Theorem 2.3.1.

Theorem 4.1.1. *If a picture $\mathcal{P} = (\mathbb{B}, 14, 14, B)$ contains one or more simple points, then reduction $\mathcal{R}_{asymm\_(14,14)}$ deletes at least one point from $\mathcal{P} = (\mathbb{B}, 14, 14, B)$.*

Proof. Let $Q \subseteq B$ be the set of simple points in picture $\mathcal{P}$, and let us assume that $Q \neq \emptyset$. Furthermore, let $p \in Q$ such that for any $q \in Q \setminus \{p\}$, $q \prec p$, i.e., let p be the lexicographically greatest element in Q. (Note that p uniquely exists since '$\prec$' is a total ordering relation.)

Henceforth, we prove that reduction $\mathcal{R}_{\text{asymm}\_(14,14)}$ deletes p, i.e., p satisfies each condition of that reduction. For this aim it is sufficient to verify that all the criteria of the asymmetric point-based condition (see Theorem 3.1.3) hold:

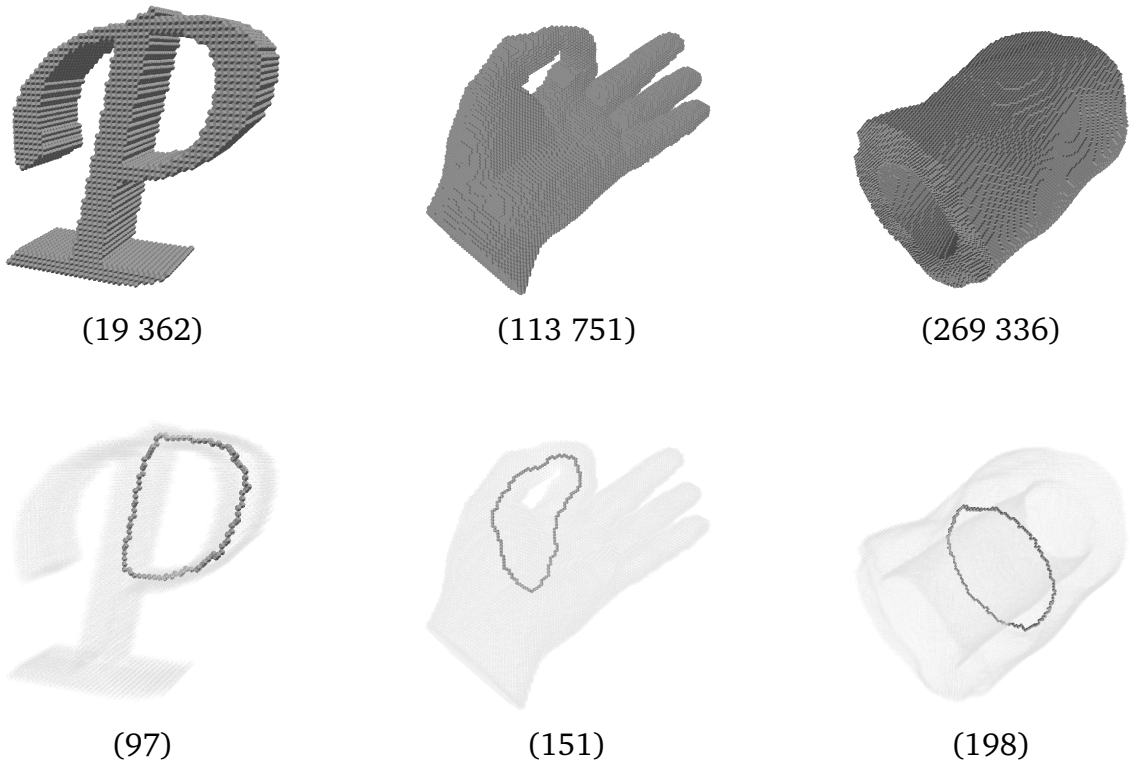
- Condition 1 is satisfied, since $p \in Q$.
- Condition 2 is satisfied, since if there exists point $q \in Q$ mentioned in that condition, then $q \prec p$.
- Condition 3 is fulfilled, since if there exists points $q, r \in Q$ mentioned in that condition, then $q \prec p$ and $r \prec p$, hence p is not the smallest element in set $\{p, q, r\}$.
- Condition 4 is fulfilled, since if $\{p, q, r, s\}$ is the object mentioned in that condition, then not only p but also q, r, and s are simple by Proposition 4.1.1, i.e., $q, r, s \in Q$, which implies $q \prec p$, $r \prec p$ and $s \prec p$. Hence, again, p is not the smallest element of that object.

Herewith, we can conclude that point p is deleted by $\mathcal{R}_{\text{asymm}\_(14,14)}$, and the theorem is proved. □

Corollary 4.1.1. *Algorithm* $FP\text{-}KT\_(14, 14)$ *produces topological kernels.*

Proof. By Theorem 4.1.1, each iteration step of our algorithms deletes at least one simple point from a picture that contains one or more simple points. Furthermore, it is obvious that the set of simple points in a picture is finite. Consequently, the outputs of our topology-preserving algorithms do not contain any simple point, which implies the statement of the corollary. □

Figure 4.1 shows three examples to illustrate the constructed thinning algorithm's behaviour. Notice that the extracted topological kernels are correcly placed at the center of the objects.



(19 362) (113 751) (269 336)

(97) (151) (198)

Figure 4.1: *Topological kernels (bottom row) of three test objects (upper row) produced by the proposed algorithm. The numbers in parentheses under the input objects are the counts of object points. Numbers in parentheses under the produced topological kernels indicate the counts of skeletal points.*

4.1.2 Algorithms on the FCC grid

Similarly to the BCC case, we specify the following deletion criterion for the FCC grid:

Definition 4.1.2. *A black point is said to be $\mathcal{R}_{asymm\_(k,\bar{k})}$-deletable if reduction $\mathcal{R}_{asymm\_(k,\bar{k})}$ (see Def. 3.3.3) classify it as deletable* $((k,\bar{k}) \in \{(18,12),(12,18),(12,12)\})$.

We are now ready to present our kernel-thinning algorithms on the FCC grid as well:

Algorithm 4.2: $FP\text{-}KT\_(k,\bar{k})$ $(\ (k,\bar{k}) \in \{(18,12),(12,18),(12,12)\}\ )$

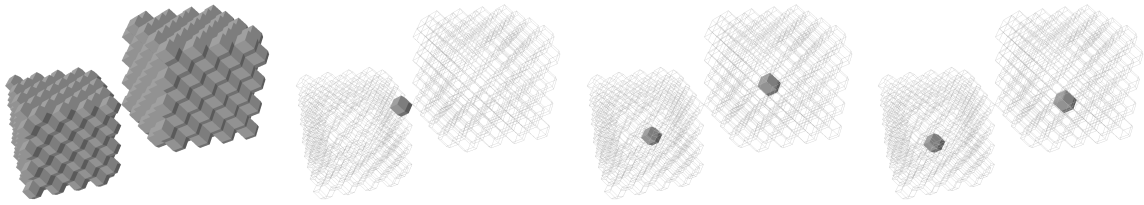
Input: picture $(\mathbb{F},k,\bar{k},X)$
Output: picture $(\mathbb{F},k,\bar{k},Y)$
`// initialize resulted black points`
1 $Y \leftarrow X$
2 **repeat**
`// initialize deleted points`
3 $D \leftarrow \emptyset$
`// collect deletable points`
4 $D \leftarrow \{\ p \in Y \mid p$ is $\mathcal{R}_{\text{asymm}\_(k,\bar{k})}$-deletable $\}$
`// parallel deletion`
5 $Y \leftarrow Y \setminus D$
6 **until** $D = \emptyset$

The three proposed algorithms (see Alg. 4.2) are able to produce topological kernels, which can be proved by the same principle as for the BCC case.

Theorem 4.1.2. [36] *If a picture $\mathcal{P} = (\mathbb{F},k,\bar{k},B)$ contains one or more simple points, then reduction $\mathcal{R}_{asymm\_(k,\bar{k})}$ deletes at least one point from $\mathcal{P}$* $(\ (k,\bar{k}) \in \{(18,12),(12,18),(12,12)\}\ )$.

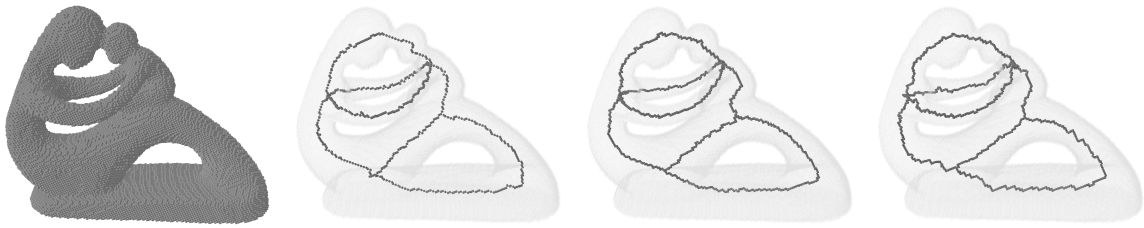
Corollary 4.1.2. [36] *All the three proposed algorithms (see Algorithm 4.2) produce topological kernels.*

We present three illustrative examples below (see Figs. 4.2–4.4). Similarly to Fig. 4.1, the numbers in parentheses under the original objects and the topological kernels indicate the counts of object points and skeletal points, respectively. Notice that while algorithms $FP\text{-}KT\_(12,12)$ and $FP\text{-}KT\_(12,18)$ transform each of the two solid cubes (with no holes nor cavities) in Fig. 4.2 to an isolated point, the topological kernel produced by algorithm $FP\text{-}KT\_(18,12)$ consists only of a single voxel in this case. The reason for this difference is that the only 18-connected object in the above test image coincides with two 12-connected objects. The produced topological kernels of holey objects contain one-voxel thin closed curves as it is illustrated in Figs. 4.3–4.4.



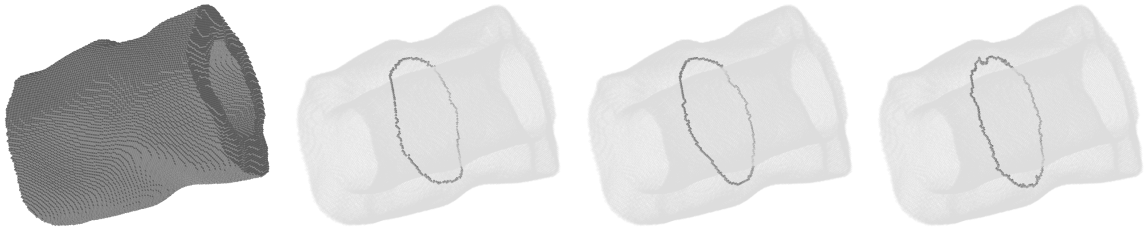
original object (512) | $FP\text{-}KT\_(18,12)$ (1) | $FP\text{-}KT\_(12,12)$ (2) | $FP\text{-}KT\_(12,18)$ (2)

Figure 4.2: *Topological kernels of two cubes produced by algorithms* $FP\text{-}KT\_(18,12)$*,* $FP\text{-}KT\_(12,12)$*, and* $FP\text{-}KT\_(12,18)$*.*



original object (200 424) | $FP\text{-}KT\_(18,12)$ (486) | $FP\text{-}KT\_(12,12)$ (630) | $FP\text{-}KT\_(12,18)$ (649)

Figure 4.3: *Topological kernels of a 'fertility' object produced by algorithms* $FP\text{-}KT\_(18,12)$*,* $FP\text{-}KT\_(12,12)$*, and* $FP\text{-}KT\_(12,18)$*.*



original object (538 800) | $FP\text{-}KT\_(18,12)$ (230) | $FP\text{-}KT\_(12,12)$ (246) | $FP\text{-}KT\_(12,18)$ (274)

Figure 4.4: *Topological kernels of a deformed tube produced by algorithms* $FP\text{-}KT\_(18,12)$*,* $FP\text{-}KT\_(12,12)$*, and* $FP\text{-}KT\_(12,18)$*.*

4.2 Subfield-based algorithms on the BCC grid

In this section, two subfield-based parallel kernel-thinning algorithms are reported. For this purpose, we propose the partitions of $\mathbb{B}$ into four and eight subfields as shown in Fig. 4.5.

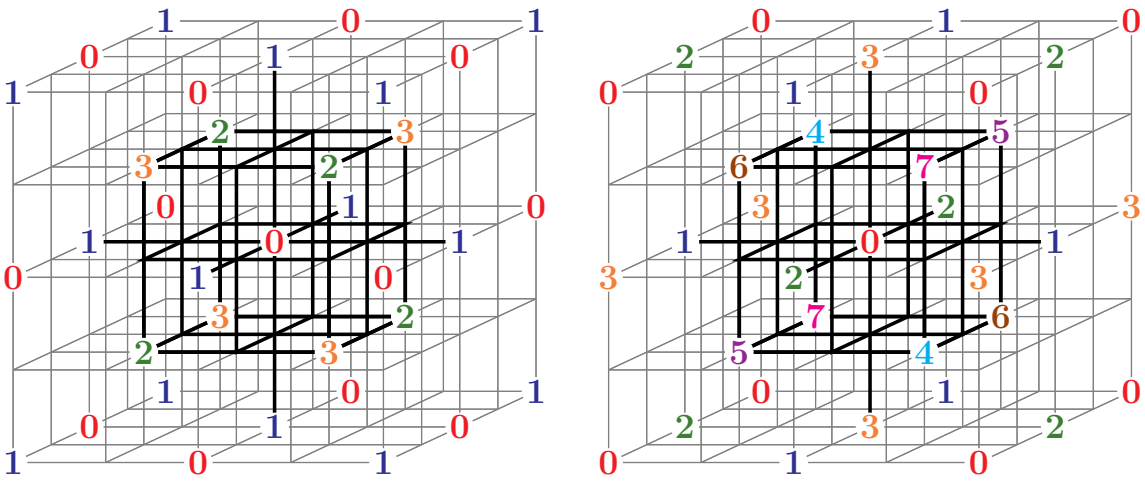


Figure 4.5: *Partition of $\mathbb{B}$ into four (left) and eight subfields (right). All points marked 'i' are in subfield $S_4(i)$ $(i = 0, 1, 2, 3)$, and in $S_8(i)$ $(i = 0, 1, \ldots, 7)$, respectively. (Note that unmarked elements in $\mathbb{Z}^3$ are not points in $\mathbb{B}$.)*

Let us state now an important property of these partitionings:

Proposition 4.2.1. *If $p \in S_k(i)$ $(k = 4, 8;\ i = 0, 1, \ldots, k-1)$ and $q \in N_{14}(p)$, $q \notin S_k(i)$.*

It is obvious by careful examination of Fig. 4.5.

By Proposition 4.2.1, our configuration-based sufficient condition (see Theorem 3.1.1) can by simplified for reductions of subfield-based thinning algorithms:

Theorem 4.2.1. [35] *A reduction that deletes a subset of $S_k(i) \cap B$ $(k = 4, 8;\ i = 0, 1, \ldots, k-1)$ from picture $(\mathbb{B}, 14, 14, B)$ is topology-preserving if only simple points are deleted.*

With the help of Proposition 4.2.1, it can be readily seen that there is no mutually 14-adjacent pair (neither triplet nor quadruple) of points in the same subfield according to the proposed partitionings. Thus the last three conditions of Theorem 3.1.1 are irrelevant here.

In Algorithm 4.3 the kernel of the **repeat** loop corresponds to one iteration step that comprises k $(k = 4, 8)$ subiterations (i.e., parallel reductions) in which the k subfields (see Fig. 4.5) are alternatively activated. The thinning process is terminated if no points are deleted within an iteration step (i.e., $D = \emptyset$). Note that some black points may become simple directly after a subiteration process, which may overdeform an object, and may cause the resulting skeletal points not to be centered on the input object. To prevent this phenomenon, we must ensure that only those black points are visited that were border points at the beginning of the actual iteration. That is why the border points are collected into a set before the first subiteration.

Algorithm 4.3: SF-k-KT $(k = 4, 8)$

Input: picture $(\mathbb{B}, 14, 14, X)$
Output: picture $(\mathbb{B}, 14, 14, Y)$
// initialize resulted black points
1 $Y \leftarrow X$
2 **repeat**
  // border tracking
3 $S \leftarrow \{\, p \mid p$ is a border point for $Y \,\}$
  // initialize deleted points
4 $D \leftarrow \emptyset$
  // k subiterations
5 **for** $i \leftarrow 0$ **to** $k - 1$ **do**
    // collect deletable points
6    $D(i) \leftarrow \{\, p \mid p \in S_k(i) \cap S$ and simple for $Y \,\}$
    // parallel deletion
7    $Y \leftarrow Y \setminus D(i)$
    // collect deleted points
8    $D \leftarrow D \cup D(i)$
9 **until** $D = \emptyset$

Notice that only simple points can be deleted in each subiteration of all the proposed subfield-based parallel thinning algorithms (see Algorithm 4.3). Thus we can state the following:

Theorem 4.2.2. [35] *Algorithms SF-4-KT and SF-8-KT are both topology-preserving.*

Lastly, since the output of Algorithm 4.3 does not contain any simple point, the presented algorithms are indeed kernel-thinning ones:

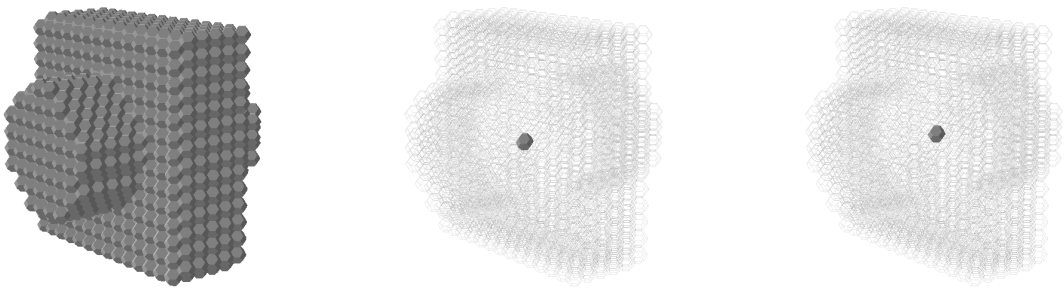
Proposition 4.2.2. [35] *Algorithms SF-4-KT and SF-8-KT extract the topological kernel of the input object.*

The proposed two algorithms were tested on objects of different shapes. We present ten illustrative examples below (see Figs. 4.6–4.15). The numbers in parentheses under the original images are the counts of object points. The pairs of numbers in parentheses under the produced topological kernels indicate the count of object points and the number of required iterations, respectively. Notice that the algorithms transform the synthetic object with no hole in Figure 4.6 to a single voxel, while the topological kernels of holey objects in Figs. 4.7–4.15 are one-voxel thin connected closed curve segments. Notice that in Fig. 4.15, our algorithms produced exactly the same skeleton-like features, and even the corresponding number of iterations is equal.

Our implementations were run on a usual desktop (HP ProDesk 400 G4; 3.20 GHz Intel Core i5-6500; Windows 10 x64) and written in C++. The `std::chrono` library was used for runtime measurement. Table 4.1 contains the computation times of our algorithms for each object shown in Figs. 4.6–4.15. Note that reading the input image and writing the output image were not considered here. We can observe that the computational cost does not depend on the number of subfields, due to the fact that any border point belongs to exactly one subfield. Thus, any of those points is examined exactly once in each iteration.

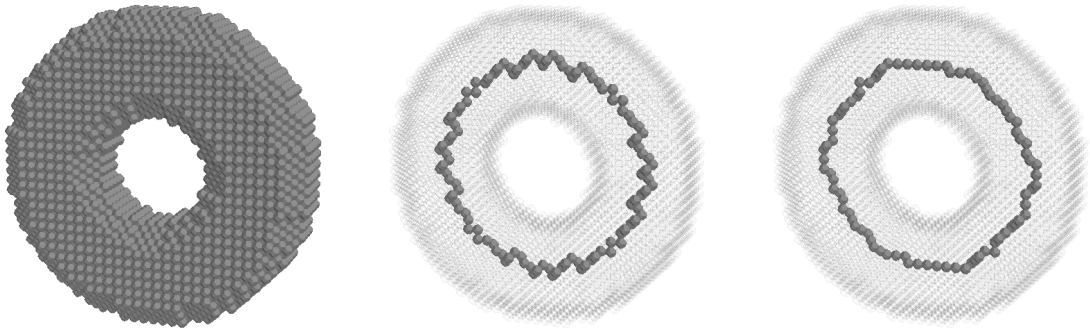
Table 4.1: *Computation times (in millisec.) of Algorithm 4.3.*

Test object	SF-4-KT	SF-8-KT
Syntetic object	0.993	0.992
Torus	1.994	1.995
Gear	36.869	36.889
Helicopter	38.890	38.895
Hand	31.947	32.897
Letter A	9.942	9.954
'Fertility'	29.908	29.920
Cube with 1 hole	37.866	37.890
Cube with 2 holes	33.914	33.909
Cube with 3 holes	31.891	31.902



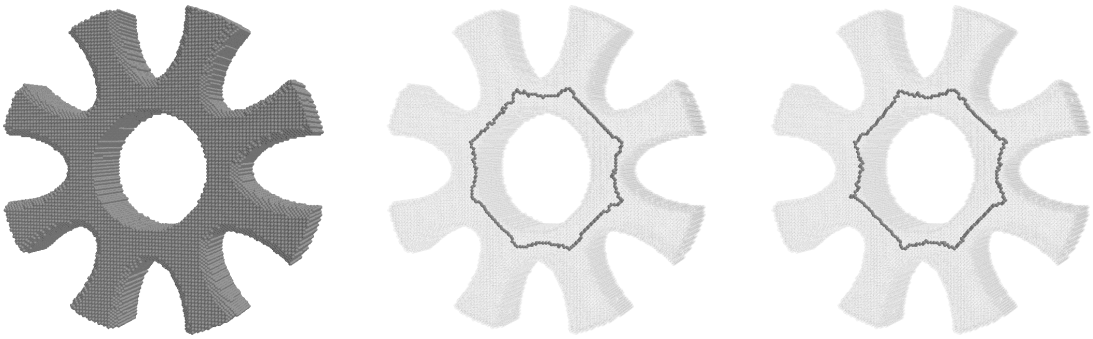
(2 806) (1, 8) (1, 10)

Figure 4.6: *A $32 \times 32 \times 32$ image of a syntetic object (left) and its topological kernels produced by algorithms SF-4-KT (middle) and SF-8-KT (right).*



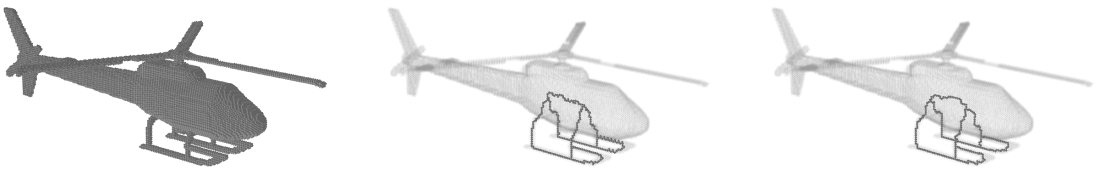
(10 028) (112, 7) (84, 8)

Figure 4.7: *A* $64 \times 64 \times 19$ *image of a torus (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*



(149 142) (218, 25) (234, 27)

Figure 4.8: *A* $45 \times 191 \times 191$ *image of a gear (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*

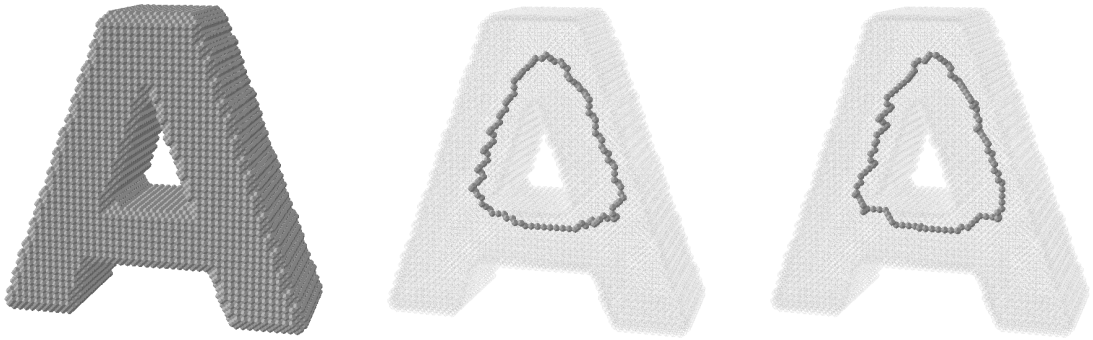


(68 432) (304, 85) (272, 72)

Figure 4.9: *A* $381 \times 103 \times 255$ *image of a helicopter (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*

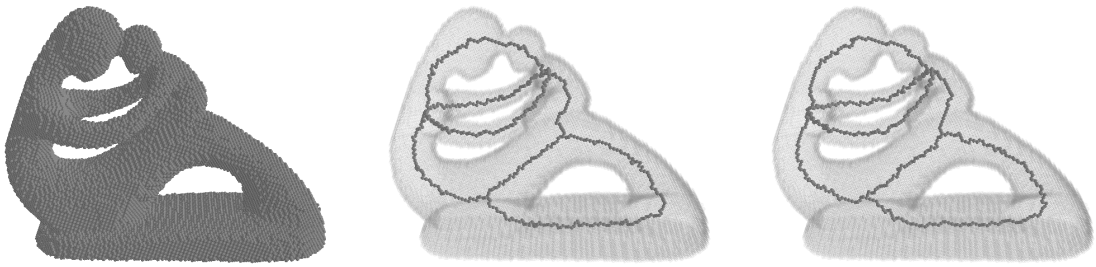
(113 751) (151, 51) (165, 43)

Figure 4.10: *A* $191\times96\times114$ *image of a hand (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*



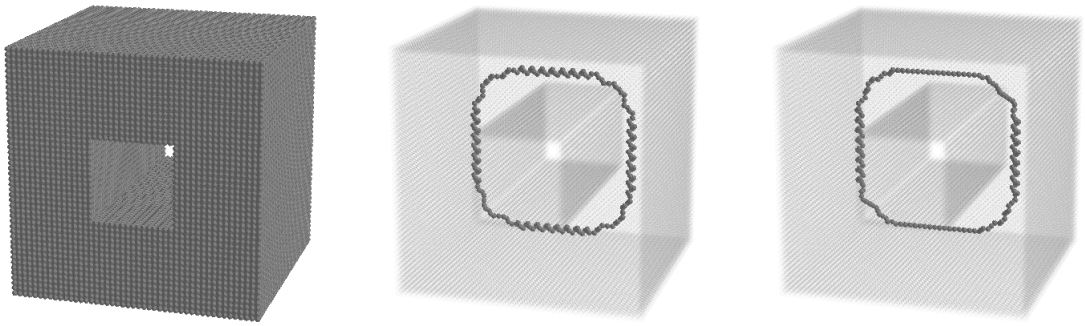
(41 964) (109, 17) (109, 20)

Figure 4.11: *A* $100\times100\times40$ *image of a letter (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*



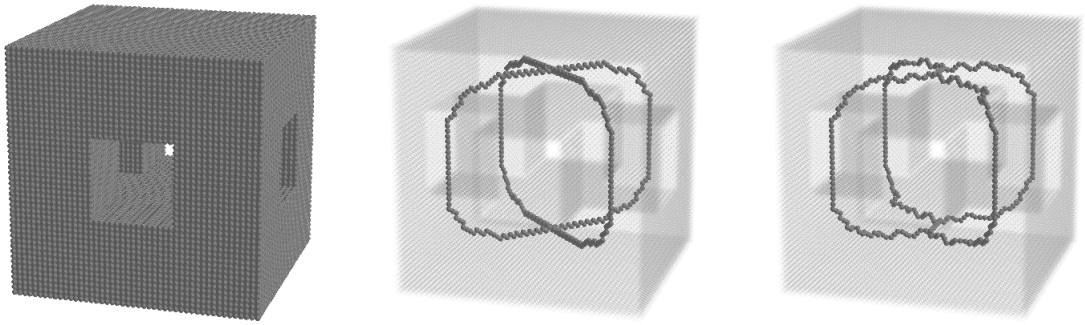
(100 188) (471, 18) (459, 17)

Figure 4.12: *A* $138\times70\times189$ *image of 'fertility' (left) and its topological kernels produced by algorithms* SF*-4-*KT *(middle) and* SF*-8-*KT *(right).*



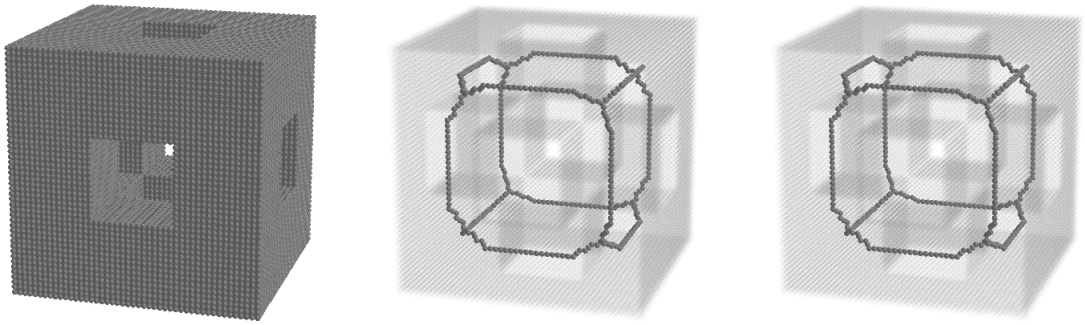
(178 808) (184, 20) (156, 21)

Figure 4.13: *A* $93 \times 93 \times 93$ *image of a holey cube (left) and its topological kernels produced by algorithms* SF-4-KT *(middle) and* SF-8-KT *(right).*



(163 928) (330, 18) (320, 18)

Figure 4.14: *A* $93 \times 93 \times 93$ *image of a holey cube with more holes (left) and its topological kernels produced by algorithms* SF-4-KT *(middle) and* SF-8-KT *(right).*



(149 048) (368, 11) (368, 11)

Figure 4.15: *A* $93 \times 93 \times 93$ *image of a holey cube with even more holes (left) and its topological kernels produced by algorithms* SF-4-KT *(middle) and* SF-8-KT *(right).*

4.3 Concluding remarks

This chapter described some research belonging to the *second thesis group*, in which we proposed fully parallel kernel-thinning algorithms. One and three of them act on pictures sampled on the BCC and FCC grids, respectively. Furthermore, two subfield-based kernel-thinning algorithms on the BCC grid are reported. We showed that all the constructed thinning methods are capable of producing topological kernels. The topological correctness of our algorithms are guaranteed by the safe designing technique: a sufficient condition for topology-preserving reductions is combined with a parallel thinning strategy.

In future work, we plan to combine these thinning schemes with various geometric constraints to establish a variety of parallel 3D curve-, and surface-thinning algorithms acting on these non-conventional grids.

Chapter 5

Distance-oriented sequential thinning

In this chapter, we focus on sequential thinning algorithms combined with distance information. In Section 5.1, we recall the details of distance transform on the considered grids. Section 5.2 describes surface skeletonization algorithms. First, we review Strand's algorithms working on $(14, 14)$ pictures of the BCC grid and $(12, 12)$ pictures of the FCC grid [72]. To the best of our knowledge, there is not any other similar result in the later literature to extract the medial surface of objects on these alternative grids. We show that these methods have nonlinear time complexity. Then, we present two altered versions for both grids, which are proved to be linear. Finally, in Section 5.3, two distance-driven curve-thinning method is proposed, one for $(14, 14)$ pictures of the BCC grid, and one for $(18, 12)$ pictures of the FCC grid. This chapter is based on papers [32, 33, 34].

5.1 Distance transform on the BCC and FCC grids

As we mentioned during the discussion of distance-based skeletonization techniques (see Subsect. 2.4.3), we denote the computed distance map of a picture by DT. The most frequently used distance functions are neighbor-based distances [52], and the Euclidean distance – denoted by d_e. In $\mathbb{B}$, neighbor-based distances d_8 and d_{14} can be applied, while in $\mathbb{F}$, distances d_{12} and d_{18} are taken into consideration. For better approximations of the exact Euclidean distance, the path of the minimal sum of weighted moves called as *Chamfer distances* are calculated [12]. Computation of the *Chamfer DT* requires two raster scans [75], while the non-errorfree *Euclidean distance transform* (EDT in short) takes four scans on the BCC and FCC grid as well [75]. Note that, in the implementation of the discussed methods, we computed the square of the Euclidean distance transform (i.e, we omitted the square root operation), since that was more beneficial for our purpose.

Let $\langle a, b\rangle^{\mathbb{B}}$ denote the general *Chamfer mask* for the BCC grid, where a and b are the weights assigned to all points in $N_8(p)$ and $N_{14}(p) \setminus N_8(p)$, respectively. Note

that distances d_8 and d_{14} are equivalent to the $\langle 1, 2\rangle^{\mathbb{B}}$ and $\langle 1, 1\rangle^{\mathbb{B}}$ Chamfer distances, respectively. Similarly to the BCC case, $\langle a, b, c\rangle^{\mathbb{F}}$ denotes the general Chamfer mask for the FCC grid, where a and b are the weights assigned to all points in $N_{12}(p)$ and $N_{18}(p)\backslash N_{12}(p)$, respectively, and weight c is assigned to all q points such that $d_e(p, q) = \sqrt{6}$, see Fig. 5.1. If c is not used, notion $\langle a, b\rangle^{\mathbb{F}}$ is applied. Notice that distances d_{12} and d_{18} are equivalent to the $\langle 1, 2\rangle^{\mathbb{F}}$ and $\langle 1, 1\rangle^{\mathbb{F}}$ Chamfer distances, respectively. Some weight combinations for both grids are presented in [25, 75].

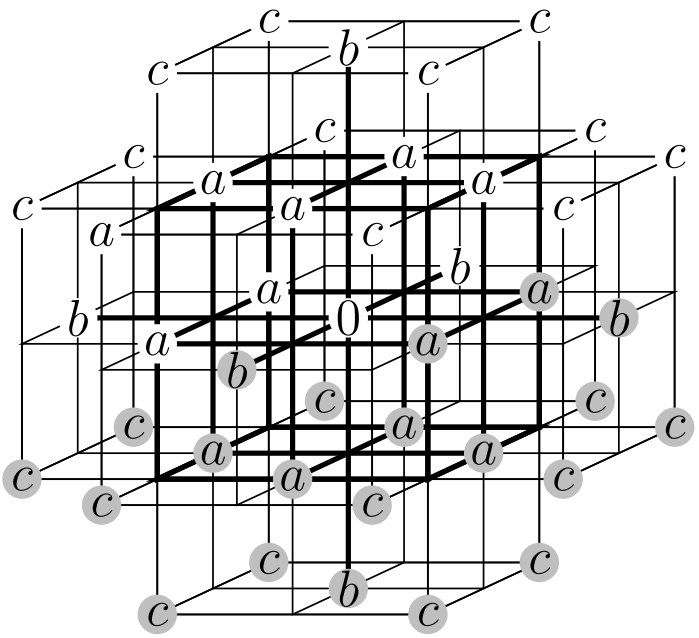


Figure 5.1: *Chamfer mask for distance transform on $\mathbb{F}$, in which the three weights $0 < a \leq b < c$ are taken into consideration. There are 42 examined positions in total, and they all fit into the $5 \times 5 \times 5$ local environment of the centered point marked 0. Note that unmarked elements of $\mathbb{Z}^3$ are not in $\mathbb{F}$. Neighbors labeled with white and lightgray background are investigated in the forward and backward scan during distance transform, respectively.*

5.2 Surface-thinning

In this section, we study Strand's surface-thinning algorithms, then we discuss their modified variants for both considered grids, which are faster and – according to our experiments – less sensitive to the visiting order of border points in the sequential thinning phase.

To completely understand these methods, we need to define the concept of a local maximum, the center of a maximal inscribed ball, and two opposite neighbors of a

point. Point p is called a *local maximum* if for any point $q \in N_i(p)$, $DT(p) \geq DT(q)$, where i is the considered neighborhood. In a Chamfer distance map, point p is a *center of maximal ball* (*CMB*, for short) [2] if $DT(q) < DT(p) + w$ for any point q concerned in the Chamfer mask, where w is the weight that corresponds to q in the Chamfer mask. Note that local maxima coincide with CMBs in case of neighbor-based distances. Two points $q, r \in B$ are opposite neighbors of $p \in B$ if both q and r are adjacent to p, and $q - p = p - r$.

It is showed that the original object can be completely reconstructed from the skeleton if all the CMBs are included in the produced medial surface [78].

5.2.1 Strand's algorithms

The sequential thinning method proposed by Strand [72] assumes $(14, 14)$ pictures on the BCC grid and $(12, 12)$ pictures on the FCC grid. These procedures rely on the d_{14} and d_{12} distance functions for the BCC and FCC grids, respectively. Both variants consider CMBs as safe skeletal (anchor) points, and can produce surface skeletons by preserving non-simple points and surface edge points. A point $p \in B$ is called a *surface edge point* if the following conditions all hold:

- it has two opposite black neighbors in $N_k(p)$ $(k \in \{14, 12\})$, and
- there is no point $s \in N_k(p) \cap B$ such that $N_k(s) \cap N_l(p) \subseteq B$, where $(k, l) \in \{(14, 14), (12, 18)\}$.

The thinning process consists of two phases. Forward thinning reduces the input objects to 2–3 voxel thick surface patches, which are peeled further in the backward thinning phase. The unusual aspect in this process is that during backward thinning, object points are visited in descending order of their distance value. The sequential thinning is rather sensitive to the visiting order of border points. As a result, the final skeleton usually has some unwanted branches or surface segments. This situation is illustrated on both grids in Fig. 5.2, and examples for its occurrence can be found in Figs. 5.3 and 5.4 on the BCC and the FCC grids, respectively.

Note that the produced skeleton is fully reversible due to the fact that the extracted medial surface contains all the CMBs (as safe skeletal points).

In Algorithm 5.1, most steps (i.e., distance mapping, detection of CMBs, and forward thinning) have linear time complexity, but the backward thinning is slower. During the *for* loop in Line 10, all remaining object points are visited, but only a one-unit layer from each segment can be peeled due to the descending distance-based visiting order, since only simple points can be deletable. Hence, each object point will be visited at most as many times as the object's thickness until the algorithm terminates. That is why we showed that Strand's algorithm for both grids is nonlinear.

Algorithm 5.1: $Strand\_(k, \bar{k})$ $\big((k, \bar{k}) \in \{(14, 14), (12, 12)\}\big)$

Input: picture $(\mathcal{V}, k, \bar{k}, X) \in \{(\mathbb{B}, 14, 14, X), (\mathbb{F}, 12, 12, X)\}$
Output: picture $(\mathcal{V}, k, \bar{k}, Y) \in \{(\mathbb{B}, 14, 14, Y), (\mathbb{F}, 12, 12, Y)\}$
`// Distance mapping and identifying CMBs`
$DT \leftarrow$ `computeDT(`*X,* d_k`)`
$A \leftarrow$ `CMB(`*DT,* d_k`)`
$Y \leftarrow X$
`// Forward thinning`
for $i \leftarrow 1$ **to** `max(`*DT*`)` **do**
  foreach $p \in Y$ **do**
    if $DT(p) = i$, p is simple and $N_k(p) \cap A = \emptyset$ **then**
      $Y \leftarrow Y \setminus \{p\}$
`// Backward thinning`
repeat
  $changed\_any \leftarrow false$
  for $i \leftarrow$ `max(`*DT*`)` **downto** 1 **do**
    repeat
      $changed \leftarrow false$
      $D \leftarrow \{p \in Y \setminus A \mid DT(p) = i,$ and p is *simple*$\}$
      foreach $p \in D$ **do**
        if p is *simple* and not a *surface edge point* **then**
          $Y \leftarrow Y \setminus \{p\}$
          $D \leftarrow D \setminus \{p\}$
          $changed\_any \leftarrow true$
          $changed \leftarrow true$
    until $changed = false$
until $changed\_any = false$

Theorem 5.2.1. [32, 34] *The runtime complexity of Strand's algorithms (see Algorithm 5.1) is* $O(|\mathcal{V}|^{4/3})$.

5.2.2 Two modified versions on the BCC grid

To construct linear-time algorithms, we merge the thinning phases in Algorithm 5.1 and simplify the organization of the thinning iterations. Our deletion rule also preserves non-simple points and surface edge points. We introduce a new parameter N which gives an upper limit to the visiting number of all black points during the thinning phase (i.e. each iteration will be repeated at most N times). Note that setting N to ∞ would mean that the visiting number is unlimited. This parameter is applied in both of our improved algorithms. Our main goal is to reduce the sensitivity to the vis-

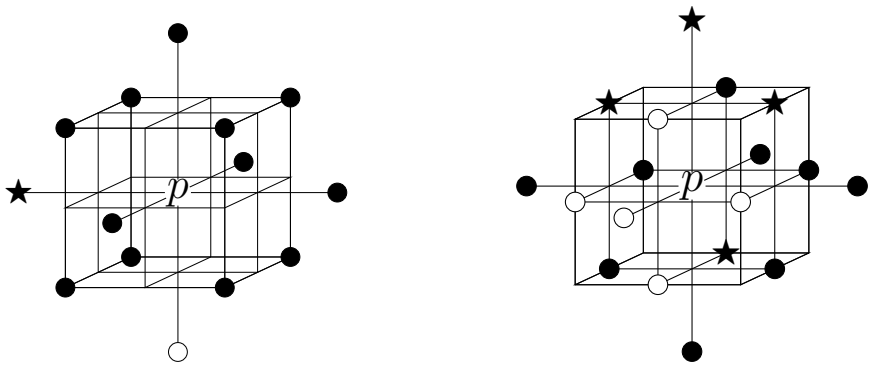
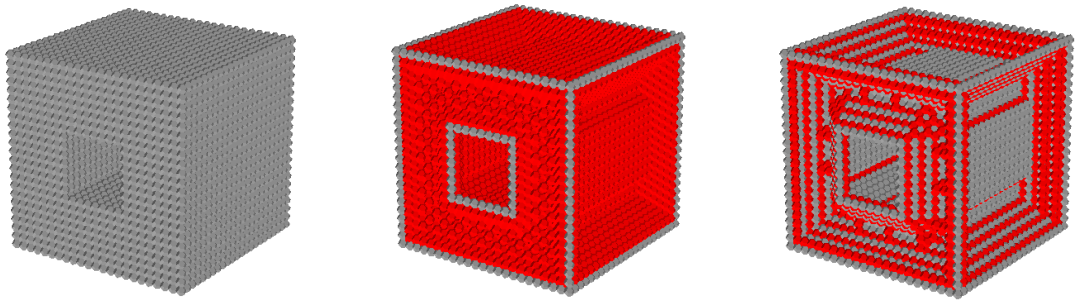


Figure 5.2: *Examples for an unfavorable visiting order during sequential thinning on the BCC (left) and the FCC grid (right). Border point p becomes non-simple after deletion of its neighbors marked "★".*

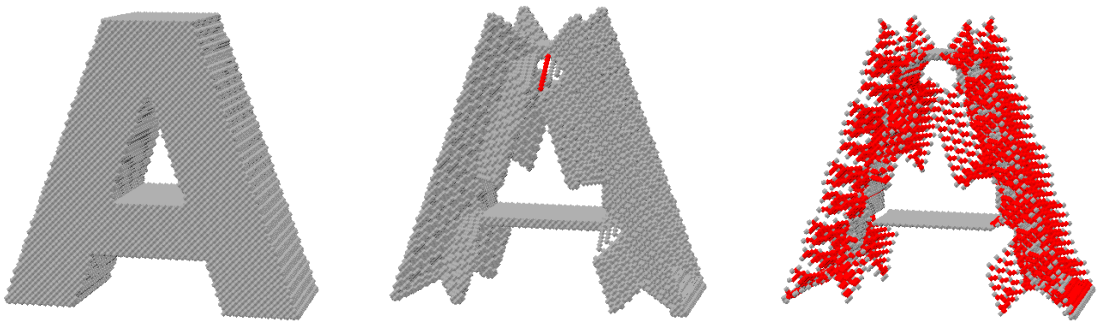


original object (20 280) *result of forward thinning* *final skeleton* (4 926)

Figure 5.3: *Result of Strand's algorithm on a holey cube sampled on the BCC grid. In the middle and right figure, anchor voxels are gray and further skeleton voxels are red. Numbers in parentheses show the number of black points.*

iting order of border points. The sequential thinning part consists of two substeps in our modified algorithms. First, we collect the deletable points from the actual picture into set D. Then we individually delete each point in this set if they are still simple when visited. In order to lessen the occurrence of false skeleton points, elements of D are visited in lexicographical order.

Our first improved variant, called $APT\_(14,14)\_N$, is an anchor-based thinning method that considers local maxima instead of CMBs to be anchor points because many CMBs are proved to be "false" skeleton points on weighted distance maps [13]. It is important to note that the input object is fully reconstructible only if d_{14} distance function is chosen. In this approach, we consider only strong border points to ensure that the strong and weak border points will be visited in different iterations. At the end of each iteration, all visited but non-deleted points are insterted in the set S of skeleton points (see Line 18 of Alg. 5.2). This operation guarantees that no border



original object (84 911) *result of forward thinning* *final skeleton* (6 847)

Figure 5.4: *Result of Strand's algorithm on an A-shaped object sampled on the FCC grid. In the middle figure, the longest branch is depicted in red, whose endpoint was a border point in the initial picture. In the right figure, CMBs are gray and further skeleton voxels are red. Numbers in parentheses show the number of black points.*

points will be examined during the further iterations which already have been visited.

Our second improved variant, called $DDT\_(14,14)\_N$, is a purely distance-driven thinning method that omits the detection of anchor points, i.e., set A is not used in this version. The border points are visited in ascending order of their distance value during the thinning phase.

We can state that the maximal number of times the same black point is visited in Algorithms 5.2 and 5.3 is equal to parameter N.

Theorem 5.2.2. [34] *The runtime complexity of Algorithms 5.2 and 5.3 is linear if* $N \in \mathbb{N}$.

The extracted surface skeletons of the holey cube by our improved algorithms are shown in Fig. 5.5. It is easy to see that these surface skeletons contain less insignificant skeleton points compared to the result in Fig. 5.3.

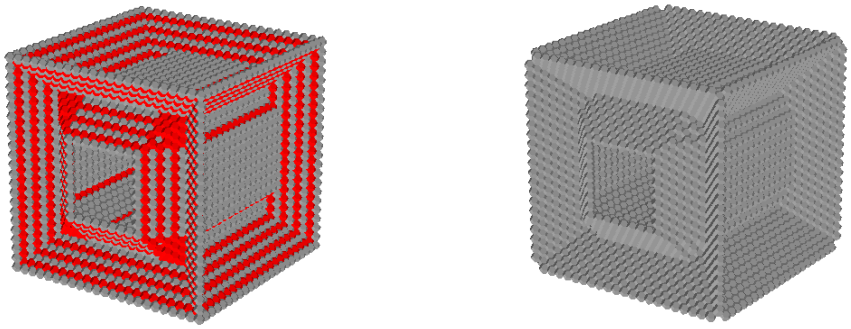
Our algorithms were tested on numerous objects of different shapes. Figures 5.6–5.9 show the extracted surface skeletons. To make the difference between the applied parameters more visible, some of the following figures consist of fused images, where red, green, and gray voxels belong to the first, second, and both skeletons extracted with the indicated parameters, respectively. In the case of Strand's method the gray and red points mean the anchor points and further skeletal points, respectively. In case of any $\langle a,b\rangle^{\mathbb{B}}$ weighted distances, local maxima are considered as anchor points instead of CMBs even for Strand's method. Numbers in parentheses show the number of object or skeleton points.

As we discussed in Subsect. 5.2.1, Strand's algorithm leaves many insignificant skeleton segments in case of d_{14} distance. It is easy to see that Strand's method remarkably overshrinks the input object, when weighted distance is used. Also note

Algorithm 5.2: Anchor-preserving thinning – $APT\_(14,14)\_N$

Input: picture $(\mathbb{B}, 14, 14, X)$, distance d, and visiting limit N
Output: picture $(\mathbb{B}, 14, 14, S)$

```
1  DT ← computeDT(X, d)
2  A ← LocalMaxima(DT, d)
3  S ← ∅
4  repeat
5      L ← {p ∈ X \ (S ∪ A) | p is a strong border point}
6      D ← {p ∈ L | p is simple for X and not a surface edge point}
7      t ← 0
8      repeat
9          t ← t + 1
10         changed ← false
11         foreach p ∈ D do
12             if p is simple for X then
13                 X ← X \ {p}
14                 L ← L \ {p}
15                 D ← D \ {p}
16                 changed ← true
17     until t = N or changed = false
18     S ← S ∪ L
19 until D = ∅
```



$APT\_(14,14)\_N$
(4 432)

$DDT\_(14,14)\_N$
(3 881)

Figure 5.5: *Result of the modified algorithms on a holey cube on d_{14} distance map with $N = 2$. In the left figure anchor voxels are gray and further skeleton voxels are red. Numbers in parentheses indicate the number of skeleton points.*

that the more accurate approximation of the Euclidean distance is used, the less anchor points are detected. In the case of $N = 1$, algorithm *DDT_(14,14)_N* leaves

Algorithm 5.3: Distance-ordered thinning – $DDT\_(14,14)\_N$

Input: picture $(\mathbb{F}, 12, 12, X)$, distance d, and visiting limit N
Output: picture $(\mathbb{F}, 12, 12, Y)$

```
1  DT ← computeDT(X, d)
2  Y ← X
3  for i ← 1 to max(DT) do
4      D ← {p ∈ Y | DT(p) = i, p is simple for Y and not a surface edge point}
5      t ← 0
6      repeat
7          t ← t + 1
8          changed ← false
9          foreach p ∈ D do
10             if p is simple for Y then
11                 Y ← Y \ {p}
12                 D ← D \ {p}
13                 changed ← true
14     until t = N or changed = false
```

one side of the letter A almost unchanged due to the unfavorable visiting order of border points. This phenomenon is successfully handled by setting N to 2. However, the modified versions may overshrink the object if N is too large, especially in d_8 or d_{14} distance maps (see Fig. 5.7). Hence, option $N = 2$ is usually sufficient. We can also observe that the medial surface is strongly jagged due to the nature of sequential thinning.

A similar phenomenon can be observed on the gear and the amphora. By setting N to 2, we get a much cleaner skeleton. Note that the number and distribution of skeletal points also depend on the thinning strategy.

5.2.3 Two modified versions on the FCC grid

With the same principle as in the BCC case (see Subsect. 5.2.2), we simplify Strand's algorithm for the FCC grid by merging its thinning phases. We can observe that, in the BCC case, visiting of each point exactly ones during thinning (i.e., setting N to 1) is insufficient. That is why, in the FCC case, we omit parameter N, and set the visiting upper limit to 2 for the precollected border points.

Let us call our anchor-based thinning method $APT\_(12,12)$ and our distance-driven algorithm $DDT\_(12,12)$. Only the distance-driven variant has a significant difference compared to the BCC case. If d_{18} distance is applied, collection of deletable points must be repeated twice, since some points with corresponding distance value may be interior points during the first examination. Note that the only medial sur-

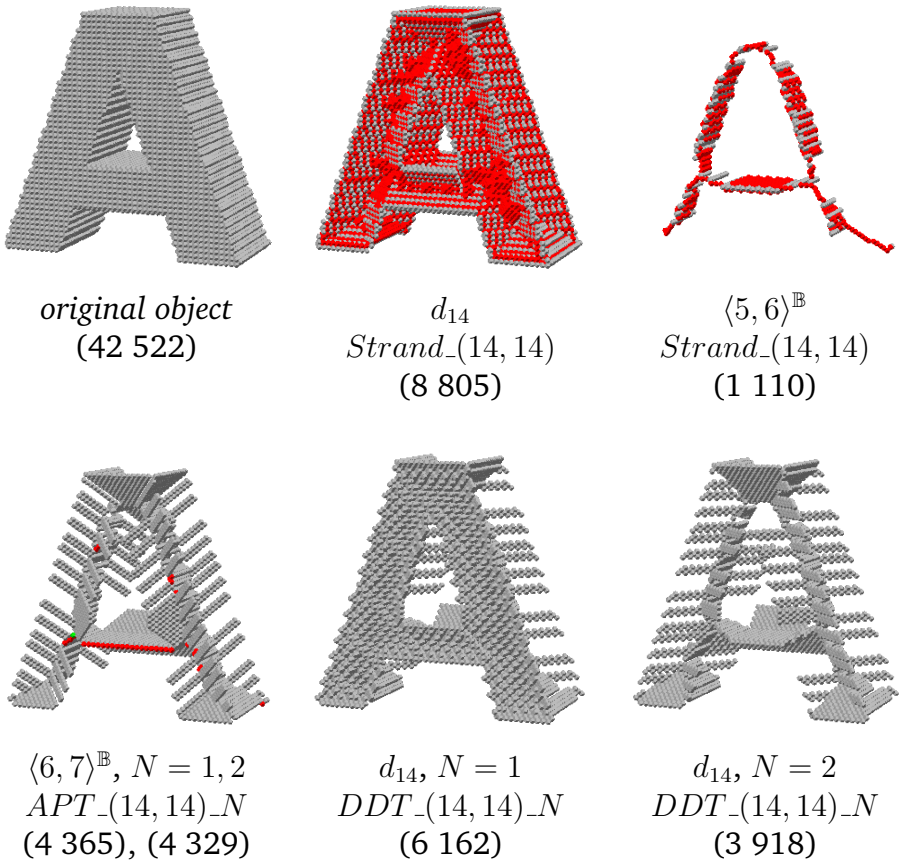


Figure 5.6: *Produced surface skeletons of the letter A with various parameters.*

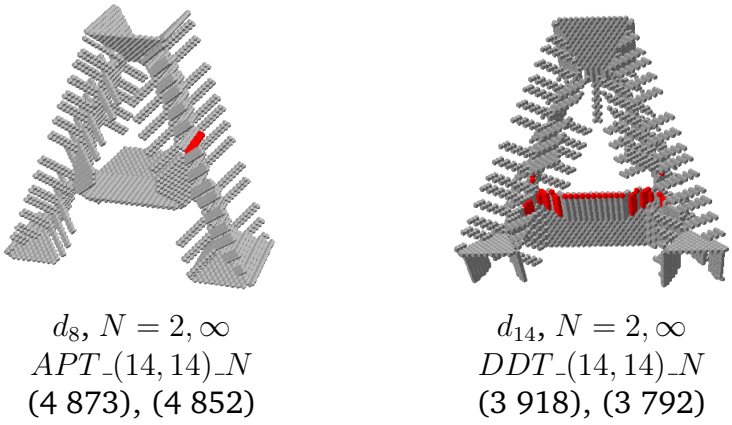


Figure 5.7: *Overshrinked surface skeletons of the letter A. For $N=\infty$, the highest repetition number, i.e. the highest value of t in Algorithm 5.2 and Algorithm 5.3 was 14 and 12, respectively.*

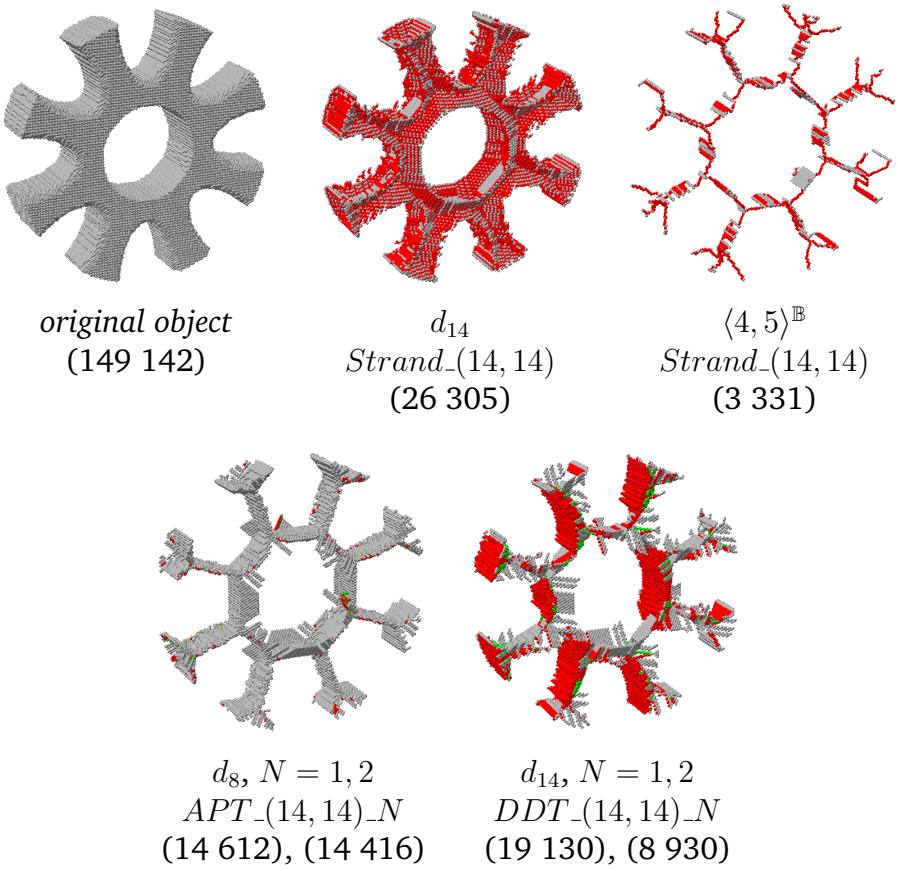


Figure 5.8: *Produced surface skeletons of a gear with various parameters.*

faces produced by algorithm $APT\_(12,12)$ are fully reversible if a neighbor-based distance function (i.e., d_{12} or d_{18}) is used.

It is easy to see that each border point is visited up to k times in each thinning iteration in Algs. 5.4 and 5.5, where $k = 4$ for Algorithm 5.5 with $d = d_{18}$, otherwise $k = 2$. As a consequence, all object points are visited maximum two or four times during the thinning phase. Hence, the computational cost is $O(k \cdot |B|)$, $k \in \{2, 4\}$.

Theorem 5.2.3. [32] *The runtime complexity of Algs 5.4 and 5.5 is linear.*

The reported algorithms were tested on numerous objects of different shapes. Figures 5.10–5.12 show the produced surface skeletons from different distance maps. In the case of methods $Strand\_(12,12)$ and $APT\_(12,12)$ the gray and red points indicate the anchor points and further skeletal points, respectively. Numbers in parentheses show the number of black points. In order to visualise the sensitivity of the sequential thinning to the visiting order of border points, we also extracted the surface skeleton of two rotated versions of the amphora-shaped object (see Fig. 5.12).

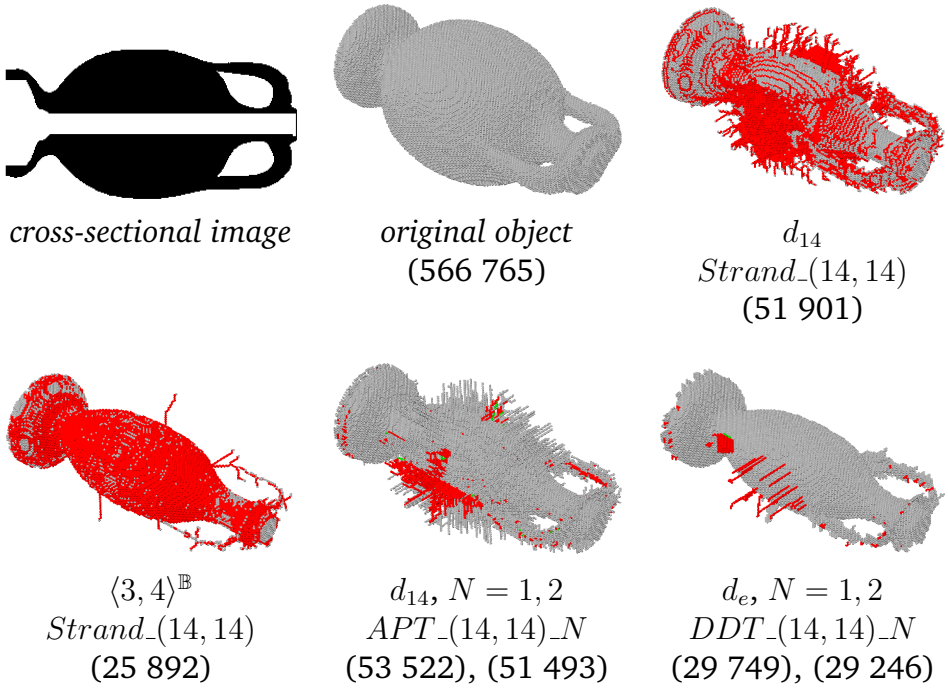


Figure 5.9: *Produced surface skeletons of an amphora with various parameters.*

We can observe that the anchor-preserving thinning method generates less insignificant surface patches than Strand's algorithm. Most of the unwanted branches are not generated by the purely distance-ordered algorithm. However, it may overshrink the object, especially for $d = d_{12}$. This phenomenon is successfully handled by choosing a weighted distance suggested in [25].

5.3 Curve-thinning

In this section, we present two distance-driven curve skeletonization methods with line-endpoint criteria. One of them works on $(14, 14)$ pictures of the BCC grid, and the other one considers $(18, 12)$ pictures of the FCC grid. Furthermore, for the $(18, 12)$ variant, we measure the required time of the thinning process, and also the reconstructibility of objects from their centerline.

5.3.1 Curve-thinning on the BCC grid

By modifying our distance-ordered variant, we can also extract the curve skeleton directly from the original object. For this purpose, instead of surface edge points we retain either of two types of curve endpoints. Point $p \in B$ is an *E1 endpoint* if

Algorithm 5.4: Anchor-preserving thinning – $APT\_(12, 12)$

Input: picture $(\mathbb{F}, 12, 12, X)$, and distance d
Output: picture $(\mathbb{F}, 12, 12, S)$

```
1  DT ← computeDT(X, d)
2  A ← LocalMaxima(DT, d)
3  S ← ∅
4  repeat
5      L ← {p ∈ X \ (S ∪ A) | p is a border point}
6      D ← {p ∈ L | p is simple for X and not a surface edge point}
7      t ← 0
8      repeat
9          t ← t + 1
10         changed ← false
11         foreach p ∈ D do
12             if p is simple for X then
13                 X ← X \ {p}
14                 L ← L \ {p}
15                 D ← D \ {p}
16                 changed ← true
17     until t = 2 or changed = false
18     S ← S ∪ L
19 until D = ∅
```

$|N_{14}(p) \cap B| = 1$, i.e., p has only one black neighbor in $N_{14}(p)$. Let u be this black neighbor. Point p is an *E2 endpoint* if p is an *E1* endpoint and u is a line point. Point u is a *line point* if there are exactly 2 black points in $N_{14}(u)$ and they are not 14-adjacent to each other. We consider a border point deletable if it is simple but not an endpoint. In the corresponding curve-thinning algorithm called $DDCT\_(14, 14)$, we reorganised the thinning iterations used in our surface-thinning methods: the non-deleted black points must be visited again during the further iterations in order to shrink the surface patches until line branches are only left. Parameter N is not used since the sufficient number of iterations depends on the structure of the objects in the input picture. This algorithm terminates only if there are no more deletable points. Therefore, it is impossible to set a constant upper limit k to the visiting number of object points. As a result, the extraction of the curve skeleton has nonlinear time complexity.

It is important to note that anchor point condition should not be used because the detected ridge points belong to the surface skeleton, so surface segments will probably be also generated.

To ensure distance-driven thinning, all distinct distance values are collected into

Algorithm 5.5: Distance-driven thinning – $DDT\_(12, 12)$

Input: picture $(\mathbb{F}, 12, 12, X)$, and distance d
Output: picture $(\mathbb{F}, 12, 12, Y)$

```
1  DT ← computeDT(X, d)
2  Y ← X
3  if d = d_18 then
4  |  it ← 2
5  else
6  |  it ← 1
7  for i ← 1 to max(DT) do
8  |  for l ← 1 to it do
9  |  |  D ← {p ∈ Y | DT(p) = i, p is simple for Y and not a surface edge point}
10 |  |  t ← 0
11 |  |  repeat
12 |  |  |  t ← t + 1
13 |  |  |  changed ← false
14 |  |  |  foreach p ∈ D do
15 |  |  |  |  if p is simple for Y then
16 |  |  |  |  |  Y ← Y \ {p}
17 |  |  |  |  |  D ← D \ {p}
18 |  |  |  |  |  changed ← true
19 |  |  until t = 2 or changed = false
```

a set right after the distance transform. The *for* loop (see lines 3–16) iterates through this set in ascending order. As a consequence, the total number of iterations is equal to the number of (positive) unique distance values, which depends on the shape of objects and the chosen distance function.

According to our experiments, visiting elements of D in lexicographical order has a relevant effect only on d_8 and d_{14} distance maps. The reason behind this phenomenon is the fact that on Chamfer or Euclidean distance maps the number of considered border points in a thinning iteration is less than in d_{14} or d_8 distance map, so there is a lower chance of unfavorable visiting order. Furthermore, there are much more configurations for surface edge points than for line-endpoints. Hence, the curve endpoint criteria are less sensitive to the visiting order, especially the $E2$ condition.

Results of three objects with different shapes are presented in Figs. 5.13–5.15. Following the same color marking applied in Subsect. 5.2.2, the red, green, and gray voxels belong to the first, second, and both skeletons extracted with the indicated parameters, respectively. Numbers in parentheses show the number of object or skeleton points. It can be well observed that significantly less false line segments were produced on the curve skeletons with condition $E2$ compared to $E1$.

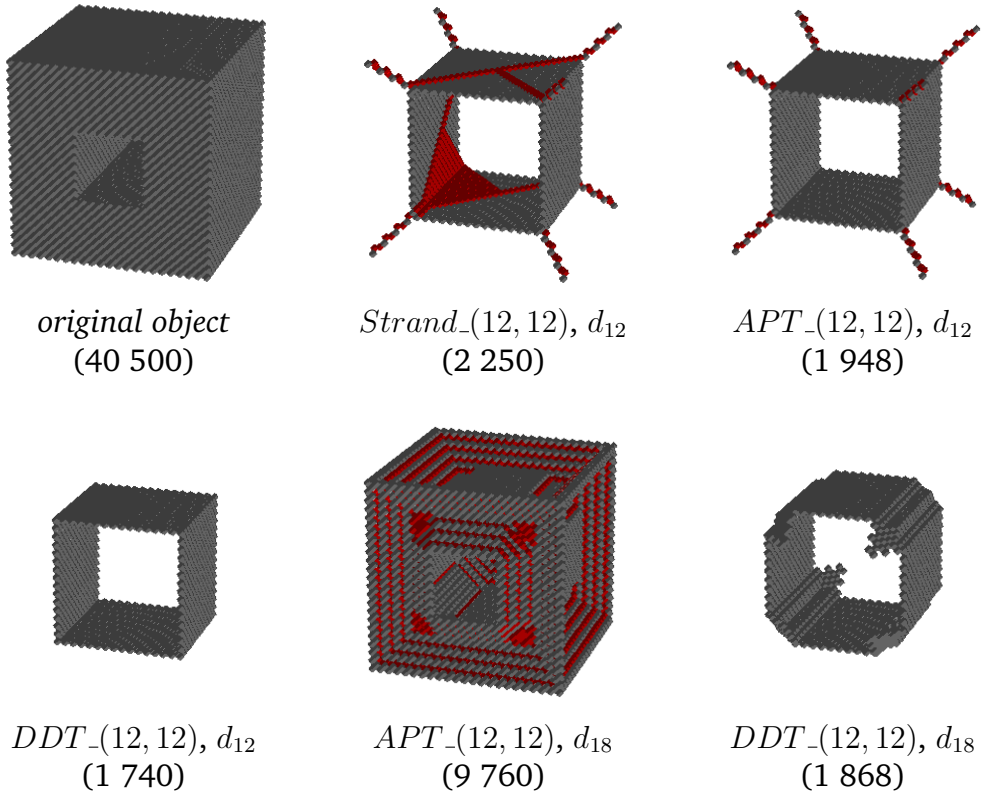


Figure 5.10: *Produced surface skeletons of a holey cube with various parameters.*

5.3.2 Curve-thinning on the FCC grid

Similarly to the BCC case (see Subsect. 5.3.1), we can produce the centerlines of objects on $(18, 12)$ pictures of the FCC grid. Here we rely on the $E1$ and $E2$ endpoint criteria as well. Point $p \in B$ is an *E1 endpoint* if p has only one black neighbor in $N_{18}(p)$. Let u denote this black neighbor in this case too. Point p is an *E2 endpoint* if p is an *E1* endpoint and u is a line point. Point u is a *line point* if there are exactly 2 black points in $N_{18}(u)$ and they are not 18-adjacent to each other. Since we cannot set a constant upper limit to the visiting number of object points, this thinning algorithm has nonlinear time complexity.

The reported algorithm was tested on several objects of different shapes, see Fig. 5.16. We examined the reconstructibility of objects from their centerline (see Figs. 5.17–5.20) by performing reverse distance transform [71].

The produced centerlines with different line endpoint criteria are fused (just like in the previous subsection), where red, green, and gray voxels belong to the $E1$, $E2$, and both produced centerlines with the indicated parameters, respectively. Furthermore, the transparent contour of the original object is displayed in order to verify the centeredness of the resulting centerlines. We can observe that preserving endpoints

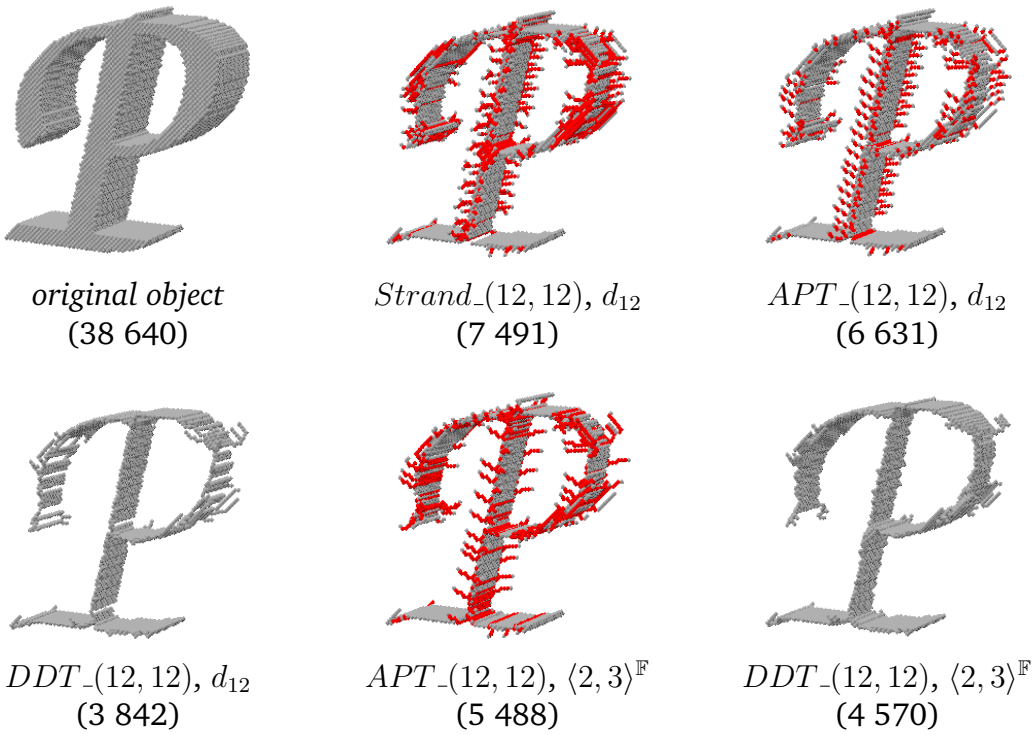
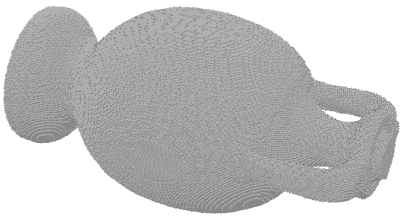


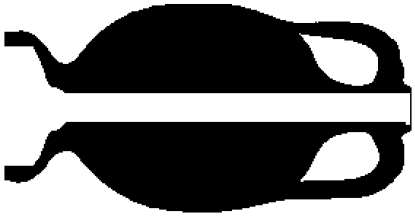
Figure 5.11: *Produced surface skeletons of a P-shaped object with various parameters.*

of type $E2$ leaves fewer branches compared to $E1$ on the FCC grid as well. Thus, the centerline with condition $E2$ gives a more concise representation of the object. However, the reconstructed object is usually less complete because of the loss of details of its shape. We got the worst quality of reconstruction in the case of the holey cube, because it is the least tubular object without any curved surface. Hence, centerline is not a suitable shape descriptor for it. Since sequential thinning algorithms are sensitive to the visiting order of border points, they may produce asymmetric skeletons, just like criterion $E1$ from the holey cube.

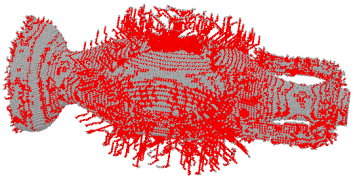
We also measured the required time for the curve-thinning method to process. For this purpose, the `std::chrono` library was applied in the implementation written in C++ on a usual desktop (HP ProDesk 400 G4; 3.20 GHz Intel Core i5-6500; Windows 10 x64). A detailed list can be found in Table 5.1. Note that just the distance transform and the iterative thinning process were considered here, file operations were not taken into account. Notice that the better the approximation to the Euclidean distance is, the more time the thinning takes. This happens due to the growing number of distinct values, which implies that the number of thinning iterations is increasing as well. Additionally, the Euclidean DT takes more raster scans than the Chamfer ones through the picture, which makes it computationally even more expensive. We can also observe that thinning with endpoint condition $E2$ takes



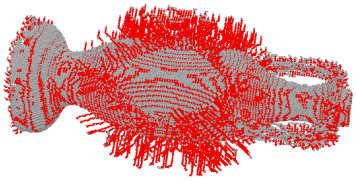
original object
(1 132 691)



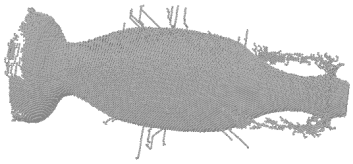
cross-sectional image



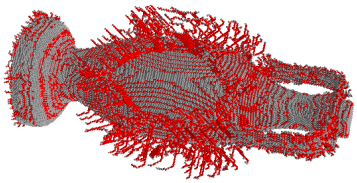
$Strand\_(12,12)$, d_{12}
(61 585)



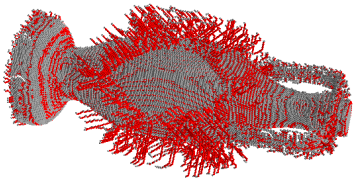
$APT\_(12,12)$, d_{12}
(58 512)



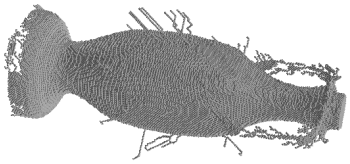
$DDT\_(12,12)$, d_{12}
(39 326)



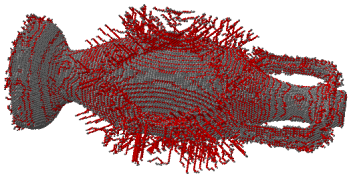
90°, $Strand\_(12,12)$, d_{12}
(60 657)



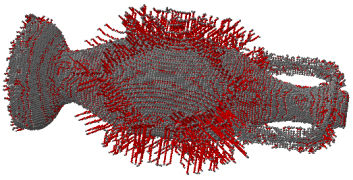
90°, $APT\_(12,12)$, d_{12}
(57 475)



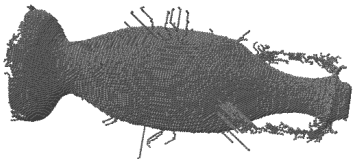
90°, $DDT\_(12,12)$, d_{12}
(39 208)



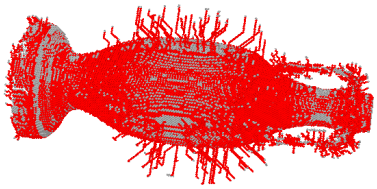
180°, $Strand\_(12,12)$, d_{12}
(61 732)



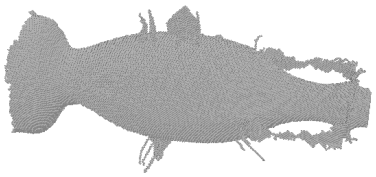
180°, $APT\_(12,12)$, d_{12}
(58 476)



180°, $DDT\_(12,12)$, d_{12}
(39 632)



$APT\_(12,12)$, $\langle 11,16,19\rangle^{\mathbb{F}}$
(47 287)



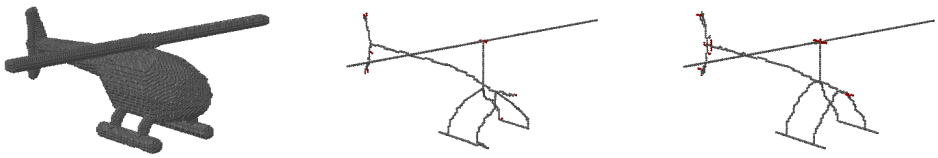
$DDT\_(12,12)$, $\langle 11,16,19\rangle^{\mathbb{F}}$
(46 439)

Figure 5.12: *Produced surface skeletons of an amphora with various parameters. Results from the rotated objects can be found in third and fourth rows. Note that all objects are depicted in the same orientation. The rotation was performed around the vertical axis.*

Algorithm 5.6: Distance-driven curve-thinning — $DDCT\_(14, 14)$

Input: picture $(\mathbb{B}, 14, 14, X)$, distance d, and endpoint of type $\varepsilon \in \{E1, E2\}$
Output: picture $(\mathbb{B}, 14, 14, Y)$

```
1  DT ← computeDT(X, d)
2  Y ← X
3  for k ← 1 to max(DT) do
4      repeat
5          D ← {p ∈ Y | DT(p) ≤ k, p is simple for Y and not an endpoint of
             type ε}
6          changed_any ← false
7          repeat
8              changed ← false
9              foreach p ∈ D do
10                 if p is simple for Y then
11                     Y ← Y \ {p}
12                     D ← D \ {p}
13                     changed ← true
14                     changed_any ← true
15         until changed = false
16     until changed_any = false
```



original object (94 031) | d_8 $E1$ & $E2$ (539) & (524) | $\langle 3, 4 \rangle^{\mathbb{B}}$ $E1$ & $E2$ (576) & (537)

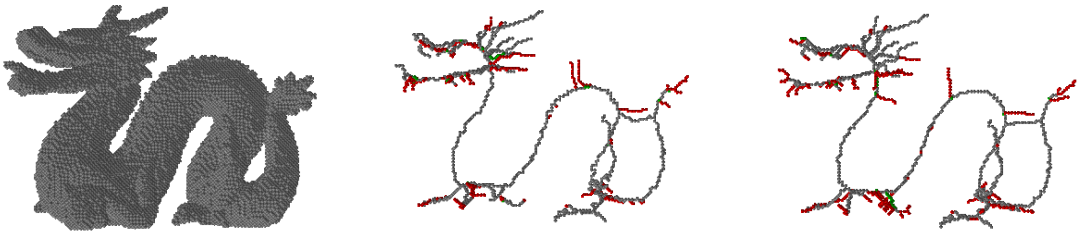
Figure 5.13: *Extracted curve skeletons of a helicopter with various parameters.*

less time because it leaves fewer border points to visit for possible deletion in the remaining iterations.

5.4 Concluding remarks

In this chapter, sequential thinning methods belonging to the *third thesis group* are reported.

We showed that the reference surface-thinning algorithms proposed by Strand [72]

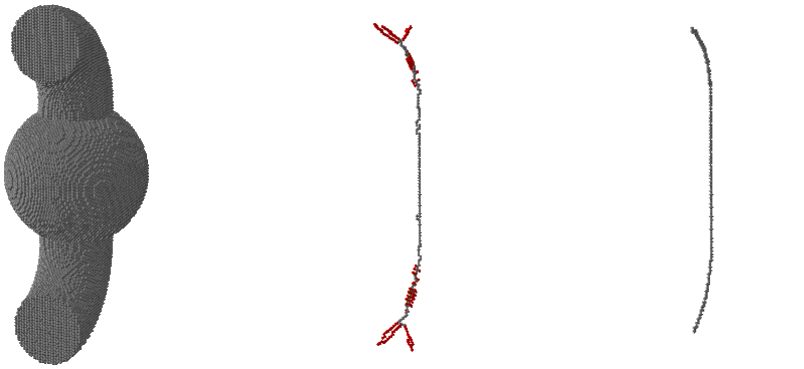


original object
(105 933)

$\langle 4,5\rangle^{\mathbb{B}}$
$E1$ & $E2$
(1 131) & (861)

d_e
$E1$ & $E2$
(1 153) & (818)

Figure 5.14: *Extracted curve skeletons of a dragon with various parameters.*



original object
(212 639)

d_{14}
$E1$ & $E2$
(283) & (145)

d_e
$E1$ & $E2$
(121) & (121)

Figure 5.15: *Extracted curve skeletons of a tube with various parameters. In the right figure, the $E1$- and $E2$-skeletons coincide with each other since no $E2$-endpoint was detected.*

are nonlinear on neither $(14,14)$, nor $(12,12)$ pictures. In contrast, all our constructed surface skeletonization methods have linear time complexity, and are less sensitive to the visiting order of border points. Furthermore, unlike Strand's method, our algorithms are not constrained to a given distance.

In addition, two novel algorithms capable of producing curve skeletons are also described. Two types of line-endpoints are introduced for $(14,14)$ and $(18,12)$ pictures, and we showed that the obtained centerlines allow recovery of a significant subset of the object, especially in case of tubular-like models. All examined algo-

Algorithm 5.7: Distance-driven curve-thinning — $DDCT\_(18, 12)$

Input: picture $(\mathbb{F}, 18, 12, X)$, distance d, and endpoint of type $\varepsilon \in \{E1, E2\}$
Output: picture $(\mathbb{F}, 18, 12, Y)$

```
1  DT ← computeDT(X, d)
2  Y ← X
3  for k ← 1 to max(DT) do
4      repeat
5          D ← {p ∈ Y | DT(p) ≤ k, p is simple for Y and not an endpoint of
             type ε}
6          changed_any ← false
7          repeat
8              changed ← false
9              foreach p ∈ D do
10                 if p is simple for Y and not an endpoint of type ε then
11                     Y ← Y \ {p}
12                     D ← D \ {p}
13                     changed ← true
14                     changed_any ← true
15         until changed = false
16     until changed_any = false
```

holey cube $93 \times 93 \times 93$ | *dragon* $135 \times 86 \times 191$ | *fertility* $138 \times 70 \times 189$ | *gear* $45 \times 191 \times 191$

Figure 5.16: *The selected four test objects.*

rithms preserve the topology due to the fact that only a single simple point is deleted at a time.

Future research should be devoted to adapting the presented curve-thinning methods to the $(12, 12)$ and $(12, 18)$ pictures of the FCC grid, and constructing similar distance-based algorithms combined with parallel thinning strategies.

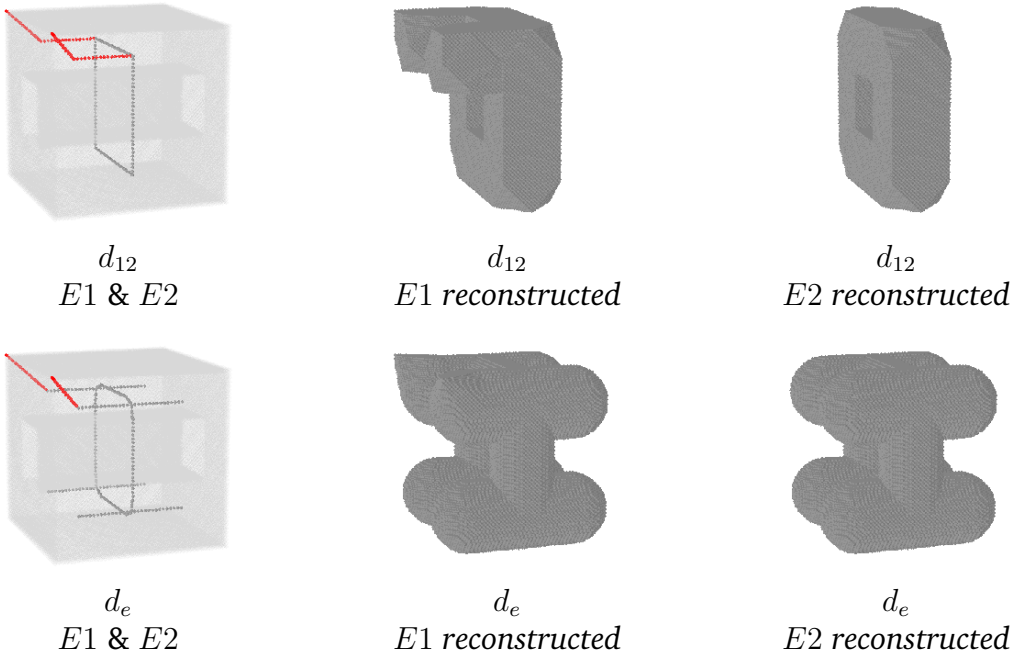


Figure 5.17: *Produced centerlines of the holey cube by using d_{12} and Euclidean distance (left) and the reconstructed objects (middle and right).*

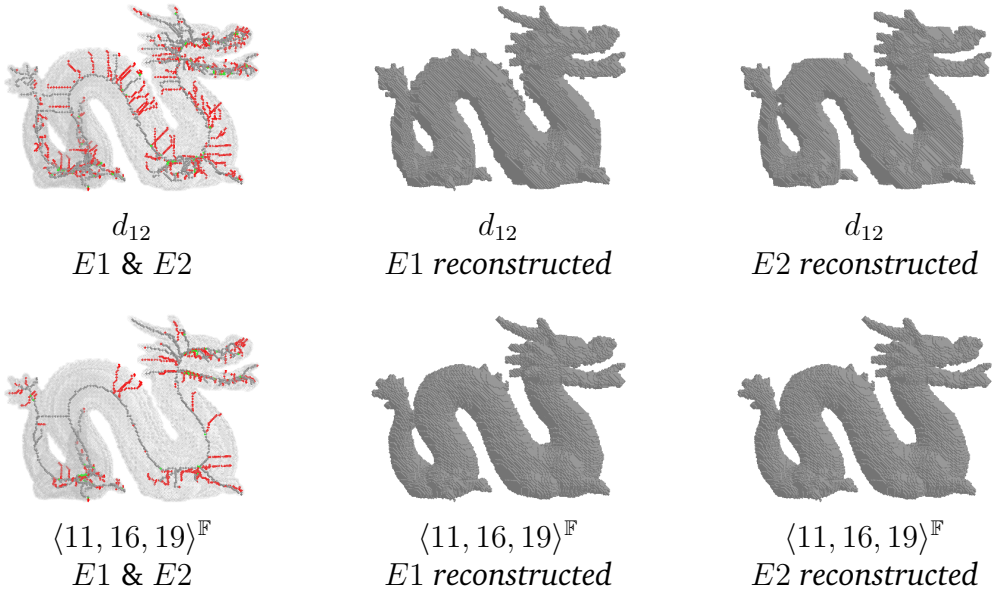
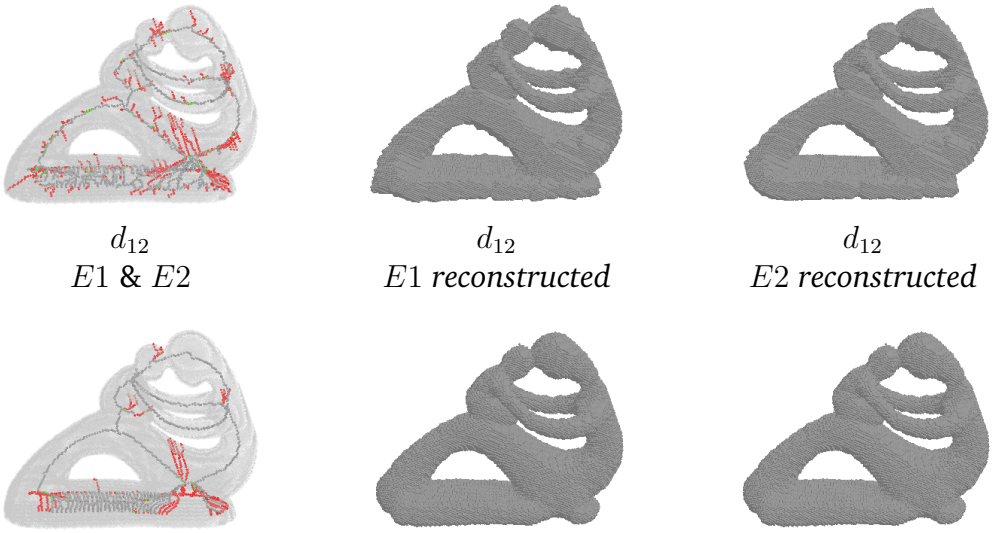


Figure 5.18: *Produced centerlines of the dragon by using d_{12} and a Chamfer distance (left) and the reconstructed objects (middle and right).*



d_{12}
$E1$ & $E2$

d_{12}
$E1$ *reconstructed*

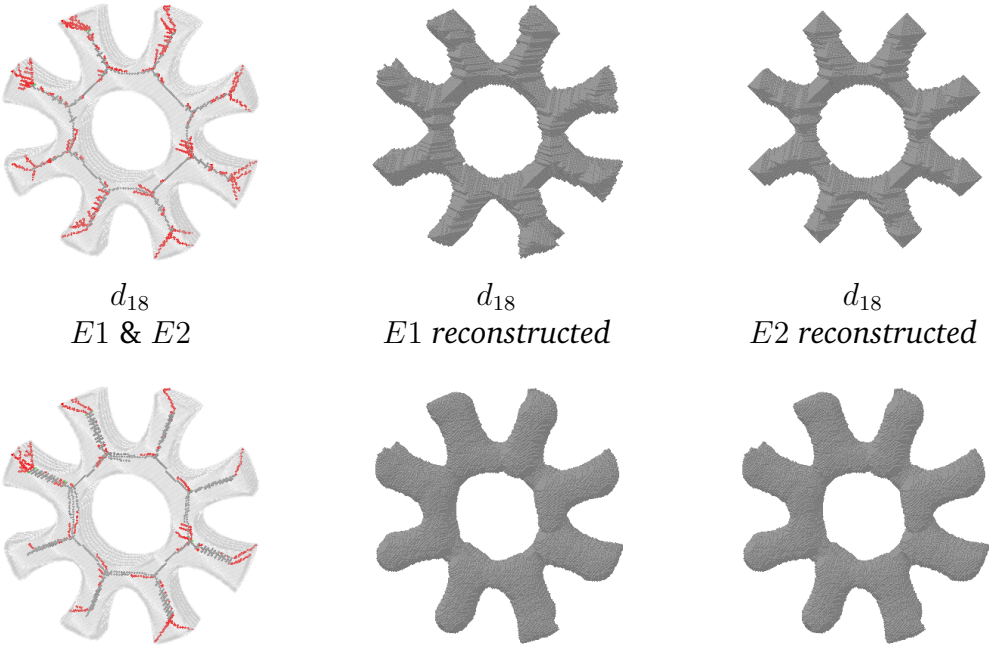
d_{12}
$E2$ *reconstructed*

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E1$ & $E2$

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E1$ *reconstructed*

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E2$ *reconstructed*

Figure 5.19: *Produced centerlines of the 'fertility' by using d_{12} and a Chamfer distance (left) and the reconstructed objects (middle and right).*



d_{18}
$E1$ & $E2$

d_{18}
$E1$ *reconstructed*

d_{18}
$E2$ *reconstructed*

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E1$ & $E2$

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E1$ *reconstructed*

$\langle 11, 16, 19\rangle^{\mathbb{F}}$
$E2$ *reconstructed*

Figure 5.20: *Produced centerlines of the gear by using d_{18} and a Chamfer distance (left) and the reconstructed objects (middle and right).*

Table 5.1: *Evaluation of Algorithm 5.7 on objects showed in Fig. 5.16. The investigated distance functions are listed in ascending order of their geometric accuracy.*

Test object	Distance function	Endpoint type	Running time (millisec)	Object reconstruction (%)
holey cube	d_{18}	$E1$	135	48.5
		$E2$	130	47.5
	d_{12}	$E1$	139	40.7
		$E2$	139	32.6
	$\langle 11, 16, 19\rangle^{\mathbb{F}}$	$E1$	177	57.9
		$E2$	176	51.8
	d_e	$E1$	299	58.8
		$E2$	289	58.0
dragon	d_{18}	$E1$	122	73.3
		$E2$	120	65.7
	d_{12}	$E1$	139	82.6
		$E2$	137	77.1
	$\langle 11, 16, 19\rangle^{\mathbb{F}}$	$E1$	177	86.4
		$E2$	170	80.8
	d_e	$E1$	265	88.0
		$E2$	257	82.6
fertility	d_{18}	$E1$	111	70.1
		$E2$	110	64.4
	d_{12}	$E1$	123	83.5
		$E2$	122	78.4
	$\langle 11, 16, 19\rangle^{\mathbb{F}}$	$E1$	163	87.3
		$E2$	159	82.3
	d_e	$E1$	238	88.5
		$E2$	232	81.1
gear	d_{18}	$E1$	130	53.4
		$E2$	127	38.6
	d_{12}	$E1$	142	81.3
		$E2$	141	75.0
	$\langle 11, 16, 19\rangle^{\mathbb{F}}$	$E1$	164	82.1
		$E2$	159	76.2
	d_e	$E1$	310	80.3
		$E2$	294	73.0

Bibliography

[1] Amenta, N., Choi, S.: Voronoi methods for 3D medial axis. In *Siddiqi, K., Pizer, S.M. (eds) Medial Representations – Mathematics, algorithms and applications. Comp. Im. and Vision*, vol. 37, Springer, 2008

[2] Arcelli, C., Sanniti di Baja, G.: Finding local maxima in a pseudo-Euclidean distance transform. *Comp. Vision, Graphics, and Im. Proc.* vol. 43(3), 361–367, 1988

[3] Arcelli, C., Sanniti di Baja, G.: Ridge points in Euclidean distance maps. *Patt. Rec. Letters* vol. 13(4), 237–243, 1992

[4] Arcelli, C., Sanniti di Baja, G., Serino, L.: Distance-Driven Skeletonization in Voxel Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33(4), 709–720, 2011

[5] Attali, D., Montanvert, A.: Computing and simplifying 2D and 3D continuous skeletons. *Comp. Vision and Im. Underst.*, vol. 67, 261–273, 1997

[6] Bertrand, G.: A parallel thinning algorithm for medial surfaces. *Patt. Rec. Letters* vol. 16, 979–986, 1995

[7] Bertrand, G.: On P-simple points. *Comptes Rendus De L Academie Des Sciences Serie I-mathematique* vol. 321, 1077–1084, 1995

[8] Bertrand, G., Aktouf, Z.: A 3D thinning algorithm using subfields. In *SPIE Proc. of Conf. on Vision Geometry*, 113–124, 1994

[9] Biswas, R., Largeteau-Skapin, G., Zrour, R., Andres, E.: Rhombic Dodecahedron Grid – Coordinate System and 3D Digital Object Definitions. In *Proc. 21st Int. Conf. on Discrete Geometry for Computer Imagery - DGCI*, 27–37, 2019

[10] Biswas, R., Largeteau-Skapin, G., Zrour, R., Andres, E.: Digital Objects in Rhombic Dodecahedron Grid. *Math. Morphol. Theory Appl.*, vol. 4(1), 143–158, 2020

[11] Blum, H.: A transformation for extracting new descriptors of shape. In *Wathen-Dunn, W. (ed.) Models for the Perception of Speech and Visual Form*, 362–380, MIT Press, 1967

[12] Borgefors, G.: Distance transformations in arbitrary dimensions. *Comp. Vision, Graphics, and Im. Proc.* vol. 27(3), 321–345, 1984

[13] Borgefors, G., Nyström, I., Sanniti di Baja, G.: Discrete Skeletons from Distance Transforms in 2D and 3D. In *Siddiqi, K., Pizer, S.M. (eds) Medial Representations – Mathematics, algorithms and applications. Comp. Im. and Vision*, vol. 37, Springer, 2008

[14] Brandt, J.W.: Describing a solid with the three-dimensional skeleton. In *Proc. SPIE Conf. Curves and Surfaces in Computer Vision and Graphics III*, Vol. 1830, 258–269, 1992

[15] Brandt, J.W., Algazi, V.R.: Continuous skeleton computation by Voronoi diagram. *CVGIP: Image Understanding*, vol. 55, 329–337, 1992

[16] Csébfalvi, B.: Cosine-Weighted B-Spline Interpolation: A Fast and High-Quality Reconstruction Scheme for the Body-Centered Cubic Lattice. *IEEE Transactions on Visualization and Computer Graphics* vol. 19(9), 1455–1466, 2013

[17] Čomić, L., Magillo, P.: Repairing 3D binary images using the BCC grid with a 4-valued combinatorial coordinate system. *Information Sciences* vol. 499, 47–61, 2019

[18] Čomić, L., Magillo, P.: Repairing 3D Binary Images Using the FCC Grid. *J. Math. Imaging Vis.* vol. 61, 1301–1321, 2019

[19] Čomić, L., Magillo, P.: On Hamiltonian cycles in the FCC grid. *Computers & Graphics* vol. 89, 88–93, 2020

[20] Čomić, L., Nagy, B.: A combinatorial coordinate system for the body-centered cubic grid. *Graphical Models* vol. 87, 11–22, 2016

[21] Čomić, L., Nagy, B.: A combinatorial 3-coordinate system for the face centered cubic grid. In *Proc. 9th Int. Symposium on Image and Signal Processing and Analysis – ISPA 2015*, 298–303, 2015

[22] Čomić, L., Nagy, B.: A topological 4-coordinate system for the face centered cubic grid. *Patt. Rec. Letters* vol. 83(1), 67–74, 2016

[23] Damon, J.: Geometry and medial structure. In *Siddiqi, K., Pizer, S.M. (eds) Medial Representations – Mathematics, algorithms and applications. Comp. Im. and Vision*, vol. 37, Springer, 2008

[24] Edelsbrunner, H., Iglesias-Ham, M., Kurlin, V.: Relaxed disk packing. In *Proc. 27th Canadian Conf. on Computational Geometry - CCCG 2015*, 128–135, Queen's University, Ontario, Canada, 2015

[25] Fouard, C., Strand, R., Borgefors, G.: Weighted distance transforms generalized to modules and their computation on point lattices. *Patt. Rec.* vol. 40(9), 2453–2474, 2007

[26] Gastineau, N., Togni, O.: Coloring of the d^{th} power of the face-centered cubic grid. *Discussiones Mathematicae - Graph Theory* vol. 41, 1001–1020, 2021

[27] Gau, C.J., Kong, T.Y.: Minimal nonsimple sets of voxels in binary pictures on a face-centered cubic grid. *Int. J. Patt. Rec. Artif. Intell.* vol. 13(4), 485–502, 1999

[28] Giblin, P.J., Kimia, B.B.: Local forms and transitions of the medial axis. In *Siddiqi, K., Pizer, S.M. (eds) Medial Representations – Mathematics, algorithms and applications. Comp. Im. and Vision*, vol. 37, Springer, 2008

[29] Hall, R.W.: Parallel Connectivity-Preserving Thinning Algorithms. *Machine Intell. and Patt. Rec.* vol. 19, 145–179, 1996

[30] Hall, R.W., Kong, T.Y., Rosenfeld, A.: Shrinking Binary Images. *Machine Intell. and Patt. Rec.* vol. 19, 31–98, 1996

[31] Herman, G.T.: Geometry of digital spaces. Birkhäuser, 1998

[32] Karai, G.: Distance-Oriented Surface Skeletonization on the Face-Centered Cubic Grid. In *Proc. 1st Int. Joint Conf. on Discrete Geometry and Mathematical Morphology - DGMM 2021*, LNCS vol. 12708, 177–188, Springer, 2021

[33] Karai, G.: Distance-Driven Curve-Thinning on the Face-Centered Cubic Grid. In *Proc. 2nd Int. Joint Conf. on Discrete Geometry and Mathematical Morphology - DGMM 2022*, LNCS vol. 13493, 354–365, Springer, 2022

[34] Karai, G., Kardos, P.: Distance-Based Skeletonization on the BCC Grid. *Acta Cybernetica* vol. 25(2), 351–367, 2021

[35] Karai, G., Kardos, P., Palágyi, K.: Subfield-based Parallel Kernel-thinning Algorithms on the BCC Grid. In *Proc. 11th Int. Conf. on Patt. Rec. Applications and Methods - ICPRAM 2022*, 288–295, SciTePress, 2022

[36] Karai, G., Kardos, P., Palágyi, K.: Fully Parallel Kernel-Thinning on the Face-Centered Cubic Grid. In *Proc. 11th Hungarian Conference on Computer Graphics and Geometry – GRAFGEO 2024*, ISBN: 978-963-421-948-4, 32–37, 2024

[37] Karai, G., Kardos, P., Palágyi, K.: Sufficient Conditions for Topology-Preserving Parallel Reductions on the Face-Centered Cubic Grid. *J. Math. Imaging Vis.*, 66, 271–292, 2024

[38] Karai, G., Kardos, P., Palágyi, K.: Topology-Preserving Reductions on (18,12) Pictures of the Face-Centered Cubic Grid. In *Proc. 12th Int. Conf. on Patt. Rec. Applications and Methods - ICPRAM 2022*, SciTePress, 254–261, 2023

[39] Kardos, P.: Topology preservation on the BCC grid. *J. Comb. Optim.* vol. 44, 2981–2995, 2022

[40] Kardos, P., Palágyi, K.: Topology preservation on the triangular grid. *Annals of Math. Artif. Intell.* vol. 75, 53–68, 2015

[41] Kittel, C.: Crystal Structures. In *Introduction to Solid State Physics, 8th Edition*. Wiley, New York, 2004

[42] Kong, T.Y.: On topology preservation in 2-D and 3-D thinning. *Int. J. Patt. Rec. Artif. Intell.* vol. 9(5), 813–844, 1995

[43] Kong, T.Y., Rosenfeld, A.: Digital topology: Introduction and survey. *Comp. Vision, Graphics, and Im. Proc.* vol. 48(3), 357–393, 1989

[44] Lam, L., Lee, S.-W., Suen, C.Y.: Thinning methodologies - A comprehensive survey. *IEEE Trans. on Pattern Analysis and Machine Intell.* vol. 14(9), 869–885, 1992

[45] Lohou, C., Dehos, J.: An Automatic Correction of Ma's Thinning Algorithm Based on P-simple Points. *J. Math. Imaging Vis.* vol. 36, 54–62, 2010

[46] Lohou, C., Dehos, J.: Automatic correction of Ma and Sonka's thinning algorithm using P-simple points. *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 32, 1148–1152, 2010

[47] Ma, C.M.: A 3D fully parallel thinning algorithm for generating medial faces. *Patt. Rec. Letters* vol. 16, 83–87, 1995

[48] Ma, C.M., Wan, S.-Y.: Parallel thinning algorithms on 3D $(18, 6)$ binary images. *Comp. Vision and Im. Underst.* vol. 80, 364–378, 2000

[49] Ma, C.M., Wan, S.Y.: A medial-surface oriented 3-d two-subfield thinning algorithm. *Patt. Rec. Letters* vol. 22, 1439–1446, 2001

[50] Ma, C.M., Wan, S.Y., Chang, H.K.: Extracting medial curves on 3D images. *Patt. Rec. Letters* vol. 23, 895–904, 2002

[51] Ma, C.M., Wan, S.Y., Lee, J.D.: Three-dimensional topology preserving reduction on the 4-subfields. *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 24, 1594–1605, 2002

[52] Marchand-Maillet, S., Sharaiha, Y.M.: Binary Digital picture Processing: A Discrete Approach. Academic Press, 2000

[53] Matej, S., Lewitt, R.M.: Efficient 3D grids for image reconstruction using spherically-symmetric volume elements. *IEEE Transactions on Nuclear Science* vol. 42(4), 1361–1370, 1995

[54] Matheron, G.: Examples of topological properties of skeletons. In *J. Serra (Ed.) Image analysis and mathematical morphology* vol. 2: Theoretical Advances, 217–238, Academic Press, 1988

[55] McAndrew, A., Osborne, C.: The Euler characteristic on the face-centred cubic lattice. *Patt. Rec. Letters* vol. 18(3), 229–237, 1997

[56] Nackman, L.R., Pizer S.M.: Three-dimensional shape description using the symmetric axis transform I: Theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, 187–202, 1985

[57] Németh, G., Kardos, P., Palágyi, K.: Topology preserving 2-subfield 3D thinning algorithms. In *Proc. 7th Int. Conf. Signal Proc., Patt. Rec. and App. – SPPRA 2010*, 310–316, 2010

[58] Németh, G., Kardos, P., Palágyi, K.: Topology preserving 3D thinning algorithms using four and eight subfields. In *Proc. Int. Conf. Im. Analysis and Rec. - ICIAR 2010* LNCS vol. 6111, 316–325, Springer, 2010

[59] Ogniewicz, R.L., Ilg, M.: Voronoi skeletons: Theory and applications. In *Proc. IEEE Comp. Soc. Conf. Comp. Vis. Patt. Rec. – CVPR 1992*, 63–69, 1992

[60] Ogniewicz, R.L., Kübler, O.: Hierarchic Voronoi skeletons. *Patt. Rec.*, vol. 28, 343–359, 1995

[61] Palágyi, K.: A 3-subiteration surface-thinning algorithm. In *Proc. 12th Int. Conf. Comp. Analysis of Im. Patt. – CAIP 2007*, LNCS vol. 4673, 628–635, Springer, 2007

[62] Palágyi, K.: A 3D fully parallel surface-thinning algorithm. *Theor. Comput. Sci.* vol. 406(1–2), 119–135, 2008

[63] Palágyi, K.: Equivalent sequential and parallel reductions in arbitrary binary pictures. *Int. J. Patt. Rec. Artif. Intell.* vol. 28, 1460009, 2014

[64] Palágyi, K., Karai, G., Kardos, P.: Sufficient Conditions for Topology-Preserving Parallel Reductions on the BCC Grid. In *Proc. 21st Int. Workshop on Combinatorial Image Analysis – IWCIA 2022*, LNCS vol. 13348, 71–83, Springer, 2023

[65] Preparata, F.P., Shamos, M.I.: Computational geometry. Springer-Verlag, 1985

[66] Ronse, C.: Minimal test patterns for connectivity preservation in parallel thinning algorithms for binary digital images. *Discrete Applied Mathematics* vol. 21(1), 67–79, 1988

[67] Saha, P.K., Borgefors, G., Sanniti di Baja, G.: A survey on skeletonization algorithms and their applications. *Patt. Rec. Letters*, vol. 76(1), 3–12, 2016

[68] Saha, P.K., Borgefors, G., Sanniti di Baja, G.: Skeletonization: Theory, Methods and Applications 1st edition, Academic Press, 2017

[69] Saito, T., Toriwaki, J.: A sequential thinning algorithm for three dimensional digital pictures using the Euclidean distance transformation. In *Proc. 9th Scandinavian Conf. on Im. Analysis – SCIA 1995*, 507–516, Uppsala, Sweden, 1995

[70] Schmitt, M.: Some examples of algorithms analysis in computational geometry by means of mathematical morphological techniques. In *Boissonnat, J.-D., Laumond J.-P. (eds) Proc. French Workshop on Geometry and Robotics – GeoRob 1988*, LNCS vol. 391, 225–246, Springer, 1988

[71] Sobiecki, A., Jalba, A., Telea, A.: Comparison of curve and surface skeletonization methods for voxel shapes. *Patt. Rec. Letters* vol. 47, 147–156, 2014

[72] Strand, R.: Surface skeletons in grids with non-cubic voxels. In *Proc. 17th Int. Conf. Patt. Rec. – ICPR 2004*, vol. 1, 548–551, 2004

[73] Strand, R.: The Euclidean Distance Transform Applied to the FCC and BCC Grids. In *Proc. Patt. Rec. Im. Analysis – IbPRIA 2005*, LNCS vol. 3522, 243–250, Springer, 2005

[74] Strand, R.: The face-centered cubic grid and the body-centered cubic grid: a literature survey. Technical Report 35, Centre for Image Analysis, Uppsala University, Uppsala, Sweden, 2005

[75] Strand, R., Borgefors, G.: Distance transforms for three-dimensional grids with non-cubic voxels. *Comp. Vision and Im. Underst.* vol. 100(3), 294–311, 2005

[76] Strand, R., Brunner, D.: Simple Points on the Body-Centered Cubic Grid. Technical Report 42, Centre for Image Analysis, Uppsala University, Uppsala, Sweden, 2006

[77] Strand, R., Nagy, B., Borgefors, G.: Digital distance functions on three-dimensional grids. *Theor. Comp. Sci.* vol. 412(15), 1350–1363, 2011

[78] Svensson, S.: Reversible Surface Skeletons of 3D Objects by Iterative Thinning of Distance Transforms. In *Bertrand G., Imiya A. (eds) Digital and picture Geometry*, LNCS vol. 2243, 400–411, Springer, 2001

[79] Székely, G.: Voronoi skeletons. In *Siddiqi, K., Pizer, S.M. (eds) Medial Representations – Mathematics, algorithms and applications. Comp. Im. and Vision*, vol. 37, 191–221, Springer, 2008

[80] Tagliasacchi, A., Delame, T., Spagnuolo, M., Amenta, N., Telea, A.: 3D Skeletons: A State-of-the-Art Report. *Computer Graphics Forum* vol. 35(2), 573–597, 2016

[81] Theussl, T., Möller, T., Gröller, M.E.: Optimal regular volume sampling. In *Proc. Visualization, VIS'01*, 91–98, 2001

Summary

This dissertation presents my results in the field of topology-preserving reductions and skeletonization (i.e., the extraction of skeleton-like shape features in digital binary pictures) on the body-centered cubic (BCC) grid and the face-centered cubic (FCC) grid. The topological kernels and the centerlines can be extracted from 2D objects, and in addition, 3D objects can be represented by their medial surfaces as well. The resulting 'skeleton' must be topologically equivalent to the original object, which concludes they must be produced by topology-preserving reductions. Thinning is an iterative object reduction, which is a suitable approach for extracting all skeleton-like shape features.

Chapter 2 describes the basic definitions and existing results of digital topology, topology-preserving reductions, and skeletonization. In the subsequent three chapters, the three thesis groups are discussed.

Thesis Group I.

Sequential reductions may delete only one black point at a time. Hence, topology preservation is guaranteed if only a simple point is deleted. In contrast, parallel reductions can delete multiple points simultaneously. To make sure that a parallel reduction preserves the topology, we relied on Ronse's theorem about minimal non-simple (MNS) sets: if a parallel reduction never deletes MNS sets, it is topology-preserving. In their seminal work, Gau and Kong discovered all the possible MNS sets on the FCC grid. Furthermore, Kardos gave characterization of MNS sets on the BCC grid.

In Chapter 3, we gave eight configuration-based sufficient conditions for topology-preserving parallel reductions on the considered grids. They are only capable of verifying topology preservation of previously designed parallel reductions, but not suitable to construct new deletion rules for reduction operations. That is why we also proposed, in total, ten point-based sufficient conditions that directly provide deletion rules of topology-preserving parallel reductions on these non-conventional grids. The proofs for their topology-preserving property can be found in my corresponding publications. Among the point-based rules, using the lexicographical order,

we distinguished the asymmetric conditions from the symmetric ones.

In addition, we generated topology-preserving reductions from our proposed point-based conditions. The asymmetric reductions are simpler than the symmetric ones, since their supports contain fewer points to be taken into account. Consequently, we can establish symmetric and asymmetric families of thinning algorithms.

Thesis Group II.

In Chapter 4, we derived topology-preserving fully parallel thinning algorithms from our constructed asymmetric reductions discussed in Chapter 3. Since we did not apply any geometrical constraint, these algorithms can be classified as kernel-thinning methods. We showed that they do not leave any simple point in the resulting object. Thus, they all extract the topological kernels of objects sampled on the BCC and FCC grids.

Furthermore, we partitioned the BCC grid into four and eight subfields such that each point belongs to a different subfield than any of its 14-adjacent one. Due to this beneficial property, we could simplify the configuration-based condition for topology-preserving parallel reductions on the BCC grid: the simultaneous deletion of all simple points in the corresponding subfield is a topology-preserving reduction. We established two subfield-based parallel kernel-thinning algorithms on the BCC grid by applying these reductions. We showed that these two methods have a similar computational cost due to the fact that any border point is examined exactly once in each iteration.

Thesis Group III.

Distance-based skeletonization methods rely on the distance transform (DT), which assigns to all black points the distance from its closest white point. With a geometrically precise distance function (i.e., the Euclidean distance or its 'good' approximations), this strategy can guarantee that the skeleton is placed in the center of the object. To ensure topology preservation and produce geometrically correct skeleton, distance information is often applied in thinning. Strand proposed the first – and, to the best of our knowledge, the only existing – surface-thinning algorithms for $(14, 14)$ pictures of the BCC grid and $(12, 12)$ pictures of the FCC grid. Both can be categorized as 'hybrid', since they combine the anchor-based and distance-driven approches.

In Chapter 5, we showed that Strand's methods have nonlinear time complexity. Furthermore, we established two, modified versions of Strand's methods – an anchor-based and a distance-driven one - for each of the two considered picture types, which are proved to be linear. In addition, we presented two distance-driven thinning al-

gorithms, which directly extract the centerlines of object on $(14, 14)$ pictures of the BCC grid and $(18, 12)$ pictures of the FCC grid with two line-endpoint criteria for each considered picture types. Moreover, we measured the reconstructibility of the original objects from their produced centerline on the FCC grid. We determined that the reconstructed object from a more compact representation usually have significantly less black points because of the loss of details of its shape.

Contributions of the thesis

In the **first thesis group**, my contributions are related to constructing sufficient conditions for topology-preserving parallel reductions. Detailed discussion can be found in Chapter 3.

I/1. I verified the proposed configuration-based sufficient conditions for topology-preserving parallel reductions on $(14, 14)$ pictures of the BCC grid, and $(18, 12)$, $(12, 18)$, and $(12, 12)$ types of pictures of the FCC grid.

I/2. I verified the proposed symmetric and asymmetric point-based sufficient conditions for topology-preserving parallel reductions on $(14, 14)$ pictures of the BCC grid, and all three types of pictures of the FCC grid.

I/3. I implemented the two topology-preserving parallel reductions derived from the symmetric and asymmetric point-based conditions on $(14, 14)$ pictures of the BCC grid, and all three types of pictures of the FCC grid, and identified their support.

In the **second thesis group**, my contributions are related to constructing parallel kernel-thinning algorithms. Detailed discussion can be found in Chapter 4.

II/1. I implemented the (topology-preserving) fully parallel kernel-thinning algorithm on $(14, 14)$ pictures of the BCC grid, and proved its capability of producing topological kernels.

II/2. I implemented the (topology-preserving) fully parallel kernel-thinning algorithms on all three types of pictures of the FCC grid.

II/3. I implemented and evaluated the proposed kernel-thinning algorithms working with 4 and 8 subfields on $(14, 14)$ pictures of the BCC grid.

In the **third thesis group**, my contributions are related to constructing computationally efficient sequential thinning methods to extract medial surfaces and centerlines. Detailed discussion can be found in Chapter 5.

III/1. I reproduced Strand's sequential surface-thinning algorithms combined with neighbor-based distance information on $(14, 14)$ pictures of the BCC grid and $(12, 12)$ pictures of the FCC grid, and proved their runtime complexity.

III/2. I modified Strand's methods to produce surface skeletons in linear-time on both considered types of pictures.

III/3. I constructed a distance-driven sequential thinning method with two line endpoint criteria to extract curve skeletons directly from the input object on $(14, 14)$ pictures of the BCC grid.

III/4. I constructed a distance-driven sequential thinning method with two line endpoint criteria to extract curve skeletons directly from the input object on $(18, 12)$ pictures of the FCC grid. I measured this algorithm's execution time and the reconstructibility of the original objects from their produced centerline.

Összefoglalás

Dolgozatom a topológia-megőrző redukciók és a vázkijelölés (vagyis a digitális képek vázszerű alakjellemzőinek meghatározása) területén elért eredményeimet mutatja be, melyek a tércentrált (BCC) és a lapcentrált (FCC) kockarácsokon értelmezett képekre vonatkoznak. A 2D objektumok vázszerű jellemzői a topológiai mag és a középvonal, 3D-ben pedig még a középfelszín is. Az eredményül kapott ,,váznak" topológikusan ekvivalensnek kell lennie a kiindulási objektummal, vagyis az eredményképeket topológia-megőrző redukciókkal kell előállítani. A vékonyítás, vagyis az objektumok iteratív redukciója alkalmas módszer valamennyi vázszerű jellemző meghatározására.

Az értekezés második fejezetében áttekintem a digitális topológia alapfogalmait, a topológia megőrzés és a vázkijelölés területének előzményeit. A következő három fejezetben rendre a három téziscsoport eredményeit ismertetem.

1. rész

A szekvenciális redukciók egyszerre csak egy objektumpontot törölhetnek, ezért a topológia megőrzése garantált, ha a törölt pont egyszerű. Ezzel szemben a párhuzamos redukciók egyszerre több pontot is törölhetnek. Hogy megbizonyosodjunk arról, hogy egy párhuzamos redukció megőrzi a topológiát, Ronse minimálisan nem egyszerű (MNS) halmazokra vonatkozó tételére támaszkodtunk: ha egy párhuzamos redukció soha nem töröl MNS halmazokat, akkor topológia-megőrző. Gau és Kong korábbi munkájuk során feltérképezték az FCC rácson lehetséges összes MNS halmazt. Kardos pedig a BCC rácson előforduló MNS halmazokra adott jellemzést.

A 3. fejezetben nyolc konfiguráció-alapú elegendő feltételt adtunk meg topológia-megőrző párhuzamos redukciókhoz a vizsgált rácsokon. Ezek csupán már létező párhuzamos redukciók topológia megőrzésének ellenőrzésére alkalmasak, de nem használhatók új törlési szabályok meghatározására redukciós műveletekhez. Ezért összesen tíz pont-alapú elégséges feltételt is javasoltunk, melyekből közvetlenül származtathatók törlési szabályok topológia-megőrző párhuzamos redukciók számára ezeken a nem konvencionális rácsokon. Az összes feltétel topológia-megőrző jellegének bizonyítása a vonatkozó publikációimban található meg. A pont-alapú

szabályok közül, a lexikografikus rendezés alkalmazásával, elkülönítettük az aszimmetrikus feltételeket a szimmetrikusaktól.

Ezenkívül a javasolt pont-alapú feltételeinkből topológia-megőrző redukciókat származtattunk. Az aszimmetrikus redukciók egyszerűbbek, mint a szimmetrikusak, ugyanis a szupportjuk kevesebb pontot vesz figyelembe. Mindezek alapján szimmetrikus és aszimmetrikus vékonyító algoritmusok családjait hozhatjuk létre.

2. rész

A 4. fejezetben topológia-megőrző teljesen párhuzamos vékonyító algoritmusokat konstruáltunk a 3. fejezetben tárgyalt, általunk származtatott aszimmetrikus redukciók felhasználásával. Mivel nem alkalmaztunk geometriai kényszerfeltételt, ezek a eljárások zsugorító algoritmusoknak tekinthetők. Bebizonyítottuk, hogy ezek a módszerek nem hagynak egyszerű pontot az eredményképeken, így mindegyikük a BCC és FCC rácsokon mintavételezett objektumok topológiai magját jelöli ki.

Ezenkívül felosztottuk a BCC rács pontjait négy és nyolc almezőre úgy, hogy minden pont más almezőbe kerüljön, mint bármelyik 14-szomszédja. Ennek a kedvező tulajdonságnak köszönhetően leegyszerűsíthettük a BCC rácson értelmezett, topológia-megőrző párhuzamos redukciókra vonatkozó konfiguráció-alapú feltételt: az azonos almezőbe tartozó összes egyszerű pont egyidejű törlése topológia-megőrző redukcióhoz vezet. Ezeket a redukciókat fehasználva két darab almező-alapú párhuzamos zsugorító algoritmust konstruáltunk a BCC rácson. Beláttuk, hogy ennek a két eljárásnak hasonló a futási ideje, mivel minden határpontot pontosan egyszer vizsgálunk meg minden iterációban.

3. rész

A távolság-alapú vázkijelölő módszerek a távolság transzformáción (DT) alapulnak, amely minden fekete ponthoz a hozzá legközelebbi fehér ponttól való távolságát rendeli hozzá. Geometriailag pontos távolság függvénnyel (vagyis az euklideszi távolsággal vagy annak ,,jó" közelítéseivel) ez a stratégia garantálja, hogy a vázszerű jellemző az objektum közepén helyezkedjen el. A topológia megőrzése és a geometriailag pontos váz kinyerése érdekében a távolság információt gyakran alkalmazzák a vékonyítás során. Strand javasolta az első – és legjobb tudásunk szerint csak ezen két létező – középfelszínre vékonyító algoritmust a BCC rács $(14, 14)$ képeire és az FCC rács $(12, 12)$ képeire. Mindkettő a ,,hibrid" eljárások közé sorolható, mert ötvözik a horgony-alapú és a távolság-rendezett módszereket.

Az 5. fejezetben megmutattuk, hogy Strand eljárásainak futási komplexitása nemlineáris. Továbbá kidolgoztuk két módosított változatukat – egy horgony-alapú és

egy távolság-rendezett verziót – mindkét vizsgált képtípushoz, amelyeknek lineáris futási komplexitását bebizonyítottuk. Ezenkívül konstruáltunk két távolság-rendezett vékonyító algoritmust, amelyek közvetlenül az objektumok középvonalát állítják elő a BCC rács $(14, 14)$ képein és az FCC rács $(18, 12)$ képein, méghozzá két vonalvégpont-kritériummal mindkét vizsgált képtípusra. Mindezeken túl megvizsgáltuk az eredeti objektumok rekonstruálhatóságát a kinyert középvonalukból kiindulva az FCC rácson. Megállapítottuk, hogy a kompaktabb középvonalból rekonstruált objektumok általában jelentősen kevesebb fekete pontot tartalmaznak, mivel az eredeti alakjuk részletei elvesznek a tömörebb reprezentáció miatt.

A disszertáció tézisei

Az **első téziscsoportban** a hozzájárulásaim az elegendő feltételek kidolgozásához kapcsolódnak topológia-megőrző párhuzamos redukciók számára. A részletes bemutatás a 3. fejezetben található.

I/1. Ellenőriztem a topológia-megőrző párhuzamos redukciókra javasolt konfiguráció-alapú elegendő feltételeket a BCC rács $(14, 14)$ képeire, valamint az FCC rács mindhárom képtípusára, vagyis a $(18, 12)$, $(12, 18)$ és $(12, 12)$ képekre.

I/2. Ellenőriztem a topológia-megőrző párhuzamos redukciókra javasolt szimmetrikus és aszimmetrikus pont-alapú elegendő feltételeket a BCC rács $(14, 14)$ képeire, valamint az FCC rács mindhárom képtípusára.

I/3. Implementáltam a topológia-megőrző párhuzamos redukciókra javasolt szimmetrikus és aszimmetrikus pont-alapú elegendő feltételekből származtatott párhuzamos redukciókat a BCC rács (14,14) képeire, valamint az FCC rács mindhárom képtípusára, és meghatároztam a szupportjukat.

A **második téziscsoportban** a hozzájárulásaim a párhuzamos zsugorító algoritmusok konstruálásához kapcsolódnak. A részletes bemutatás a 4. fejezetben található.

II/1. Implementáltam egy teljesen párhuzamos (topológia-megőrző) zsugorító algoritmust a BCC rács $(14, 14)$ képeire, és bebizonyítottam, hogy az algoritmus topológiai magot állít elő.

II/2. Implementáltam a teljesen párhuzamos (topológia-megőrző) zsugorító algoritmusokat az FCC rács mindhárom képtípusára.

II/3. Implementáltam és kiértékeltem a 4- és 8-almezős zsugorító algoritmusokat a BCC rács $(14, 14)$ képeire.

A **harmadik téziscsoportban** a hozzájárulásaim hatékony számításigényű szekvenciális vékonyító algoritmusok konstruálásához kapcsolódnak középfelszín és középvonal kijelölésére. Részletes bemutatás az 5. fejezetben található.

III/1. Reprodukáltam Strand szekvenciális, középfelszínre vékonyító, szomszédság-alapú távolság információval kombinált algoritmusait a BCC rács $(14, 14)$ képeire és az FCC rács $(12, 12)$ képeire, valamint bebizonyítottam a futási komplexitásukat.

III/2. Kidolgoztam Strand algoritmusainak két, bizonyítottan lineáris időigényű módosított változatát mindkét vizsgált képtípuson.

III/3. Konstruáltam egy távolság-rendezett bejárási sorrendet követő, középvonalra vékonyító eljárást két vonalvégpont-kritériummal a BCC rács $(14, 14)$ képeire.

III/4. Konstruáltam egy távolság-rendezett bejárási sorrendet követő, középvonalra vékonyító eljárást két vonalvégpont-kritériummal az FCC rács $(18, 12)$ képeire. Megvizsgáltam az eljárás futási idejét és az eredeti objektumok rekonstruálhatóságát a kinyert középvonalukból.

Publications

Journal publications

[34] **Karai, G.**, Kardos, P.: Distance-Based Skeletonization on the BCC Grid. *Acta Cybernetica* vol. 25(2), 351–367, 2021

[37] **Karai, G.**, Kardos, P., Palágyi, K.: Sufficient Conditions for Topology-Preserving Parallel Reductions on the Face-Centered Cubic Grid. *J. Math. Imaging Vis.* vol. 66, 271–292, 2024

Full papers in conference proceedings

[32] **Karai, G.**: Distance-Oriented Surface Skeletonization on the Face-Centered Cubic Grid. In *Proc. 1st Int. Joint Conf. on Discrete Geometry and Mathematical Morphology – DGMM 2021*, LNCS vol. 12708, 177–188, Springer, 2021

[35] **Karai, G.**, Kardos, P., Palágyi, K.: Subfield-based Parallel Kernel-thinning Algorithms on the BCC Grid. In *Proc. 11th Int. Conf. on Patt. Rec. Applications and Methods – ICPRAM 2022*, 288–295, SciTePress, 2022

[33] **Karai, G.**: Distance-Driven Curve-Thinning on the Face-Centered Cubic Grid. In *Proc. 2nd Int. Joint Conf. on Discrete Geometry and Mathematical Morphology – DGMM 2022*, LNCS vol. 13493, 354–365, Springer, 2022

[64] Palágyi, K., **Karai, G.**, Kardos, P.: Sufficient Conditions for Topology-Preserving Parallel Reductions on the BCC Grid. In *Proc. 21st Int. Workshop on Combinatorial Image Analysis – IWCIA 2022*, LNCS vol. 13348, 71–83, Springer, 2023

[38] **Karai, G.**, Kardos, P., Palágyi, K.: Topology-Preserving Reductions on (18,12) Pictures of the Face-Centered Cubic Grid. In *Proc. 12th Int. Conf. on Patt. Rec. Applications and Methods – ICPRAM 2023*, 254–261, SciTePress, 2023

[36] **Karai, G.**, Kardos, P., Palágyi, K.: Fully Parallel Kernel-Thinning on the Face-Centered Cubic Grid. In *Proc. 11th Hungarian Conference on Computer Graphics and Geometry – GRAFGEO 2024*, ISBN: 978-963-421-948-4, 32–37, 2024

Supporting projects

Researches presented in [32, 34] were supported by the project “Integrated program for training new generation of scientists in the fields of computer science”, no EFOP-3.6.3-VEKOP-16-2017-00002. The project has been supported by the European Union and co-funded by the European Social Fund.

The further researches were supported by project TKP2021-NVA-09. Project no TKP2021-NVA-09 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021-NVA funding scheme.

Acknowledgments

First of all, I would like to thank my supervisors, **Kálmán Palágyi** and **Péter Kardos**, for their professional guidance during my PhD studies. Kálmán's professional routine facilitated smooth scientific progress for me. Péter inspired me to take a deep dive into the world of skeletonization during my BSc studies. He introduced me to several alternative grids – first the hexagonal grid, then the BCC and FCC grids – and some open problems related to skeletonization. It was thanks to him that I started my research career. **Gábor Németh** for his valuable advice on both research and educational matters. Finally, I would like to thank **Iván Devosa** who, together with Péter, taught me about craft beers, which helped me through the tough times.