



Traitements d'images

Description de contours et de formes

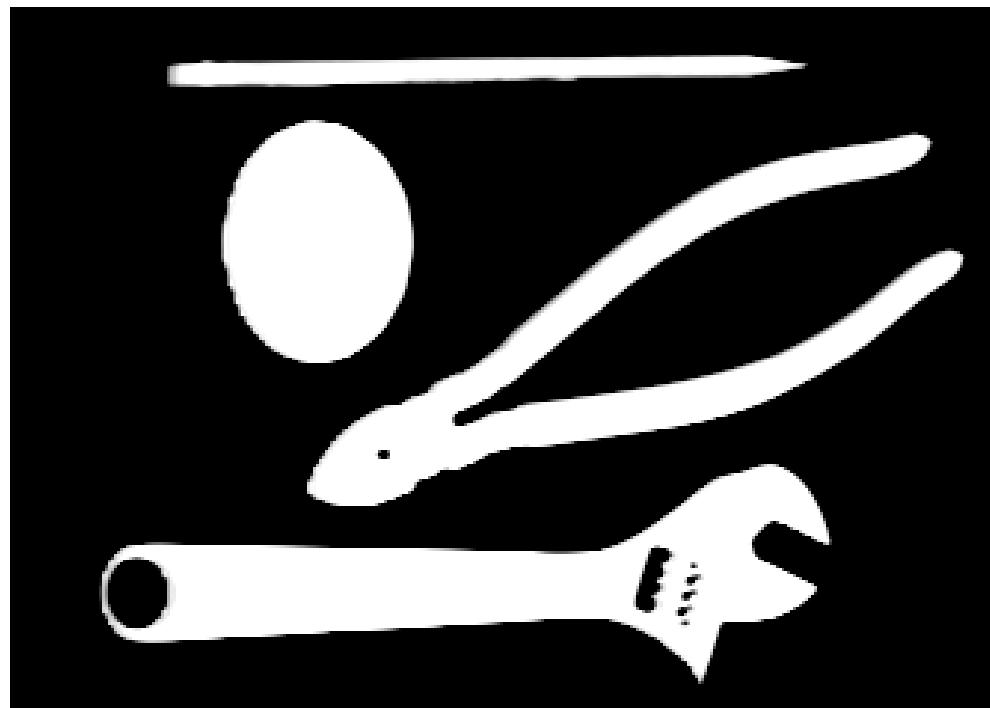
Michel Roux

Description de contours et de formes

- **Analyse des objets présents dans une image**
 - description de formes
 - description de contours
 - transformées de Hough

Objets

- Les objets sont les composantes connexes d'une image binaire.



Préliminaire : étiquetage

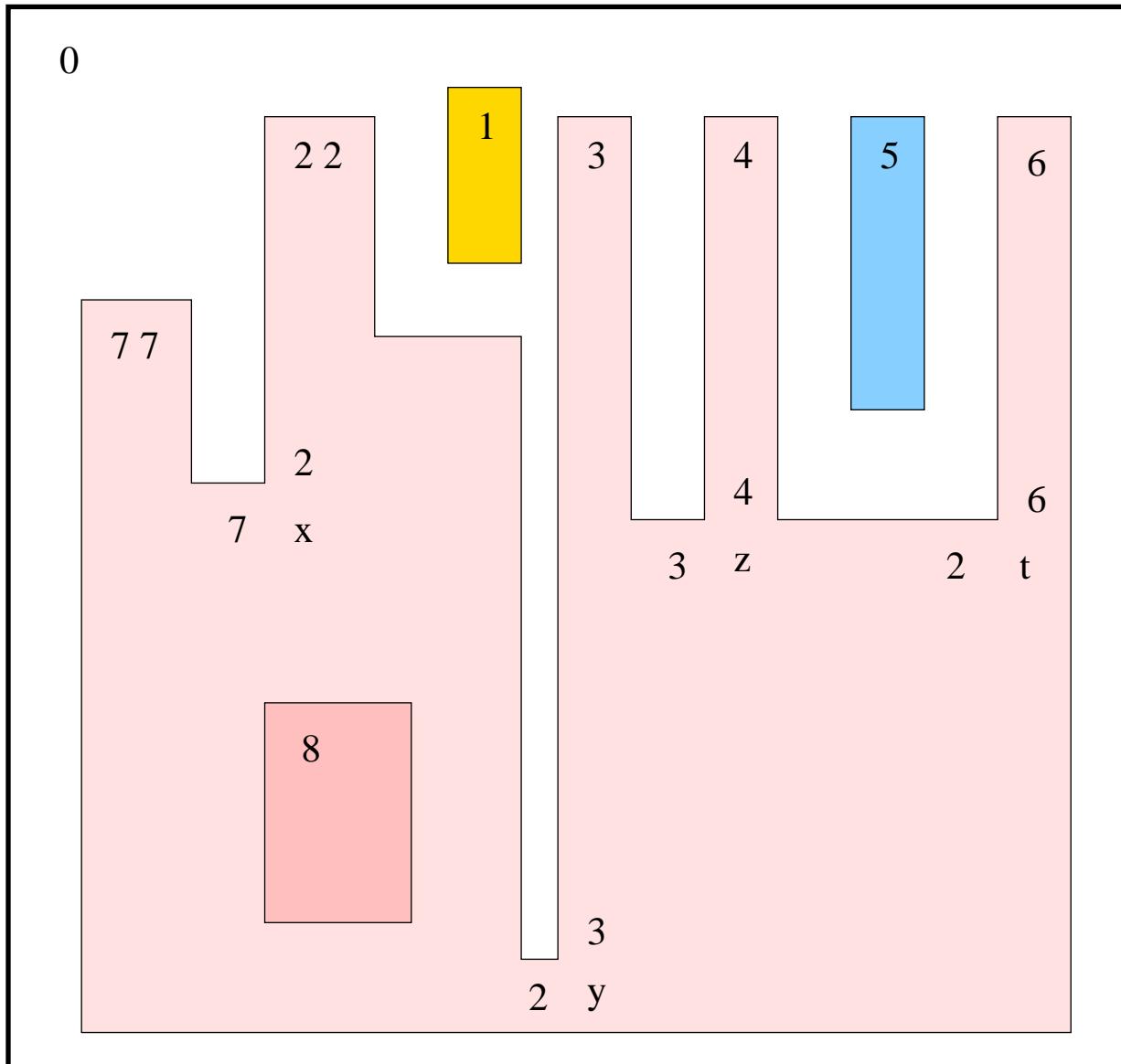
- **Principe**

associer une étiquette à chaque composante connexe
(dénombrer et isoler les objets)

- **Algorithme**

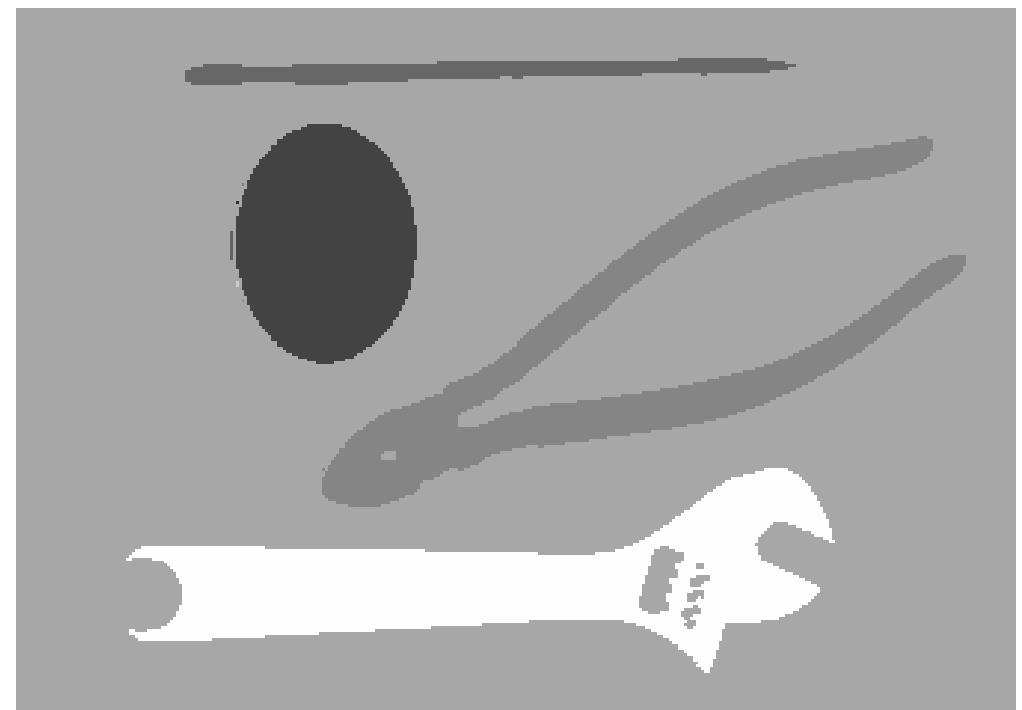
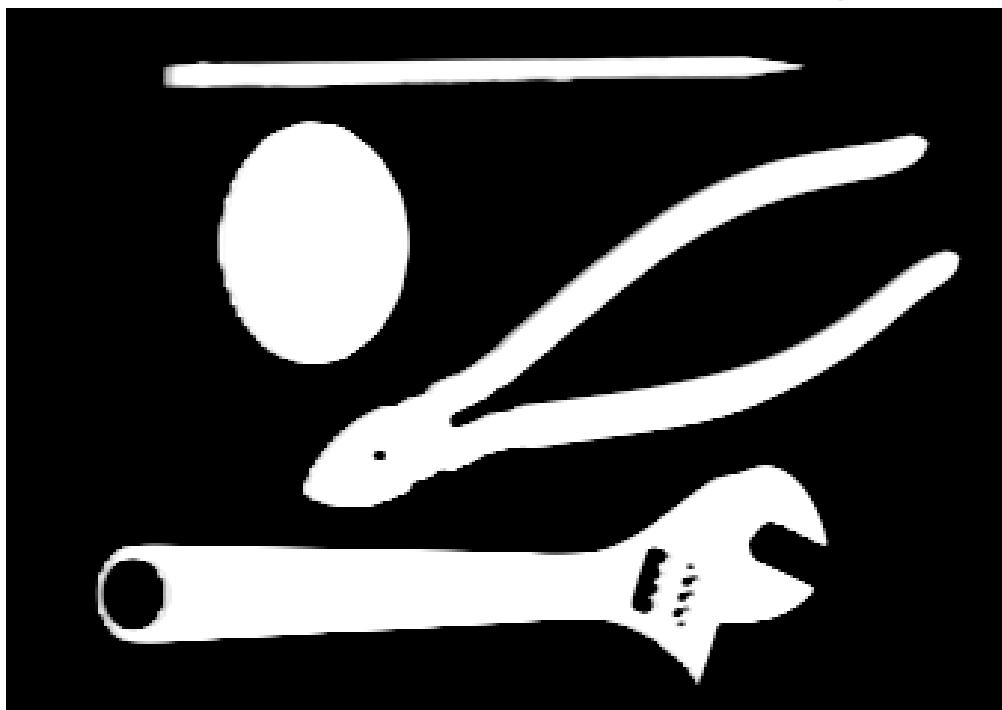
- 2 passes sur l'image
- liste d'étiquettes mise à jour
- étiquetage des points
- remplacement des étiquettes par les étiquettes terminales le cas échéant
(! choix de la connexité important)

Préliminaire : étiquetage



étiquette initiale	pointeur	étiquette finale
0		0
1		1
2		2
3	→ 2	2
4	→ 3	2
5		3
6	→ 2	2
7	→ 2	2
8		4

Préliminaire : étiquetage



Préliminaire : étiquetage

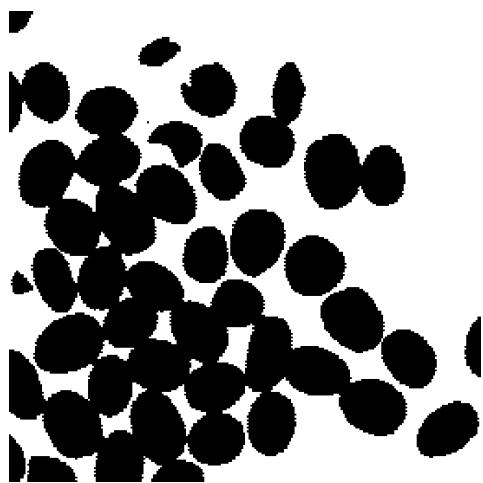
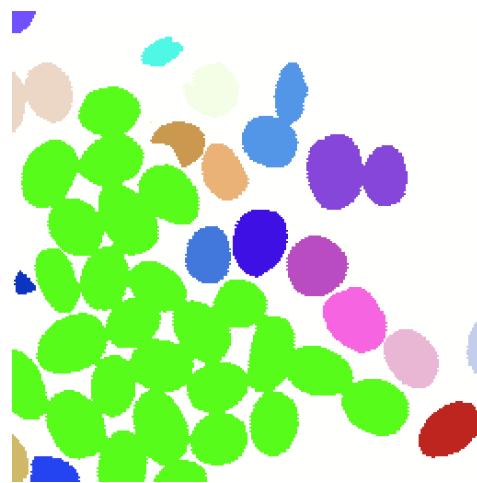
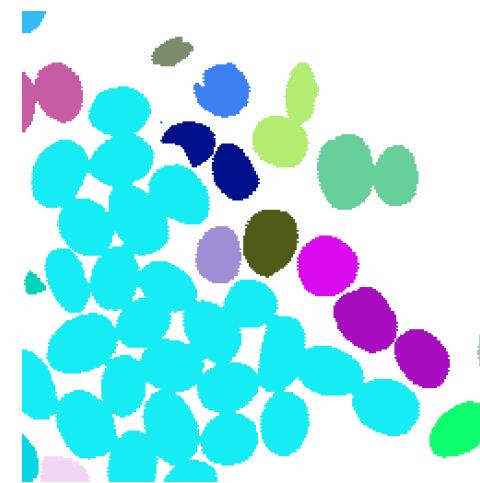


image binaire



4-connexité

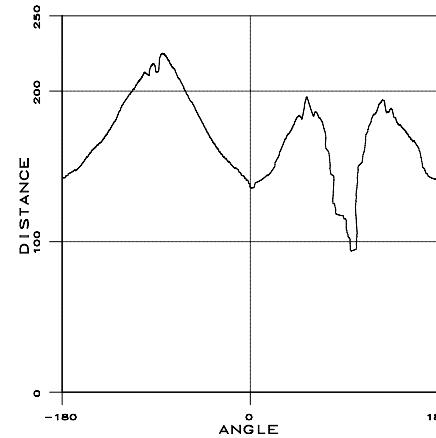
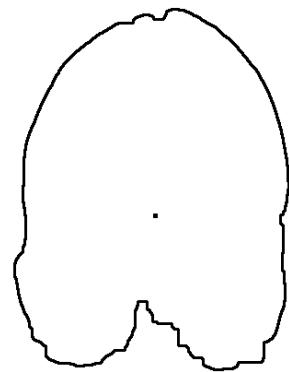


8-connexité

Signature d'un contour

○ Principe

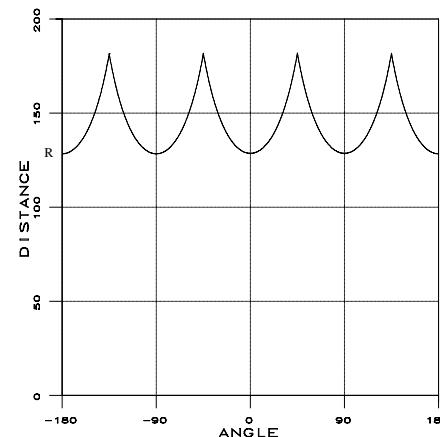
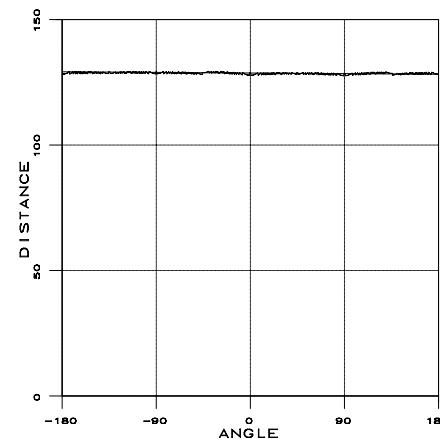
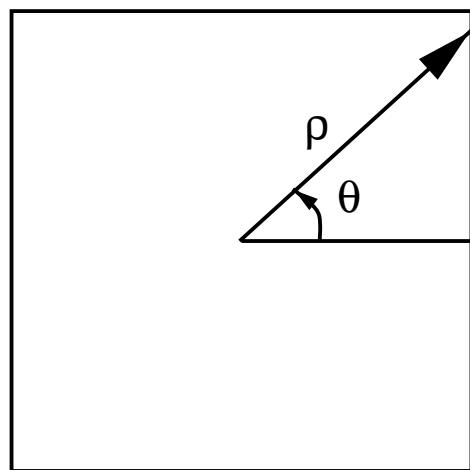
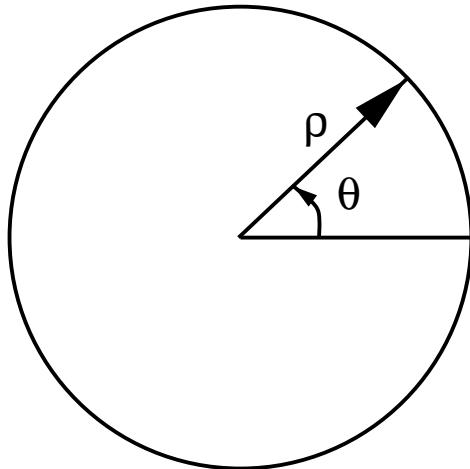
- centre de gravité = origine d'un système de coordonnées polaires
- liste des distances à l'origine en fonction de l'angle \Rightarrow fonction $\rho(\theta)$
- comparaison à un dictionnaire de formes par corrélation des signatures



○ Propriétés

- invariant par translation
- détection de petits défauts
- rotation = translation selon θ

Signature d'un contour - exemples



Signature d'un contour - filtrage d'objets

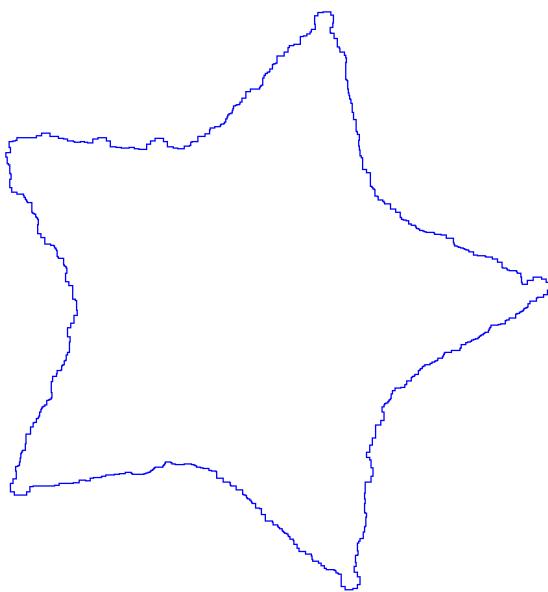
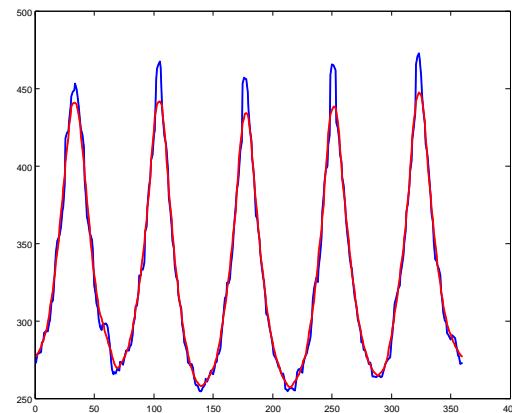
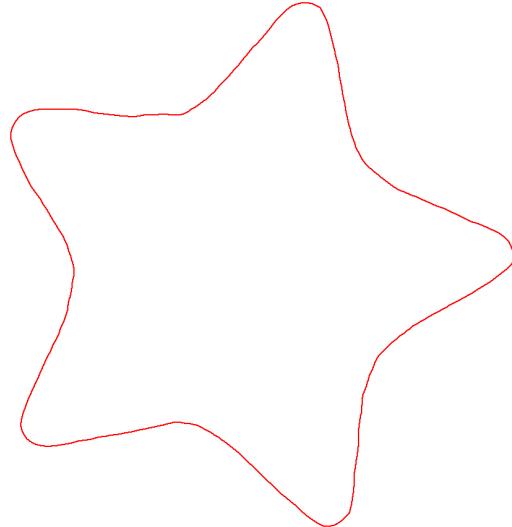


image originale



signature filtrée



forme filtrée

Signature d'un contour - reconnaissance d'objets



image originale



régions détectées



régions reconnues

Signature d'un contour - reconnaissance d'objets



image originale



régions détectées



régions reconnues

Signature d'un contour - reconnaissance d'objets



image originale



régions détectées



régions reconnues

Signature d'un contour - reconnaissance d'objets



image originale



régions détectées



régions reconnues

Descripteurs de Fourier

○ Représentation complexe

- forme décrite par un ensemble M_j de points de contours ($M_j = x_j + iy_j$)
- descripteurs de Fourier = coefficients de la transformée en Z des z_j :

$$Z_k = \frac{1}{N} \sum_{j=1}^N z_j \exp\left(-2\pi i \frac{jk}{N}\right)$$

coefficients Z_k , pour $k \in [-N/2 + 1, N/2]$

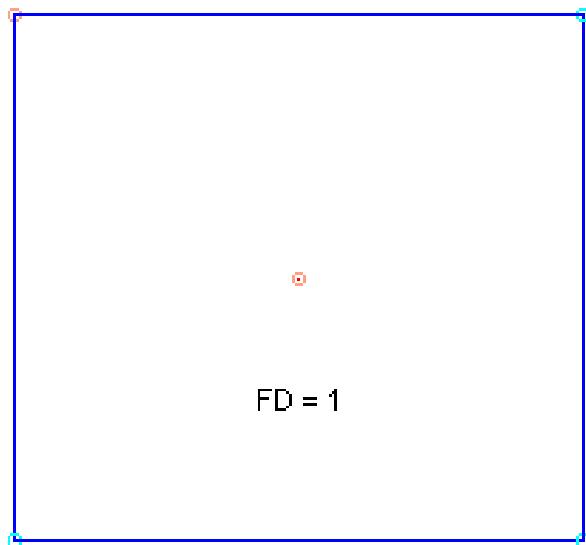
○ Propriétés

- $k = 0$, Z_0 = centre de gravité de la forme
- $Z_k = 0$ sauf pour $k = 1$, forme = cercle de rayon Z_1
(Z_1 = facteur d'échelle ; si normalisation, forme invariante par homothétie)
- coefficients $Z_{|k|}$ et $Z_{|1-k|}$ = rôles opposés mais symétriques sur la courbe

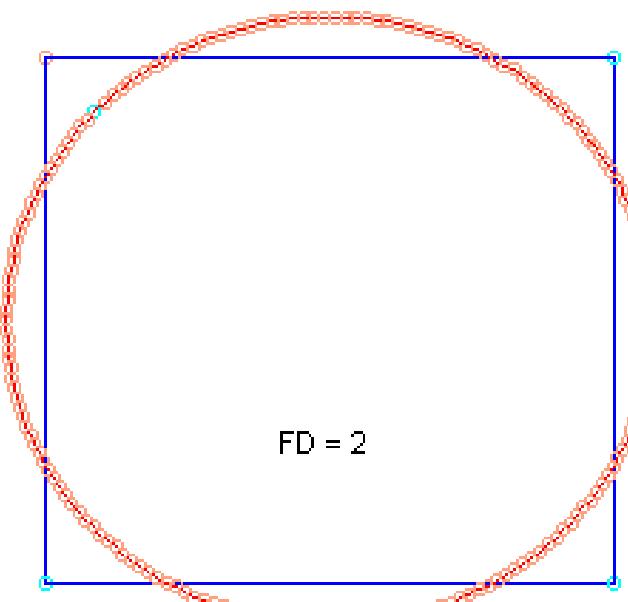
Descripteurs de Fourier

- **ordre k** = nombre d'actions sur le cercle unité (entre 0 et 2π) :
 - 1 action pour $k = 2$ et $k = -1$,
 - 2 actions pour $k = 3$ et $k = -2$,
 - 3 actions pour $k = 4$ et $k = -3$, etc.
 - valeurs de $k > 0$: actions de traction sur la courbe (pour la déformer vers l'extérieur du cercle unité),
 - valeurs de $k < 0$: actions de pression sur la courbe (pour creuser la courbe vers son centre)
 - phase du nombre complexe Z_k : ϕ_k : lieu sur le cercle unité où s'exerce l'action
- **nombre de coefficients**
 - coefficients nombreux \Rightarrow forme complexe
 - coefficients d'ordre élevés \Rightarrow détails fins sur la courbe.

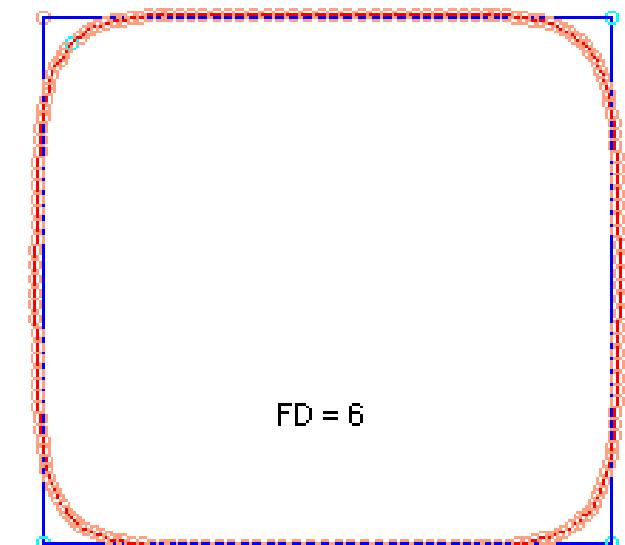
Descripteurs de Fourier



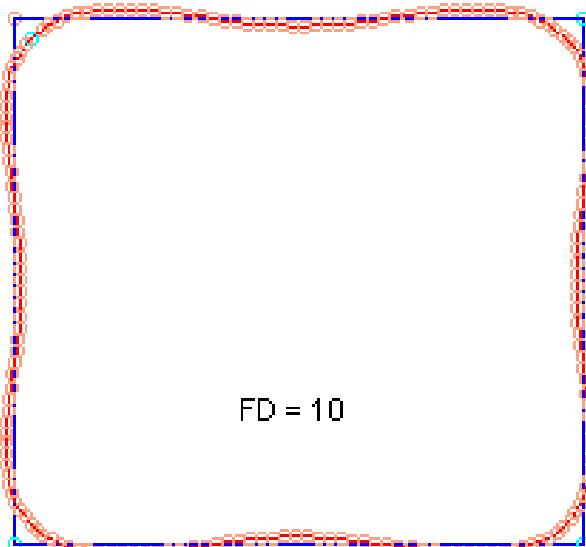
FD = 1



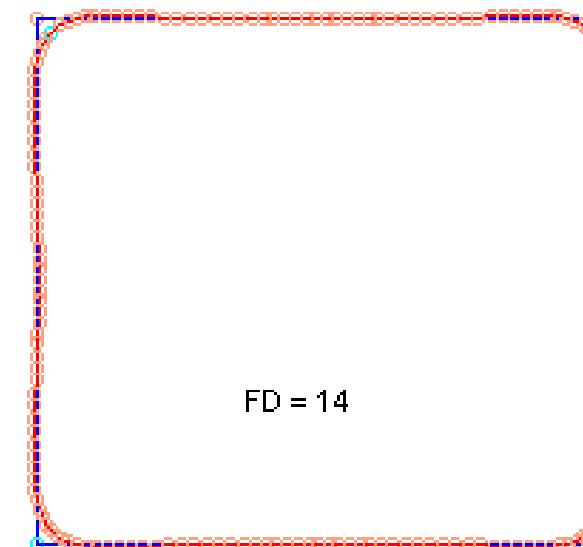
FD = 2



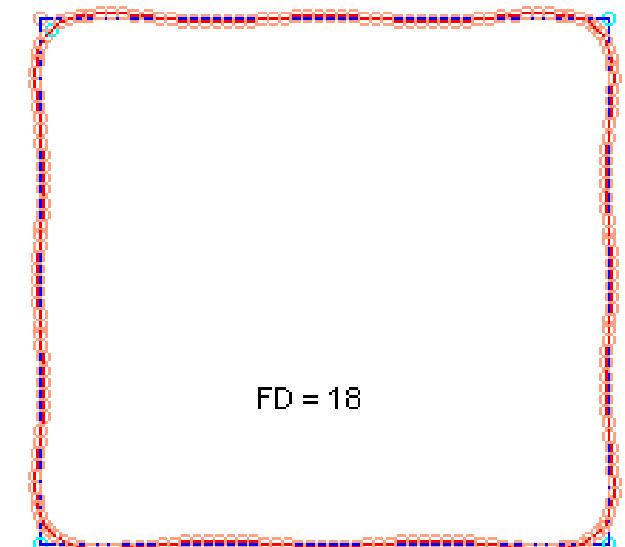
FD = 6



FD = 10

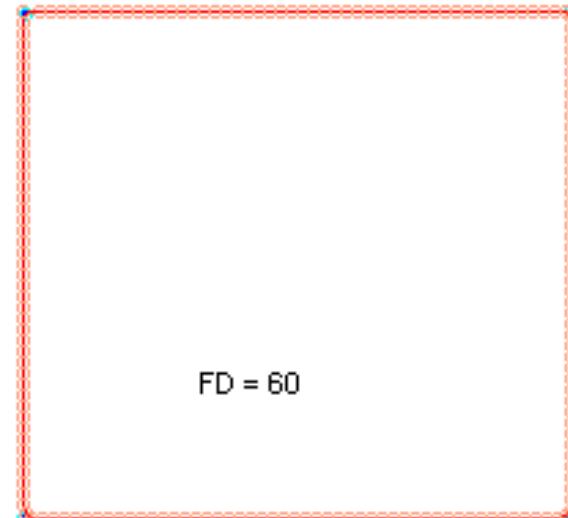
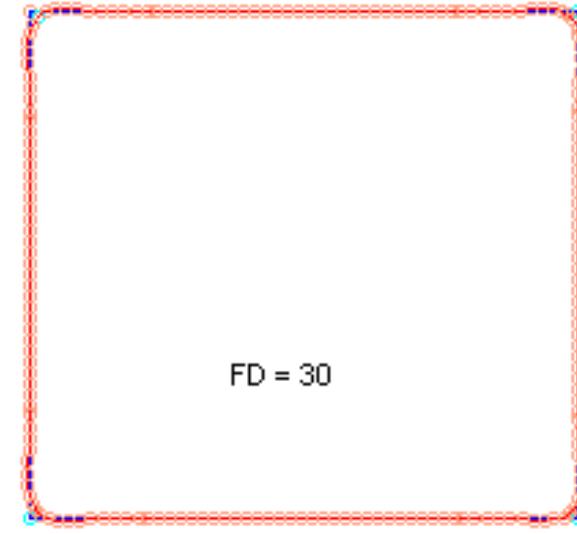
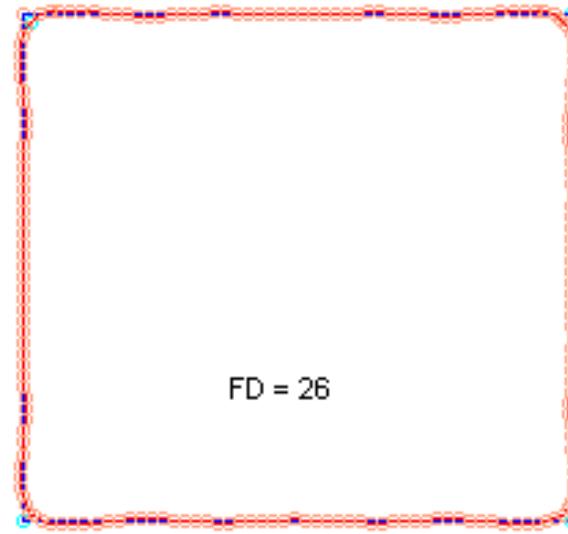
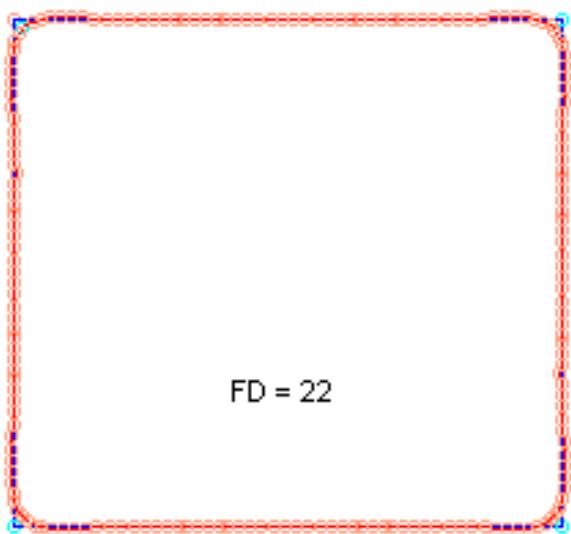


FD = 14

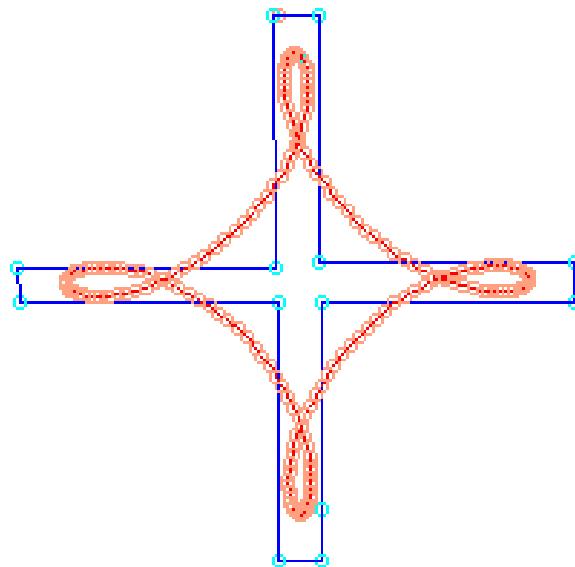


FD = 18

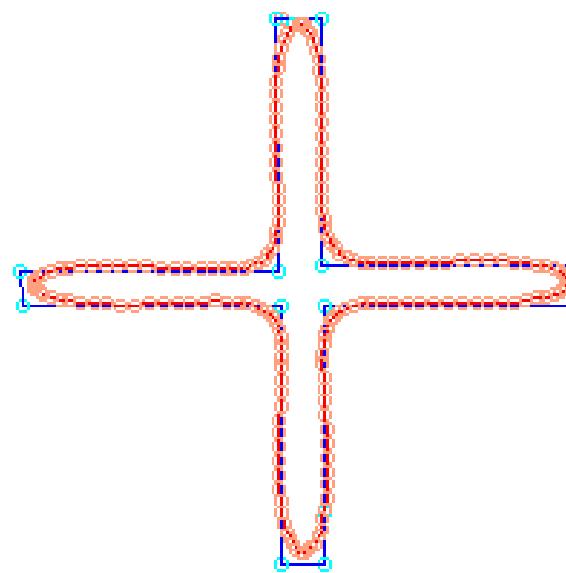
Descripteurs de Fourier



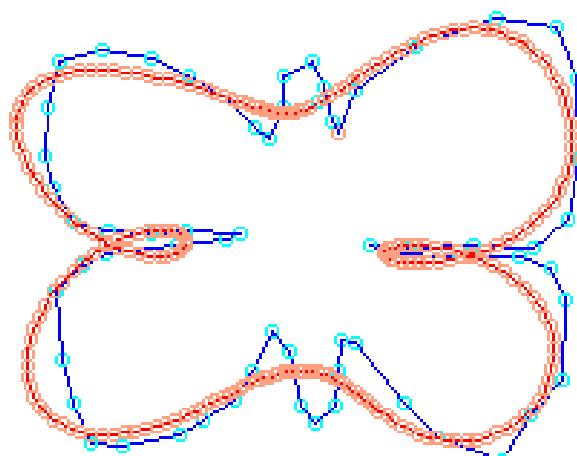
Descripteurs de Fourier



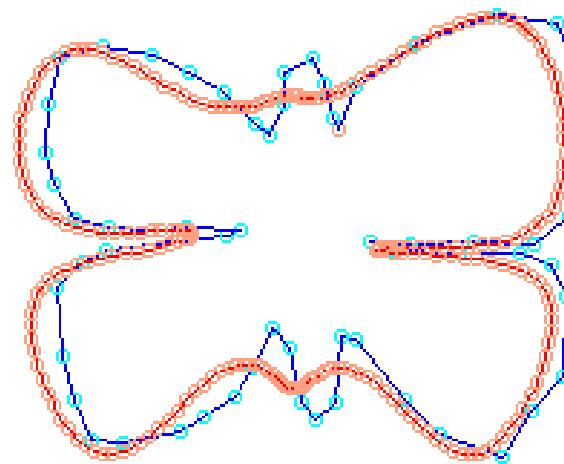
Number of FD: 9



Number of FD: 10

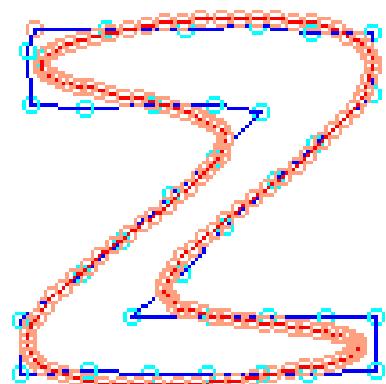


Number of FD: 17

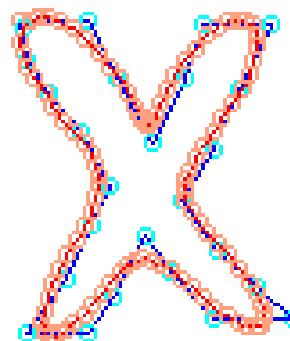


Number of FD: 20

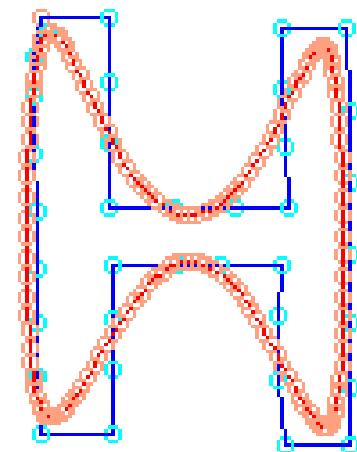
Descripteurs de Fourier



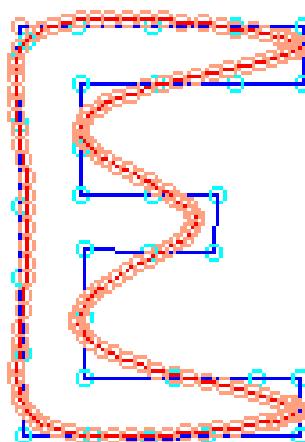
Number of FD: 6



Number of FD: 10

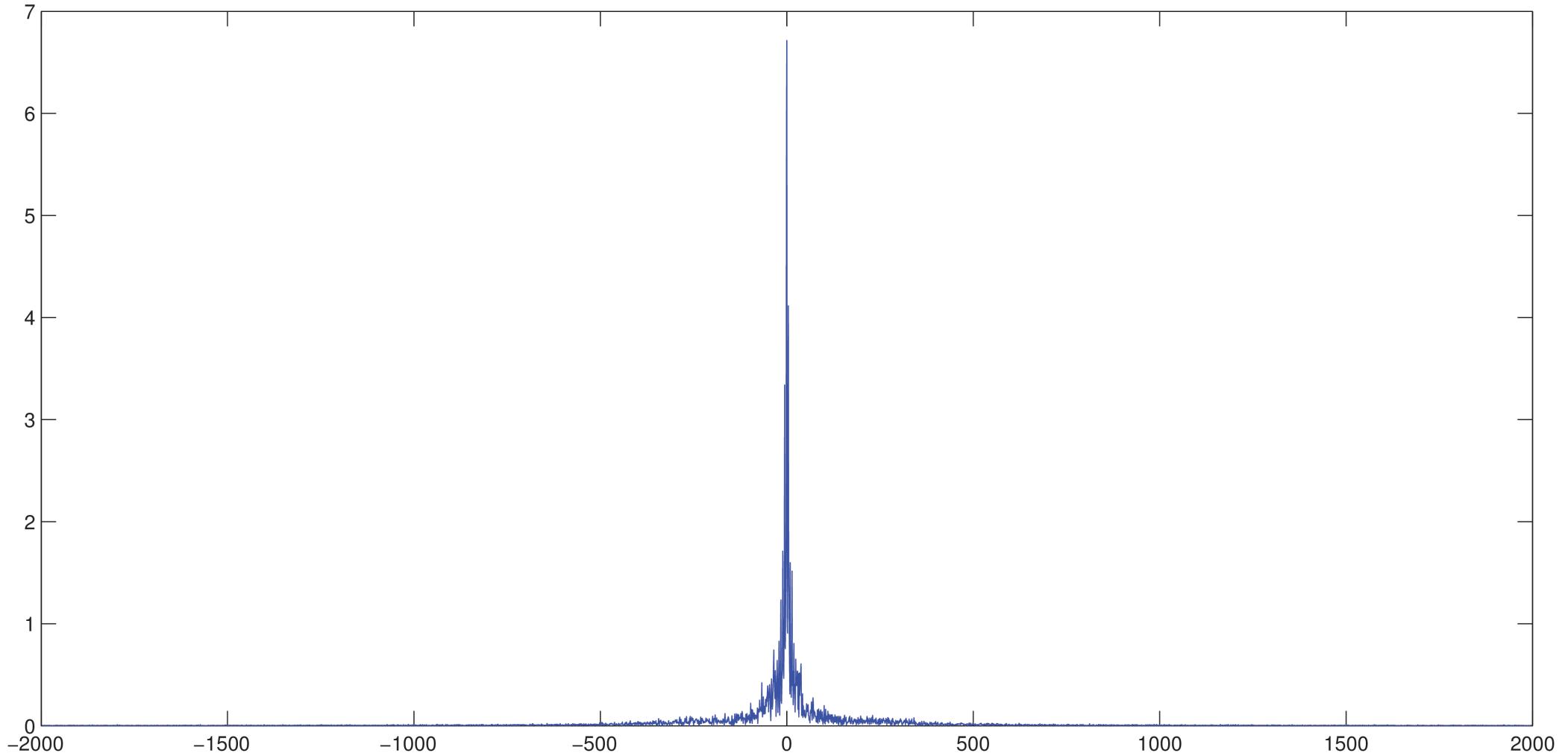


Number of FD: 8

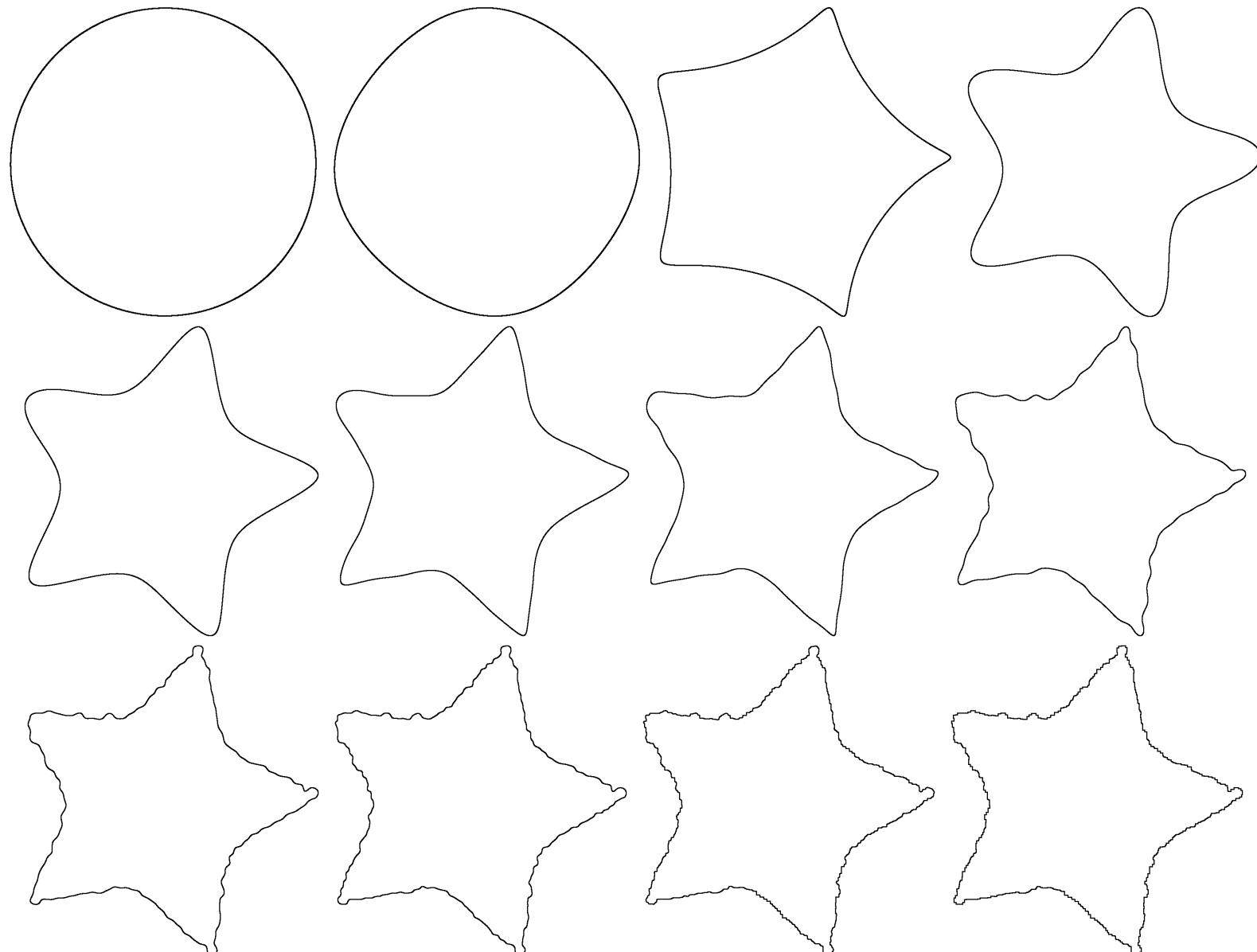


Number of FD: 8

Descripteurs de Fourier



Descripteurs de Fourier



Approximations polynomiales

Préliminaire : approximation d'un nuage de points par une droite unique

- **Approximation par régression linéaire**

$M_i = (x_i, y_i)$, N points

approche par moindres carrés

droite Δ : $y = ax + b$ minimisant l'équation :

$$d_1^2 = \sum_i^N [y_i - (ax_i + b)]^2$$

solution :

$$A = X^\# Y = (X^t X)^{-1} X^t Y$$

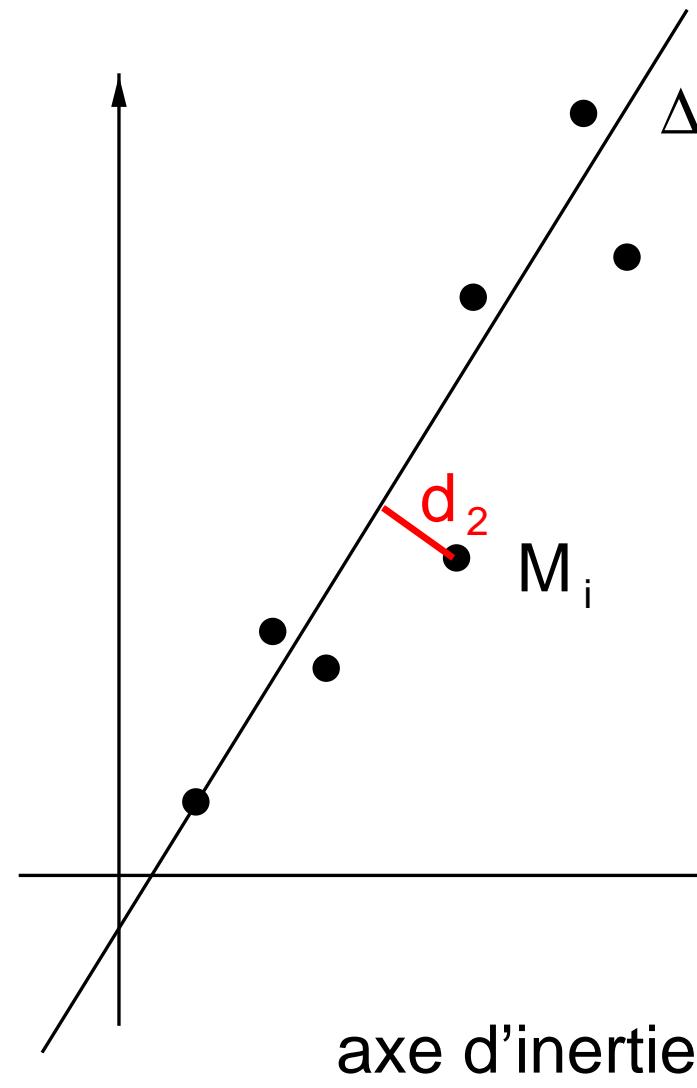
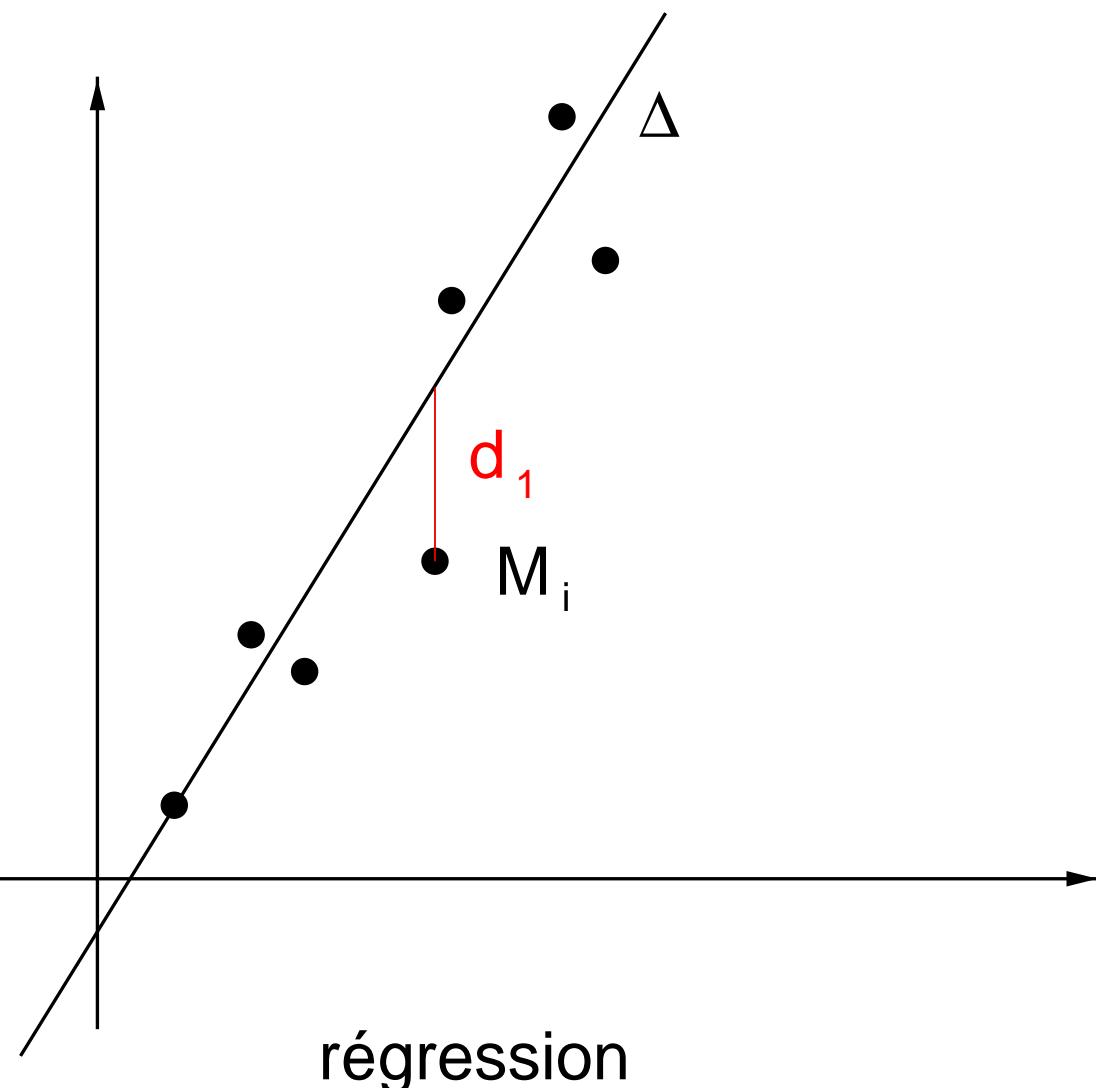
$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ .. & .. \\ 1 & x_N \end{bmatrix}$$

$$Y = [y_1, \ y_2, \ .., \ y_N]^t \quad \text{et} \quad A = [ab]^t$$

ce qui s'écrit :

$$a = \frac{\text{cov}(X, Y)}{\text{var}X} \quad \text{et} \quad b = E(Y) - aE(X)$$

Approximations polynomiales



Approximations polynomiales

- **Approximation par axe principal d'inertie**

droite Δ : minimisant l'équation $ax + by + c = 0$

(avec la contrainte $a^2 + b^2 = 1$) :

$$d_2^2 = \sum_i^N [ax_i + by_i + c]^2 + \lambda(1 - (a^2 + b^2))$$

- (a, b) le vecteur propre associé à la plus petite valeur propre de la matrice de covariance
- c tel que $aE(x) + bE(y) + c = 0$

Approximations polynomiales

Approximations robustes :

problème du bruit dans les observations (*outliers*)

- **Moindres carrés tronqués**

- première approximation aux moindres carrés
- suppression des points les plus éloignés (test statistique)
- itération...

- **Moindres médians**

- tirage aléatoire d'ensembles de points
- calcul des paramètres
- sélection des valeurs médianes
- au maximum 20% d'outliers

Approximations polynomiales

Approximations robustes :

- **Moindres médians bis**
 - tirage aléatoire d'ensembles de points
 - calcul des paramètres
 - calcul des erreurs pour tous les points et sélection de l'erreur médiane
 - sélection des paramètres correspondant à la plus petite valeur médiane
 - au maximum 50% d'outliers
- **Méthode RanSac (Random Sample Consensus)**
 - tirage aléatoire d'ensembles de points
 - calcul des paramètres
 - calcul du nombre de points dont l'erreur est inférieure à un seuil ϵ
 - sélection des paramètres maximisant ce nombre

Approximations polynomiales

Approximations robustes :

- **Les M-estimateurs**

- initialisation (par exemple aux moindres carrés)
- calcul des erreurs (distances) pour chaque point
- pondération des points en fonction de leur distance (quadratique tronquée)
- nouveau calcul des paramètres
- itération du procédé

Approximations polygonales des contours

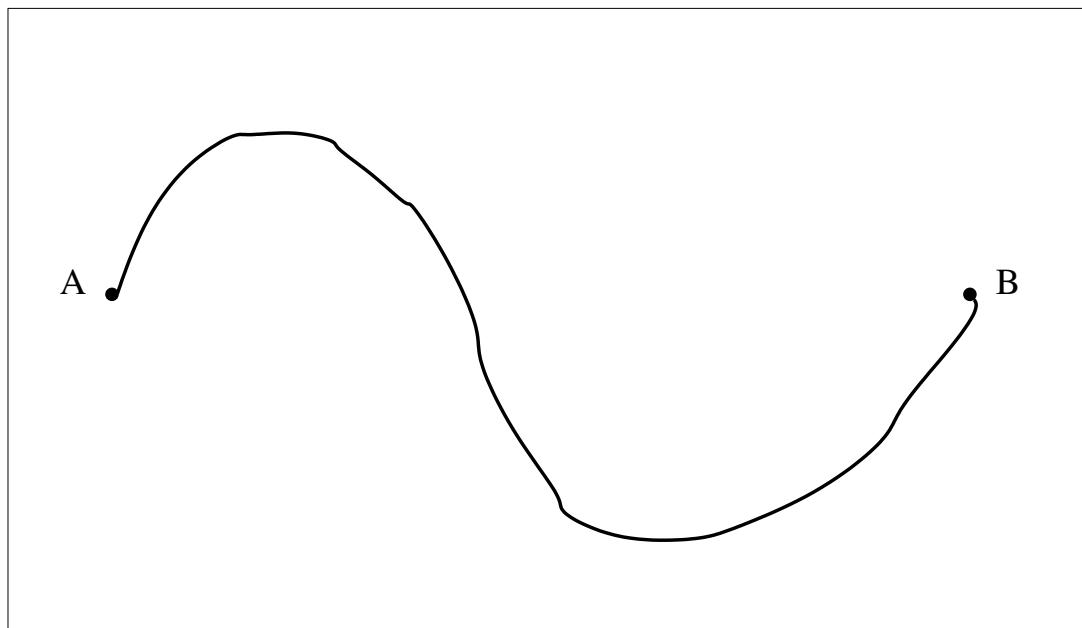
réduction d'une courbe Γ continue ou finement échantillonnée à une ligne polygonale

- **Algorithme de la corde**

- subdivision récursive ou itérative à chaque étape
- sommets du polygone = points de Γ les plus éloignés des cordes précédemment tirées
- arrêt du processus : nouvelle distance candidate inférieure à un seuil ϵ fixé.

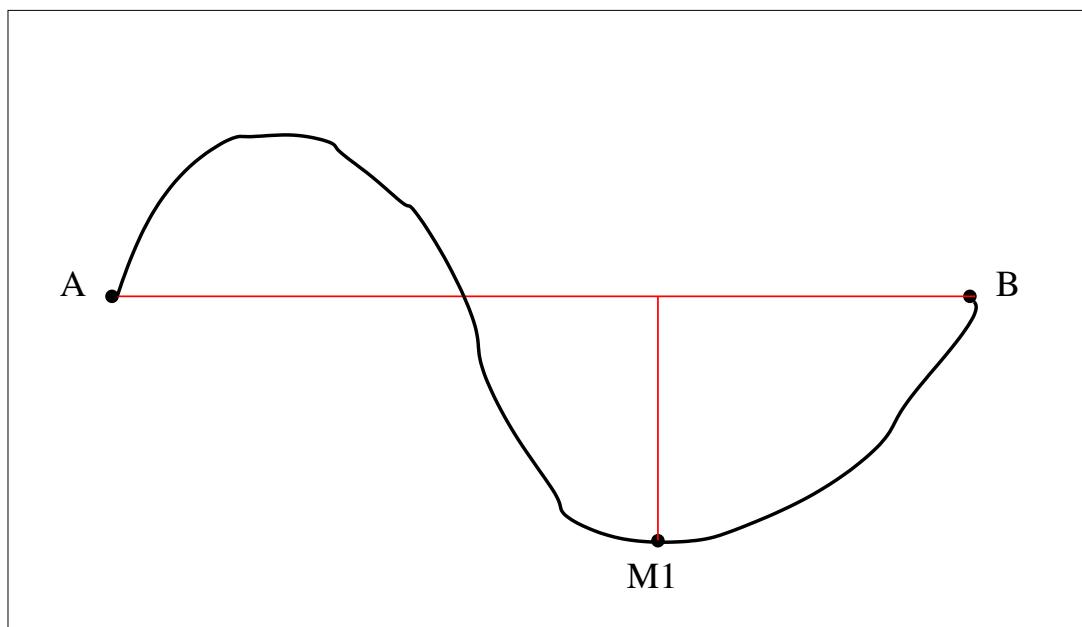
Approximations polygonales des contours

- Algorithme de la corde



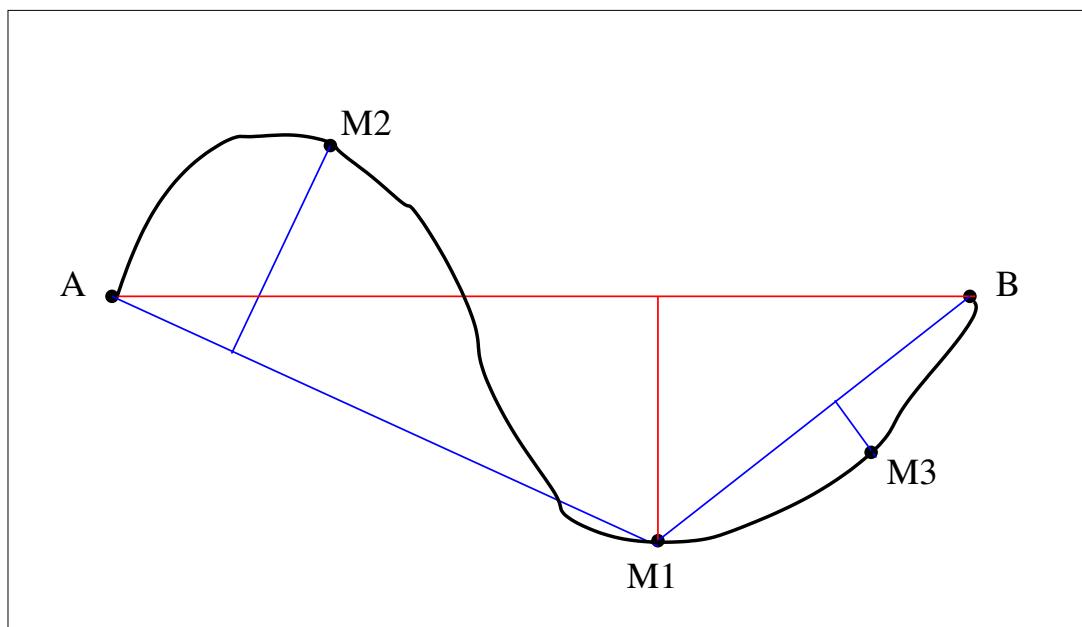
Approximations polygonales des contours

- Algorithme de la corde



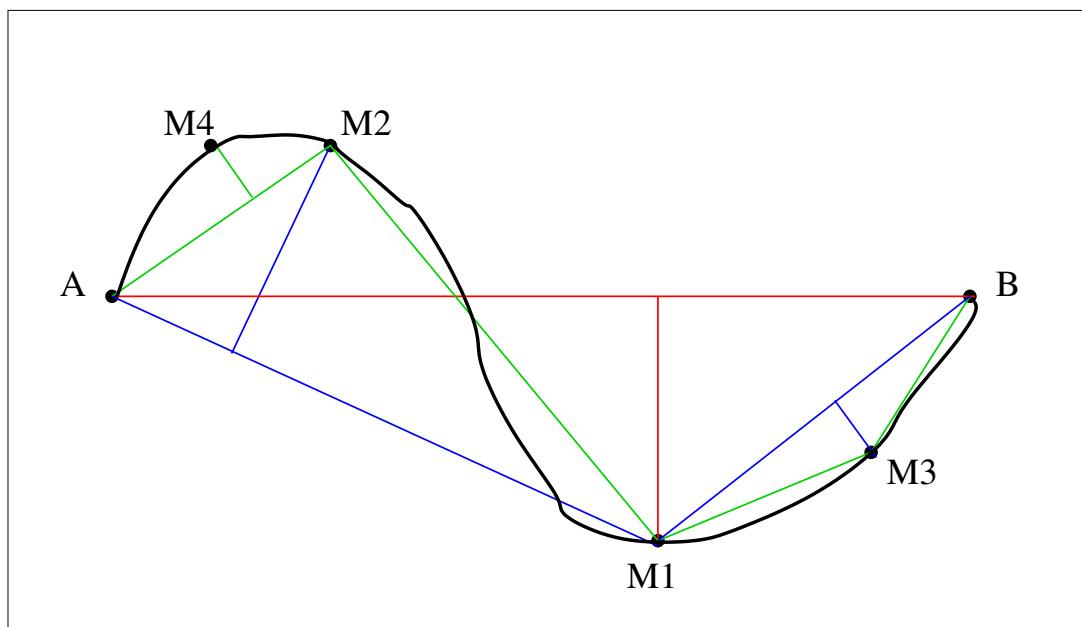
Approximations polygonales des contours

- Algorithme de la corde



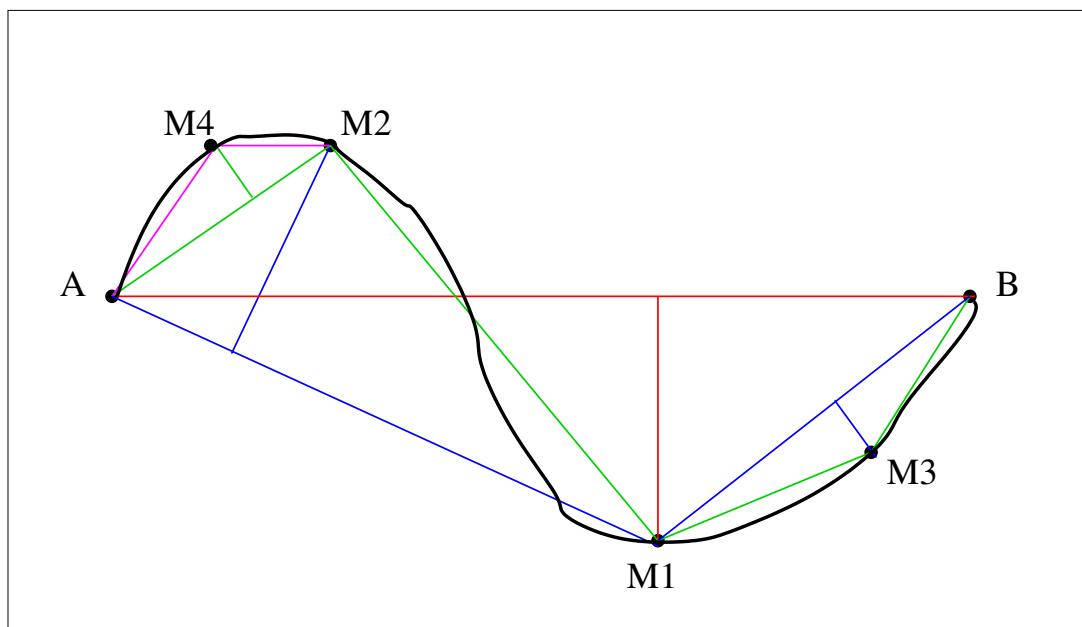
Approximations polygonales des contours

- Algorithme de la corde



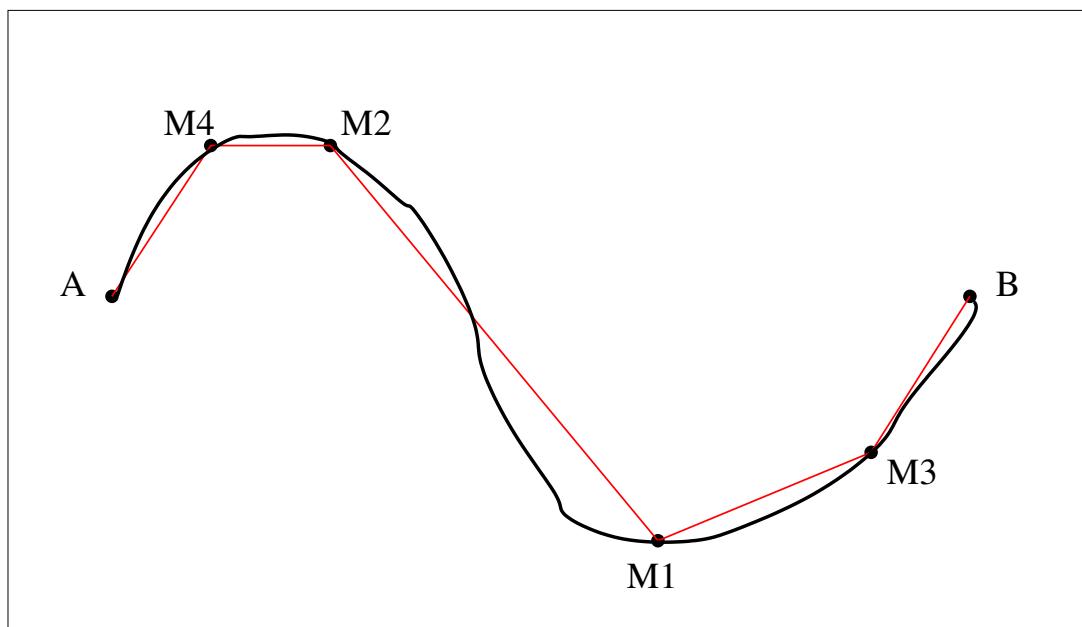
Approximations polygonales des contours

- Algorithme de la corde



Approximations polygonales des contours

- Algorithme de la corde



Transformation de Hough

Reconnaissance de courbes *paramétrées*

- **Principe**

associer au plan image un espace d'accumulateurs lié aux paramètres de la courbe recherchée

Détection de courbes dans le plan image (espace \mathcal{I})

\Updownarrow

Recherche de maxima dans l'espace des accumulateurs (espace \mathcal{H})

Transformation de Hough

Recherche de droites

- **TH de 1 à m**

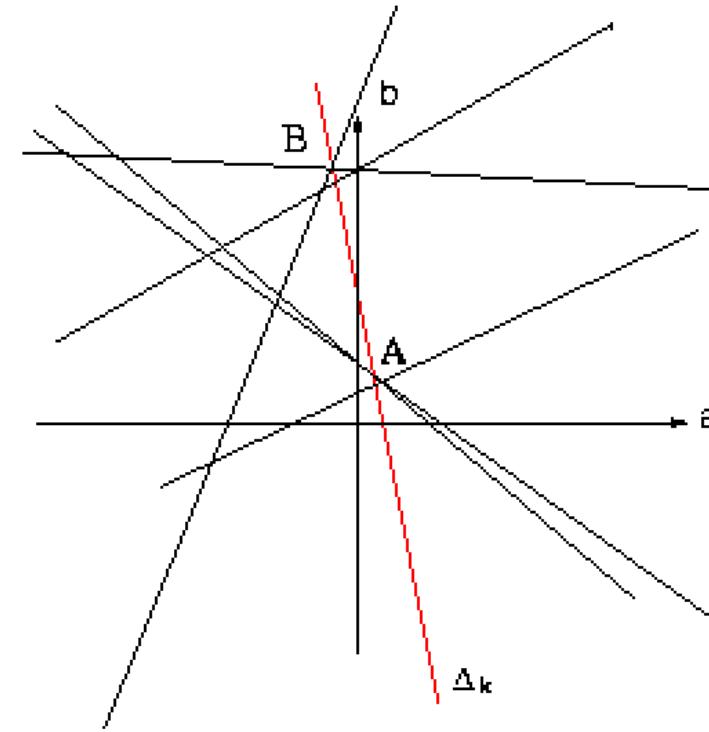
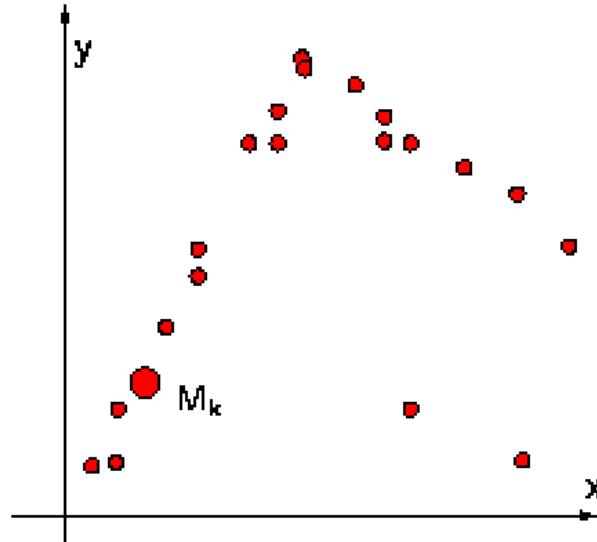
un point de l'espace image \Rightarrow une courbe dans l'espace des paramètres

soit $M_i = (x_i, y_i)$

si $M_i \in D_i$ alors $y_i = ax_i + b$ soit $b = -x_i a + y_i$

\Rightarrow équation d'une droite dans l'espace (a, b)

(à un point de \mathcal{I} est associé m points de \mathcal{H})



recherche des intersections des droites dans \mathcal{H}

Transformation de Hough

Recherche de droites

- **TH de m à 1**

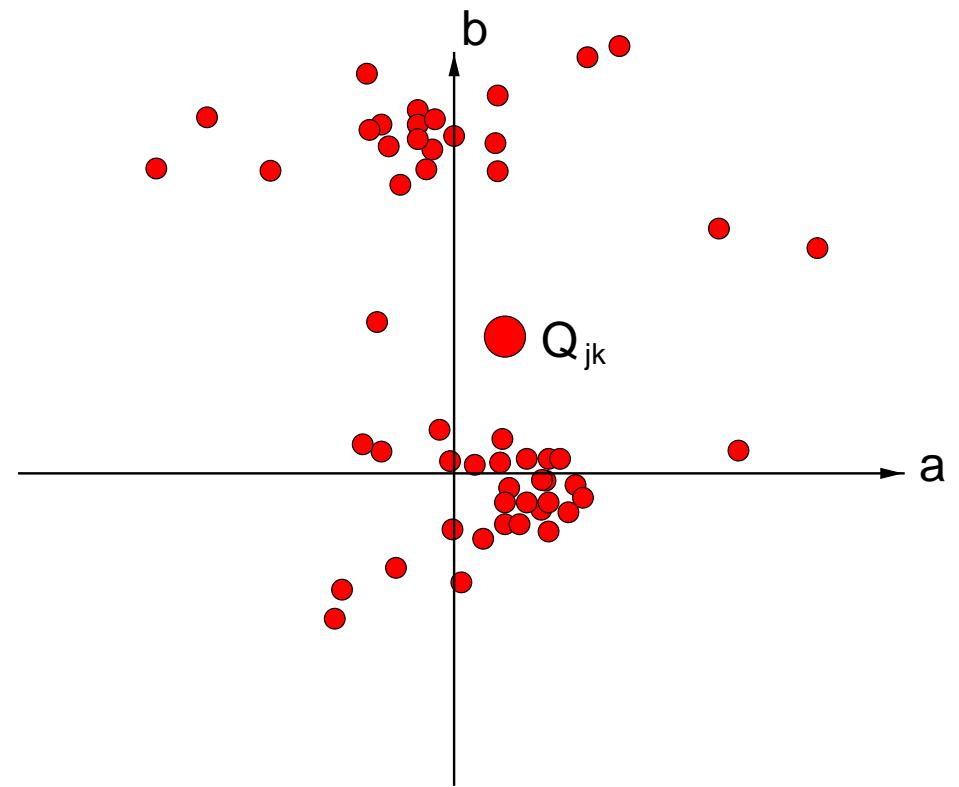
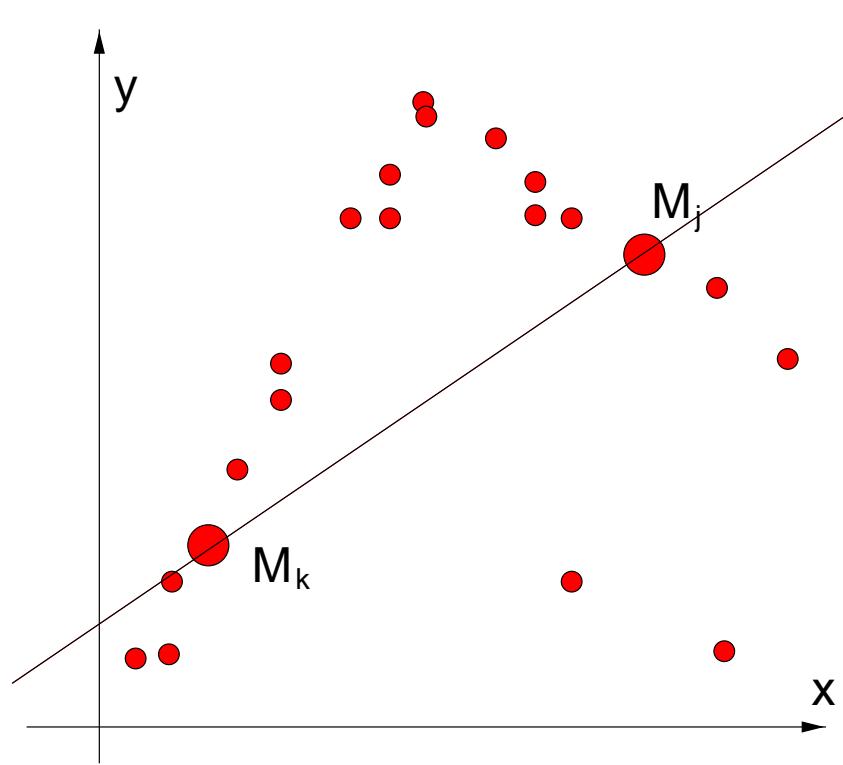
un bipoint de l'espace image \Rightarrow un point dans l'espace des paramètres

soit M_i, M_j de \mathcal{I} \rightarrow une seule droite $\Delta_{ij} \rightarrow$ un point Q_{ij} de \mathcal{H}

$$a_{ij} = \frac{y_i - y_j}{x_i - x_j}$$

$$b_{ij} = \frac{x_i y_j - x_j y_i}{x_i - x_j}$$

(à m points de \mathcal{I} est associé 1 point de \mathcal{H})



recherche des points les plus représentés dans \mathcal{H}

$(m=2!...)$

$(\text{TH de } 1 \text{ à } m \Leftrightarrow \text{TH de } m \text{ à } 1)$

Transformation de Hough

- **Discrétisation de \mathcal{H}**

- accumulateur = case de \mathcal{H}
- hypothèse = vote dans \mathcal{H} = incrémentation de l'accumulateur

Choix de la taille des cases délicat :

- accumulateurs trop grands = mauvaise précision dans la détection
- accumulateurs trop petits = peu de votes si courbe “bruitée”
- problème de la taille de l'espace mémoire
(surtout si de nombreux paramètres pour la forme)

Transformation de Hough

o Paramétrisation de la courbe

Nécessité d'une bonne paramétrisation des courbes

- Ex : pour les droites $y = ax + b$, $a, b \in]-\infty; +\infty[$

$a \in [0, 1[\Rightarrow 25\%$ des droites du plan

$a \in [1, +\infty[\Rightarrow 25\%$ des droites

\Rightarrow taille des cellules très inégale

Autre paramétrisation : $\rho = x_i \cos \theta + y_i \sin \theta$

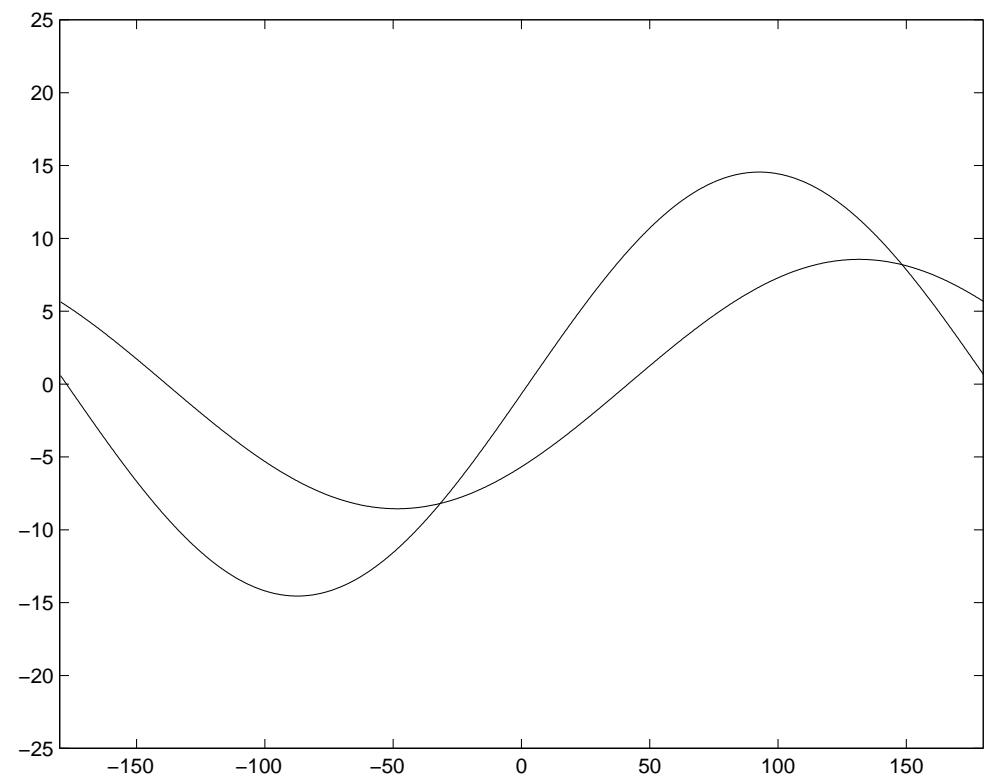
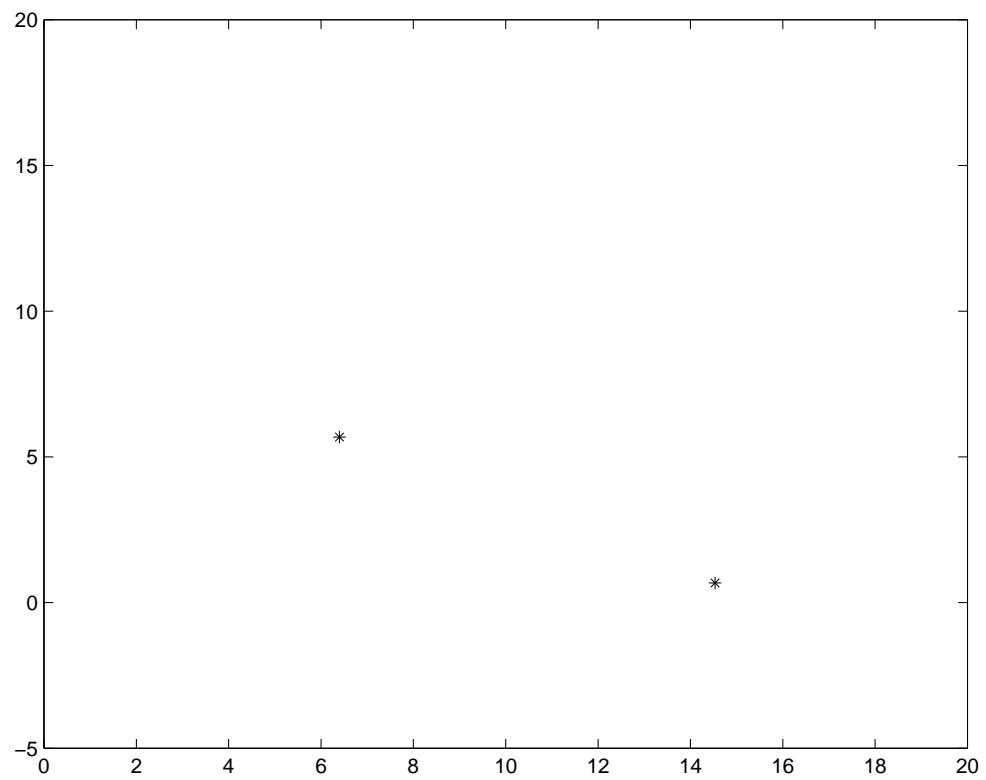
Transformée de Hough 1 à m :

$\Rightarrow \rho = f(\theta)$ sinusoïde

$$\rho = \sqrt{x_i^2 + y_i^2} \cos(\theta + \phi)$$

$$\cos(\phi) = x_i / \sqrt{x_i^2 + y_i^2}$$

$$\Delta\theta = [-\pi, +\pi[\text{ et } \Delta\rho = [0, \sqrt{2}L] \text{ (} L \text{ côté de l'image)}$$



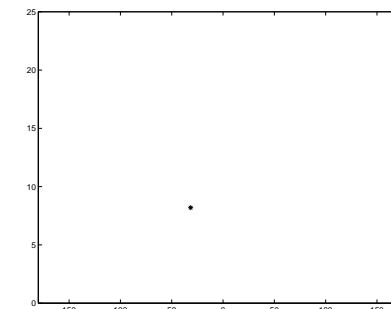
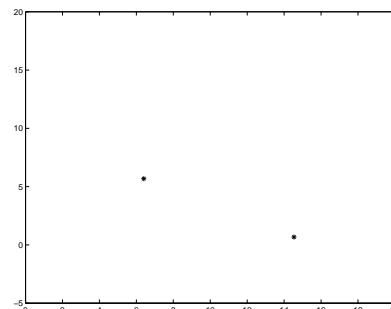
Transformation de Hough

- **Paramétrisation de la courbe**

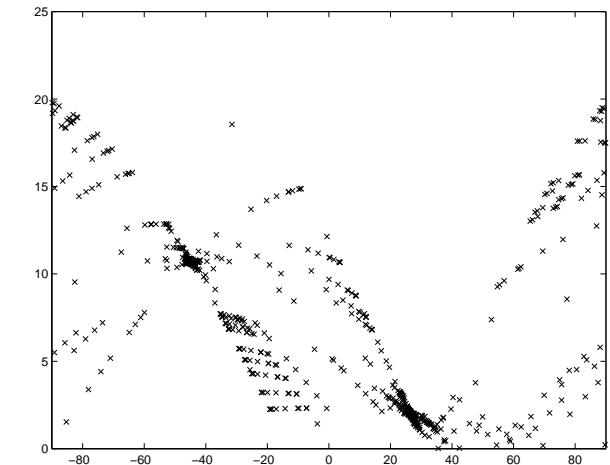
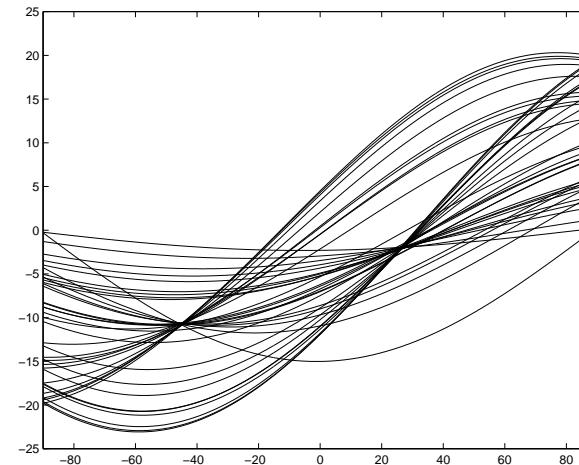
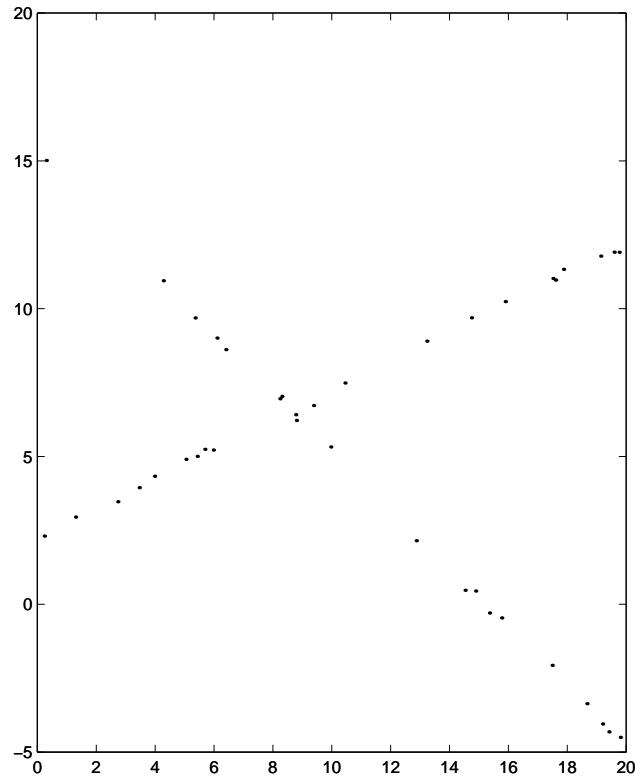
Transformée de Hough m à 1 :

doublet M_i, M_j de \mathcal{I} \Rightarrow point Q_{ij} de \mathcal{H}

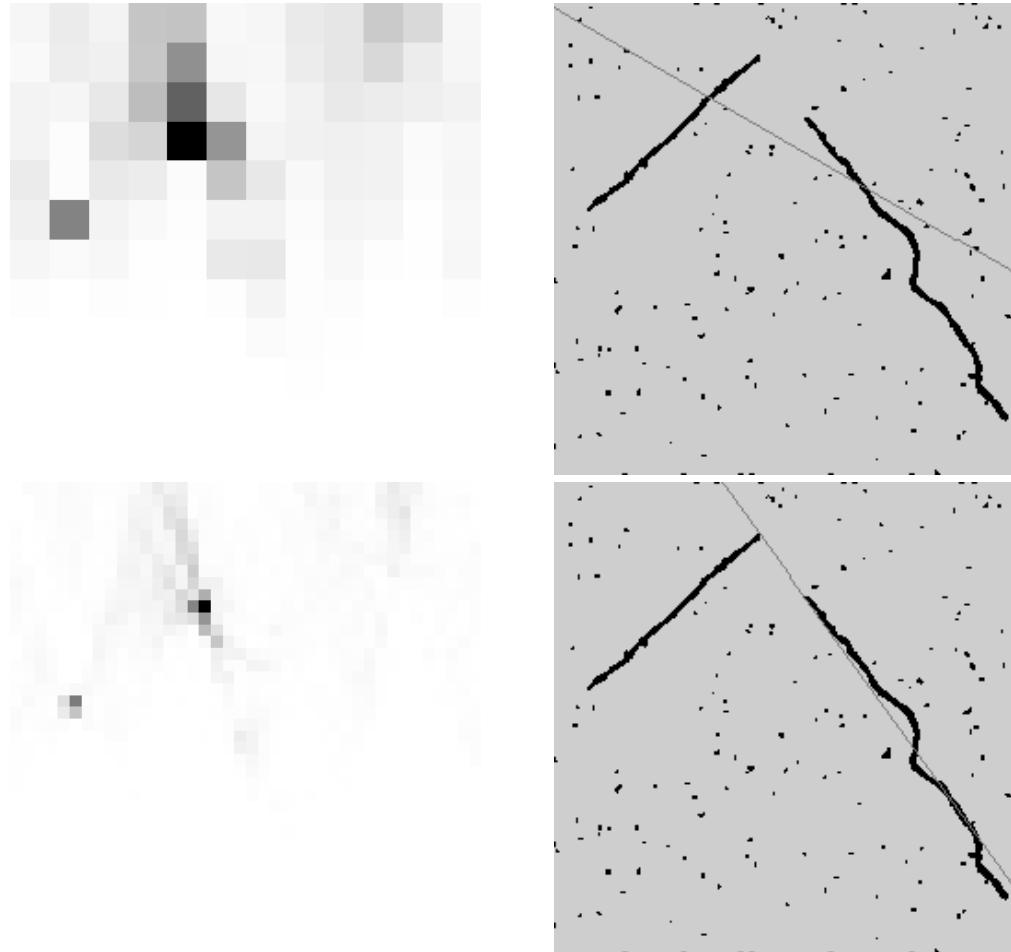
$$\rho_{ij} = \frac{|x_i y_j - x_j y_i|}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \quad \text{et} \quad \theta_{ij} = -\text{Arctg} \left[\frac{x_j - x_i}{y_j - y_i} \right]$$



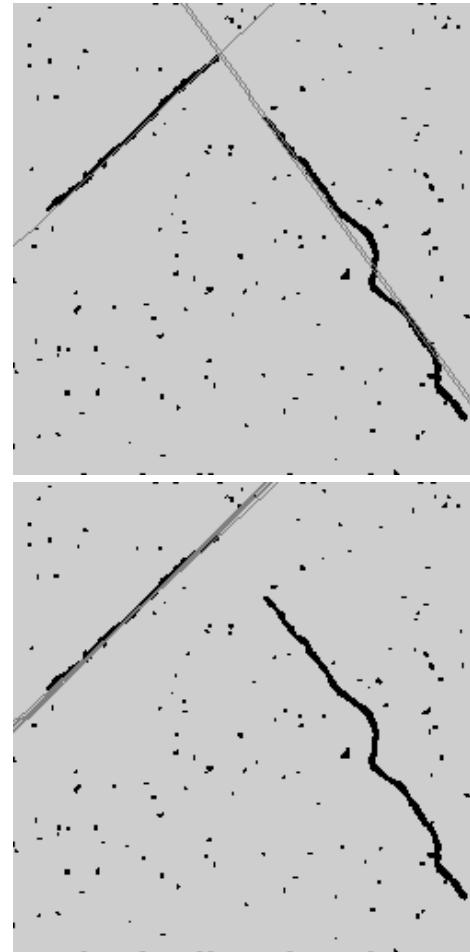
Transformation de Hough - exemple



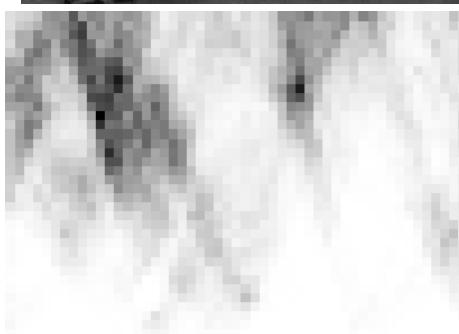
Transformation de Hough - exemple



Transformation de Hough - exemple



Transformation de Hough - exemple



Transformation de Hough

- **Détection d'objets par TH**

- cercles (3 paramètres : centre + rayon)
- ellipses (5 paramètres centre + rayon+ direction de l'axe + ellipticité)
- paraboles (3 paramètres si on connaît la direction de l'axe, 5 sinon)

Transformation de Hough - exemple

