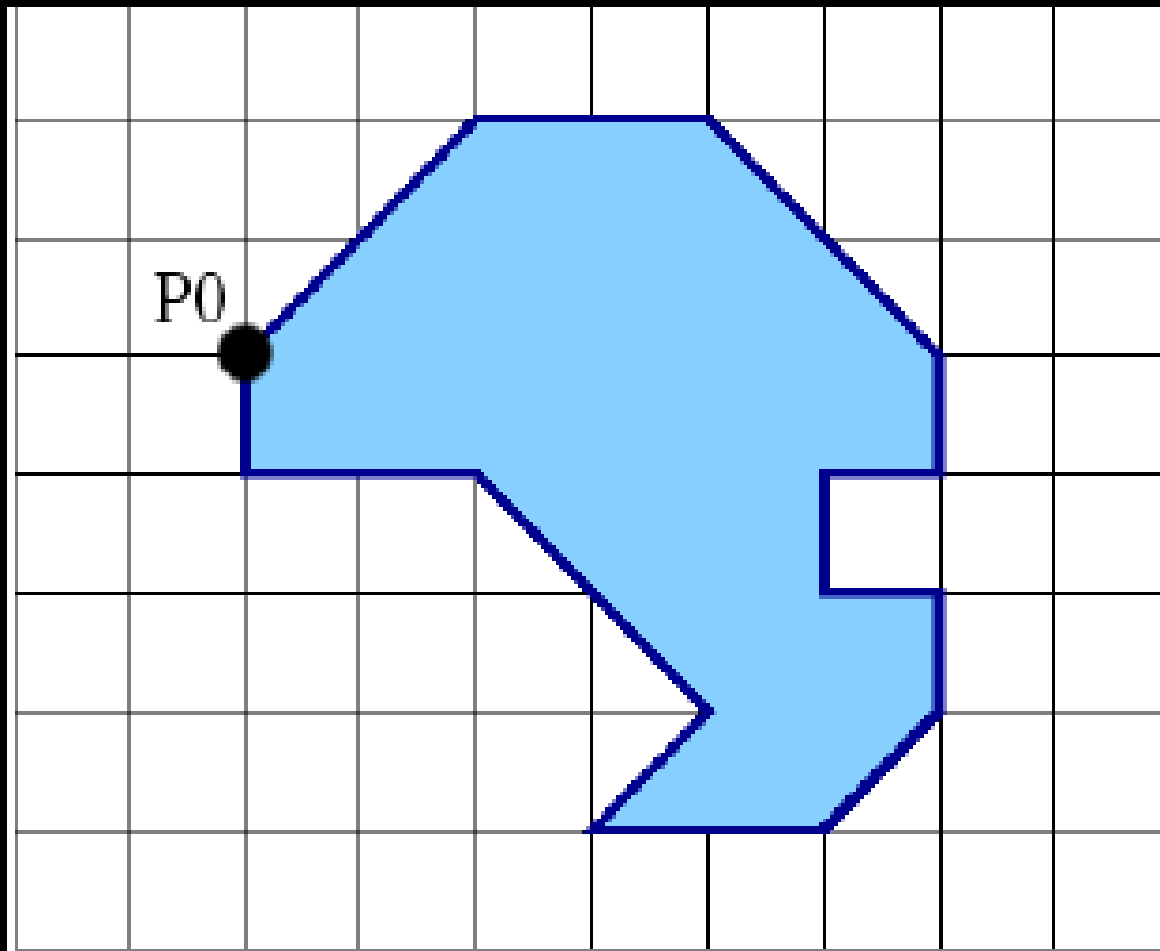


Contours dans une image

Pour aller plus loin avec l'image des contours...

- Comment coder le contour ?



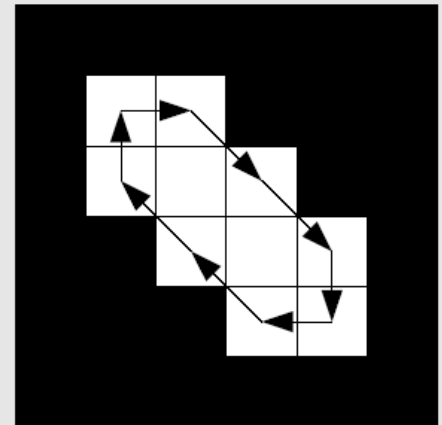
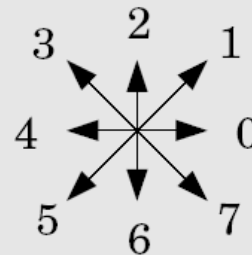
Codage de Freeman

Principe

- codage avec un nombre **limité de bits** de la direction locale d'un élément de contour défini dans une **image discrète**.
- constitution d'une **chaîne de codes** à partir d'un pixel initial, considérant qu'un élément de contour relie **2 pixels connexes**.

Méthode

- 1) on choisit un **pixel initial** du contour et un **sens de parcours**;
- 2) on code la direction qui permet de passer d'un pixel du contour à son **voisin immédiat**;
- 3) on continue jusqu'à revenir au **pixel initial**.



0 7 7 6 4 3 3 2

Codage de Freeman relatif

Principe

on code le **changement** de direction plutôt que la direction.



Propriétés

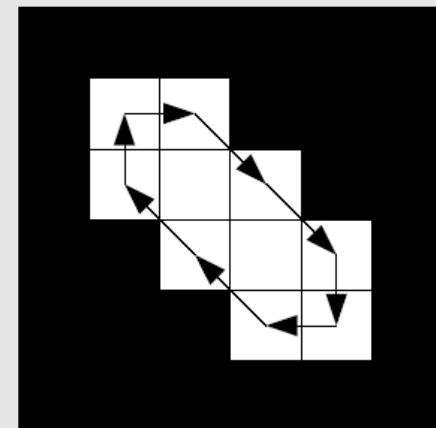
- le code de Freeman standard est invariant en **translation** uniquement.
- le code de Freeman relatif est invariant en **translation** et aux **rotations** de 45 deg.

Variantes

codage sur 2 bits pour **connexité 4**

codage sur 3 bits pour **connexité 8**

codage sur 4 bits pour **connexité 8 + longueur 2**



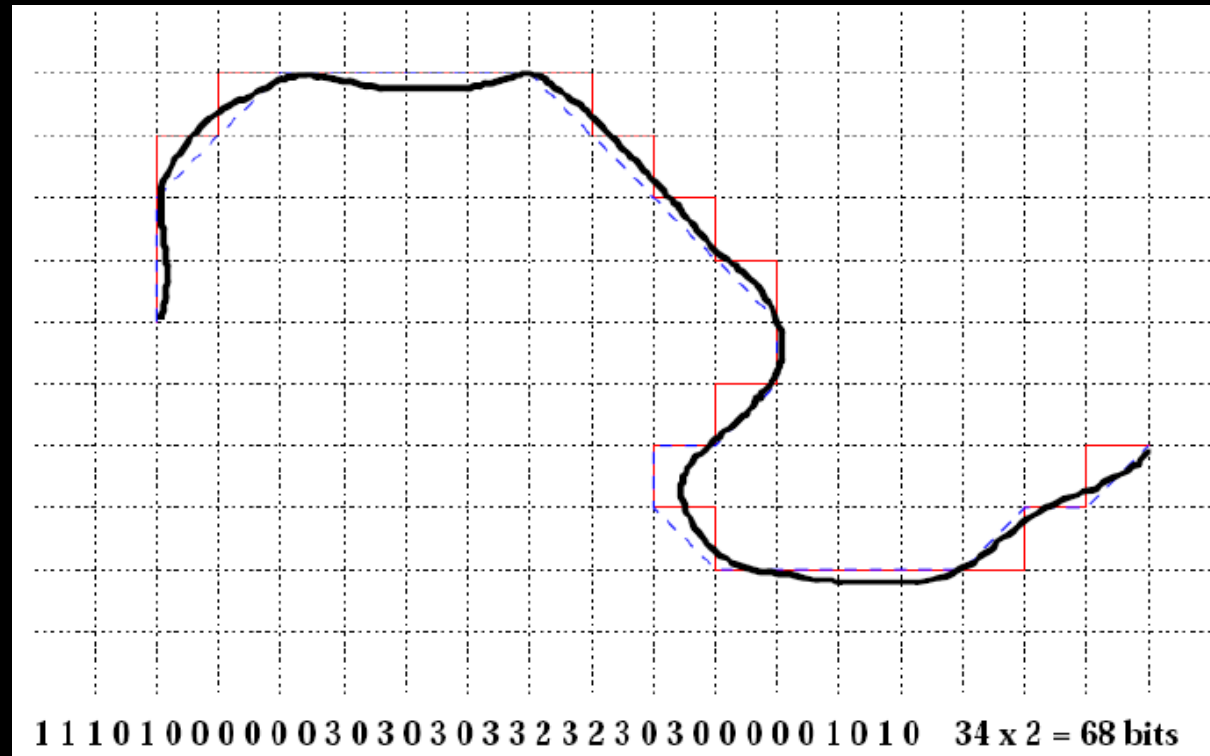
0 7 0 7 6 7 0 7

The figure shows a smooth curve (solid black line) and its piecewise linear approximation (dashed blue line) on a grid. The approximation is formed by connecting the midpoints of the vertical segments of the curve, with red rectangles highlighting the steps.

1110100000030303033232303000001010	34 x 2 = 68 bits
221100007777654670000101	24 x 3 = 72 bits
1098815156546788101	16 x 4 = 64 bits

Codage de Freeman

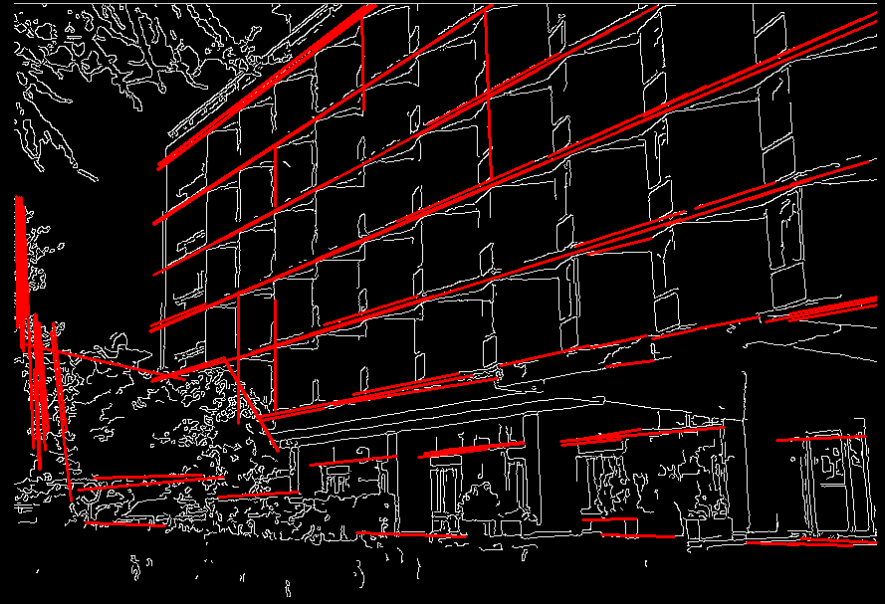
- A partir du codage de Freeman, on peut caractériser le contour
- Histogramme des orientations ?



Détection de lignes droites

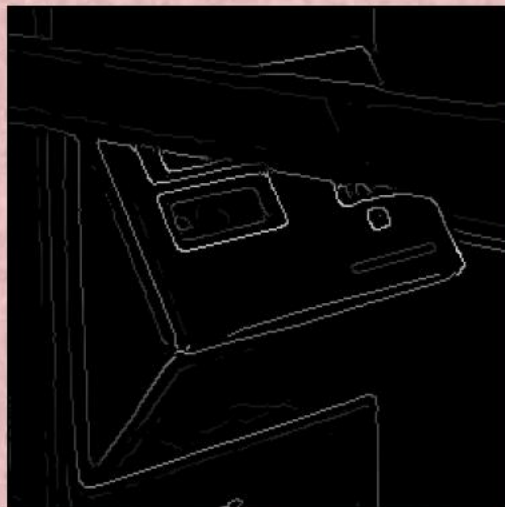
avec la Transformée de Hough

- à partir d'une image binarisée des contours, on veut détecter les **lignes droites** dans une image

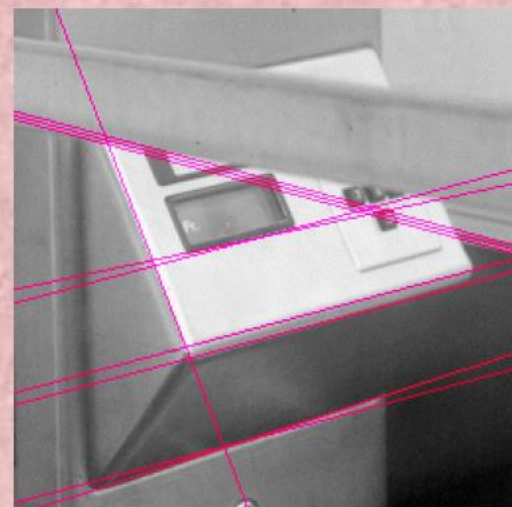




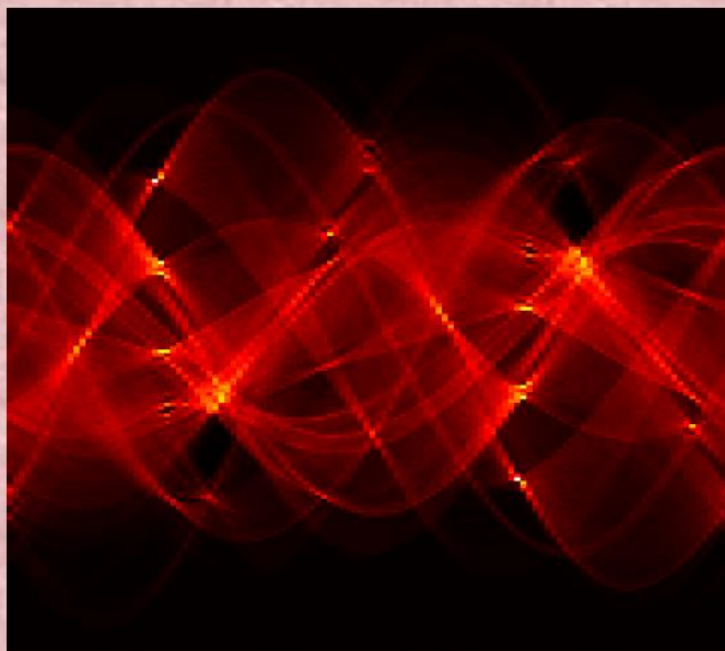
**Original
Image**



**Edge
Detection**



**Detected
Lines**

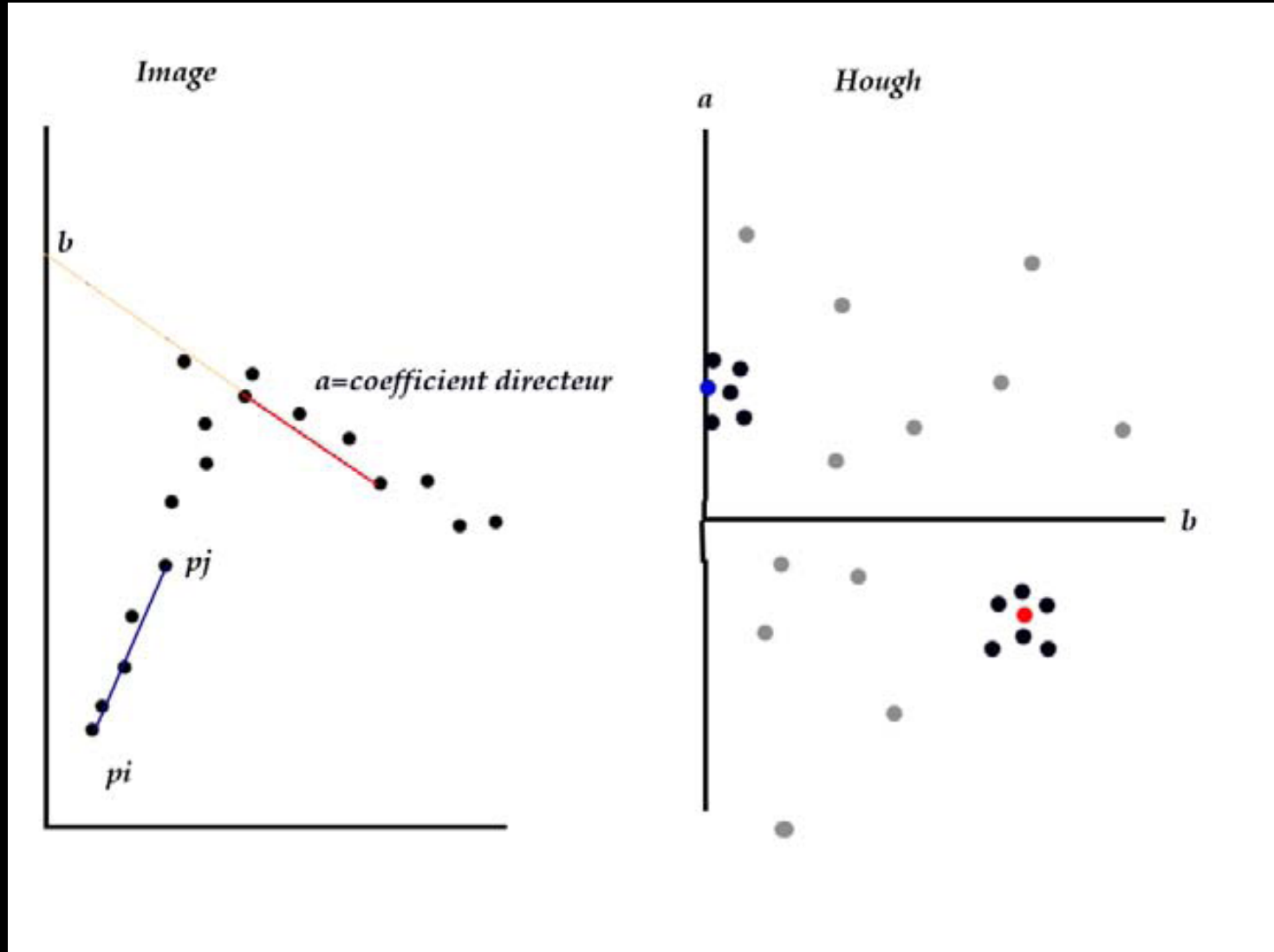


**Parameter
Space**

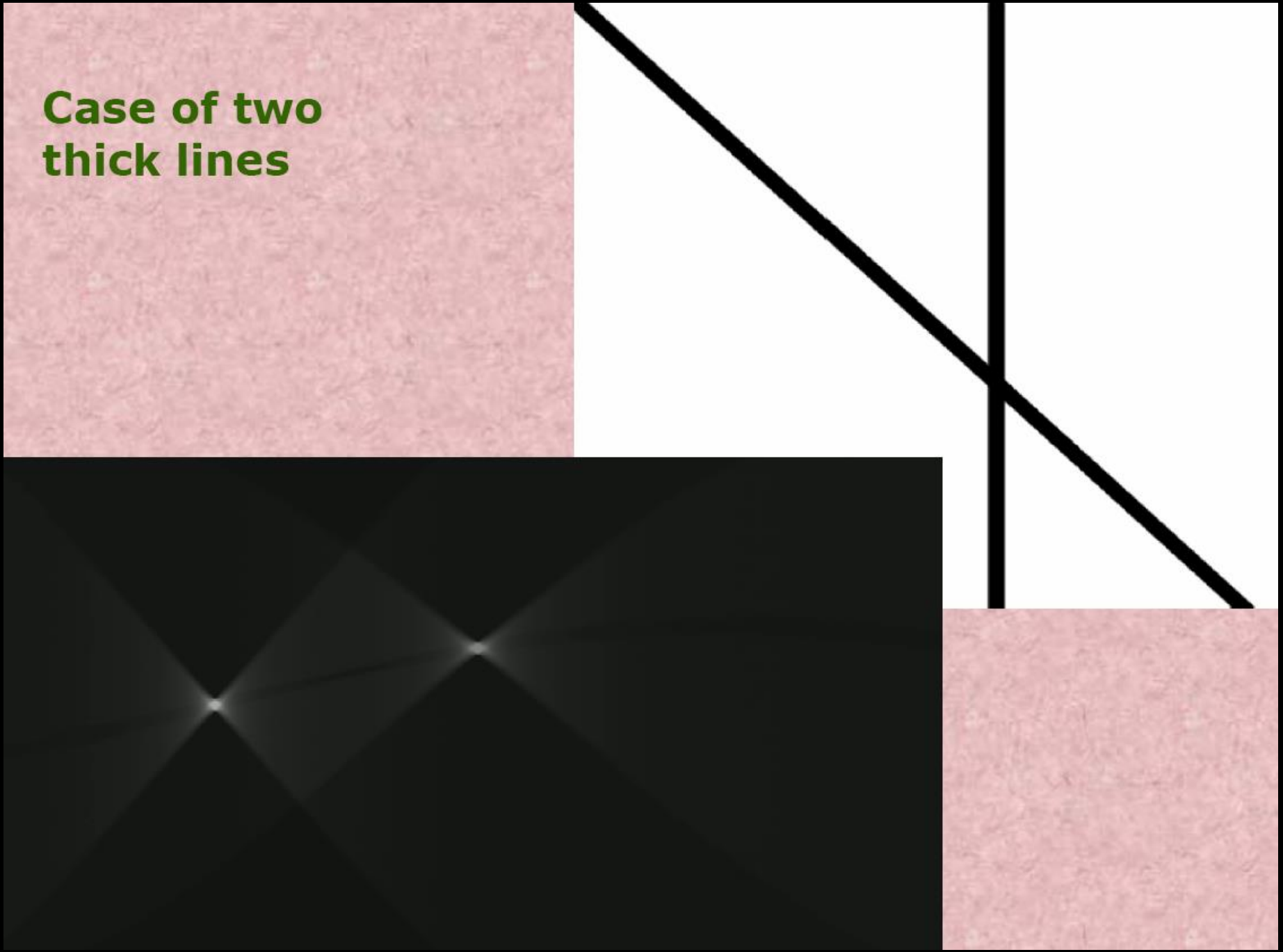
Transformée de Hough

- Principe : pour tous les couples de points $P1=(X1,Y1)$ et $P2=(X2,Y2)$ appartenant au gradient, on cherche la droite (a,b) qui passe par ces points :
 - $Y1 = aX1 + b$
 - $Y2 = aX2 + b$
- On compte le nombre de couples (a,b) obtenus
- On conserve les maximaux

Transformée de Hough



**Case of two
thick lines**



Transformée de Hough

- Principe

Initialiser la matrice H à 0.

Pour tout couple de pixels $\{p_i, p_j\}$ (étiquetés) :

- déterminer les paramètres a_{ij} et b_{ij} de la droite (p_i, p_j)
- ajout d'un vote dans la matrice H aux coordonnées $\{a_{ij}, b_{ij}\}$.

Recherche dans H des nuages de points les plus denses.

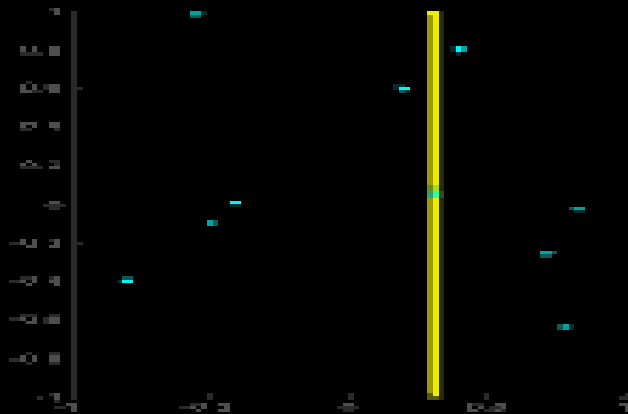
Déterminer les centres (=caractéristiques des droites détectées dans l'image)

Problèmes de mise en œuvre

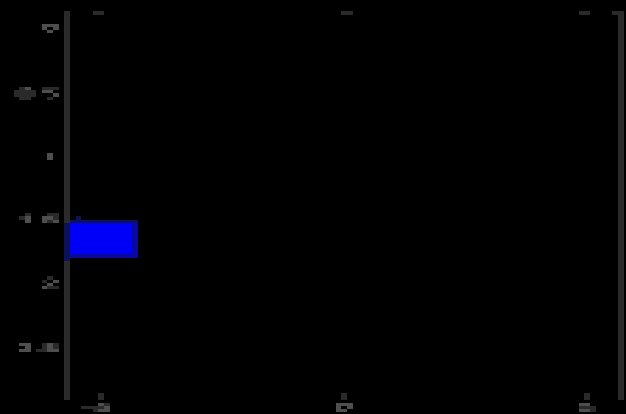
- a_{ij} et b_{ij} doivent être bornés et quantifiés.
- Précision du calcul des centres.

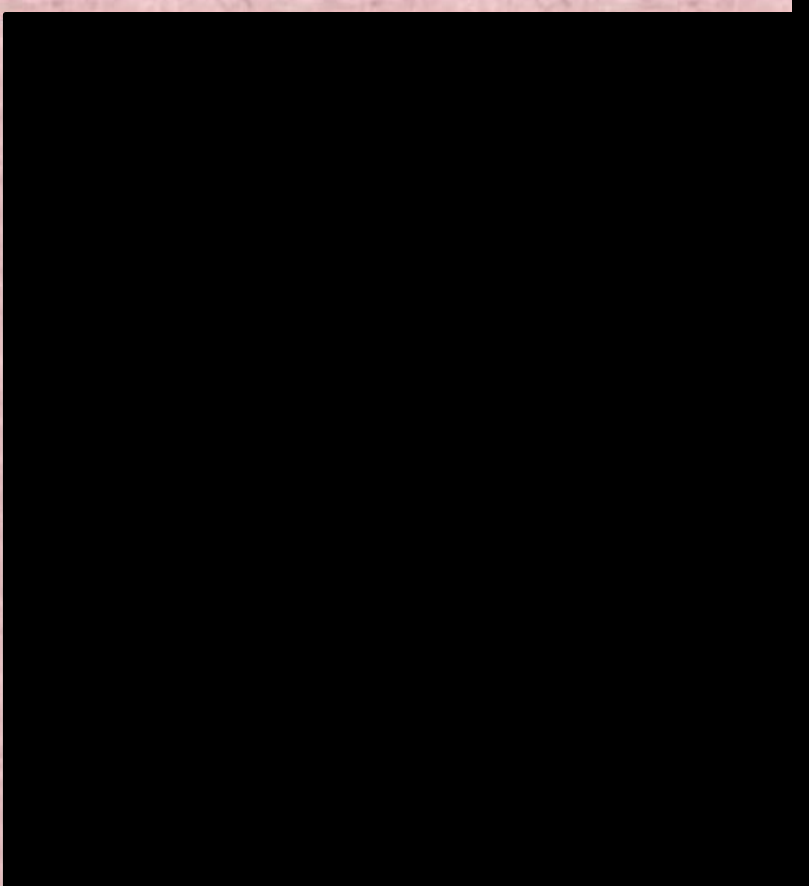
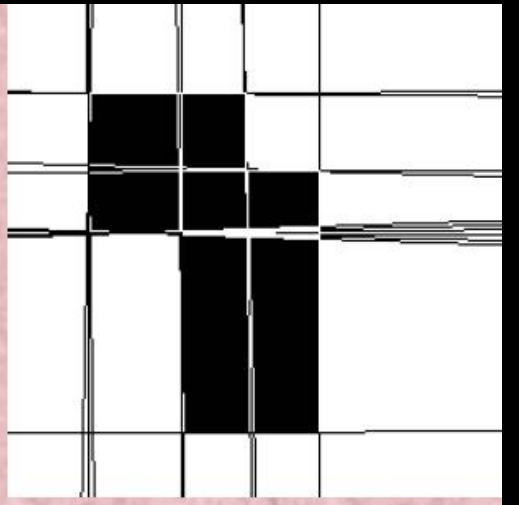
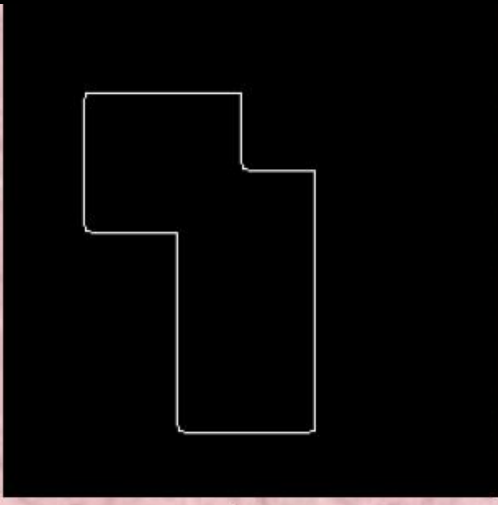
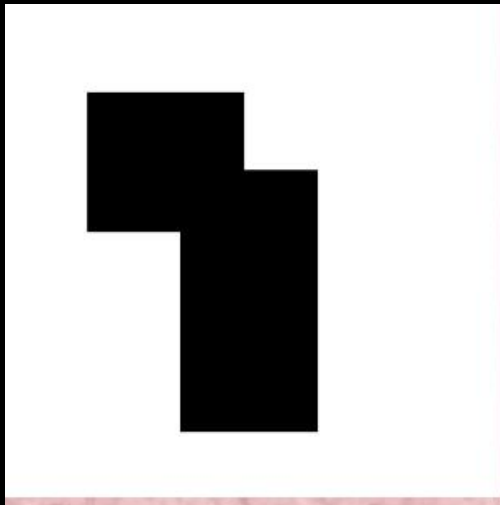
Transformée de Hough

Image



Espace (a,b)





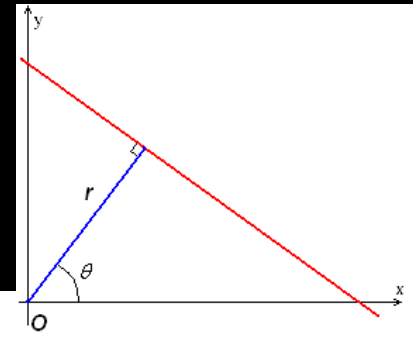
Algo Transfo. De Hough

- Coordonnées cartésiennes

```
Pour chaque point  $(x, y)$  de l'image n'appartenant pas au fond
  Pour chaque valeur  $a$ 
    calcul de  $b = y - ax$ 
    ajout d'un vote pour  $(a, b)$ 
     $\text{vote}[a][b] ++;$ 
```

- Coordonnées polaires

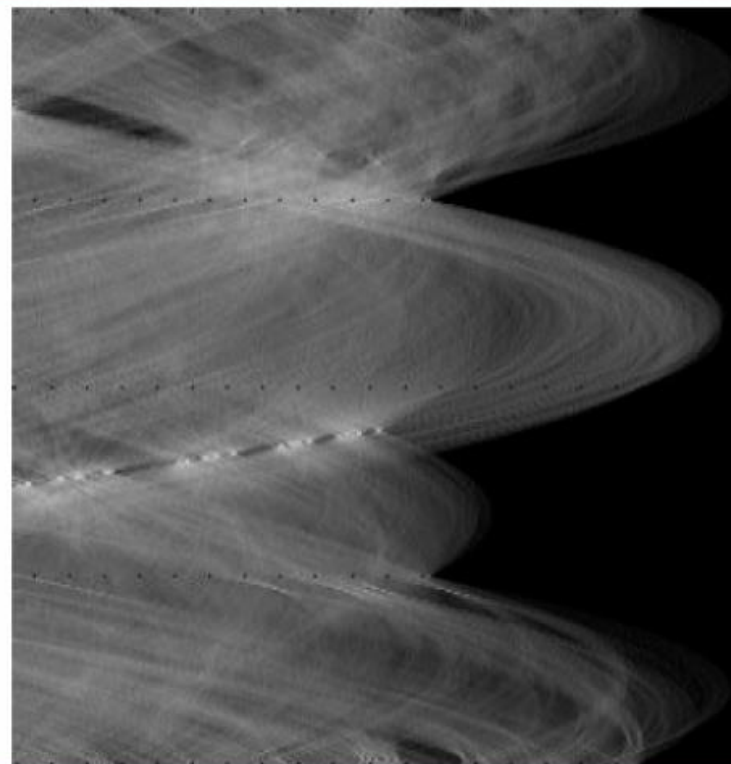
$$r = x \cos\theta + y \sin\theta$$



```
Pour chaque point  $(x, y)$  de l'image n'appartenant pas au fond
  Pour chaque valeur  $\theta$ 
    calcul de  $\rho = x \cos(\theta) + y \sin(\theta)$ 
    ajout d'un vote pour  $(\rho, \theta)$  :
     $\text{vote}[\rho][\theta] ++;$ 
```




peaks →

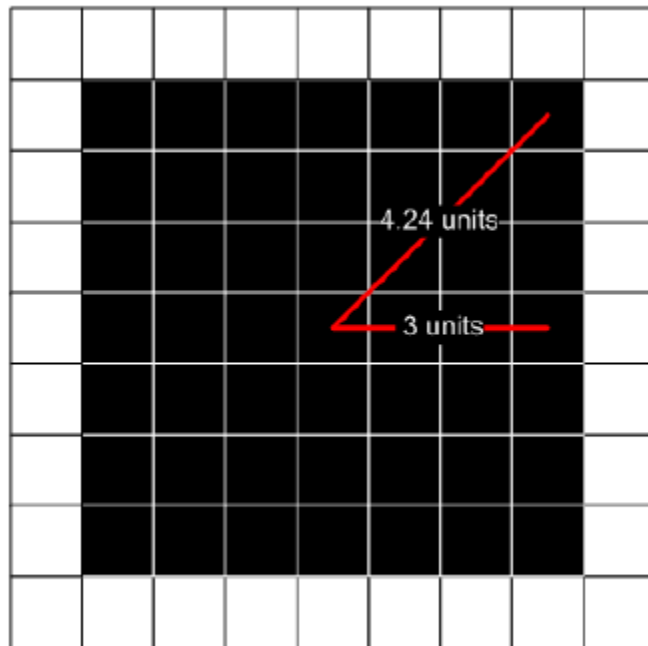


Transformée de Hough

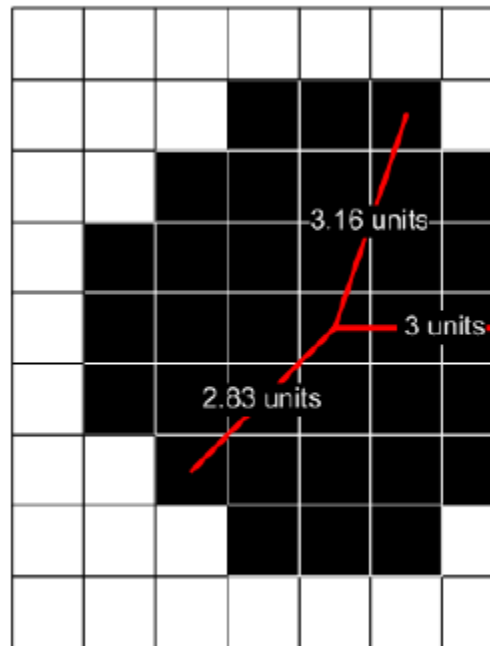
- Détection de courbe type

```
Pour chaque valeur de a entre a_min et a_max  
  pour chaque valeur de b entre b_min et b_max  
    calcul de c t.q.  $ax^2 + bxy + cy^2 = 1$   
    ajout d'un vote pour (a,b,c) : vote[a][b][c]++;
```

- Transformée de Hough généralisée
 - « Matcher » un modèle dans l'image



Max. Variation in Euclidean Distance = 1.24 units



Max. Variation in Euclidean Distance = 0.83 units