Algorithms for pixelwise shape deformations preserving digital convexity

Lama Tarsissi, Yukiko Kenmochi, Hadjer Djerroumi, David Coeurjolly, Pascal Romon, Jean-Pierre Borel

Algorithms for pixelwise shape deformations preserving digital convexity

Lama Tarsissi[1,2], Yukiko Kenmochi[3], Hadjer Djerroumi[1], David Coeurjolly[4], Pascal Romon[5], and Jean-Pierre Borel[6]

[1] Univ Gustave Eiffel, ESIEE Paris, CNRS, LIGM, Champs-sur-Marne, France
[2] Sorbonne University Abu Dhabi, United Arab Emirates
[3] Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France
[4] Univ Lyon, CNRS, LIRIS, Lyon, France
[5] Univ Gustave Eiffel, CNRS, LAMA, Champs-sur-Marne, France
[6] Univ Limoges, CNRS, XLIM, Limoges, France

Abstract. In this article, we propose algorithms for pixelwise deformations of digital convex sets preserving their convexity using the combinatorics on words to identify digital convex sets via their boundary words, namely Lyndon and Christoffel words. The notion of removable and insertable points are used with a geometric strategy for choosing one of those pixels for each deformation step. The worst-case time complexity of each deflation and inflation step, which is the atomic deformation, is also analysed.

1 Introduction

Convexity is an elementary geometric property of digital sets in digital image processing. There are various applications which require deforming digital convex sets while preserving their convexity. Various definitions of digital convex sets exist, among which we choose the one based on the convex hull [15]. Indeed, Brleck et al. have characterized such digital convex sets via the boundary words, which are encoded by the Freeman chain code [14]; for short, a 4-connected digital set is digital convex if and only if the Lyndon factorization of its boundary word is made of Christoffel words [7]. Thanks to this approach based on combinatorics on words, we recently considered the following question: given a finite 4-connected, digital convex set C, how can one find a point x of C (resp. its complement $\overline{C}$) such that $C \setminus \{x\}$ (resp. $C \cup \{x\}$) is still 4-connected and digitally convex? In order to answer this question, we characterized the two types of points; they are called removable and insertable points [21, 22].

In this article, following the approach based on combinatorics on words, we propose algorithms for pixelwise deformations of digital convex sets that preserve their convexity using the characterizations of removable and insertable points. The main contribution of this article is factorizing the inflation and deflation algorithms, whose time complexities are analysed, in order to propose the general deformation algorithm. A geometric strategy based on distance map is also used for choosing one of removal and insertable pixels for each deformation step.

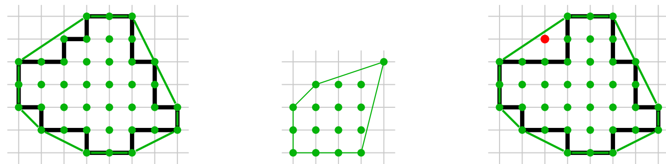

Fig. 1. Digitally convex sets with and without 4-connectivity (left and center) and digitally non-convex set (right). The sequence of border points is also illustrated by a thick black polygonal line for each 4-connected set (left and right).

Given a pair of digital convex sets, we show that the proposed algorithms create a sequence of digital convex sets, which is such a deformation between them. Some experimental results are illustrated.

2 Basic notions

2.1 Digital convex set

In $\mathbb{R}^2$, a subset S is convex if for any pair of points $x, y \in R$, every point on the straight line segment joining x and y is also within S. This notion, however, cannot be straightforwardly applied to subsets in $\mathbb{Z}^2$; various notions of convexity of a subset X of $\mathbb{Z}^2$ have been proposed. In this article, we focus on the following one [15] based on the convex hull, denoted by $conv(X)$ and also called H-convexity [13].

Definition 1 ([15]). *A subset X of $\mathbb{Z}^2$ is digitally convex if $X = conv(X) \cap \mathbb{Z}^2$.*

Figure 1 illustrates examples of digital convex and non-convex sets, based on this notion. The following remark warns us to pay attention to the connectivity separately from the convexity in $\mathbb{Z}^2$ (see Figure 1 (center)).

Remark 1. Digital convexity does not imply connectivity in $\mathbb{Z}^2$.

Concerning the connectivity of a digital convex set, there exists a homeomorphism that makes the set almost 4-connected [10] while an alternative definition for digital convexity, called full convexity, that encompasses arithmetic lines and naturally entails connectivity has been proposed [17].

Let us call convex polygons with vertices in $\mathbb{Z}^2$, digital convex polygons. The following property [1, 3] will help us to analyse the complexity of our deformation algorithms later.

Property 1 ([1, 3]). Given a digital convex polygon of diameter N, the number of its vertices is bounded by $\mathcal{O}(N^{\frac{2}{3}})$.

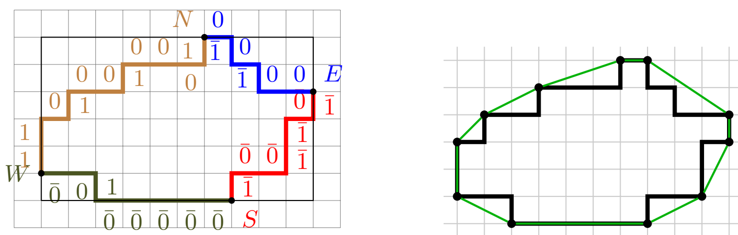

Fig. 2. The boundary word of a digital convex 4-connected set decomposed into four parts such that each part consists of a binary word (left), and the Lyndon points (black points) of the boundary word drawn as the black thick polygonal line (right).

2.2 Boundary words and some basic notions of words

Let $C \subset \mathbb{Z}^2$ be a finite, 4-connected digitally convex set. The border points of C can be tracked by classical border following algorithms (for example, see [2] for "left-hand-on-wall" border following), which generate a 4-connected sequence of the border points of C. Note that the sequence can include dead-ends and thus sometimes turnaround sub-sequences if the set contains thin parts of one pixel width. Here, we encode the sequence with Freeman code [14], called the boundary word of C, denoted by $Bd(C)$, in the clockwise order of border points. Boundary words are thus defined over an alphabet of four letters $0, 1, \bar{0}, \bar{1}$, which are associated to the right, up, left and down directions, respectively. The boundary word of a digital convex 4-connected set C is decomposed into four parts such that each part consists of two letters, as seen in Figure 2 (left): WN, NE, ES and SW.

Let us present some basic notions of words (see [18] for more complete overview): a nonempty finite set of letters is called an alphabet A; in this article, we have the four letters $0, 1, \bar{0}, \bar{1}$ as mentioned above. A word w is a sequence of concatenated letters from A. The empty word ϵ is a sequence of zero letter. A^* denotes the set of all finite words over A. The length of w is denoted by $|w|$ while $|w|_a$ represents the number of occurrences of a in w. The n-times concatenation of w is written by w^n. A word is said *primitive* if it is not the power of a nonempty word. A word w is *conjugate* of a word w' if w' can be obtained from w by cyclically shifting the letters.

2.3 Lyndon words and Lyndon factorization

We give the definition of Lyndon words, which is a necessary notion for the sequel.

Definition 2 ([19]). *A word w over a totally ordered alphabet is a Lyndon word if it is the smallest among all its conjugates.*

For example, $w = 00101$ where $0 < 1$ is a Lyndon word as w is the smallest among all its conjugates. The following proposition will play a leading role in our algorithms.

Proposition 1 ([8]). *Every non-empty word w over a totally ordered alphabet can be written uniquely as $w = \ell_1^{n_1}\ell_2^{n_2}\ldots\ell_k^{n_k}$ such that every factor ℓ_i is a Lyndon word and $\{\ell_i\}_i$ is a lexicographically decreasing sequence.*

This decomposition of w into ℓ_i is called Lyndon factorization. Given a word w of length N, the Lyndon factorization of w is calculated in $\mathcal{O}(N)$ time with a constant space [12]. The points on a word w that separate different Lyndon factors are called Lyndon points. Note that the two extremities of w are also Lyndon points. Let us consider a finite, 4-connected digitally convex set $C \subset \mathbb{Z}^2$ and its boundary word w. Then, the Lyndon points of w correspond to the vertices of the convex hull of C geometrically (see Figure 2 (right)).

2.4 Christoffel words

Christoffel words are another important notion in this article. Their geometrical definition can be formulated as:

Definition 3 ([4]). *The lower Christoffel word of slope $\frac{b}{a}$ is determined by encoding with Freeman chain code the Christoffel path, which is the discrete path from the origin O to the point $P(a, b)$ such that:*

- *the path lies below the line segment OP;*
- *the integer points in the region enclosed by the path and the line segment OP are exactly those of the path.*

Any Christoffel word with $\gcd(a, b) = 1$ is called primitive. Some properties of Christoffel words are presented as follows:

- A Christoffel word describes a shortest discrete path, so that it is always composed from two letters.
- Let c_1, c_2 be two Christoffel words over the alphabet $\{0, 1\}$. Then lexicographically $c_1 < c_2$ *iff* $slope(c_1) < slope(c_2)$ [6].
- Every primitive Christoffel word is a Lyndon word [5].

The converse of the last one is not true; for example, 0011 is a Lyndon word but not a Christoffel word.

In this article, we need the following specific points, called furthest points of Christoffel words: Figure 3 (left) illustrates an example of the lower Christoffel word of slope $\frac{4}{7}$ with its furthest point.

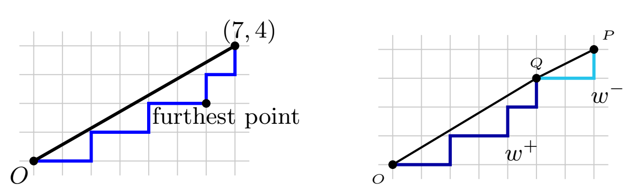

Fig. 3. The lower Christoffel word of slope $\frac{4}{7}$ with the furthest point (left) and its split (right).

Definition 4. *Given a primitive Christoffel word of slope* $\frac{b}{a}$*, the furthest point is uniquely defined on the path as the point whose vertical distance to the line segment joining* $(0,0)$ *and* (a,b) *is maximum.*

We will also need the diagonally opposite point of a furthest point, which is above the line segment OP (see Fig. 3), called the closest upper point.

2.5 Digital convex sets with combinatorics on words

By using Lyndon and Christoffel words, digital convex sets are characterized.

Proposition 2 ([7]). *A* 4*-connected set* $\mathcal{C} \subset \mathbb{Z}^2$ *is digitally convex iff its boundary word is decomposed into four binary subwords and each subword has the unique Lyndon factorization* $\ell_1^{n_1}\ell_2^{n_2}\dots\ell_k^{n_k}$ *such that all* ℓ_i *are primitive Christoffel words.*

The geometric interpretation of this proposition is that $\mathcal{C}$ is digitally convex iff the Lyndon factorization of $Bd(\mathcal{C})$ exactly corresponds to the segments of the convex hull of $\mathcal{C}$ (see Figure 1 (left and right) for positive and negative examples). This characterization will be used in the rest of this article.

3 Removable and insertable points

In this section, we consider the following problem: given a finite 4-connected digitally convex set C, how can one find a point x of C (resp. the complement $\overline{C}$) such that $C \setminus \{x\}$ (resp. $C \cup \{x\}$) is still digitally convex and 4-connected? The former points are called removable points while the latter ones are called insertable points (see Fig. 4 for examples). Their characterizations have been studied previously [21, 22]. We recall them in this section.

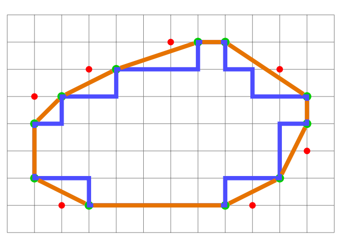

Fig. 4. Removable and insertable points, depicted in green and red respectively, for the boundary path of a digitally convex set, drawn as the blue polygonal line.

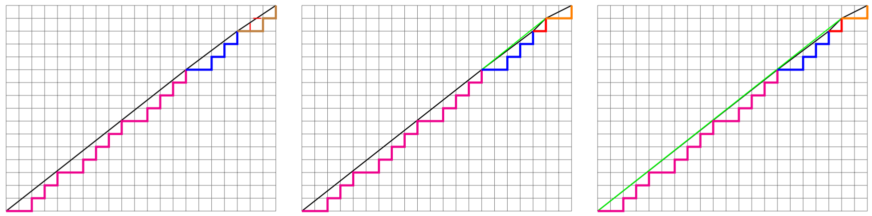

Fig. 5. Procedure of insertability verification on the left with propagation: the closest upper point $(19, 15)$ of ℓ_i (ℓ_i in brown) is inserted (left); as $\ell_{i-1} \leq L_0$ (ℓ_{i-1} in blue and $L_0 = \ell^+$ in red) is not satisfied but we have $\ell_{i-1} = \ell_i L_0$, we obtain $L_1 = \ell_{i-1} L_0$ (L_1 in green (=red+blue)) (center); as $\ell_{i-2} \leq L_1$ (ℓ_{i-2} in pink) is not satisfied but we have $\ell_{i-2} = \ell_{i-1} L_1^2$, we obtain $L_2 = \ell_{i-2} L_1$ (L_2 in green (=red+blue+pink)) (right).

3.1 Removable points

Let us consider that the boundary word w of a digitally convex 4-connected set C and its Lyndon factorization $\mathcal{L}(w)$ are given. Then we have the following theorem.

Theorem 1 ([21]). *A point x of C is removable iff x is a Lyndon point of $\mathcal{L}(w)$ and a simple point with respect to C.*

As the digital convexity does not imply the connectivity, as mentioned above, we need to add the simpleness condition that is also locally characterized [16]. Thanks to this theorem, we can find a position k where we can apply the following switch operator, which corresponds to removing the point at k. The switch operator on a word $w = a_1 \dots a_n$ at position $k < n$ is defined by $\text{switch}_k(w) = a_1 \dots a_{k-1} a_{k+1} a_k a_{k+2} \dots a_n$ where each a_i is a letter. If $a_k a_{k+1}$ consists of consecutive reverse letters, namely $0\bar{0}$, $\bar{0}0$, $1\bar{1}$, $\bar{1}1$, this operator will simply remove both of them, instead of the substitution.

Once a chosen removable point is removed by the switch operator, the following proposition tells us that updating the Lyndon factorization, namely updating the list of Lyndon points, can be made locally.

Proposition 3 ([21]). *Let u and v be two consecutive Christoffel words of the Lyndon factorization of a boundary word such that $u > v$. After applying the switch operator at $|u|$ on the binary word uv, if we obtain its Lyndon factorization $\mathcal{L}(switch_{|u|}(uv)) = \ell_1^{n_1} \dots \ell_m^{n_m}$, then $u > \ell_1$ and $\ell_1 > \dots > \ell_m$.*

3.2 Insertable points

Let us consider that the boundary word w of a digitally convex 4-connected set C and its Lyndon factorization $\mathcal{L}(w)$ are given. Then we have the following proposition.

Proposition 4 ([22]). *If a point x of $\overline{C}$ is insertable, then x is a closest upper point of $\mathcal{L}(w)$.*

It should be mentioned that the converse is not always true. Indeed, an insertable point is geometrically a point such that its convex hull with C does not contain any other integer point, and the proposition indicates that the union of C and a closest upper point is not always digitally convex.

Instead of the switch operator for removable points, here we use the split operator that is defined for a primitive Christoffel word c, $|c| > 1$, such that $split(c) = switch_k(c)$ where k is the furthest point of c (see [22] for the definition in the case of $|c| = 1$). In order to insert a closest upper point of C, the following standard factorization is used.

Definition 5 ([5]). *Any Christoffel word c with $|c| > 1$ can be written in a unique way as a product $c = uv$ such that u and v are both primitive Christoffel words. The couple (u, v) is called the standard factorization of c.*

Note that the standard factorization of c can be computed in $\mathcal{O}(\log |c|)$ due to its geometric interpretation [20].

The proposition below implies that the standard factorization gives the result of the split operator without knowing the position of the furthest point (see Fig. 3 (right) for an example of application of this split operator).

Proposition 5 ([11]). *Let c be a primitive Christoffel word, $|c| > 1$, such that its standard factorization is given by $c = c^-c^+$. Then, we have $split(c) = c^+c^-$ with $c^+ > c^-$.*

The following is the characterization of the insertability of such a closest upper point $x \in \overline{C}$.

Proposition 6 ([22]). *Given the boundary word w of a digitally convex 4-connected set C and its Lyndon factorization $\mathcal{L}(w) = \ell_1^{n_1} \dots \ell_m^{n_m}$, let x be the closest upper point in $\overline{C}$ of the j-th Lyndon factor of $\ell_i^{n_i}$ in $\mathcal{L}(w)$ such that $split(\ell_i) = \ell_i^+ \ell_i^-$ where $\ell_i = \ell_i^- \ell_i^+$. Let us say that:*

1. *x is insertable on the left if $\exists k \in \mathbb{Z}^*, \ell_{i-k-1} \geq L_k$ such that for every $h \leq k$, L_h is recursively defined by*

$$L_h = \begin{cases} \ell_i^{j-1}\ell_i^+ & \textit{for } h = 0 \\ \ell_{i-h}^{n_{i-h}} L_{h-1} & \textit{for } h \geq 1 \textit{ if } \exists m_{h-1} \in \mathbb{Z}^+, \ell_{i-h} = \ell_{i-h-1} L_{h-1}^{m_{h-1}} \end{cases}$$

2. *similarly, x is insertable on the right if $\exists k \in \mathbb{Z}^*, \ell_{i+k+1} \leq R_k$ such that for every $h \leq k$, R_h is recursively defined by*

$$R_h = \begin{cases} \ell_i^- \ell_i^{n_i - j} & \textit{for } h = 0 \\ R_{h-1} \ell_{i+h}^{n_{i+h}} & \textit{for } h \geq 1 \textit{ if } \exists m_{h-1} \in \mathbb{Z}^+, \ell_{i+h} = R_{h-1}^{m_{h-1}} \ell_{i+h-1} \end{cases}$$

Then, x is insertable if x is insertable on both sides.

This proposition indicates that the insertability cannot always be verified locally; see Figure 5 for an example with propagation. On the other hand, Lyndon re-factorization is not necessarily applied after adding an insertable point as a simple concatenation of Christoffel words provides new Lyndon factors, multiplicities and points.

Algorithm 1: Deflation

input : digitally convex 4-connected sets $\mathcal{A}$, $\mathcal{B}$ such that $\mathcal{A} \supset \mathcal{B}$
output : a sequence $\mathcal{T}_\ominus$ of points to remove from $\mathcal{A}$ to obtain $\mathcal{B}$

$w \leftarrow$ the boundary word of $\mathcal{A}$, $\mathcal{L} \leftarrow$ Lyndon Factorization of w;
calculate $d_{\mathcal{A}}^{\mathcal{B}}(x)$ for all $x \in \mathcal{A} \setminus \mathcal{B}$;
set the current deflated set $\mathcal{C} \leftarrow \mathcal{A}$, $\mathcal{T}_\ominus \leftarrow \varnothing$;
$\mathcal{Q}_\ominus \leftarrow$ UpdateRemovable$(\mathcal{Q}_\ominus, \mathcal{L}, \varnothing, (1, \ldots, |\mathcal{L}|), d_{\mathcal{A}}^{\mathcal{B}}, \mathcal{C})$;
while $\mathcal{Q}_\ominus \neq \varnothing$ **do**
    $i \leftarrow$ find_max$(\mathcal{Q}_\ominus)$;
    $(\ell, n, p) \leftarrow \mathcal{L}[i]$, push p to $\mathcal{T}_\ominus$, $\mathcal{C} \leftarrow \mathcal{C} \setminus \{p\}$;
    $I_{old} \leftarrow (i-1, i)$;
    $(\mathcal{L}, i') \leftarrow$ UpdateLyndonFactorizationDueToSwitch$(\mathcal{L}, i)$;
    $I_{new} \leftarrow (i-1, i, \ldots, i')$;
    $\mathcal{Q}_\ominus \leftarrow$ UpdateRemovable$(\mathcal{Q}_\ominus, \mathcal{L}, I_{old}, I_{new}, d_{\mathcal{A}}^{\mathcal{B}}, \mathcal{C})$;
end
return $\mathcal{T}_\ominus$

Function 2: UpdateLyndonFactorizationDueToSwitch

input : Lyndon factorization $\mathcal{L}$ and switch operator position k
output : updated Lyndon factorization $\mathcal{L}$ and last new factor index h

$(\ell_1, n_1, p_1) \leftarrow \mathcal{L}[k-1]$, $(\ell_2, n_2, p_2) \leftarrow \mathcal{L}[k]$;
$w \leftarrow \text{switch}_{|\ell_1|}\ell_1\ell_2$;
$\mathcal{L}_{new} \leftarrow$ the Lyndon factorization of w;
remove $\mathcal{L}[k-1], \mathcal{L}[k]$;
$h = k - 1$;
if $n_1 > 1$ **then** insert $(\ell_1, n_1 - 1, p_1)$ at $\mathcal{L}[h]$, $h \leftarrow h + 1$;
insert $\mathcal{L}_{new}$ at $\mathcal{L}[h]$, $h \leftarrow h + |\mathcal{L}_{new}|$;
if $n_2 > 1$ **then** insert $(\ell_2, n_2 - 1, p_2 - |\ell_2|_0 e_1 - |\ell_2|_1 e_2)$ at $\mathcal{L}[h]$;
else $h \leftarrow h - 1$;
return $\mathcal{L}, h$

4 Deformation preserving digital convexity

We now achieve our purpose of this article: given a pair of 4-connected digital convex sets, $\mathcal{A}, \mathcal{B} \subset \mathbb{Z}^2$, such that $\mathcal{A} \cap \mathcal{B} \neq \varnothing$, we would like to make a sequence of 4-connected digital convex sets, which represents a pixelwise deformation from $\mathcal{A}$ to $\mathcal{B}$. For each step, we remove or add a point of $\mathbb{Z}^2$ thanks to the notions of removable and insertable points. In order to choose a point among all the removable and insertable points, we use the following geometric information based on the distance map.

4.1 Priority distance for pixel choices

Let $d(x, \mathcal{A})$ be the Euclidean distance between a point $x \in \mathbb{Z}^2$ and $\mathcal{A}$. Then we define the relative distance for $x \in \mathcal{A} \setminus \mathcal{B}$ from $\mathcal{A}$ to $\mathcal{B}$ such that $d_{\mathcal{A}}^{\mathcal{B}}(x) =$

Function 3: UpdateRemovable

input : removable point set $\mathcal{Q}_\ominus$, Lyndon factorization $\mathcal{L}$, old and new factor lists I_{old} and I_{new}, priority d, digital set $\mathcal{C}$
output : updated removable point set $\mathcal{Q}_\ominus$

$i \leftarrow \mathrm{pop}(I_{old})$, $j \leftarrow \mathrm{pop}(I_{new})$;
replace the Lyndon point of $\mathcal{L}[i]$ by that of $\mathcal{L}[j]$ in $\mathcal{Q}_\ominus$;
while $I_{old} \neq \varnothing$ **do** $i \leftarrow \mathrm{pop}(I_{old})$, remove the Lyndon point of $\mathcal{L}[i]$ from $\mathcal{Q}_\ominus$;
while $I_{new} \neq \varnothing$ **do**
  $i \leftarrow \mathrm{pop}(I_{new})$, $x \leftarrow$ the Lyndon point of $\mathcal{L}[i]$;
  if x *is simple with respect to* $\mathcal{C}$ **then**
    push x in $\mathcal{Q}_\ominus$;
    foreach $x' \in \mathcal{N}_8(x) \setminus \{x\}$ **do**
      if $\exists j \in \mathcal{Q}_\ominus, x'$ *is the Lyndon point of* $\mathcal{L}[j]$ *and not simple to* $\mathcal{C}$ **then**
        remove x' from $\mathcal{Q}_\ominus$
      end
    end
end
return $\mathcal{Q}_\ominus$

$\frac{d(x,\mathcal{B})}{d(x,\mathcal{B})+d(x,\overline{\mathcal{A}})}$. We can observe that $d_{\mathcal{A}}^{\mathcal{B}}(x)$ is close to 0 when x is close to $\mathcal{B}$, $d_{\mathcal{A}}^{\mathcal{B}}(x)$ is close to 1 when x is close to $\overline{\mathcal{A}}$, and all the distances are between 0 and 1. Note that discrete points in $\mathcal{A} \setminus \mathcal{B}$ will be removed during the deformation while those in $\mathcal{B} \setminus \mathcal{A}$ will be added. For the points in $\mathcal{B} \setminus \mathcal{A}$, we use $d_{\mathcal{B}}^{\mathcal{A}}$.

4.2 Deflation algorithm

Let us first consider the easiest case such that $\mathcal{A} \supset \mathcal{B}$. Let $\mathcal{L}$ be the Lyndon factorization of the boundary word of $\mathcal{A}$. During deflation, $\mathcal{L}$ is updated for each step of removing a point, which is chosen by the priority $d_{\mathcal{A}}^{\mathcal{B}}$. The priority $d_{\mathcal{A}}^{\mathcal{B}}$ is in descending order; the highest priority is given to pixels of largest $d_{\mathcal{A}}^{\mathcal{B}}$. The following data structures are used in the deflation algorithm:

- $\mathcal{L}$: Lyndon factorization of a boundary word whose i-th element is (ℓ_i, n_i, p_i); ℓ_i is the Lyndon factor, n_i is the multiplicity, and p_i is the (left) Lyndon point,
- $\mathcal{Q}_\ominus$: set of removable points represented by Lyndon factor indices i,
- $\mathcal{T}_\ominus$: sequence of removed points.

In the following, N represents the length of the boundary word of $\mathcal{A}$ (or $\mathcal{B}$) so that the number of the Lyndon points of the Lyndon factorization $\mathcal{L}$ is bounded by $\mathcal{O}(N^{\frac{2}{3}})$ according to Property 1.

Algorithm 1 shows the procedure of deflation from $\mathcal{A}$ to $\mathcal{B}$, which call the two functions, UpdateLyndonFactorizationDueToSwitch (Function 2) and UpdateRemovable (Function 3). All the information of Lyndon factorization is stored in $\mathcal{L}$. The kernel of the algorithm is updating $\mathcal{L}$ efficiently for each removal step,

Algorithm 4: Inflation

input : digitally convex 4-connected sets $\mathcal{A}, \mathcal{B}$ such that $\mathcal{A} \subset \mathcal{B}$
output : a sequence $\mathcal{T}_{\oplus}$ of points to add

1 $w \leftarrow$ the boundary word of $\mathcal{A}$, $\mathcal{L} \leftarrow$ Lyndon Factorization of w;
2 calculate $d_{\mathcal{B}}^{\mathcal{A}}(x)$ for all $x \in \mathcal{B} \setminus \mathcal{A}$;
3 set the current inflated set $\mathcal{C} \leftarrow \mathcal{A}$, $\mathcal{T}_{\oplus} \leftarrow \varnothing$;
4 $\mathcal{Q}_{\oplus} \leftarrow$ AddInsertable$(\mathcal{Q}_{\oplus}, \mathcal{L}, (1, \ldots, |\mathcal{L}|), d_{\mathcal{B}}^{\mathcal{A}})$;
5 **while** $\mathcal{Q}_{\oplus} \neq \varnothing$ **do**
6 $\quad (i, j, left, right) \leftarrow$ find_max$(\mathcal{Q}_{\oplus})$;
7 $\quad (\ell, n, p) \leftarrow \mathcal{L}[i]$;
8 $\quad (\ell^{+}, \ell^{-}) \leftarrow split(\ell)$;
9 $\quad x \leftarrow$ the closed upper point of the j-th ℓ, push x to $\mathcal{T}_{\oplus}$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{x\}$;
10 $\quad I_{old} \leftarrow (i - left, i - left + 1, \ldots, i + right)$;
11 $\quad$ remove $\mathcal{L}[i - left, \ldots, i + right]$;
12 $\quad$ insert $\ell_{i-left}^{n_{i-left}} \ldots \ell_{i-1}^{n_{i-1}} \ell^{-}$ at $\mathcal{L}[i - left]$;
13 $\quad$ insert $\ell^{+} \ell_{i+1}^{n_{i+1}} \ldots \ell_{i+right}^{n_{i+right}}$ at $\mathcal{L}[i - left + 1]$;
14 $\quad I_{new} \leftarrow (i - left, i - left + 1)$;
15 $\quad \mathcal{Q}_{\oplus} \leftarrow$ DelInsertable$(\mathcal{Q}_{\oplus}, I_{old})$;
16 $\quad \mathcal{Q}_{\oplus} \leftarrow$ AddInsertable$(\mathcal{Q}_{\oplus}, \mathcal{L}, I_{new}, d_{\mathcal{B}}^{\mathcal{A}})$
17 **end**
18 **return** $\mathcal{T}_{\oplus}$

which is described in Function 2: only the two Lyndon factors adjacent to a chosen removable point are modified by the Lyndon factorization after the switch operation. In other words, we can observe that the update is made locally. As the length of each Lyndon factor is $\mathcal{O}(N)$ in worst case, the time complexity of Function 2 is $\mathcal{O}(N)$. Finding the maximum element of $\mathcal{Q}_{\ominus}$ (Line 6) and its update (Line 11) need $\mathcal{O}(\log N)$ for each removal step, if we store the sorted removable points of $\mathcal{Q}_{\ominus}$ in a tree structure such as a heap [9], as the size of $\mathcal{Q}_{\ominus}$ is bounded by $\mathcal{O}(N^{\frac{2}{3}})$, which is the same size of $\mathcal{L}$. Note that simplicity can be verified efficiently by using its local characterization [16] (see Function 3). Thus, the overall complexity of each deflation step of Algorithm 1 is $\mathcal{O}(N)$.

4.3 Inflation algorithm

Let us consider the case such that $\mathcal{A} \subset \mathcal{B}$. Here we add points one-by-one to $\mathcal{A}$ until obtaining $\mathcal{B}$ with the priority $d_{\mathcal{B}}^{\mathcal{A}}$. The inflation algorithm requires the following data structures with the Lyndon factorization $\mathcal{L}$ presented for the deflation algorithm.

- $\mathcal{Q}_{\oplus}$: set of insertable points, each of which is represented by a pair of a Lyndon factor and a multiplicity index (i, j), and their propagation ranges for left and right, $(left, right)$
- $\mathcal{T}_{\oplus}$: sequence of inserted points.

Function 5: AddInsertable

input : insertable point set $\mathcal{Q}_\oplus$, Lyndon factorization $\mathcal{L}$, indext set I to verify
output : updated $\mathcal{Q}_\oplus$

while $I \neq \varnothing$ **do**
  $i \leftarrow \mathrm{pop}(I)$, $(\ell, n, p) \leftarrow \mathcal{L}[i]$;
  foreach $j = 1, \ldots, n$ **do**
    $(insertable^-, k^-) \leftarrow InsertableLeft(i, j, \mathcal{L})$;
    $(insertable^+, k^+) \leftarrow InsertableRight(i, j, \mathcal{L})$;
    if $insertable^- \wedge insertable^+$ **then** push (i, j, k^-, k^+) to $\mathcal{Q}_\oplus$;
  end
end
return $\mathcal{Q}_\oplus$

Function 6: DelInsertable

input : insertable point set $\mathcal{Q}_\oplus$, Lyndon factorization $\mathcal{L}$, index set I to delete
output : updated $\mathcal{Q}_\oplus$

while $I \neq \varnothing$ **do**
  $i \leftarrow \mathrm{pop}(I)$, $(\ell, n, p) \leftarrow \mathcal{L}[i]$;
  foreach $j = 1, \ldots, n$ **do**
    remove the element associated to the point index (i, j) from $\mathcal{Q}_\oplus$
  end
end
return $\mathcal{Q}_\oplus$

Note that any insertable point is a closest upper point (Proposition 4), which exists uniquely for each j-th Lyndon factor ℓ_i.Thus, keeping $\mathcal{L}$ in the same way as the deflation is also enough for the inflation. When we add a point in the boundary of a digital convex set, the simplicity is obviously satisfied; no simplicity verification is necessary.

Algorithm 4 shows the inflation procedure from $\mathcal{A}$ to $\mathcal{B}$ guided by the priority $d^{\mathcal{A}}_{\mathcal{B}}$. Similarly to the deflation, the kernel of the inflation algorithm is also updating $\mathcal{L}$ efficiently for each point insertion. However, this update may affect $left$ and $right$ neighbors in the left and right propagations where $left, right$ can be more than 1, contrary to the deflation case, in which $left = right = 1$. Instead, those affected neighboring Lyndon factors are always replaced by exactly two Lyndon factors (see Lines 12 and 13 in Algorithm 4). In other words, no Lyndon re-factorization is needed for the insertion. These left and right neighboring ranges, $left$ and $right$, are respectively calculated in the functions, InsertableLeft and InsertableRight, both of which are called in Function 5 (see Function 7 for the InsertableLeft; InsertableRight is omitted here due to its similarity). In fact, those functions verify the instability of the point corresponding to the given factor and multiplicity indecencies, i and j, in left and right sides with the propagation verification. This part is based on Proposition 6. As this

Function 7: InsertableLeft

input : insertion factor index i, multiplicity index j, Lyndon factorization $\mathcal{L}$
output : boolean *insertable*, left propagation range r
$(\ell, n, p) \leftarrow \mathcal{L}[i]$, $(\ell^-, \ell^+) \leftarrow$ standard factorization of ℓ;
$w \leftarrow \ell^{j-1}\ell^+$;
$propag \leftarrow true$, $insertable \leftarrow false$, $r \leftarrow 0$;
while $propag = true$ **do**
  $(\ell_{prev}, n_{prev}, p_{prev}) \leftarrow \mathcal{L}[i - r - 1]$;
  if $\ell > w$ **then** $insertable \leftarrow true$, $propag \leftarrow false$;
  else if $\ell_{prev} = w$ **then** $insertable \leftarrow true$, $propag \leftarrow false$, $r \leftarrow r + 1$;
  else if $\exists n \in \mathbb{Z}^+, \ell_{prev} = \ell w^n$ **then** $w \leftarrow \ell_{prev}^{n_{prev}} w$, $r \leftarrow r + 1$;
  else $propag \leftarrow false$;
end
return $(insertable, r)$

Iteration: 0 Iteration: 500 Iteration: 1000 Iteration: 1544

Fig. 6. Deformation from a digitized disk to a digitized ellipse, both of which are digitally convex.

propagation cannot be theoretically bounded, the complexity of Function 5 is in $\mathcal{O}(N)$. In other words, if there is no propagation, this complexity can be reduced to $\mathcal{O}(\log N)$. This can be done if we strengthen the insertability condition such that $\ell_{i-1} \geq \ell_i^{j-1}\ell_i^+$ and $\ell_i^-\ell_i^{n_j-1} \geq \ell_{i+1}$ instead of those of Proposition 6.

Note that the size of $\mathcal{Q}_\oplus$ is almost equal to the number of furthest points, which can be given by $\sum_i m_i$ where m_i is the multiplicity for the i-th factor ℓ_i of the Lyndon factorization of the boundary word of the current deformed shape. If we set $M = \max_i m_i$, then we can also say that the size of $\mathcal{Q}_\oplus$ is in $\mathcal{O}(MN^{\frac{2}{3}})$. Thus the time complexity of updating $\mathcal{Q}_\oplus$ (Functions 6 and 5) are in $\mathcal{O}(\log N)$ as $M \leq N$. Then the overall complexity of each inflation step of Algorithm 4 is $\mathcal{O}(N)$ due to the propagation in the insertability verification. We remind that Lyndon re-factorization is not necessary for the inflation case, so that this $\mathcal{O}(N)$ comes only from the insertability verification propagation.

4.4 General deformation algorithm

Now let us consider more general case such that $\mathcal{A} \cap \mathcal{B} \neq \varnothing$. We start from an initial digitally convex set $\mathcal{A}$ and obtain $\mathcal{B}$ by adding points of $\mathcal{B} \setminus \mathcal{A}$ and removing points of $\mathcal{A} \setminus \mathcal{B}$. The algorithm is given by Algorithm 8, which is a

Algorithm 8: Digital convexity preserving deformation

input : overlapped digitally convex 4-connected sets $\mathcal{A}$, $\mathcal{B}$
output : a sequence $\mathcal{T}$ of pixels to remove (with $-$) and to add (with $+$)

$w \leftarrow$ the boundary word of $\mathcal{A}$, $\mathcal{L} \leftarrow$ Lyndon Factorization of w;
calculate $d^{\mathcal{B}}_{\mathcal{A}}(x)$ for $x \in \mathcal{A} \setminus \mathcal{B}$ and $d^{\mathcal{A}}_{\mathcal{B}}(x)$ for all $x \in \mathcal{B} \setminus \mathcal{A}$;
set the current deflated set $\mathcal{C} \leftarrow \mathcal{A}$, $\mathcal{T} \leftarrow \varnothing$;
$\mathcal{Q}_{\ominus} \leftarrow \text{UpdateRemovable}(\mathcal{Q}_{\ominus}, \mathcal{L}, \varnothing, (1, \dots, |\mathcal{L}|), d^{\mathcal{B}}_{\mathcal{A}}, \mathcal{C})$;
$\mathcal{Q}_{\oplus} \leftarrow \text{AddInsertable}(\mathcal{Q}_{\oplus}, \mathcal{L}, (1, \dots, |\mathcal{L}|), d^{\mathcal{A}}_{\mathcal{B}})$;
while $\mathcal{Q}_{\ominus} \cup \mathcal{Q}_{\oplus} \neq \varnothing$ **do**
  $x \leftarrow$ the Lyndon point corresponding to find_max($\mathcal{Q}_{\ominus}$);
  $y \leftarrow$ the closest upper point corresponding to find_max($\mathcal{Q}_{\oplus}$);
  if $d^{\mathcal{B}}_{\mathcal{A}}(x) \leq d^{\mathcal{A}}_{\mathcal{B}}(y)$ **then**
    push $(x, -)$ to $\mathcal{T}$;
    ... `// Deflation (Lines 6-10 of Algorithm 1)`
  else
    push $(y, +)$ to $\mathcal{T}$;
    ... `// Inflation (Lines 6-14 of Algorithm 4)`
  end
  $\mathcal{Q}_{\ominus} \leftarrow \text{UpdateRemovable}(\mathcal{Q}_{\ominus}, \mathcal{L}, I_{old}, I_{new}, d^{\mathcal{B}}_{\mathcal{A}}, \mathcal{C})$;
  $\mathcal{Q}_{\oplus} \leftarrow \text{DelInsertable}(\mathcal{Q}_{\oplus}, I_{old})$, $\mathcal{Q}_{\oplus} \leftarrow \text{AddInsertable}(\mathcal{Q}_{\oplus}, \mathcal{L}, I_{new}, d^{\mathcal{A}}_{\mathcal{B}})$;
end
return $\mathcal{T}$

simple fusion of Algorithms 1 and 4. Figure 6 shows an experimental result for a deformation from a digitized disk of 2821 points (most left) to a digitized ellipse (most right).

5 Conclusion

In this article, using the combinatorics on words to identify digital convex sets via their boundary words, we proposed algorithms for pixelwise inflation, deflation and more general deformation of digital convex sets preserving their convexity. Given a pair of digital convex sets, we showed that each proposed algorithm creates a sequence of digital convex sets, namely a deformation between them. The worst-case time complexity for each inflation and deflation iteration step was analyzed: $\mathcal{O}(N)$ for both where N is the length of the boundary word of a given digital convex.

References

1. Acketa, D.M., Žunić, J.: On the Maximal Number of Edges of Convex Digital Polygons Included into an m x m-Grid. Journal of Combinatorial Theory **Series A 69**, 358–368 (1995)

2. Alexander, J.C., Thaler, A.I.: The Boundary Count of Digital Pictures. Journal of the ACM **18**(1), 105–112 (Jan 1971)
3. Andrews, G.: A Lower Bound for the Volume of Strictly Convex Bodies with Many Boundary Lattice Points. Transactions of the American Mathematical Society **106**, 270–279 (1963)
4. Berstel, J.: Tracé de droites, fractions continues et morphismes itérés. In: Mots, pp. 298–309. Hermès (1990)
5. Borel, J.P., Laubie, F.: Quelques mots sur la droite projective réelle. Journal de Théorie des Nombres de Bordeaux **5**(1), 23–51 (1993)
6. Borel, J.P., Laubie, F.: Construction de mots de Christoffel. Comptes Rendus de l'Académie des Sciences. Série I. Mathématique **313**(8), 483–485 (1991)
7. Brlek, S., Lachaud, J.O., Provençal, X., Reutenauer, C.: Lyndon + christoffel = digitally convex. Pattern Recognition **42**(10), 2239–2246 (2009)
8. Chen, K.T., Fox, R.H., Lyndon, R.C.: Free differential calculus, iv. the quotient groups of the lower central series. Annals of Mathematics **68**(1), 81–95 (1958)
9. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to algorithms. MIT Press, 2nd ed edn. (2001)
10. Crombez, L.: Digital convex + unimodular mapping = 8-connected (all points but one 4-connected). In: Lindblad, J., Malmberg, F., Sladoje, N. (eds.) Discrete Geometry and Mathematical Morphology. pp. 164–176. Lecture Notes in Computer Science, Springer International Publishing
11. Dulio, P., Frosini, A., Rinaldi, S., Tarsissi, L., Vuillon, L.: First steps in the algorithmic reconstruction of digital convex sets. In: International Conference on Combinatorics on Words. pp. 164–176. Springer (2017)
12. Duval, J.P.: Factorizing words over an ordered alphabet. Journal of Algorithms **4**(4), 363–381 (1983)
13. Eckhardt, U.: Digital Lines and Digital Convexity. In: Digital and Image Geometry, vol. 2243, pp. 209–228. Springer (2001)
14. Freeman, H.: On the Encoding of Arbitrary Geometric Configurations. IRE Transactions on Electronic Computers **EC-10**(2), 260–268 (Jun 1961)
15. Kim, C.E.: On the Cellular Convexity of Complexes. IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-3**(6), 617–625 (1981)
16. Kong, T.Y., Rosenfeld, A.: Digital topology: Introduction and survey. Computer Vision, Graphics, and Image Processing **48**(3), 357–393 (Dec 1989)
17. Lachaud, J.O.: An Alternative Definition for Digital Convexity. Journal of Mathematical Imaging and Vision (Apr 2022)
18. Lothaire, M.: Algebraic combinatorics on words, Encyclopedia of Mathematics and its Applications, vol. 90. Cambridge University Press, Cambridge (2002)
19. Lyndon, R.C.: On burnside's problem. Transactions of the American Mathematical Society **77**(2), 202–215 (1954)
20. Roussillon, T.: An Arithmetical Characterization of the Convex Hull of Digital Straight Segments. In: DGCI 2014, LNCS, vol. 8668, pp. 150–161. Springer (2014)
21. Tarsissi, L., Coeurjolly, D., Kenmochi, Y., Romon, P.: Convexity Preserving Contraction of Digital Sets. In: ACPR 2019. LNCS, vol. 12047, pp. 611–624. Springer (2020)
22. Tarsissi, L., Kenmochi, Y., Romon, P., Coeurjolly, D., Borel, J.P.: Convexity preserving deformations of digital sets: characterization of removable and insertable points. Tech. rep., LIGM (2022)