

Reconnaissance des formes par transformée de Hough



Bruno VALLET

LaSTIG - IGN

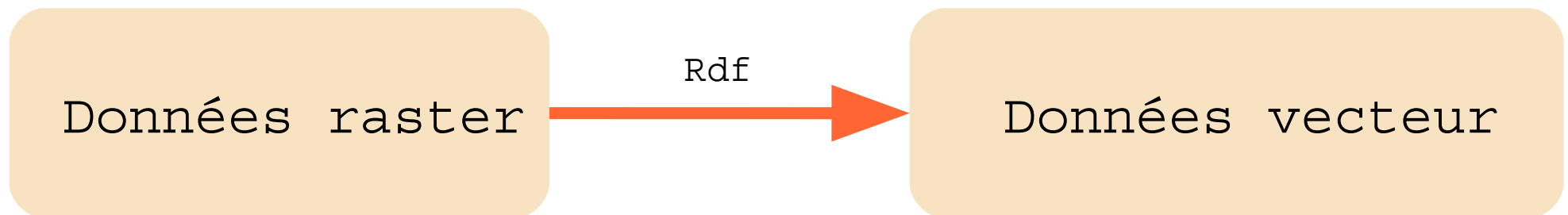
<https://www.umr-lastig.fr/bruno-vallet/>
bruno.vallet@ign.fr



ÉCOLE NATIONALE
DES SCIENCES
GÉOGRAPHIQUES

ENSG
Ingénieur 2

Le problème de la RdF:

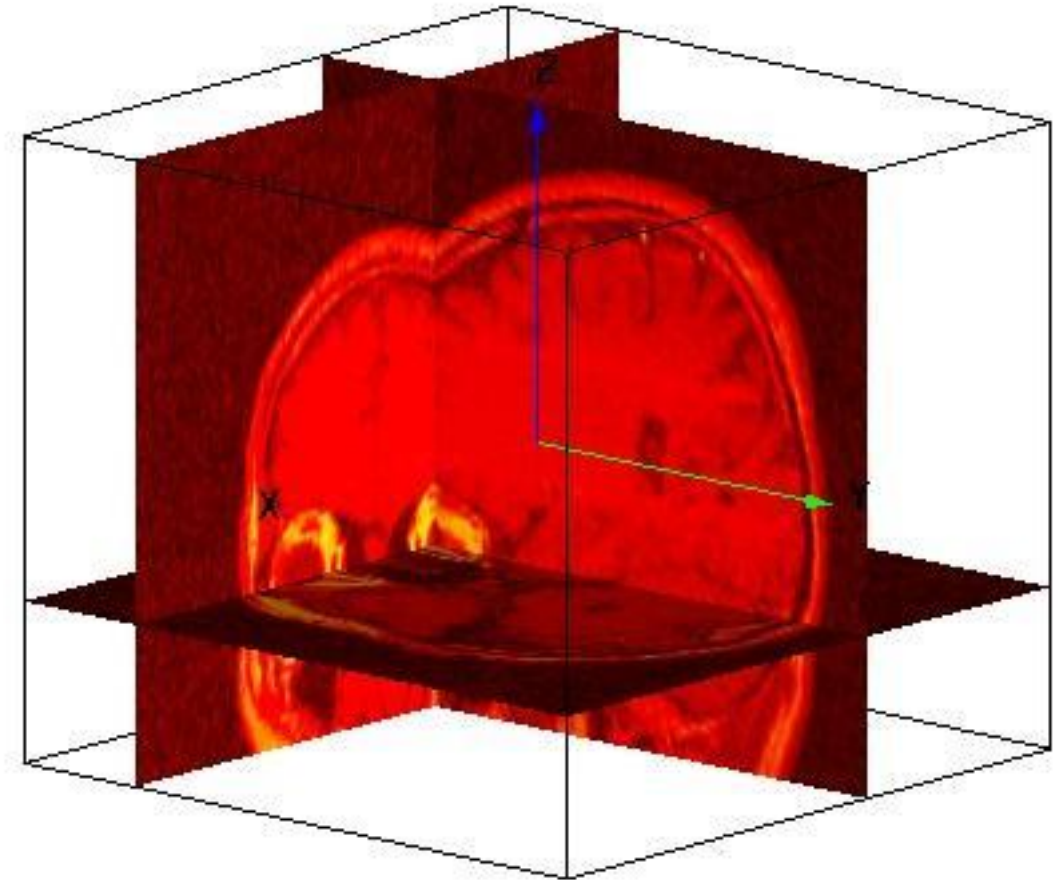


Données raster

- Données échantillonnées sur une grille régulière



2D



3D

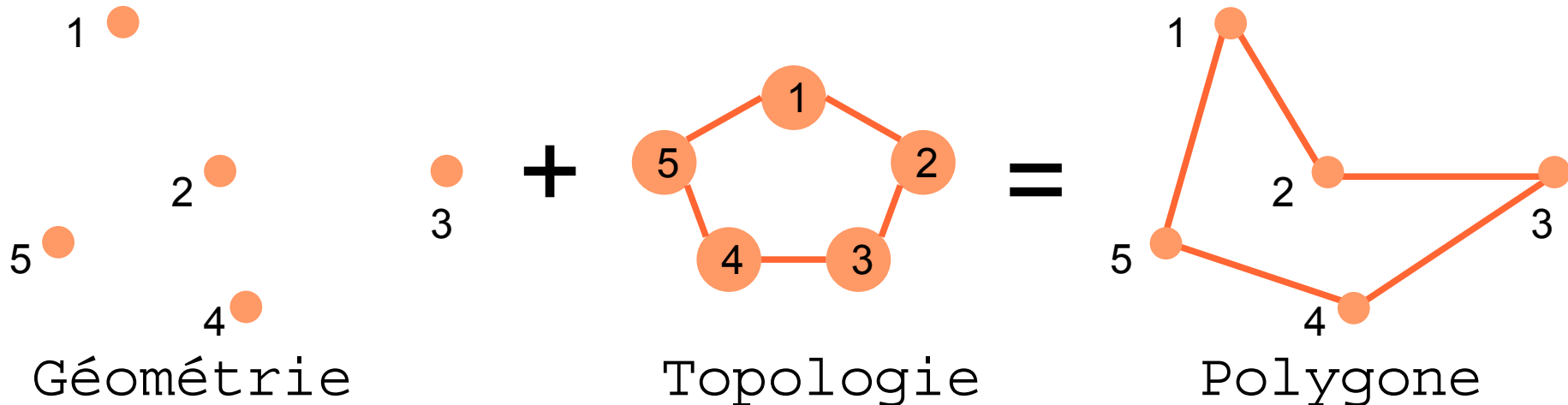
Données vecteur

Modèle structuré d'un objet contenant:

- De la géométrie
- De la topologie

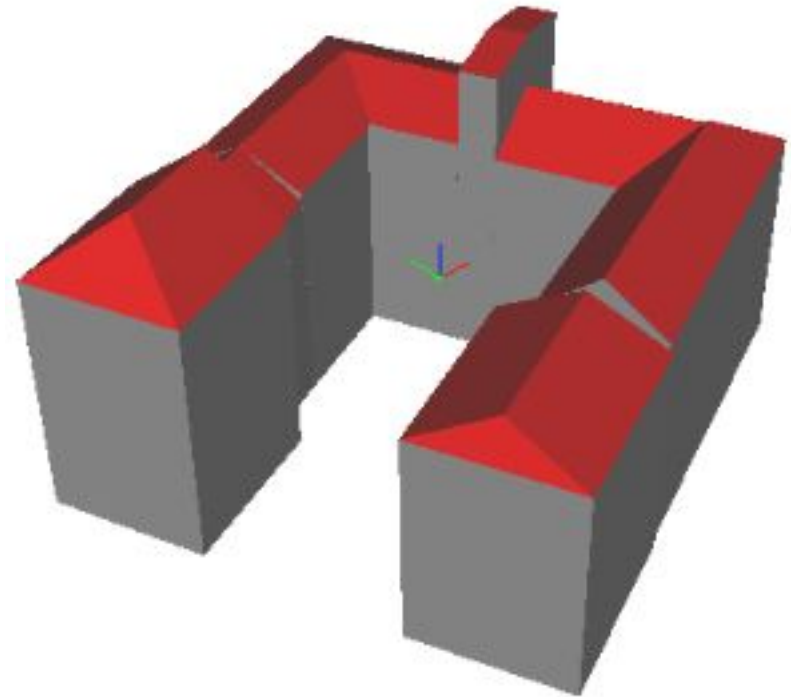
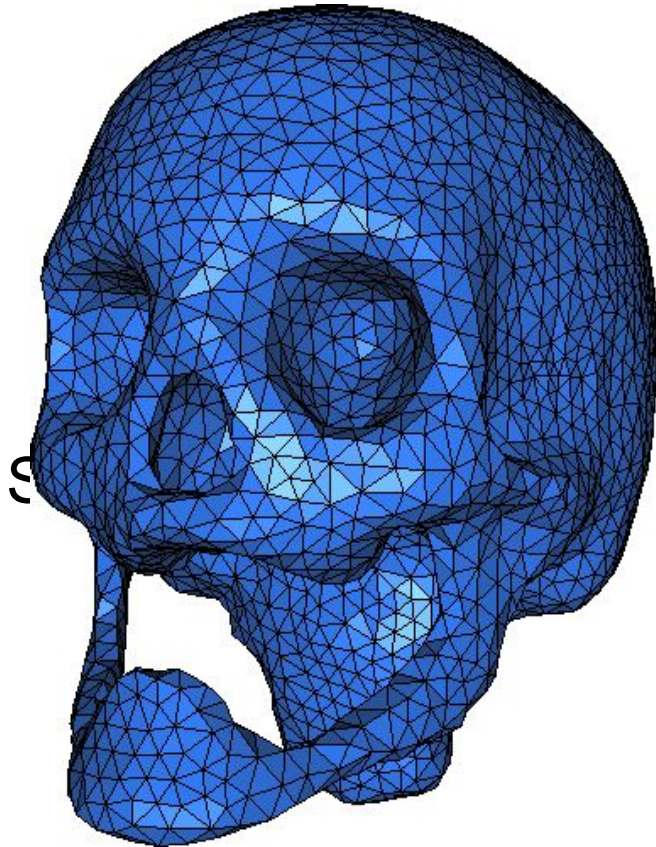
Ex: un polygone est défini par:

- Les coordonnées de ses points (géométrie)
- L'ordre de ses points (topologie)



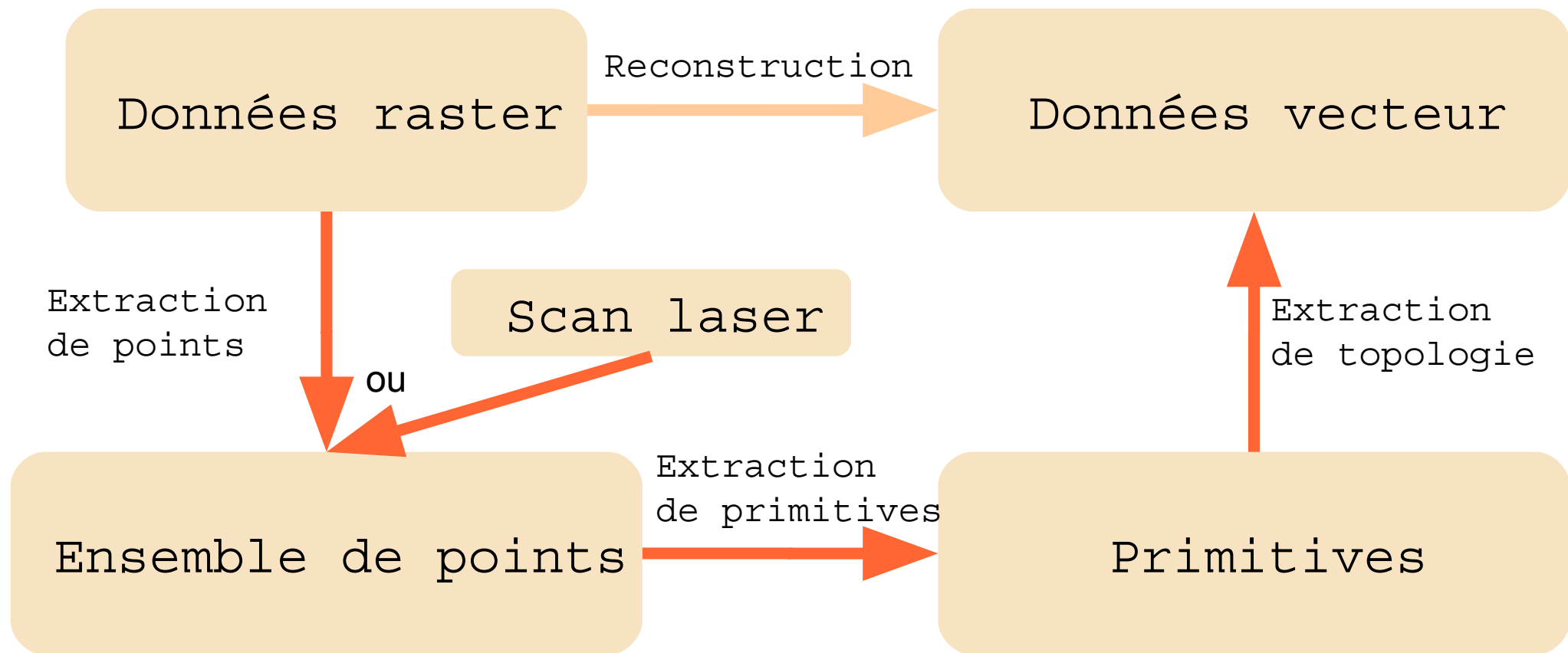
Données vecteur

- Exemples



Le problème de la reconstruction

- 3 étapes



Le problème de l'extraction de points

Données raster



Ensemble de points

- Extraction de points d'intérêt:

Image



Points d'intérêt

- Extraction de contours: (cf. Cours Extraction de contours)

Image



Points de contours

- Corrélation éparses:

Images orientées



Nuage de points 3D

- Corrélation dense:

Images orientées



MNE

Primitives

- Données continues décrites par une équation mathématique:
 - En 2D:
 - Droite
 - Segment
 - Cercle
 - Ellipse
 - Triangle
 - Rectangle
 - Carré
 - Spline
 - En 3D:
 - Plan
 - Triangle
 - Sphère
 - Droite
 - Spline
 - Quadrique

Elles dépendent d'un nombre limité de **paramètres**

Problème: détection de primitives multiples

Nuage de points

$$\mathcal{X} = \{\mathbf{x}_i = (x_i, y_i, z_i)^t \in \mathbb{R}^3\}_{i=1..n}$$

Primitives

$$\{\mathcal{P}(a_i, b_i, c_i, \dots)\}_{i=1..n}$$

Principe de la Reconnaissance de Formes:

- Extraire des objets paramétriques simples qui expliquent bien les points de données
- Les primitives doivent être proches des points expliqués
- Il doit y avoir le moins possible de points non expliqués

Problème: détection de primitives multiples

Nuage de points

$$\mathcal{X} = \{\mathbf{x}_i = (x_i, y_i, z_i)^t \in \mathbb{R}^3\}_{i=1..n}$$

Primitives

$$\{\mathcal{P}(a_i, b_i, c_i, \dots)\}_{i=1..n}$$

Principe de la Reconnaissance de Formes:

- Le nombre d'objets peut être
 - fixe (+ facile)
 - inconnu (+difficile): trouver le moins de primitives possible

Principe de la transformée de Hough

Nuage de points

$$\mathcal{X} = \{\mathbf{x}_i = (x_i, y_i, z_i)^t \in \mathbb{R}^3\}_{i=1..n}$$

Primitives

$$\{\mathcal{P}(a_i, b_i, c_i, \dots)\}_{i=1..n}$$

- Extraire des objets paramétriques simples en se projetant dans l'espace des paramètres de l'objet
- Accumuler dans l'espace des paramètres
- Les primitives cherchées correspondent aux maxima

Principe de la transformée de Hough

Nuage de points

$$\mathcal{X} = \{\mathbf{x}_i = (x_i, y_i, z_i)^t \in \mathbb{R}^3\}_{i=1..n}$$



Primitives

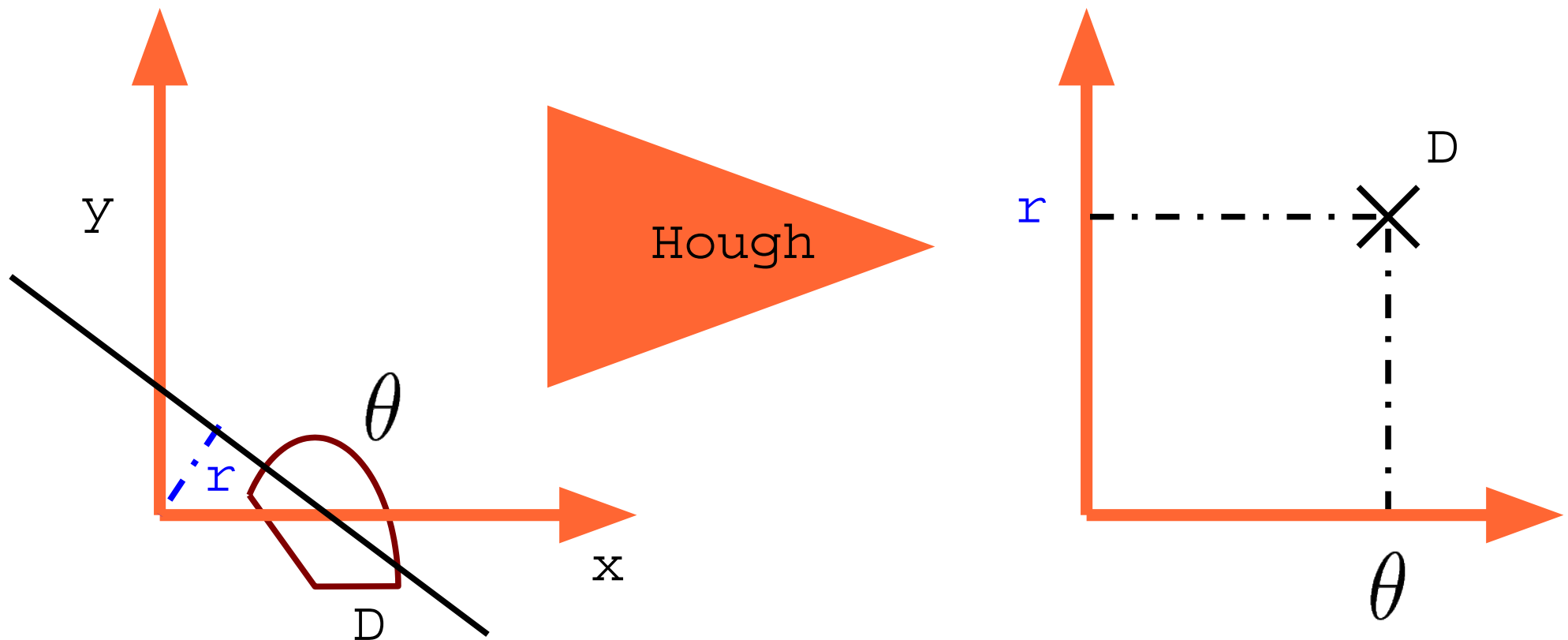
$$\{\mathcal{P}(a_i, b_i, c_i, \dots)\}_{i=1..n}$$

La primitive dépend de plusieurs paramètres :

- Droite : 2 $(r, \theta) : x \cos(\theta) + y \sin(\theta) - r = 0$
- Cercle : 3 $(x_0, y_0, r) : (x - x_0)^2 + (y - y_0)^2 - r^2 = 0$
- Plan : 4 $(r, \theta, \phi) : x \sin(\theta) \cos(\phi) + y \sin(\theta) \sin(\phi) + z \cos(\phi) - r = 0$
- Sphère : 4 $(x_0, y_0, z_0, r) : (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - r^2 = 0$
- Droite 3D: ???

Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Transformation d'une **droite** dans l'espace des paramètres



Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Transformation d'une **droite** dans l'espace des paramètres :
- Soient deux points de la droite $P_1 = (x_1, y_1)$ et $P_2 = (x_2, y_2)$
- Ils appartiennent à la droite donc :

$$x_1 \cos(\theta) - y_1 \sin(\theta) = r$$

$$x_2 \cos(\theta) - y_2 \sin(\theta) = r$$

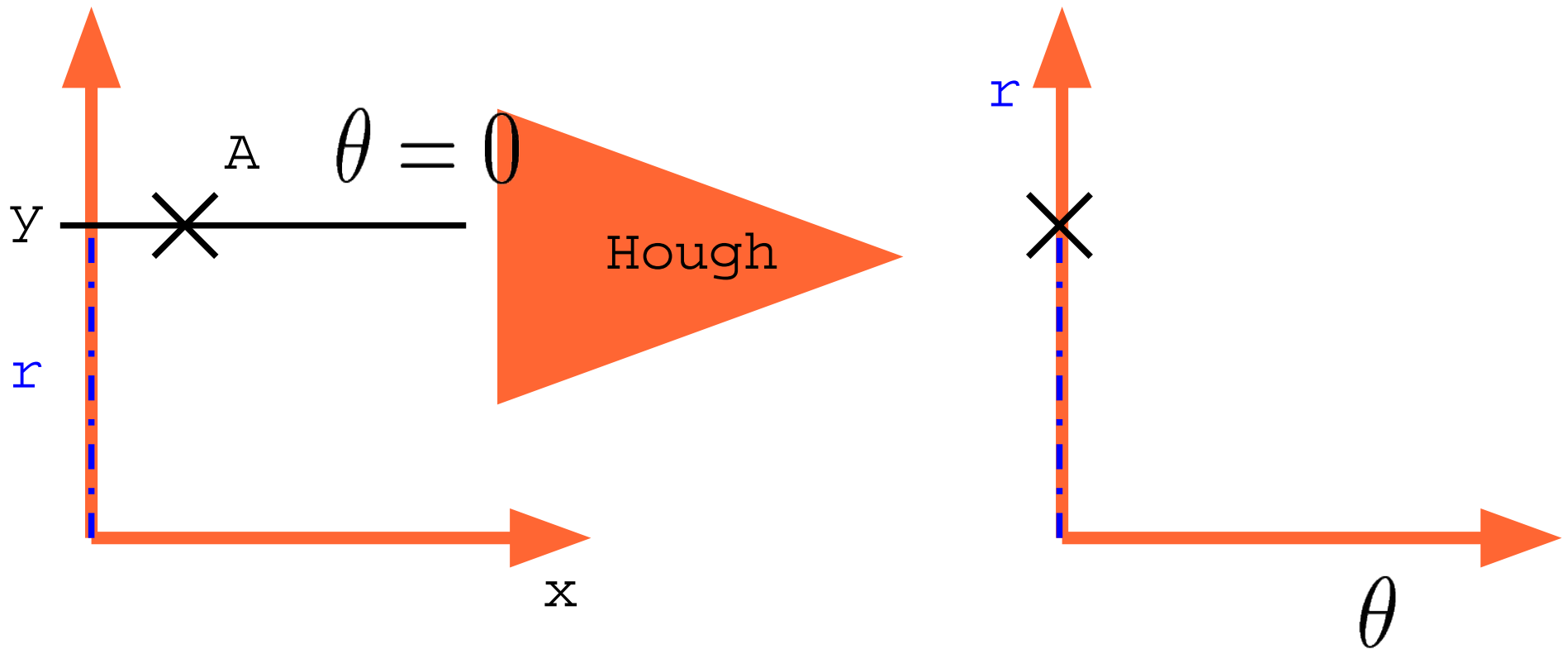
$$(x_1 - x_2) \cos(\theta) - (y_1 - y_2) \sin(\theta) = 0$$

$$\frac{\sin(\theta)}{\cos(\theta)} = \tan(\theta) = -\frac{x_1 - x_2}{y_1 - y_2} \quad \theta = \text{atan} \left(\frac{x_1 - x_2}{y_1 - y_2} \right) \in [-\pi, \pi]$$

- Dans une image (n_c, n_l) avec un repère centré en $(n_c/2, n_l/2)$
 $r = x_1 \cos(\theta) - y_1 \sin(\theta) \in [-r_{max}, r_{max}] \quad r_{max} = \sqrt{n_c^2 + n_l^2}/2$

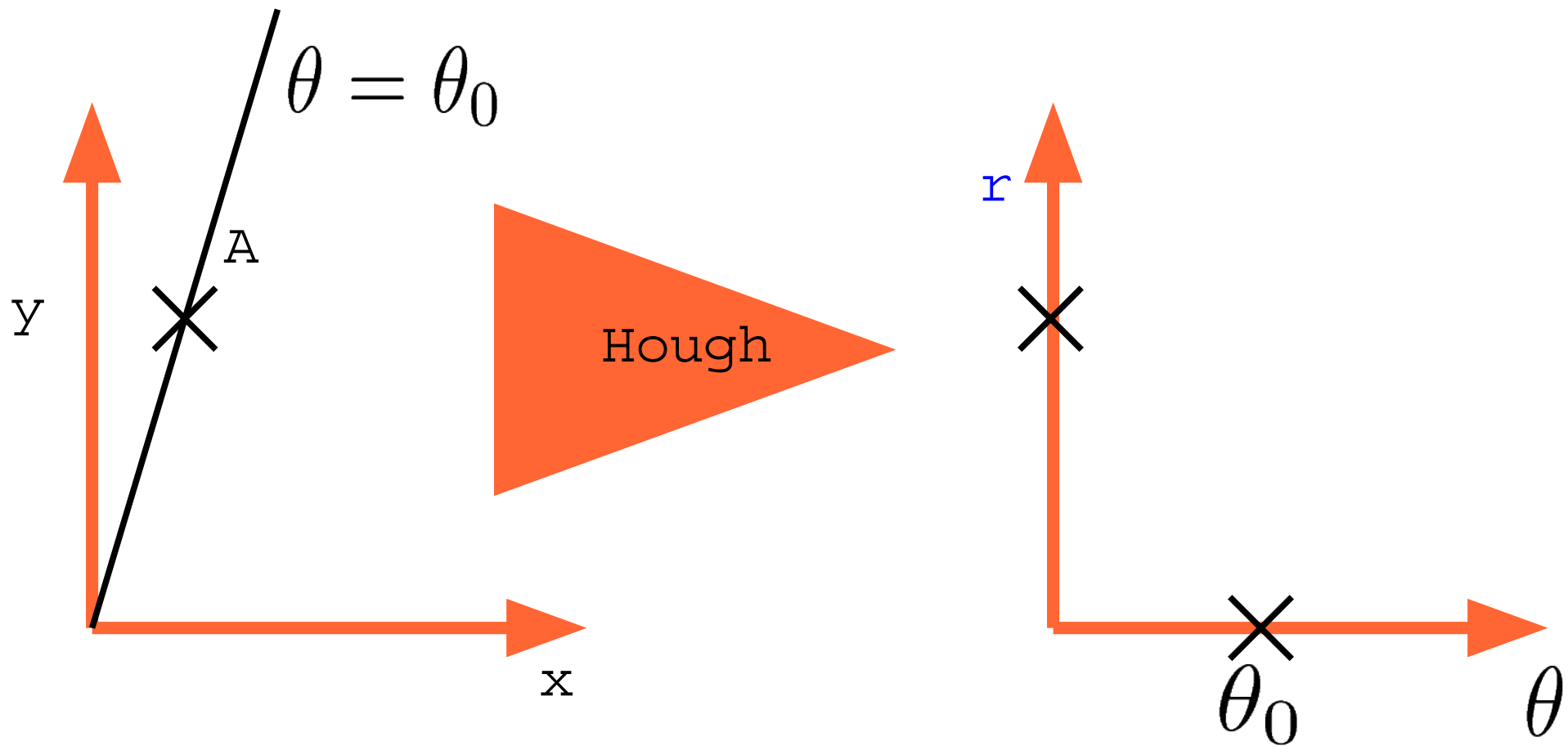
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Transformation d'un **point** dans l'espace des paramètres



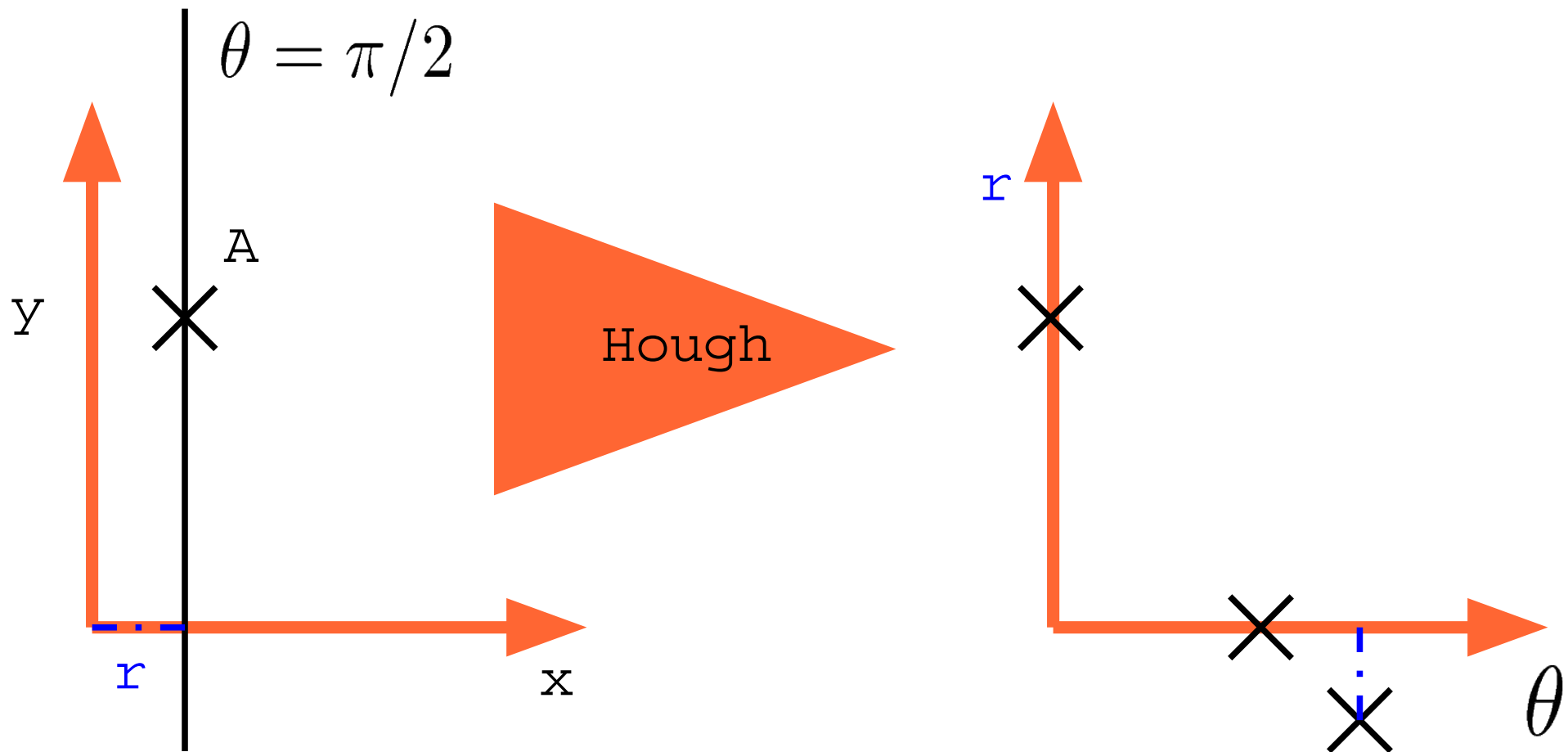
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Transformation d'un **point** dans l'espace des paramètres



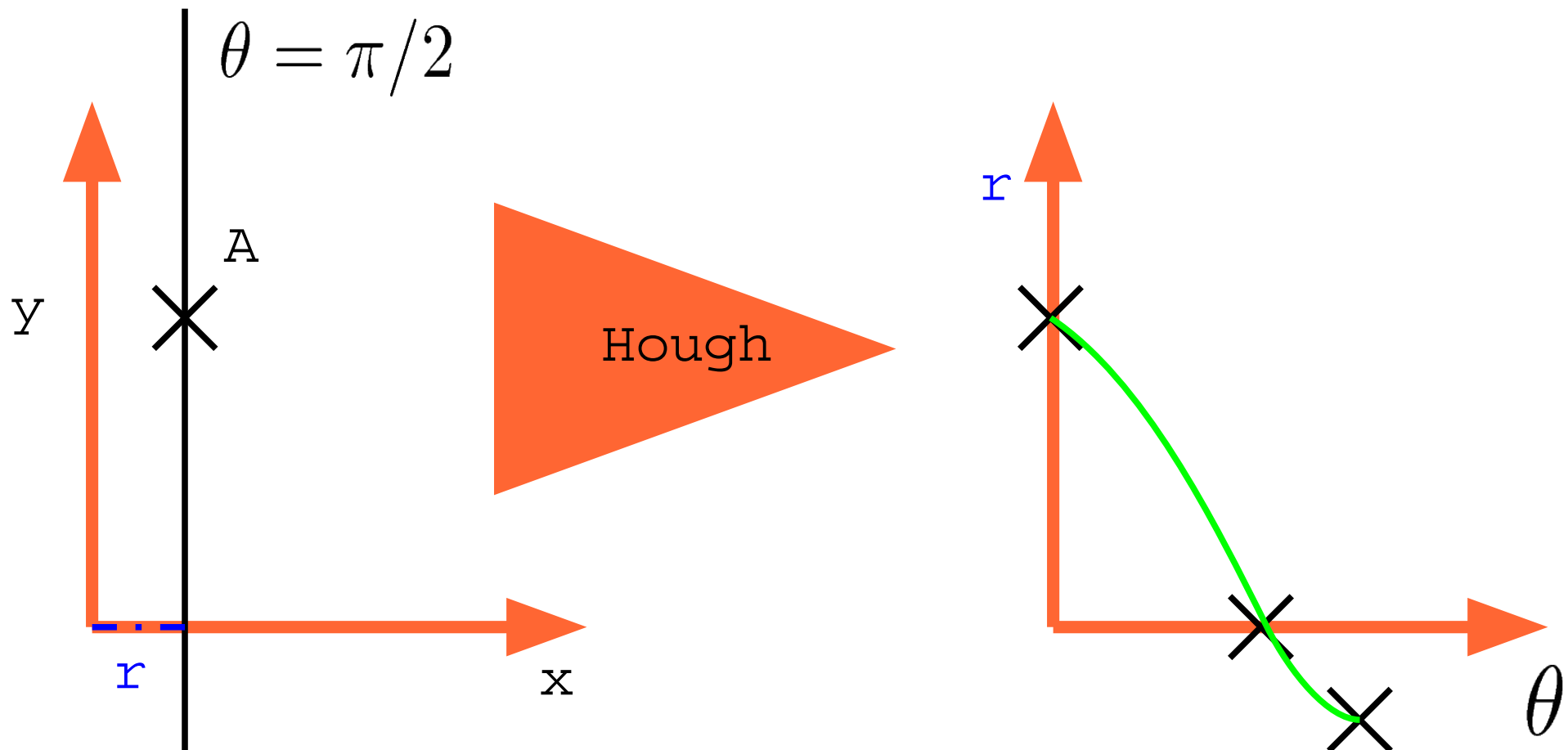
Exemple de la droite:

- Transformation d'un **point** dans l'espace des paramètres



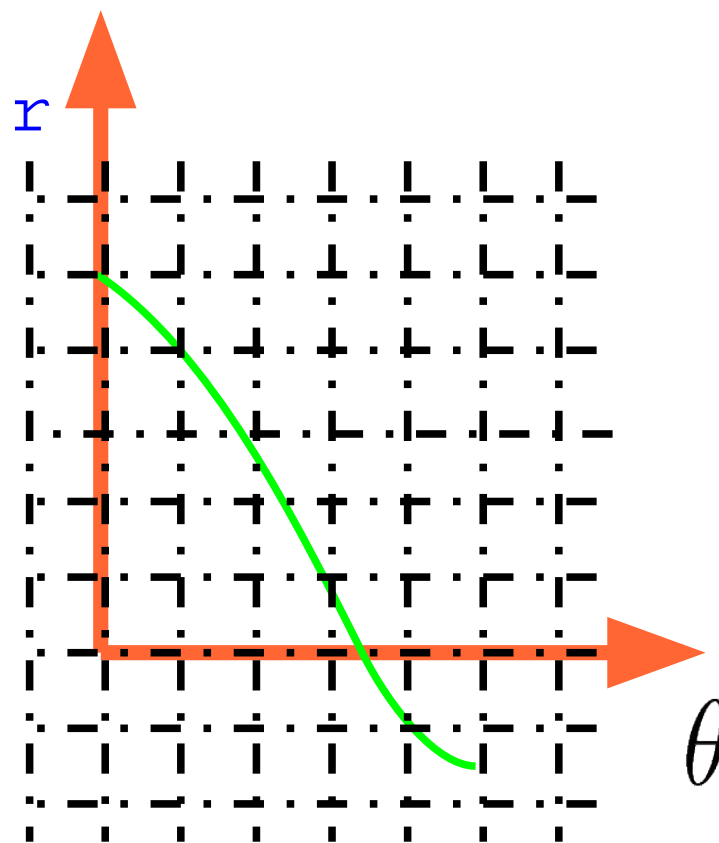
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Transformation d'un **point** dans l'espace des paramètres



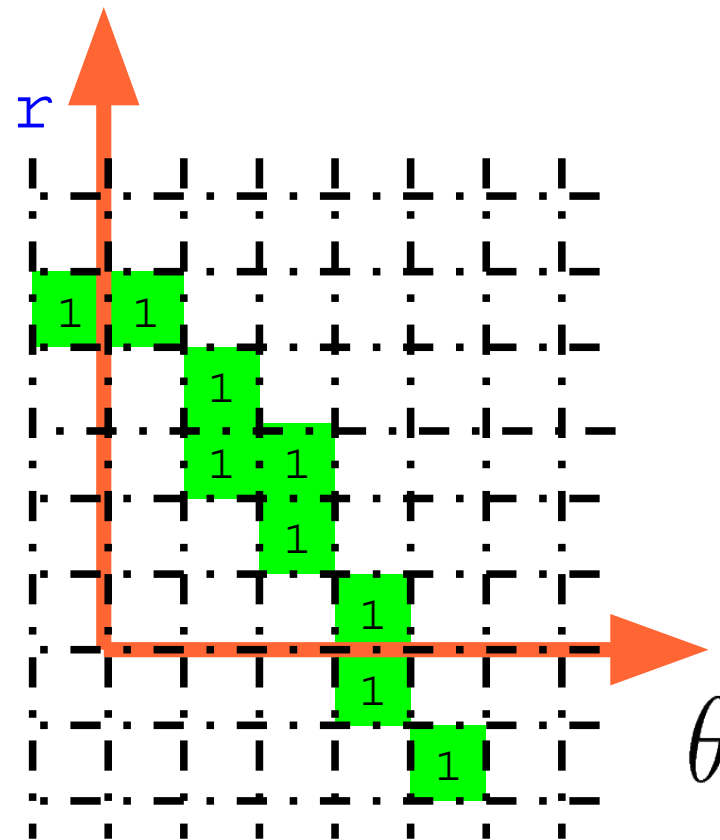
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Discrétisation de l'espace des paramètres:



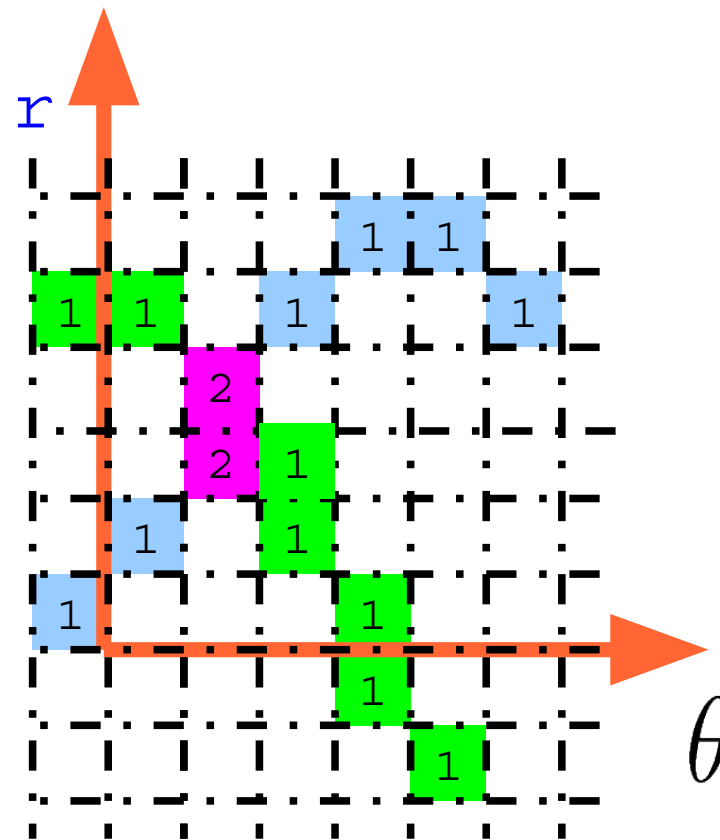
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Accumulation dans l'espace des paramètres:



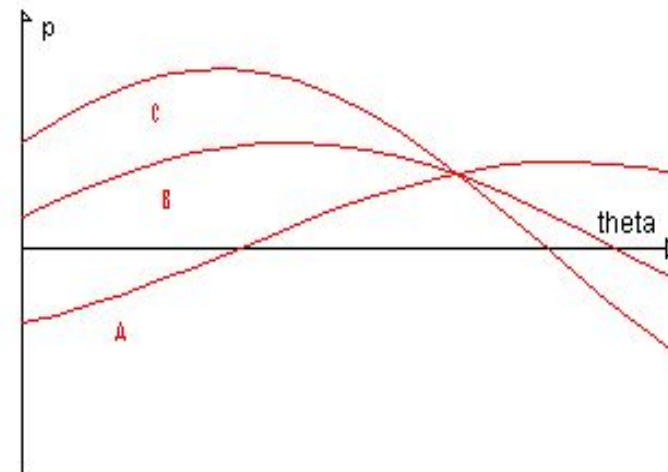
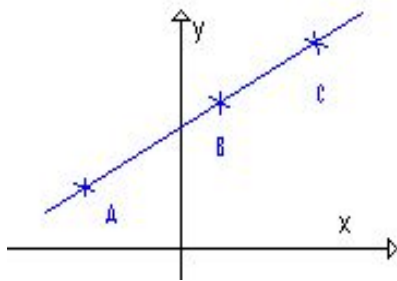
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

- Accumulation dans l'espace des paramètres:

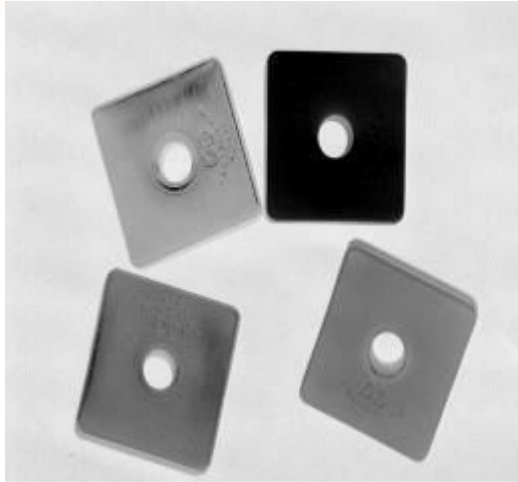


Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$

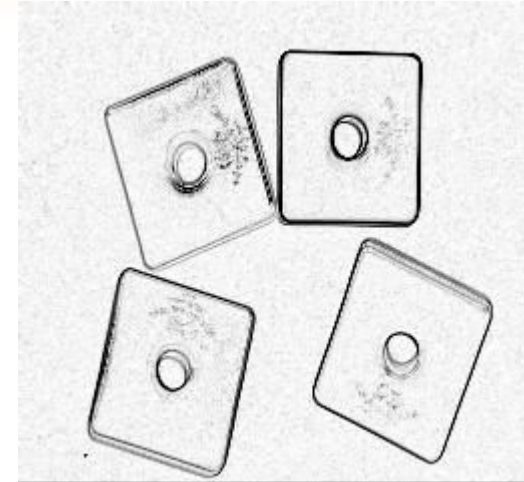
- Extraction des maxima locaux



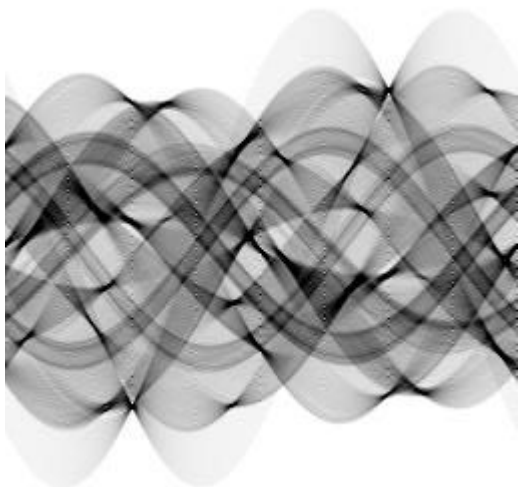
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$



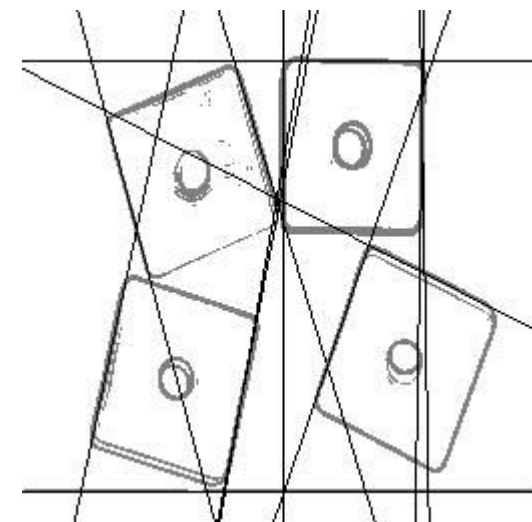
Image



Contours

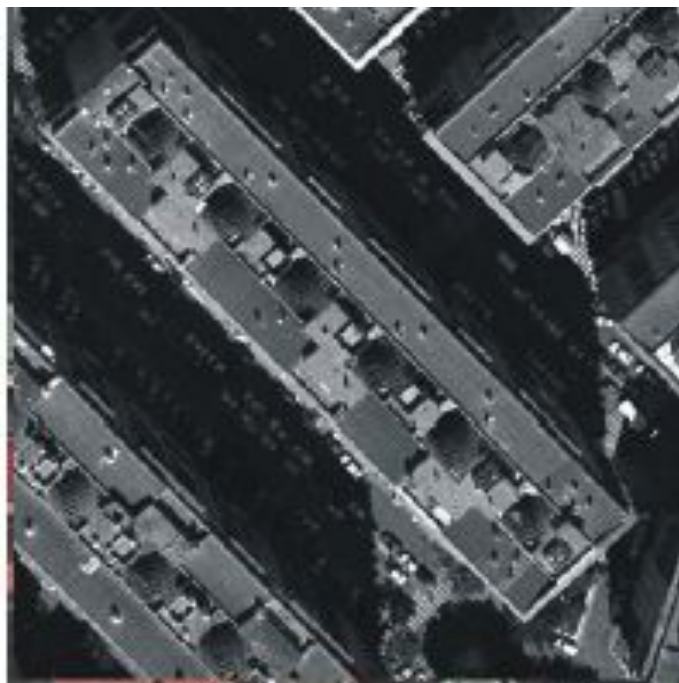


Accumulateur

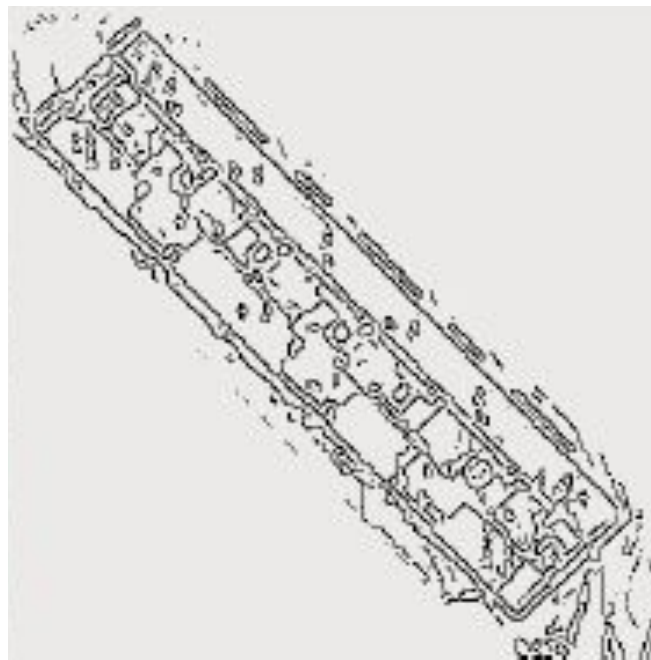


Droites

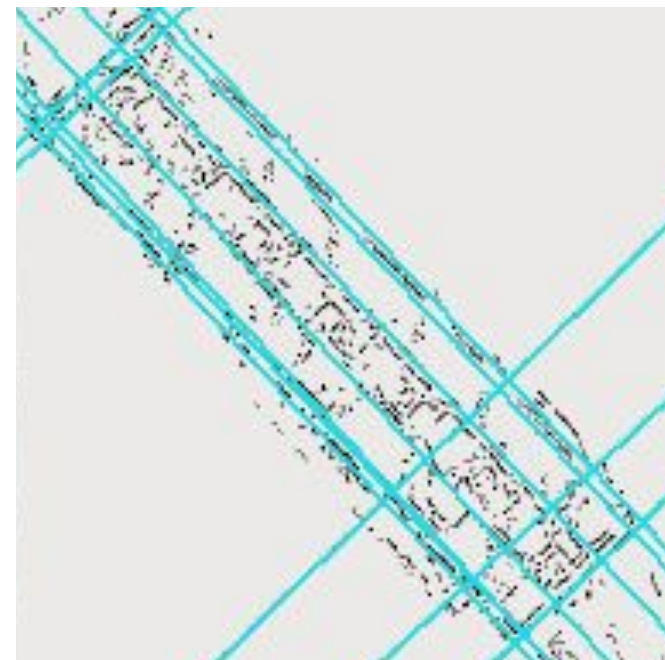
Exemple de la droite: $(r, \theta) : x \cos(\theta) - y \sin(\theta) = r$



Image



Contours



Droites

Exemple du cercle



Conclusion

- La reconnaissance des formes est une discipline indispensable à :
 - l'exploitation géométrique de la donnée brute (image/laser)
 - L'interprétation de la donnée
- Elle se repose sur l'extraction de formes géométriques
- La transformée de Hough permet cette extraction en se plaçant dans l'espace des paramètres de la forme géométrique