Recognition of Blurred Pieces of Discrete Planes

Laurent Provot, Lilian Buzer, Isabelle Debled-Rennesson

HAL Id: hal-00166285

https://hal.archives-ouvertes.fr/hal-00166285

Recognition of Blurred Pieces of Discrete Planes

L. Provot[1] and L. Buzer[2] and I. Debled-Rennesson[1]

[1] LORIA Nancy
Campus Scientifique - BP 239
54506 Vandœuvre-lès-Nancy Cedex, FRANCE
{provot,debled}@loria.fr
[2] Laboratory CNRS-UMLV-ESIEE, UMR 8049
ESIEE, 2, boulevard Blaise Pascal
Cité DESCARTES, BP 99
93162 Noisy le Grand CEDEX, FRANCE
buzerl@esiee.fr

Abstract. We introduce a new discrete primitive, the blurred piece of a
discrete plane, which relies on the arithmetic definition of discrete planes.
It generalizes such planes, admitting that some points are missing and
then permits to adapt to noisy discrete data. Two recognition algorithms
of such primitives are proposed: the first one is a geometrical algorithm
and minimizes the Euclidean distance and the second one relies on linear
programming and minimizes the vertical distance.

1 Introduction

The recognition of discrete primitives as digital straight lines and digital planes
is a deeply studied problem in digital geometry (see a review in the book [1]).
This problem consists in determining if a set of discrete points corresponds to
a known discrete primitive and, in such case, in identifying its characteristics.
Three main classes of algorithms can be defined:

- Structural algorithms: based on geometric (convex hull, chords) or com-
  binatorial (size of the steps) properties of discrete primitives. Indeed, the
  structural regularity of these primitives can lead to efficient algorithms.
- Arithmetic algorithms: based on the definition of discrete primitives as Dio-
  phantine inequalities, these algorithms make profit of the well defined arith-
  metical structure of discrete primitives.
- Dual space algorithms: the recognition problem is translated in a dual space
  where each grid point is represented by a double linear constraint. The recog-
  nition problem is then defined as a linear programming problem, optimized
  using particular knowledge on the constraints geometry.

Recently, a new discrete primitive, the blurred segment [2, 3], was introduced
to deal with the noise or artefacts due to the acquisition tools or methods. Re-
lying on an arithmetic definition of discrete lines [4], it generalizes such lines,
admitting that some points are missing. Efficient blurred segments recognition

algorithms were proposed [2, 3, 5] and they were used in applications in image analysis [6]. In the same framework, we introduce in the paper the new notion of blurred pieces of discrete planes, relying on the definition of arithmetic discrete plane [7] by considering a variable thickness.

Two recognition algorithms of blurred pieces of discrete planes are proposed. The first one is based on a structural approach: the computation of the convex hull of the given voxels is done while we search for the two parallel planes that mark out this convex hull and that minimize the Euclidean distance between themselves. An incremental algorithm is given. The second one is based on a dual space approach in the context of linear programming: the recognition problem is modelled by a system of linear constraints defined by the initial set of points. The simplex algorithm is then used to solve the problem by minimizing the vertical distance between two parallel planes containing all the points of the initial set. A geometrical interpretation of this method is also given. The codes of these algorithms and examples are available on `http://www.loria.fr/~debled/BlurredPlane`.

In section 2, after recalling definitions and basic properties of arithmetic discrete planes, we define the related notion of blurred pieces of discrete planes and optimal bounding planes. Then, in section 3, a geometrical method is proposed to solve the recognition problem by minimizing the Euclidean distance. The second method, based on linear programming, is presented in section 4 as well as a geometrical interpretation of the dual problem. The paper ends up with some conclusions and perspectives in section 5.

2 Blurred Pieces of Discrete Planes

An **arithmetic discrete plane** [7], named $\mathcal{P}(a, b, c, \mu, \omega)$, is a set of integer points (x, y, z) verifying $\mu \leq ax + by + cz < \mu + \omega$ where $(a, b, c) \in \mathbb{Z}^3$ is the normal vector. $\mu \in \mathbb{Z}$ is named the translation constant and $\omega \in \mathbb{Z}$ the arithmetical thickness.

The two real planes, defined by the following equations: $ax + by + cz = \mu$ and $ax + by + cz = \mu + \omega - 1$, are called the **leaning planes** of $\mathcal{P}(a, b, c, \mu, \omega)$. All the points of $\mathcal{P}$ are located between the leaning planes of $\mathcal{P}$.

We hereafter propose a generalization of the notion of discrete plane relying on the arithmetical definition and admitting that some points are missing. Consider a norm N on $\mathbb{R}^3$. We define the notion of bounding plane, relative to N, as follows:

Definition 1. *Let $\mathcal{E}$ be a set of points in $\mathbb{Z}^3$. We say that the discrete plane $\mathcal{P}(a, b, c, \mu, \omega)$ is a **bounding plane of** $\mathcal{E}$ if all the points of $\mathcal{E}$ belong to $\mathcal{P}$. We call **width of** $\mathcal{P}(a, b, c, \mu, \omega)$, the value $\frac{\omega - 1}{N(a,b,c)}$.*

Interpretation of the Width:

1. if $N = \|\cdot\|_2$, the width $\frac{\omega-1}{N(a,b,c)}$ represents the Euclidean distance between the two leaning planes of the bounding plane $\mathcal{P}(a,b,c,\mu,\omega)$. Indeed, let $P_1 :$ $ax+by+cz = \mu$ and $P_2 : ax+by+cz = \mu+\omega-1$ be the two leaning planes of $\mathcal{P}$. As P_1 and P_2 are parallel, the distance between P_1 and P_2 is equal to $\frac{|\mu+\omega-1-\mu|}{\sqrt{a^2+b^2+c^2}}$, i.e. $\frac{\omega-1}{\|(a,b,c)\|_2}$ since $\omega > 0$.

2. if $N = \|\cdot\|_\infty$, the width $\frac{\omega-1}{N(a,b,c)}$ represents the distance according to the main direction of the vector (a,b,c). Indeed and without loss of generality we can assume that $\max(|a|,|b|,|c|) = |c|$, which means the main direction is the Oz axis. Let $M_1(x_1,y_1,z_1) \in P_1$ and $M_2(x_2,y_2,z_2) \in P_2$ such that $x_1 = x_2$ and $y_1 = y_2$. The distance between P_1 and P_2 is equal to $|z_1 - z_2| = \frac{|c(z_1-z_2)|}{|c|} = \frac{|a(x_1-y_2)+b(y_1-y_2)+c(z_1-z_2)|}{|c|} = \frac{|\mu-(\mu+\omega-1)|}{|c|}$ because $M_1 \in P_1$ and $M_2 \in P_2$, i.e. $\frac{\omega-1}{\|(a,b,c)\|_\infty}$ since $\omega > 0$.

(a)   (b)

(c)   (d)
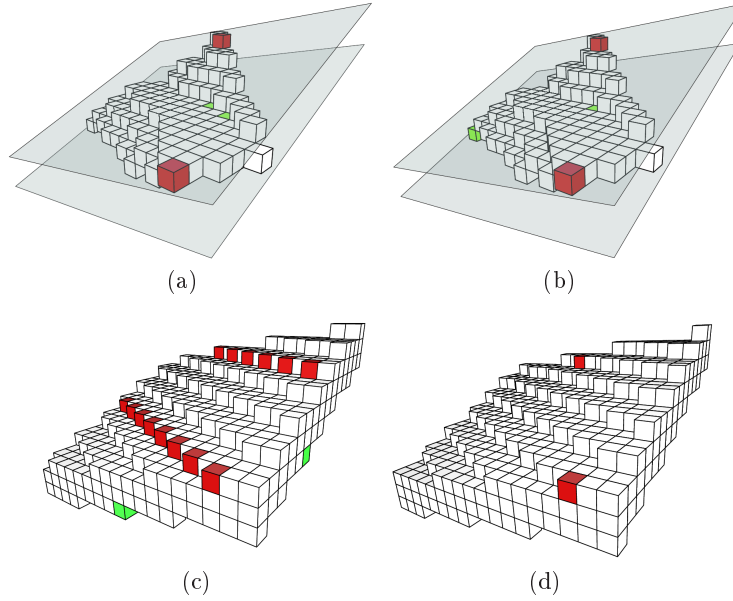
Fig. 1. A width-3 blurred piece of discrete plane (a and b), its optimal bounding planes (c) for Euclidean norm: $\mathcal{P}_2(4,8,19,-80,49)$ and the width of $\mathcal{P}_2 = 2.28$ (d) for infinity norm: $\mathcal{P}_\infty(31,65,157,-680,397)$ and the width of $\mathcal{P}_\infty = 2.52$. The leaning planes and corresponding leaning points of $\mathcal{P}_2$ and $\mathcal{P}_\infty$ are respectively drawn on (a,c) and (b,d).

Definition 2. *Let $\mathcal{E}$ be a point set in $\mathbb{Z}^3$. A bounding plane of $\mathcal{E}$ is said* **optimal** *if its width is minimal.*

This leads us to the definition of a blurred piece of discrete plane (Fig. 1).

Definition 3. *A point set $\mathcal{E}$ in $\mathbb{Z}^3$ is a **width-ν blurred piece of discrete plane** if and only if the width of its optimal bounding plane is less or equal to ν.*

In the following sections we propose two algorithms which solve the recognition problem of blurred pieces of discrete planes. For a given set of points $\mathcal{E}$ in $\mathbb{Z}^3$ and a width ν these algorithms decide whether $\mathcal{E}$ is a width-ν blurred piece of discrete plane. In addition, they give the characteristics of an optimal bounding plane of $\mathcal{E}$ for which the width is minimal. We also show how these algorithms can be made incremental.

3 Geometrical Method for the Recognition of Blurred Pieces of Discrete Planes

The first approach allows to solve the problem in terms of the norm $\| \cdot \|_2$. It relies on the computation of the width of a point set in 3-space [8, 9].

Definition 4. *Let E be a set of points in $\mathbb{R}^3$ and P a real plane. We say that P is a **plane of support** of E if all the points of E are located in one of the two half-spaces delimited by P and such that $P \cap E \neq \emptyset$.*

Definition 5. *The **width** of E is the smallest (Euclidean) distance between two parallel planes of support of E called* width planes.

The link with our problem is the following: if E is a set of points in $\mathbb{Z}^3$ then the width planes coincide with the leaning planes of an optimal bounding plane of E and the width of E is equal to the width of this optimal bounding plane. For that reason, computing the width and deducing the width planes allow to recognize blurred pieces of discrete planes.

3.1 Width Computation

We are looking for two parallel planes $P_1 : \alpha x + \beta y + \gamma z + \delta_1 = 0$ and $P_2 : \alpha x + \beta y + \gamma z + \delta_2 = 0$ which minimize the distance $\frac{|\delta_2 - \delta_1|}{\sqrt{\alpha^2 + \beta^2 + \gamma^2}}$ between P_1 and P_2 and such that, for all points $p(p_x, p_y, p_z) \in E$, we have $p_x + \beta p_y + \gamma p_z + \delta_1 \leq 0$ and $p_x + \beta p_y + \gamma p_z + \delta_2 \geq 0$. For this purpose we can see that the width of E is the same as the width of its convex hull $\mathcal{CH}(E)$ [8]. It is due to the fact that $\mathcal{CH}(E)$ is the intersection of all the half-spaces containing all the points of E. We can then simplify the problem by introducing antipodal pairs. Consider the convex hull of a set of points E in 3-space. Two of its edges form an **antipodal edge-edge (E-E) pair** when two parallel planes of support of E contain these edges. Similarly, we define **vertex-vertex (V-V)**, **face-face (F-F)**, **vertex-face (V-F)**, **vertex-edge (V-E)** and **edge-face (E-F) pairs**.

In [8], M.E. Houle and G.T. Toussaint show that, to compute the width of E, it is sufficient to focus only on parallel planes which contain an E-E pair or a V-F pair. Therefore, we will enumerate all the E-E and V-F pairs of $\mathcal{CH}(E)$ and keep the ones whose distance is minimal.

In [9], B. Gärtner and T. Herrmann propose a direct approach relying on the geometry and combinatorial properties of the convex hull. The method is inspired from the *rotating calipers* [10] but generalized to the three-dimensional space. They start with an arbitrary face f of $\mathcal{CH}(E)$ and determine its antipodal vertices $V = \{v_1, \ldots, v_k\}$ by exploring all the vertices of $\mathcal{CH}(E)$. Thus, they obtain an initial V-F pair and the two parallel planes P_1 and P_2 supporting V and f respectively. Next, they rotate the two planes about an incident edge e of f until P_2 supports the other facet f' incident to e. During this rotation the parallelism and the supporting property of the two planes are preserved and all E-E pairs belonging to e as well as the antipodal vertices of f' are reported.

The important part is as follows: given a V-E pair (w, e) and two parallel planes P_1 and P_2 supporting w and e respectively, two events of interest might happen during the rotation of P_2 about e:

1. P_2 supports a new face f' incident to e, a new V-F pair (w, f') is found.
2. P_1 supports an additional vertex v, a new E-E pair $((wv), a)$ is found.

Thus, a rotation about an edge e of $\mathcal{CH}(E)$ allows to get all E-E pairs belonging to e and all V-F belonging to the two incident faces of e. Hence, by rotating about all the edges of $\mathcal{CH}(E)$ we get all the possible E-E and V-F pairs of $\mathcal{CH}(E)$. At least one of them belongs to the width planes and the distance between these planes is the width $\mathcal{W}$ of E.

As $\mathcal{W}$ represents the width of an optimal bounding plane of E, if $\mathcal{W} \le \nu$ then E is a width-ν blurred piece of discrete plane.

Furthermore, we can obtain the characteristics of this optimal bounding plane. As the width planes coincide with the leaning planes of the bounding plane $\mathcal{P}(a, b, c, \mu, \omega)$ of E, we have $a = \alpha$, $b = \beta$ and $c = \gamma$. Relying on the width interpretation in Section 2, we get $\omega = |\delta_2 - \delta_1| + 1$. Lastly, owing to the leaning planes equations, $\mu = \min(-\delta_1, -\delta_2)$.

3.2 Incremental Algorithm

Here we propose an incremental version, in order to get an algorithm which gives the characteristics of an optimal bounding plane of E each time we add a new point. A naive method consists in recomputing the width of E each time we add a point. Nevertheless some observations allow to improve this process.

On the one hand, only one point differs from one step to another. Thus, we can advantageously replace the computation of the convex hull of all the points of E by an incremental computation ([11] pp 235–246). Let us briefly recall the procedure. At a general step i of the algorithm, a convex hull C_i is given and we add a new point M. If it lies inside C_i or on its boundary, then there is nothing to be done. Otherwise we look for all the visible[3] faces of C_i, standing from

---

[3] Consider a plane P_f containing a face f of the convex hull. By convexity, this convex hull is completely contained in one of the closed half-spaces defined by P_f. The face f is *visible* from a point if that point is located in the open half-space on the other side of P_f
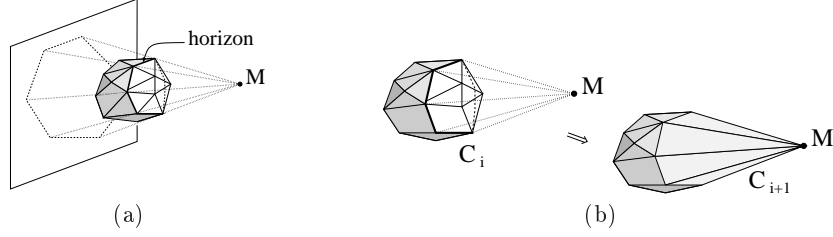
Fig. 2. (a) The horizon from M; (b) Adding a point to the convex hull.

M. This set of faces is enclosed by a curve called **_horizon_** (Fig. 2(a)). All the visible faces are removed from C_i and replaced by new ones created by joining each vertex of the horizon to the point M (Fig. 2(b)). Some of them could be coplanar with non-visible faces so they have to be merged together. The resulting polytope is the new convex hull C_{i+1}.

On the other hand, we can observe that, at each step of the algorithm, we know the characteristics of an optimal bounding plane $\mathcal{P}(a,b,c,\mu,\omega)$ of $\mathcal{E}$. So, if we add a point $M(x_M, y_M, z_M)$, we can compute the remainder value of M relative to $\mathcal{P}$: $r_{\mathcal{P}}(M) = ax_M + by_M + cz_M - \mu$. According to a property of discrete planes, if $r_M \in [0, \omega - 1]$ then $M \in \mathcal{P}$, so it is useless to recompute the width of $\mathcal{E}$ since it does not change.

Algorithm 1: Incremental Recognition

> **Data**: $\mathcal{E} \in \mathbb{Z}^3$, the convex hull C of $\mathcal{E}$, the characteristics a, b, c, μ and ω of the optimal bounding plane of $\mathcal{E}$
> **Input**: A point $M \in \mathbb{Z}^3$
> **Result**: The updated data after the addition of M

1 **begin**
2     $\mathcal{E} \longleftarrow \mathcal{E} \cup M$
3   Update C using the incremental process
4     $r_M \longleftarrow ax_M + by_M + cz_M - \mu$
5   **if** $r_M \notin [0, \omega - 1]$ **then**
6        $\langle \alpha, \beta, \gamma, \delta_1, \delta_2 \rangle \longleftarrow$ `ComputeWidthPlanes`(C)
7        $a \longleftarrow \alpha$
8        $b \longleftarrow \beta$
9        $c \longleftarrow \gamma$
10       $\mu \longleftarrow \min(-\delta_1, -\delta_2)$
11       $\omega \longleftarrow |\delta_2 - \delta_1| + 1$
12 **end**

This leads to the incremental procedure described in Algorithm 1. The function `ComputeWidthtPlanes`(C) at line 6 computes the width planes of C according to the method described in Section 3.1. The returned tuple contains the

coefficients of these planes.

Complexity: In [9], Gärtner and Herrmann showed that the complexity of computing the function `ComputeWidthPlanes`(C) is $\mathcal{O}(n^2)$, where n is the number of points in $\mathcal{E}$. As the other instructions of Algorithm 1 run in constant time, we obtain a complexity of $\mathcal{O}(n^2)$ for our incremental procedure. We need to use this incremental procedure each time we add a point to $\mathcal{E}$. Thus, we obtain an $\mathcal{O}(n^3)$ worst case complexity for a set $\mathcal{E}$ of n points. Nevertheless, in practice, the recognition process seems rather linear.

4 Linear Programming Method

The second method relies on linear programming and permits to solve the problem by considering the norm $\|\cdot\|_\infty$. We recall in the following section the general formulation of a linear programming problem and the simplex algorithm. The problem of recognition of blurred pieces of discrete planes is then modelled in that way in Section 4.2.

4.1 The Simplex Algorithm

Formulation. We try to identify a minimum point $x^* \in \mathbb{R}^d$ of a function $f(x) : \mathbb{R}^d \to \mathbb{R}$ where $x = (x_1, \ldots, x_d)$. Moreover, x^* must satisfy a set of n constraints $G = (g_i(x) \leq b_i)_{1 \leq i \leq n}$. LP is the specialization of mathematical programming to the case where both, the *objective function* f and the problem constraints G are linear. Let $A(n \times d)$ denote a matrix of n rows and d columns. Let $c(d)$, $b(n)$ and $x(d)$ denote three column vectors of size d and n. Thus, we can write our LP problem in such a way: Min $c^t.x$ subject to $A \cdot x \leq b$ and $x \geq 0$. We call the *standard form* the equivalent rewriting: Min $c'^t.x'$ subject to $A' \cdot x' = b$ and $x' \geq 0$ where $A' = [A|Identity(n \times n)]$, $c' = [c|Zero(n)]$. The n inserted variables in the standard form are called the *slack variables*.

The simplex algorithm. This method, developed by George Dantzig 1947, provides a powerful computational tool (see [12] for details). It operates on the formulation of the standard form. We have $n + d$ variables and n equalities in the system $Ax = b$, we can extract a nonsingular matrix B of rank n relative to this system of equations.
The *basis* corresponds to the indices of the columns extracted from A to create B. In the simplex method, the *nonbasic variables*, denoted by $x_N = (x_i)_{1 \leq i \leq n+d, i \notin basis}$ are forced to be zero. The *basic variables* $x_B = (x_i)_{i \in basis}$ are thus equal to $B^{-1}b$. A solution x associated with a basis B is called *feasible* when it verifies $x_B \geq 0$.

The simplex algorithm starts from a feasible solution. At each iteration, the program computes a new basis in such a way that the new basic solution is feasible and that the objective function has decreased or remains unchanged. To build the new basis, one nonbasic variable is reclassified as basic and vice versa.

Which variable can we choose ? Let N denote the columns of A whose indices are not in the basis. From $Ax = b$, we have: $[B|N].[x_B, x_N] = b$. As B is a non-singular matrix, we obtain: $x_B = B^{-1}.(b - N.x_N)$. The objective function can be rewritten as: $f(x) = c^t.x = c_B{}^t.x_B + c_N{}^t.x_N = (c_N{}^t - c_B{}^t B^{-1} N)x_N + c_B^t B^{-1} b$. This rewriting is not depending on the variables x_B. Thus, as the variables are positive, if there exists no negative value in the *reduced cost vector* $rc^t = c_N{}^t - c_B{}^t B^{-1} N$, we have found the minimum x^*.

If there exists a negative value, then we can decrease the current value of the objective function by increasing the corresponding variable x_l of x_N. As x_l is no more zero, at the next iteration, it will be reclassified as a basic variable. By increasing x_l, the values of the basic variables change. If they all increase, the problem is *unbounded*, it means that the minimum value for the objective function is $-\infty$. In the other case, where some basic variables decrease when x_l increases, the first basic variable x_k that reaches zero will stop the increase of x_l. Thus, x_k leaves the basis.

To determine the index k, let consider the equalities $x_B = B^{-1}.(b - N.x_N)$. Only x_l is now nonzero among x_N, so we have $x_B = B^{-1}.b - B^{-1} A_l x_l$. Let $\overline{b}$ and P denote $B^{-1}b$ and $B^{-1}A_l$. Values in $\overline{b}$ are positive, so only the indices associated with a positive value in P are of interest. The previous condition $\overline{b} - P.x_l \geq 0$ implies that for all i in the basis with $P_i > 0$, we have: $x_l \leq \overline{b}_i/P_i$. It follows that $k = index\ of\ min_{i,P_i>0}\{\overline{b}_i/P_i\}$.

> **function Min-Simplex**(A,b,c,basis)
> **Repeat**
>   1- **Extract** B, c_B from A // relative to the current basis
>   2- $\overline{b} = B^{-1}b$
>   3- $rc' = c^t - (c_B{}^t B^{-1}).A$ // equivalent version of rc
>   4- **If** $(rc' \geq 0)$ **return** $\overline{b}$ // optimum found ($\leq$ for a Max)
>   5- **Choose** l such that $rc'_l < 0$ // x_l enters the basis (> 0 for a Max)
>   6- $P = B^{-1}A_l$
>   7- **If** $P \leq 0$ **return** unbounded
>   8- $k = min_{i,P_i>0}\{\overline{b}_i/P_i\}$ // (same thing for a Max)
>   9- $basis \leftarrow basis \backslash \{k\} \cup \{l\}$

Duality theorem. Associated with each *Primal* LP problem is a companion problem called the *Dual*. The main theorem of LP proves that the Primal problem is infeasible iff the Dual problem is unbounded and vice versa. Moreover, one problem has an optimum iff the other problem has an optimum. The two optimum values are equal. Moreover, if c_B and B are the matrices associated with the optimum in the Dual, then the optimum in the Primal is equal to $c_B^t B^{-1}$.

Primal:	(i) Min $c^t.x$	$\longleftrightarrow$	**Dual:**	Max $b^t.\lambda$
Subject to:	(ii) $A.x \geq b$	$\longleftrightarrow$	Subject to:	$\lambda \geq 0$
	(iii) $A.x = b$	$\longleftrightarrow$		$\lambda \in \mathbb{R}$
	(iv) $x \geq 0$	$\longleftrightarrow$		$A^t.\lambda \leq c$
	(v) $x \in \mathbb{R}$	$\longleftrightarrow$		$A^t.\lambda = c$

4.2 Modelling the Recognition Problem

In this way, we compute the minimum vertical distance between two parallel planes whose slopes relative to the x-axis and the y-axis are between $\pm\pi/2$. Indeed, let us recall the given problem, we are looking for the characteristics a, b, c, μ, ω of an optimal discrete plane bounding $\mathcal{P}$ for a set of n points by minimizing the vertical distance between its two leaning planes. By considering $\alpha = -\frac{a}{c}$, $\beta = -\frac{b}{c}$, $h = \frac{\mu}{c}$ and $e = \frac{\omega-1}{c}$, the problem may be reformulated as follows: for a given set of n points (x_i, y_i, z_i), we want to find two planes $P : z(x, y) = \alpha.x + \beta.y + h$ and $P' : z'(x, y) = \alpha.x + \beta.y + h + e$ such that all the points are located between P and P' and such that e is minimal. We obtain one couple of inequalities for each entered point: $\alpha.x_i + \beta.y_i + h \leq z_i$ and $\alpha.x_i + \beta.y_i + h + e \geq z_i$.

Primal

Min e

$$\begin{cases} -\alpha.x_i - \beta.y_i - h \geq -z_i \\ \alpha.x_i + \beta.y_i + h + e \geq z_i \\ i = 1, \ldots, n \end{cases}$$

$|\alpha| \leq 1,\ |\beta| \leq 1$
$\alpha, \beta, h \in \mathbb{R},\ e \geq 0$

Dual standard form

Max $[-z_1 \ldots -z_n \mid z_1 \ldots z_n \mid$ -1 -1 -1 -1 0 $].\lambda$

$$\begin{bmatrix} -x_1 & \ldots & -x_n & x_1 & \ldots & x_n & \text{-1} & 1 & 0 & 0 & 0 \\ -y_1 & \ldots & -y_n & y_1 & \ldots & y_n & 0 & 0 & \text{-1} & 1 & 0 \\ -1 & \ldots & -1 & 1 & \ldots & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ldots & 0 & 1 & \ldots & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \ldots \\ \lambda_{2n+5} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$\lambda \geq 0$

We gather the two different types of inequalities on each side of the matrix. Working in the Primal problem with the standard form forces to manage a large sparse matrix of size $(2n+4) \times (2n+8)$. The Dual allows to bypass this problem with a $4 \times (2n+5)$ matrix ((i), (ii) and (v) in 4.1). We can easily check that the basis $\{\lambda_1, \lambda_{2n+1}, \lambda_{2n+3}, \lambda_{2n+5}\}$ where $B^{-1}b = [0\ 0\ 0\ 1]^t \geq 0$ is always a feasible basis for the Dual problem.

Geometrical Interpretation of the Dual Problem.

The basis of the Dual problem is associated with four inequalities in the Primal problem. So when λ_i is in the basis, the ith inequality in the Primal problem corresponds to an equality. For example, when $\lambda_i, 1 \leq i \leq n$ is in the basis, the ith inequality implies $\alpha.x_i + \beta.y_i + h = z_i$, this means that the point p_i belongs to the lower plane P. When $n < i \leq 2n$, the point p_{i-n} belongs to the upper plane P'. In the same way, the variables $\lambda_{2n+1}, \ldots, \lambda_{2n+5}$ are associated with the cases: $\alpha = 1, \alpha = -1, \beta = 1, \beta = -1$ or $e = 0$.
The vector $c_B^t B^{-1}$ in the Dual transforms the current basis into the primal variables. This follows from the previous remark. Let K denote the matrix corresponding to the equalities retained in the Primal problem. The current system verifies: $K \cdot [\ \alpha\ \beta\ h\ e\ ]^t = b_B^{Primal}$. Thus, we have: $[\ \alpha\ \beta\ h\ e\ ] = (K^{-1} \cdot b_B^{Primal})^t = (b_B^{Primal})^t \cdot (K^t)^{-1} = c_{B\ Dual}^t \cdot B_{Dual}^{-1}$
Reduced cost optimality condition. The simplex algorithm maximizes a function in the Dual. So, it stops when it finds an rc vector with negative values

(line 4). We easily verify that: $rc^t = c^t - (c_B{}^t B^{-1}).A = [$ $(-z_i + [\alpha.x_i + \beta.y_i + h])_{1 \leq i \leq n}$ | $(z_i - [\alpha.x_i + \beta.y_i + h + e])_{1 \leq i \leq n}$ | -1+α | -1-α | -1+β | -1-β | -e]. As all these values are negative, this implies that the inequalities of the Primal are all verified. The Dual program stops when it finds two parallel planes that include all the points and that have valid slopes.

The **objective function** in the Dual is quite obscure. Nevertheless relative to the theorem of Duality, the dual objective function must represent the same thing than the Primal function. In fact, we have $f(\lambda) = c_{BDual}^t (B^{-1} b_{Dual}) = (c_{BDual}^t B^{-1}) b_{Dual} = [\ \alpha\ \beta\ h\ e\ ] \times c_{Primal} = e$.

The core of the algorithm. Each iteration is associated with a feasible basis. We only consider in the following the two most important cases with all the basic variables λ_i such that $1 \leq i \leq 2n$. Other subcases can be processed without difficulty. The configuration $1 \leq i, j, k, l \leq n$ for the indices of the basic variables is not possible because the corresponding matrix B would be singular.

Configuration 1: $1 \leq i, j, k \leq n < l \leq 2n$. In this case, the three points $p_i(x_i, y_i), p_j(x_j, y_j), p_k(x_k, y_k)$ define the lower plane P and the parallel plane P' is supported by p_{l-n}. The matrix B is equal to $[$ -x_i -y_i -1 0 | -x_j -y_j -1 0 | -x_k -y_k -1 0 | x_{l-n} y_{l-n} 1 1 $]$. Wlog, we can assume that the point p_{l-n} corresponds to the origin, this allows to simplify the writing of the matrix B to $[$ -x_i -y_i -1 0 | -x_j -y_j -1 0 | -x_k -y_k -1 0 | 0 0 1 1 $]$. Let N_i denote the two-dimensional vector (x_i, y_i). The vector $B^{-1}b$ is equal to: $[\frac{N_k \wedge N_j}{det(B)}; \frac{N_i \wedge N_k}{det(B)}; \frac{N_j \wedge N_i}{det(B)}; 1]$. As the matrix B is nonsingular and $det(B) = -det([\ x_i\ y_i\ 1\ |\ x_j\ y_j\ 1\ |\ x_k\ y_k\ 1])$ the three points p_i, p_j, p_k must not be colinear. Suppose that the three points $N_i N_j N_k$ lie in clockwise order, so $det(B) \geq 0$. As $B^{-1}b \geq 0$, $N_k \wedge N_j, N_i \wedge N_k$ and $N_j \wedge N_i$ are positive. Such a situation can appear only when the point p_{l-n} lies inside the triangle $N_i N_j N_k$ relative to the projection into the O_{xy} plane.

Configuration 2: $1 \leq i, j \leq n < k, l \leq 2n$. The planes P (resp. P') is supported by the segment $p_i p_j$ (resp. $p_{k-n} p_{l-n}$). As they are parallel, this couple of planes is unique. Consider that the point p_{l-n} is centered on the origin, we have $B^{-1}b = [N_k \wedge N_j / \Delta; N_i \wedge N_k / \Delta; ...; ...]$ with $\Delta = (N_i - N_j) \wedge N_k$. Δ is nonzero iff the segment $p_i p_j$ and the segment $p_{k-n} p_{l-n}$ are not colinear in the Oxy plane. It follows that $N_k \wedge N_j$ and $N_k \wedge N_i$ have not the same sign. Thus, the segment $p_i p_j$ crosses the line $(p_{k-n} p_{l-n})$. When we center the origin on p_i, we symmetrically obtain the same result. Thus, this case is associated with two segments $p_i p_j$ and $p_{k-n} p_{l-n}$ that intersect each other relative to a projection into the Oxy plane.

Configuration 3: it is equivalent to the first configuration.

Variables interchanging. We traverse all the set of points. For each point, we consider its vertical distance from P when it lies under P or from P' when it lies above P'. If no points are found, our problem is solved. Otherwise, we select the point that is the vertically farthest point from P and P'. The associated variable λ_u enters the basis. In the **Configuration 1**, we have three equalities of the type: $\alpha.x + \beta.y + h = z$. When we select a variable λ_u of the same type, it means with $1 \leq u \leq n$, we can not withdraw p_{l-n}, otherwise we would obtain a
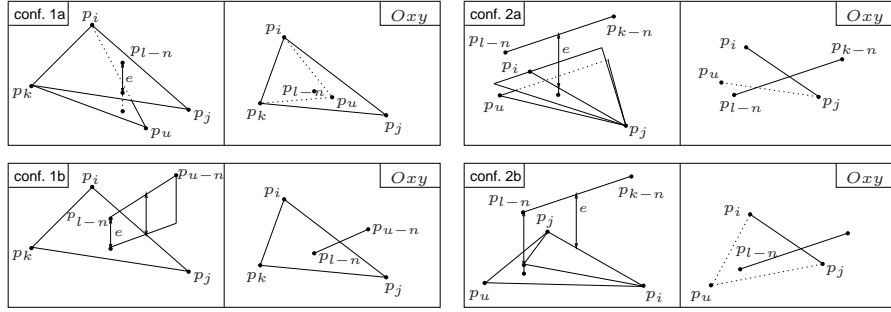
Fig. 3. Different configurations relative to the basis and the entering variable.

basis with four equalities of the same type and this configuration is not possible. Thus the new basis will remain in configuration 1. So, the new point p_u replaces the sole point among p_i, p_j and p_k that will preserve the constraint: p_{u-n} lies inside the new triangle relative to the Oxy plane. As p_{u-n} is under the plane P defined by $p_i p_j p_k$, the current thickness e has also increased (see Fig. 3.1a). In the other case where $n < u \leq 2n$, two possibilities can appear. When p_{u-n} lies inside the triangle, it simply replaces its equivalent point p_l and e increases. When p_{u-n} lies outside, we cannot achieve a configuration of type 1, thus we move to a configuration of type 2. For this, the segment that supports P' is also $p_{u-n}p_{l-n}$. The other segment corresponds to the sole edge of the triangle that crosses this segment relative to the Oxy plane. p_{u-n} lies at a vertical distance greater than the one defined by the triangle and p_{l-n}. Moreover, this distance is equal to the distance between the two retained segments, so the new configuration increases the value of e (see Fig. 3.1b). In the **Configuration 2**, when a variable λ_u, $1 \leq u \leq n$ is selected, we have two possibilities. To remain in the same configuration, p_u must replace a point in such a way that the two new segments cross each other relative to the Oxy plane (see Fig. 3.2a). When this is not attainable, one of the two points p_{k-n} or p_{l-n} inevitably belongs to the triangle $p_i p_j p_u$ and we then shift to a configuration of type 1 (see Fig. 3.2b). Other interchangings can be deduced from the ones exposed in this section.

Convergence and complexity. As the Primal is feasible (choose a large value for e), the Dual is never unbounded and we can suppress the processing of this particular case. As we select a point outside of two parallel planes, we know that the vertical distance (the objective function) strictly increases at each iteration. Thus, unlike in the general case, the simplex algorithm applied to this recognition problem can not cycle. Moreover, we have at most $C_{2n+5}^4 = \mathcal{O}(n^4)$ possible feasible basis. Thus, we obtain an $\mathcal{O}(k^4)$ time complexity where k represents the number of the vertices of the convex hull of the given points. In practice, this quantity is relatively small compared to the number of points.

The incremental version. When a new point is inserted, it may lie between

the two planes P and P'. In this case, the previous solution remains optimal and nothing has to be done. Otherwise, two columns are added to the matrix A in the Dual. Next, using the last processed feasible basis, we launch a new sequence of iterations until the new optimum solution is found.

5   Conclusion

We proposed in this paper a new definition of discrete primitives: the blurred pieces of discrete planes. These discrete primitives allow to deal with the noise present in discrete data by varying a parameter. Two recognition algorithms are given. The first one is a geometric algorithm, based on the convex hull of the considered set of points and its result is the optimal bounding plane for which the Euclidean distance is minimal. The second one is based on the simplex algorithm and its output corresponds to the optimal bounding plane for which the vertical distance is minimal. The codes of these two algorithms and examples of use are available on `http://www.loria.fr/~debled/BlurredPlane`. A work about the comparison between these two methods is in progress. Moreover we intend to use these algorithms in the framework of the boundary segmentation of 3D noisy discrete objects. Our aim is to obtain an algorithm of polyhedrization of 3D noisy discrete objects by controlling the approximations done.

References

1. Klette, R., Rosenfeld, A.: Digital Geometry. Morgan Kaufmann publishers, Elsevier (2004)
2. Debled-Rennesson, I., Rémy, J.L., Rouyer-Degli, J.: Linear segmentation of discrete curves into fuzzy segments. Discrete Applied Math. **151** (2005) 122–137
3. Debled-Rennesson, I., Feschet, F., Rouyer-Degli, J.: Optimal blurred segments decomposition of noisy shapes in linear time. Computers & Graphics **30**(1) (2006)
4. Reveillès, J.: Géométrie discrète, calculs en nombres entiers et algorithmique. Thèse d'Etat – Université Louis Pasteur (1991)
5. Buzer, L.: An elementary algorithm for digital line recognition in the general case. In: DGCI-2005. Number 3429 in LNCS, Springer-Verlag (2005) 299–310
6. Debled-Rennesson, I., Tabbone, S., Wendling, L.: Multiorder polygonal approximation of digital curves. Electronic Letters on Computer Vision and Image Analysis **5**(2) (2005) 98–110 Special Issue on Document Analysis.
7. Andres, E.: Le plan discret. In: Colloque de géométrie discrète en imagerie: fondements et applications, Strasbourg, France (1993)
8. Houle, M., Toussaint, G.: Computing the width of a set. IEEE Trans. on Pattern Analysis and Machine Intelligence **10**(5) (1988) 761–765
9. Gärtner, B., Herrmann, T.: Computing the width of a point set in 3-space. J. Exp. Algorithmics **4** (2001)   3
10. Toussaint, G.: Solving geometric problems with the rotating calipers. In: Proceedings of IEEE MELECON'83, Athens, Greece (1983)
11. de Berg, M., van Kreveld, M., Overmars, M., Schwarzkopf, O.: Computational Geometry : Algorithms and Applications. Springer-Verlag, Heidelberg (2000)
12. Chvatal, V.: Linear Programming. Freeman, New York (1983)