

## **RdF – Reconnaissance des Formes**

### **Semaine 3 : attributs pour la segmentation**

**Master ASE** : <http://master-ase.univ-lille1.fr/>  
**Master Informatique** : <http://www.fil.univ-lille1.fr/>  
**Spécialité IVI** : <http://master-ivi.univ-lille1.fr/>

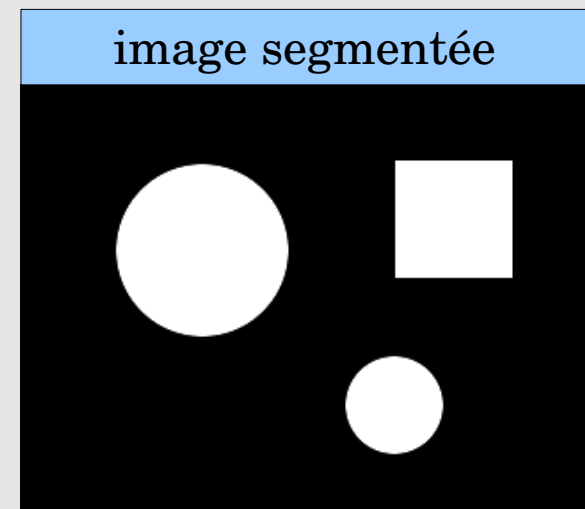
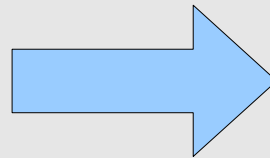
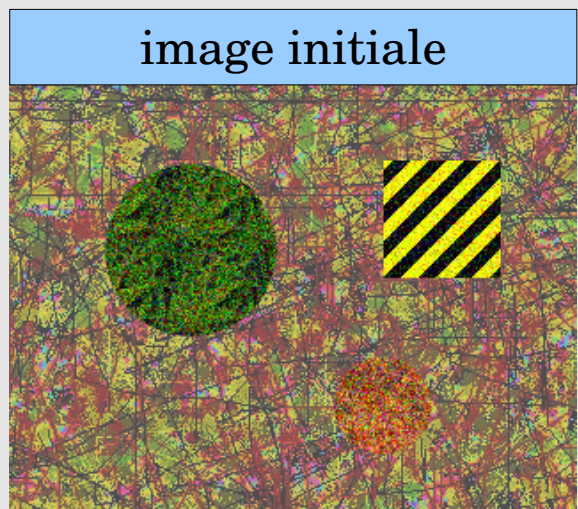
# Segmentation par classification (régions)

## Objectif

associer une **étiquette** à chaque pixel d'une image, pour obtenir une nouvelle image dans laquelle les **régions** sont marquées.

## Méthode

- 1) associer un **vecteur attribut** à chaque pixel de l'image;
- 2) **déterminer** des classes regroupant les pixels similaires;
- 3) **assigner** une classe à chaque pixel selon ses attributs.



# Plan du cours

## 1 – Apprentissage et classification

étapes d'une procédure de classification

exemple: binarisation par analyse de l'histogramme

attributs d'un pixel: niveau de gris, couleur, texture ...

## 2 – Textures

propriété complexe à définir, approche intuitive

approche statistique de la définition d'une texture

## 2 – Texture : approche par matrices de co-occurrences

matrices des co-occurrences

propriétés des matrices des co-occurrences: attributs d'Haralick

## 3 – Texture : approche par filtrage

filtres de Gabor: orientation et étendue d'une texture

décomposition en ondelettes

# Classification

## Étapes d'un processus de reconnaissance

- 1) **acquisition** des données grâce à un ou plusieurs capteurs;
- 2) extraction d'**attributs** caractéristiques des observations;
- 3) catégorisation ou description des observations: **classification**.

## Extraction des attributs d'une observation

grâce à une **fonction**  $A(.)$  définie sur l'espace des observations et à valeur dans l'espace des attributs (souvent sous-espace de  $\mathbb{R}^n$ ):

$$A: O \rightarrow \mathbb{R}^n , \\ o \rightarrow (A_1(o), A_2(o), \dots, A_n(o))^T$$

## Classification

fonction  $L(.)$  définie sur l'espace des attributs vers un ensemble de cardinal fini, dont les éléments sont les **étiquettes**.

$$L: \mathbb{R}^n \rightarrow \{1, \dots, C\} , \quad x \rightarrow L(x)$$

# Apprentissage automatique

## Comment déterminer les étiquettes ?

la combinaison des fonctions  $A(.)$  et  $L(.)$  définit une **partition** de l'espace des observations **en classes** (une classe par étiquette). l'apprentissage est l'étape durant laquelle sont déterminées les fonctions  $A(.)$  et  $L(.)$  les **plus adaptées** à classification.

## Classification supervisée

on connaît a priori le **nombre de classes** et on dispose d'un ensemble d'observations dont les **étiquettes sont connues**. cet ensemble est appelé **ensemble d'apprentissage**.

## Classification non supervisée

on doit déterminer le nombre de classes et la partition de l'espace des observations en analysant directement les données. mise en évidence des **groupes** d'observations (**clusters**).

## Exemple : binarisation par analyse de l'histogramme (1/2)

### Binarisation

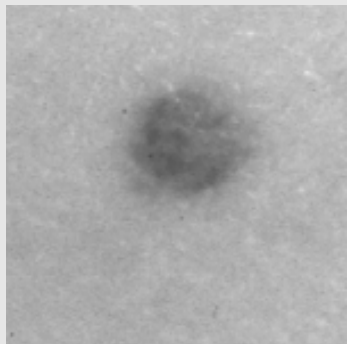
on suppose que l'image ne contient que **deux** types de régions: les **objets** et le **fond**.

la binarisation consiste à affecter à chaque pixel une **étiquette** indiquant s'il appartient à un objet (1) ou au fond (0).

méthode de **classification binaire**, car uniquement 2 classes.

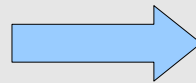
### Vecteur attribut du pixel

vecteur à **une seule dimension** : le niveau de gris du pixel.



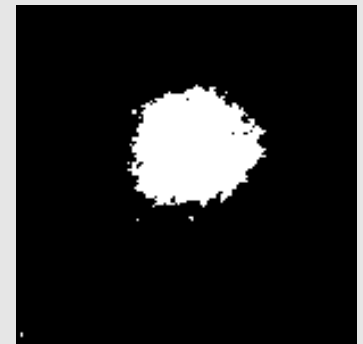
$$A: \mathbb{R}^2 \rightarrow \mathbb{R},$$

$$P = (x \ y)^T \rightarrow A(P)$$



$$L: \mathbb{R} \rightarrow \{0,1\},$$

$$g \rightarrow L(g)$$



## Exemple : binarisation par analyse de l'histogramme (2/2)

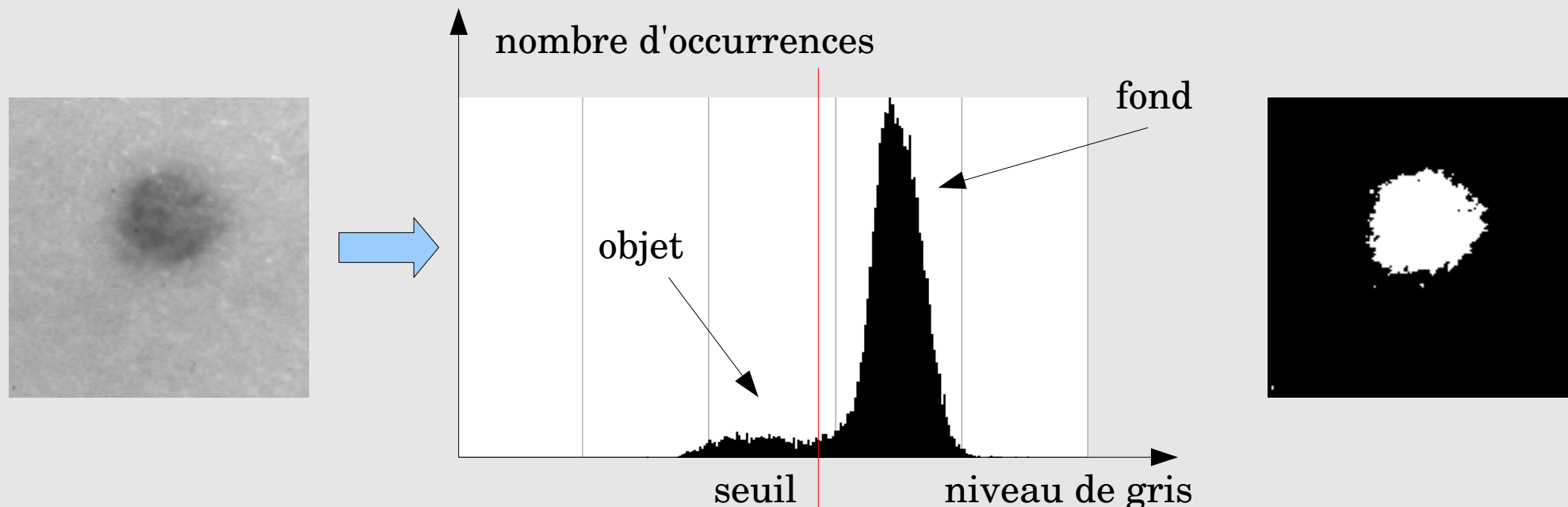
### Classification = seuillage

le seuillage est une **partition** de  $\mathbb{R}$  en deux sous-espaces, ex:

$$L_s: \mathbb{R} \rightarrow \{ \cdot, \backslash \},$$

$$g \rightarrow \cdot \text{ si } g < s, \text{ et } g \rightarrow \backslash \text{ si } g \geq s$$

### Histogramme des niveaux de gris





## Attributs d'un pixel

### Propres au pixel

niveau de gris : mesure de l'**intensité** lumineuse

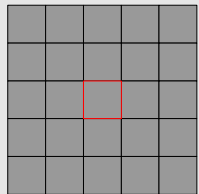
composantes **couleur** : mesures de l'éclairement dans différentes gammes de **longueurs d'ondes**

et ensuite ???

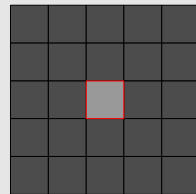
### Attributs du pixel et de son voisinage

pour définir un vecteur d'attributs, il faut disposer d'un nombre **plus élevé** d'informations, donc on utilise un **voisinage**.

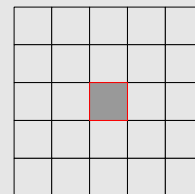
exemple: pixel et son voisinage 5x5



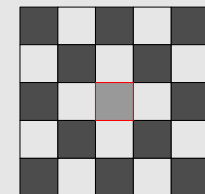
région uniforme



isolé dans du foncé



isolé dans du clair



isolé dans un damier

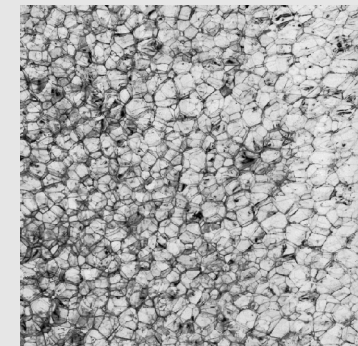
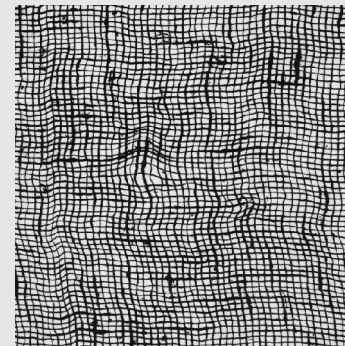
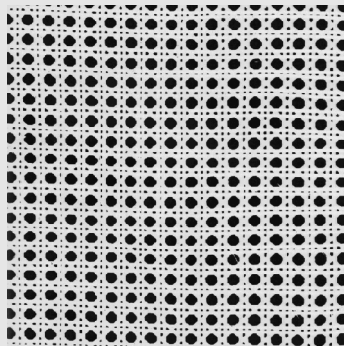
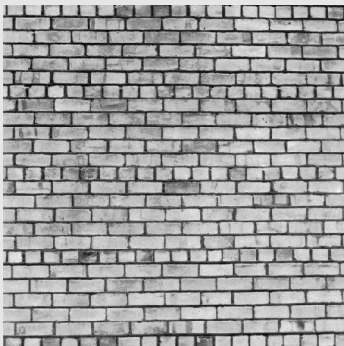
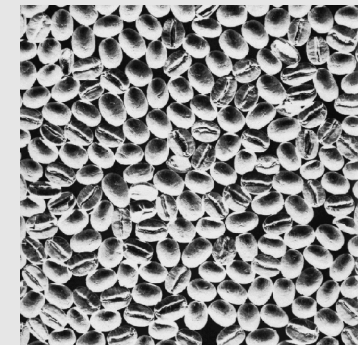
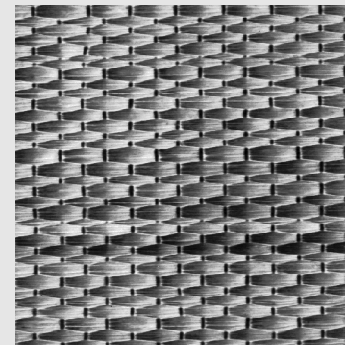
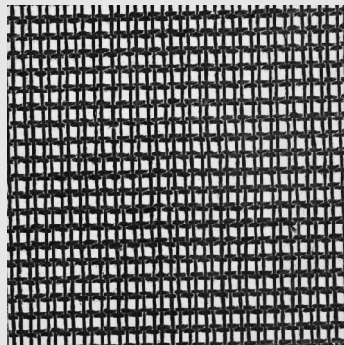
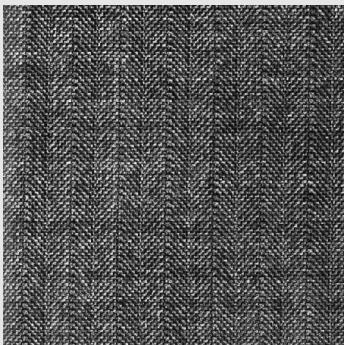


## Texture (1/2)

### Définition ?

**variations** de l'intensité lumineuse dans le voisinage, qui peut refléter des **propriétés** de l'objet (rugosité, granularité, etc.)

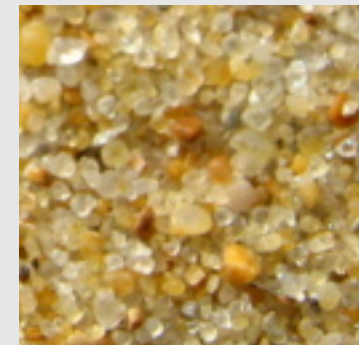
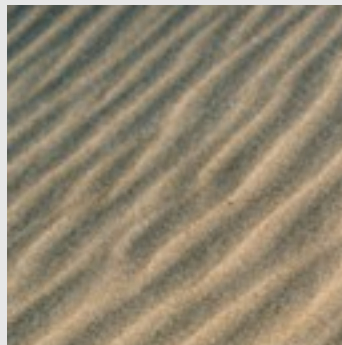
### Exemples, base de textures de Brodatz



## Texture (2/2)

### Problèmes

il n'existe pas de modèle mathématique suffisamment **général**  
pour décrire toutes les **configurations** possibles d'un voisinage.  
il n'existe pas de **distance** entre deux textures différentes.  
les propriétés d'une texture sont très liées à l'**échelle** de l'image.



### Approches principales pour quantifier les textures

**statistiques** calculées sur tous les pixels du voisinage  
matrices des co-occurrences et **indices d'Haralick**  
**modélisation** paramétrique, par exemple ondelettes de **Gabor**

## Attributs statistiques

### Calculés à partir de l'histogramme

un histogramme  $h(g)$  des niveaux de gris  $g$  est calculé sur un voisinage de taille fixe centré sur **chaque pixel** de l'image

**moyenne**  $M = \sum_g g \cdot h(g)$

**variance**  $\sigma^2 = \sum_g (g - M)^2 \cdot h(g)$

**cumulant**  $C = \sum_g (g - M)^3 \cdot h(g)$

**kurtosis**  $K = \sum_g (g - M)^4 \cdot h(g)$

**entropie**  $E = - \sum_g \log[h(g)] \cdot h(g)$

# Matrice des co-occurrences

## Principe

déterminer le nombre de **configurations** dans un voisinage pour lesquelles les niveaux de gris d'une **paire** de pixels (suivant cette configuration) prennent des valeurs particulières.

## Calcul

$T_x$  et  $T_y$  sont les dimensions de la **région** analysée,

$C_x$  et  $C_y$  sont les dimensions de l'ensemble des **configurations**,

$G$  est le **nombre** de niveaux de gris:  $0 \leq g \leq G-1$

la matrice des co-occurrences, de dimension  $G \times G$ , est donnée par:

$$M_{dx, dy}(u, v) = \sum_{i=1}^{N_x - C_x} \sum_{j=1}^{N_y - C_y} \delta \left[ (I(x, y) - u) \cdot (I(x + dx, y + dy) - v) \right]$$

dans laquelle  $\delta[x] = 1$  ssi  $x = 0$

## Exemple de calcul

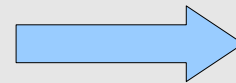
### Cas traité dans l'exemple

niveaux de gris codés entre 0 et 7 (3 bits)

configuration  $d_x = 1$  et  $d_y = 2$

région de taille 4x4

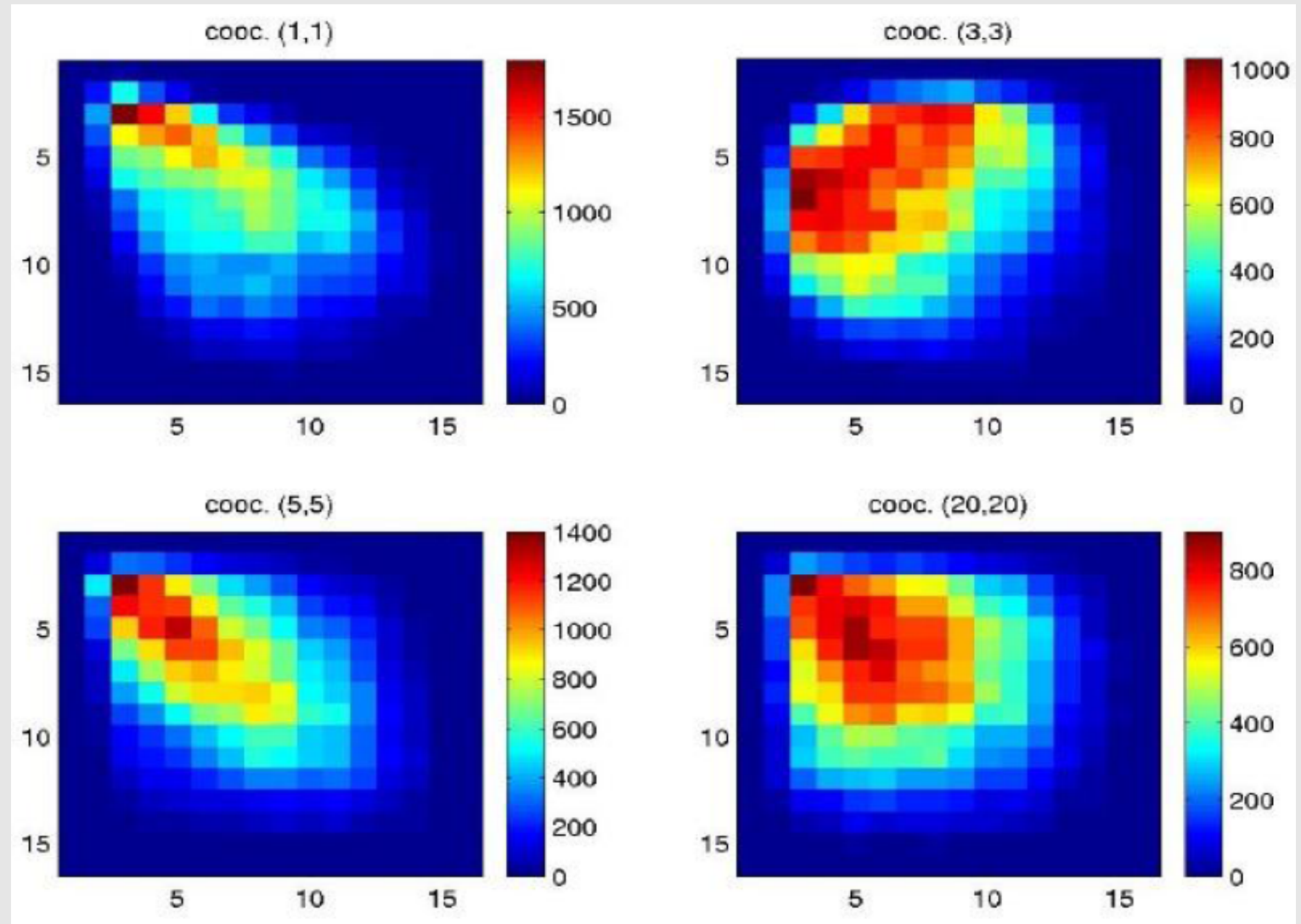
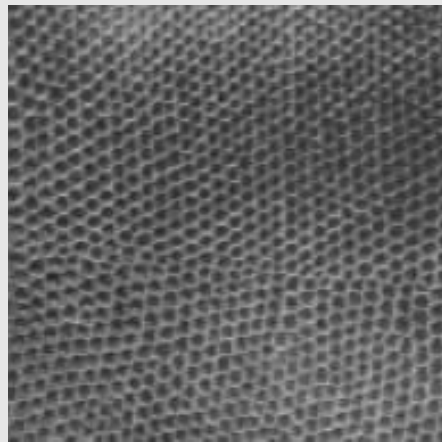
2	2	1	0
1	4	4	3
1	1	7	4
5	6	7	7



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7	3							



## Exemple sur une image (16 niveaux de gris)



# Attributs d'Haralick

## Objectif

**simplifier** la représentation des matrices des co-occurrences

## Exemples d'attributs

$(u,v)$  désignent les **coordonnées** d'un élément de la matrice  
 $M(u,v)$  désigne la **valeur normalisée** (divisée par le nb de pixels)

**moyenne**  $M = \sum_{u,v} M(u,v)$

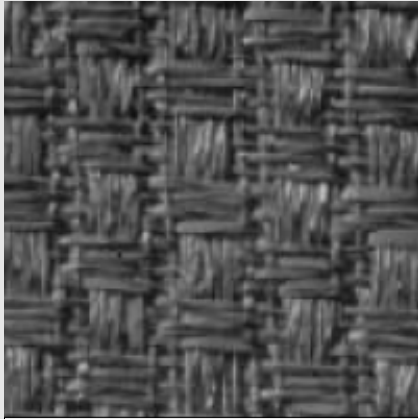
**variance en u**  $V_u = \sum_{u,v} (u - M)^2 \cdot M(u,v)$

**entropie**  $E = - \sum_{u,v} \log(M(u,v)) \cdot M(u,v)$

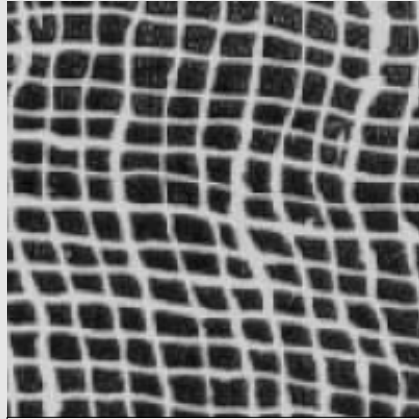
**moment différentiel inverse**  $MDI = - \sum_{u,v} \frac{1}{1 + (u - v)^2} \cdot M(u,v)$



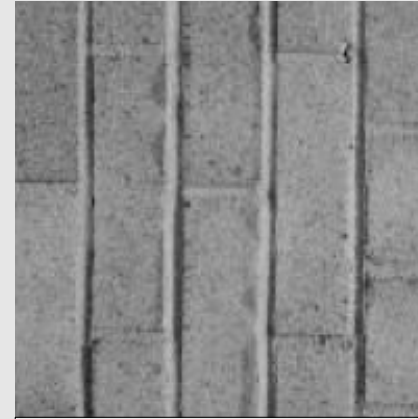
## Exemple d'attributs d'Haralick



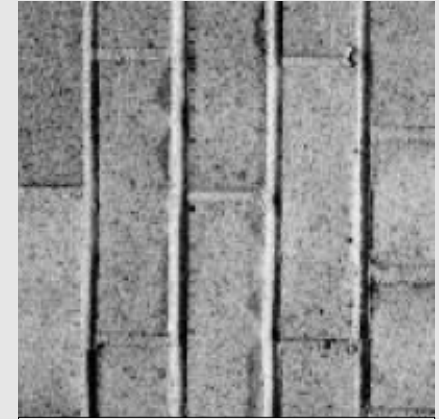
contraste = 120



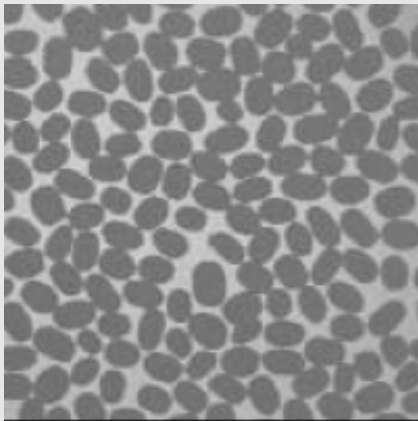
contraste = 626



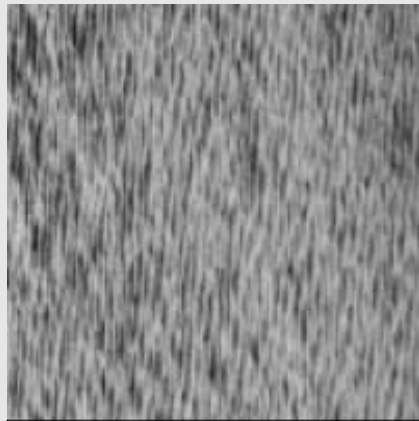
contraste = 42



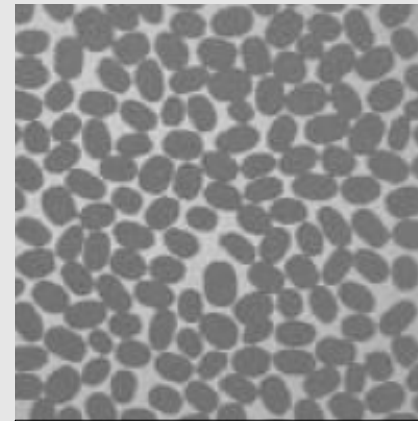
contraste = 142



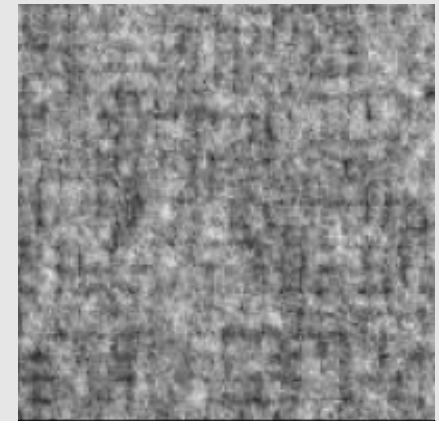
entropie = 6.7



entropie = 4.7



mdi = 0.03



mdi = 0.002

# Description fréquentielle d'une texture

## Principe

une texture est une **répétition** dans l'image d'éléments avec une **fréquence spatiale** déterminée.

## Détermination de la texture

**filtrage** de l'image pour mettre en évidence les répétitions :  
par transformation de **Fourier**,  
par transformation en **ondelettes**,  
par banc de filtres adaptés, exemple : filtres de **Gabor**

## Fonction de Gabor

produit d'une fonction **gaussienne** par un sinus ou un cosinus :

$$G_c(x, y) = \cos(\omega_x x + \omega_y y) \cdot e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad G_s(x, y) = \sin(\omega_x x + \omega_y y) \cdot e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

# Analyse par filtres de Gabor

## Principe

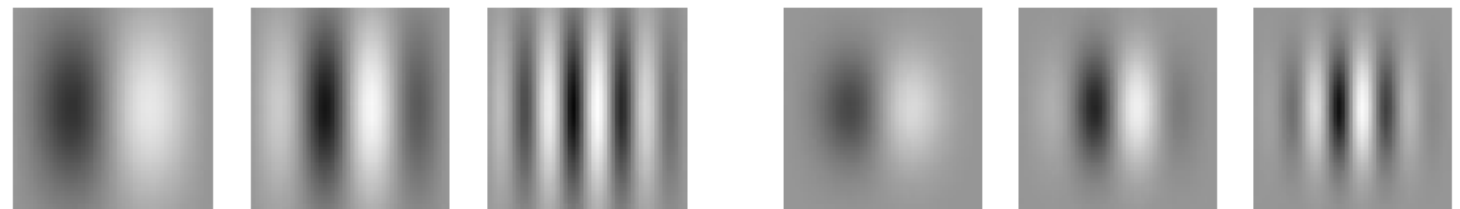
une texture est caractérisée par les **statistiques** (moyenne, variance, etc.) calculées sur les réponses d'un **ensemble** de filtres de Gabor.

## Réponse d'un filtre

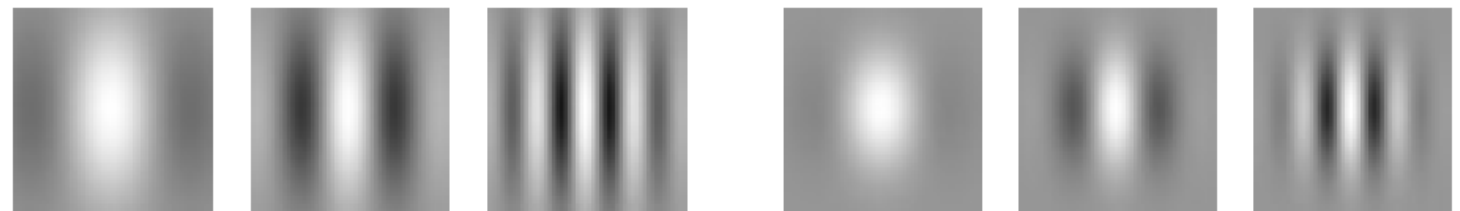
**convolution** de l'image avec une fonction de Gabor donnée

## Exemples de fonctions de Gabor pour l'analyse de textures

anti-symétriques  
fonction sinus



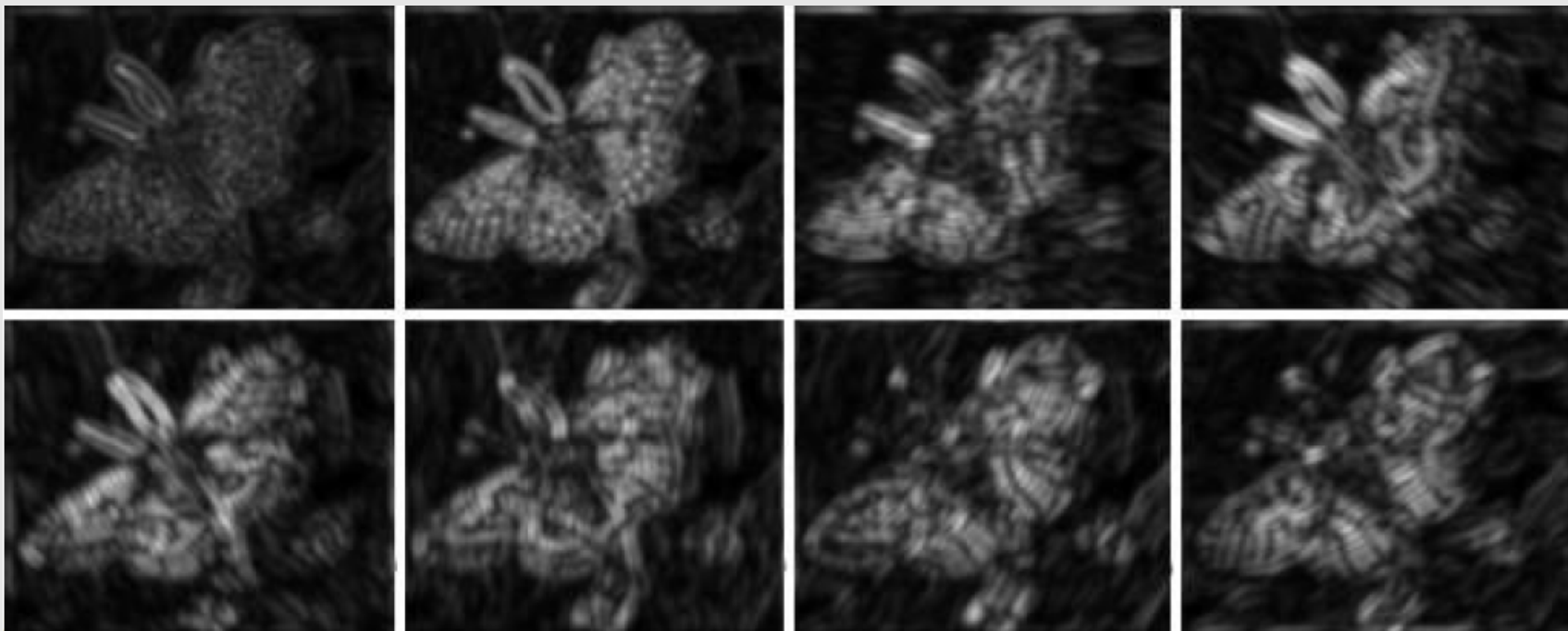
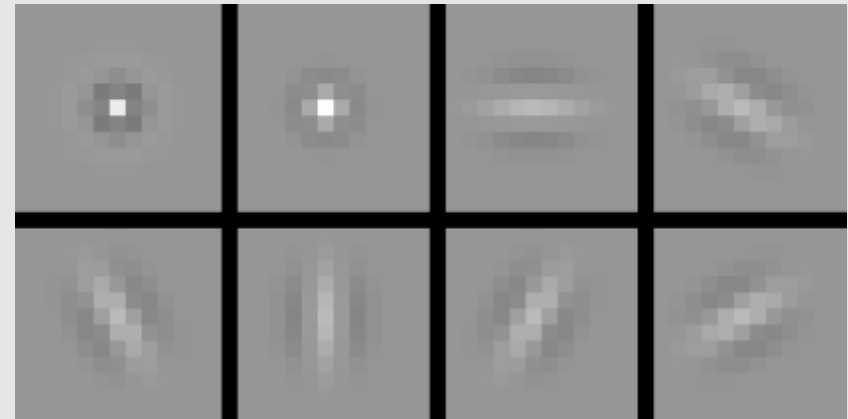
symétriques  
fonction cosinus



## Exemple de filtrage d'image



ensemble de  
filtres de Gabor



## Pour approfondir

**Duda, Hart, Stork, « Pattern Classification », 2ème édition, Wiley-Interscience, 2001.**

<http://rii.ricoh.com/~stork/DHS.html>

**cours de Ronan Fablet sur les textures (ENST Bretagne)**

[perso.telecom-bretagne.eu/ronanfablet/data/coursTexture\\_F4B202v2.pdf](http://perso.telecom-bretagne.eu/ronanfablet/data/coursTexture_F4B202v2.pdf)