Electronic Notes in Theoretical Computer Science 46 (2001)
URL: http://www.elsevier.nl/locate/entcs/volume46.html 11 pages

An Approach for the Estimation of the Precision of a Real Object from its Digitization

Fabien Feschet [1]

UMR 5823 - LASS
Université Claude Bernard Lyon 1
69622 Villeurbanne, France

Laure Tougne [2]

Laboratoire ERIC
Université Lumière Lyon 2
69676 Bron, France

Abstract

In this article, we study the problem of the estimation of the rigid transformations, composed by translations and rotations, of a real object such that the discretizations before and after the transformation are the same. Our method is based on the polyhedral simplex representation associated to the set of lines compatible with each border of a real polygon. We give some algorithms that allow to decide whether a transformation is "valid" for a polygon and to compute the "maximal possible transformations".

Keywords: discrete curve, polygonalisation, discretization, error estimation.

1 Introduction

In the following, we tackle the problem: having obtained a discretization of an object of $\mathbb{R}^2$, we try to recover the set of real objects generated by rigid transformations of the original one such that the discretization viewed as a set of $\mathbb{Z}^n$ remains fixed. This problem comes from medical imaging. In the context of cancer therapy, the problem is to irradiate a tumor. Dosimetry techniques aim at computing the beam energy level which is necessary to damage the tumor at most. The beams must be concentrated on the tumor itself through the use of multi-leaves collimators which deform the beam into a shape similar to the

[1] Email: feschet@univ-lyon1.fr
[2] Email: ltougne@univ-lyon2.fr

one of the tumor in the projection plane of irradiation. However, dosimetry and beam correction suppose that the tumor is at the center of the irradiation beam. To ensure this, there are mainly two techniques. The first one uses moulds of the patient so to forbid all moves. This is routinely used but is not a simple protocol. In the second technique, it is proposed to use automatic or semi-automatic registration algorithms [6,3] to detect mispositioning. The current position of the patient is compared to its theoretical one. However, one important point in those algorithms is to have an idea of the precision of the registration in order to check if the result is compatible or not with the limits imposed by medical constraints. The precision is computed on the digital image of the tumor. So the purpose of this study is to check, given a discrete set, if the set of real objects, whose digitization is the given set, has a low scattering or not. Indeed, we conjecture that from a given discretization of an object, the set of possible real objects is not large. One result in that direction is the one of Newman [7]. Given a positive ε, there is a positive δ such that any closed convex plane curve whose curvature is bounded by δ must come within ε of a lattice point. Thus, for any closed convex curve there exist some anchors points which limit the permissible movements. We can also make a close relation to the studies done in discrete geometry. Considering objects as a set of n-cells, these works are related to the discovery of geometrical structure. In theses approaches there are mainly two different ways. One consists in defining discrete geometrical primitives and to search for subsets of n-cells verifying them. We can cite the recognition of lines [5] or circles [2]. An other approach consists in finding continuous primitives inside a discrete set so to recover a continuous structure compatible with it. We refer to the work of Vittone and Chassery [10] for the recovery of plane and to the work of Vialard and Braquelaire [9,4] with the introduction of Euclidean Paths.

In the following, we first introduce the discretization scheme we use and propose an analysis in the case of a real segment. This analysis is then extended to the case of a polygonal curve. This approach is based on the use of the polyhedral domain associated to the set of lines compatible with the discretization scheme. This domain can be defined for every edges of the polygonal curve. We first study the problem that consists in deciding whether a transformation is valid or not. Valid in this case means that the discretizations before and after the transformations remain the same. We then solve the problem of computing the "largest transformations" allowed. Such a work leads us to consider a more general one that consists in looking for the "farthest real polygons" such that their digitizations are the same given one. Finally, we propose some future works.

2 Study of one segment case

Let us consider one border $\mathcal{B}$ of a polygon $\mathcal{P}$. Let us denote respectively by m_l and m_r the extremities of the border $\mathcal{B}$. Without lost of generality, we can suppose $\mathcal{B}$ to be in the first octant, we can proceed symmetrically in the others cases. Consequently, m_l and m_r respectively correspond to the left and right extremities of the border $\mathcal{B}$. Let us denote by D a process of digitization of the segment $\mathcal{B}$. We call B the corresponding digitization, that is to say: $B = D(\mathcal{B})$. Let us denote by M_l and M_r the respective left and right extremities of the discrete segment B. The goal of this part is to characterize the family of straight lines $\mathcal{F}$ that have the same discretization between the points M_l and M_r.

2.1 Discretization scheme

We suppose $\mathcal{B}$ to be a segment without restriction concerning either its slope or its extremities. Consequently, we can define it as the set of points of $\mathbb{R}^2$ such that $y = ax + b$ with a the slope and b the second axis coordinate, with x between m_l and m_r (points of $\mathbb{R}^2$).

Many discretization schemes are possible. For the sake of clarity of the explanations and figures, we can fix the process of digitization D. For example, we can consider $D(\mathcal{B})$ to be the set of integer points (x, y) such that $x_{M_l} \leq x \leq x_{M_r}$ with $x_{M_l} = \lceil x_{m_l} \rceil$ and $x_{M_r} = \lfloor x_{m_r} \rfloor$, and $y = [ax + b]$ where $[x]$ denote the nearest integer from x. But, the algorithms we will describe in the following will be available for every “good” discretization process. The figure 1 shows an example of real segment and its discretization obtained by such a digitization process.

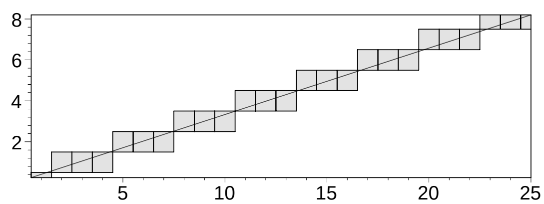

Fig. 1. An example of real segment and its discretization

2.2 Intervals

Let us remark that for a given integer abscissa x, with the previous process of digitization D, all the points (x, y') with y' between $[ax + b] - 0.5$ and $[ax+b]+0.5$ have the same digitization: the point of coordinates $(x, [ax+b])$. We can represent it with intervals as in figure 2.

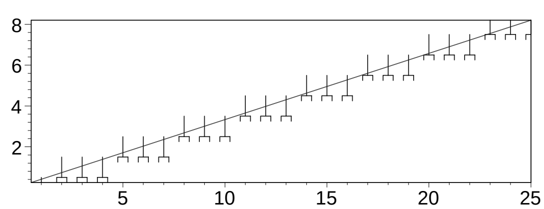

Fig. 2. An example of real segment and its intervals

This remark leads us to formulate the problem of one straight line case as follows.

Problem 2.1 *What are the parameters A and B of the real lines of equation $y = Ax + B$ that satisfy:*

$$\text{for all } x \text{ in } [M_l, M_r],\ [ax+b] - 0.5 < Ax + B \leq [ax+b] + 0.5$$

In the following part, we explain a way to solve such a problem.

2.3 Resolution of one segment problem

The set of previous equations can be viewed as the constraint equations of a two variables (A and B) linear programming problem and our goal is to find the "feasible region" R in the A-B parameters space.

Such a problem has already been studied in a more general context by O'Rourke [8]. In his paper, O'Rourke presents an on-line linear algorithm for fitting straight lines between data ranges. As a matter of fact, each equation of the previous system represents two half-planes with parallel edges in the A-B space and the "feasible region" R can be constructed as the intersection of such half-planes.

Consequently, if we suppose a segment to be given by these extremities (m_l and m_r), we can summarize the process as follows.

Feasible-Region-Segment (m_l,m_r)
Compute the slope a
Compute the second axis coordinate b
Compute the coordinates of M_l and M_r
$S = \emptyset$
For all x in $[x_{M_l}, x_{M_r}]$
  $S = S \cup \{[ax+b] - 0.5 < Ax + B, Ax + B \leq [ax+b] + 0.5\}$
$R = SolveORourke(S)$

The so generated convex region, for the example of the figure 1, is the one given in figure 3 where all the lines are drawn for explanation purposes.

Let us now study what we obtain if we consider all the borders of a polygon.

3 Study of polygonal curve

Let $\mathcal{P}$ be a polygon given by its n vertices denoted by S_i with $i = 0, ..., n-1$. In this part we present algorithms that allow to compute the various polygons that have the same discretization as the polygon $\mathcal{P}$. The first part deals with a direct algorithm that computes all the possible polygons without preserving the angles and the second one studies rigid transformations.

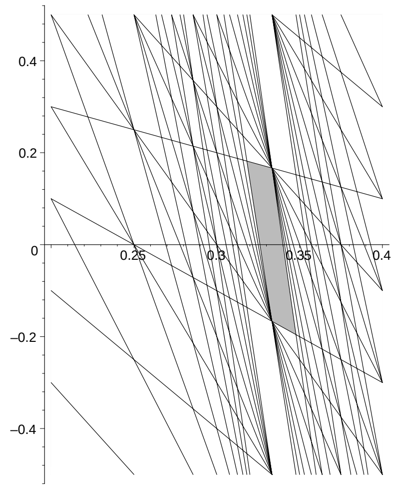

Fig. 3. The feasible region associated to the segment of the figure 1

3.1 Naive algorithm

On its n borders, we can apply the previous algorithm in order to determine for each of them the straight lines that have the same digitization. That is to say, for each border $[S_i, S_{(i+1)\%n}]$ we compute the feasible region R_i of the A-B space.

Just remark that each border of the polygon $\mathcal{P}$ corresponds to a point inside each feasible region. Consequently, if we consider one point of each feasible corresponding region R_i (distinct from the previous ones), we obtain a new polygon that have the same digitization as the polygon $\mathcal{P}$. In other words, such a process allows to decide whether transformations T_i applied on each vertex S_i of the polygon $\mathcal{P}$ give a polygon that has the same digitization as $\mathcal{P}$ or not. We can summarize it as follows:

Same-digitizations (S_i, T_i $(i = 0, ..., n-1)$)
For i in $[0, n-1]$
  Compute R_i=Feasible-Region-Segment $(S_i, S_{(i+1)\%n})$
  $S'_i = T_i(S_i)$
same=true
For i in $[0, n-1]$
  $V'_i = [S'_i, S'_{(i+1)\%n}]$
  If $V'_i \notin R_i$ then same=false
If same=true then return(true) else return(false)

The figure 4 illustrates this algorithm.

It shows two polygons that have the same discretization. Just remark that according to the slope of the borders, only very small moves are allowed.

Notice that the polygons we obtain with this process have not necessarily the same angles as the polygon $\mathcal{P}$.

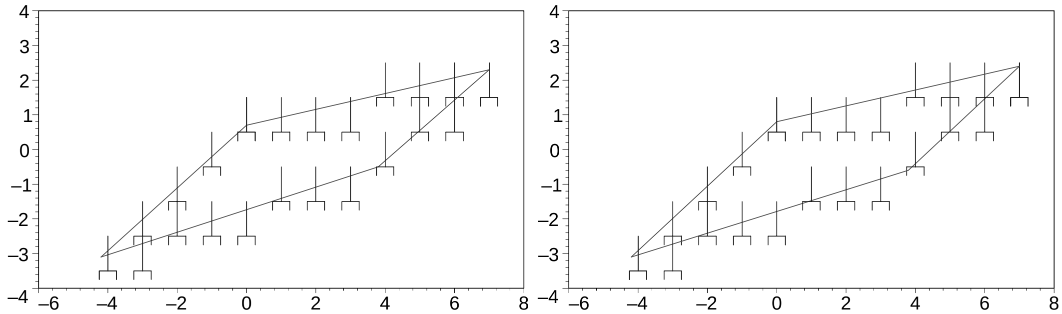

Fig. 4. Two polygons that have the same discretization

The goal of the following subsection is to consider rigid transformations in order to preserve the "geometry of the object".

3.2 Rigid transformations

In this paragraph we make two studies. The first one supposes that the polygon $\mathcal{P}$ is known, in other words we apply the same transformation to each starting border. In such conditions we are lead to consider the maximal possible translation according to a direction and the maximal possible rotation centered on a given point. In the second part, only the "geometry of the object" is known that is to say the length of the borders and the angles of the polygon are supposed given but not the borders themselves. The goal here is, starting from a given discretization, to determine the maximal translation and the maximal rotation that a real polygon if we suppose that this starting polygon and its image have the same given transformation.

3.2.1 Starting object is known

In this part, we consider a unique transformation T which will be applied on the polygon $\mathcal{P}$. So, we consider the following problem.

Problem 3.1 *Let $\mathcal{P}$ be a polygon. Let T be a rigid transformation. Does T be a valid transformation for $\mathcal{P}$ or not ?*

We say that a transformation is valid if and only if the polygon, obtained from the polygon $\mathcal{P}$ using the transformation T, has the same discretization as $\mathcal{P}$.

We can use the previous algorithm in order to solve this problem. In this case, the transformations applied to the borders would be the same ones : the transformation T. So, for each border $[S_i, S_{(i+1)\%n}]$ of the starting polygon, we compute the feasible region R_i and its parameters after the transformation T. Such transformed borders correspond to points (a_i, b_i) in the A-B parameters space. So, the transformation T is valid if and only if each point (a_i, b_i) is in R_i.

Consequently, we obtain the following algorithm.

Valid-Transformation $(S_i,\, T)$
For i in $[0, n-1]$
  Compute R_i=Feasible-Region-Segment $(S_i, S_{(i+1)\%n})$
  $S'_i = T(S_i)$
valid=true
For i in $[0, n-1]$
  $V'_i = [S'_i, S'_{(i+1)\%n}]$
  If $V'_i \notin R_i$ then valid=false
If valid=true then return(true) else return(false)

The figure 5 illustrates this method. It shows the feasible region of each border of the polygons of the figure 4 and the corresponding points of the started (points) and obtained (crosses) borders.

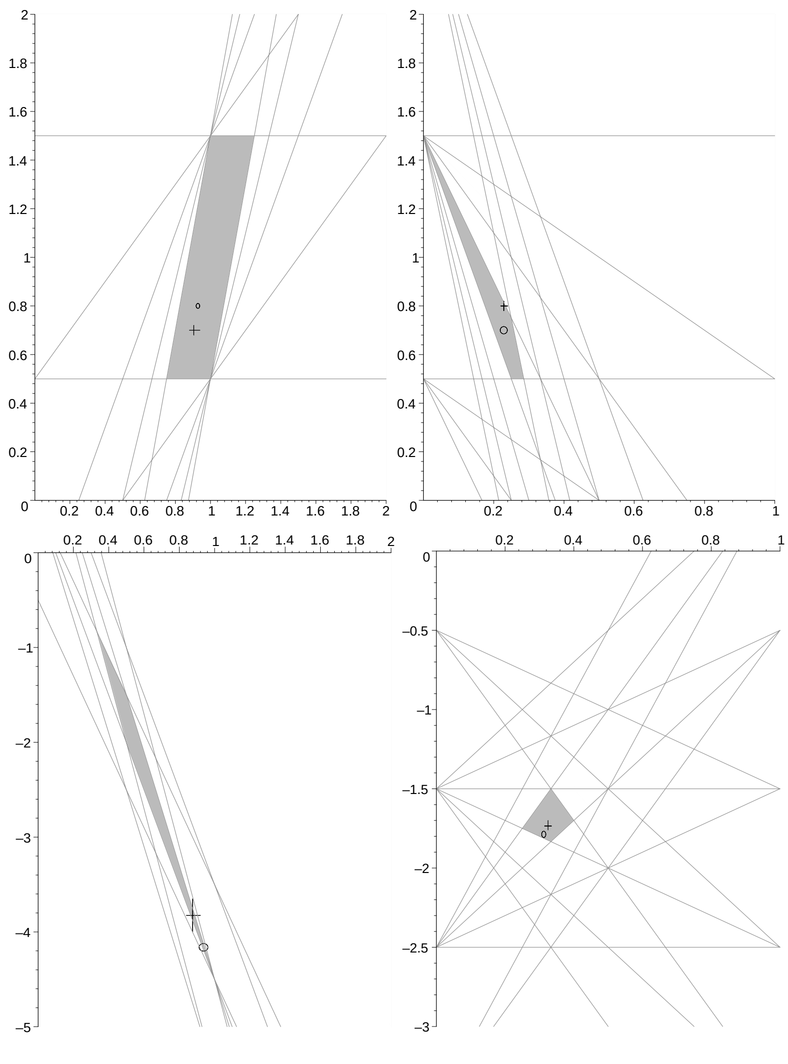

Fig. 5. Feasible region of each border of the polygons of the figure 4 and the corresponding points of the started (points) and obtained (crosses) borders

Such an approach leads us to study the "maximal possible transformations". In fact, we first interest ourselves to the maximal possible translation according to a direction and generalize the result to general transformations.

Let us consider the case of translation. Then the goal is to compute the largest α such that the translation of vector $\alpha(v_x, v_y)$ is valid for the starting polygon $\mathcal{P}$. As a translation corresponds to a vertical move of the corresponding point in the feasible region, it is enough to compute for each border the corresponding parameter B'_i obtained after translation (function of α). Then the maximal possible translation is given by taking the largest α such that all the obtained points are inside the corresponding feasible regions. The algorithm is the following one.

Maximal-Translation $(S_i, (v_x, v_y))$
For i in $[0, n-1]$
  Compute R_i=Feasible-Region-Segment $(S_i, S_{(i+1)\%n})$
  $B'_i = B_i + \alpha(v_y - A_i v_x)$
  Compute $\alpha_i^{\max}$ such that $B'_i \in R_i$
$\alpha = \min\{\alpha_i^{\max} \mid i = 0 \ldots n-1\}$
return(α)

Just remark that the case of rotation is quite similar. As a matter of fact, we can decompose a rotation centered on any point into a rotation centered on an extremity of the border and a translation. As previously, the translation corresponds to a vertical move of the corresponding point in the feasible region and the rotation centered on an extremity of the border corresponds to an horizontal move (only the slope of the border changes). Consequently, we just have to compute for each border the corresponding parameters obtained in the feasible region. Such parameter are functions of α the angle of the rotation and the maximal rotation is given by the maximal angle such that all the corresponding points are inside the feasible regions.

More generally, every transformation that can be decomposed into a rotation and a translation can be proceeded as previously.

Until here we have supposed the starting polygon $\mathcal{P}$ to be known. In fact, it corresponds to a preliminary work to a more general one which consists in finding the "farthest real polygons" which correspond to the same given digitization. In the following part, we explain such a problem and give some ideas according to the previous paragraphs to solve it.

3.2.2 Unknown borders

In the following, we do not suppose to know the borders of the real polygon but only its angles (α_i). Thus, the data is composed of those angles and the digitization of the real polygon. From this digitization, we can compute all the feasible regions of the different borders using the previous algorithms. We also suppose that the moves are all rigid thus the original angles are preserved by any transformations considered here. The figure 6 presents an example of two real objects having the same digitization. One is obtained from the other by the translation of vector $\pm(u_x, u_y)$. This corresponds to the extremal positions of the real polygon according to the given transformation.

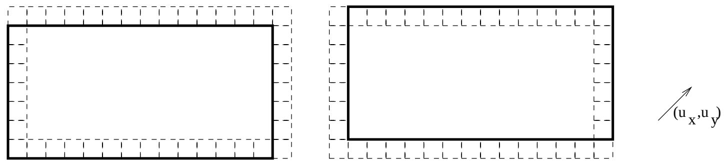

Fig. 6. The two farthest objects having the same discretization for the given translation (u_x, u_y)

An idea to solve such a problem consists in taking into account the problem of joining two consecutive borders of the real polygon. Let us recall that the feasible region of a border is a convex polygon having at most four borders[8,1]. For a given corner of the discrete object, we can thus define, for the last two intervals of two consecutive borders, two limiting lines having $(B_{\max}, A_{\min})$ and $(B_{\min}, A_{\max})$ such that all points inside the strip belong to a line of the feasible region (see Fig. 7).

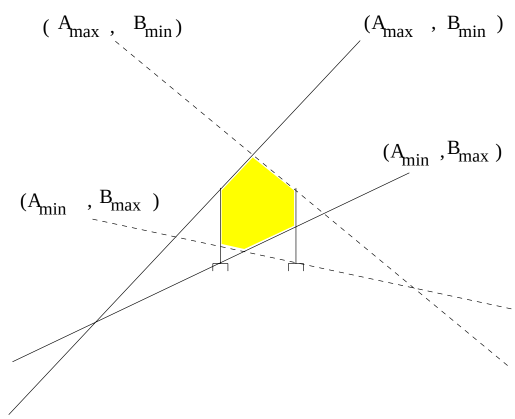

Fig. 7. The set of possible joining points without taking into account the angle

For each point of the previously defined region, there exists one line compatible with the left discrete segment and one line compatible with the right discrete segment such that those lines meet at the given point. However, the angle of the junction has not been taken into account. Doing this require to restrict the joining region. To do so, let us choose one point has a reference in the region of junction, say $C = (x_c, y_c)$. For each border, we need the set of lines $D = (A, B)$ passing trough C thus we need that $y_c = Ax_c + B$ which is equivalent to $B = -Ax_c + y_c$. Thus, this set is a line in the (A, B)-space. The intersection of this line and the feasible region of each border is a segment. It is clear that for each point of the segment of the left feasible region, we have a slope A_{left} for which we can associate the slope of the line having a fixed angle α with it. Thus, it is possible to determine the set of possible slopes A_{right}. The set of possible lines for the right border is thus the subset of the segment generated by D, having a slope in the set of possible A_{right} (see Fig. 8).

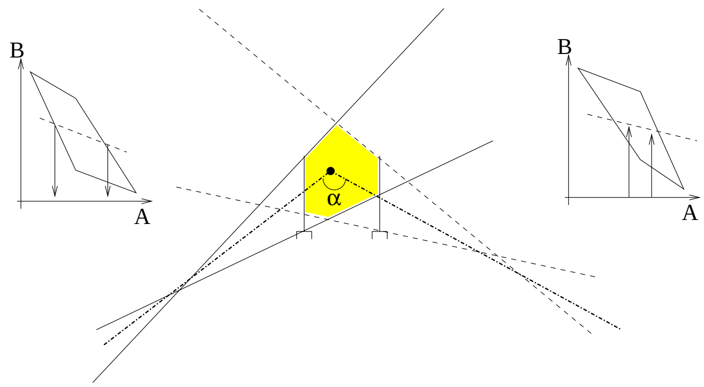

Fig. 8. The correspondence between one point in the left feasible region and the set of possible lines in the right one

Of course, because of complexity of the previous procedure, this can not be done for each point of the joining region. However, this can be done for the border of the region generating the border of the region of the right feasible region compatible with an α-join with a line of the left feasible region. Then, having defined the subset of the feasible region for each consecutive border, the algorithms of the previous sections can be applied to check what transform is valid and what are the *biggest* transforms.

4 Conclusions and future works

In this paper, we have presented an approach to characterize all the real objects that have the same discretization. Considering a polygonal object, we have first studied the case of one segment and have shown that in this case, we can reduce the problem to a two-variables linear programming problem. Then we have applied such a result on the whole object and have described an algorithm that allows to determine whether a transformation applied to a polygon is valid or not. We have also studied the problem of "maximal" valid transformations. Such a work leads us to the problem that consists in finding the "farthest real polygons" which correspond to the same given digitization. Some elements to solve such a problem have been given but need to be studied more.

A direct extension could be to extend these results to general curves. One way to do this is to approach the curve interiorly and exteriorly by two polygonal lines. Finally, we could try to study the case of 3D real object and in particular of polyhedral ones.

Acknowledgments

We thank Professor Serge Miguet of University of Lyon 2 for his constant support.

References

[1] Anderson, T. A. and C. E. Kim, *Representation of digital line segments and their preimages*, in: *Seventh International Conference on Pattern Recognition (Montreal, Canada, July 30-August 2, 1984)*, IEEE Publ. 84CH2046-1 (1984), pp. 501–504.

[2] Andrès, E., *Discrete Circles, Rings and Spheres*, Computer and Graphics **18** (1994), pp. 695–706.

[3] Bansal, R., L. Staib, Z. Chn, A. Rangarajan, J. Knisely, R. Nath and J. Duncan, *A novel approach for the registration of 2d portal and 3d ct images for treatment setup verification in radiotherapy*, in: W. Wells, A. Colchester and S. Delp, editors, *MICCAI 98*, LNCS **1496** (1998).

[4] Braquelaire, J. P. and A. Vialard, *Euclidean Paths: A new Representation of Boundary of Discrete Regions*, Graphical Models and Image Processing **61** (1999), pp. 16–43.

[5] Debled-Rennesson, I. and J. P. Reveillès, *A linear algorithm for segmentation of digital curves*, Parallel Image Analysis: Theory and Applications (1993), pp. 73–100.

[6] Feschet, F., "Technique d'imagerie pour l'aide au positionnement en dosimétrie conformationnelle," Ph.D. thesis, Insa de Lyon (1999).

[7] Newman, D. J., *A gradually turning curve must come near a lattice point*, Journal of Number Theory **6** (1974), pp. 7–10.

[8] O'Rourke, J., *An on-line algorithm for fitting straight lines between data ranges*, Communications of the ACM **24** (1981), pp. 574–578.

[9] Vialard, A., *Geometrical parameters extraction from discrete paths*, in: S. Miguet, A. Montanvert and S. Ubéda, editors, *6th International Conference on Discrete Geometry for Computer Imagery (DGCI'96)*, LNCS **1176** (1996), pp. 24–35.

[10] Vittone, J. and J. M. Chassery, *(n-m)-cubes and farey nets for naive planes understanding*, in: G. Bertrand, M. Couprie and L. Perroton, editors, *8th International Conference on Discrete Geometry for Computer Imagery (DGCI'99)*, LNCS **1568** (1999), pp. 76–87.