

# Cours : Indexation d'informations MultiMedia

## Chapitre 2: Les descripteurs globaux des images

présenté par:  
Mohamed Sahbi Bahroun

# Plan

---

- 1) Introduction
- 2) Descripteurs de couleur
- 3) Descripteurs de texture
- 4) Descripteurs de forme

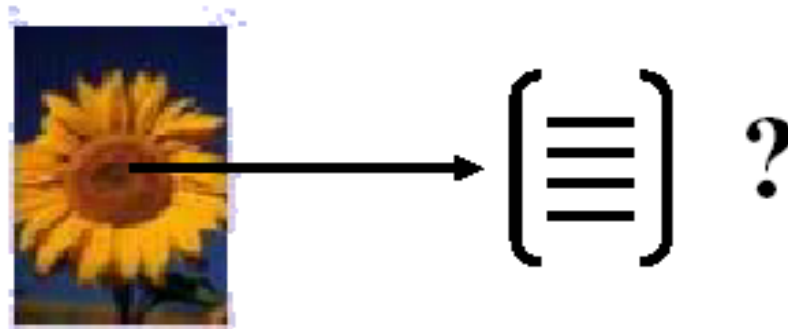
---

# 1- Introduction

# Indexation par descripteurs globaux

---

- Extraire automatiquement d'une image des descripteurs **significatifs** et **compacts**, qui seront utilisés pour la recherche ou la structuration



- Prise en compte directe du contenu visuel  
... mais manifestation fréquente du **gap sémantique**

# Description globale de l'image

---



Description  
des couleurs



Description  
des textures



Description  
des formes

Assembler les différents types de description pour  
décrire de façon plus complète chaque image

---

## 2- Descripteurs de couleur

# Les signatures de couleur

---

- Modélisation de la distribution des couleurs
  - Couleur moyenne
  - Histogramme des couleurs
  - Moments
  - ...
- Choix d'un espace de représentation des couleurs

# Espaces colorimétriques

- Espace de représentation RGB (Red, Green, Blue)
  - ▶ représentation additive des couleurs sur la base de la lumière émise
  - ▶ le plus souvent, chaque canal est représenté sur 8 bits  
⇒ 16,7 millions de couleurs possibles
- YUV (Y: luminosité, U et V: chrominance)
  - ▶ utilisé dans PAL
- YCbCr, Y'CbCr
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.13983 \\ 1 & -0.39465 & -0.58060 \\ 1 & 2.03211 & 0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix}$$
  - ▶ Mise à l'échelle de YUV pour les signaux digitaux
  - ▶ utilisé dans JPEG, MPEG2
  - ▶ permet une quantification plus grossière des valeurs de chrominance
- HSV (Hue, Saturation, Value) = HSB
  - ▶ Bonne représentation perceptuelle des couleurs, utilisé dans les logiciels
- CMYK : Espace de couleur soustractive (application des encres)



# Espaces colorimétriques

- Afin de créer des images encore plus riches en
- l'idée de mélanger des couleurs primaires en « couches
- Il faut savoir qu'il existe deux systèmes de représentation des couleurs par mélange, selon qu'on les reproduisent sur un écran d'ordinateur ou sur support papier via une imprimante :

La synthèse additive : c'est le phénomène qui se passe lorsqu'un écran affiche une image par la lumière. On part du noir (lumière éteinte) et on va vers le blanc. L'addition du rouge, du vert et du bleu donne le blanc: RVB

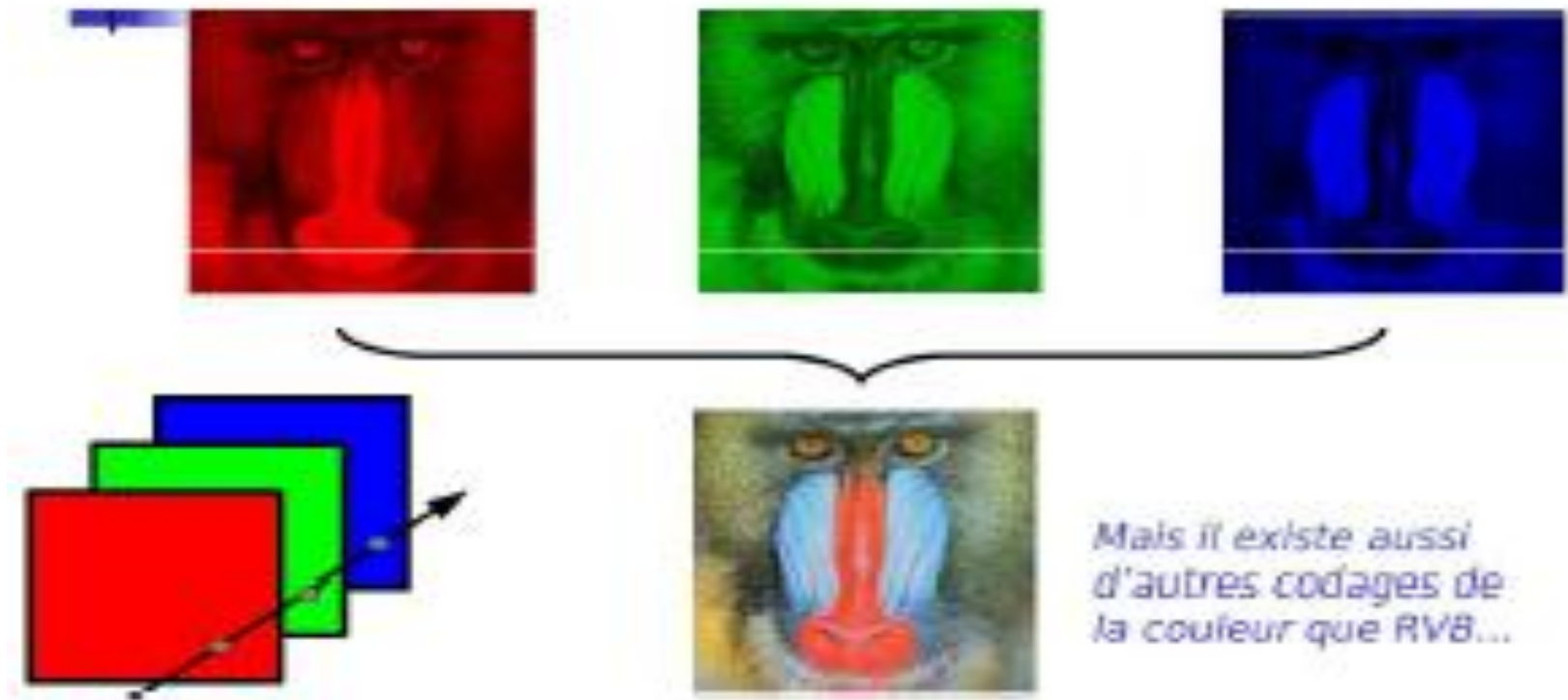


La synthèse soustractive : c'est le phénomène qui se passe lorsqu'on mélange des pigments colorés en peinture. On part du blanc (support papier) pour aller vers le noir. L'addition du Cyan, du Magenta et du Jaune donne le Noir: CMJN



# Image couleur RVB

---



# Moments de couleur

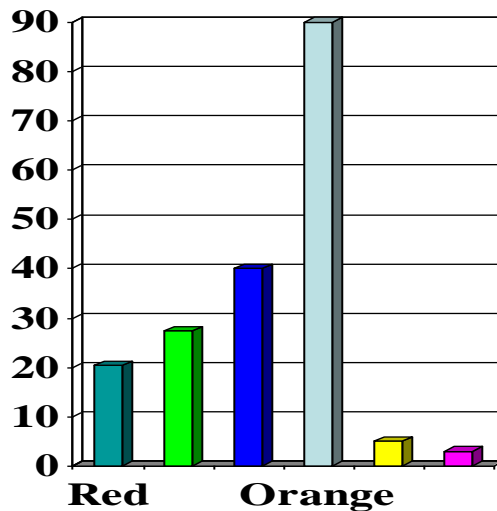
---

- **Moyennes** (image  $(R[i][j], G[i][j], B[i][j])$ ,  $N$  pixels):  
 $mR = (\sum R)/N$ ,  $mG = (\sum G)/N$ ,  $mB = (\sum B)/N$
- **Moyennes + variances : + composantes :**  
 $mRR = (\sum (R-mR)^2)/N$ ,  $mGB = (\sum (G-mG)(B-mB))/N$ , ...
- **Moments « supérieurs » :**  $mRRR$ ,  $mRGB$ , ...
- **Moments associés aux composantes spatiales :**  
 $mRX = (\sum (R-mR)(X-mX))/N$ ,  $mRGX$ ,  $mBXY$ , ...
- **Avantages :** faciles/rapides à calculer, descripteurs très compacts,
- **Inconvénients :** peu précis et peu sélectifs.

# Histogramme

L'histogramme d'une image  $I$  est défini comme suit:

Pour une couleur  $C_i$ ,  $H_{ci}(I)$  représente le nombre de pixels de couleur  $C_i$  dans l'image  $I$ .



- (+) invariant par translation et rotation
- (-) dépend des conditions d'illumination et de l'angle de vue
- (-) ne tient pas compte de la position relative des pixels

# Histogramme : algorithme

---

***Histogramme (Matrice I [ nbcol][nbligne ], entier nbcol,  
entier nbligne)***

*Debut*

*H = new Array [0..255]*

*/\*Initialisation du tableau\*/*

*Pour i= 0 à 255*

*H[i]= 0;*

*Fin pour*

*Pour i= 0 à nbcol*

*Pour j= 0 à nbligne*

*H[ I[i][j] ] ++;*

*Fin pour*

*Fin pour*

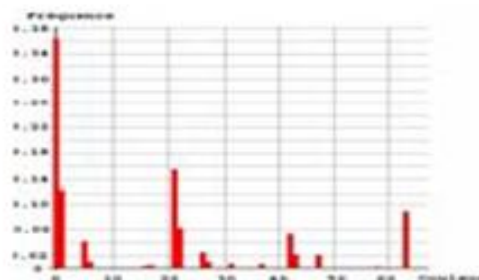
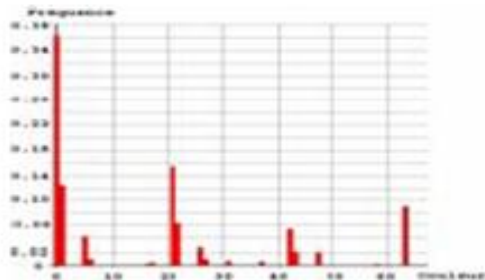
*Fin*

# Limitations histogramme

## ○ Limitations



Contenu visuel différent



Histogrammes similaires

→ **Problème** de dimension avec l'histogramme : Si l'image  $I$  est codée sur 8 bits alors taille histogramme = 256  
**Descripteur non compact**



# Solution 1

---

calcul de paramètres statistiques à partir de l'histogramme

$$Moy = \sum_{i=0}^{2^k-1} i * H(i)$$
$$asymétrie = \frac{\left[ \sum_{i=0}^{2^k-1} (i - Moy)^3 * H(i) \right]^2}{E^6}$$

$$E = \sqrt{\sum_{i=0}^{2^k-1} (i - Moy)^2 * H(i)}$$
$$aplatissement = \frac{\left[ \sum_{i=0}^{2^k-1} (i - Moy)^4 * H(i) \right]}{E^4}$$

# solution 2 : Quantification des images

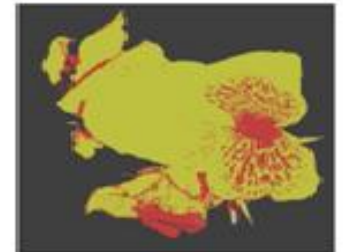
- On ne prend qu'un nombre fini de niveaux gris ou de couleurs :
  - niveaux de gris: 256
  - nombre de couleurs:  $256 * 256 * 256$
- Conséquences :
  - perte de qualité
  - apparence d'artefacts



$256 * 256 * 256$   
= 16777216 colors



$8 * 8 * 8$   
= 512 colors



$2 * 2 * 2 = 8$  colors



# Distance entre histogrammes

---

❖ Intersection :  $d_{SB} (H^1, H^2) = 1 - \frac{\sum_{i=1}^n \min (h_{ci}^1, h_{ci}^2)}{\sum_{i=1}^n h_{ci}^1}$

❖ chi2 :  $d(h_1, h_2) = \sum_{i=1}^N \frac{[h_1(i) - h_2(i)]^2}{h_1(i) + h_2(i)}$

# Exemple de recherche à base de couleur



---

## 3- Descripteurs de texture

Définition de texture

Histogramme pondéré

Corrélogramme

Matrices de cooccurrence

Descripteurs fréquentiels

# Textures

---

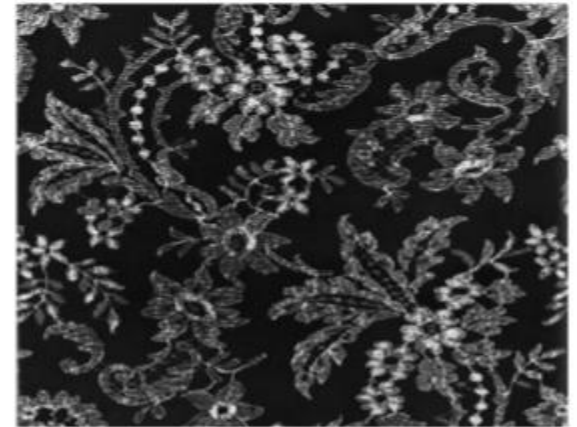
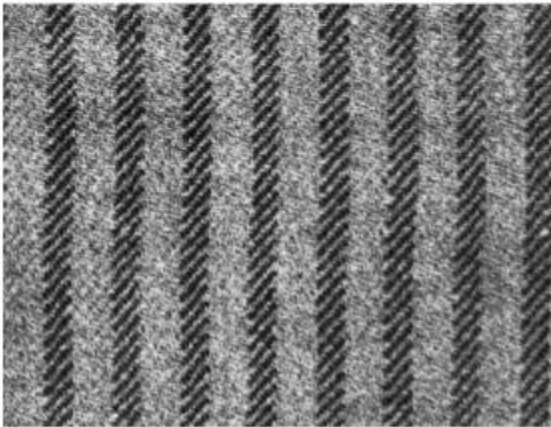
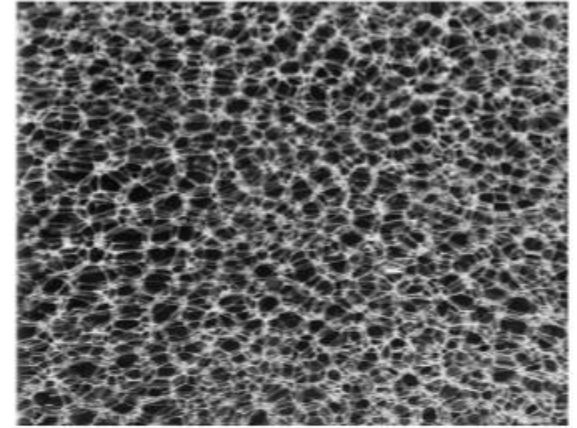
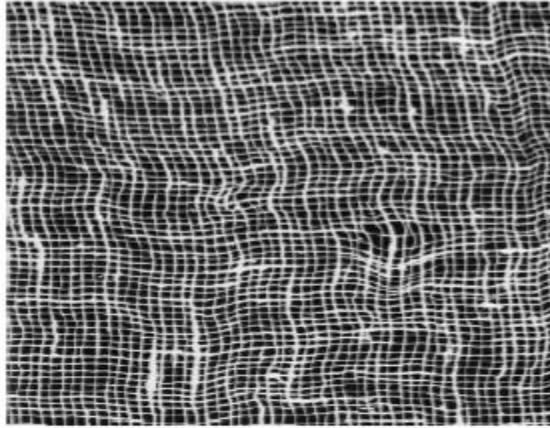
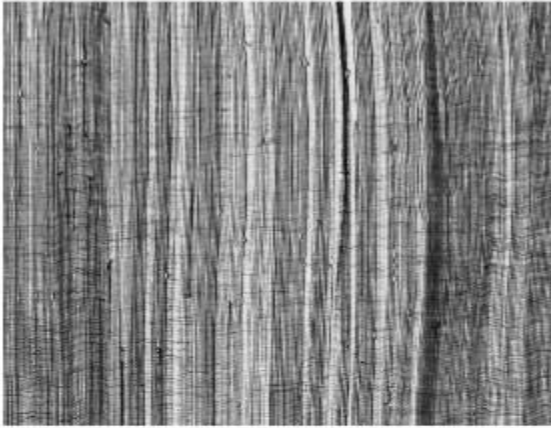
La texture peut être définie comme une variation répétitive de l'intensité.

La variation de l'intensité peut être :

- Régulière : répétition des couleurs d'une façon régulière (Exemple : champ d'olivier vu du ciel)
- Aléatoire : variation aléatoire de la couleur (Exemple : ciel avec des nuages, champs d'herbes et de fleurs sauvages)

# Exemple de textures

---



Exemples de textures de Brodatz

# Comment analyser des textures?

---

## 1) Statistiques d'ordre 2

- ❑ Interaction entre 1 pixel et ses voisins séparés par une distance  $d$  + direction spécifique
- ❑ Attributs calculés à partir :
  - Histogramme bidimensionnel : matrice de cooccurrence et corrélogramme, Histogrammes pondéré

## 2) Méthodes fréquentielles : filtres de Gabor, Transformée en ondelettes, Transformée de Fourier

# Histogramme de couleur pondéré

---

- ❖ chaque pixel participe à toutes les cellules de l'histogramme classique avec une pondération correspondant à la similarité entre la couleur du pixel et celle de chaque cellule de son voisinage.

$$h'(\mathbf{c}) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N w(i, j) \delta(f(i, j) - \mathbf{c}), \forall \mathbf{c} \in C$$

$w(i, j)$  est une mesure locale dans le voisinage de chaque pixel.

# Histogramme de couleur pondéré

---

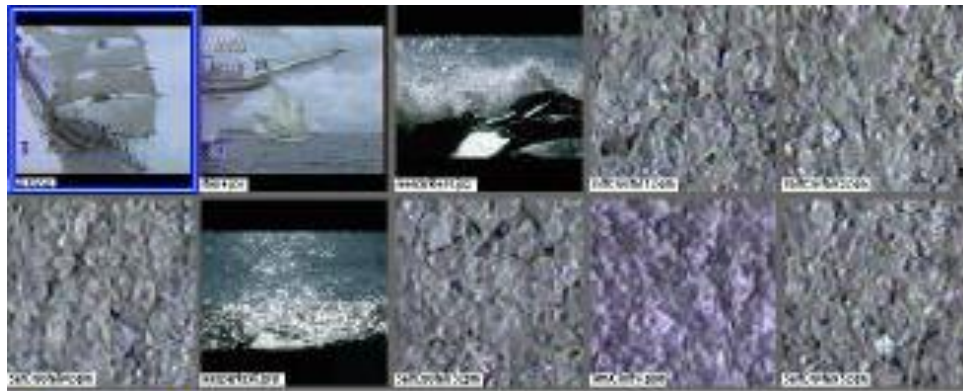
## ❖ Choix du critère de pondération $w(i,j)$ : *plusieurs critères*

- Moyenne des couleurs du voisinage ayant la même couleur que le pixel courant
- Moyenne des couleurs du voisinage ayant la même couleur que le pixel courant plus ou moins un seuil  $S$  choisi

$H'$  (histogramme pondéré) a la même dimension qu'un histogramme de couleur normal. Il inclut l'information de **texture**



# Histogramme de couleur pondéré



Signature couleur  
classique

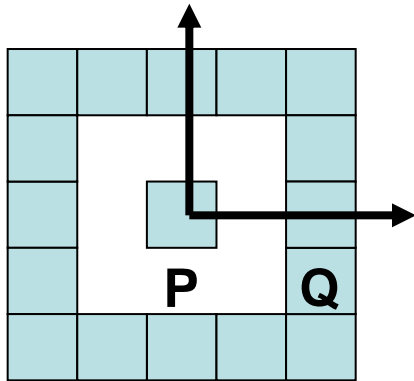


Signature intégrant la  
Répartition spatiale  
de la couleur

# Corrélogramme : principe

- ❖ Descripteur statistique d'ordre 2 qui a comme seul paramètre la distance de voisinage **d**

$$Corr_{C_i, C_j}^{(d)}(I) = \left\{ \begin{array}{l} (p_1, p_2) \in I / \\ p_2 \in I_{C_j}, \\ p_1 \in I_{C_i}, \\ |p_1 - p_2| = d \end{array} \right\}$$



$$D_8(p, q) = \max(|p_x - q_x|, |p_y - q_y|) = d$$

- ❖ Le corrélogramme est une matrice de taille  $Ng \times Ng$   
 $Ng$  : nombre de niveaux de gris dans l'image I

# Corrélogramme : principe

2	2	0	0	1	0	1
0	0	2	1	1	0	0
1	0	0	1	2	1	2
0	2	1	0	2	0	0
1	0	2	1	0	1	1
2	1	1	2	1	0	0
0	0	1	1	0	2	2

Corrélogramme pour  $d = 1$

	0	1	2
0	2	6	6
1	0	0	0
2	0	0	0

Corrélogramme pour  $d = 2$

	0	1	2
0	6	7	6
1	0	0	0
2	0	0	0

# Matrices de cooccurrence : principe

- ❖ Descripteur statistique d'ordre 2 qui a comme paramètre la distance de voisinage **d** et l'angle de voisinage  $\theta$

$$Cooc_{C_i, C_j}^{(d)}(I) = \left\{ \begin{array}{l} (p_1, p_2) \in I / \\ p_2 \in I_{C_j}, \\ p_1 \in I_{C_i}, \\ |p_1 - p_2| = d, \\ angle(p_1, p_2) = \theta \end{array} \right\} \quad \begin{array}{l} \theta = \frac{2k\pi}{8}; k = 1..8 \\ D_8(p, q) = \max(|p_x - q_x|, \\ |p_y - q_y|) = d \end{array}$$

- ❖ La matrice de cooccurrence est de taille Ng\*Ng  
Ng : nombre de niveaux de gris dans l'image I

# Matrices de cooccurrence

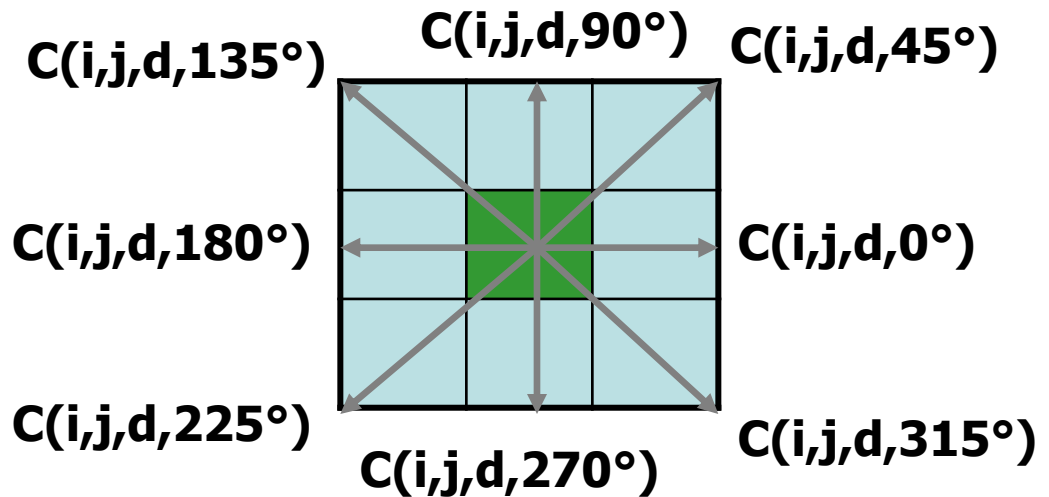
❖ Caractérisation de la directivité et la périodicité des textures

❖ Paramètres :

$$\mathcal{G} = \left\{ \frac{2k\pi}{8}, k = 1..8 \right\}$$

Une distance d  
entre pixels voisins

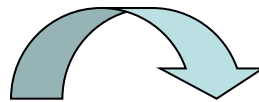
$$\begin{Bmatrix} \Delta x \\ \Delta y \end{Bmatrix} = \begin{Bmatrix} p_x - q_x \\ p_y - q_y \end{Bmatrix}$$



$$C(i, j, d, \mathcal{G}) = \sum_{m=\Delta x}^{NL-\Delta x} \sum_{n=\Delta y}^{NC-\Delta y} \delta(I(m, n) = i \wedge I(m + \Delta x, n + \Delta y) = j)$$

# Problèmes avec les MC

d	c	b
e		a
f	g	h



b	a	h
c		g
d	e	f

**Matrices de cooccurrence  
non invariantes à la rotation**

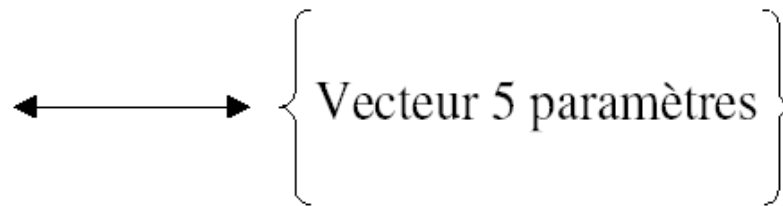
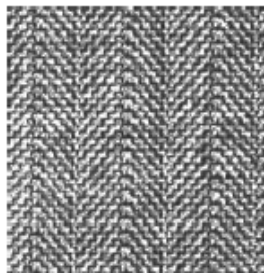
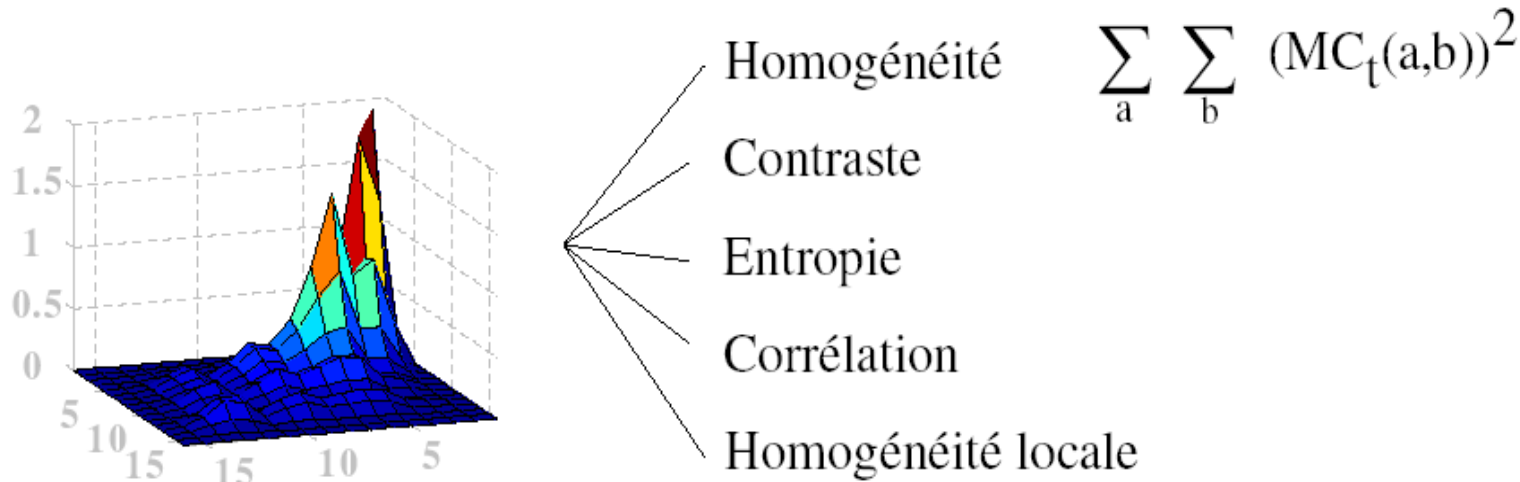
❖ On peut aussi rajouter le problème de dimension. Pour y remédier on peut :

❖ Calculer les paramètres statistiques de Haralick.

❖ Prendre comme descripteur la diagonale ou l'anti-diagonale

# Matrices de cooccurrence et paramètres de Haralick

Haralick définit 14 paramètres pour définir une texture dont 5 sont reconnus comme étant discriminant (Conners et Harlow).



# Quelques descripteurs

---

## ■ Calcul de descripteurs

$$MOYENNE = \sum_j \sum_i p(i, j)$$

$$VARIANCE_i = \sum_j \sum_i (i - MOYENNE)^2 p(i, j)$$

$$ENERGIE = \sum_j \sum_i (p(i, j))^2$$

$$INERTIE = \sum_j \sum_i (i - j)^2 p(i, j)$$

$$ENTROPIE = - \sum_j \sum_i p(i, j) \log(p(i, j))$$

$$MOMENT\ DIFFERENTIEL\ INVERSE = \sum_i \sum_j \frac{1}{1 + (i - j)^2} p(i, j)$$

**où**

**(i,j)** : coordonnées dans la matrice

**p(i,j)** : valeurs normalisées de la matrice



# Analyse fréquentielle

---

Une texture est une répétition d'éléments avec une certaine **fréquence**

Filtrage de l'image en utilisant des méthodes spectrales

- n Fourier
- n **Filtres de Gabor**
- n Analyse en ondelette...

# Histogramme de fourrier

---

- calculé uniquement sur un seul plan de l'image (niveau de gris)
- contient principalement des informations sur la texture.
- basé sur la transformée de fourrier

# Histogramme de fourrier

---

- calcul du spectre de l'image:

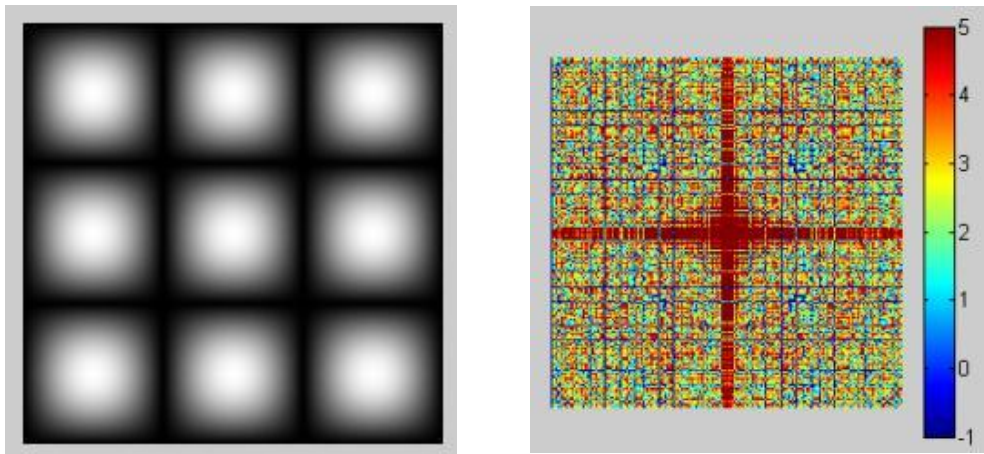


Image initiale et son spectre

# Histogramme de fourrier

---

## Transformée de Fourier

- Formulation:

$$F(u, v) = \iint I(x, y) e^{-i(ux+vy)} dx dy$$

- Invariance

phase invariante à la luminosité, stable jusqu'à un changement d'échelle de 20%

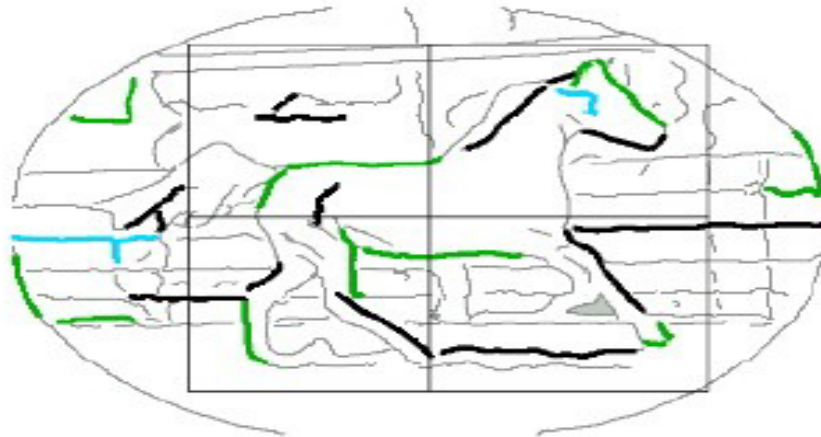
Amplitude invariante à la translation

---

# Descripteurs de forme

# Indices de forme

- Nombreux algorithmes d'extraction de formes
- Bestiaire (non exhaustif) :
  - ▶ extraction des contours
  - ▶ chaîne de points de contours ou approximation polygonale
  - ▶ mise en correspondance de graphes
  - ▶ distance de Hausdorff
- indice visuel différente des régions (1D vs. 2D)



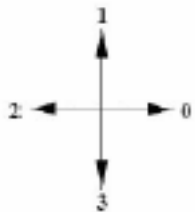
# Catégories de descripteurs

---

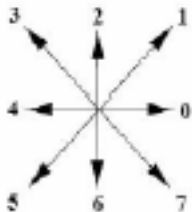
- Méthodes de description de la forme
  - Méthodes basées *contour*
    - Méthodes structurelles
      - Codage de chaîne, polygones, B-splines, etc
    - Méthodes globales
      - Périmètre, compacité, excentricité, descripteurs de Fourier, etc
  - Méthodes basées *région*
    - Méthodes structurelles
      - Enveloppe convexe, axe médian, etc
    - Méthodes globales
      - Surface, compacité, moments géométriques, etc

# Codage de Freeman

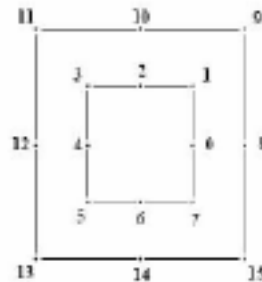
- Représentation structurelle du contour en chaînes de code
  - Codage de Freeman
    - 4, 8 ou 16 orientations



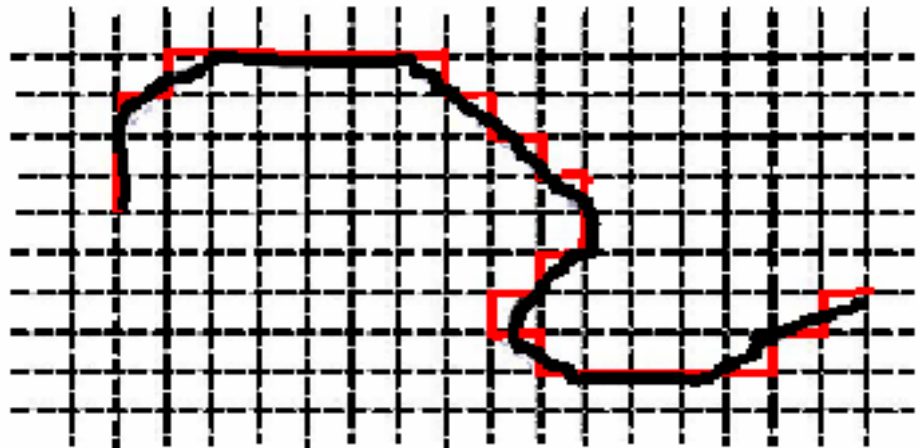
4 directions : 2bits



8 directions : 3bits



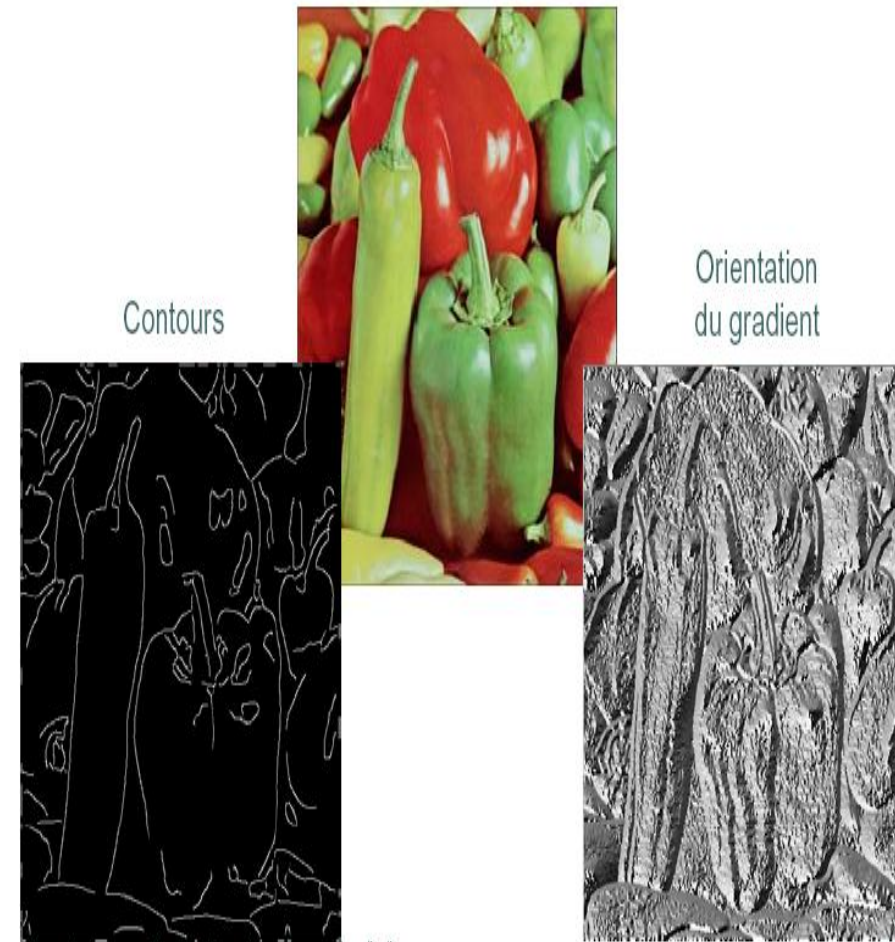
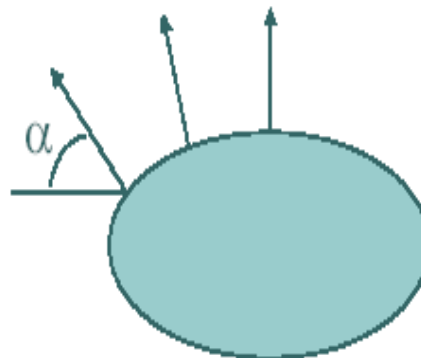
16 directions : 4bits





# Edges Orientation Histogram : EOH

- EOH : Edge Orientation Histogram
  - Principe
    - Extraction des contours de l'image
    - Histogramme de l'angle des gradients sur les contours
  - Invariance aux changements d'illumination affines



# Edges Orientation Histogram : EOH

---

- Calculé sur une image niveau de gris
- Information sur l'orientation des contours

## ***Principe:***

- Extraction des contours de l'image
  - Histogramme de l'angle des gradients sur les contours
- Invariance aux changements d'illumination affines

# EOH

---

Détection des contours : descripteur de forme

- **Algorithme**

- 1) Transformer l'image en niveau de gris
- 2) calculer les dérivées de l'image ou le gradient (derX et derY)
- 3) calculer l'orientation du gradient

Pour chaque pixel :

$\text{angle}[i] = \text{atan2}((\text{double})\text{derY}[i], (\text{double})\text{derX}[i]);$

- 4) Détecter les contours : algorithme de Canny/Deriche

➤ **Calculer l'amplitude de gradient :**

$\text{gradient}[i] = \text{sqrt}((\text{double})(\text{derX}[i] * \text{derX}[i] + \text{derY}[i] * \text{derY}[i]));$

# EOH

---

- **Etape 1 : calcul du gradient**

Méthode la plus répandue calcul d'un filtre dérivatif 1D centré  $[-1, 0, 1]$  et  $[-1, 0, 1]^T$

Dans le cas des images couleurs le gradient est calculé séparément pour chaque bande

Calcul de l'amplitude du gradient :

$$\text{gradient}[i] = \text{sqrt}((\text{double})(\text{derX}[i] * \text{derX}[i] + \text{derY}[i] * \text{derY}[i]))$$

calcul de l'orientation du gradient :

$$\text{angle}[i] = \text{atan2}((\text{double})\text{derY}[i], (\text{double})\text{derX}[i])$$

# EOH

---

- **Etape 2 : construction de l'histogramme**
  - Ceci est fait dans des cellules carrées de petite taille (de 4x4 à 12x12 pixels).
  - Chaque pixel de la cellule vote alors pour une classe de l'histogramme, en fonction de l'orientation du gradient à ce point
  - Le vote du pixel est pondéré par l'intensité du gradient en ce point

# Laplacien Histogramme

---

Donne des informations sur le contenu de la forme des objets d'une image.

Donne des informations sur la présence d'arêtes dans l'image. Même si pour certaines bases de l'information présente dans cette signature est une information supplémentaire lorsqu'on l'utilise conjointement avec d'autres signatures d'images.

# Laplacien Histogramme

---

- **Algorithme**

Calcul du Laplacien sur l'image

$$\nabla^2 I(x, y) = \frac{\partial^2 I}{\partial x^2}(x, y) + \frac{\partial^2 I}{\partial y^2}(x, y)$$

Calcul par convolution avec les masques

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

# Histogramme de Hough

---

Détection des contours : descripteur de forme

- **Algorithme**

- 1) Transformer l'image en niveau de gris
- 2) calculer les dérivées de l'image ou le gradient (derX et derY)
- 3) calculer l'orientation du gradient

Pour chaque pixel :

**$\text{angle}[i] = \text{atan2}((\text{double})\text{derY}[i]/(\text{double})\text{derX}[i]);$**

- 4) Détecter les contours : algorithme de Canny/Deriche

➤ Calculer l'amplitude de gradient :

**$\text{gradient}[i] = \text{sqrt}((\text{double})(\text{derX}[i] * \text{derX}[i] + \text{derY}[i] * \text{derY}[i]));$**



# Histogramme de Hough

---

- derY contient les maxima du gradient ; valeurs comprises entre 0 et 1
- maximum local de gradient : présence d'un contour

## *Méthode pour obtenir une image des contours :*

1. estimation du gradient en chaque point de l'image (vu précédemment)
2. extraction des maxima locaux de la norme du gradient dans la direction du gradient
3. sélection des maxima locaux significatifs par seuillage
4. fermeture des contours en traçant les chemins suivant une ligne de crête dans l'image de la norme du gradient

# Histogramme de Hough

---

- 5) Appliquer hystérésis; la différence entre le seuil haut et le seuil bas. Quand on commence en étant en bas, on change d'état quand on franchi le seuil haut. Puis on repasse à l'état bas en franchissant le seuil bas.
- 6) En utilisant un masque on calcule l'histogramme sur la carte d'orientation

# Histogramme de Hough

---

## Pseudo code

```
i=0
pour tous les pixels de l'image binarisée
| si le pixel courant appartient à un contour
| | pour tous les angles teta entre -pi/2 et 2 pi
| | | ro = x_pixel*cos(teta) + y_pixel*sin(teta)
| | | detec_rec[i].val++
| | | detec_rec[i].teta = teta
| | | detec_rec[i].ro = ro
| | i++
| fin
| fin
fin
```

# Histogramme de Hough

---

## Pseudo-code

```
pour toutes les équations i appartenant au seuil
| x1 = detec_rec[i].ro/cos(detec_rec[i].teta)
| y1 = 0
| x2 = 0
| y2 = detec_rec[i].ro/sin(detec_rec[i].teta)
| tracer_ligne_entre_x1y1_et_x2y2()
fin
```

# Difficultés

- Difficultés
  - ▶ l'étape d'extraction des contours est souvent instable
  - ▶ séparer la forme du fond est difficile (problème de segmentation)
- En général, ces algorithmes mènent à une description qui n'est pas invariante à des gros changement d'échelle, ni aux rotations (car invariance en rotation introduit alors forte perte de discriminance pour cet indice visuel)



V. Ferrari, L. Février, F. Jurie, C. Schmid, *Groups of Adjacent Contour Segments for Object Detection*, PAMI 2008

# Comparaison des descripteurs visuels

- Pour une utilisation "brute" des différents indices visuels
- Notion de descripteur associé à un indice visuel

Indice visuel	Descripteur Utilisé	Avantages	Limitations
Couleur	Histogramme, moments	Extraction très facile, propriétés globales de l'images	Pas d'information spatiale, peu de lien avec la sémantique
Forme	Polygones, approximations, invariants moments	Haut niveau d'abstraction, permet la recherche d'objets	Coût de calcul élevé, dépend d'une segmentation de bas niveau
Texture	Matrices de co-occurrences, filtre de Gabor, Fourier	Information globales et locales	Coût de calcul élevé, difficile à définir