
Introduction à la détection d'objets dans les images

Jean-Philippe Tarel

LCPC Paris

Tarel@lcpc.fr

Séminaire Master STIC- Informatique

Spécialité MOCS, finalité Recherche

Cours de Vision par Ordinateur

Plan

1. Détecter un objet modélisé par une image?
2. Détecter un objet avec une modélisation Gaussienne de la variabilité de son aspect?
3. Détecter un objet avec une modélisation SVM de la variabilité de son aspect?
4. Détecter un objet avec une modélisation paramétrée de la variabilité de son aspect?

Partie 1

1. Détecter un objet modélisé par une image?
 - Méthode par corrélation.
 - Transformée de Hough généralisée.

Exemple simple

Les images à traiter :



L'objet à détecter (modélisé par une image) :



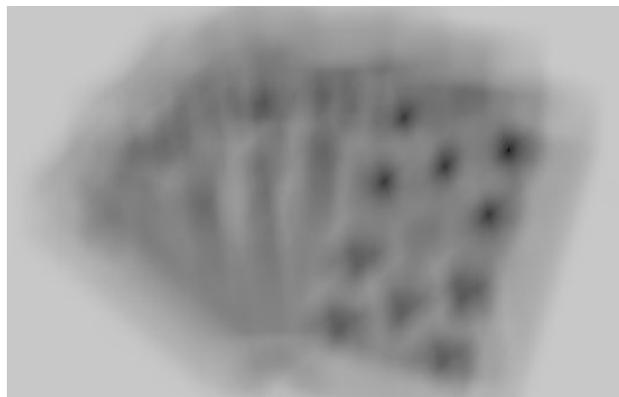
Méthode par corrélation (1)

Image originale :



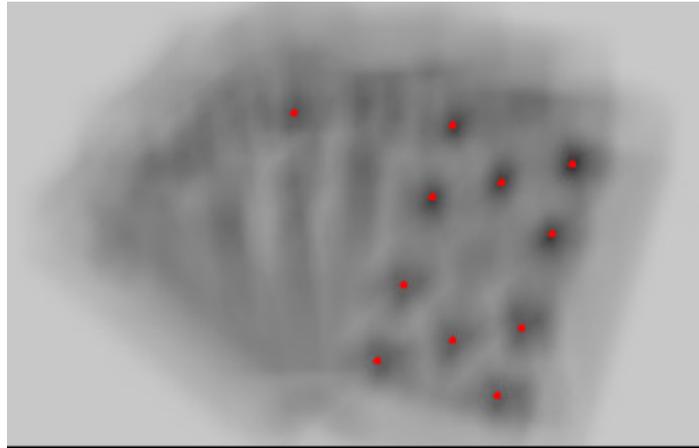
Calcul de la carte de corrélation :

Pour chaque position, calculer la distance entre l'image modèle et la zone de l'image associée:

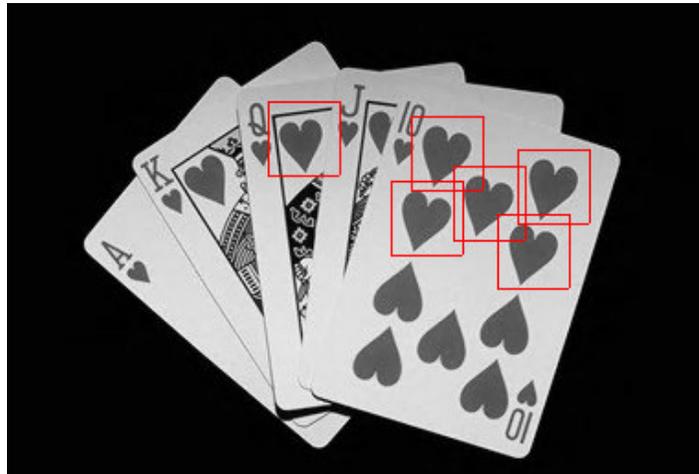


Méthode par corrélation (2)

Extraction des minima locaux :

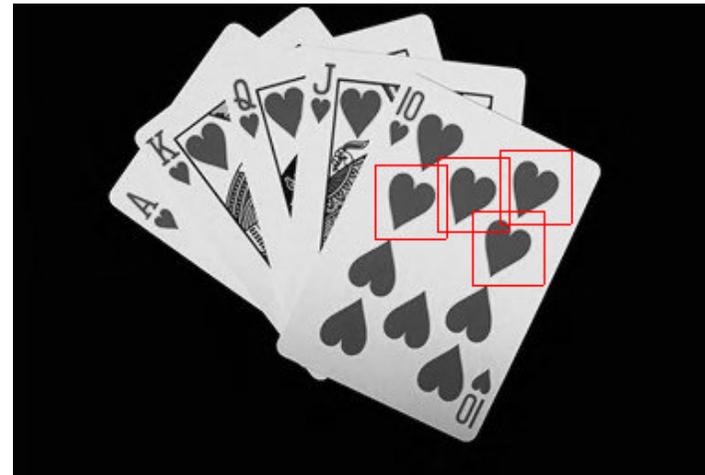
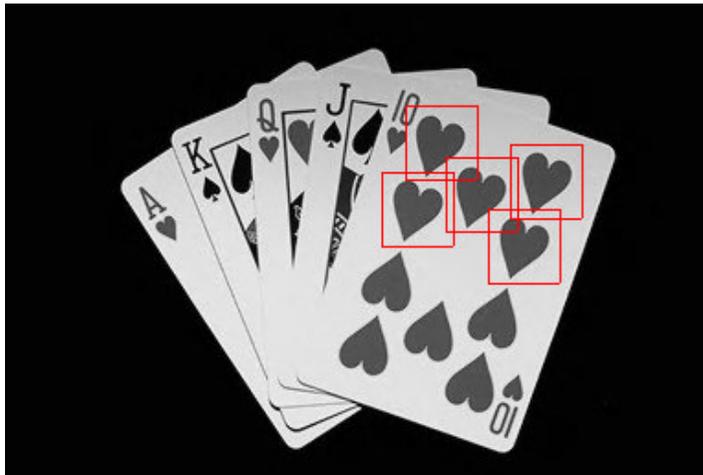
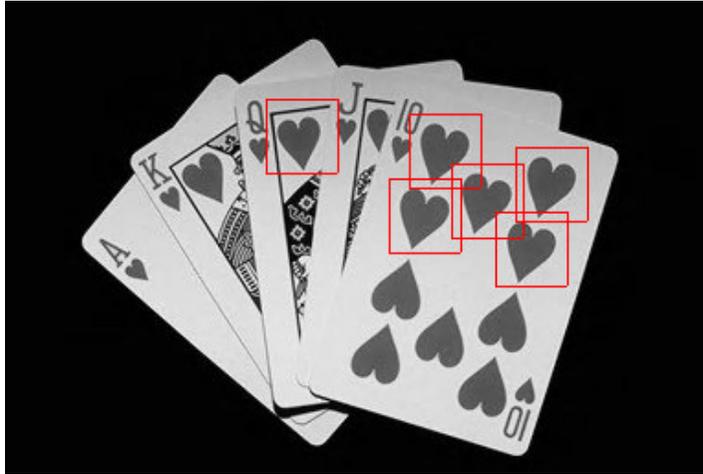


et décision par seuillage sur la valeur de la distance :



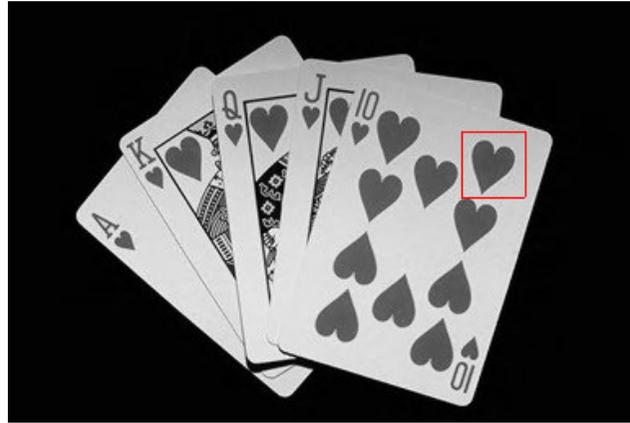
Détection par corrélation

Résultats :

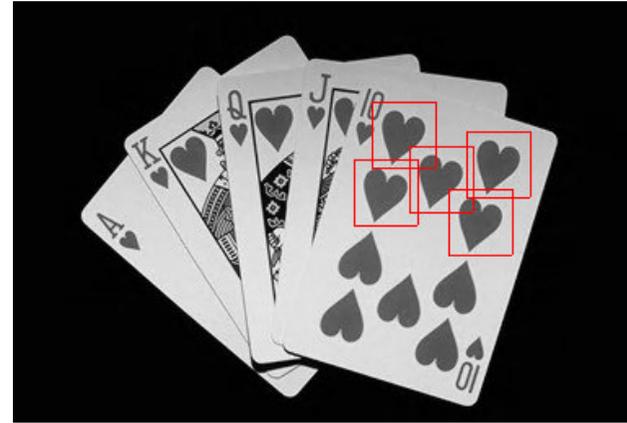


Temps de détection par image : environ 10s.

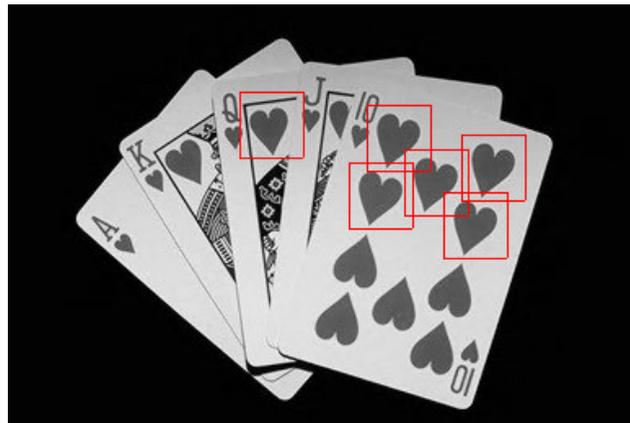
Choix du seuil de décision



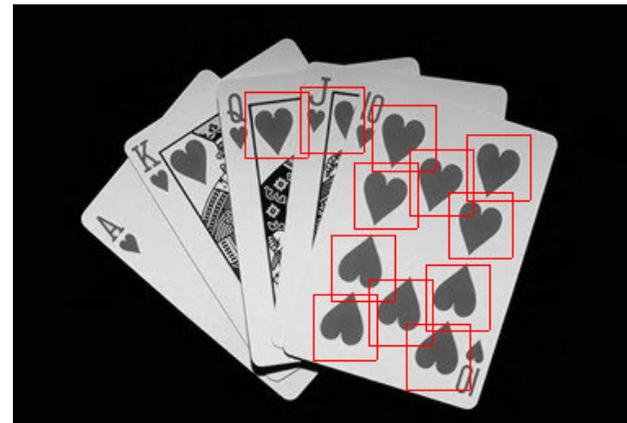
Seuil de décision=10



Seuil de décision=20



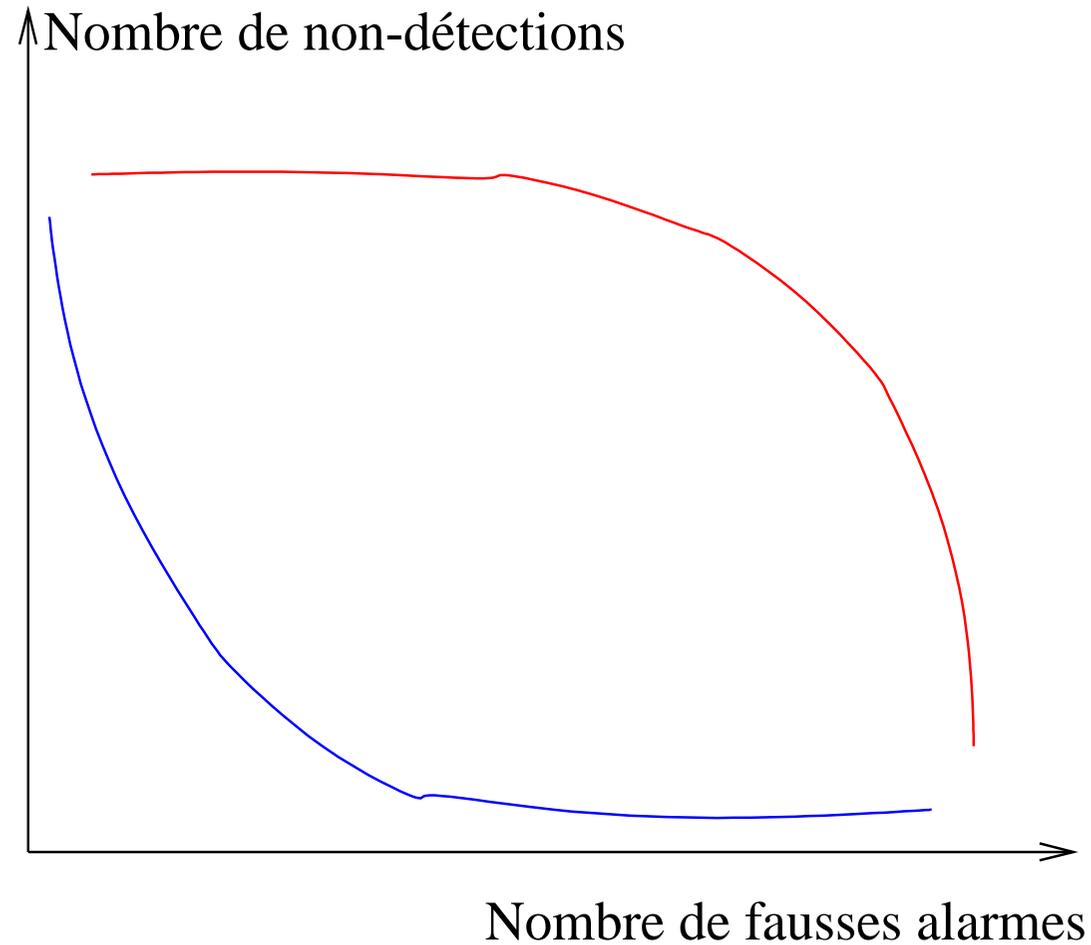
Seuil de décision=30



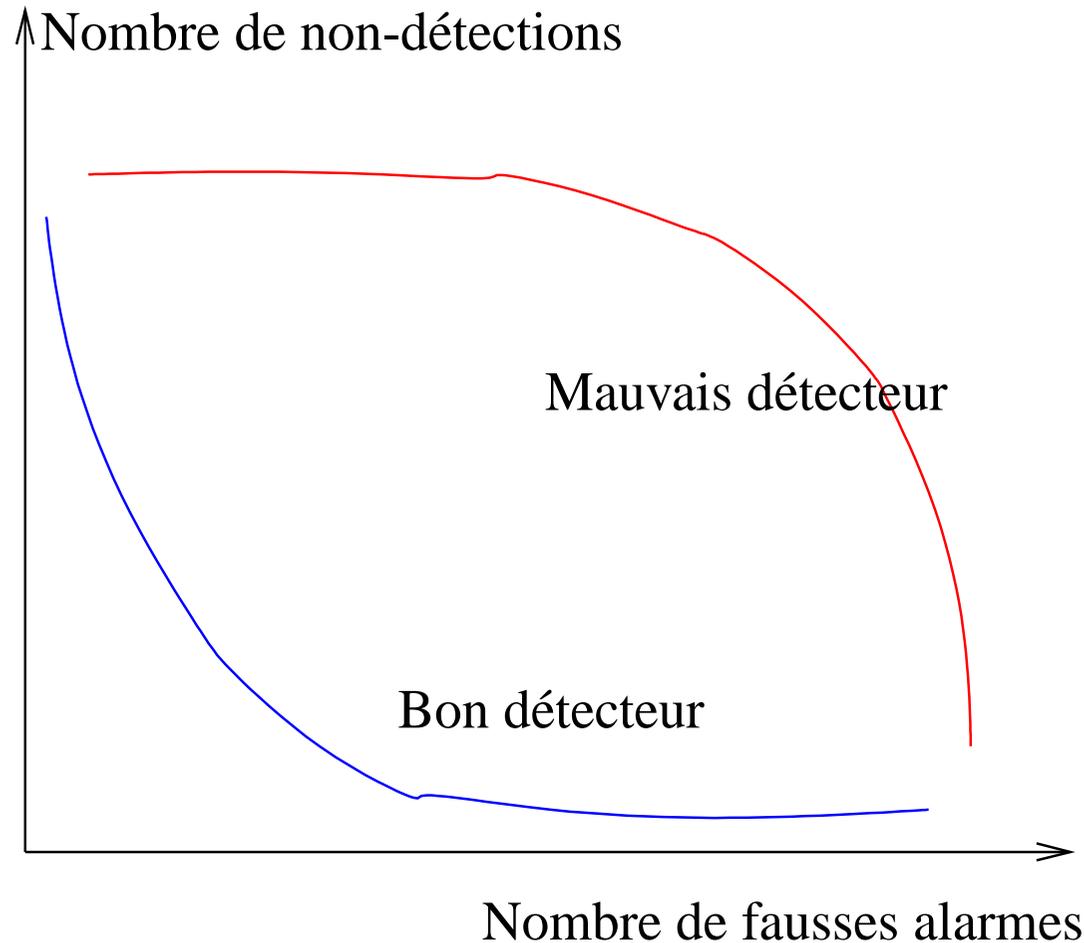
Seuil de décision=40

⇒ Compromis entre le nombre de fausses alarmes et le nombre de non-détections.

Comparaison de deux détecteurs



Comparaison de deux détecteurs



La comparaison entre deux détecteurs doit être réalisée par la comparaison des courbes fausse-alarme/non-détection, obtenues en faisant varier le seuil de décision.

Choix de la distance?

- On suppose implicitement que les perturbations sont additives, indépendantes et identiquement distribuées (iid) :

$$I_{obs}(x, y) = I_{réf}(x, y) + b$$

en chaque pixel, avec b une variable aléatoire centrée.

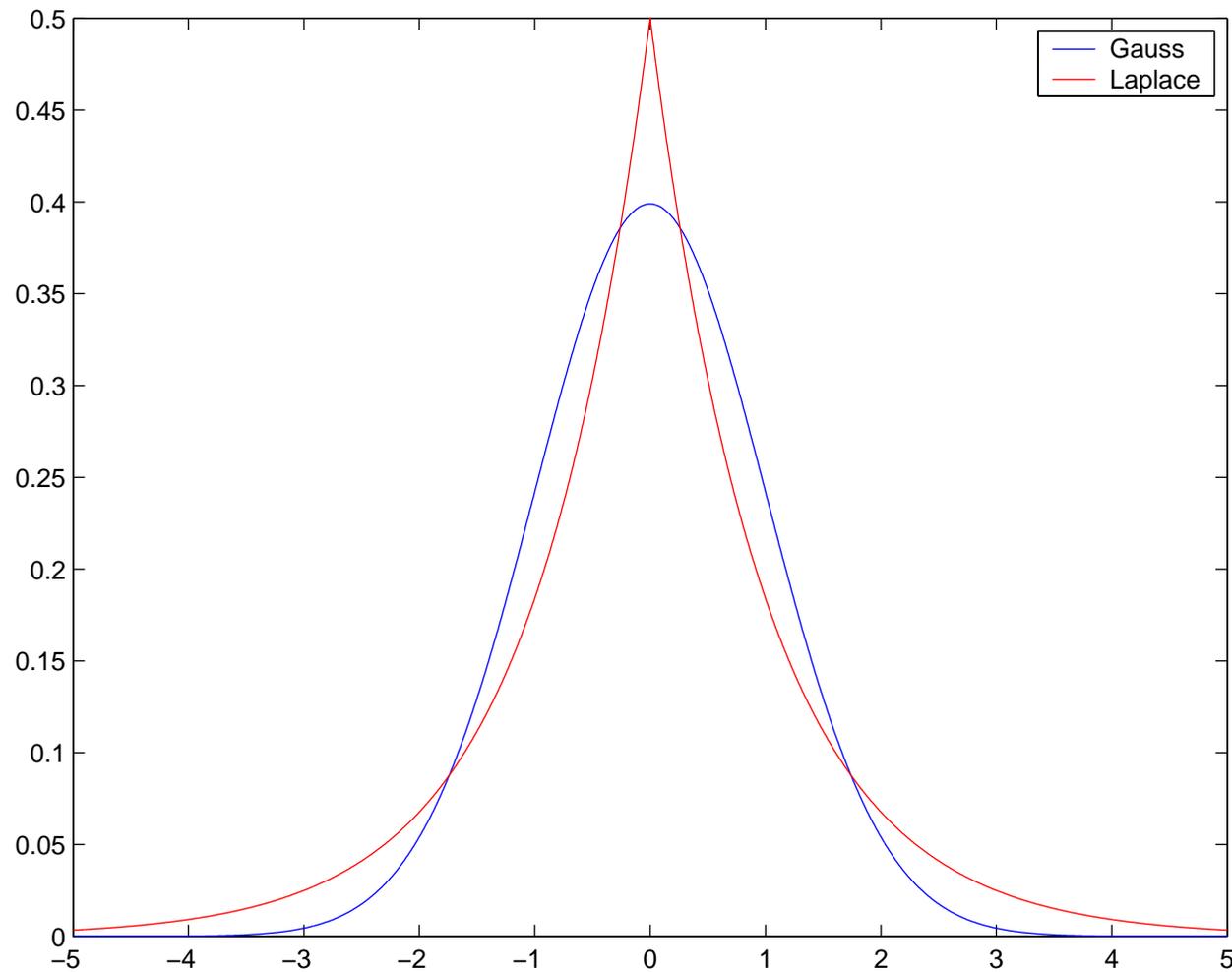
- Lorsque b est Gaussienne $b \propto e^{-\frac{b^2}{2\sigma^2}}$, il faut utiliser la distance Euclidienne :

$$D(I_{obs}, I_{réf}) = \sqrt{\sum_{(x,y)} (I_{obs}(x, y) - I_{réf}(x, y))^2}$$

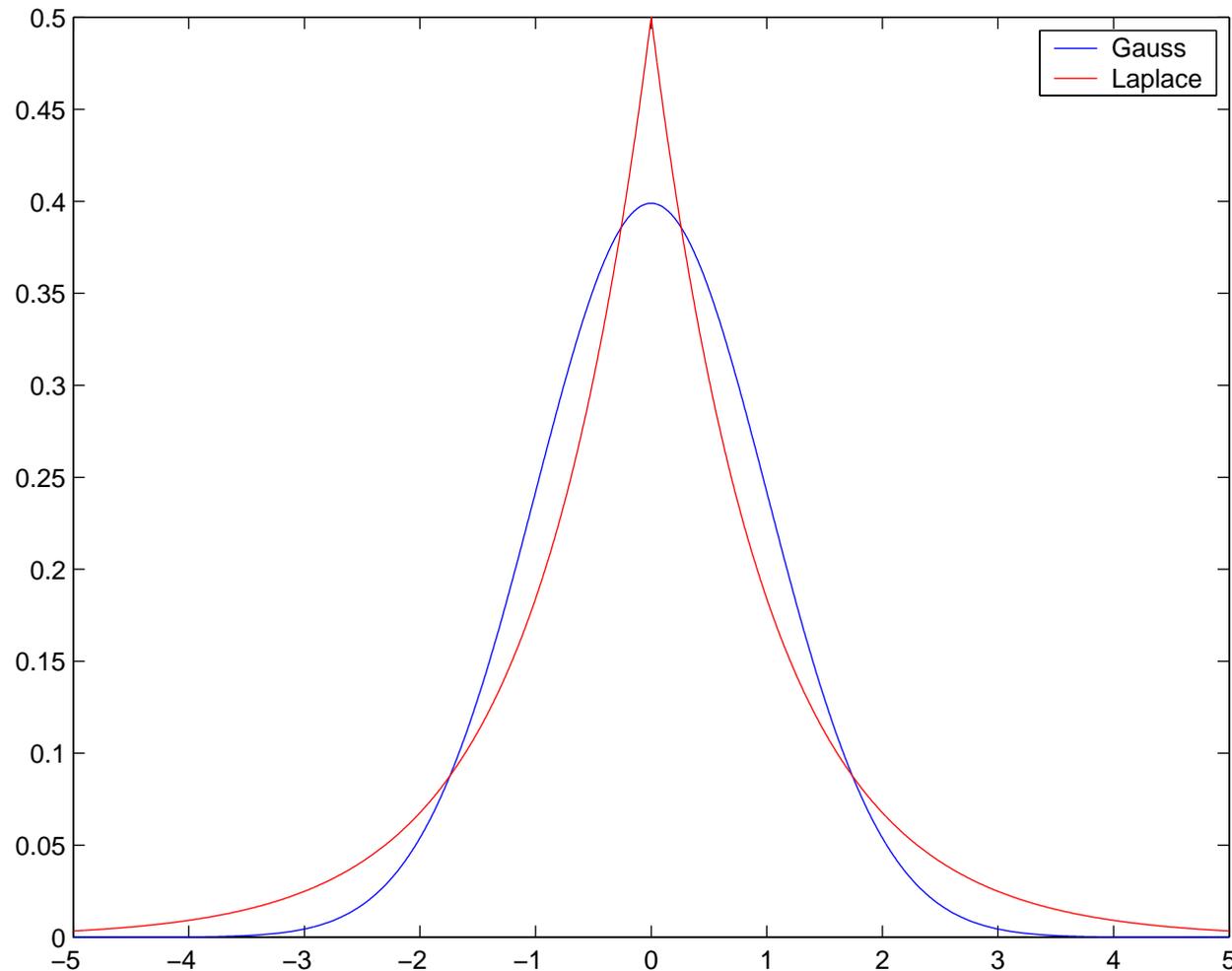
- Lorsque b est Laplacienne $b \propto e^{-\frac{|b|}{\sigma}}$, il faut utiliser la distance L1 :

$$D(I_{obs}, I_{réf}) = \sum_{(x,y)} |I_{obs}(x, y) - I_{réf}(x, y)|$$

Laplace/Gauss (1)



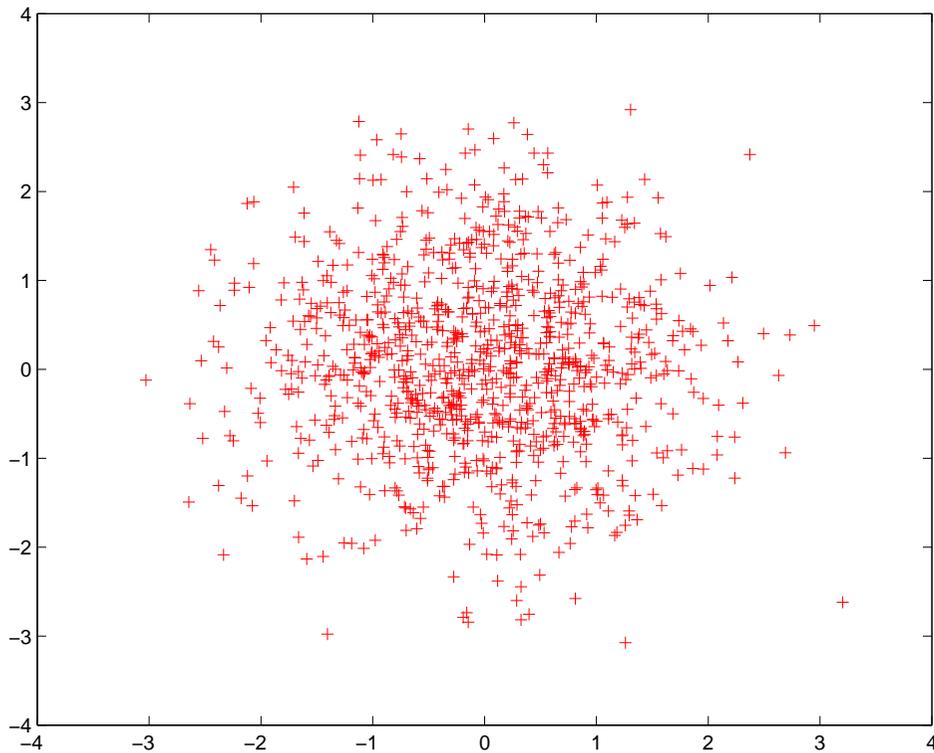
Laplace/Gauss (1)



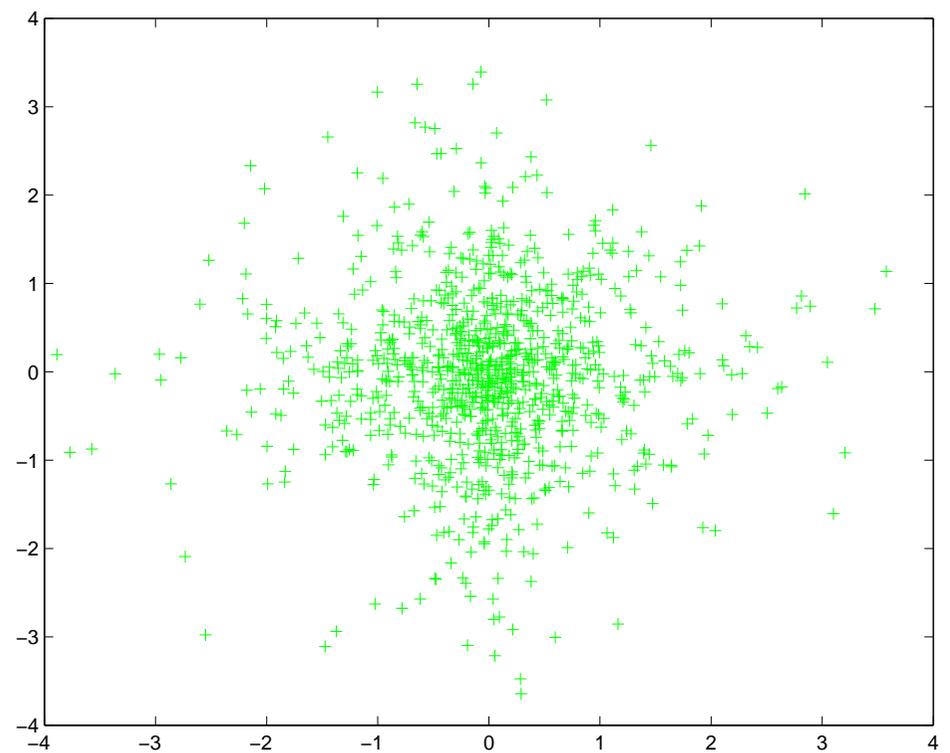
⇒ La distance L_1 donne des résultats plus robustes aux grandes perturbations que la distance L_2 , pour une même échelle.

Laplace/Gauss (2)

Bruits de moyenne=0, variance=1:



Gauss



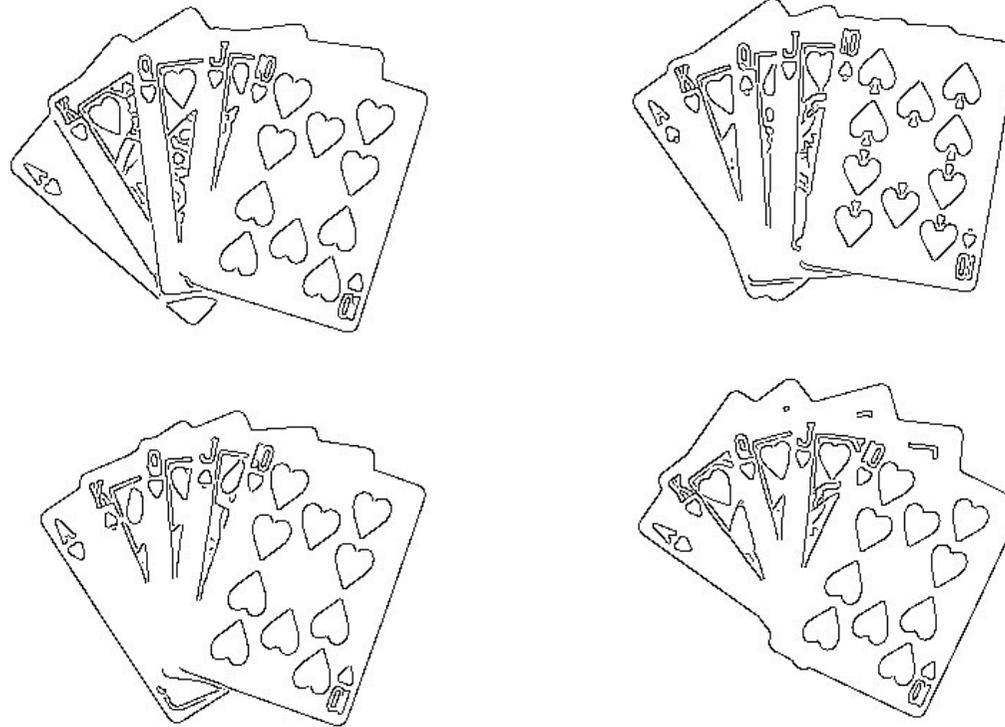
Laplace

Limites de la corrélation

- Coûteux en calculs (mais parallélisable).
- Modèle d'objet très restreint.
- Modèle des perturbations peu réaliste.

Détection des contours

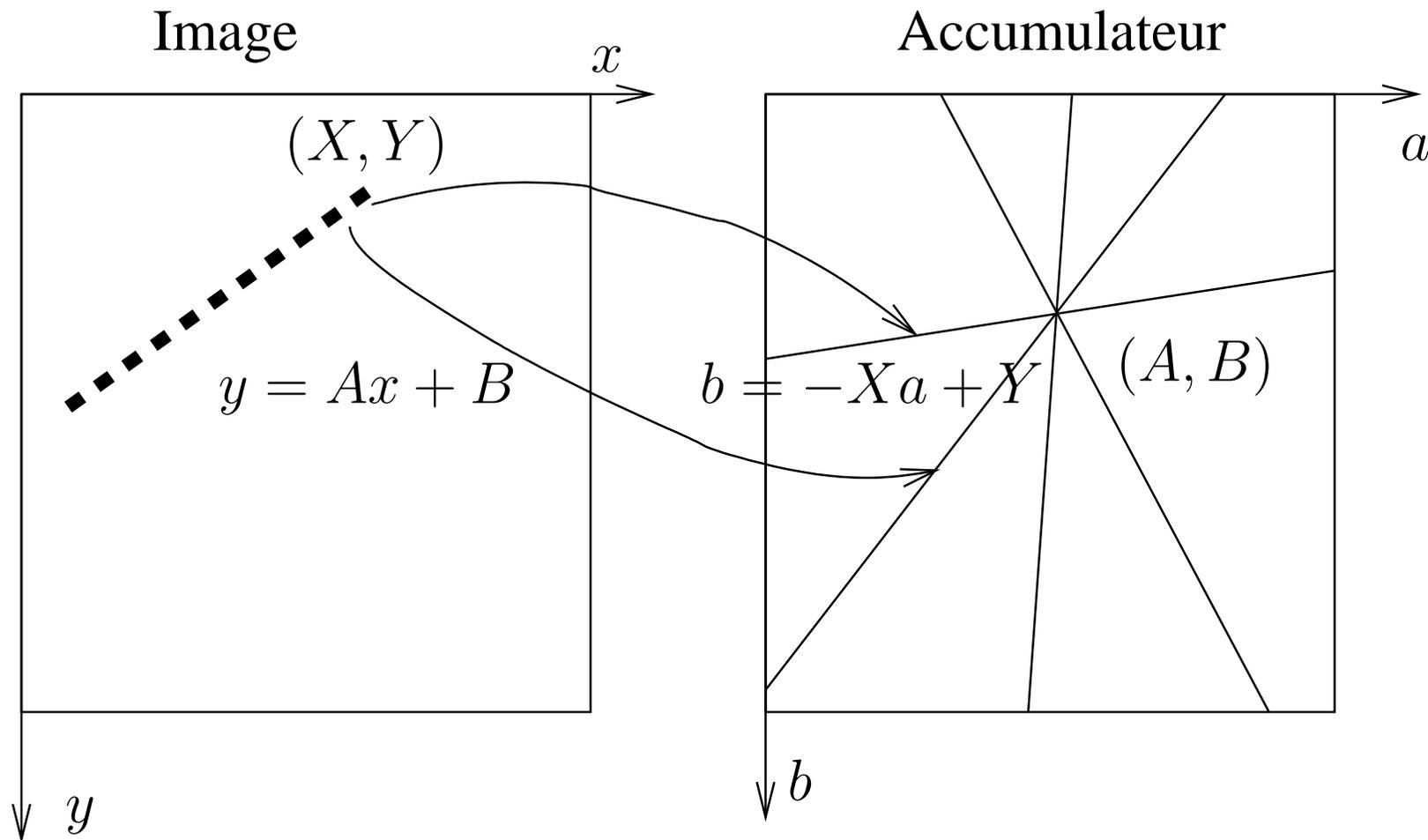
L'extraction des contours permet de réduire la variabilité due aux changements d'éclairage. Les images des contours :



La forme à détecter :



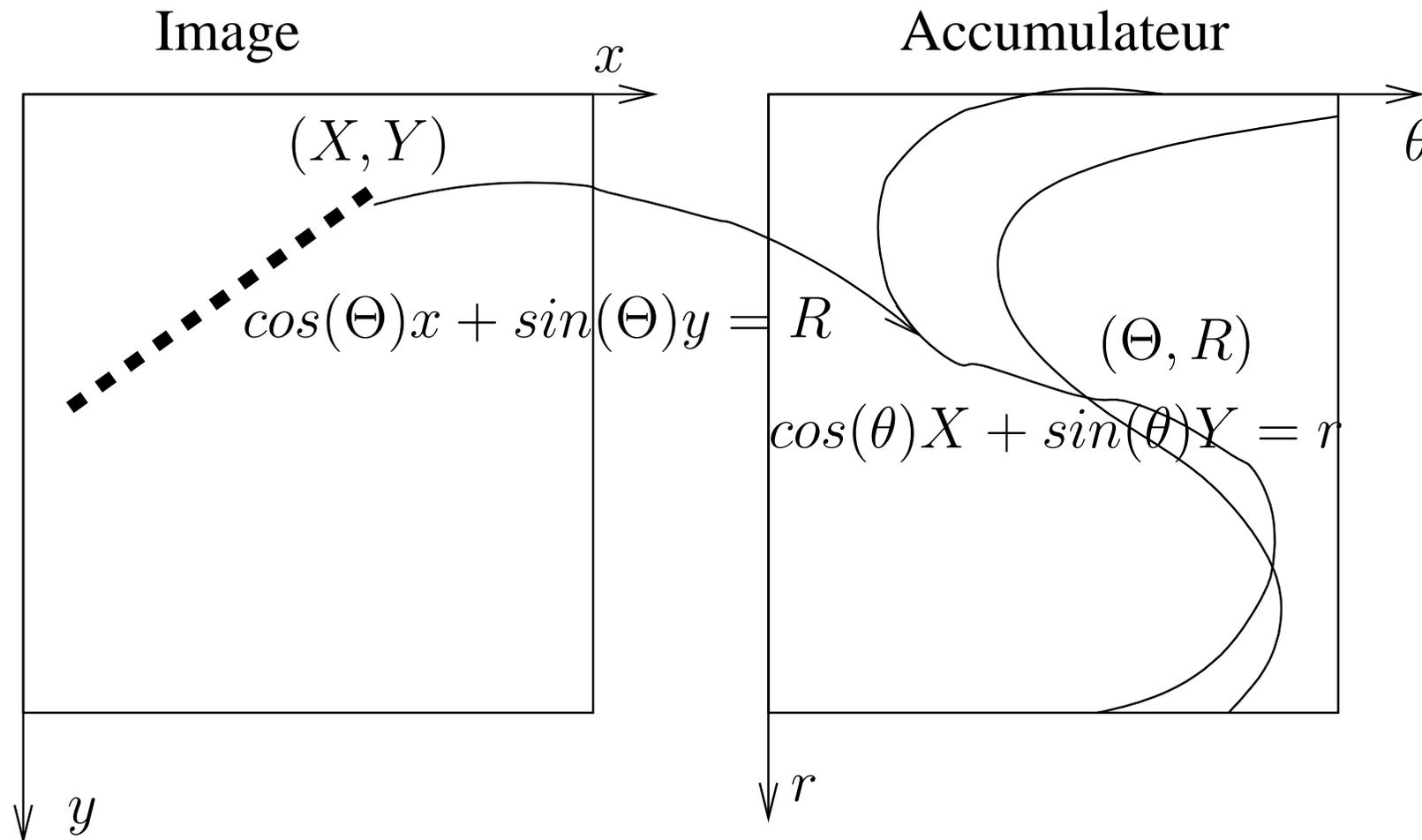
Transformée de Hough



Forme à détecter: ligne (utilisation de la dualité ligne-point).

Paramétrisation cartésienne \Rightarrow problèmes avec les verticales.

Transformée de Hough en polaire



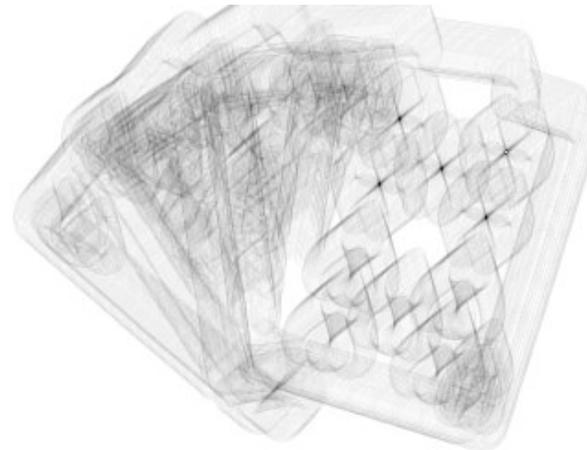
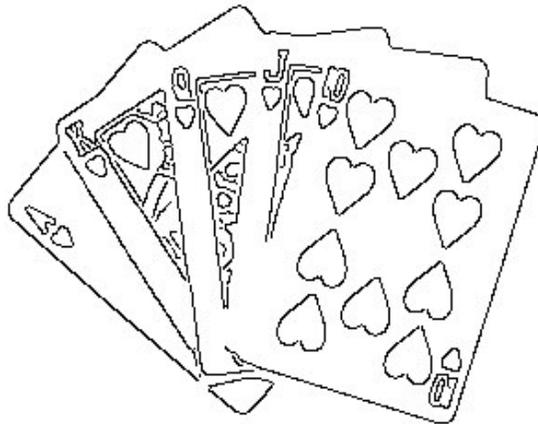
Forme à détecter: ligne.

Paramétrisation polaire plus lourde mais pas de problème avec les verticales.

Méthode Hough généralisée (1)

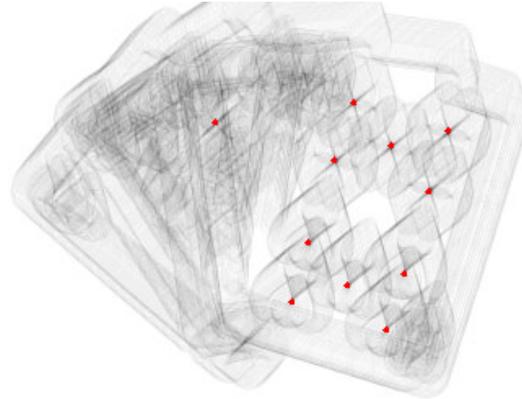
Accumulation par transformée de Hough généralisée :

L'accumulateur de Hough est une image des positions possibles de l'objet à détecter. Pour chaque pixel de contours, on ajoute un vote dans l'accumulateur pour toutes les positions de l'objet à détecter qui pourraient expliquer la présence de ce pixel de contours.



Méthode Hough généralisée (2)

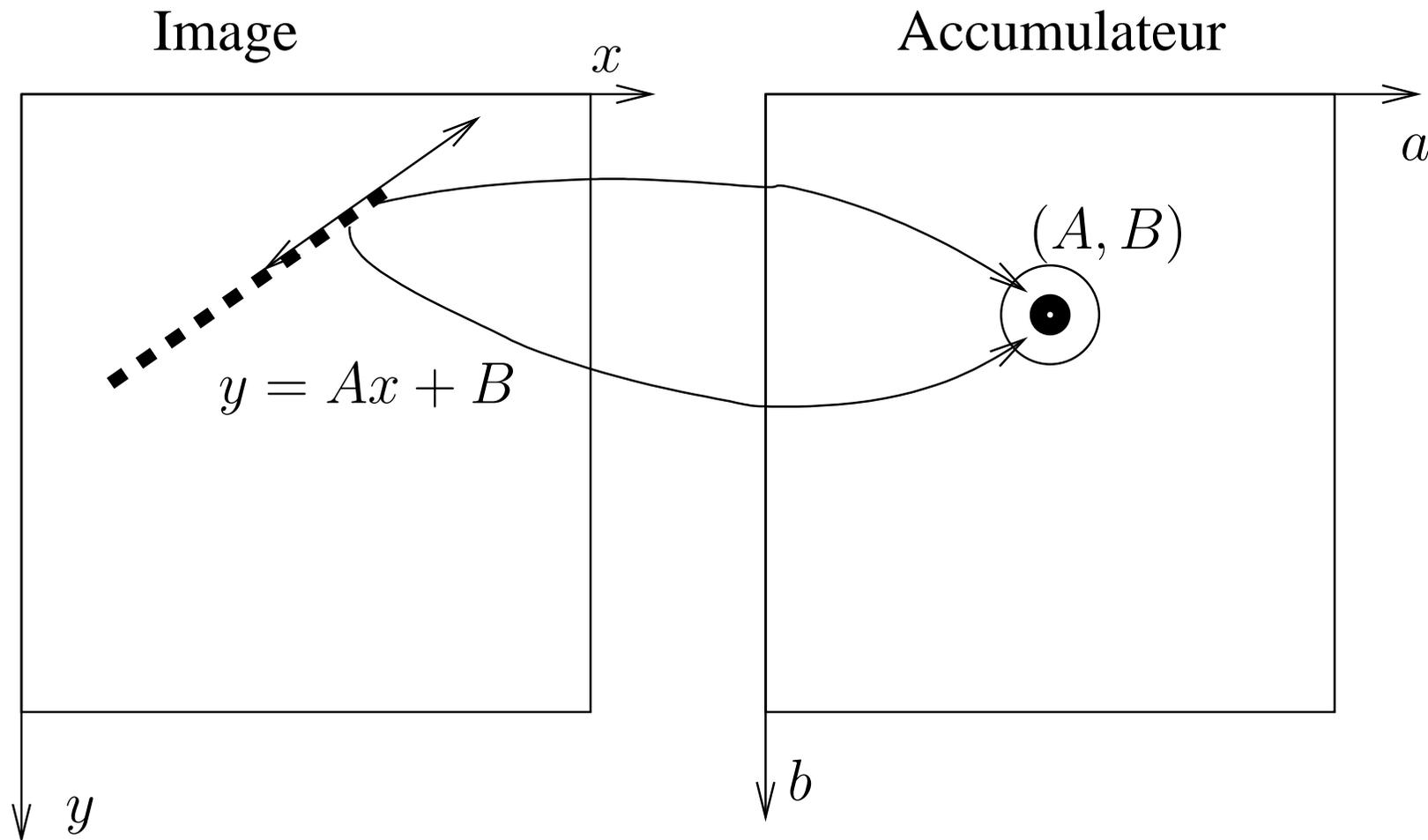
Extraction des minima locaux :



et décision par seuillage sur le nombre de pixels de contours :



Introduction de la tangente

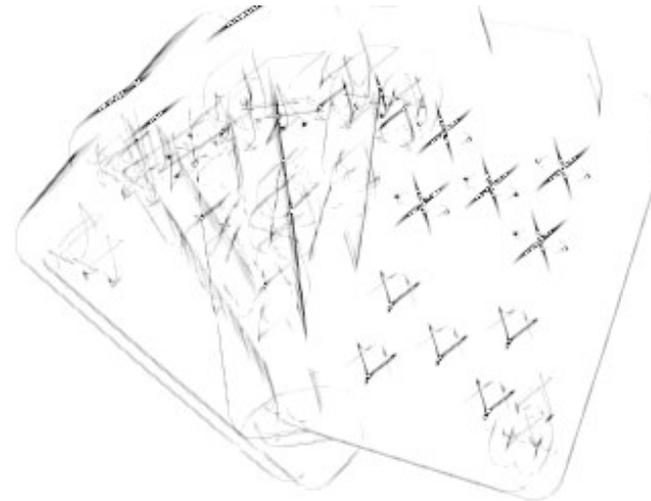
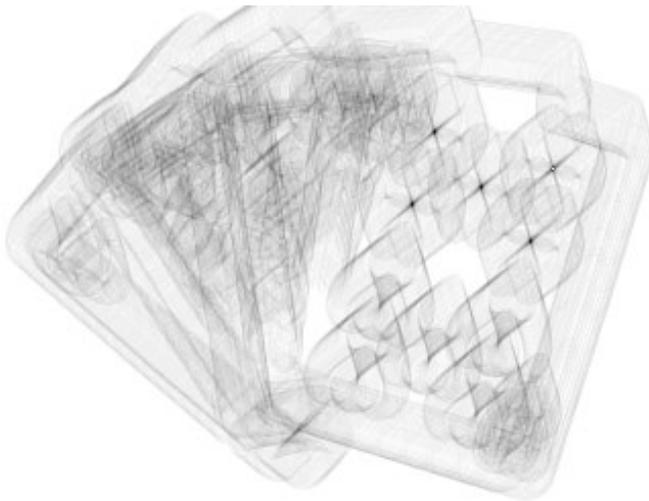
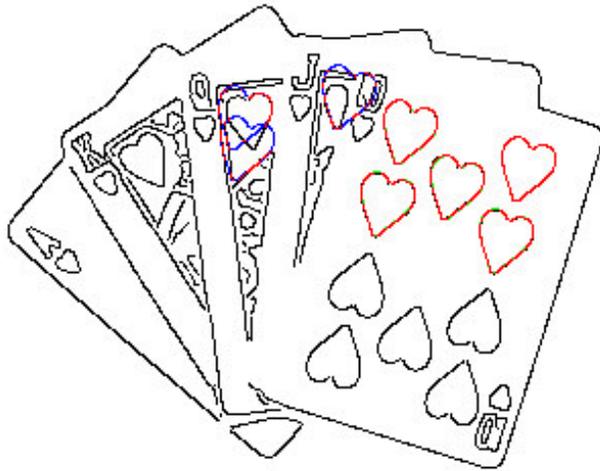


Forme à détecter: ligne.

Paramétrisation cartésienne.

Nécessite le calcul du gradient mais est plus rapide.

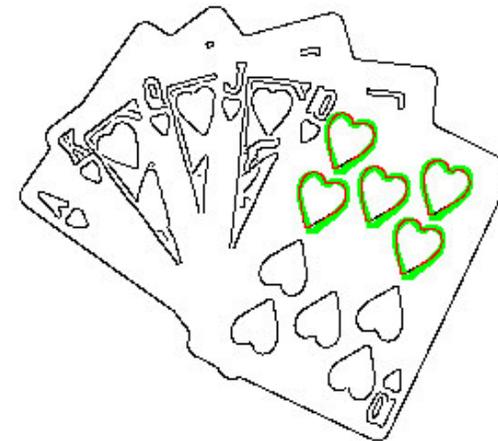
Avec ou sans tangente



⇒ Version avec la tangente est plus discriminante.

Hough généralisée

Résultats :



Temps de détection par image : environ 0.1s.

Perturbations dans les images (1)

Perturbations photométriques :

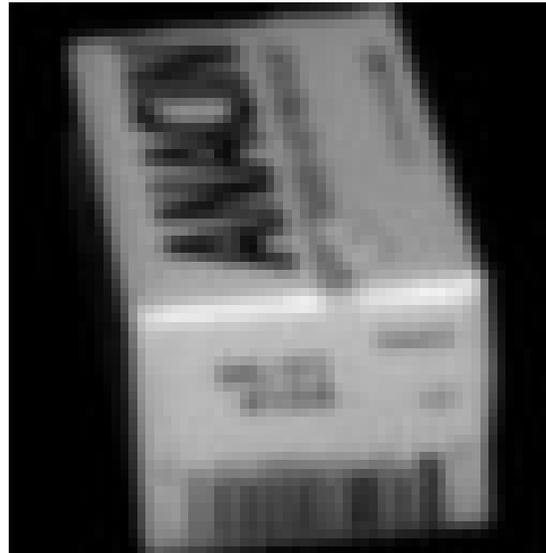
- changement de l'intensité et de la couleur de l'éclairage,
- ombres d'autres objets,
- conditions atmosphériques (ex : brouillard).



Perturbations dans les images (2)

Perturbations géométriques :

- en position (translation) et angle (rotation),
- en taille (échelle, homothétie),
- changement de point de vue (approché par une transformation affine ou homographique),
- occultations, parties manquantes.



Partie 2

1. Détecter un objet modélisé par une image?
2. Détecter un objet avec une modélisation Gaussienne de la variabilité de son aspect?
 - Modélisation de la variabilité par une Gaussienne multi-dimensionnelle.
 - Détection SVD.

Modélisation de la variabilité

Comment tenir compte de cette variabilité?
⇒ l'observer et la modéliser.

Deux types :

- Les perturbations impliquent une faible variabilité des observations ⇒ modéliser ou apprendre ces perturbations.
ex : variation de l'angle de vue entre -5 et 5 degrés.
- Les invariances à certaines variables ⇒ modéliser cette invariance.
ex : invariance à la rotation de l'image.

Approches possibles

Comment modéliser la variabilité des aspects de l'objet à détecter?

- Concevoir un modèle paramétrique et estimer ses paramètres de façon robuste.
- Construire un modèle par apprentissage à partir d'exemples.

Exemple réaliste

Les images à traiter :



Objet à détecter : une voiture **sombre** vue de coté.

Vocabulaire en apprentissage (1)

Définitions :

- Apprentissage **supervisé** : un médiateur indique pour chaque donnée son label (objet/non objet).
- Apprentissage **non supervisé** : l'algorithme doit faire lui même la part des choses.

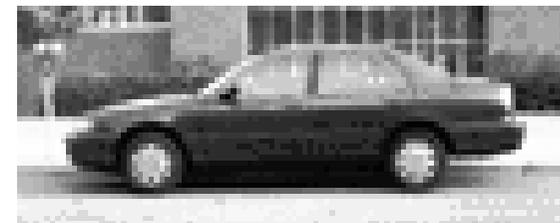
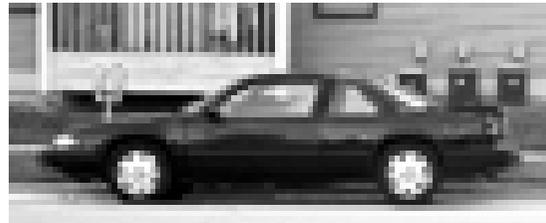
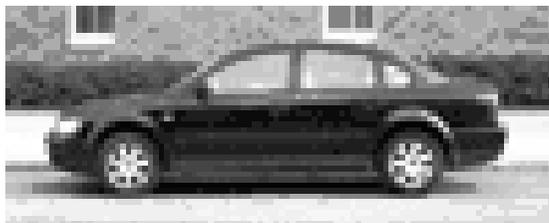
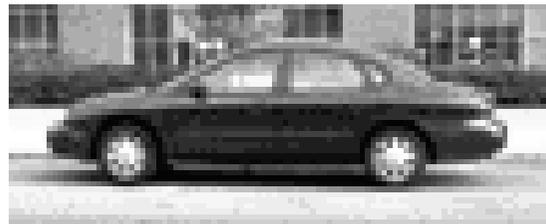
Vocabulaire en apprentissage (2)

Définitions :

- Base d'**apprentissage** (Training database) : images qui permettent de construire le modèle de l'objet à détecter et qui représente bien ses différents aspects.
- Base de **test** (Test database) : images différentes de la base d'apprentissage qui permettent de valider la qualité de l'apprentissage.
- Base de **validation** (Validation database) : images différentes des bases précédentes qui permettent éventuellement d'optimiser le choix des paramètres de l'algorithme de détection.

Base d'apprentissage

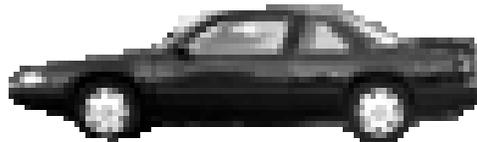
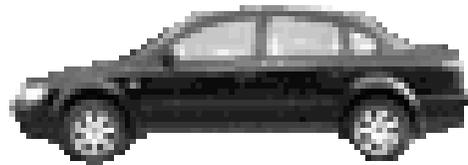
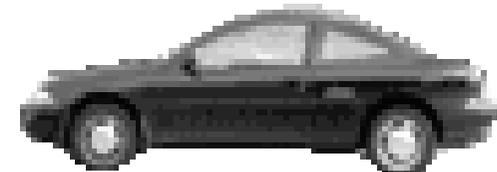
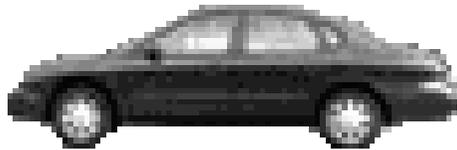
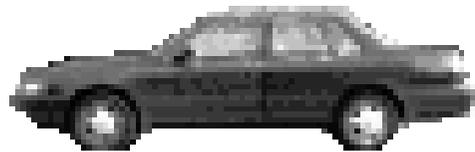
Exemples de voiture sombre vue de coté :



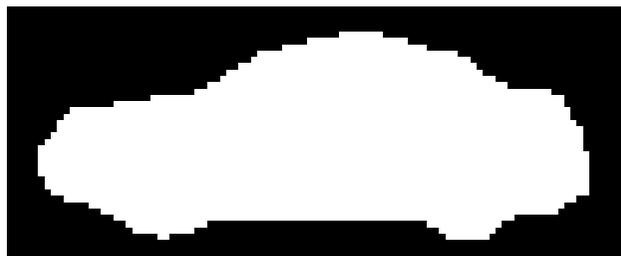
Base de $p = 212$ images d'exemples **positifs**.

Masquage

Exemples de voiture sombre vue de coté, sans fond :



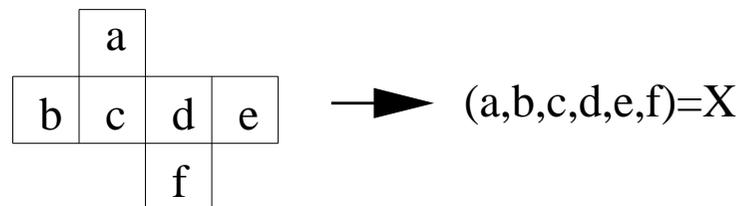
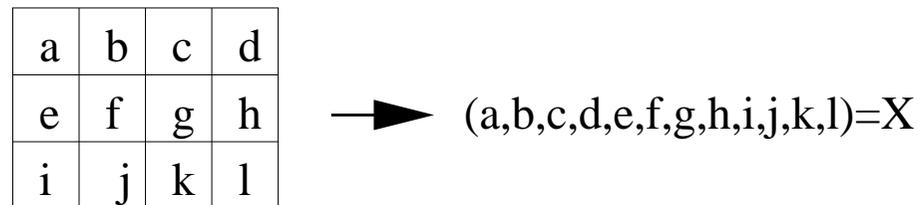
Masque du fond commun:



Choix de la signature?

Nombreux choix possibles pour la représentation de chaque image de la base d'apprentissage, en fonction de l'application.

Un choix simple:



Base d'apprentissage = ensemble de vecteurs X_i .

Modélisation Gaussienne (1)

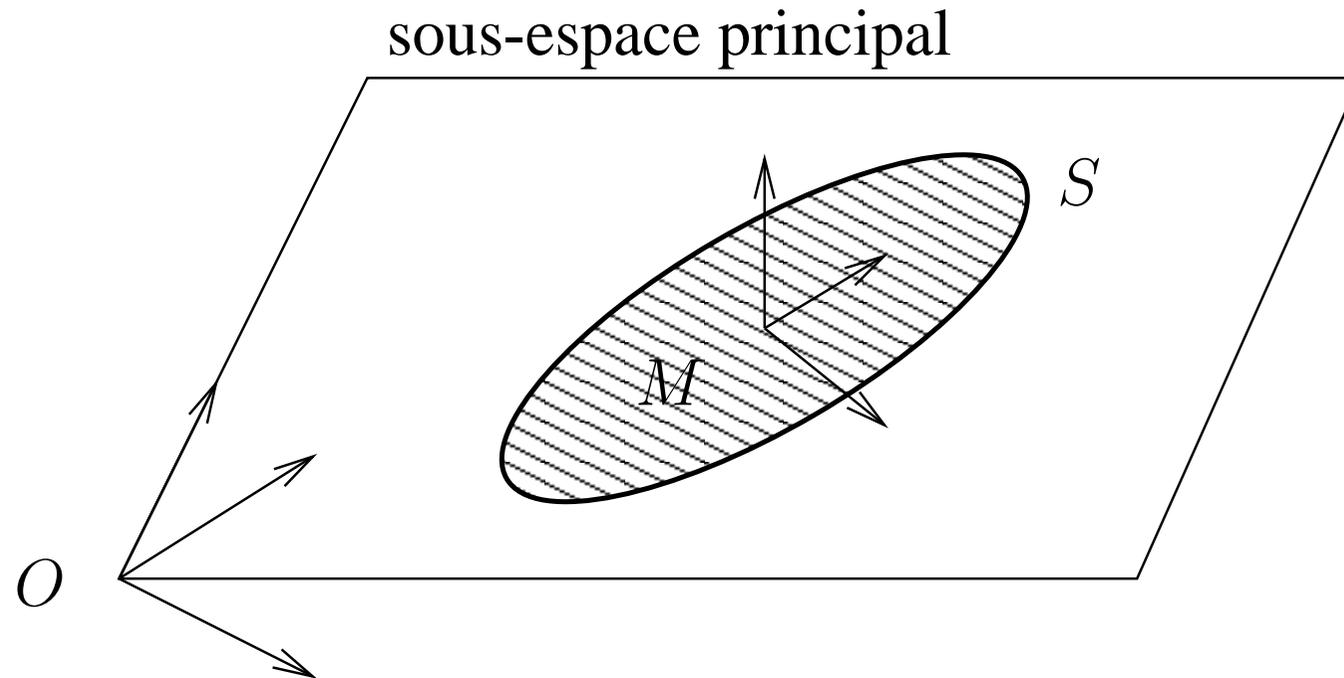
Chaque image est un vecteur X_i dans un espace de dimension d grande. Si la variabilité des images d'un même objet est Gaussienne dans un sous-espace, la probabilité pour X d'être une image de l'objet est :

$$prob(X) = \frac{1}{\sqrt{(2\pi)^n det(S)}} e^{-\frac{1}{2}(X-M)^t S^{-1}(X-M)}$$

où M est sa moyenne et S sa matrice de covariance.

La variabilité de l'objet est généralement de dimension n beaucoup plus petite que d .

Modélisation Gaussienne (2)



Comment estimer le sous-espace principal, la moyenne M et la sous matrice de covariance S ?

\Rightarrow Analyse en Composantes Principales (ACP ou PCA) des p vecteurs de la base d'apprentissage.

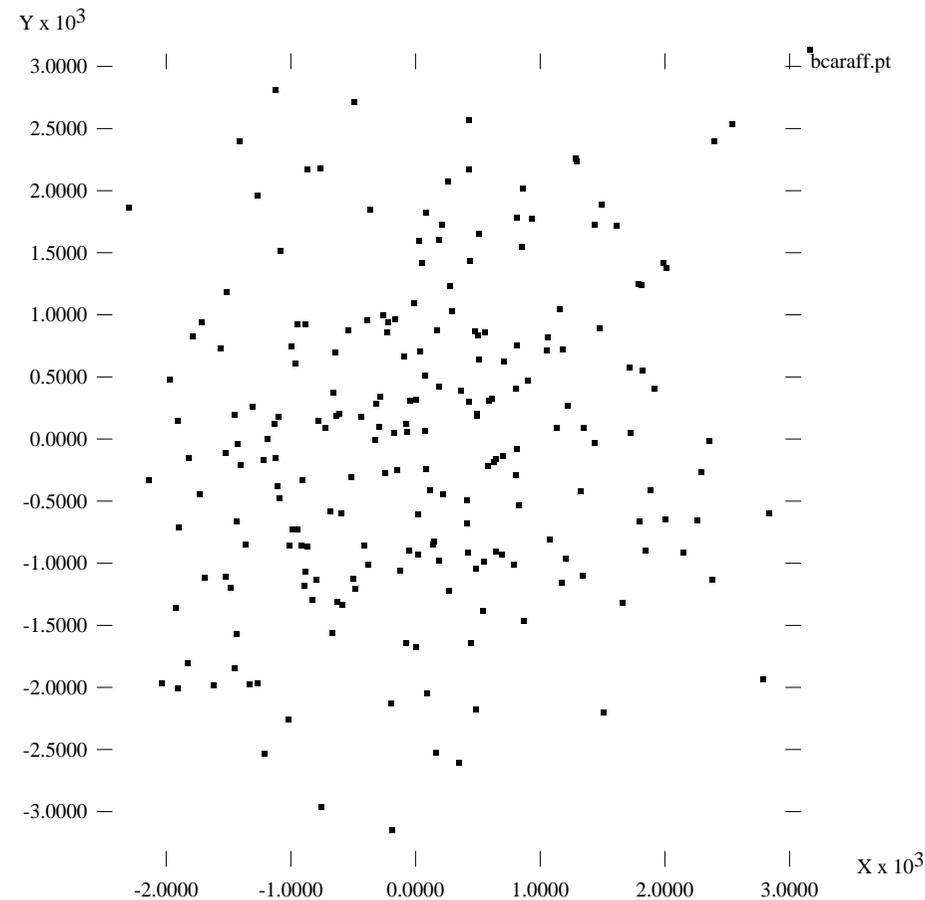
ACP (1)

Analyse en composantes principales des p vecteurs X_i :

- estimer M comme la moyenne des X_i , centrer les données $Y_i = X_i - M$, et construire la matrice des données (Y_i) ,
- diagonaliser la matrice d'inertie $I = (Y_i)(Y_i)^t$ en $I = RLR^t$, avec $L = (\lambda_i)_{1 \leq i \leq d}$ diagonale et R une rotation,
- conserver uniquement les n plus grandes valeurs propres $(\lambda_i)_{1 \leq i \leq n} = \Lambda$ et les n vecteurs propres associées ρ . ρ est une matrice $d \times n$.
- Le sous espace principal est le perpendiculaire à ρ et $S = \rho\Lambda\rho^t$.

\Rightarrow permet de réduire la dimension de l'espace des attributs.

ACP (2)



Projection sur l'espace réduit aux deux premières composantes principales de la base d'apprentissage.

ACP (3)

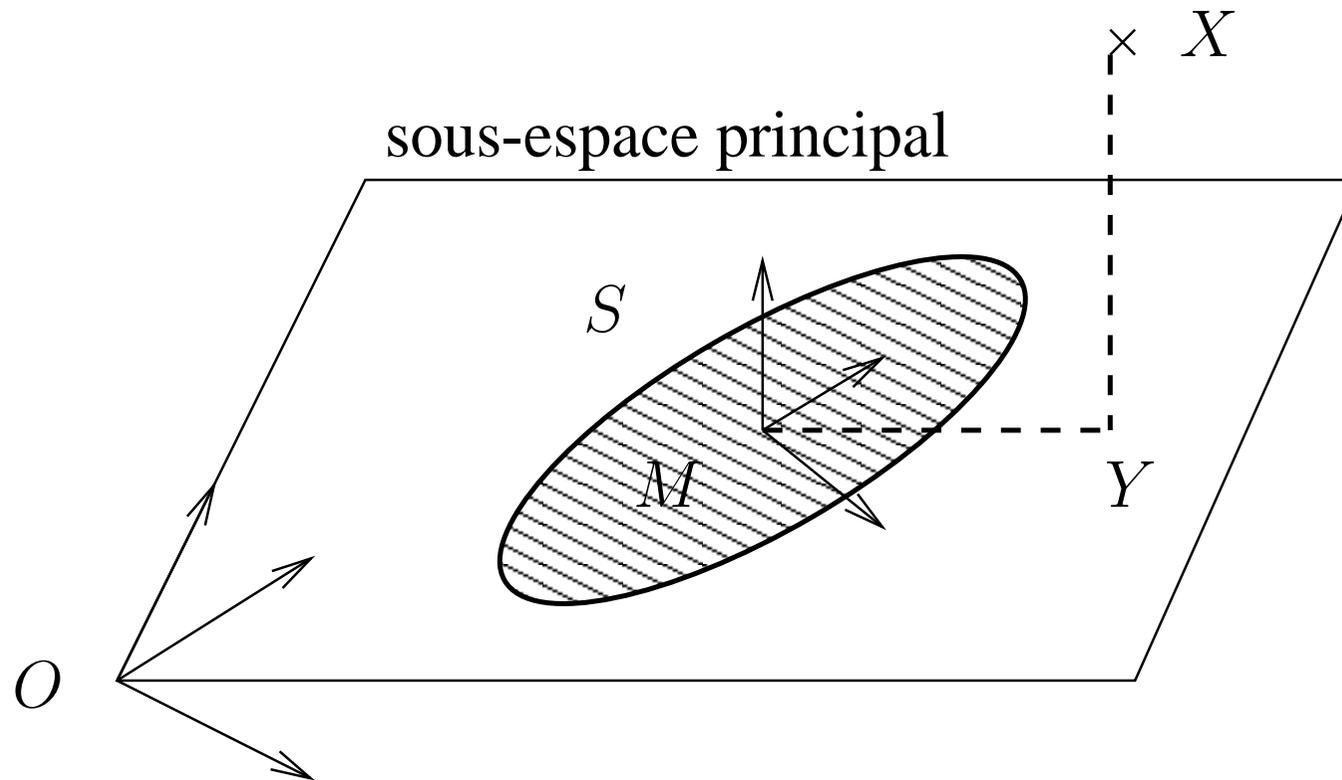
En pratique, décomposition en valeurs singulières :

- estimer M comme la moyenne des X_i , centrer les données $Y_i = X_i - M$, et construire (Y_i) .
- SVD de la matrice des données $(Y_i) = RVU^t$, avec R rotation de taille $d \times d$, V matrice diagonale des valeurs singulières de taille $d \times p$, et U rotation de taille $p \times p$.
- conserver uniquement les n plus grandes valeurs singulières de V $\lambda_i = v_i^2$ et les n vecteurs associées ρ .
- La transformation $Z = \rho^t Y$ projette le vecteur Y dans une base orthogonale du sous-espace principal.

Méthode SVD de détection (1)

La distance à l'objet moyen est

$d(X, M) = (\rho^t(X - M))^t \Lambda^{-1} \rho^t(X - M)$, lorsque X est dans le sous-espace principal.



Comment estimer la distance sur tout l'espace entre un vecteur X et l'objet moyen M ?

Méthode SVD de détection (2)

Après ACP :

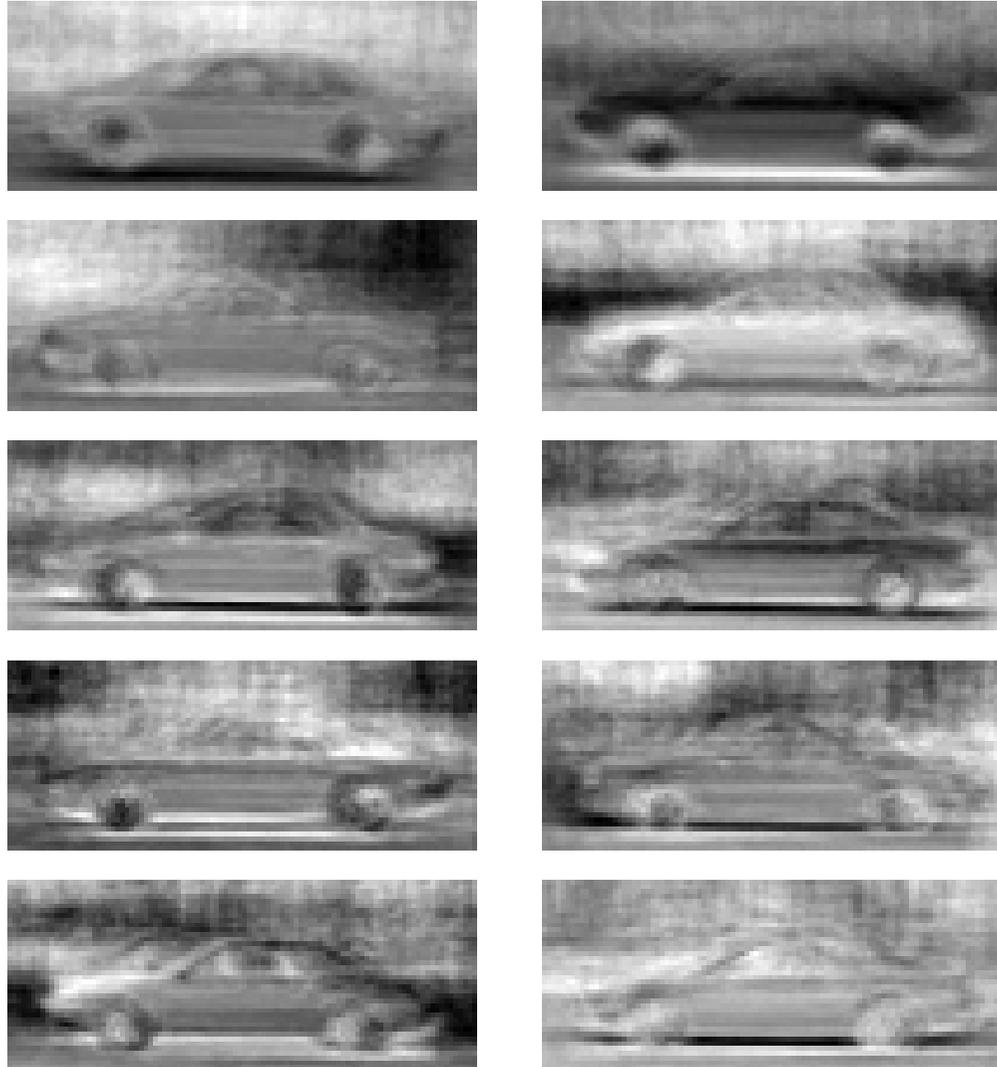
- La transformation $Z = \rho^t(X - M)$ projette le vecteur X dans le sous-espace principal. La distance à l'objet moyen est $d(X, M) = Z^t \Lambda^{-1} Z$, lorsque X est dans le sous-espace principal.
- Pour tenir compte de la distance de X au sous-espace principal il faut ajouter à $d(X, M)$ le terme $\frac{1}{r} \|X - Z\|^2$, avec

$$r = \frac{1}{d-n} \sum_{i=n+1}^d \lambda_i = \frac{1}{d-n} \sum_{i=n+1}^d v_i^2.$$

$$\Rightarrow d(X, M) = \|V^{-1} Z\| + \frac{d-n}{\|V\|^2} \|X - Z\|^2$$

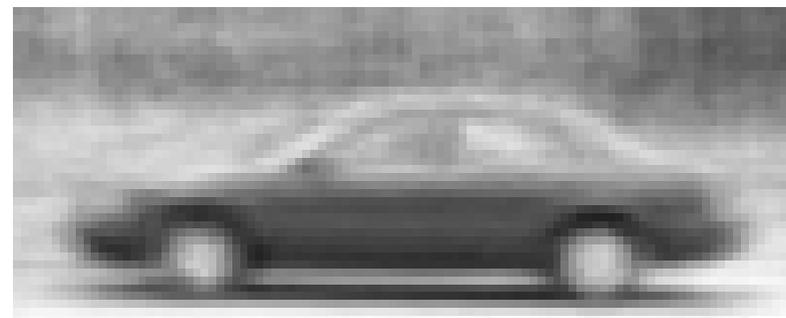
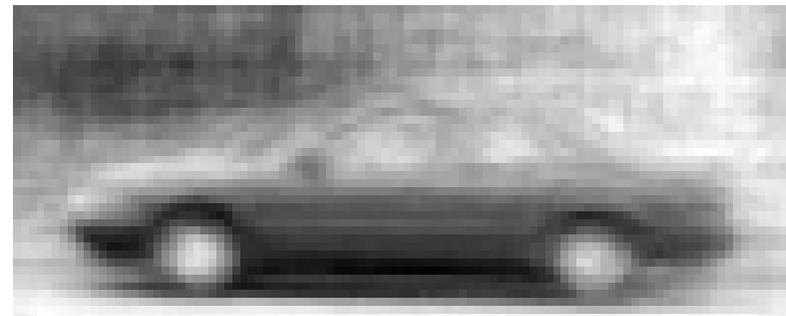
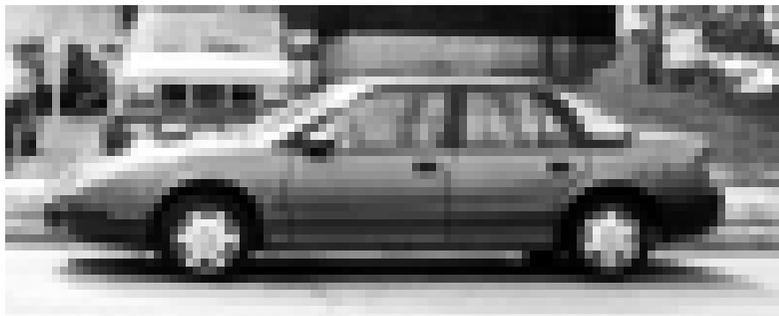
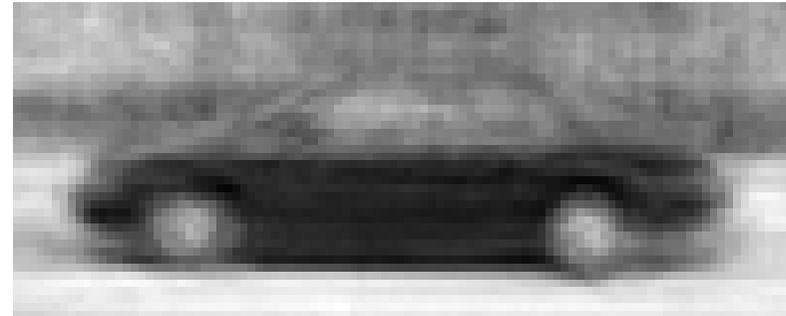
Vecteurs propres

Les $n = 10$ composantes principales :



Reconstruction

Permet un codage grossier de l'objet avec $n = 10$ coefficients :

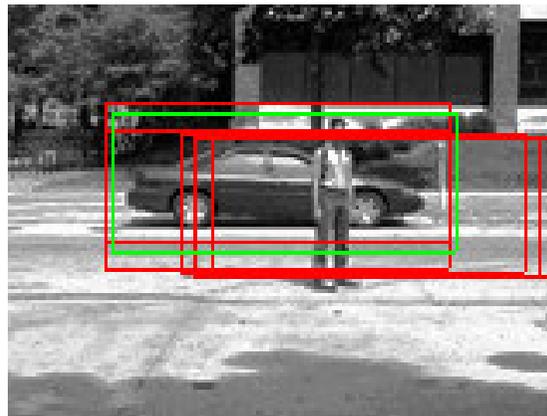
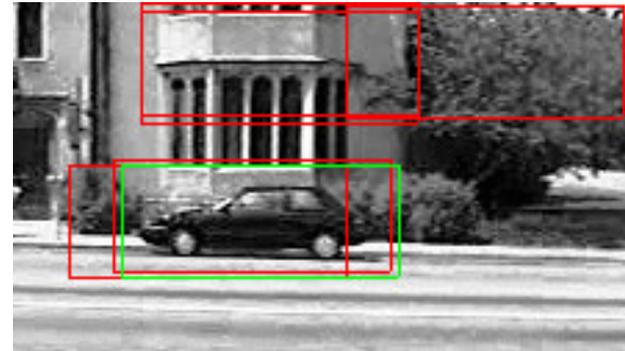


Original

Reconstruction

Résultats détection SVD

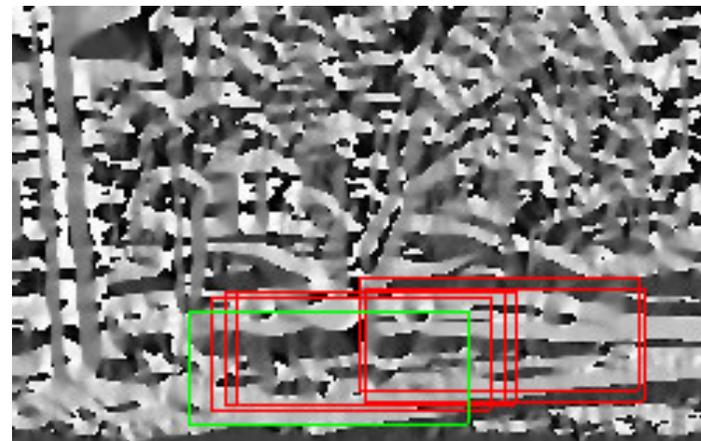
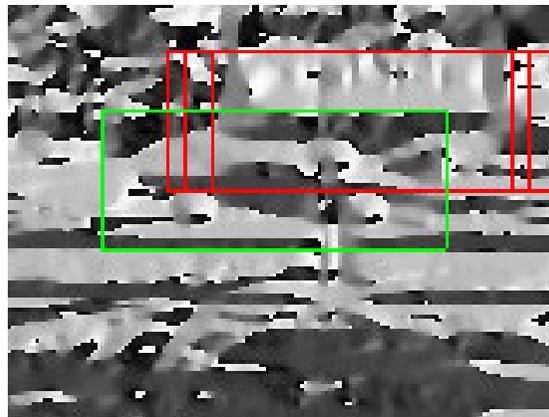
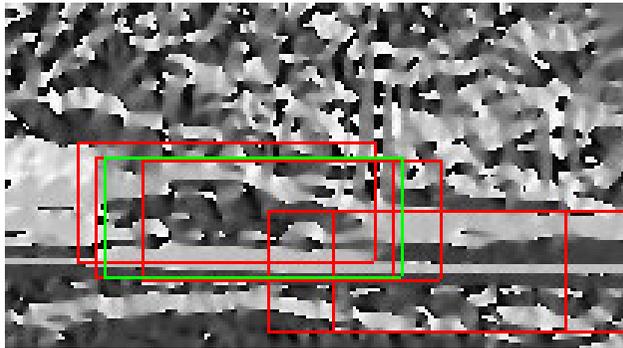
Base ACP avec les $n = 10$ composantes précédentes.



Temps moyen = $n \times$ méthode par corrélation.

Résultats de détection

Base ACP avec $n = 10$ composantes sur les images d'angles.



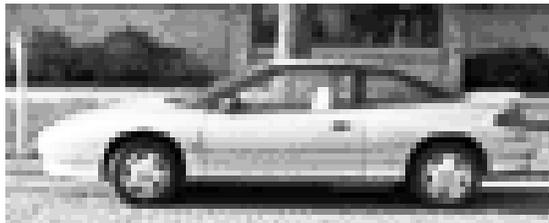
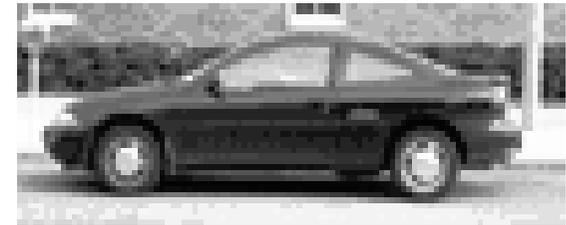
Temps moyen = $n \times$ méthode par corrélation.

Partie 3

1. Détecter un objet modélisé par une image?
2. Détecter un objet avec une modélisation Gaussienne de la variabilité de son aspect?
3. Détecter un objet avec une modélisation SVM de la variabilité de son aspect?
 - Modélisation de variabilités complexes.
 - Détection SVM (Support Vector Machine).

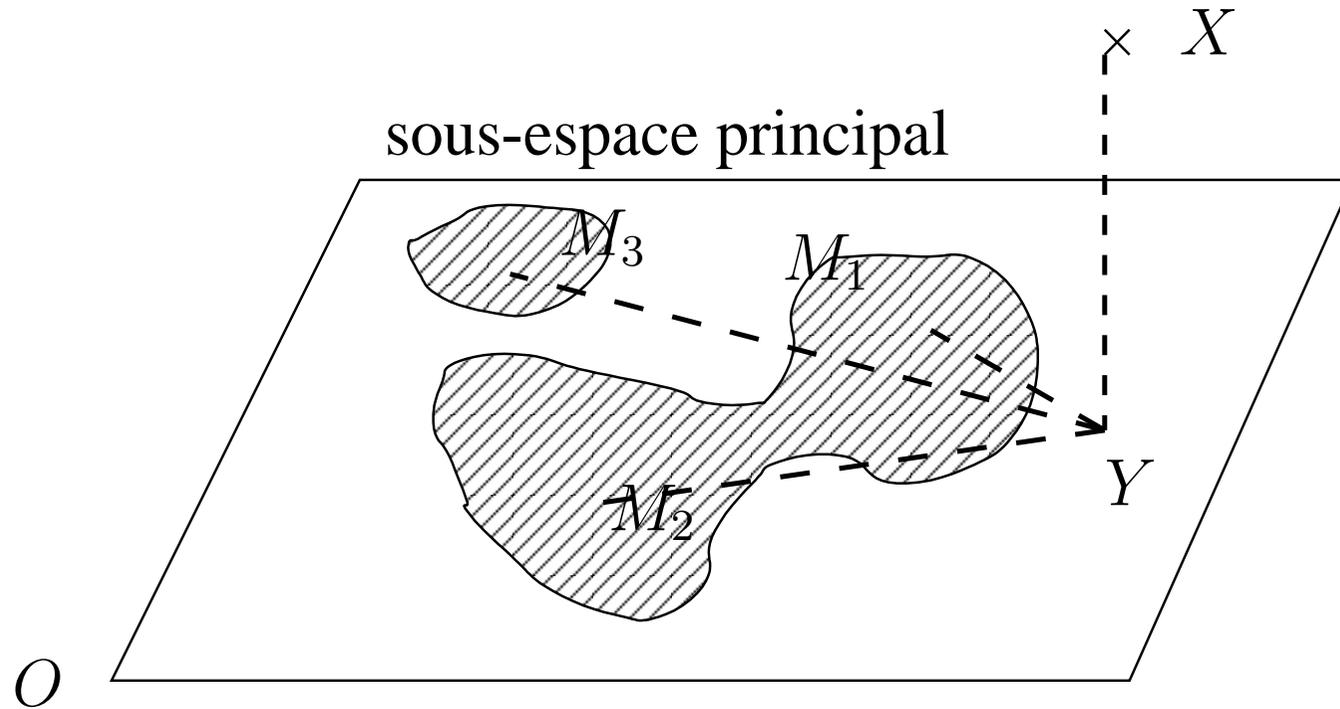
Base d'apprentissage plus riche

Exemples de voiture vue de coté :



⇒ Présence de plusieurs classes.

Modélisation par un mélange (1)



Modélisation par un mélange (2)

La distribution des aspects est modélisée par un mélange de m Gaussienne de moyennes M_j et de matrice de covariance C_j :

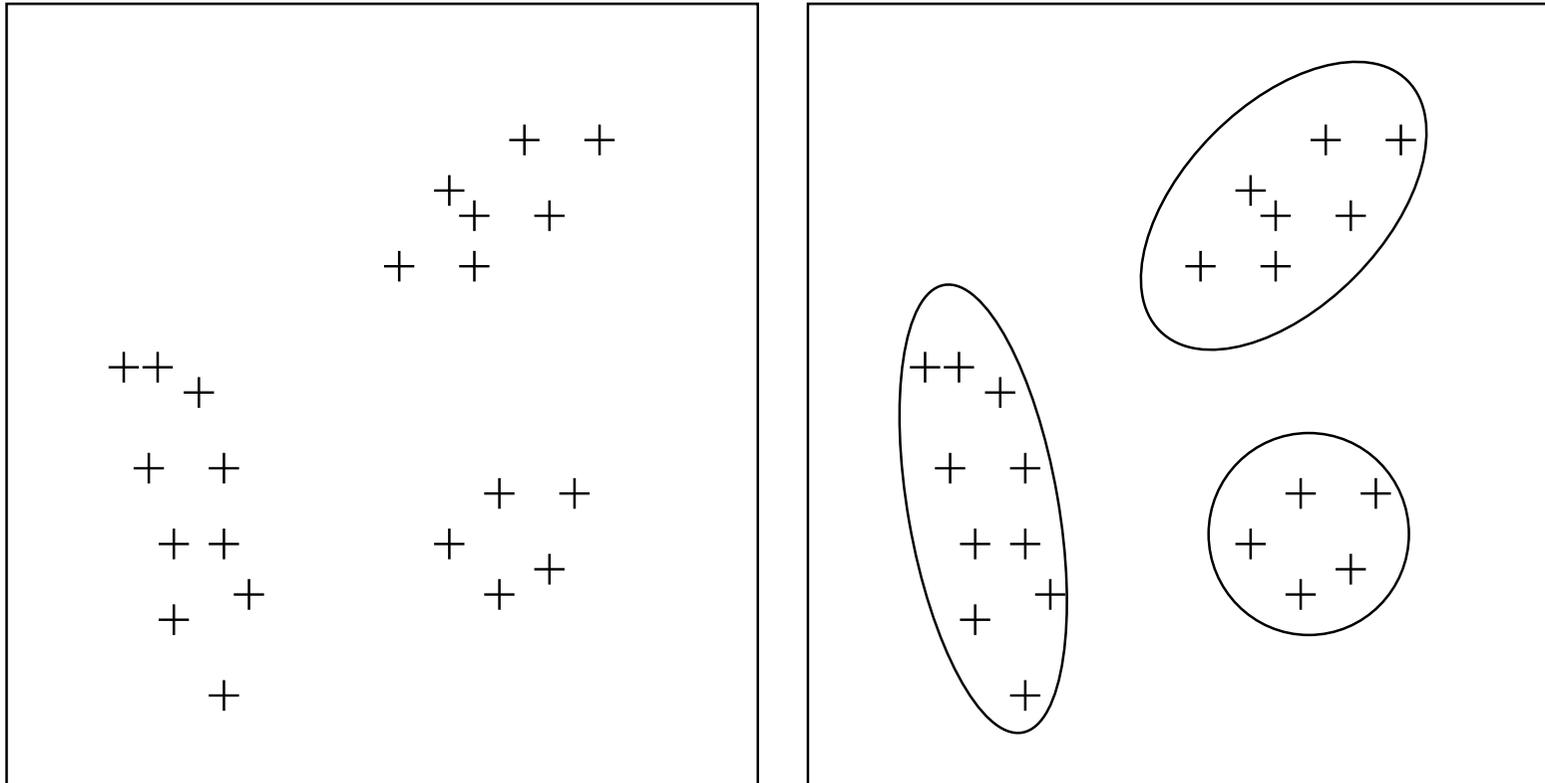
$$prob(Y) = \sum_{j=1}^{j=m} w_j \frac{1}{\sqrt{\det C_j}} g((Y - M_j)^t C_j^{-1} (Y - M_j))$$

avec $g(x) = e^{-\frac{1}{2}x^2}$. Le plus souvent $C_j = \sigma^2 Id$, avec σ fixé ou à estimer. Mais comment estimer les M_j et éventuellement C_j ?

- Si seulement des exemples positifs \Rightarrow méthodes de classification.
- Si exemples positifs et négatifs \Rightarrow SVM.

Classification

Mélange de 3 classes avec adaptation de la matrice de covariance :



Une méthode de classification (1)

Classer les vecteurs X_i de la base d'apprentissage en m classes Gaussiennes de poids équivalents ($w_j = \frac{1}{m}$, $C_j = Id$)?

Initialiser aléatoirement les centres des classes M_j .

- estimer l'appartenance de chaque donnée à chaque

classe :
$$a_{ij} = \frac{g(\|X_i - M_j\|)}{\sum_{k=1}^m g(\|X_i - M_k\|)}$$

- estimer le barycentre de chaque classe :

$$M_j = \frac{\sum_{i=1}^p a_{ij} X_i}{\sum_{i=1}^p a_{ij}}$$

- itérer jusqu'à stabilisation.

Si des poids différents?

Une méthode de classification (2)

Classer les vecteurs X_i de la base d'apprentissage en m classes Gaussiennes ($C_j = Id$)?

Initialiser les centres des classes M_j et les poids w_j .

- estimer l'appartenance de chaque donnée à chaque

classe :
$$a_{ij} = \frac{w_j g(\|X_i - M_j\|)}{\sum_{k=1}^m w_k g(\|X_i - M_k\|)}$$

- estimer le centre de chaque classe :
$$M_j = \frac{\sum_{i=1}^p a_{ij} X_i}{\sum_{i=1}^p a_{ij}}$$

- estimer le poids de chaque classe :
$$w_j = \frac{\sum_{i=1}^p a_{ij}}{\sum_{k=1}^m \sum_{i=1}^p a_{ik}}$$

- itérer jusqu'à stabilisation.

Détection SVD multiple

A partir des résultats de la classification, il est possible de généraliser la détection SVD au cas de la modélisation par un mélange de Gaussiennes :

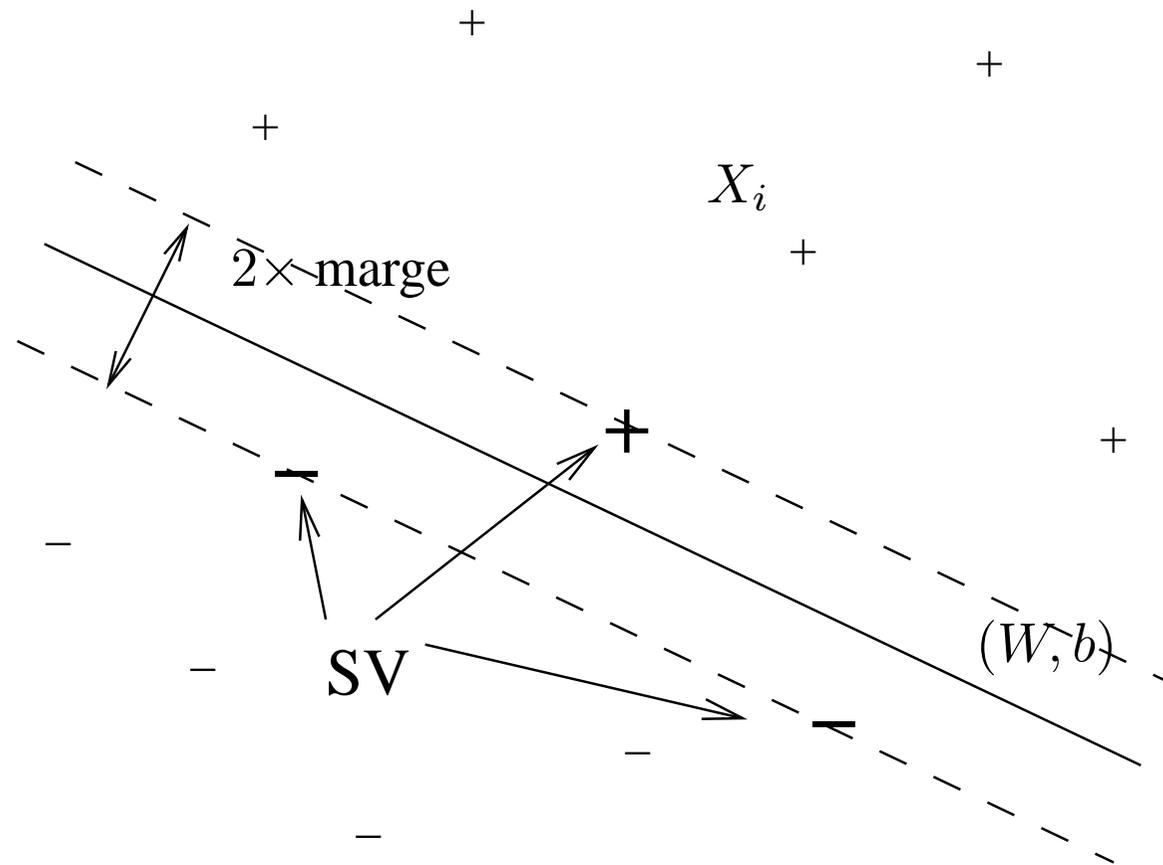
- classification de la base d'apprentissage pour obtenir la modélisation par un mélange de Gaussiennes,
- ACP sur chaque mode en utilisant la décomposition SVD,
- Pour tester une image, calculer chaque probabilité d'appartenir à un mode, et sommer sur les m modes pour obtenir la probabilité totale d'être l'objet recherché.

⇒ Temps moyen = $m \times$ méthode SVD.

Séparation linéaire

Comment séparer linéairement les exemples positifs ($X_i, y_i = +1$) de ceux négatifs ($X_i, y_i = -1$)?

Chercher un hyperplan (W, b) tel que $y_i(W^t X_i + b) \geq 1$.



Séparation linéaire marge max (1)

Plusieurs solutions \Rightarrow séparation à marge maximale (Maximal Margin Classifier). Sur les vecteurs de support $W^T X + b = \pm 1 \Rightarrow$ la marge d'un hyperplan (W, b) est $\frac{1}{\|w\|}$.

Chercher l'hyperplan (W, b) qui minimise :

$$\frac{1}{2} W^T W$$

avec les contraintes $(1 \leq i \leq p)$:

$$y_i (W^t X_i + b) \geq 1$$

Comment résoudre cela efficacement?

Séparation linéaire marge max (2)

Minimisation sous contraintes \Rightarrow Lagrangien (Théorème de Khun et Tucker) :

$$J = \frac{1}{2}W^T W - \sum_{i=1}^p \alpha_i (y_i (W^t X_i + b) - 1)$$

avec $\alpha_i \geq 0$ multiplicateurs de Lagrange.

- Minimisation en W , $\frac{\partial J}{\partial W} = 0 \Rightarrow W = \sum_{i=1}^p \alpha_i y_i X_i$,
- Minimisation en b , $\frac{\partial J}{\partial b} = 0 \Rightarrow \sum_{i=1}^p \alpha_i y_i = 0$,
- Maximisation en α_i .

Séparation linéaire marge max (3)

La formulation duale du problème est donc la maximisation quadratique :

$$L(\alpha_i) = \sum_{i=1}^p \alpha_i - \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p y_i \alpha_i y_j \alpha_j X_i^t X_j$$

avec les contraintes $\alpha_i \geq 0$ et $\sum_{i=1}^p \alpha_i y_i = 0$.
Fonction de classification solution :

$$f(X) = \sum_{i=1}^p y_i \alpha_i^{opt} X_i^t X + b^{opt}$$

avec $b^{opt} = -\frac{1}{2} (\max_{y_i=-1} \sum_{j=1}^p \alpha_j^{opt} y_j X_i^t X_j + \min_{y_i=1} \sum_{j=1}^p \alpha_j^{opt} y_j X_i^t X_j)$.

Méthode SVM

Comment étendre au cas non-linéaire?

⇒ Remplacer le produit scalaire $X^t X'$ par une mesure de dissimilarité : un noyau $K(X, X')$. La formulation duale devient :

$$L(\alpha_i) = \sum_{i=1}^p \alpha_i - \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p y_i \alpha_i y_j \alpha_j K(X_i, X_j)$$

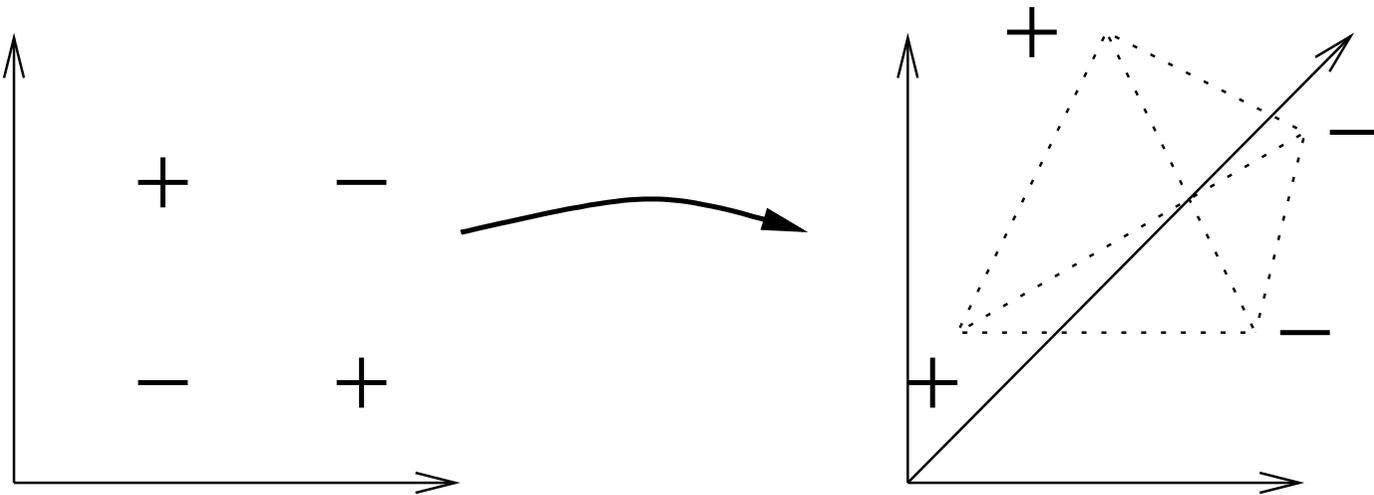
avec $\alpha_i \geq 0$ et $\sum_{i=1}^p \alpha_i y_i = 0$. Fonction de classification

$$f(X) = \sum_{i=1}^p y_i \alpha_i^{opt} K(X_i, X) + b^{opt} \text{ avec}$$

$$b^{opt} = -\frac{1}{2} (\max_{y_i=-1} \sum_{j=1}^p \alpha_j^{opt} y_j K(X_i, X_j) +$$

$$\min_{y_i=1} \sum_{j=1}^p \alpha_j^{opt} y_j K(X_i, X_j)).$$

Interprétation (1)



⇒ Dans le plan, l'espace d'origine, les quatre points ne sont pas linéairement séparables, mais dans l'espace implicite de dimension plus grande ils sont linéairement séparables.

Interprétation (2)

Séparation linéaire dans un espace implicite induit par le noyau K .

Exemple en 2D : $X = (x, y)$, $X' = (x', y')$. Noyau polynomial de degré 2 : $K(X, X') = (1 + X^t X')^2$.
En développant :

$$K(X, X') = \begin{pmatrix} 1 \\ \sqrt{2}x \\ \sqrt{2}y \\ x^2 \\ \sqrt{2}xy \\ y^2 \end{pmatrix}^t \begin{pmatrix} 1 \\ \sqrt{2}x' \\ \sqrt{2}y' \\ x'^2 \\ \sqrt{2}x'y' \\ y'^2 \end{pmatrix}$$

\Rightarrow C'est un produit scalaire en dimension 6.

Choix du Noyau SVM

- Le noyau K est Mercer (ou positif) si par définition :
Pour tout $p > 0$, et pour tout $(X_i), 1 \leq i \leq p$:

$$\sum_{i=1}^p \sum_{j=1}^p c_i c_j K(X_i, X_j) \geq 0$$

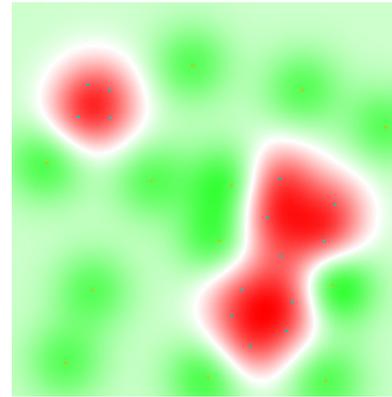
quelque soit $c_i, 1 \leq i \leq p$.

- Noyau Mercer \Rightarrow **unicité** de la solution SVM.

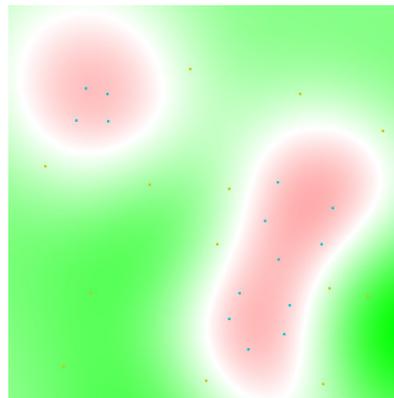
Choix de l'échelle



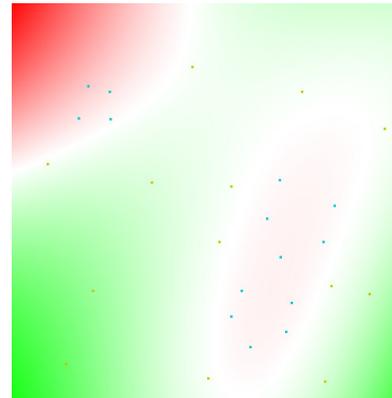
$\sigma = 0.001, 27 \text{ SV}$



$\sigma = 0.01, 26 \text{ SV}$



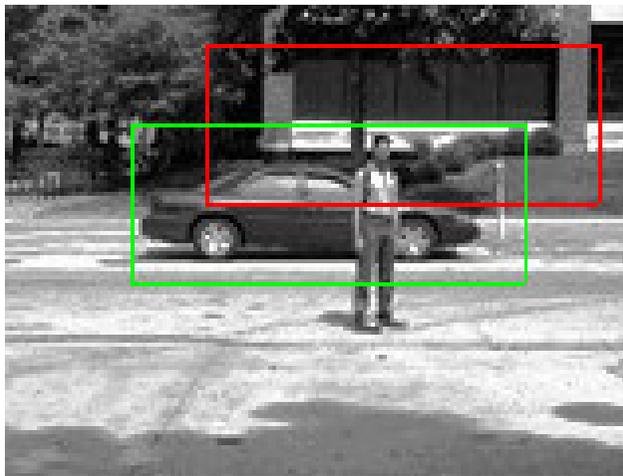
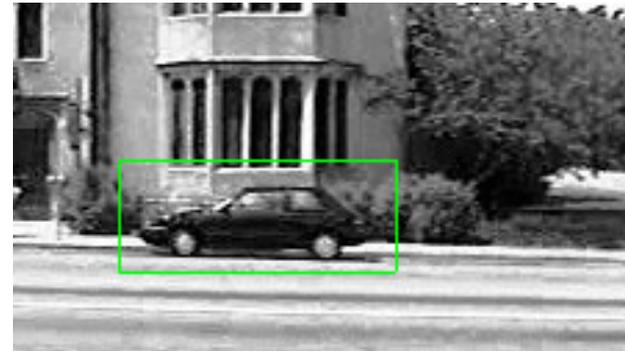
$\sigma = 0.1, 12 \text{ SV}$



$\sigma = 1.0, 10 \text{ SV}$

Noyau Gaussien $K(X, X') = \exp\left(-\frac{\|X - X'\|^2}{\sigma^2}\right)$. Les m vecteurs de support (SV) sont les X_i avec un $\alpha_i \neq 0$

Résultats SVM

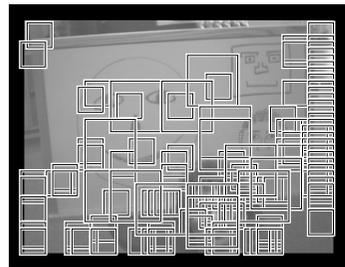


Noyau Laplacien $K(X, X') = \exp\left(-\frac{\|X - X'\|}{\sigma}\right)$. 200
exemples positifs et 200 négatifs.

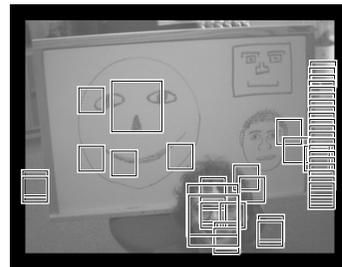
Temps moyen = $m \times$ méthode par corrélation. m est plus
petit que p ($p = 400$, $m = 146$).

Approche graduelle

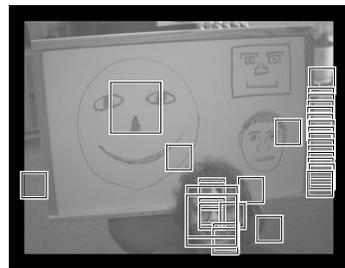
En pratique, il est plus efficace de faire une détection graduelle (coarse-to-fine) avec de plus en plus de SV.



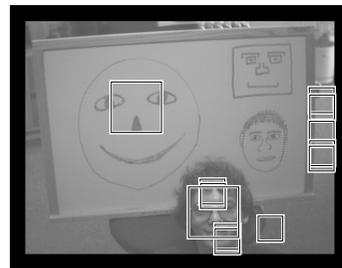
alarms=117/1960
l=1



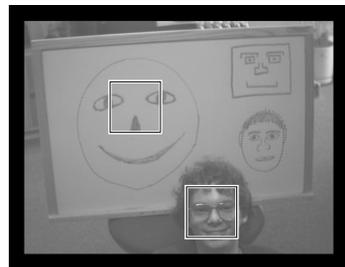
alarms=43/1960
l=2



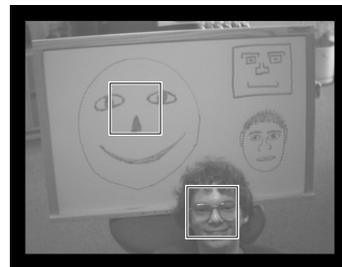
alarms=30/1960
l=3



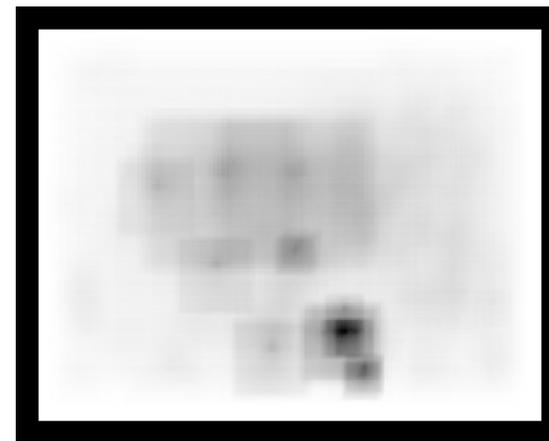
alarms=12/1960
l=4



alarms=2/1960
l=5



alarms=2/1960
l=6



Extrait Hichem Sahbi [3].

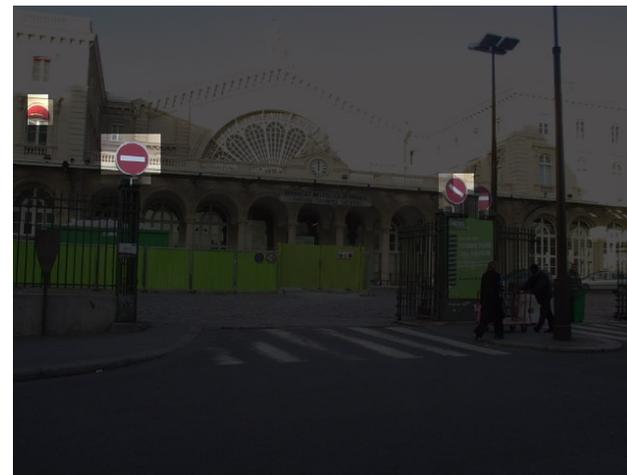
Prédétection

Il est aussi possible de faire une chaîne de détecteurs utilisant des représentation des images de plus en plus complètes. Prédécteur : détecteur en début de chaîne.



Prédétection

Il est aussi possible de faire une chaîne de détecteurs utilisant des représentation des images de plus en plus complètes. Prédécteur : détecteur en début de chaîne.



Avantages et difficultés

Avantages :

- Méthode générique : il faut bien choisir le type de signatures et la mesure de similarité.
- Ces choix sont une façon d'introduire des connaissances a priori.

Difficultés :

- Représentativité de la base d'apprentissage et qualité
- Comment introduire d'autres connaissances a priori?

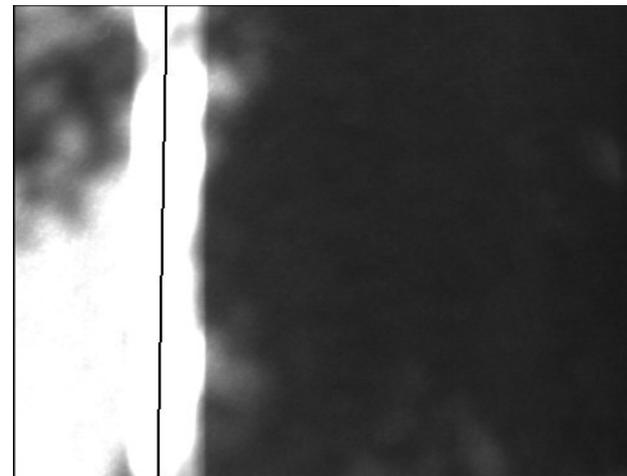
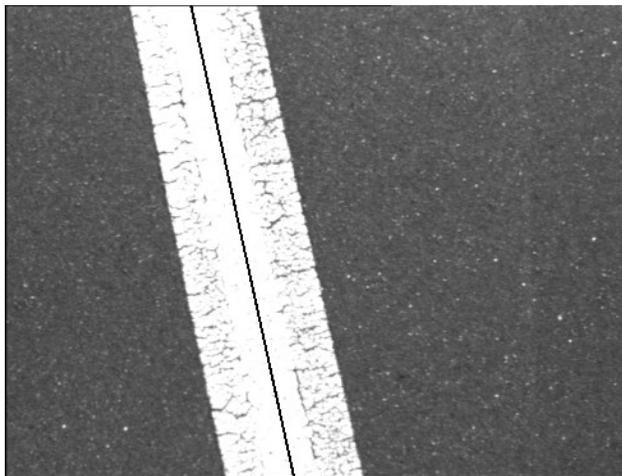
⇒ approches par modèle paramétré.

Partie 4

1. Détecter un objet modélisé par une image?
2. Détecter un objet avec une modélisation Gaussienne de la variabilité de son aspect?
3. Détecter un objet avec une modélisation SVM de la variabilité de son aspect?
4. Détecter un objet avec une modélisation paramétrée de la variabilité de son aspect?
 - Régression robuste.

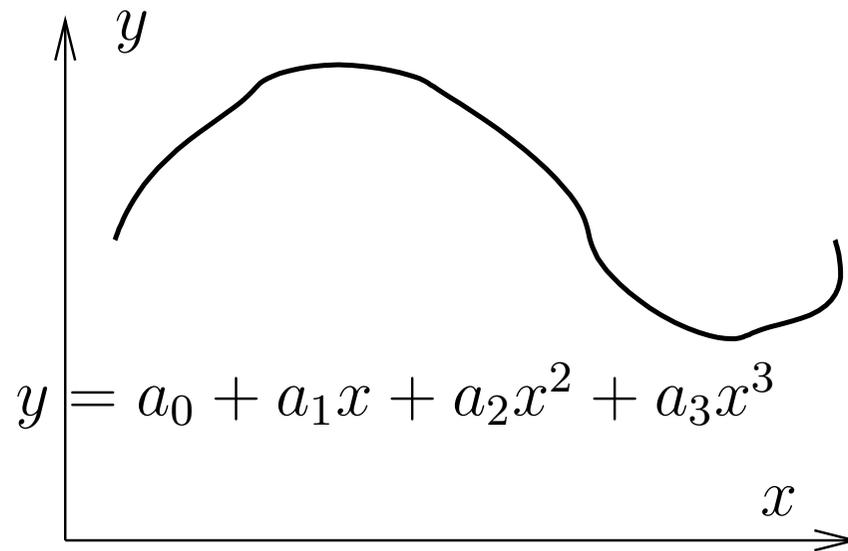
Un nouvel exemple

Détection des marquages routiers par une caméra embarquée sur un véhicule.



Extrait : LIVIC [1].

Représenter une courbe



Courbe 2D explicite de paramètre A :

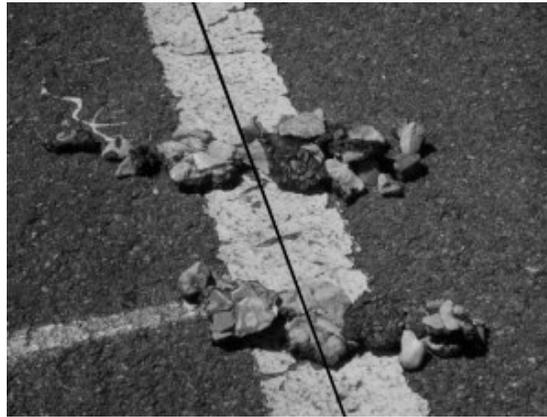
$$y = \sum_{i=0}^d f_j(x) a_j = X(x)^t A$$

⇒ linéaire par rapport aux paramètres. Si $f_j(x) = x^j$,
c'est une courbe polynomiale.

⇒ simple problème de régression de courbe.

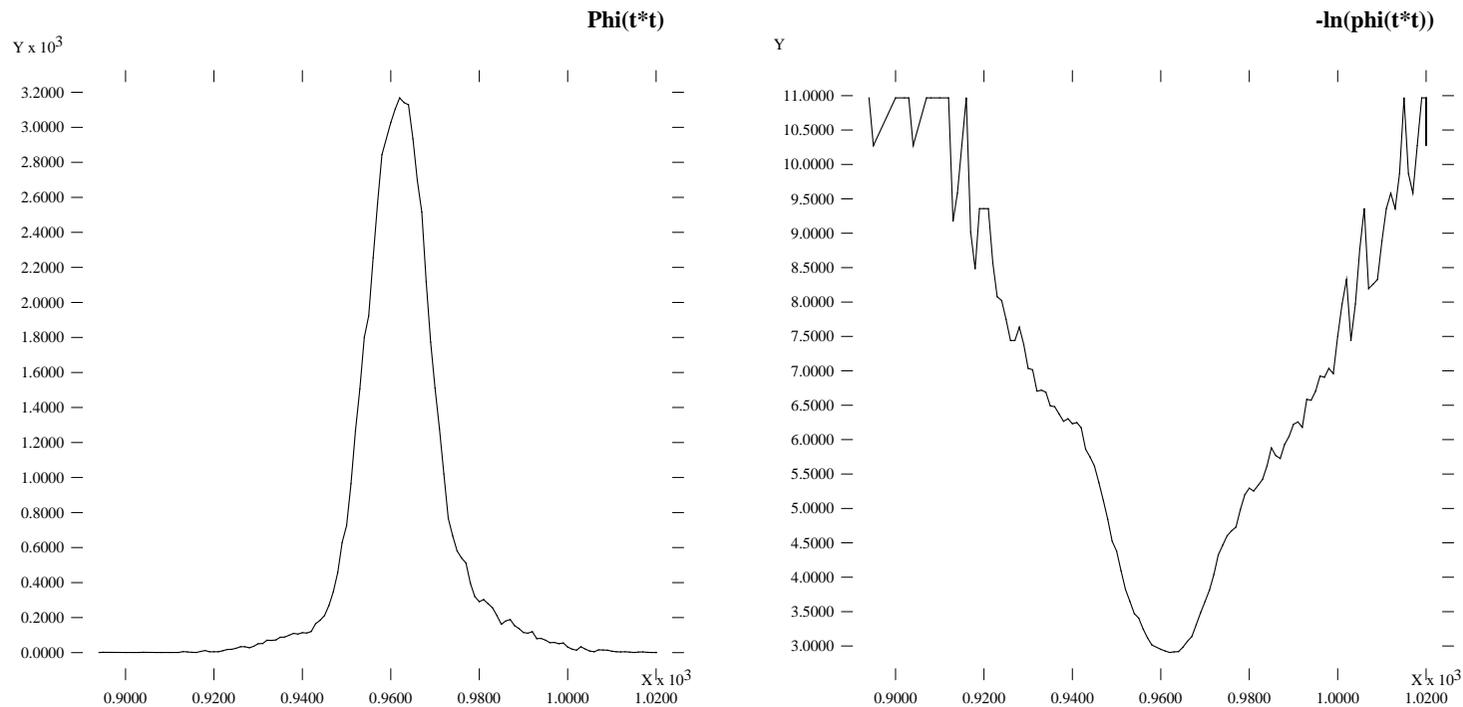
Modéliser les perturbations (1)

Même image avec des perturbations différentes :



150 images. Référence donnée à la main.

Modéliser les perturbations (2)



Les perturbations ne sont pas Gaussiennes, elles semblent plutôt avoir une distribution du type Laplace :

$$prob(x) = \frac{1}{2\sigma} e^{-\frac{|x|}{\sigma}} \text{ où } \sigma \text{ est l'échelle du bruit.}$$

Régression Robuste

La probabilité d'avoir le point (x_i, y_i) , sachant que la courbe a pour paramètres A , est:

$$prob_i((x_i, y_i)/A) \propto e^{-\frac{1}{2}\phi\left(\left(\frac{X_i^t A - y_i}{\sigma}\right)^2\right)}$$

Principe du maximum de vraisemblance : trouver A qui maximise la probabilité des données. Après application de la fonction $-\log$, c'est équivalent à trouver le vecteur de paramètres qui minimise l'erreur :

$$e(A) = \sum_{i=1}^{i=n} \phi\left(\left(\frac{X_i^t A - y_i}{\sigma}\right)^2\right)$$

Moindres carrés itérés

Initialiser le vecteur de paramètres A .

- Calculer la pondération des données :

$$w_i = \phi' \left(\frac{X_i^t A - y_i}{\sigma} \right)^2.$$

- Calculer le vecteur de paramètres en résolvant le système linéaire : $\sum_{i=1}^{i=n} w_i X_i X_i^t A = \sum_{i=1}^{i=n} w_i X_i y_i$.
- Itérer jusqu'à stabilisation.

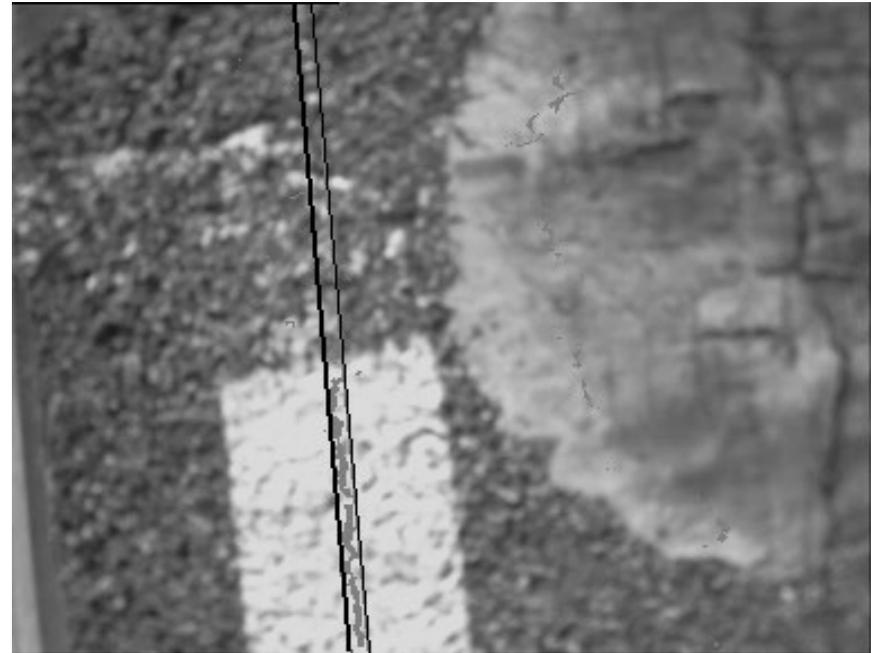
⇒ Conséquence du théorème de Khun et Tucker.

⇒ Convergence graduelle en faisant décroître σ vers la valeur correcte.

Résultats



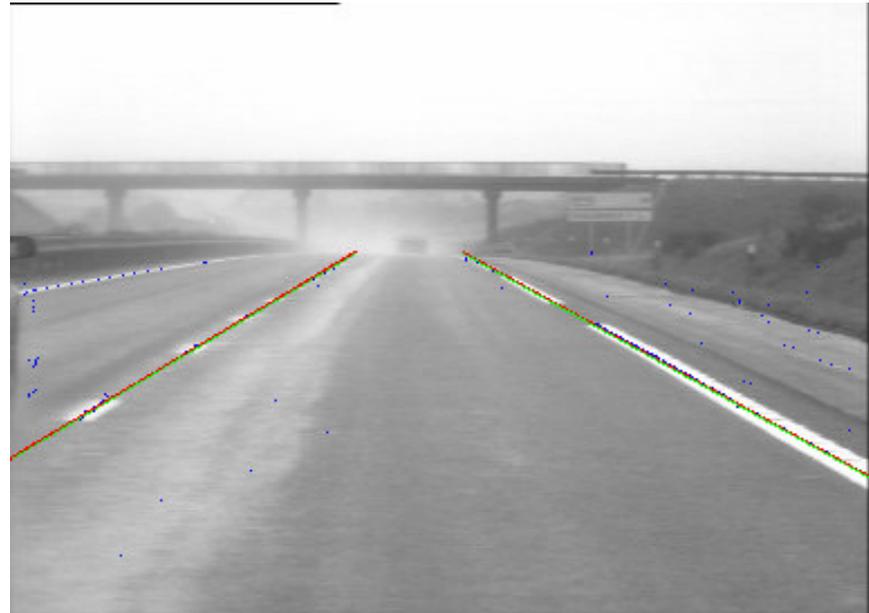
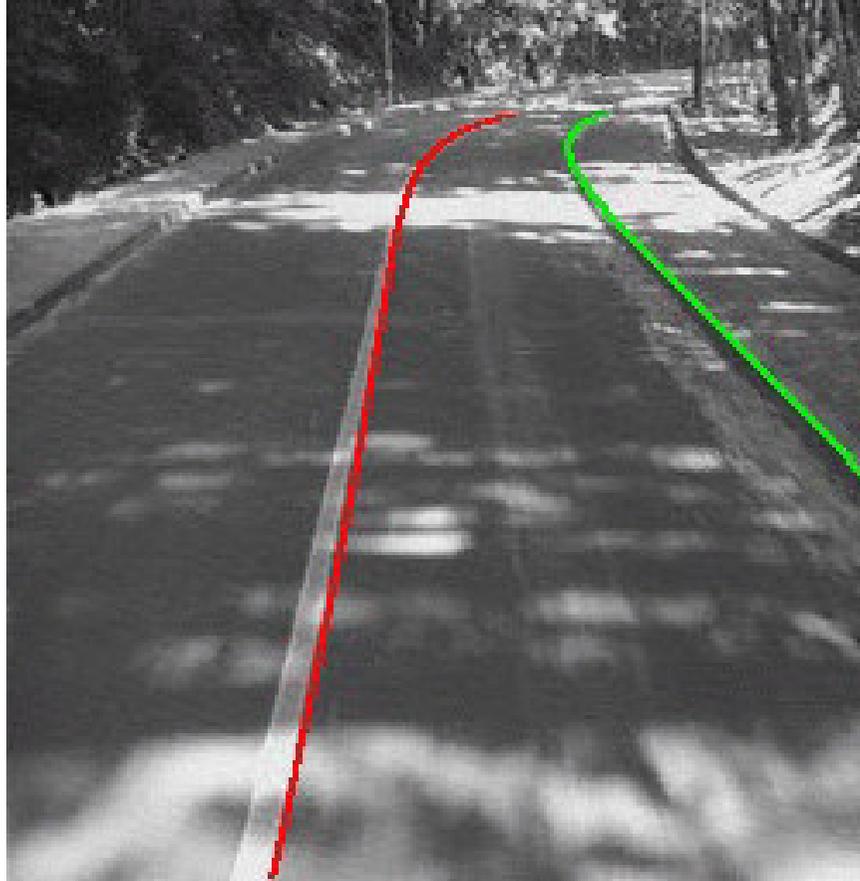
Gauss



Cauchy

Bruit de Cauchy : $prob(x) \propto \frac{1}{1 + \frac{x^2}{\sigma^2}}$.

Suivi



Détection et suivi de la voie.

Conduite Automatique



Extraits : LIVIC.

Détection d'accident

Quel modèle?



Extraits : INRETS/CITILOG (www.citilog.fr).

Recommandations

- Il faut choisir avec une grande attention l'objet que l'on veut détecter dans les images pour chaque application.
- Il faut bien modéliser les perturbations pour mieux choisir les types d'algorithmes et leurs paramètres.
- Il faut essayer de poser le problème comme une optimisation, car cela permet souvent d'appliquer le théorème de Kuhn et Tucker (dualité) et permet de dériver des algorithmes ayant une convergence rapide et garantie.

References

- [1] Anonymous. 2002.
- [2] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines (and other kernel based learning methods)*. Cambridge University Press, 2000.
- [3] H. Sahbi, D. Geman, and N. Boujema. Face detection using coarse-to-fine support vector classifiers. *In Proceedings of the IEEE International Conference on Image Processing*, pages 925–928, 2002. <http://www-rocq.inria.fr/sahbi/>.