

Introduction to Discrete Soft Transforms

Bastien Laboureix, Eric Andres, Isabelle Debled-Rennesson

▶ **To cite this version:**

Bastien Laboureix, Eric Andres, Isabelle Debled-Rennesson. Introduction to Discrete Soft Transforms. 2nd International Joint Conference on Discrete Geometry and Mathematical Morphology (DGMM 2022), Etienne Le Quentrec, Oct 2022, Strasbourg, France. pp.422-435, 10.1007/978-3-031-19897-7_33 . hal-03982600

HAL Id: hal-03982600
https://hal.science/hal-03982600

Submitted on 10 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Introduction to Discrete Soft Transforms

Bastien Laboureix[1], Eric Andres[2], and Isabelle Debled-Rennesson[3]

[1] Université de Lorraine, CNRS, LORIA, 54000 Nancy, FRANCE
bastien.laboureix@loria.fr
[2] Université de Poitiers, XLIM UMR CNRS 7252, 86962 Chasseneuil-Futuroscope.
eric.andres@univ-poitiers.fr
[3] Université de Lorraine, CNRS, LORIA, 54000 Nancy, FRANCE
isabelle.debled-rennesson@loria.fr

Abstract. In this paper a new class of discrete transforms of discrete straight segments (DSS), called *Discrete Soft Transforms* is introduced. The soft transformation of a segment consists in moving a single discrete point at each step while keeping the segment property. We propose the soft rotation and soft translation of a segment and extend these results to the soft translation of a tree.

Keywords: discrete straight segment, discrete tree, soft translation, soft rotation, discrete soft transform

1 Introduction

Commonly transformations in the discrete domain come in the form of discretized continuous transformations. The concern with such an approach is that it is difficult to obtain natural properties such as bijectivity or preservation geometric features or topology. An alternative approach promoted by the discrete geometry community is to consider transformations directly in the discrete domain. This led to interesting results on bijectivity or topology preservation for discrete rigid motions [1,6]. Such results are difficult to obtain and even more difficult to extend because, as global transformations on the whole domain, local properties are not easy to guarantee. In this paper we propose to introduce a new class of discrete transforms called *discrete soft transform*. The idea is to decompose a discrete transform into a sequence of atomic steps where only one discrete point at a time is moved/added/suppressed while maintaining a set of given properties. The recently, proposed morphing method by Lama Tarsissi and al. [9] that preserves the convexity of discrete objects by adding/removing one point at each step can be seen as an example of discrete soft transform.

As a proof of concept, we propose a discrete soft rotation and translation of discrete straight segments (DSS). The discrete lines have interesting arithmetical and combinatoric properties studied for a long time (see [3] for an historical review). The soft transformation of a segment consists in moving a single discrete point at each step while keeping the segment property. We present shortly the soft rotation of a DSS, but the focus of the paper is on the soft translation.

The soft translation can be seen as akin to a subpixel translation method when compared to the discretized continuous translation. After proving a fundamental result on the movable points in a DSS, we propose an algorithm for the soft translation of a DSS and the soft translation of tree embeddings where vertices are embedded as discrete points and edges as DSS. The next step will be to propose a soft translation for graphs of DSS and thus to the soft translation of segmented images.

This paper is decomposed into five parts, the first being this introduction. In section two, we recall basic notions about discrete lines and segments. Then, in the third section, the discrete soft tranform on a DSS is introduced and a fundamental result about movable points in a DSS is proved. In section four, we propose a first simple algorithm for soft rotations of a DSS before focusing on the soft translation of a DSS. There are in particular an important distinction that has to be made between the soft translation of a DSS of slope between 0 and 1 and a DSS of slopes greater or equal to 1. An extension to a set of DSS that embed a tree concludes section four. In the last section, we conclude and present perspectives for this work.

2 Preliminaries

Let $\{e_1, e_2\}$ denote the canonical basis of the 2-dimensional Euclidean vector space. In this paper we are dealing with discrete points in $\mathbb{Z}^2$. Two points $p, q \in \mathbb{Z}^2$ are said to be 4-neighbours iff $\|p-q\|_1 = 1$, and said to be 8-neighbours iff $\|p-q\|_\infty = 1$. A Digital Straight Line (DSL for short) $\mathcal{D}(a,b,\mu)$ of integer characteristics (a,b,μ) is the set of digital points $(x,y) \in \mathbb{Z}^2$ such that $0 \leqslant ax - by + \mu < \omega$ where $\omega = \max(|a|,|b|)$ and $\gcd(a,b) = 1$. These DSL are 8-connected and called naive DSL [8]. The slope of the DSL is the fraction $\frac{a}{b}$ (when $b \neq 0$). The value μ is sometimes called the translation constant or the offset. The value $R(a,b,\mu)(x,y) = ax - by + \mu$ is called the remainder of the DSL. A DSL can also be defined as the integer points of a strip delimited by the lower leaning line $ax - by + \mu = \omega - 1$ and the upper leaning line $ax - by + \mu = 0$ [2]. Upper (resp. Lower) leaning points are the digital points of the DSL lying on the upper (resp. lower) leaning lines. A weakly exterior point is a point of a DSL that verifies $ax - by + \mu = -1$ (in this case we speak also of a weakly upper exterior point) or $ax - by + \mu = \omega$ (in this case we speak also of a weakly lower exterior point) [2]. We note $Inf(S)$ (resp. $Sup(S)$) the set of lower (resp. Upper) leaning points of S.

A digital straight line segment (DSS) $\mathcal{D}(a,b,\mu,E_0,E_1)$ is a finite 8-connected subset of the DSL $\mathcal{D}(a,b,\mu)$ with the end-points E_0 and E_1. We speak of horizontal (resp. vertical) segments, segments in the first octant (resp. second octant) with slope $0 \leqslant \frac{a}{b} < 1$ (resp. $\frac{a}{b} \geq 1$). For an horizontal line S, for a given x, there is one and only one $y = \lfloor \frac{ax+\mu}{b} \rfloor$ such that $(x,y) \in S$. Equivalently, in vertical lines, for a given y there is only one x.

For a given DSS $S = \mathcal{D}(a,b,\mu,E_0,E_1)$, there exists an infinite number of parameters (a',b',μ') such that $S = \mathcal{D}(a',b',\mu',S_0,S_1)$.

For instance, $\mathcal{D}(5,8,0,(0,0),(11,6)) = \mathcal{D}(8,13,1,(0,0),(11,6))$. There exists however only one set of parameters with minimal b, called *minimal parameters* [2].

3 Soft Transform

Our goal is to apply transforms to discrete structures by maintaining geometric properties and a form of *continuity*. If continuity has a topological definition for classical geometric structures, we must define a notion of *continuous transform* in the discrete case, which we call *soft transform*.

Definition 1. *Two DSS S and S' are said to be neighbours iff they can be described as $X \cup \{p\}$ and $X \cup \{q\}$ where X is a set of points and p, q are 4-neighbours.*
We note this relation as $S \leftrightarrow S'$.

Definition 2 (DSS Soft Transformation)**.** *Considering a function $f : \mathbb{Z}^2 \to \mathbb{Z}^2$, and DSS S, a soft transformation of S into $f(S)$ is a sequence of segments $S_0 ... S_N$ such that:*

- $S_0 = S$
- $S_N = f(S)$
- $\forall i \in [\![0, N-1]\!], S_i \leftrightarrow S_{i+1}$

For $S \leftrightarrow S'$, the transform from one to the other DSS is called *atomic soft transform*.

This definition can be extended to a set of DSS or more generally to a digital shape with some caveats: the soft transform of a set of segments can of course be handled as the independent separate soft transform of each segment, but in general, this is not what is expected. Segments may share vertices, and thus, for what follows, we consider graphs formed by digital straight segments and add an additional constraint which is the preservation of the overall structure of the graph. As for a digital shape, a shape can always be decomposed into a set of DSS, but this decomposition is not unique, and therefore the soft transform as applied on DSS graphs depends on this decomposition.
All our algorithms are implented. See for a couple of examples that go a beyond the present paper : https://imgur.com/a/j81nI6f

3.1 Movable Points in a DSS

The goal of this paper is to define a notion of soft translation on a DSS, and introduce a first notion of soft rotation. The first question that is answered in this first subsection is the question of which points of a DSS can be moved while remaining a DSS:

Theorem 3 (Movable points in a DSS)**.** *In a DSS S with minimal parameters of slope between 0 and 1, the points which can move up (resp. down) while keeping the segment property are exactly the extremal lower (resp. upper) leaning points. The number of these points can be one or two.*

By extremal, we mean the leaning points closests to the end-points of the DSS.

Proof. Let us concentrate for this proof on points that can move up while preserving the DSS property: the goal is to characterize all the points p such that $S \cup \{p + e_2\} \setminus \{p\}$ is still a DSS. The proof is similar for points that move down.

Let us consider a segment S with end-points E and E' and of analytical equation $0 \leqslant ax - by + \mu < b$ (with a, b, μ minimal).
Let $I \stackrel{\text{def}}{=} (x_I, y_I)$ be a point of S. We denote $S' \stackrel{\text{def}}{=} S \cup \{I + e_2\} \setminus \{I\}$.
Let us show S' is a DSS iff I is a lower extremal leaning point. The proof is similar for upper leaning points.

$\Rightarrow$: Let us reason by contradiction: Two cases are possible.

Case 1 : Let us suppose that I is not a lower leaning point. This means that $R(a, b, \mu)(I) \leqslant b - 2$. So $R(a, b, \mu)(I + e_2) \leqslant -2$. Therefore $I + e_2$ is strongly exterior to S and S' is not a DSS [2].

Case 2 : Let us now suppose that I is indeed a lower leaning point, but not extremal. Let us then call A and B the extremal lower leaning points of S.
This means that I is a point of the continuous segment $[AB]$. Since $A, B \in S'$, by convexity, $I \in S'$ which is not.

$\Leftarrow$: Let us illustrate the proof of theorem 3 with Figure 1. In Figure 1, the segment S is represented in blue. Let us extend the segment S upto an upper leaning point $P \stackrel{\text{def}}{=} (x_P, y_P)$ such that $\|P - I\|_\infty \geqslant b$ (green part of the segment): we have then a cover-segment of S denoted S_0.
Since P is an upper leaning point and an end-point of S_0, by [2], $P - e_2$ is slightly exterior to S_0, therefore $S'_0 \stackrel{\text{def}}{=} S_0 \cup \{P - e_2\} \setminus \{P\}$ is a DSS.
Moreover, S'_0 admits as lower extremal leaning points I and $P - e_2$.
Let us denote a', b', μ' the parameters of S'_0, and Π (resp. Π') the restriction of S_0 (resp. S'_0) to the abscissa interval $[x_P, x_I - 1]$.
This way Π (in green) and Π' (in magenta) form respectively a period of S_0 and of S'_0 because their lengths are b'.
One can extend Π' upto E' (in red) by periodicity (since $\|P - I\|_\infty \geqslant b$), in a segment containing I and who is a cover-segment of S.
In the same way, one can extend Π upto E' (in yellow) in a segment containing $I + e_2$ and who is a cover segment of S'. Therefore S' is a segment. □

3.2 Computing the new parameters of a DSS

In [2], I. Debled-Rennesson explained how to compute the minimal parameters of a DSS when adding a new point at the end of the DSS. In our case, the

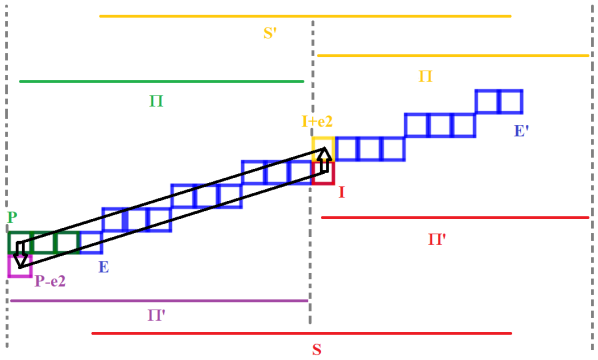


Fig. 1. Upward translation of a lower leaning point I

problem is slightly different since we move one point inside a DSS. Let us consider a DSS S with parameters (a, b, μ), of slope smaller than 1 defined on the interval $x \in [\![x_0, x_1]\!]$ and let us suppose that $m\,(x_m, y_m) \in S$ is the lower leaning point closest to x_1 (the rightmost movable point), with S' of parameters (a', b', μ') the resulting DSS after the move m to $m + e_2$. M is defined as the upper leaning point in S closest to x_1: $x_m = a^{-1}(b - 1 - \mu) + b\lfloor \frac{x_1 - a^{-1}(b-1-\mu)}{b} \rfloor$ and $y_m = \lfloor \frac{ax_m+\mu}{b} \rfloor$. M is a lower leaning point of S and $M + e_2$ is a upper leaning point in the new DSS. Let us consider the upper leaning point $u\,(x_u, y_u)$ that lies before with $x_u = -a^{-1}\mu + \lfloor \frac{x_0 - 1 + a^{-1}\mu}{b} \rfloor$ and $y_u = \lfloor \frac{ax_u+\mu}{b} \rfloor$. The parameters of the new DSS S' are given by $(b', a') = m + e_2 - u$ and $\mu = -a'x_u + b'y_u$. Now, nothing guarantees that these parameters are minimal: this means that if we want to iterate, we can use the results proposed in [5,7] to determine the minimal parameters of the new DSS, which can be made in time $O(\ln(n))$, where n is the length of the segment.

3.3 A simple exemple: the soft rotation of a DSS

In this section, we present an exemple of soft transform of a DSS that will be proved useful in future works : the soft rotation of a DSS. We consider only rotation in the first octant, the others can be obtained by symmetry. Note that our notion of rotation does not change the number of points but changes the length of the continuous segment (from n to $n\sqrt{2}$).

Problem 4. *Considering an integer $n \in \mathbb{N}$, how can we soft transform the flat segment $S_0 \overset{def}{=} \{(k, 0) \mid k \in [\![0, n-1]\!]\}$ into the diagonal segment $S_N \overset{def}{=} \{(k, k) \mid k \in [\![0, n-1]\!]\}$?*

The reverse rotation can be easily obtained in a similar way. With the previous subsections, we have all we need to make this algorithm work and prove

Algorithm: Direct Soft Rotation

Input: the length n of the considered flat segment
Output: a sequence of moves from S_0 to S_N

$S \leftarrow S_0$;
while $S \neq S_N$ **do**
  $p \leftarrow$ lower leaning point with greatest abscissa in S ;
  replace p by $p + e_2$ in S ;
  compute the new parameters of the DSS ;
end

its correction. The overall complexity of an atomic soft rotation is $O(\ln(n))$. As there is a quadratic number of points to move, the complexity for the whole process is $O(n^2 \ln(n))$.

4 Soft Translations of discrete straight line segments

The goal is henceforth to deal with the soft translation of segments. Let us first note that we only consider here the upward translation by one position. Other translations can be obtained by accumulation and symmetries.

Problem 5. *Given a discrete Straight segement S, how to compute the soft translation from S to $S + e_2$?*

By translation and symmetry, we can limit our study to segments of the first quater with an end-point in $(0, 0)$. We need however to differentiate the cases of horizontal segments (of slope between 0 and 1) from vertical segments (of slope greater than 1).

4.1 Horizontal Segments

In this section we are considering horizontal segments, i.e. of slope smaller than 1. The points of the DSS S are (k, y_k) for $k \in [\![0, n-1]\!]$, where n is the length of the segment. We propose an algorithm allowing to translate this segment simply by shifting the offset value μ.

Definition 6 (Primitive of a DSS)**.** *A DSS S defined by $0 \leqslant ax - by + \mu < b$ is said to be primitive iff it contains exactly b points.*

Proposition 7. *Let S be a primitive DSS defined by $0 \leqslant ax - by + \mu < b$, then S admits a unique lower leaning point p and the segments $S \cup \{p + e_2\} \setminus \{p\}$ is defined by $0 \leqslant ax - by + \mu + 1 < b$.*

Proof. Since $\gcd(a, b) = 1$ and S is primitive, $x \mapsto ax + \mu$ is bijective over $\mathbb{Z}/b\mathbb{Z}$. Since S is primitive, the segment is of length b, therefore there exists a unique $p = (x, y) \in S$ such that $ax - by + \mu = b - 1$. The point p is a lower leaning point and all the points of the DSS $S \cup \{p + e_2\} \setminus \{p\}$ verify $0 \leqslant ax - by + \mu + 1 < \omega$. □

Proposition 8 (Cover Segment)**.** *For each DSS S of length n, there exists a* primitive cover segment *S' of length $\leqslant 3n$ such that $S \subset S'$. In addition, the parameters of S' can be effectively computed in $O(ln(n))$.*

Proof. Let us consider a DSS S defined by $0 \leqslant ax - by + \mu < b$ with $b < n$. We are looking for parameters a', b' relative primes such that $\frac{a'}{b'} - \frac{a}{b} = \frac{1}{bb'}$ with $b' \in [\![2n, 3n]\!]$.
Such a couple exists (a', b') since $\frac{a'}{b'} - \frac{a}{b} = \frac{a'b - ab'}{bb'}$.
We consider $b' \in [\![2n, 3n]\!]$ such that $ab' \equiv -1(b)$ (which exists since $\gcd(a, b) = 1$), and then a' such that $a'b - ab' = 1$.
And so, $\frac{a'}{b'} - \frac{a}{b} = \frac{1}{bb'}$.

For the offset, we consider μ' such that $\frac{\mu}{b} \leqslant \frac{\mu'}{b'} < \frac{\mu}{b} + \frac{1}{b'}$.
We call S' the DSS defined by $0 \leqslant a'x - b'y + \mu' < b'$ of length b'.
Let's consider $(x, y) \in S$. Then $y = \left\lfloor \frac{ax+\mu}{b} \right\rfloor$.
$\frac{a'x+\mu'}{b'} - \frac{ax+\mu}{b} = x\left(\frac{a'}{b'} - \frac{a}{b}\right) + \frac{\mu'}{b'} - \frac{\mu}{b}$.
This quantity is positive and inferior to $\frac{x}{bb'} + \frac{1}{b'} \leqslant \frac{x}{2bn} + \frac{1}{2n} < \frac{1}{2b} + \frac{1}{2b} = \frac{1}{b}$.
Therefore $y \leqslant \frac{ax+\mu}{b} \leqslant \frac{a'x+\mu'}{b'} < \frac{ax+\mu+1}{b} \leqslant y + 1$.
By definition of the integer part, we obtain $y = \left\lfloor \frac{a'x+\mu'}{b'} \right\rfloor$ therefore $(x, y) \in S'$. □

We can then apply the soft translation algorithm on the cover segment S' of a DSS S, which allows us to perform the soft translation of S.

Algorithm: Horizontal DSS Soft Translation
Input: a segment S
Output: a sequence of moves from S to $S + e_2$

```
n ← |S| ;
S' ← a primitive cover segment of S of length n' ⩽ 3n ;
a', b', μ' ← parameters of S' ;
Compute a'^{-1} in ℤ/b'ℤ ;
for j from 0 to n' − 1 do
    k ← −a'^{-1}(μ' + j + 1) in ℤ/b'ℤ ;
    if k < n then
        replace p of abscissa k by p + e2 in S ;
    end
end
```

Theorem 9. *The soft translation of the horizontal DSS S is correct and has a complexity of $O(n)$ with n the length of S.*

Proof. At step j, we consider $k = -a'^{-1}(\mu' + j + 1)$ in $\mathbb{Z}/b'\mathbb{Z}$.
Henceforth, $0 \leqslant a'k + \mu + j = -1$ in $\mathbb{Z}/b'\mathbb{Z}$. Therefore, if $k < n$, the point (k, y_k)

of S verifies $a'k - b'y_k + \mu + j = -1$ since $k < n' = b'$ because S' is primitive. The point of abscissa k can therefore be moved in S', and thus in S.
As for the complexity, the length of S' is linearly proportional to the length of S and constructing S' requires a simple computation in $\mathbb{Z}/b'\mathbb{Z}$ once. The loop at each iteration has a constant time and thus operates in $O(n') = O(n)$. □

The soft translation algorithm simply modifies the offset in the parameters. The slope of the segment is invariant during the translation.

4.2 Vertical Segments

In this section we are considering vertical segments, i.e. of slope greater or equal to 1. Let us note that we need here a new operation to handle the upward soft translation. For instance, in Figure 2, we have a DSS of length 5 defined by $x = 2$ and $1 \leqslant y \leqslant 5$. No point can move upward without superposition or deconnection. To handle such situations, we propose another approach with two new operations for the soft translation of segments:

- Add a point over the point of highest ordinate (Figure 2b.)
- Remove the point of lowest ordinate (Figure 2c.).

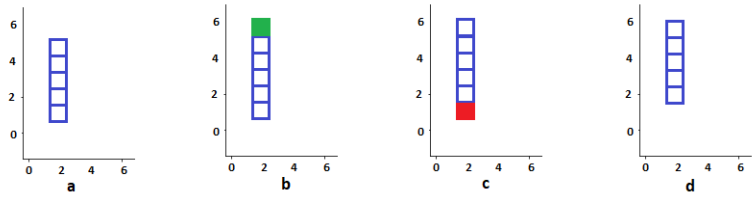


Fig. 2. A new operation for vertical segments

We propose a new algorithm that allows to create a soft translation of vertical segments in the general case by using these two operations.

Definition 10 (Pillar point)**.** *Let S be a segment of the second octant admitting a cover segment S' defined by the parameters $0 \leqslant a'x - b'y + \mu' < a'$.*
The point $(x, y) \in S$ is called a pillar of S iff $a' - b' \leqslant a'x - b'y + \mu' < a'$.

It is easy to see that a point (x, y) of a DSS is a pillar iff $(x - 1, y - 1) \in S$ (except for the lower end-point) since $a' - b' \leqslant a'x - b'y + \mu' < a'$ means that $0 \leqslant a'(x - 1) - b'(y - 1) + \mu' < b'$. The idea of the algorithm is to translate to the left the set of pillar points, by decreasing remainder, and to manage the end-points via the operations of addition and suppression: indeed, for a pillar (x, y), $(x - 1, y - 1) + e_2 = (x, y) - e_1$. There is a special case that needs to be considered when the upper end-point is also a pillar point. Figure 3 illustrates

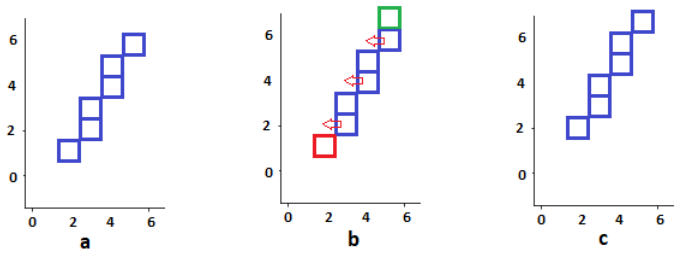


Fig. 3. Soft translation of a vertical segment

the general idea of the algorithm. The red discrete point is removed. The green discrete point is added and all the discrete points with arrows are pillars and they are moved to the left.
Note that our algorithm can change the number of points of the DSS when we add or remove a discrete point.

Theorem 11. *The soft translation algorithm of a vertical segment S is correct (in the sense that is indeed a discrete soft translation) and has a time complexity of $O(nln(n))$ with n the length of S.*

Proof. The segment at the end of the algorithm is $S + e_2$ because the only points that move, beyond the removal/addition of the end-points, are the pillars and the property of a pillar (x, y) is to take the place of the DSS point $(x - 1, y - 1) + e_2$ as we have already mentioned. A point (x, y) that is neither pillar nor an end-point means that $(x, y - 1)$ belongs to S, which means that $(x, y - 1) + e_2 = (x, y)$. What we need to show now is that the set of points remains a DSS when moving pillar points and handling end-points. There are three cases for a pillar point p in the algorithm:

- p is not an end-point: identical to the horizontal DSS soft translation with a 90° rotation. The set of points remains thus a DSS.
- p is the lower end-point: a DSS where you remove an end-point remains a DSS.
- p is the upper end-point: here, the proof is slightly less direct and is presented in what follows.

Let us consider that the next pillar p to treat is the upper end-point. Let us call S_1 the current DSS before moving p. The point p may be moved upwards at position $p + e_2$ or $p + e_1 + e_2$.
However p is a pillar and therefore it has to be moved to the left, with
$S_2 \stackrel{\text{def}}{=} S_1 \cup \{p - e_1\} \setminus \{p\}$ a DSS. This means $p + e_1 + e_2$ is not an option anymore, therefore S_2 can only be prolonged with $p + e_2$, and thus $S_1 \cup \{p + e_2\}$ is a DSS.
As for the complexity, it is easy to see that computing the pillar points is linear in the length of S and that ordering them in decreasing remainder order leads to the given complexity. □

Algorithm: Vertical DSS Soft Translation

Input: a segment S
Output: a sequence of moves from S to $S + e_2$

$S' \leftarrow$ a primitive cover segment of S of size $\leqslant 3n$;
$a', b', \mu' \leftarrow$ parameters of S' ;
$P \leftarrow$ set of the pillar points of S in decreasing order of remainder $a'x - b'y + \mu'$;
while *P is not empty* **do**
  $p \leftarrow$ first element of P ;
  if *p is not an end-point* **then**
    replace p by $p - e_1$ in S ;
    remove p from the set P
  end
  if *p is the lower end-point* **then**
    remove p from S;
    remove p from the set P
  end
  if *p is the upper end-point* **then**
    if *the lower end-point hasn't been removed yet* **then**
      remove the lower end-point from S ;
      remove the lower end-point from the set P if it is a pillar.
    end
    Add a point $p + e_2$ to S ;
    replace p by $p - e_1$ in S ;
    remove p from the set P
  end
end
Remove the lower end-point if it hasn't been done yet ;
Add a point over the upper end-point if it hasn't been done yet.

4.3 Soft Translation of DSS Trees

In imaging, discrete straight line segments come rarely alone, that's why we propose here a first extension in the form of a soft DSS tree translation algorithm. A finite DSS tree Γ can be seen as an embedding ρ of a finite tree (Σ, A) in $\mathbb{Z}^2$, where vertices of Σ become points in $\mathbb{Z}^2$ and edges of A DSSs. Note that a DSS tree does not necessarily have to be planar here.
The idea is to move each DSS of the tree independently, however, contrary to a single DSS, there are end-points to consider. An end-point can be shared by a number of segments, and in order to ensure that each set of points corresponding to a segment remains a DSS during the set of atomic transforms and that the global structure of the tree is preserved, a shared end-point can only move once. If we move the end-points independently for each segment, we may loose the DSS tree structure. This means that the points of a DSS S can move until we reach the shared end-point E of the DSS (in a tree there is one and only one if the tree is not composed of only one segment which we can suppose here). The

end-point E, and the remaining points of S, can only move when it is the turn of E to move in every other DSS S_i that shares E as end-point.

Algorithm: Soft DSS Tree Translation
Input: A DSS tree Γ
Output: A sequence of moves from Γ to $\Gamma + e_2$

$\Sigma, A, \rho \leftarrow$ Tree and embedding of Γ ;
while Γ *has not completely moved* **do**
  for $S \in \rho(A)$ *not totally translated* **do**
    $p \leftarrow$ next point of S to move ;
    if p *is not an end-point of* S **then**
      move p ;
    end
    else
      $V(p) \leftarrow \{S \in \rho(A) \mid p \text{ is an end-point of } S\}$;
      if $\forall S \in V(p)$, *the next point to move in* S *is* p **then**
        move p ;
      end
    end
  end
end

Theorem 12. *If Γ is a DSS tree with at least two vertices then the algorithm terminates, is correct and has a complexity* $O\left(\sum_{S \in \rho(A)} |S| \ln |S|\right)$.

The algorithm is correct for an empty tree but not for a tree with only one vertex, but then it means translating a structure with just one discrete point. In case of only one segment, we can use the soft DSS translation algorithms even if formally the tree algorithm still holds.

Proof. The partial correction of the algorithm follows from the correction of the soft DSS translation algorithms.
Let us prove the termination by recurrence on $|\Sigma|$: if the tree has only two vertices then Γ is a DSS and we use the DSS translation algorithm which ends. Let us suppose that $|\Sigma| \geqslant 3$. G admits a leaf s attached to G by and edge $a = \{s, s'\}$. We denote $q \stackrel{\text{def}}{=} \rho(s)$ and $q' = \rho(s')$. We denote $S \stackrel{\text{def}}{=} \rho(A)$ with end-points q and q'.
Let us assume that the algorithm does not end and let us consider the state in which it is blocked. Let p be the next point to move in the DSS S. Since p can not be moved upwards, p is necessarily an end-point of S. Moreover, $p \neq q$ because $V(q) = \{S\}$ therefore for all $S_0 \in V(q)$, p is the next point to move is S_0. Therefore $p = q'$. This proves that the algorithms does not end on the discrete tree induced by $\Sigma \setminus \{s\}$, which contradicts our recurrence hypothesis. □

An important property of this algorithm is that it preserves the slopes of the cover segments of the tree. Note that our algorithm can change the number of points in the tree when we add or remove points in vertical DSS.

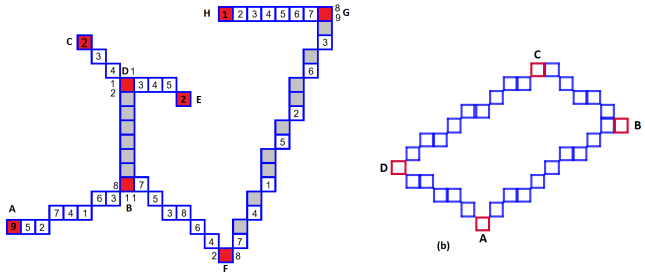


Fig. 4. (a) A DSS tree. The numbers represent the atomic movement order. The red discrete points are the vertices (that have multiple ordre movements, number next to the vertex). The discrete points in gray do not move. (b) A cyclic DSS graph that does not terminate with a naive approach

Figure 4a. illustrates the translation of a DSS tree. The numbers give the theoretical movement inside each DSS. The absolute movement order depends on the order with which you treat each DSS. If each DSS was treated independently, each discrete point with 1 would move, then those with value 2, etc. For trees, if the moving point is a end-point, the DSS is placed on hold until it is its turn to move for each of the other DSS sharing the end-point.
For example, we can start by translating the DSS $[BF]$. So we first consider the end-point B, however this point can't move because it has to wait its turn for segment $[AB]$. Meanwhile, other points in the segment $[AB]$ may move (because they are not end-points and are placed before in the order). After these moves (from 1 to 7 in $[AB]$), the next point to move in $[AB]$ is now the end-point B. Now, all the segments which share B as an end-point agree to move B so it can be done. We can continue the process of $[BF]$: the next point to move is F, which is an end-point so we have to wait that $[FG]$ agrees also to move F, etc. The theorem ensure us that the process finishes.

As we can see, there is no interlocking because there are no cycles in trees. It is different for DSS graphs. In Figure 4b. we see a polygon and here the proposed algorithm does not work: A moves before B, B before C, C before D and D before A. There is an interlocking. There is a way to solve this problem that we will present in a forthcoming paper.

5 Conclusion

In this paper we introduced Discrete Soft Transforms and showcase the soft rotation and soft translation of discrete straight segments. A discrete soft transform

as a set of atomic transforms where at each step only one discrete point is modified while preserving some properties. For a discrete straight segment, we show that at most two points that can be moved in a given direction to stay a segment. After a brief presentation of soft rotation of a segment, we focused on soft translation of segments which required distinct algorithm for different slopes. The results on the soft translation of a segment are extended to a discrete tree formed of discrete segments. Som examples of soft rotation and translation of DSS can be seen at the following repositery: https://imgur.com/a/j81nI6f. At the end of the paper, we show that there may be interlocking when we have cycles. A solution which involves a new method of soft translation will be presented in a forthcoming paper. This paper opens up many questions. A first direct question is how to adapt these methods to standard discrete line segments. This would be particularly interesting for the soft translation of segmented images by moving the discrete inter-pixel boundary between the regions [4]. The extension to the soft translation of 3D/nD planes or hyperplanes is another interesting next step. Finally, we could develop our work on soft rotations and imagine other type of transforms such as soft homothety or general continuous transformations.

References

1. E. Andres. The quasi-shear rotation. In *DCGI'96*, volume 1176 of *LNCS*, pages 307–314. Springer, 1996.
2. I. Debled-Rennesson and J-P. Reveilles. A linear algorithm for segmentation of digital curves. *IJPRAI*, 09(04):635–662, 1995.
3. R. Klette and A. Rosenfeld. Digital straightness - a review. *Discret. Appl. Math.*, 139(1-3):197–230, 2004.
4. V.A. Kovalevsky. Finite topology and image analysis. In Peter W. Hawkes, editor, *Image Mathematics and Image Processing*, volume 84 of *Advances in Electronics and Electron Physics*, pages 197–259. Academic Press, 1992.
5. J-O. Lachaud and M. Said. Two efficient algorithms for computing the characteristics of a subsegment of a digital straight line. *Discret. Appl. Math.*, 161(15):2293–2315, 2013.
6. P. Ngo, N. Passat, Y. Kenmochi, and I. Debled-Rennesson. Geometric preservation of 2d digital objects under rigid motions. *J. Math. Imaging Vis.*, 61(2):204–223, 2019.
7. J-S. D. Ouattara, E. Andres, G. Largeteau-Skapin, R.Zrour, and T. M.-Y. Tapsoba. Remainder approach for the computation of digital straight line subsegment characteristics. *Discret. Appl. Math.*, 183:90–101, 2015.
8. J-P. Reveillès. Calcul en nombres entiers et algorithmique. *Thèse d'état, Université Louis Pasteur, Strasbourg, France*, 1991.
9. L. Tarsissi, D. Coeurjolly, Y. Kenmochi, and P. Romon. Convexity preserving contraction of digital sets. In S. Palaiahnakote, G. Sanniti di Baja, L. Wang, and W. Qi Yan, editors, *Pattern Recognition - 5th Asian Conference, ACPR 2019, Auckland, New Zealand, November 26-29, 2019, Revised Selected Papers, Part II*, volume 12047 of *Lecture Notes in Computer Science*, pages 611–624. Springer, 2019.