Geometric measures on arbitrary dimensional digital surfaces

Jacques-Olivier Lachaud, Anne Vialard

LaBRI (Laboratoire Bordelais de Recherche en Informatique) - France

Outline

1- 3D example

2- nD digital surface: Definition
Tracking
Contours

3- Discrete tangent line: Definition
Recognition

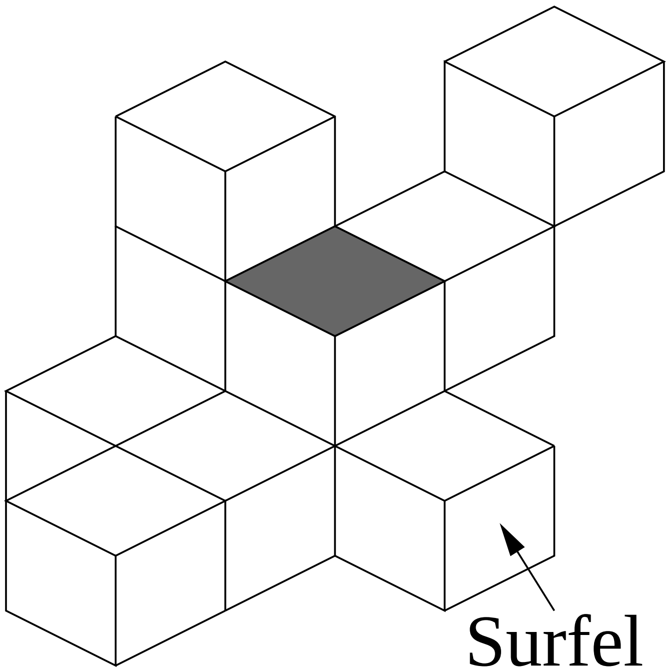
4- Geometric estimators: Normal vector
Tangent planes
Area

3D example

Problem: how to compute the normal vector to each element of a digital surface ?

[Lenoir96], *[Tellier / Debled-Renesson99]*, *[Coeurjolly02]*

3D digital surface: boundary of a voxel set

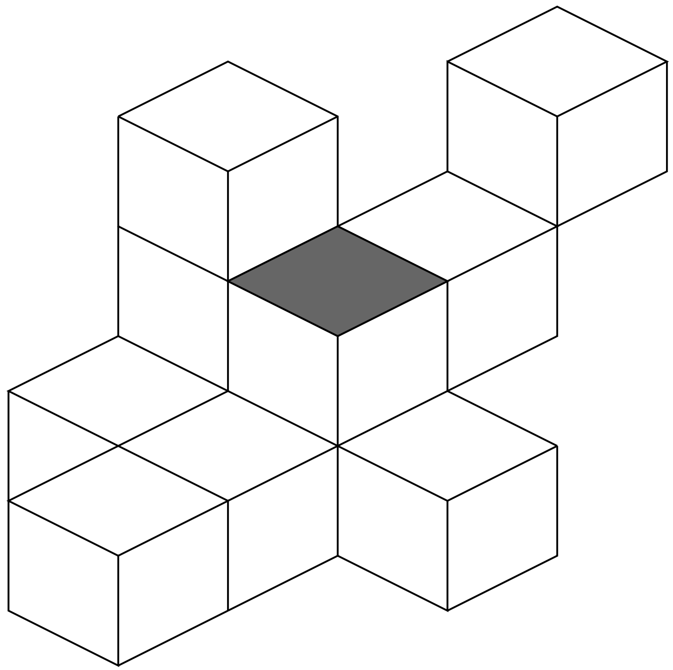


3D example

Problem: how to compute the normal vector to each element of a digital surface ?

[Lenoir96], *[Tellier / Debled-Renesson99]*, *[Coeurjolly02]*

3D digital surface: boundary of a voxel set



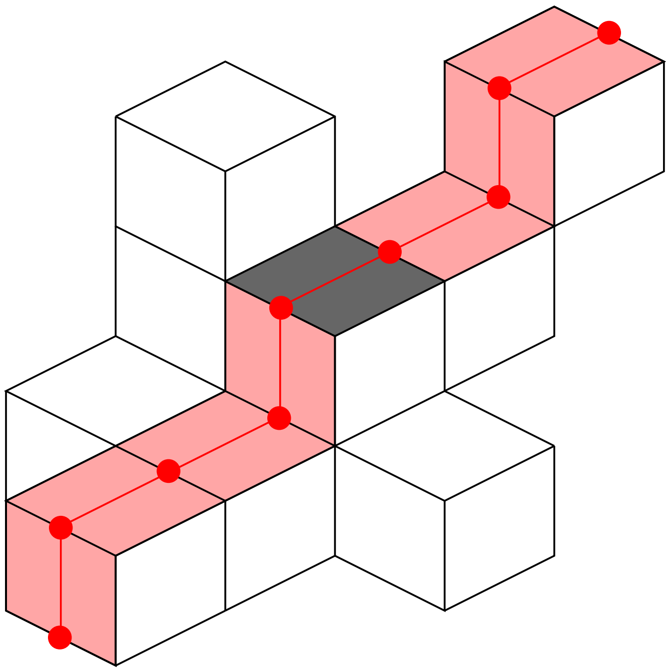
(1) Exactly two 4-connected contours cross at a given surfel

3D example

Problem: how to compute the normal vector to each element of a digital surface ?

[Lenoir96], *[Tellier / Debled-Renesson99]*, *[Coeurjolly02]*

3D digital surface: boundary of a voxel set



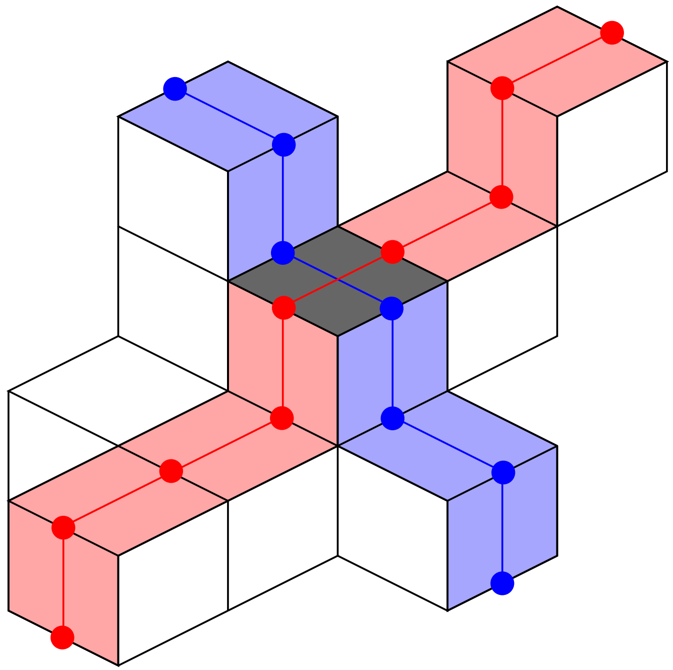
(1) Exactly two 4-connected contours cross at a given surfel

3D example

Problem: how to compute the normal vector to each element of a digital surface ?

[Lenoir96], *[Tellier / Debled-Renesson99]*, *[Coeurjolly02]*

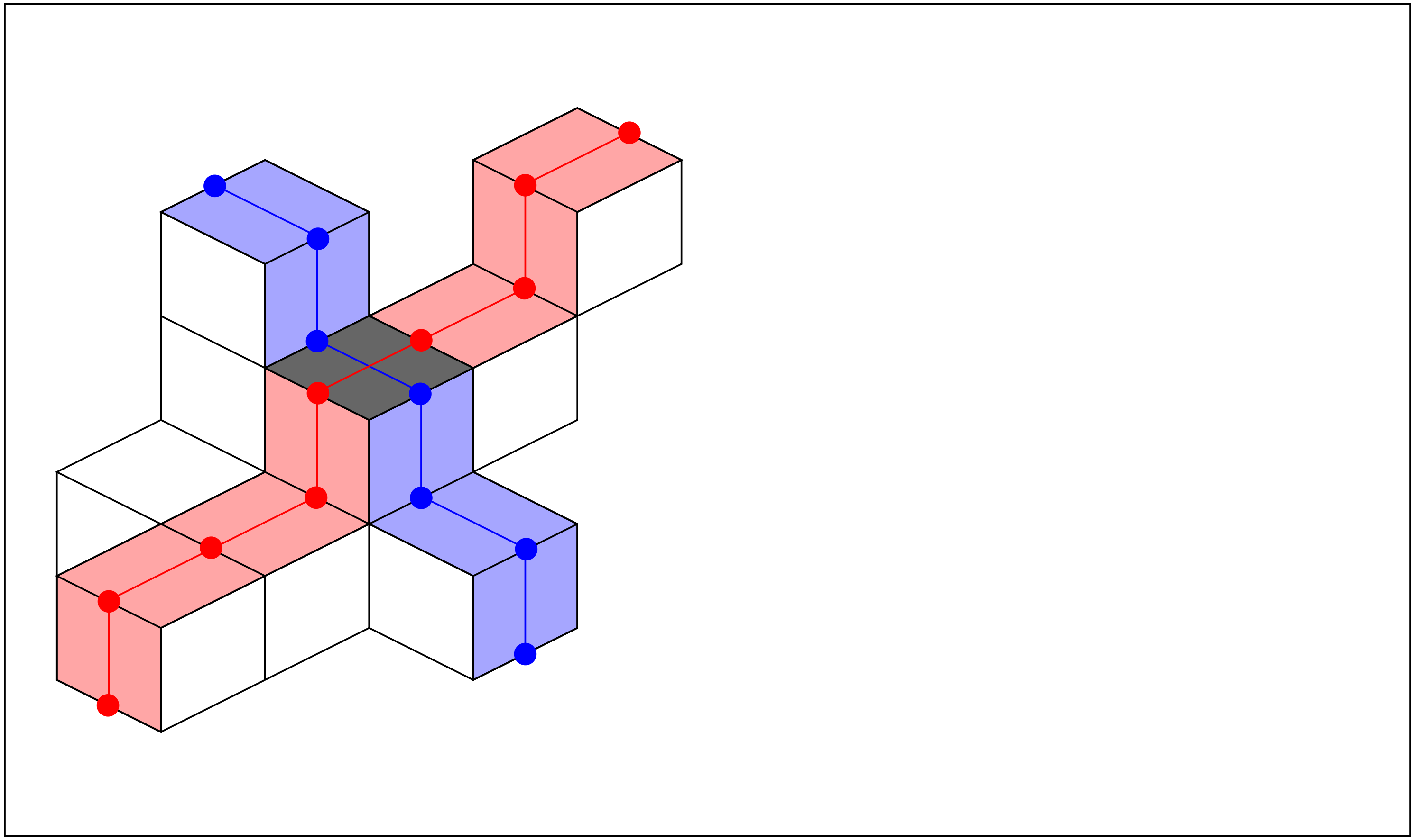
3D digital surface: boundary of a voxel set



(1) Exactly two 4-connected contours cross at a given surfel

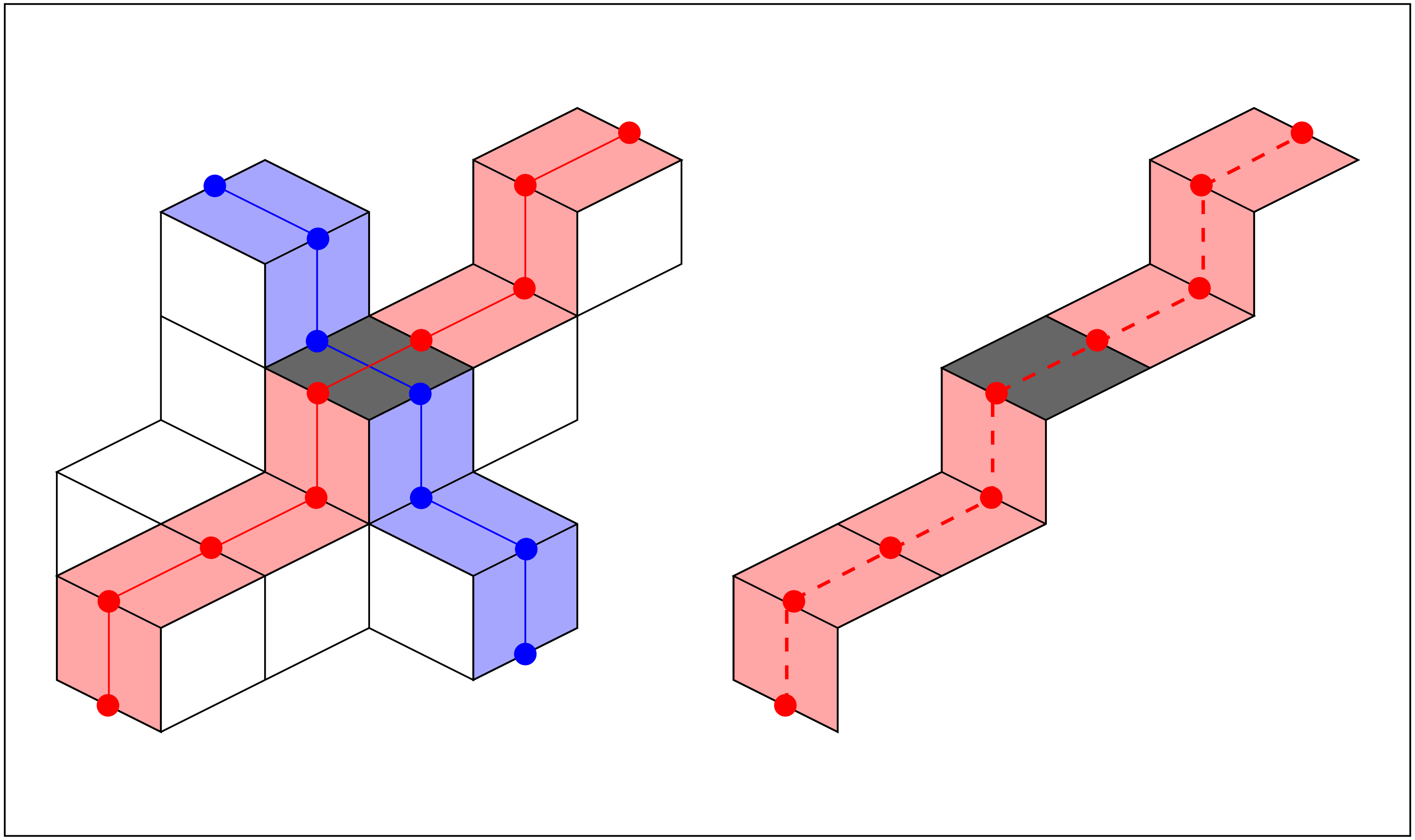
3D example

(2) Tangent computation on each 2D contour: discrete line segment recognition *[Debled95]*, *[Vialard96]*



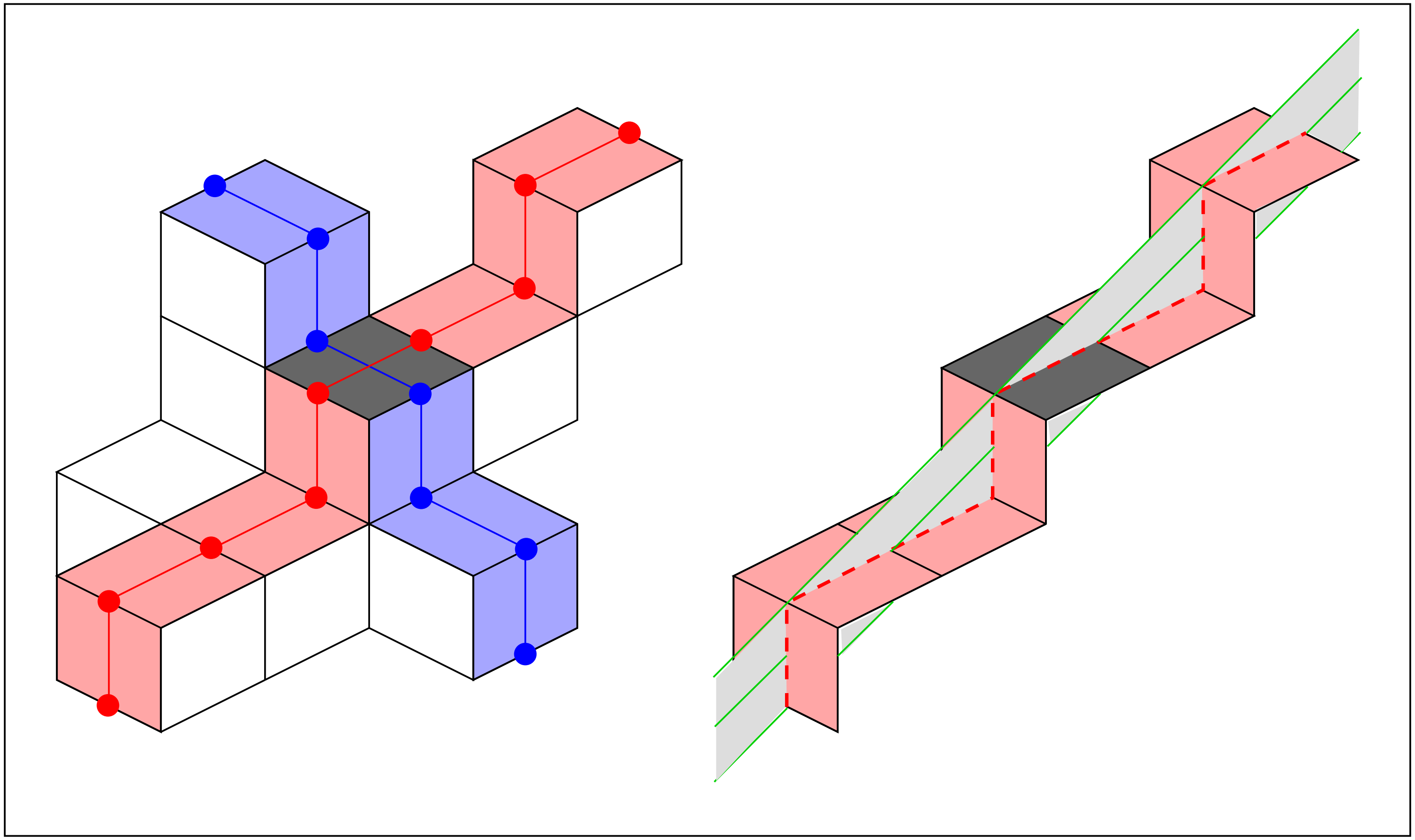
3D example

(2) Tangent computation on each 2D contour: discrete line segment recognition *[Debled95]*, *[Vialard96]*



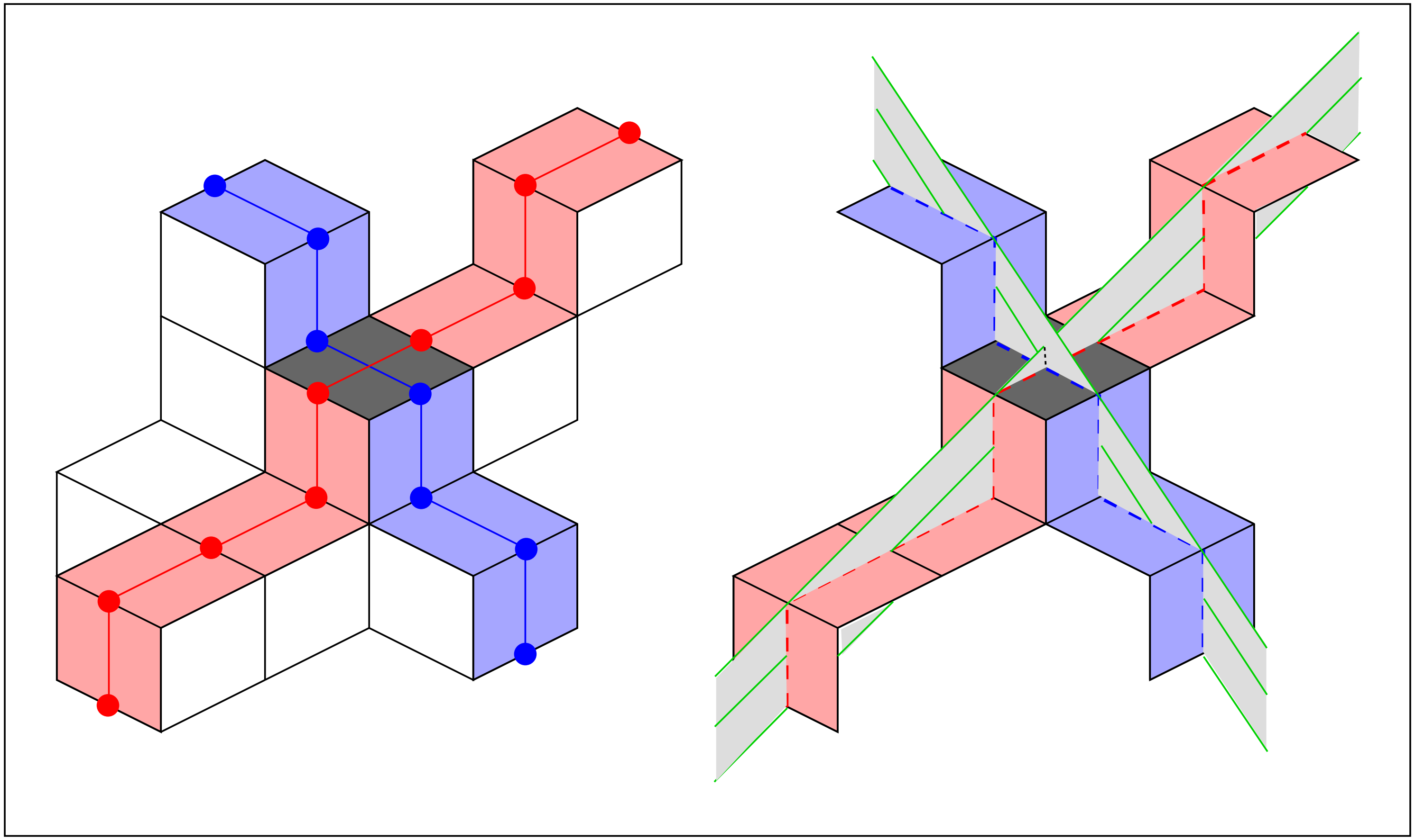
3D example

(2) Tangent computation on each 2D contour: discrete line segment recognition *[Debled95]*, *[Vialard96]*



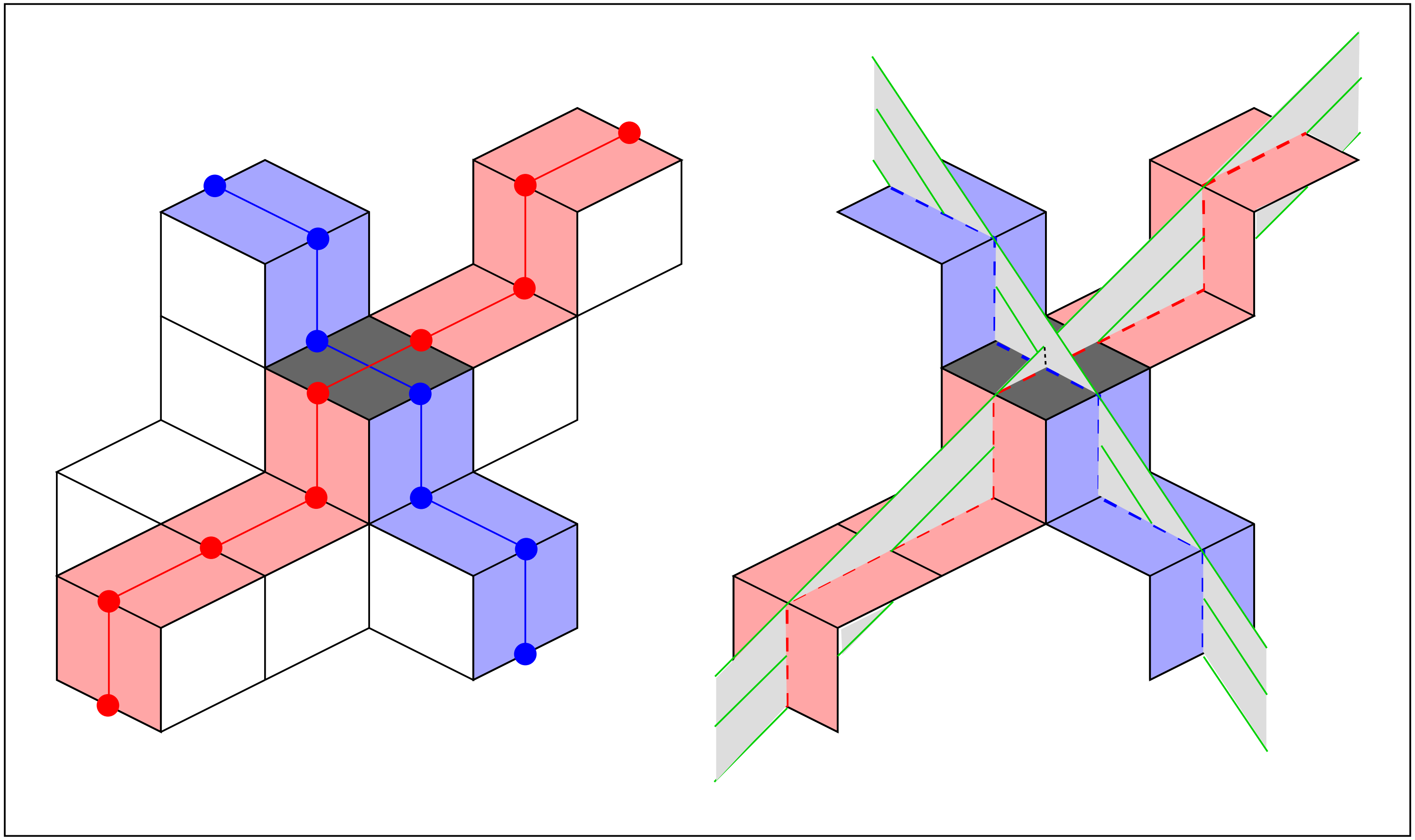
3D example

(2) Tangent computation on each 2D contour: discrete line segment recognition *[Debled95]*, *[Vialard96]*



3D example

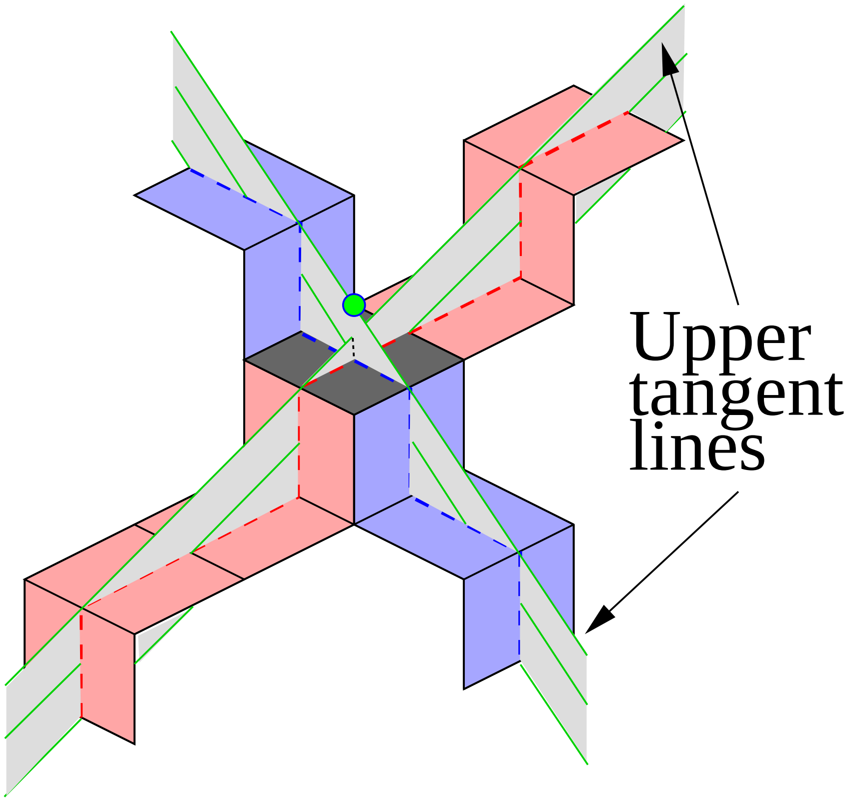
(2) Tangent computation on each 2D contour: discrete line segment recognition *[Debled95]*, *[Vialard96]*

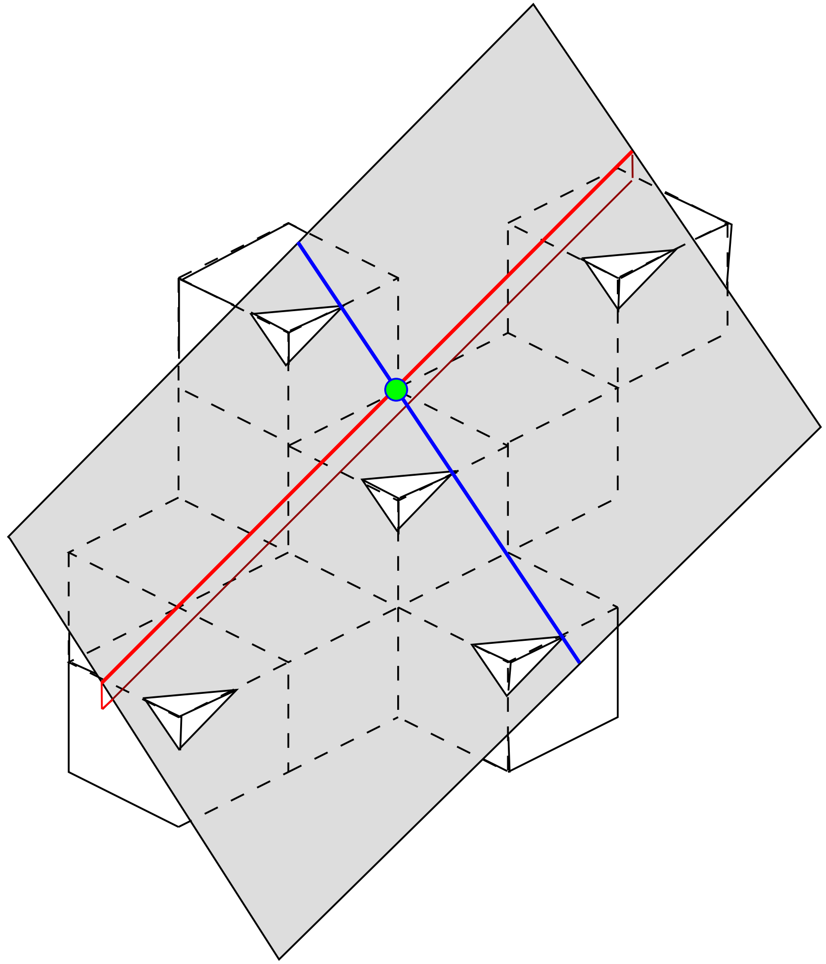


Normal vector: cross product of the two tangent vectors

3D example

(3) *Outer tangent plane:* orthogonal to the normal vector and containing P
P is the projection of the surfel centroid on the upper tangent lines (highest point).





Outline

1- 3D example

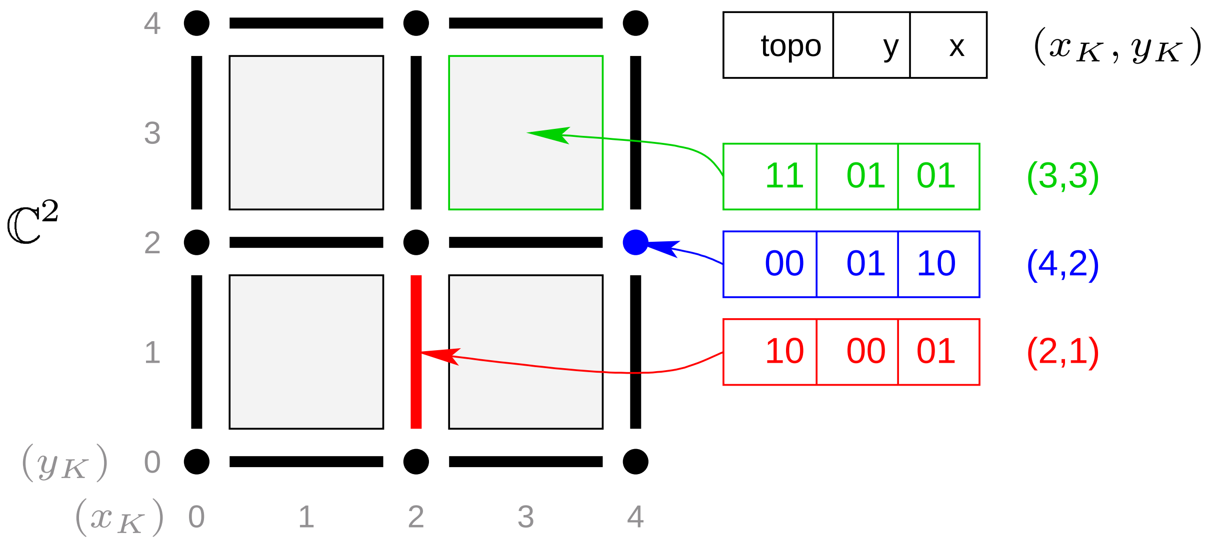
2- Multidimensional digital surface: Definition
Tracking
Contours

3- Discrete tangent line: Definition
Recognition

4- Geometric estimators: Normal vector
Tangent planes
Area

Multidimensional digital surface

Digital space: cellular decomposition of $\mathbb{R}^n$ into a regular grid *[Khalimsky90]*, *[Kovalevsky89]*, *[Herman92]*, *[Udupa94]*



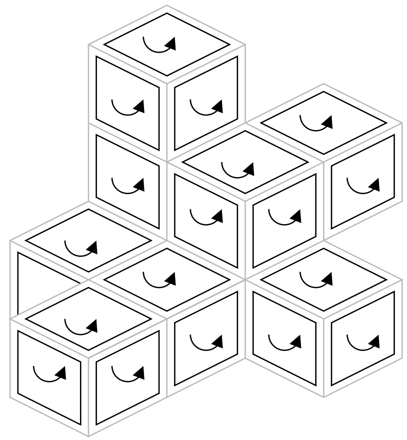
Spel: n-cell (pixel in 2D, voxel in 3D)

Surfel: $n-1$-cell

Digital surface: set of oriented surfels

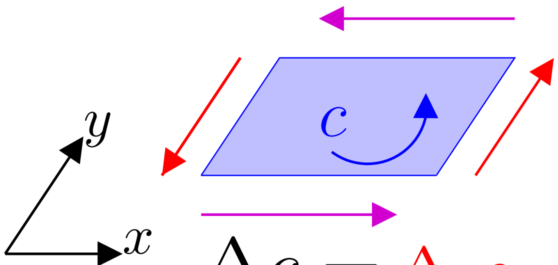
Object boundary

Object O: set of spels

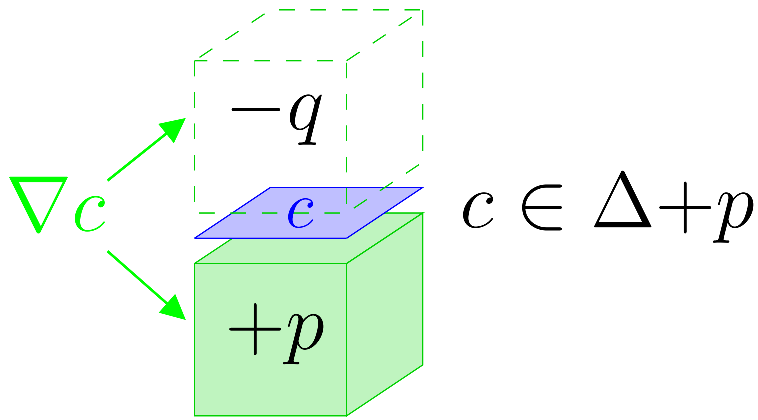


Boundary of O: surfels separating spels of O from the background with an orientation

Boundary / Coboundary operators:



$$\Delta c = \Delta_x c \cup \Delta_y c$$

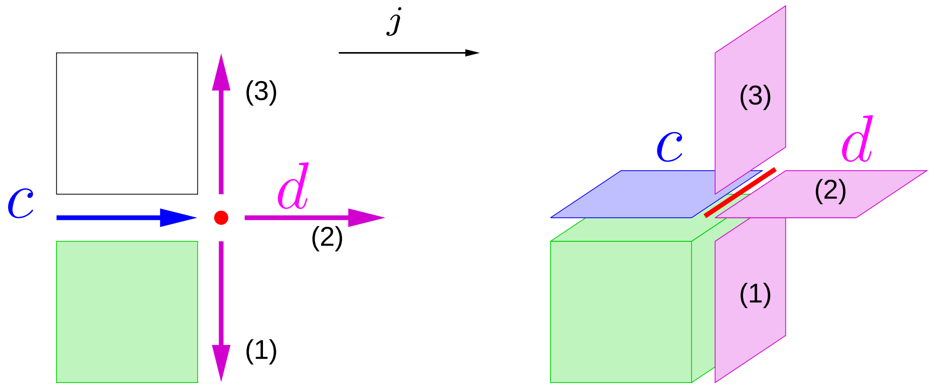


$$\boxed{\partial O = \bigcup \Delta{+p}, p \in O}$$

Boundary tracking

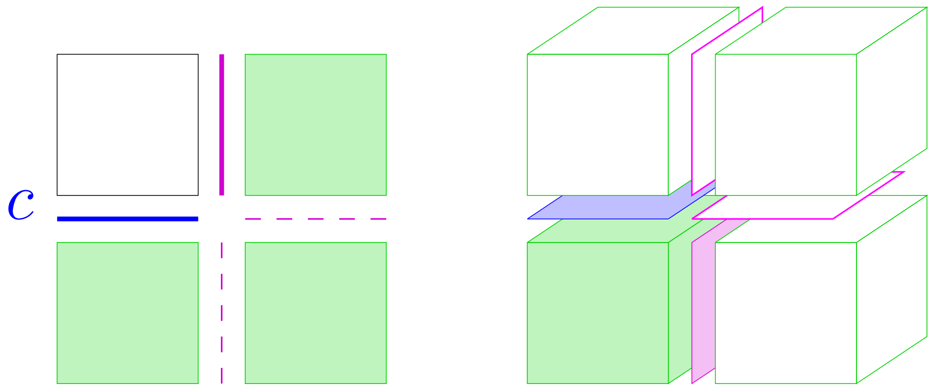
Given a bel of O, find ∂O by following the surface of the object $\Rightarrow$ adjacency between bels

Direct followers of a bel along coordinate j:



Link l: $\begin{array}{l} +l \in \Delta c \\ -l \in \Delta d \end{array}$

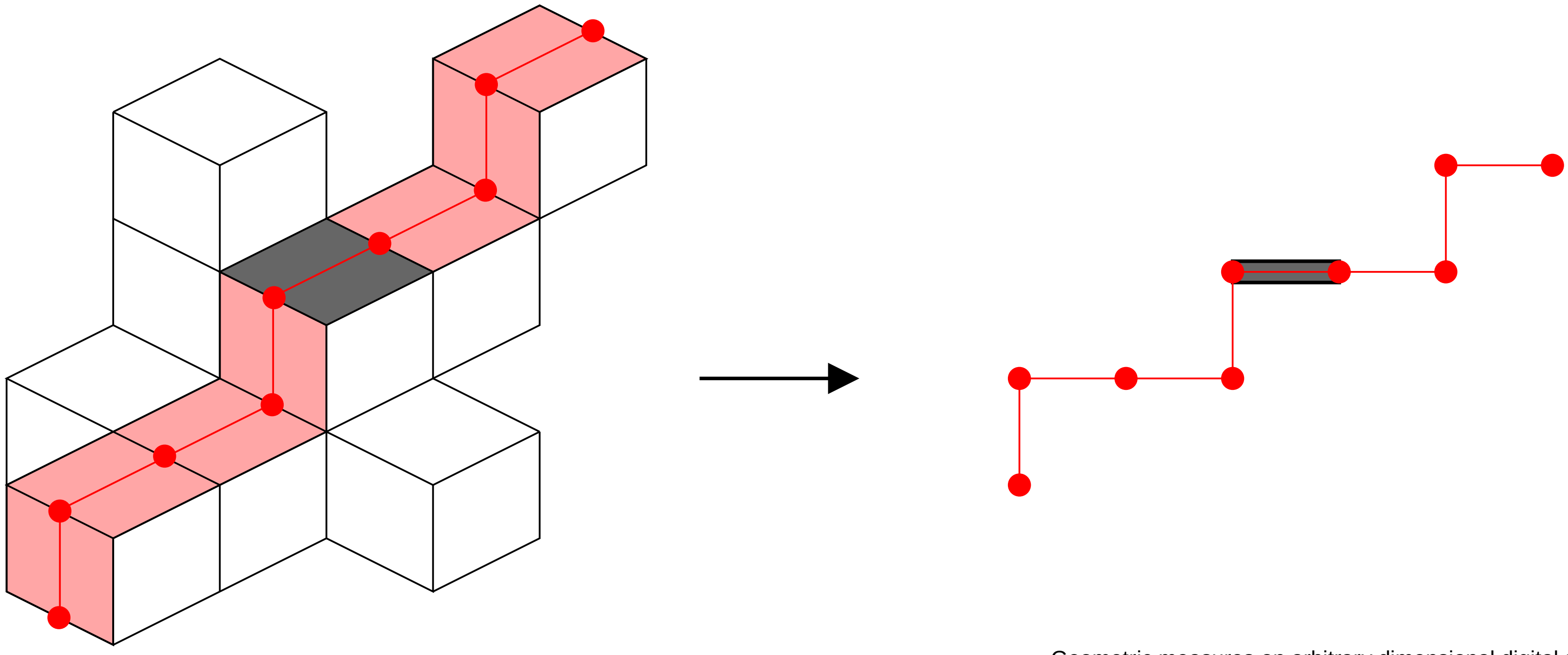
Interior direct adjacent bel: the first direct follower that is a bel of ∂O



Contour over a surface

Given a bel c of ∂O and $j \neq \perp(c)$, a *contour over the boundary* is the sequence of direct interior adjacent bels starting from c and going along directions $\perp(c)$ or j.

Such a contour is a 2D 4-connected discrete path (bels $\rightarrow$ edges, links $\rightarrow$ points).



Outline

1- 3D example

2- nD digital surface: Definition
Tracking
Contours

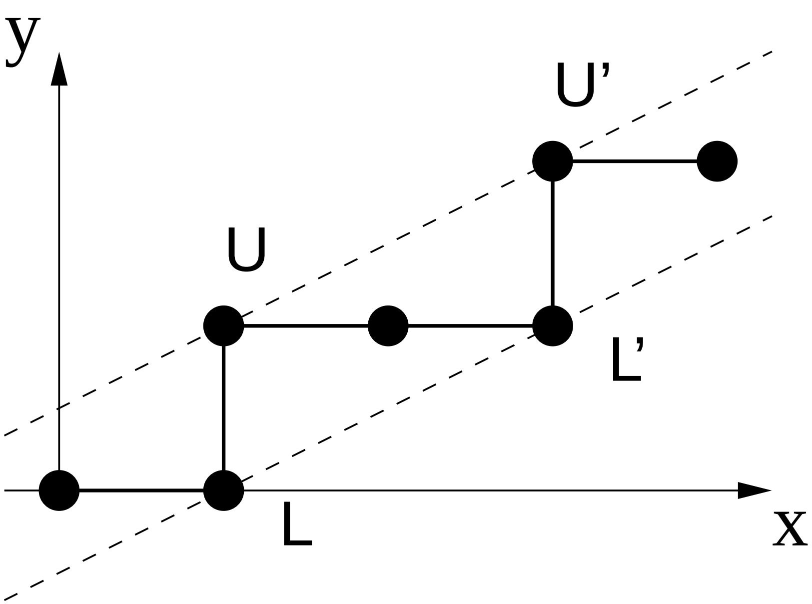
3- Discrete tangent line: Definition
Recognition

4- Geometric estimators: Normal vector
Tangent planes
Area

Discrete 2D tangent line

4-connected discrete line of characteristics $(a, b, \mu) \in \mathbb{Z}^3$:

$$\{(x, y) \in \mathbb{Z}^2, \mu \leq ax - by < \mu + |a| + |b|\}$$



$$(a, b, \mu) = (1, 2, -1)$$

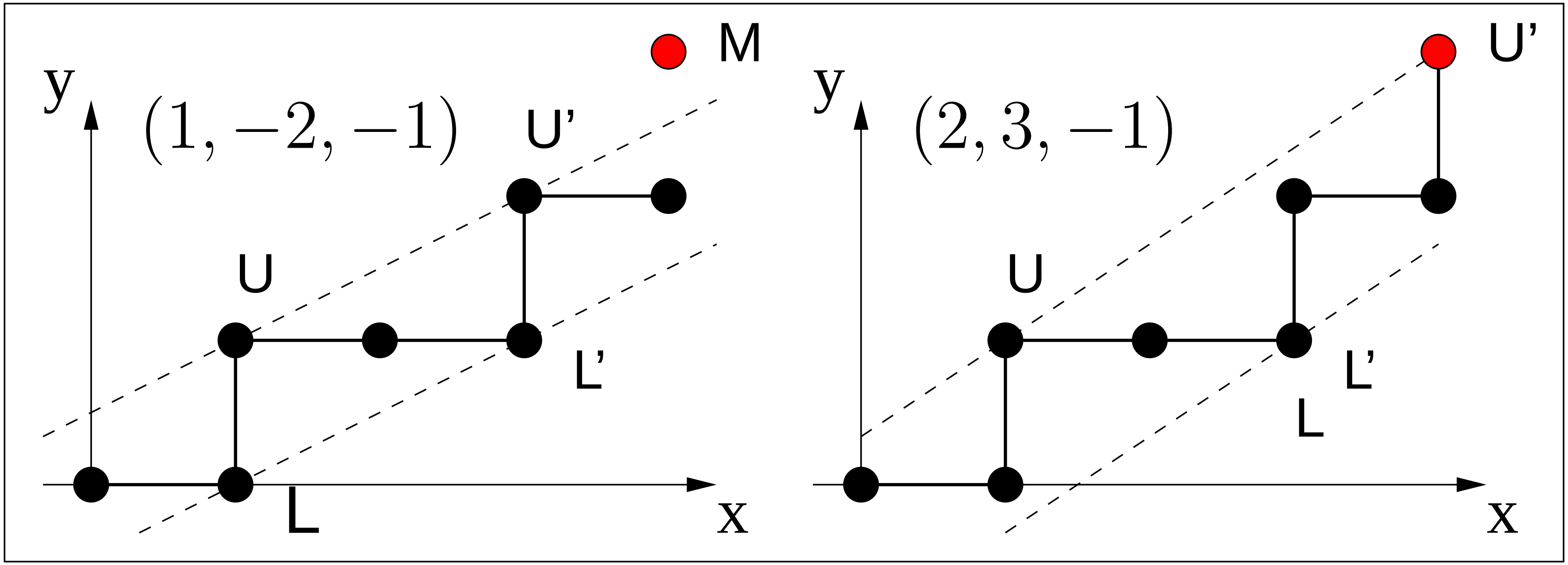
Leaning lines: $ax - by = \mu$, $ax - by = \mu + |a| + |b| - 1$

Discrete 2D tangent line

Update of a segment line when adding a point M:

(1) M is in between the leaning lines: OK

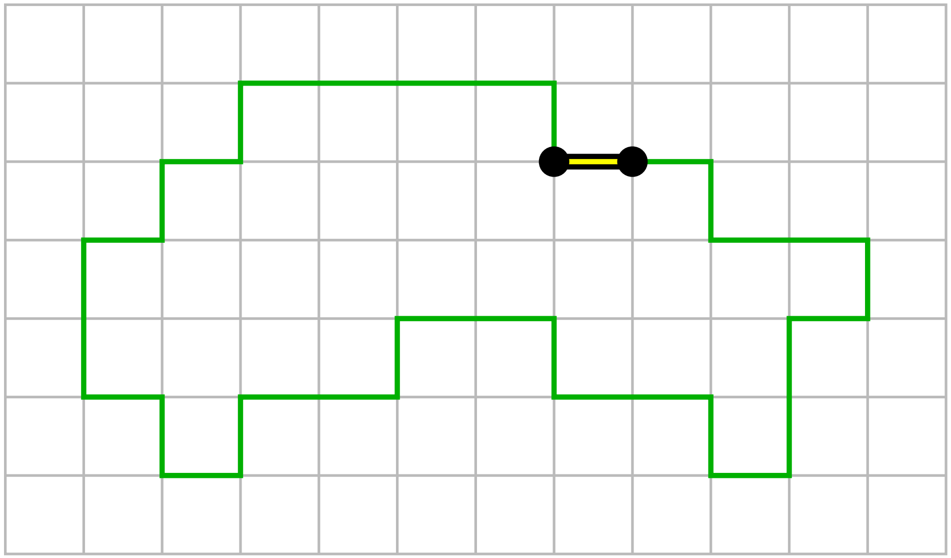
(2) $ax_M - by_M = \mu - 1$: M is just over the upper leaning line



(3) M is just under the lower leaning line: similar

Discrete 2D tangent line

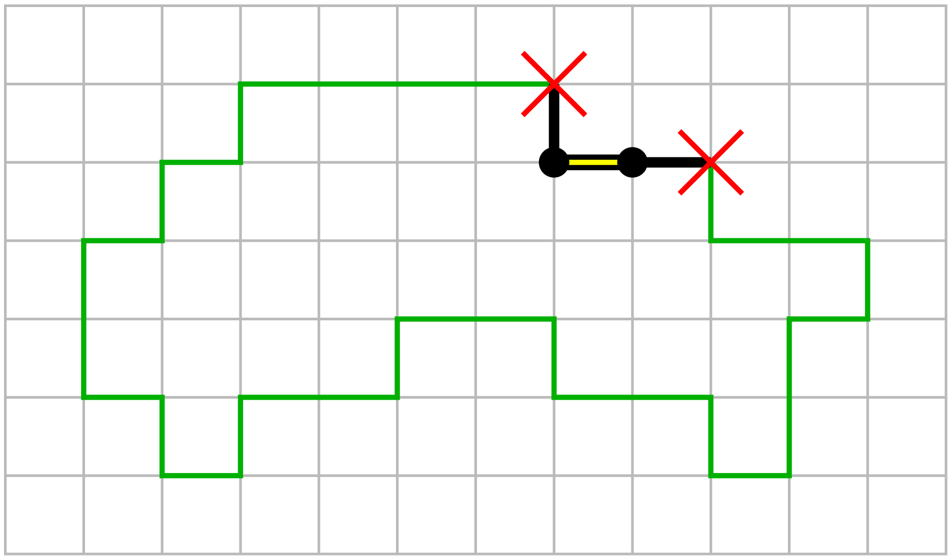
Symmetric tangent centered on an edge



$$(a, b, \mu) = (0, 1, 0)$$

Discrete 2D tangent line

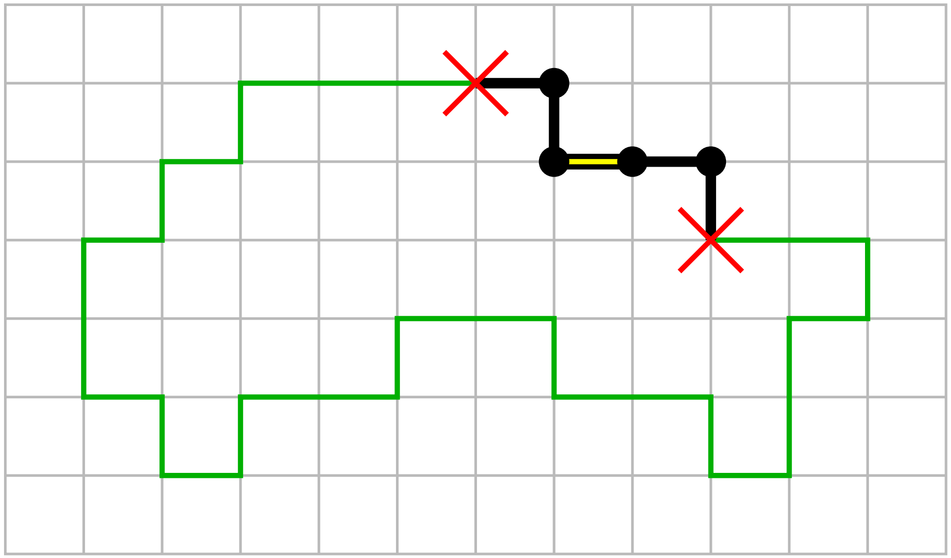
Symmetric tangent centered on an edge



$$(a, b, \mu) = (-1, 2, -2)$$

Discrete 2D tangent line

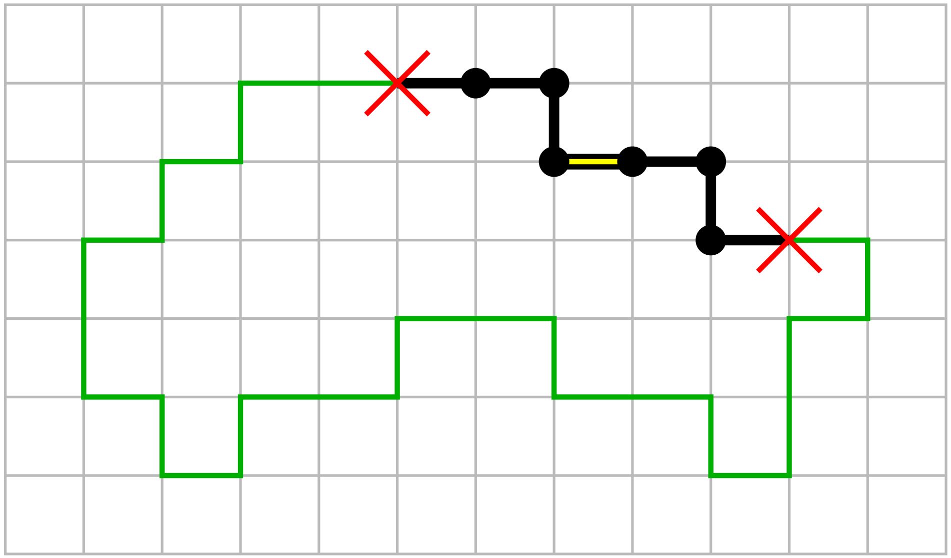
Symmetric tangent centered on an edge



$$(a, b, \mu) = (-1, 2, -2)$$

Discrete 2D tangent line

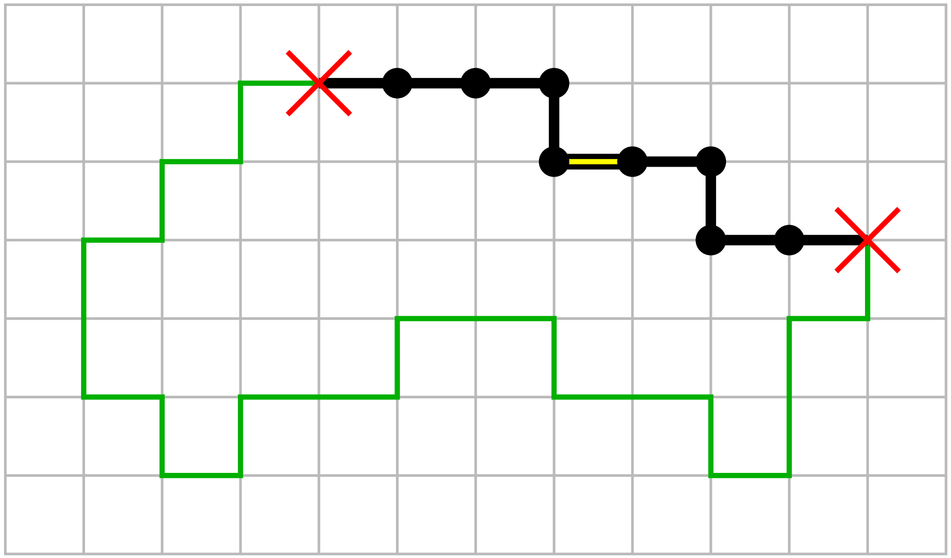
Symmetric tangent centered on an edge



$$(a, b, \mu) = (-1, 2, -2)$$

Discrete 2D tangent line

Symmetric tangent centered on an edge



$$(a, b, \mu) = (-2, 5, -5)$$

Outline

1- 3D example

2- nD digital surface: Definition
Tracking
Contours

3- Discrete tangent line: Definition
Recognition

4- Geometric estimators: Normal vector
Tangent planes
Area

Normal vector at a bel

Normal vector at bel c: unit vector orthogonal to the $n-1$ tangent vectors at c.

$i = \perp(c)$

τ_i: orientation of the cell of $\nabla_i c$ with greatest x_i

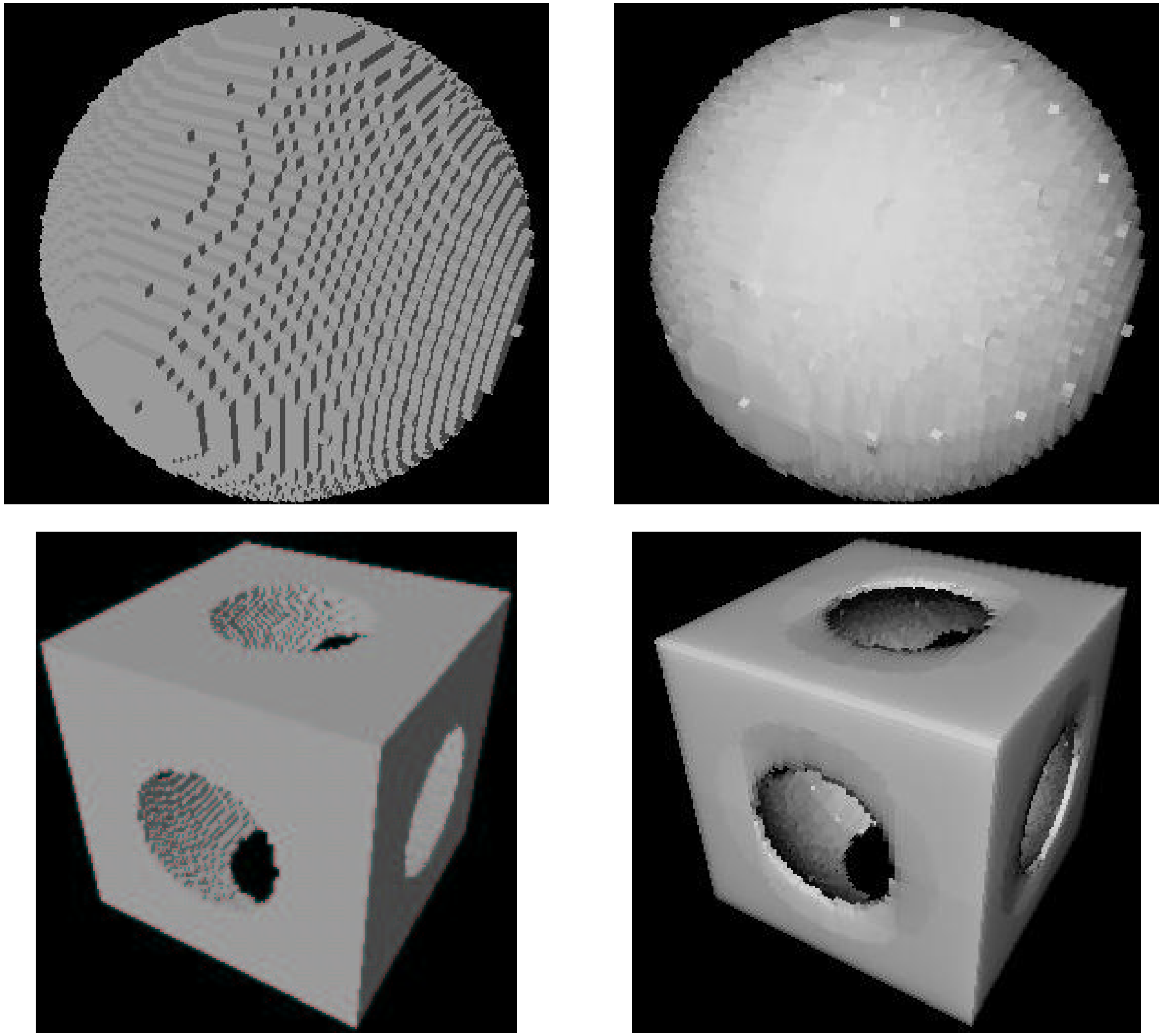
τ_j: orientation of the cell of $\Delta_j c$ with greatest x_j

$\vec{n}(c) = \frac{\vec{u}(c)}{\|\vec{u}(c)\|}$

$$\forall j \neq i, \vec{u}(c) \cdot \vec{e}_j = \tau_j \frac{\alpha_j(c)}{\beta_j(c)}$$

$$\vec{u}(c) \cdot \vec{e}_i = \tau_i$$

Visualization of 3D discrete objects



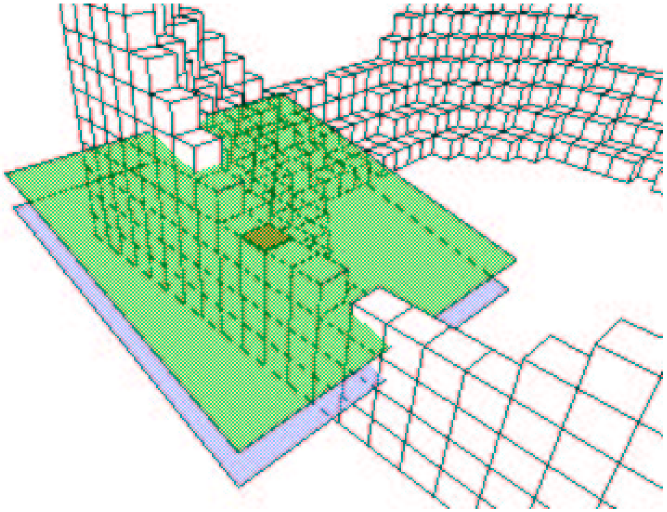
Tangent planes to an nD surface

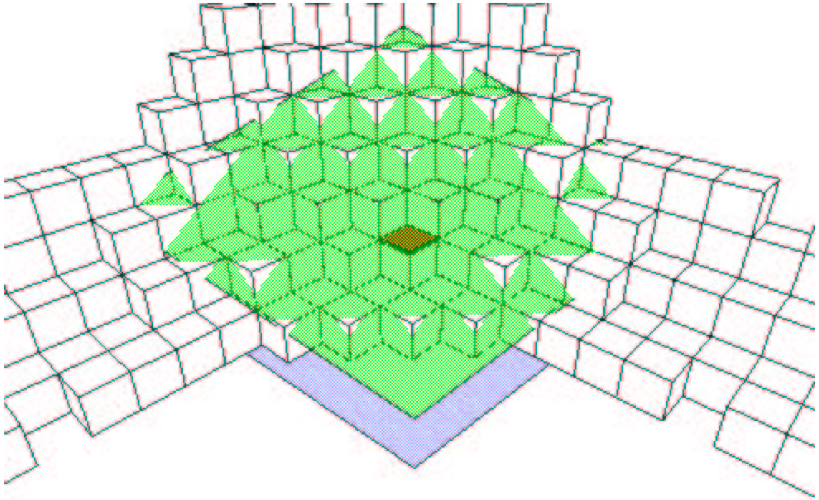
Centroid of c: $\vec{x}_c$

On each contour: $z_j^+ = \frac{0.5\alpha_j - \mu_j}{\beta_j}$, $z_j^- = z_j^+ - 1 - \frac{|\alpha_j| - 1}{\beta_j}$

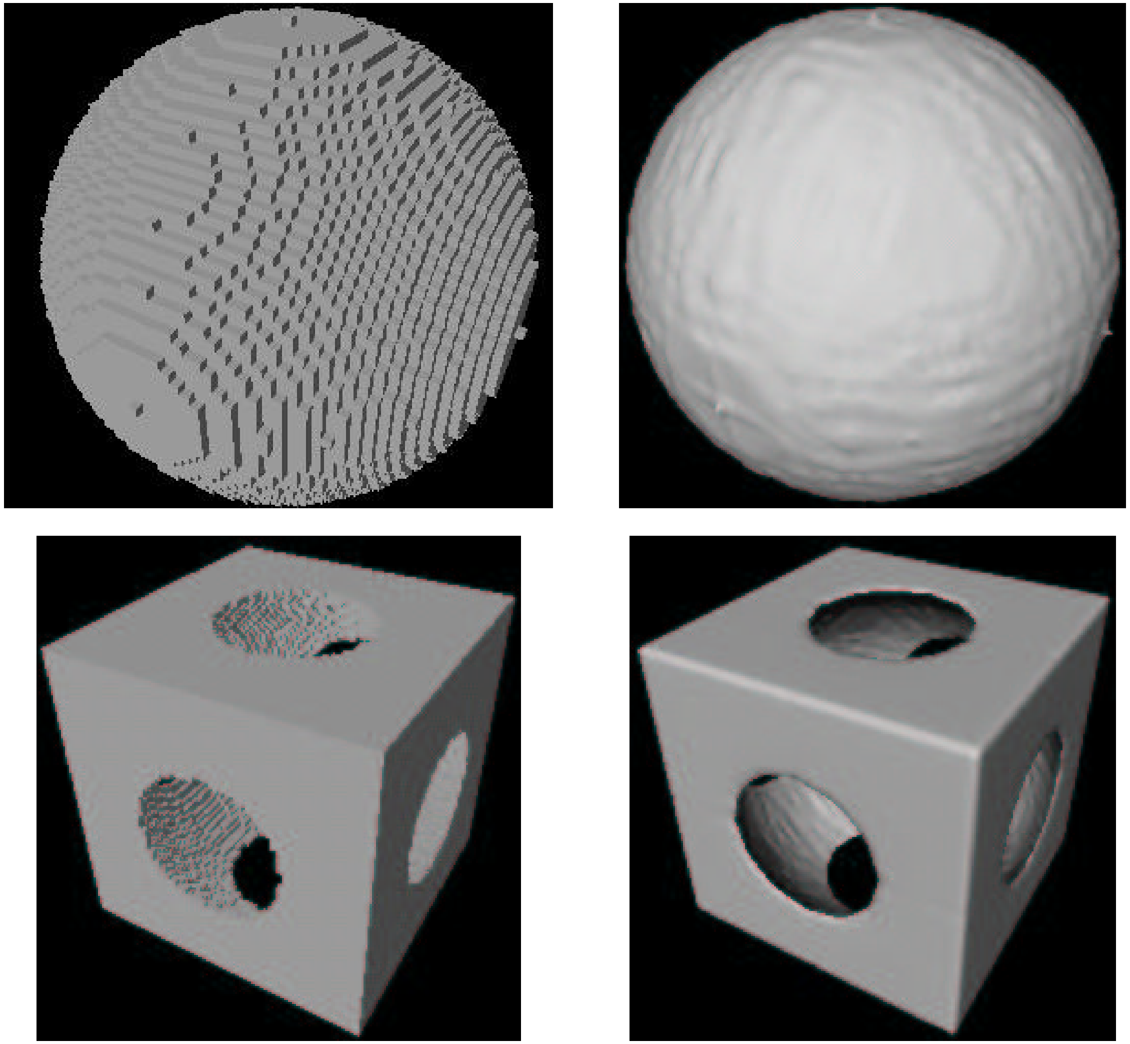
The inner tangent plane passes through

$$\vec{x}_c + \tau_i \vec{e}_i \max_{j \neq i} z_j^+$$





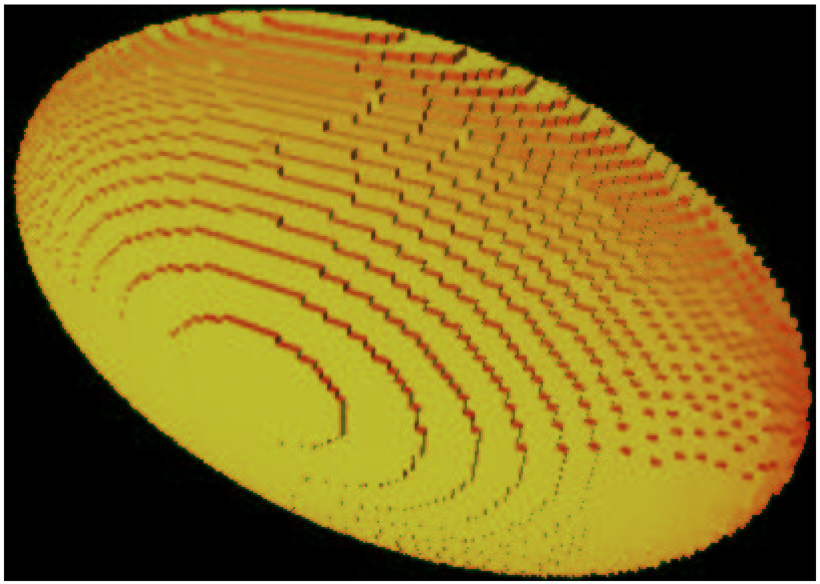
Smoothing of 3D digital surfaces

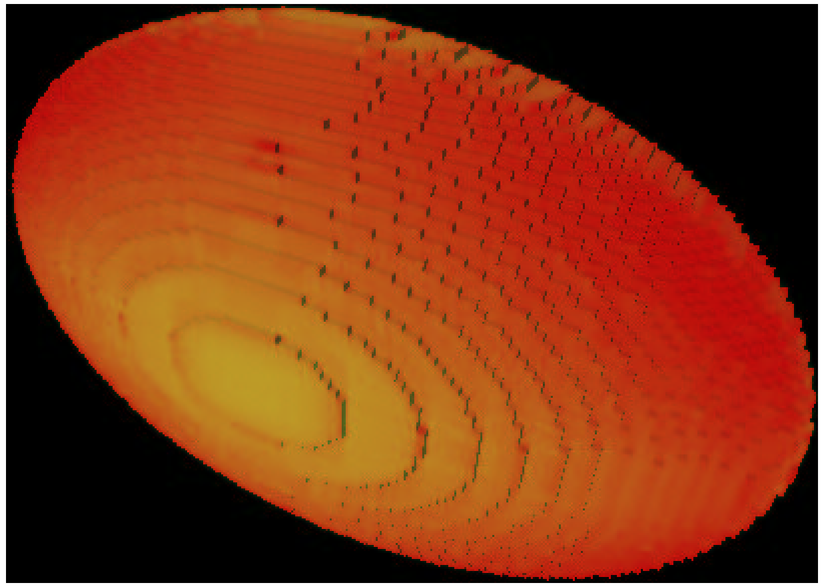


Area of an nD digital surface

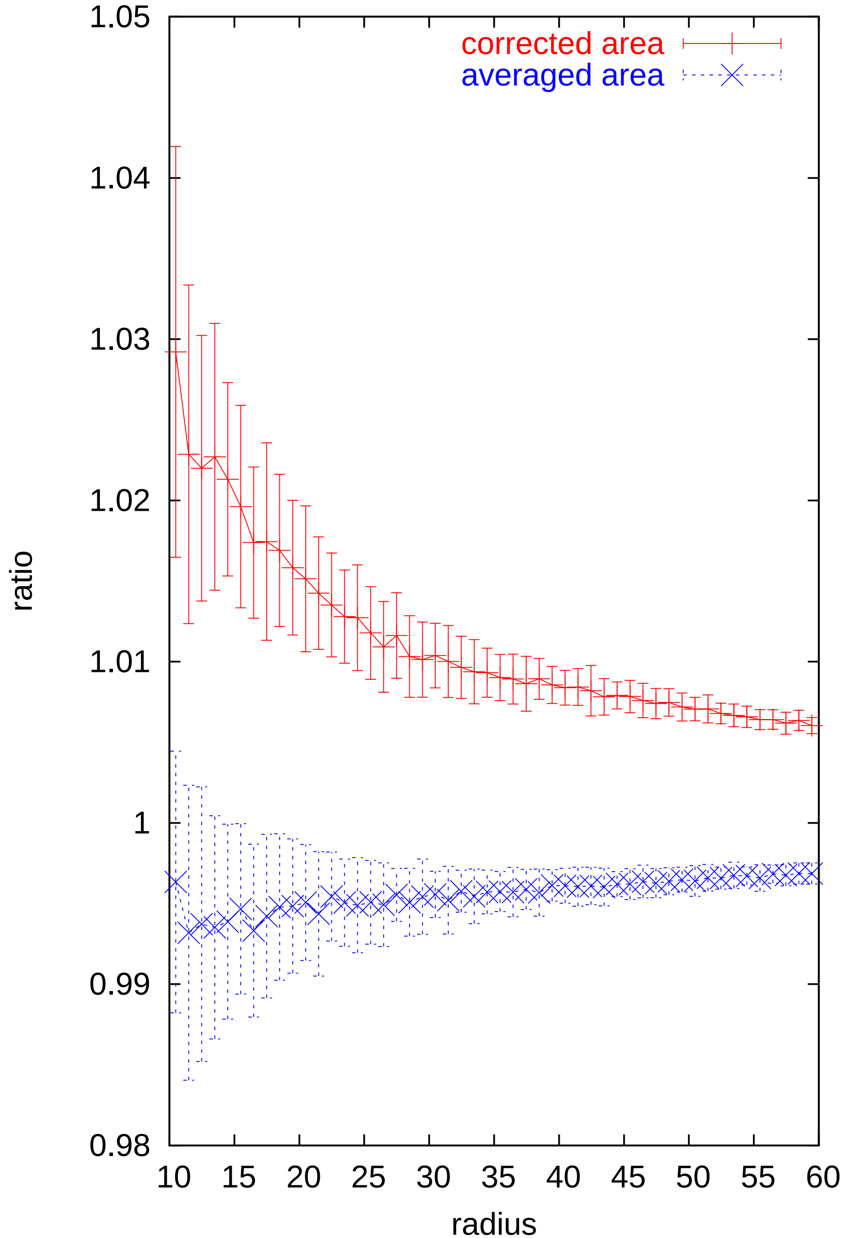
Corrected area: $\widehat{d\sigma}(c) = \vec{n}(c) \cdot \vec{e}_i$

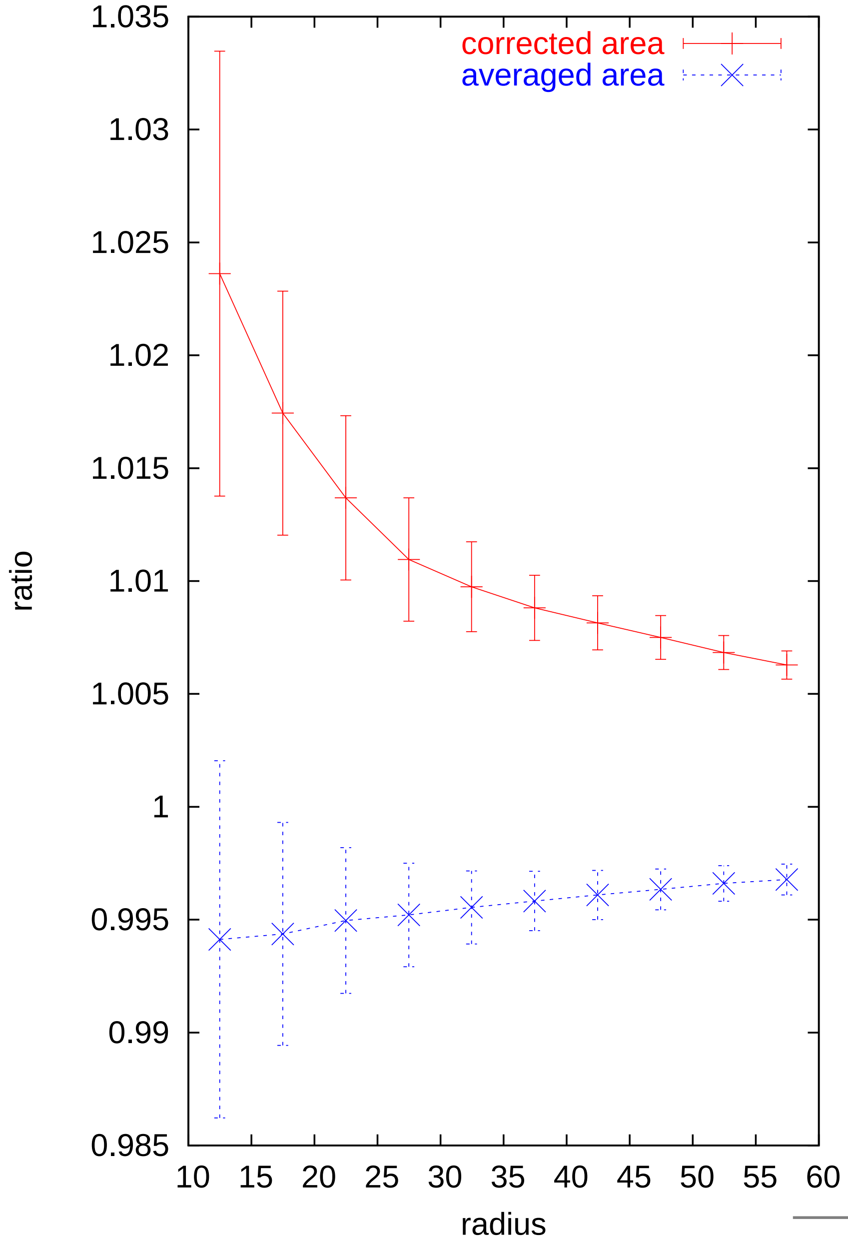
Averaged area: $\overline{d\sigma}(c) = 1/(\sum_{k=0}^{n-1} |\vec{n}(c) \cdot \vec{e}_k|)$





Area of a 3D sphere





Conclusion

Results:

- Definition of a set of geometric estimators for multidimensional surfaces
- Efficient and generic implementation *[Lachaud03]*
- Convergence of the estimators to the continuous values

Further works:

- Can we compute the normal vector field in a time linear with the number of surfels ?
- Curvature definition and computation