An Order–Independent Sequential Thinning Algorithm

Péter Kardos, Gábor Németh, and Kálmán Palágyi

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

Abstract. Thinning is a widely used approach for skeletonization. Sequential thinning algorithms use contour tracking: they scan border points and remove the actual one if it is not designated a skeletal point. They may produce various skeletons for different visiting orders. In this paper, we present a new 2-dimensional sequential thinning algorithm, which produces the same result for arbitrary visiting orders and it is capable of extracting maximally thinned skeletons.

Keywords: shape representation, skeleton, thinning, topology preservation, order-independent thinning.

1 Introduction

Thinning algorithms [11] perform iterative object reductions. They delete some border points of binary objects that satisfy certain topological and geometrical constrains. Sequential thinning algorithms delete only one non-skeleton point at a time [1,7,8], hence it is easy to preserve the topology. However, they have the disadvantage that they can result various skeletons for different point–visiting orders. This can be explained with the fact that there can occur some pairs of points in the picture, from which any arbitrary point can be deleted without altering the topology, but removing the whole pair would split an object into two components, which is not allowed.

Although the idea of order–independence is already discussed, this area still yields questions for further research. Earlier, Ranwez and Soille [9], then Iwanowski and Soille [4,5] proposed order–independent sequential thinning algorithms. However, these algorithms simply preserve the pair of points, which fulfill the property mentioned above, so they cannot extract 1–point thin skeletons. In addition, these methods themselves can be used for reductive shrinking [3] (i.e., they can transform an object into a minimal structure, which is topologically equivalent to the original one, but some important geometrical information relative to the shape of the objects are vanished). In case of the existing order–independent algorithms, thinning can only be performed by a two–phase process: first, some endpoints are detected as anchors, then the branches relative to them are created by anchor–preserving shrinking.

P. Wiederhold and R.P. Barneva (Eds.): IWCIA 2009, LNCS 5852, pp. 162–175, 2009.

In this paper, we present a novel order–independent sequential algorithm. It preserves the topology and it has two advantageous properties over the previously published algorithms: our algorithm is maximal (i.e., it produces 1–point thin skeletons) and it is shape–preserving without an endpoint–detection phase.

2 Basic Notions and Results

The points of an image can be considered as a set of points in 2-dimensional digital space denoted by $\mathbb{Z}^2$. Each point p is represented as a pair $p = (p_x, p_y)$. Let $N_4(p) = \{q = (q_x, q_y) \in \mathbb{Z}^2 : |p_x - q_x| + |p_y - q_y| \leq 1\}$ and $N_8(p) = \{q = (q_x, q_y) \in \mathbb{Z}^2 : \max(|p_x - q_x|, |p_y - q_y|) \leq 1\}$. A point $q \in N_4(p)$ is called *4–adjacent* to p, while a point $q \in N_8(p)$ is *8-adjacent* to p. Further on, the notations $N_4^*(p) = N_4(p) \setminus \{p\}$ and $N_8^*(p) = N_8(p) \setminus \{p\}$ will be used to refer to the sets of the proper neighbors of p. The sequence $\langle s_0, s_1, \ldots, s_n \rangle$ of distinct points is a *j-path* if each point of the sequence is in S and for all i $(1 \leq i \leq n)$ s_i is j–adjacent to s_{i-1}. The point $s_1 \in S$ is *j–connected* in the set S, if there exists a j–path in S between s_1 and s_2. For any two sets S, S' for which $S' \supseteq S$, S is j–connected $(j = 4, 8)$ to S', if any two points of S are j–connected in S'. The notation $\overline{S}$ will be also used to refer to the complement of S.

A 2-dimensional $(8,4)$ *binary digital picture* (in the following referred to as $(8,4)$ picture or simply as picture) can be described with the quadruple $(\mathbb{Z}^2, 8, 4, \mathbf{B})$ [6], where $\mathbb{Z}^2$ is the set of picture points, $\mathbf{B} \subseteq \mathbb{Z}^2$ is the set of black points, for which we will assign the value "1"; its complement, $\overline{\mathbf{B}} = \mathbb{Z}^2 \backslash \mathbf{B}$ is the set of white points to which the value "0" is assigned. A *black component* is a maximal 8–connected set of black points, while a *white component* is defined as a maximal 4–connected set of white points.

A black point p in the picture $(\mathbb{Z}^2, 8, 4, \mathbf{B})$ is said to be a *border point*, if it is 4–adjacent to at least one white point. A black point p is called an *interior point* if it is not a border point. Let us denote by $B(p)$ the number of elements of the set $N_8^*(p) \cap \mathbf{B}$. Let $p \in \mathbf{B}$ be a black point in $(\mathbb{Z}^2, 8, 4, \mathbf{B})$. Let us denote by $C(p)$ and $A(p)$ the number of the 8-connected and 4-connected black components in the picture $(\mathbb{Z}^2, 8, 4, \mathbf{B} \cap N_8^*(p))$, respectively.

In order to preserve the shape of the objects, the so–called endpoints of an object must be retained in the picture during the whole thinning process. For our proposed algorithm, we will use the following endpoint–criterion [2]: the black point $p \in \mathbf{B}$ in $(\mathbb{Z}^2, 8, 4, \mathbf{B})$ is an *endpoint* if and only if $B(p) = 1$ or 2 and $A(p) = 1$.

A black point p is said to be a *simple point*, if its deletion (i.e., changing it to white) preserves the topology of the picture [6]. There are numerous characterizations of simple points. We make use the following one.

Theorem 1. *[2] Black point p is simple in a (8,4) picture if and only if p is a border point and* $C(p) = 1$.

Furthermore, we say that the object point p is a *corner point*, if it suits any of the configurations in Fig. 1.

1	1	0
1	p	0
0	0	0

0	1	1
0	p	1
0	0	0

0	0	0
0	p	1
0	1	1

0	0	0
1	p	0
1	1	0

Fig. 1. Configurations of corner points

3 Decision Pairs and Their Properties

Let p and q be two 4–adjacent object points in $P = (Z^2, 8, 4, \mathbf{B})$. Let us introduce the notations $N_8(p,q) = N_8(p) \cup N_8(q)$ and $N_8^*(p,q) = N_8(p,q) \setminus \{p,q\}$. Henceforth, let $C(p,q)$ denote the number of black 8–components in $N_8^*(p,q)$.

Definition 1. *The pair of 4–adjacent border points* $\{p,q\}$ *is called a* decision pair*, if the following conditions hold:*

i) $C(p) = C(q) = 1$*,*
ii) neither p *nor* q *is an endpoint,*
iii) $C(p,q) = 2$*.*

It is easy to discover that a decision pair is a special minimal non-deletable set [10], but of course, these two notions do not coincide with each other (for example, consider a 2×2 square object). Actually, we could define a decision pair as a minimal non-deletable set of two 4-adjacent non-endpoints, however, Definition 1 is more precise and more useful for our purposes.

The fact that the set $\{p,q\}$ is a decision pair in a picture, is denoted with the term $\Gamma(p,q)$. We also define a relation $\prec$ between two 8–adjacent points $p = (p_x, p_y)$ and $q = (q_x, q_y)$ in the following way: $p \prec q$ holds if and only if $p_x + p_y < q_x + q_y$.

0	0	0	0
1	1	p	1
1	1	q	1
0	0	0	0

(a)

0	1	0	1	0
0	p	q	r	0
0	1	1	1	0
1	0	1	0	1
0	0	1	0	0

(b)

Fig. 2. Examples for comparing members of decision pairs with relation $\prec$: $p \prec q$ (a) and $p \prec q \prec r$ (b)

This relation is useful for order–independent rules in thinning. Let us consider, for example, the situation in Fig. 2(a), where $\{p,q\}$ is a decision pair, and neither p nor q is a member of any other decision pair. In such cases, it would be easy to define a proper rule for order–independent thinning: we could simply use the relation $\prec$ to decide which point to prefer, as exactly one of the conditions $p \prec q$ and $q \prec p$ holds. However, in some cases an object point can be a member of more than one decision pairs. In Fig. 2(b), for example, $p \prec q$ and $q \prec r$, and this makes clear, that if we would like to define correct rules for order–independent thinning, it is not enough to take only the relation $\prec$ into account. That is why we give the following classification of points, which will help to set up another kind of precedence between the points of a decision pair.

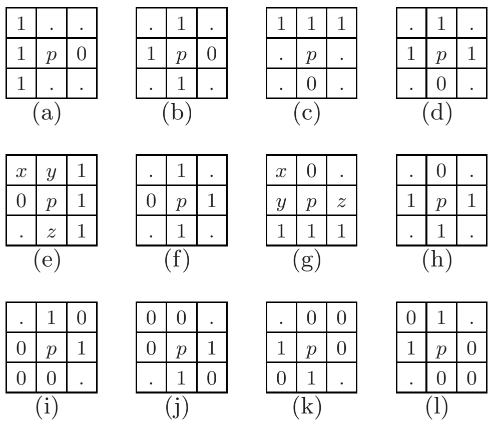


Fig. 3. In configurations (a)–(d), (e)–(h), and (i)–(l) p is an α–point, β–point, and γ–point, respectively. In configurations (e) and (g) $x \wedge y = 0$ or $z = 1$. A point marked "." may be either 0 or 1.

Let us consider the configurations shown in Fig. 3. The object point p in a picture will be called *α–point*, *β–point*, *γ–point*, and *safe γ–point*, if $N_8(p)$ matches any of the configurations in 3(a)–(d), (e)–(h), (i)–(l), and (i)–(j), respectively. The decision pairs will be called *α–pair*, *β–pair*, *γ–pair*, if both of their elements are α–, β–, γ–points, respectively, and we will talk about *$\alpha\beta$–, $\alpha\gamma$–, $\beta\gamma$–pairs*, if they contain exactly the denominated types of points. Here we discuss some important properties of decision pairs.

First, we show a necessary property of decision pairs, which is easier to check than the conditions in Definition 1, therefore we will make use of it in later proofs.

Proposition 1. *If $\{p, q\}$ is a decision pair, then $N_8(p, q)$ matches at least one of the configurations of Fig. 4 or their rotations by 90, 180, or 270.*

Proof. We give an indirect proof. Let us suppose that the decision pair $\{p, q\}$ does not fulfill the property stated in the proposition. Then, $N_8(p, q)$ must suit one of the configurations of Fig. 5 or their rotations by 90, 180 or 270. As $C(p, q) = 2$ holds, by Definition 1, it is easy to check that $\{x_2, x_4, x_5, x_7\} \cap \overline{\mathbf{B}} \neq \emptyset$. If $\{x_4, x_5\} \cap \overline{\mathbf{B}} \neq \emptyset$ held, then one of the conditions *i)* and *ii)* in Definition 1 would not be satisfied. Hence, $\{x_4, x_5\} \subset \mathbf{B}$.

.	1
p	q
.	1

1	.
p	q
1	.

1	.
p	q
.	1

.	1
p	q
1	.

Fig. 4. Possible configurations of the decision pair $\{p, q\}$, when q is the right neighbor of p. A point marked "." may be either 0 or 1.

Let us assume that $x_2 \in \overline{\mathbf{B}}$. From here, if $x_1 \in \overline{\mathbf{B}}$, then according to condition *iii*) of Definition 1, $x_7 \in \overline{\mathbf{B}}$ and $x_8 \in \mathbf{B}$, which implies $C(q) = 2$. On the other hand, $x_1 \in \mathbf{B}$ implies $C(p) = 2$. This means, that both of these cases conflict with condition *i*) of Definition 1, consequently, $x_2 \in \mathbf{B}$. Because of the symmetry of Fig. 5, we can deduce similar contradiction for the case $x_7 \in \overline{\mathbf{B}}$ as for $x_2 \in \overline{\mathbf{B}}$. □

x_1	0	0	x_8
x_2	p	q	x_7
x_3	x_4	x_5	x_6

x_3	x_4	x_5	x_6
x_2	p	q	x_7
x_1	0	0	x_8

Fig. 5. Configurations assigned to Proposition 1. $\{p, q\}$ cannot be a decision pair.

The following two propositions ensure that in α–pairs and β–pairs the relation $\prec$ can be used to unambiguously decide which point to prefer.

Proposition 2. *An object point may belong to at most one α–pair.*

Proof. Let $\{p, q\}$ be an α–pair. By examining the templates of α–points in Fig. 3(a)–(d), we can claim that the right or bottom 4-neighbor of an α–point must be a background point. Furthermore, according to Proposition 1, 4–neighbors of p and q from the same direction can not be both background points. Taking these observations into account it is easy to check that Fig. 6 shows all the possible 3×3 environments of p. If $x \in \mathbf{B}$, then x cannot be an α–point since none of its conceivable 3×3 environments match any of the configurations in Fig. 3(a)–(d). If $y_1 \in \mathbf{B}$ and y_1 is also an α–point, then, by necessity, $y_2 \in \overline{\mathbf{B}}$. But from this follows, according to Proposition 1, that the set $\{p, y_1\}$ cannot be a decision pair. □

1	q	0
y_1	p	x
y_2	0	

(a)

	x	1
y_1	p	q
y_2	0	1

(b)

1	y_1	y_2
q	p	0
0	x	

(c)

	y_1	y_2
x	p	0
1	q	1

(d)

Fig. 6. Configurations assigned to Proposition 2. $\{p, q\}$ is an α–pair.

Proposition 3. *An object point may belong to at most one β–pair.*

Proof. Note that if we rotate all the configurations of the β–points by 180, we get a subset of all the possible templates of α–points (see Fig. 3(a)–(h)). Based on this observation, the proof of this proposition may be similar to the previous one. (The reason why we can not get all α–point environments with the mentioned rotation, is the constraint "$x \wedge y = 0$ or $z = 1$" in Fig. 3(e) and (g), but this means only that this time there are even less cases to consider than in the previous proof.) □

Even if we take both the relation $\prec$ and the type of points into account in determining preference rules for our goal, it still can happen that a point q is a member of decision pairs $\{p, q\}$ and $\{q, r\}$, plus q is preferred to p, but not preferred to r. The observation below will help to formulate a suitable deleting condition for such situations as well.

Proposition 4. *Let $\{p, q\}$ be a β–pair where $p \prec q$ and let $\{r, s\}$ be an α–pair, where $r \prec s$. If $r \in N_4^*(p)$, then the sets $\{p, r\}$ and $\{q, s\}$ are $\alpha\beta$–pairs.*

Proof. The cases to consider are illustrated in Fig. 7. By examining the possible configurations of β–points, we can note that both members of a β–pair must suit Fig. 3(f) or 3(h). Hence, by necessity, $\{a, c, e, f\} \subset \mathbf{B}$. $r \neq a$, otherwise $r \prec s$ would imply $s = p$, which cannot happen. Therefore, $r = c$, and $s \in \{d, e\}$. $d \in \overline{\mathbf{B}}$, or else p would not be a border point, which contradicts Definition 1. Thus $s = e$. In addition, $b \in \mathbf{B}$, otherwise r would not be an α–point. Consequently, based on Proposition 1, the sets $\{p, r\}$ and $\{q, s\}$ are $\alpha\beta$–pairs. □

	0	b	
a	p	c	d
0	q	e	
	f		

	a	0	
0	p	q	f
b	c	e	
	d		

Fig. 7. Configurations assigned to Proposition 4. Both points of the β–pair $\{p, q\}$ must suit one of the templates in Fig. 3(f) and (h).

The next proposition tells us, that a decision pair must consist of points belonging to either of the earlier defined three classes.

Proposition 5. *If p is a member of a decision pair, then p must be an α–, β–, or γ– point.*

Proof. Let us suppose that $\{p, q\}$ is a decision pair. According to Proposition 1, $N_8(p, q)$ must match one of the templates in Fig. 4 or one of their proper rotated versions. If we complete these patterns to 3×3 environments of p in all the possible ways, then, by careful examination of the configurations in Fig. 3, we can observe that p must be an α–, β–, or γ– point. □

Finally, we point out to an important property of γ–pairs.

Proposition 6. *There is exactly one safe γ–point in a decision γ–pair.*

Proof. Let $\{p, q\}$ be a γ–pair. Without loss of generality we can assume that $p \prec q$. Fig. 8 shows the situations which can occur. In case (a),(b), and (d), only p is a safe γ–point in $\{p, q\}$, while in case (c), only q has this property in the considered pair. □

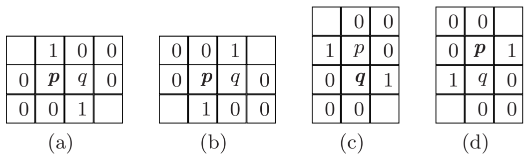


Fig. 8. Configurations assigned to Proposition 6. Only the highlighted point is safe γ-point in the γ-pair $\{p, q\}$.

4 The Proposed Algorithm KNP

Our algorithm called **KNP**[1] uses the subsets of black points denoted by S_{inner}, S_α, S_β, S_γ, S_{corner}, $S_{visited}$, and we need four additional Boolean functions. These functions have value of "1" (true) if the following conditions hold:

$\Delta_\alpha(p)$: $p \in S_\alpha$ and $\nexists q \in S_\alpha$ black point such that $\Gamma(p, q)$ and $p \prec q$.
$\Delta_\beta(p)$: $p \in S_\beta$, and the following two conditions hold:
i) If $\exists q \in S_\beta$ black point such that $\Gamma(p, q)$ and $q \prec p$, then $\exists r \in S_\alpha$ such that $\Gamma(q, r)$.
ii) For each $q \in S_\alpha$ black point, $\neg\Gamma(p, q)$ or $\exists r \in S_\alpha$ such that $\Gamma(q, r)$ and $q \prec r$.
$\Delta_\gamma(p)$: $p \in S_\gamma$, $\nexists q \in S_\alpha \cup S_\beta$ black point such that $\Gamma(p, q)$, and if p is not a safe γ–point, then $\nexists q \in S_\gamma$ black point such that $\Gamma(p, q)$.
$\Delta_{corner}(p)$: $p \in S_{corner}$ and $\exists q \in \big(S_{inner} \cap N_8^*(p)\big)$ black point or $\exists r \in \big(S_{corner} \cap N_8^*(p)\big)$ black point such that $r \prec p$.

Our preference rules for decision pairs are built in these Boolean functions. First of all, we have determined a priority order for the introduced classes of points, in which α–points are the most preferred, followed by β–points, which are preferred to γ–points, and finally, safe γ–points have higher priority than non–safe γ–points. If p has higher priority than q for every decision pair $\{p, q\}$, then p can be removed, but if q is preferred in such a pair, then p still has a chance to be removed, namely, if q is not preferred in another pair.

It can also occur that p and q have the same priority. (This can only happen in the case of α– and β–pairs, as we have already proved, that γ–pairs contain exactly one safe γ–point.) In such situations, we use the relation $\prec$ for the

[1] KNP stands for "Kardos-Németh-Palágyi", surnames of the authors.

decision: for α–points, we prefer p to q if $q \prec p$, however, in the case of β–pairs, $p \prec q$ must be fulfilled for deletion of p. (These opposite conditions for α– and β–pairs will help to ensure maximal thinning.)

Function $\Delta_\alpha(p)$ has only to check whether $q \prec p$ holds, as α–points have the highest priority. Function $\Delta_\beta(p)$ takes also care about the incidental α–points: criterion *i*) deals with the case when q is a preferred β–point in $\{p, q\}$ $(q \prec p)$, but not preferred in another decision pair, criterion *ii*) stands for the situation when q is an α–point which is not preferred in an α–pair. As it is noticeable, function $\Delta_\gamma(p)$ does not examine, whether the points with higher priority in $\alpha\gamma$– and $\beta\gamma$–pairs are also preferred in other incidental decision pairs. This has only a technical reason: sometimes we would have to analyze a relatively large environment of p for this purpose. Instead of this, we drop this check, and simply postpone the decision to a subsequent iteration, when p will be visited again. (As we will see later, the absence of the mentioned examination does not affect the expected properties of the algorithm.) Finally, function $\Delta_{corner}(p)$ has the aim to prevent the fully removal of a 2×2 square object by preserving its upper–left point.

Using these conditions we define the term

$$\Delta(p) : \Delta_\alpha(p) \vee \Delta_\beta(p) \vee \Delta_\gamma(p) \vee \Delta_{corner}(p).$$

We are now ready to give the pseudo–code of our algorithm:

Order–Independent Sequential Thinning Algorithm KNP
Input: picture $(Z^2, 8, 4, X)$
Output: picture $(Z^2, 8, 4, Y)$

```
S_α = S_β = S_γ = S_corner = S_visited = ∅
S_inner = Y = X
changed := true
repeat
        changed := false
        for each p ∈ Y do                                        /* Phase 1 */
                if p ∈ S_corner and B(p) > 0 then
                        Y := Y \ {p}
                        changed := true
                if p is a border point but not an endpoint then
                        S_inner := S_inner \ {p}
                        if p is an α–point then
                                S_α := S_α ∪ {p}
                        elseif p is a β–point then
                                S_β := S_β ∪ {p}
                        elseif p is a γ–point then
                                S_γ := S_γ ∪ {p}
                        elseif p is a corner point then
                                S_corner := S_corner ∪ {p}
```

for each $p \in \big(Y \cap (S_\alpha \cup S_\beta \cup S_\gamma \cup S_{corner})\big)$ **do** /* Phase 2 */
    if $p \notin S_{visited}$ and $C(p) = 1$ and $\Delta(p) = 1$ **then**
        $Y := Y \setminus \{p\}$
        changed := true
    if $p \notin S_\gamma$ **then**
        $S_{visited} := S_{visited} \cup \{p\}$
until changed=false

An iteration step (i.e., the kernel of the **repeat** cycle) consists of two phases. In Phase 1, border points are assigned to the sets S_α, S_β, S_γ, and S_{corner} according to their classes. Furthermore, an earlier labeled corner point can be removed in this phase if it is not an isolated point. Then, in Phase 2 the algorithm visits the actual assigned border points, and removes point p, if it fulfills the deleting conditions ($p \notin S_{visited}$ and $C(p) = 1$ and $\Delta(p) = 1$). The set $S_{visited}$ has the role to memorize the already visited but not deleted α–, β–, and corner points.

5 Discussion

Now we will show that the proposed algorithm is order–independent, topology preserving, and maximal thinning. The latter property, however, needs to be concretely defined beforehand, as there can occur some special cases, where only non–simple object points and endpoints are present in the image, yet the object is not 1–point wide (see Fig. 9, based on the examples in [2] and [12]).

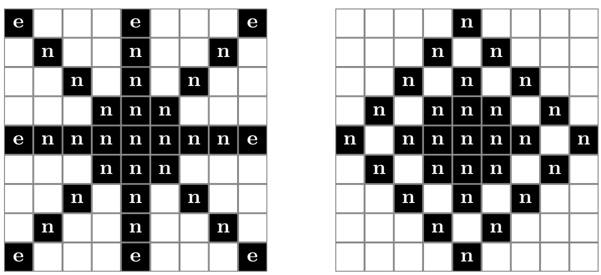


Fig. 9. Examples of objects which can not be thinned to 1-point thin line segments. Black points denoted by **e** are endpoints and points denoted by **n** are not simple.

Definition 2. *A thinning algorithm is* maximal*, if it produces a skeleton* $(Z^2, 8, 4, X)$ *such that for any* $x \in X$*,* x *is not a simple point or* x *is an endpoint.*

The following lemma is a preliminary result which shows that if $\{p, q\}$ is a decision pair, then we cannot have at the same time $\Delta(p)$ and $\Delta(q)$, implying that p or q must be retained in the picture.

Lemma 1. *In any iteration step of Algorithm* **KNP***,* $\neg\big(\Gamma(p,q) \wedge \Delta(p) \wedge \Delta(q)\big)$ *holds for any object points* p *and* q*.*

Proof. Let p and $q \in Y$ be two object points in a given iteration. It is obvious that $\Delta(p)$ can be satisfied only if $p \in S_\alpha \cup S_\beta \cup S_\gamma \cup S_{corner}$. If $\Delta_\alpha(p)$ holds, then it is easy to verify, according to Definition 1 and to the conditions $\Delta_\alpha(q), \Delta_\beta(q), \Delta_\gamma(q)$, that $\neg\big(\Gamma(p,q) \wedge \big(\Delta_\alpha(q) \vee \Delta_\beta(q) \vee \Delta_\gamma(q)\big)\big)$ holds. If $\Delta_\beta(p)$ is fulfilled, then it is obvious that $\neg\Delta_\gamma(q)$ also holds and from Proposition 4 follows that $\neg\big(\Gamma(p,q) \wedge \big(\Delta_\alpha(q) \vee \Delta_\beta(q)\big)\big)$ is satisfied. If both $\Delta_\gamma(p)$ and $\Gamma(p,q)$ holds, then $q \notin S_\alpha \cup S_\beta$, which means $\neg\big(\Delta_\alpha(q) \vee \Delta_\beta(q)\big)$ is fulfilled, and by Proposition 6, $\neg\Delta_\gamma(q)$ is also satisfied. Furthermore, due to Proposition 5, $\Delta_{corner}(p) \vee \Delta_{corner}(q)$ implies $\neg\Gamma(p,q)$. □

A necessary behavior of order–independent algorithms is that if a point p is the first visited point in an iteration and p is deletable in such a case, then p must fulfill the deleting conditions for any other visiting order as well. Of course, the same holds for the opposite case: if p could not be removed in the first iteration step, then p must be always preserved, no matter what the order of points is. Lemma 2 and Lemma 3 prepare the ground for the proof of order–independence by showing the above properties.

Lemma 2. *If $C(p) > 1$ or $\neg\Delta(p)$ holds for $p \in Y \cap \overline{S_{visited}} \cap \overline{S_\gamma}$ in the beginning of Phase 2, then the same holds when visiting p.*

Proof. Let us suppose that $C(p) > 1$ holds in either step of Phase 2 of a given iteration. Then, by examining the possible 8–neighbors of p, it can be shown that no one black component in $N_8^*(p)$ will be completely removed by the end of this iteration, because for at least one object point $q \in N_8^*(p)$ of any component, one of the following cases must occur:

- q was not a simple point or q was an endpoint in the beginning of Phase 2;
- q was a corner point for which $\neg\Delta(q)$ held in the beginning of Phase 2;
- there was an object point r in this component such that $\Gamma(q,r)$ and $\neg\Delta(q)$ held in the beginning of Phase 2.

In the first case, q is not visited in Phase 2. In the second case, it follows from the content of condition $\Delta_{corner}(q)$ that q remains in Phase 2. In the last case, if r was deleted, then according to the definition of decision pairs, $C(q)$=2 held right after the removal of r. By visiting q, the same can be stated for q as for p above, and this chain can be continued recursively until there will be an object point x for which one of the first two mentioned cases occur. As Y is finite, this chain is also finite. From this follows that q remains in the third case, too. Hence, the value $C(p)$ does not change in further steps of Phase 2. Now let us suppose that $\neg\Delta(p)$ holds in the beginning of Phase 2, and $\Delta(p)$ holds after some steps in this phase. By examining the deletion conditions, it is easy to see that in this case, there must have been an object point q for which $\Gamma(p,q)$ held in the beginning of Phase 2. But according to Definition 1, $C(p) = 2$ holds right after removing q, and we have already pointed to the fact that in such cases, $C(p)$ does not change anymore in Phase 2. □

Lemma 3. *If $C(p) = 1$ and $\Delta(p)$ holds for $p \in Y \cap \overline{S_{visited}}$ in the beginning of Phase 2 of a given iteration, then the same holds when visiting p.*

Proof. If $\Delta_{corner}(p)$ holds in the beginning of Phase 2, then based on the neighborhood of p and on the content of the mentioned condition, it is obvious that in any further step of this phase, $C(p) = 1$ and $\Delta_{corner}(p)$ holds. Let us suppose that $C(p) = 1$ and $\Delta_\alpha(p) \vee \Delta_\beta(p) \vee \Delta\gamma(p)$ is satisfied in the beginning of Phase 2, but $C(p) > 1$ or $\neg\Delta(p)$ holds in either further step of this phase. By examining the conditions and Definition 1, it can be stated that this can only happen if $\exists q \in N_4^*(p)$ such that $\Gamma(p,q)$ in the beginning of Phase 2 and q must be deleted before visiting p. That means, $C(q) = 1$ and $\Delta(q)$ must be satisfied in the beginning of this phase, or else according to Lemma 2, $C(q) > 1$ or $\neg\Delta(q)$ would hold in any step of this phase. But taking into account Lemma 1, $\Delta(p)$ and $\Delta(q)$ cannot hold at the same time. So we came to a contradiction, which means, our lemma is satisfied. □

Finally, Lemma 4 will be used to proove that our algorithm is maximal.

Lemma 4. *The output of Algorithm* **KNP** *does not contain any decision pair.*

Proof. Indirectly, let us suppose that there is at least one decision pair P in the output picture $(Z^2, 8, 4, Y)$. Y contains neither any α–points nor any β–points, or else based on Proposition 2 and Proposition 3, there would exist at least one object point $p \in Y$ satisfying condition $\Delta_\alpha(p)$ or $\Delta_\beta(p)$, and the algorithm would not have stopped. Therefore, according to Proposition 5, P must be a γ–pair. Due to Proposition 6, P must contain exactly one $p \in P$ safe γ–point, and as there is not any α–point or β–point in Y, $\Delta_\gamma(p)$ holds. This means, the algorithm continues with a further iteration, which conflicts with the initial assumption that the picture $(Z^2, 8, 4, Y)$ is the output of Algorithm **KNP**. □

Theorem 2. *Algorithm* **KNP** *is order–independent, topology preserving, and maximal.*

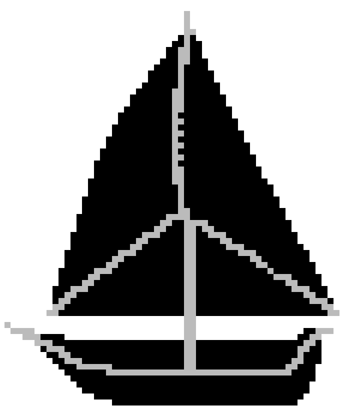
Proof. According to Lemmas 2 and 3, we can say that except for the γ–points, the same points will be deleted for any visiting order in Phase 2 of a given iteration. It is guaranteed that an object point $p \in S_\alpha \cup S_\beta$ can be visited only in one iteration, because after that, p is added to $S_{visited}$. It is also easy to see that if an object point $p \in S_\gamma$ is not deleted at the first check, then p will be deleted in the succeeding iteration if and only if for all $q \in N_4^*(p)$, $C(q) > 1$ or $\neg\Delta(q)$ holds in the beginning of Phase 2 of the iteration in which p is first checked. That means, if there exists any visiting order, for which an object point $p \in S_\gamma$ will be deleted in a given iteration, then p will surely be deleted at the latest in the succeeding iteration for each visiting order. There is a chance that a corner point, which has been visited in a given iteration, can be checked again for deletion in Phase 1 of the succeeding iteration. It is easy to see, that this can only happen to some corner points of the horizontal or vertical 2–point thick line objects and to the upper–left corner points of the 2×2 square objects, from which always the square–corner points will be preserved according to the condition $B(p) > 0$. From the observations above, it follows that the proposed algorithm is order–independent.

It is guaranteed that in Phase 2 an object point p will be removed only if it is a border point and $C(p) = 1$ holds. It is also easy to show that if we remove some (but not all) black neighbors of a corner point p, then $C(p) = 1$ still holds in Phase 1 of the next iteration. Therefore, after Theorem 1 only simple points will be deleted, which means Algorithm **KNP** preserves the topology of the objects.

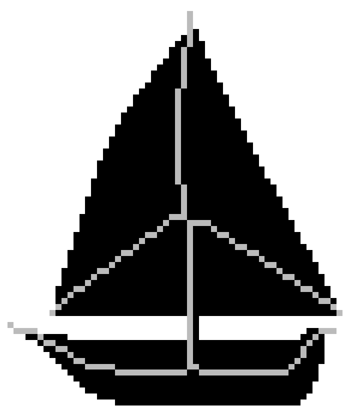
Let us suppose that our algorithm is not maximal, which means that in the last iteration of the algorithm, there is a simple point p in the picture, which is not an endpoint. Due to Lemma 4, there is not any decision pair in the skeleton. From this follows that $p \notin (S_\alpha \cup S_\beta \cup S_\gamma)$ in Phase 2 of the last iteration, otherwise $\Delta_\alpha(p) \vee \Delta_\beta(p) \vee \Delta_\gamma(p)$ would hold, which again would result in a contradiction. Hence, $p \in S_{corner}$ at the last visit of p. It can be easily verified that in this situation, p must be an upper–left corner point of a 2×2 square object $O = \{p, q, r, s\}$, or else p could be deleted. But in such an object, the conditions $\Delta_{corner}(q)$, $\Delta_{corner}(r)$, $\Delta_{corner}(s)$ hold, therefore, this situation cannot come into question. As there are no more possible cases to examine, we have indirectly proved that Algorithm **KNP** is maximal. □

6 Results

In experiments our Algorithm **KNP** and the previously published order–independent Algorithm **RS** proposed by Ranwez and Soille [9] were tested on numerous objects of different shapes and their results were compared. Here we present some illustrative examples below. In Figs. 10–13, "skeletons" produced by these two algorithms are superimposed on the original objects and numbers in parentheses mean the count of skeletal points. We can state that our algorithm (without an endpoint detection phase) does not produce spurious branches.

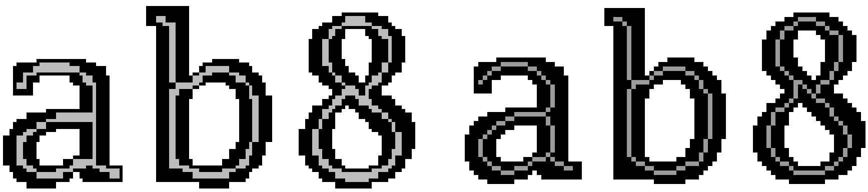


RS (247)



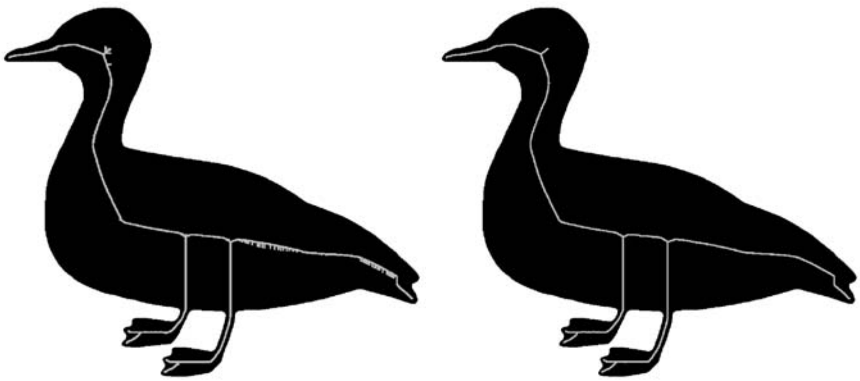
KNP (162)

Fig. 10. A 60×80 image of a ship and its "skeletons". Skeletal points are displayed in gray.



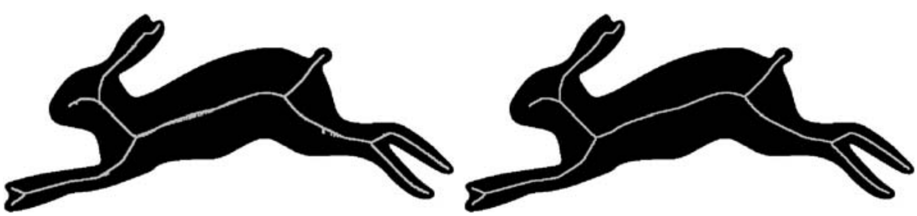
RS (331) **KNP** (235)

Fig. 11. A 100 × 40 image and its "skeletons"



RS (2740) **KNP** (1590)

Fig. 12. A 768 × 678 image of a duck and its "skeletons"



RS (1781) **KNP** (1085)

Fig. 13. A 712 × 412 image of a hare and its "skeletons"

Acknowledgements

This research was partially supported by the TÁMOP-4.2.2/08/1/2008-0008 program of the Hungarian National Development Agency.

References

1. Arcelli, C.: Pattern thinning by contour tracking. Computer Graphics and Image Processing 17, 130–144 (1981)
2. Hall, R.W.: Parallel connectivity–preserving thinning algorithm. In: Kong, T.Y., Rosenfeld, A. (eds.) Topological Algorithms for Digital Image Processing, Machine Intelligence and Pattern Recognition, vol. 19, pp. 145–179. North–Holland, Amsterdam (1996)
3. Hall, R.W., Kong, T.Y., Rosenfeld, A.: Shrinking binary images. In: Kong, T.Y., Rosenfeld, A. (eds.) Topological Algorithms for Digital Image Processing, Machine Intelligence and Pattern Recognition, vol. 19, pp. 31–98. North–Holland, Amsterdam (1996)
4. Iwanowski, M., Soille, P.: Order independence in binary 2D homotopic thinning. In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) DGCI 2006. LNCS, vol. 4245, pp. 592–604. Springer, Heidelberg (2006)
5. Iwanowski, M., Soille, P.: Fast algorithm for order independent binary homotopic thinning. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. LNCS, vol. 4432, pp. 606–615. Springer, Heidelberg (2007)
6. Kong, T.Y., Rosenfeld, A.: Digital topology: Introduction and survey. Computer Vision, Graphics, and Image Processing 48, 357–393 (1989)
7. Kwok, P.C.K.: A thinning algorithm by contour generation. Communications of the ACM 31, 1314–1324 (1988)
8. Lam, L., Lee, S.-W., Suen, C.Y.: Thinning methodologies – A comprehensive survey. IEEE Trans. Pattern Analysis and Machine Intelligence 14, 869–885 (1992)
9. Ranwez, V., Soille, P.: Order independent homotopic thinning for binary and grey tone anchored skeletons. Pattern Recognition Letters 23, 687–702 (2002)
10. Ronse, C.: Minimal test patterns for connectivity preservation in parallel thinning algorithms for binary digital images. Discrete Applied Mathematics 21, 67–79 (1988)
11. Suen, C.Y., Wang, P.S.P. (eds.): Thinning Methodologies for Pattern Recognition. Series in Machine Perception and Artificial Intelligence, vol. 8. World Scientific, Singapore (1994)
12. Verwer, B.J.H., Van Vliet, L.J., Verbeek, P.W.: Binary and grey-value skeletons: Metrics and algorithms. Int. Journal of Pattern Recognition and Artificial Intelligence 7, 1287–1308 (1993)