

# Traitement d'images et Vision

Alice POREBSKI

[alice.porebski@eilco-ulco.fr](mailto:alice.porebski@eilco-ulco.fr)

Bureau R8 – Site de Longuenesse

Bureau A015 – Site de Calais

# Introduction



- Volume horaire :
  - 11 h de cours
  - 6 h de TD
  - 6 h de TD-machine
  - 15h de TP
- Modalités d'évaluation (Credits ECTS : 4 / Coeff. : 4) :
  - Examen Final
  - Comptes-rendus de TP

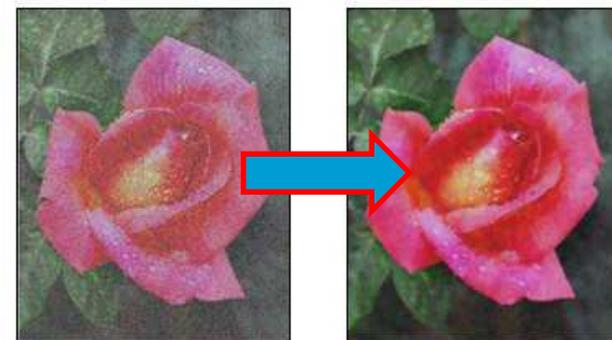
# Introduction

- Objectifs :
  - Introduction à la vision : perception visuelle humaine, présentation des éléments constitutifs de la chaîne d'acquisition, domaines et types d'applications, ...
  - Description des principaux outils de traitement d'images : prétraitement, analyse, ...
  - Illustration et applications



# Sommaire

- Approche physique et psychophysique de la perception visuelle
- Représentation numérique et dispositifs d'acquisition d'images
- Prétraitement d'images
- Les différents types de traitement d'images



# Sommaire

- Segmentation d'images



- Traitement haut-niveau :
  - > Classification d'images
  - > Recherche par le contenu
  - > Reconnaissance de formes
  - > ...



# Les différents types de traitement d'images

## Les différents types de traitement d'images



# Les différents types de traitement d'images

- Objectif du traitement : extraire l'information utile de l'image
- Traitements bas-niveau et haut-niveau :
  - Traitement bas-niveau : l'image est décrite de manière numérique (structurelle), il n'y a pas de liens avec la réalité qu'elle représente  
Exemple : segmentation
  - Traitement haut-niveau : l'image est décrite de manière symbolique, un lien existe avec la scène observée  
Exemple : classification

# Les différents types de traitements d'images

- Amélioration : modifier l'image dans le but de la rendre plus agréable à l'œil



Illustration

# Les différents types de traitements d'images

- Restauration : corriger des défauts dus à une source de dégradation



# Les différents types de traitements d'images

- Compression : réduire le volume de l'image

100% fidelity  
Image is 725kB



90%  
250kB



10%  
37kB

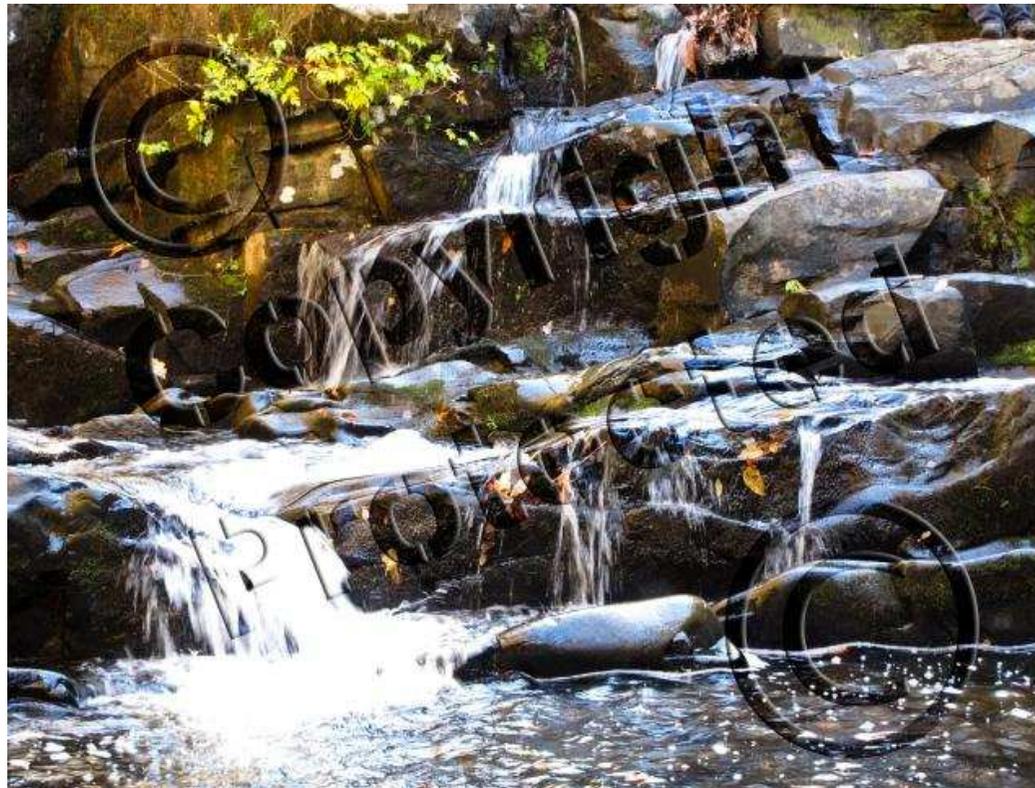


1%  
20kB



# Les différents types de traitements d'images

- Tatouage d'image : permet d'ajouter des informations, visibles ou non, dans l'image



# Les différents types de traitements d'images

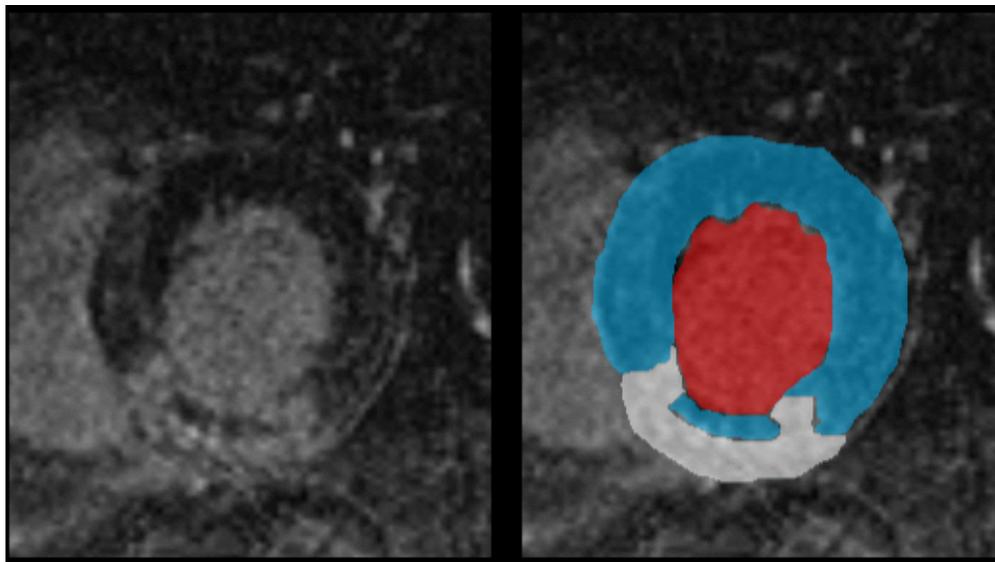
- Synthèse d'images (infographie) : modéliser par les mathématiques le contenu des images



[Vidéo](#)

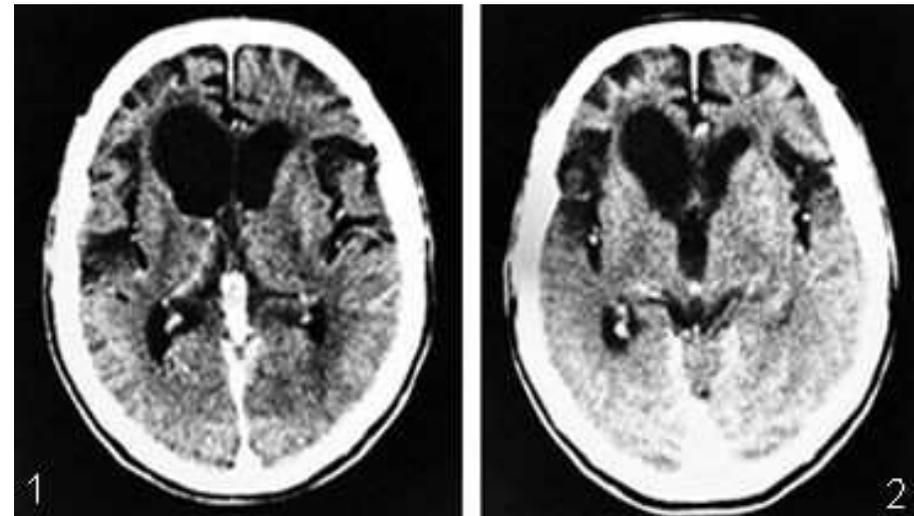
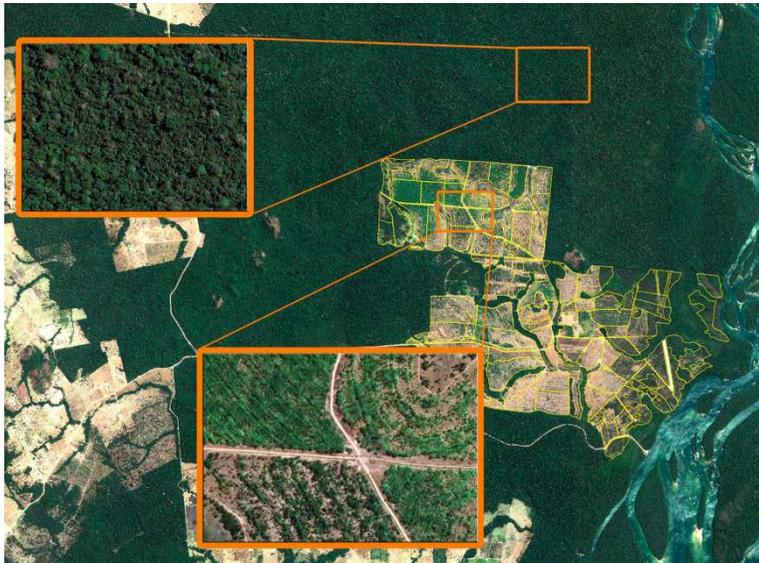
# Les différents types de traitements d'images

- Segmentation : partition de l'image



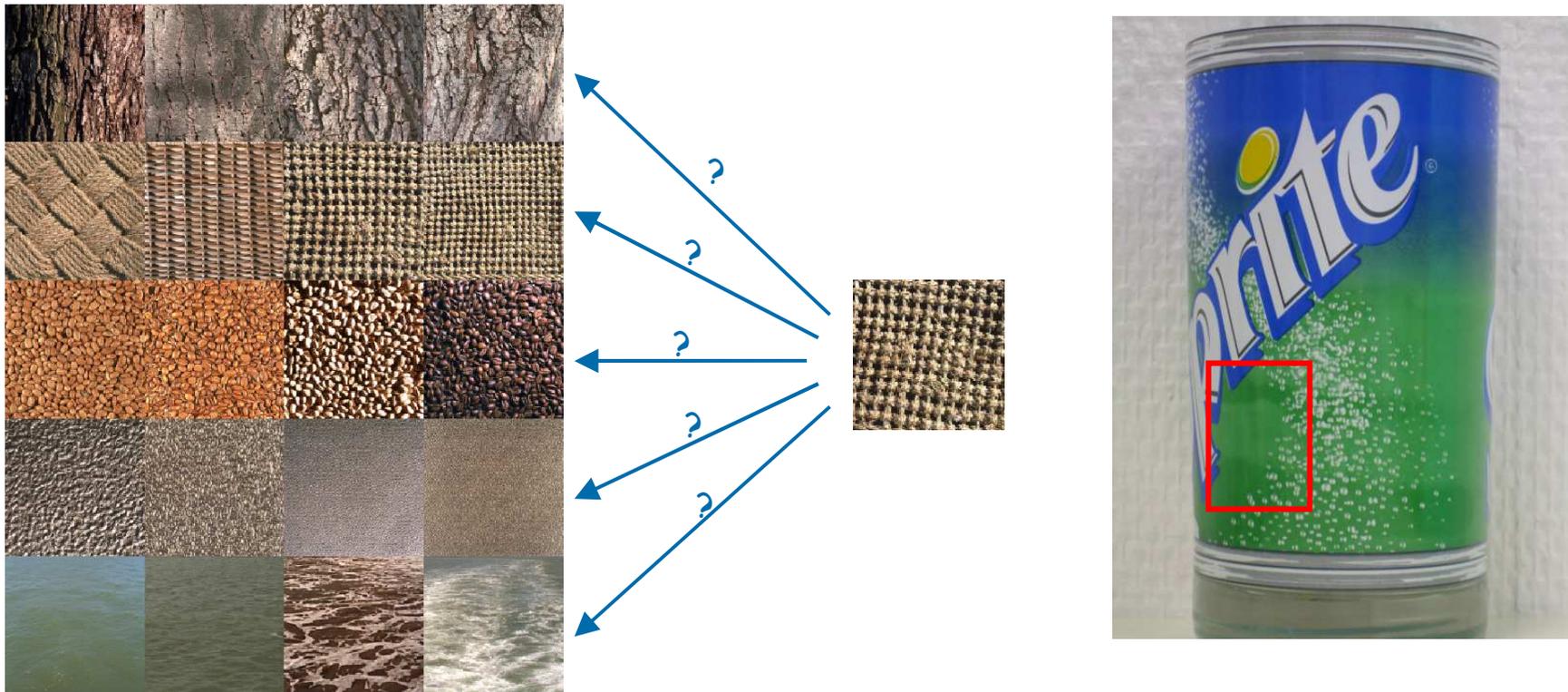
# Les différents types de traitements d'images

- Analyse : interpréter les informations et décider d'une action à engager



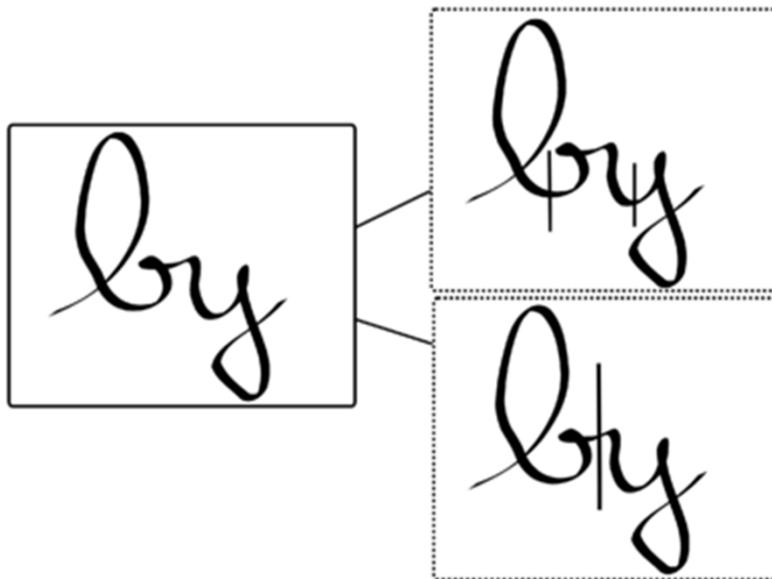
# Les différents types de traitements d'images

- Classification d'image : affectation d'une image à une classe définie



# Les différents types de traitements d'images

- Reconnaissances des formes : identification du contenu de l'image



# Les différents types de traitements d'images

- Indexation et recherche d'images : caractérisation du contenu de l'image



**Google**  
images



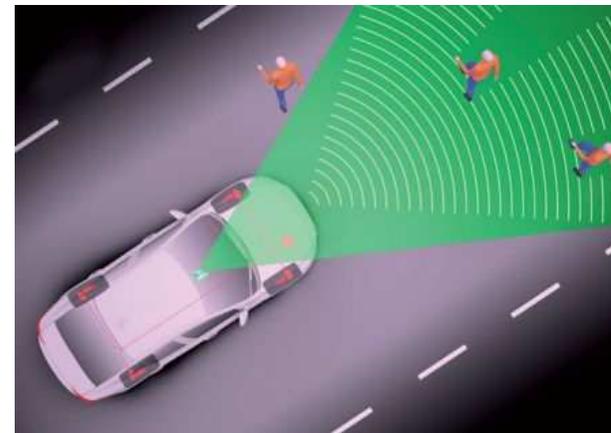
# Les différents types de traitements d'images

- Analyse du mouvement : estimation du mouvement dans les séquences d'images

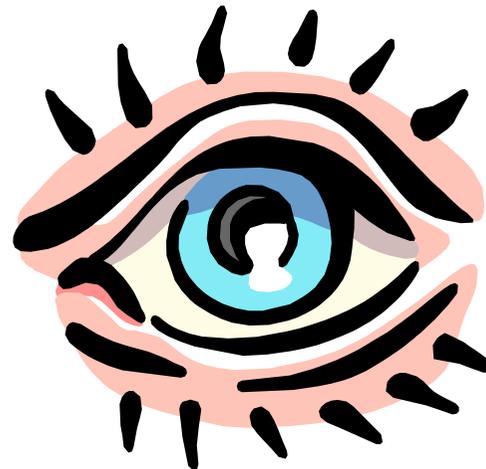


# Les différents types de traitements d'images

- Vision 3D : localisation en 3 dimensions des objets dans l'image



## Approche physique et psychophysique de la perception visuelle

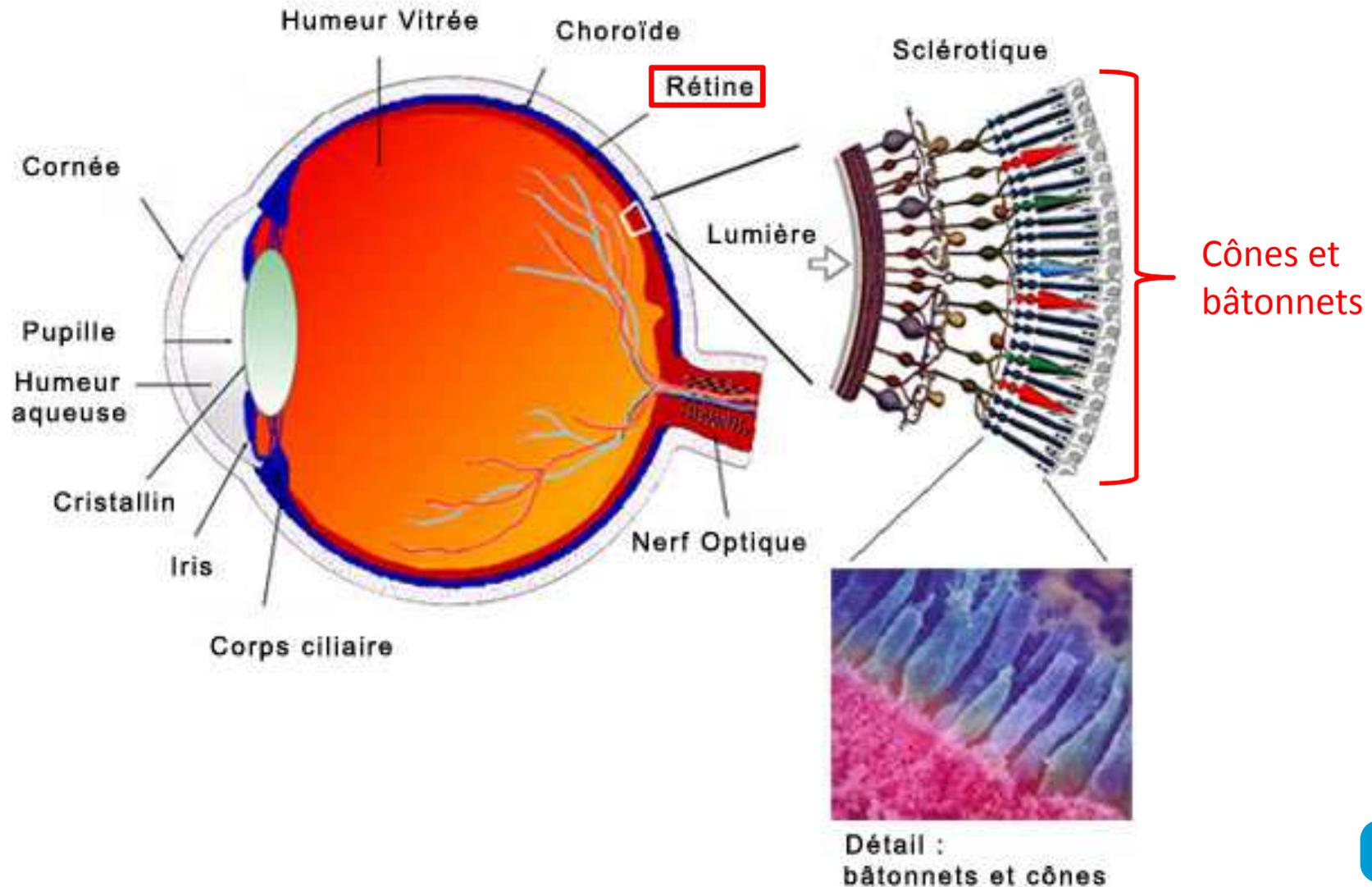


# Perception visuelle

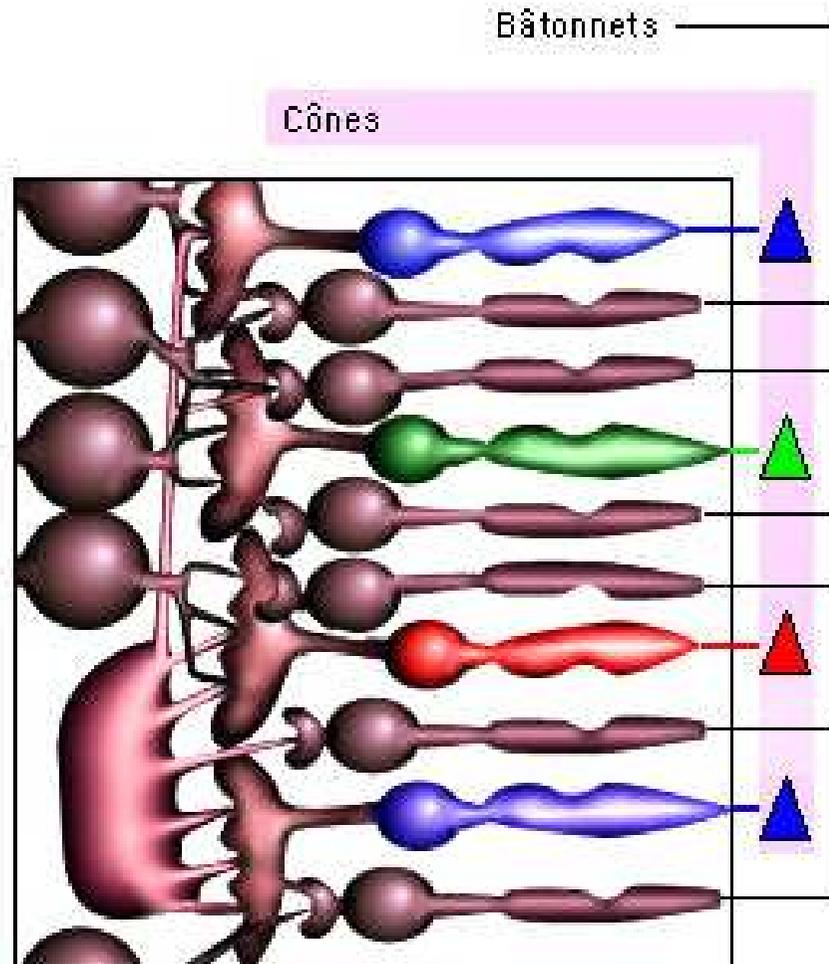
Vidéo  
introductive



# La vision humaine -> approche physique

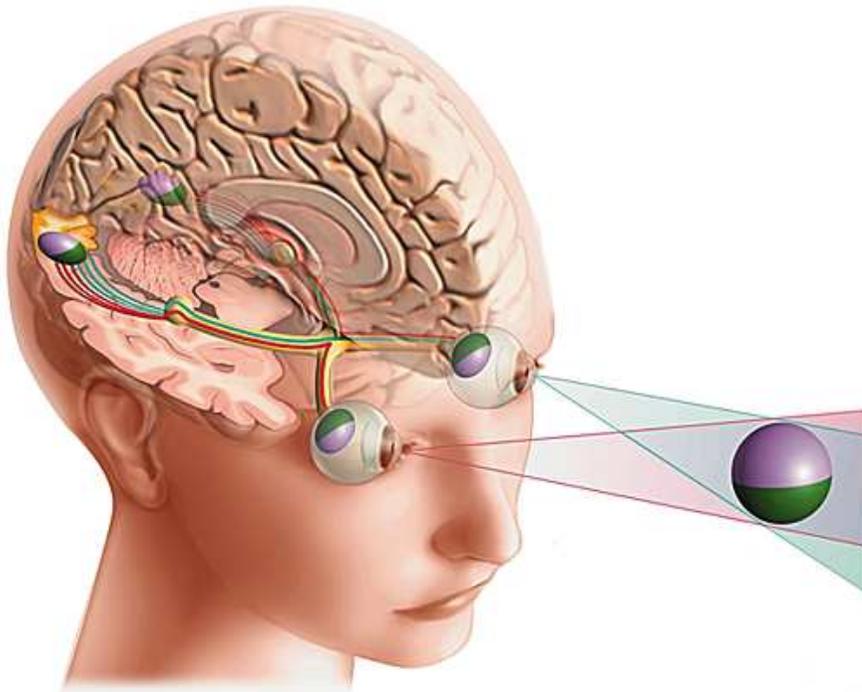


# La vision humaine -> approche physique



- Les **cônes** perçoivent les couleurs. Ils sont spécialisés : une sorte réagit au rouge, une autre au vert et une autre au bleu
- Les **bâtonnets** sont associés à la perception de l'intensité de la lumière

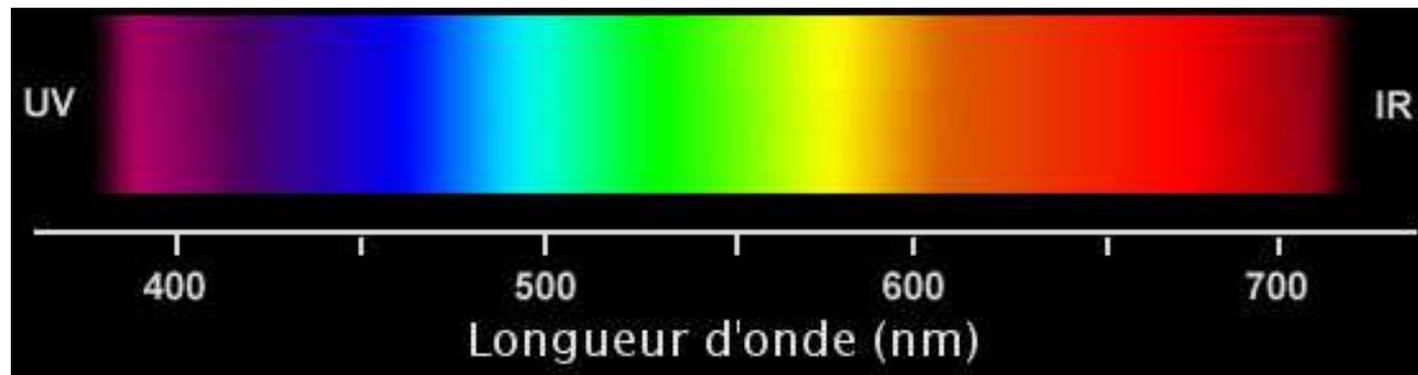
# La vision humaine -> approche physique



- Les **cônes** et les **bâtonnets** transforment la lumière ayant traversé le cristallin en impulsions électriques
- Ces impulsions sont conduites jusqu'au cerveau le long du nerf optique
- Le cerveau reconstruit alors l'image, point par point, couleur par couleur

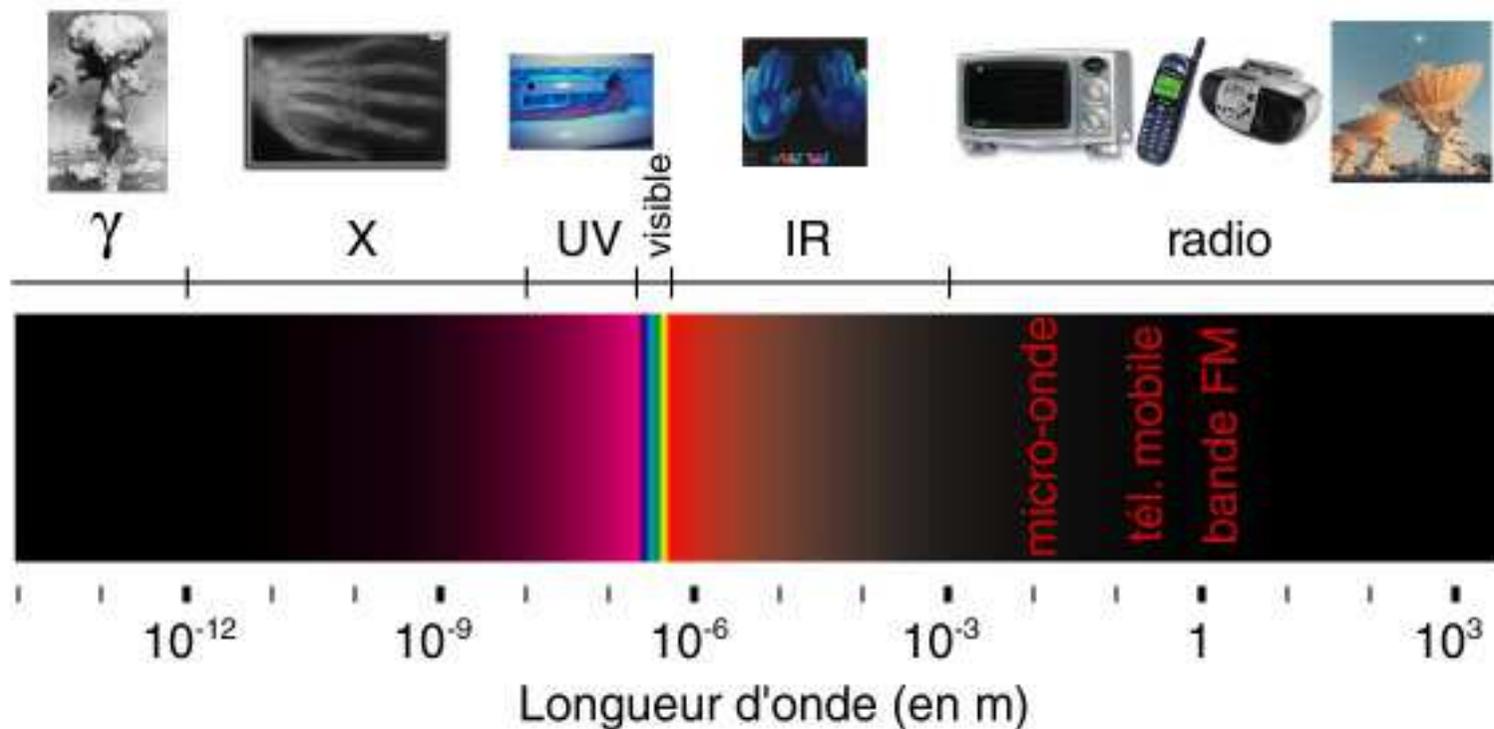
# La lumière et la couleur

- A chaque couleur perçue par l'œil humain correspond une longueur d'onde
- La longueur d'onde  $\lambda$  se mesure en mètre (m)
- $\lambda = c / F$  où  $F$  représente la fréquence exprimée en Hertz (Hz) et  $c$ , la vitesse de la lumière ( $c = 300\,000\text{ Km/s}$ )



# La lumière et la couleur

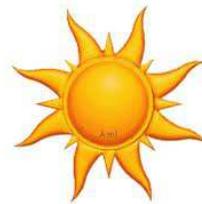
- La lumière visible correspond aux ondes visibles comprises entre 375 nm et 740 nm



# La lumière et la couleur

- La lumière peut être :

- naturelle
- artificielle



- primaire si elle produit un rayonnement électromagnétique

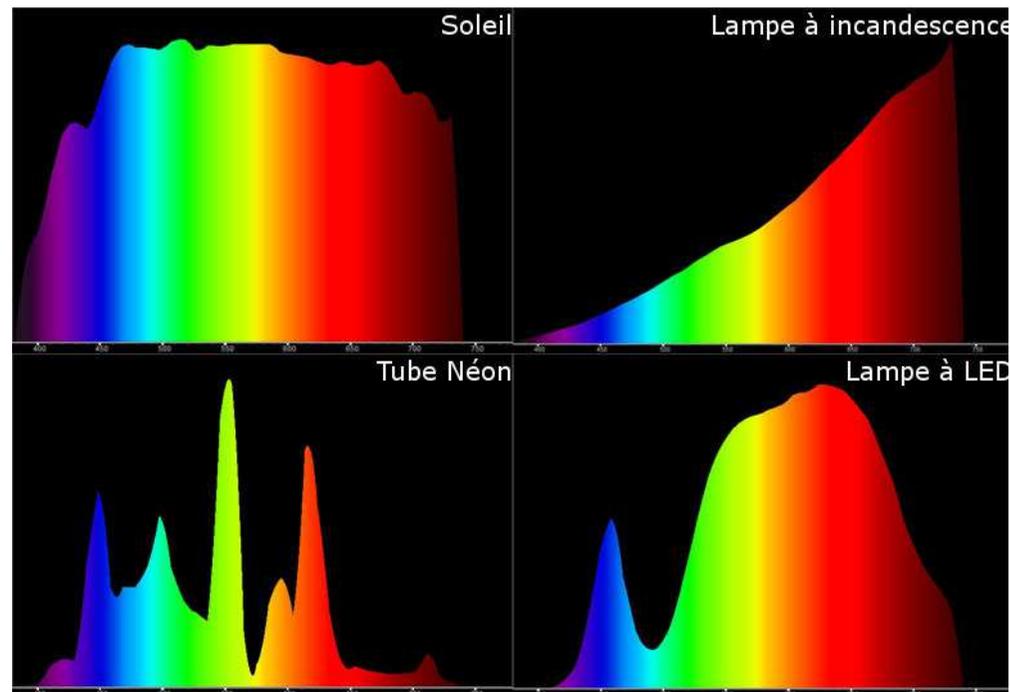


- secondaire si elle émet des rayons lumineux issus de la réflexion ou de la transmission par un matériau

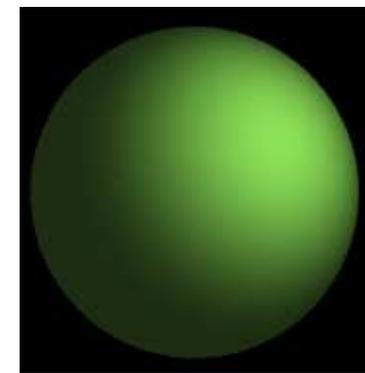
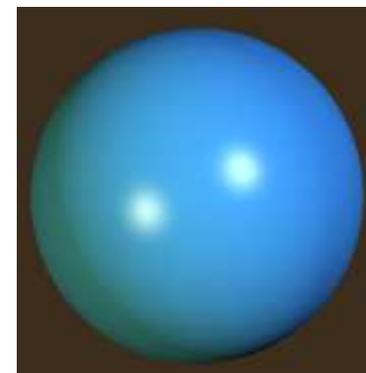
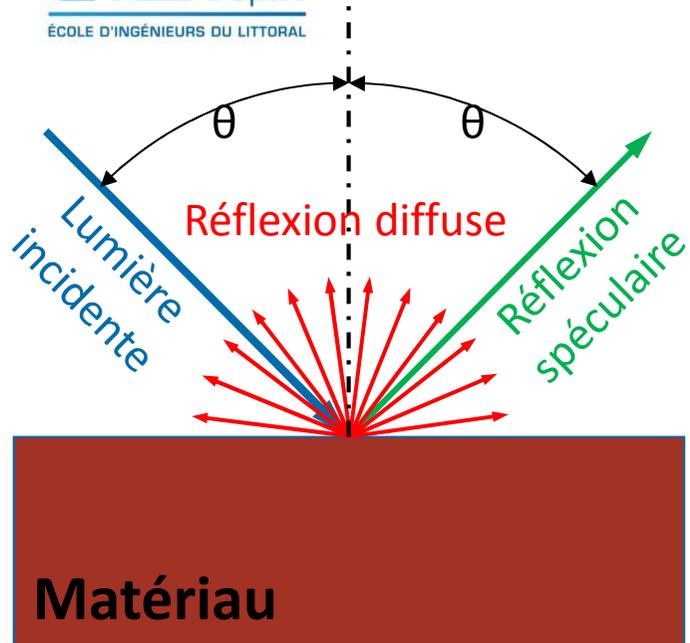


# La lumière et la couleur

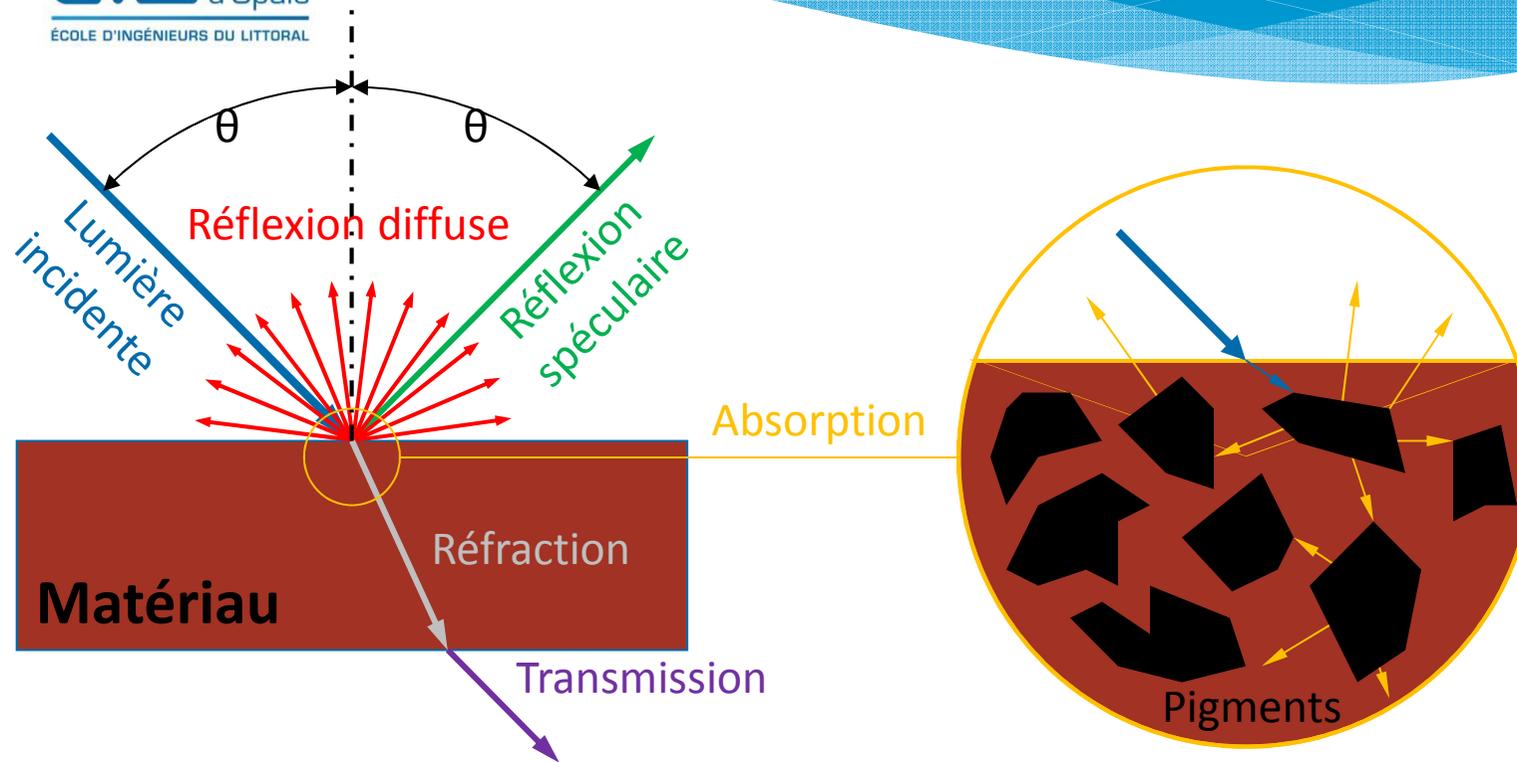
- Une source lumineuse est caractérisée par un **spectre**
- Le spectre indique les longueurs d'onde dans lesquelles la source émet



# Interaction lumière-objet



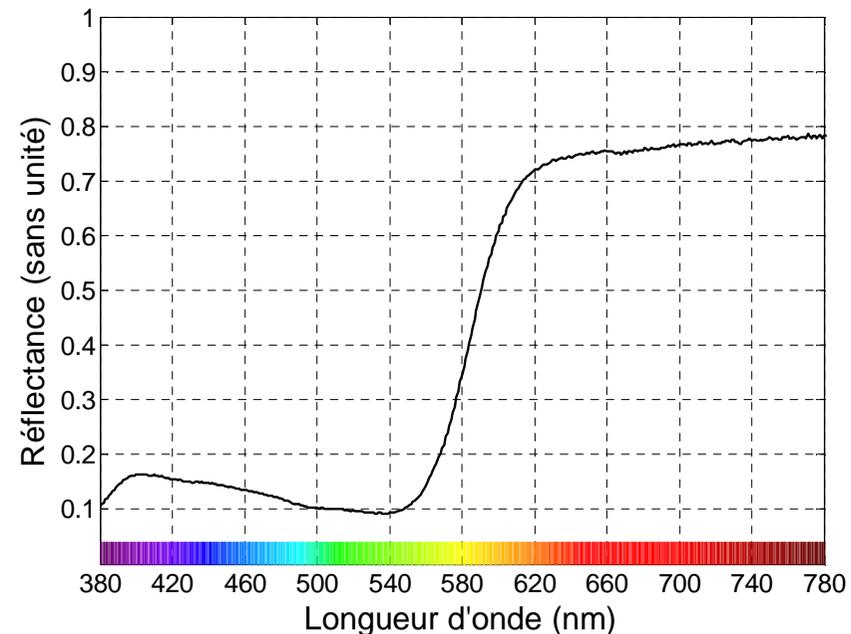
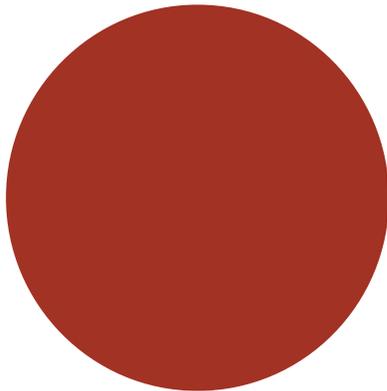
# Interaction lumière-objet



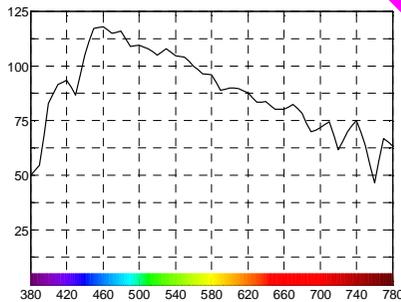
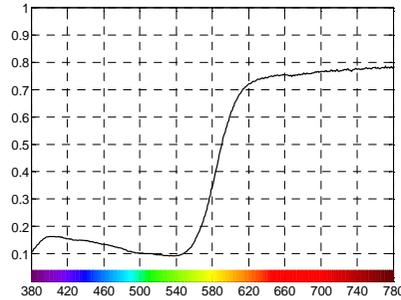
- Les pigments absorbent une partie de la lumière incidente et réfléchissent ou transmettent le reste
- Le matériau agit alors comme un filtre

# Interaction lumière-objet

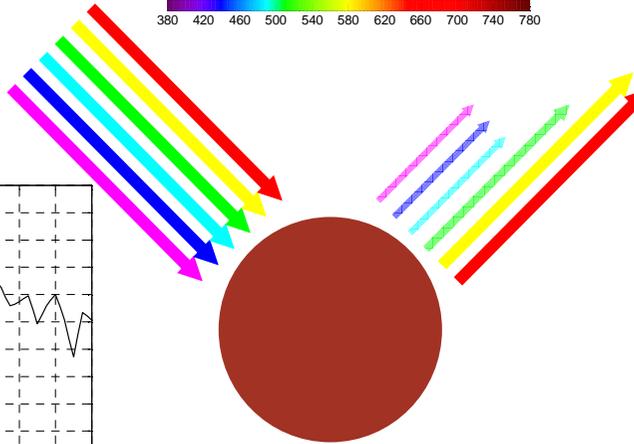
- Selon sa nature (opaque, transparent, translucide), un matériau peut être caractérisé, soit par sa capacité à réfléchir (**réflectance**) ou à transmettre (**transmittance**) l'énergie incidente, soit par sa capacité à l'absorber



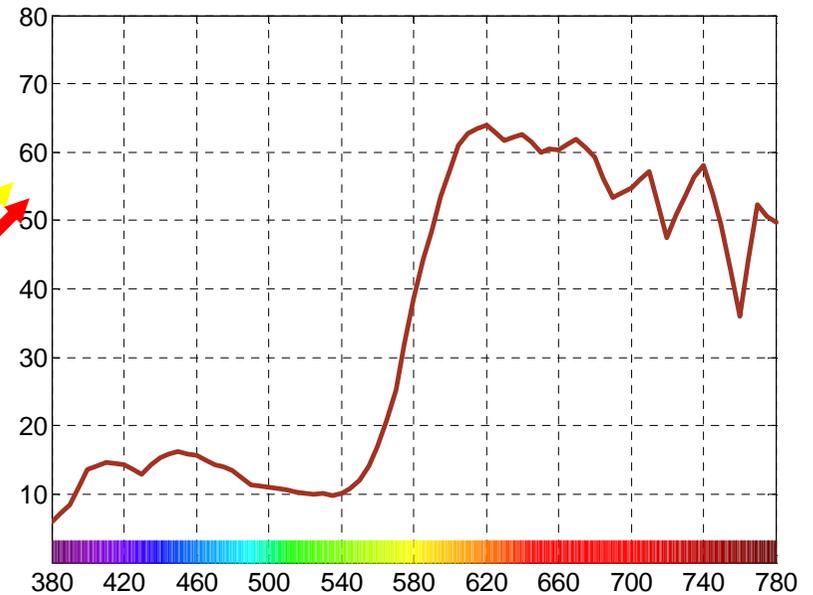
# Interaction lumière-objet



Source lumineuse



Objet



Stimulus de couleur

# Interaction lumière-objet

- Le stimulus de couleur  $S(\lambda)$  issu d'un objet résulte du produit du spectre d'une source  $I(\lambda)$  avec le spectre de réflectance de l'objet  $R(\lambda)$  :

$$S(\lambda) = I(\lambda) \times R(\lambda)$$

- Les conditions d'éclairage et d'observation modifient les caractéristiques d'un stimulus de couleur

# Influence des conditions d'éclairage



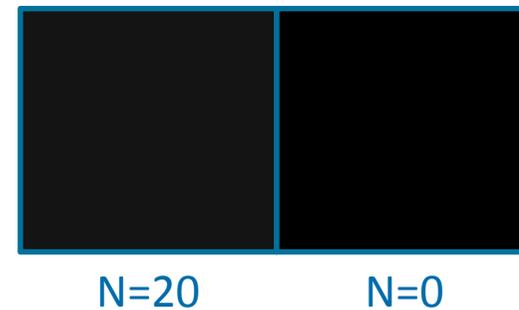
- Image issue de la base OuTex acquise avec 3 éclairages différents :
  - un éclairage du type « soleil à l'horizon »
  - une lampe à incandescence
  - une lampe fluorescente

# La vision humaine -> approche psychophysique

- Le système visuel humain ne discrimine pas avec la même efficacité toutes les couleurs. Trois phénomènes sont observés :
  - Non-uniformité de la perception aux écarts de luminance
  - Non-uniformité de la perception aux écarts de teinte
  - Métamérisme

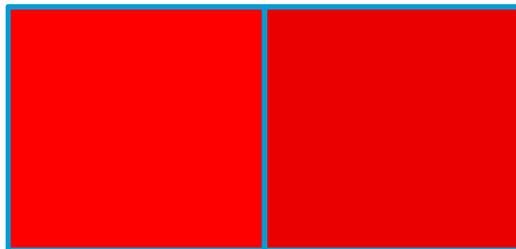
# La vision humaine -> approche psychophysique

- Non-uniformité de la perception aux écarts de luminance :
  - Expérience menée par Weber en 1834
  - Différence de sensibilité de la vision humaine aux contrastes de luminance selon le niveau de luminance (faible ou élevé)



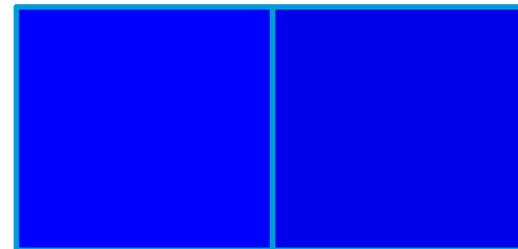
# La vision humaine -> approche psychophysique

- Non-uniformité de la perception aux écarts de teinte :
  - Expérience menée par Wright et Pitt en 1934
  - Le seuil de différentiation est différent selon la teinte



N=255

N=235

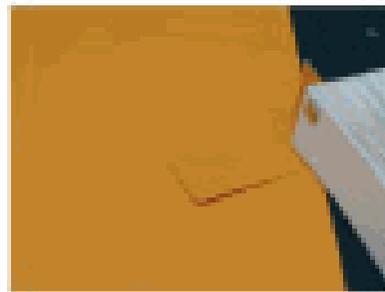


N=255

N=235

# La vision humaine -> approche psychophysique

- Métamérisme :
  - Deux couleurs absolument identiques sous un éclairage seront vues différentes sous un autre type d'éclairage

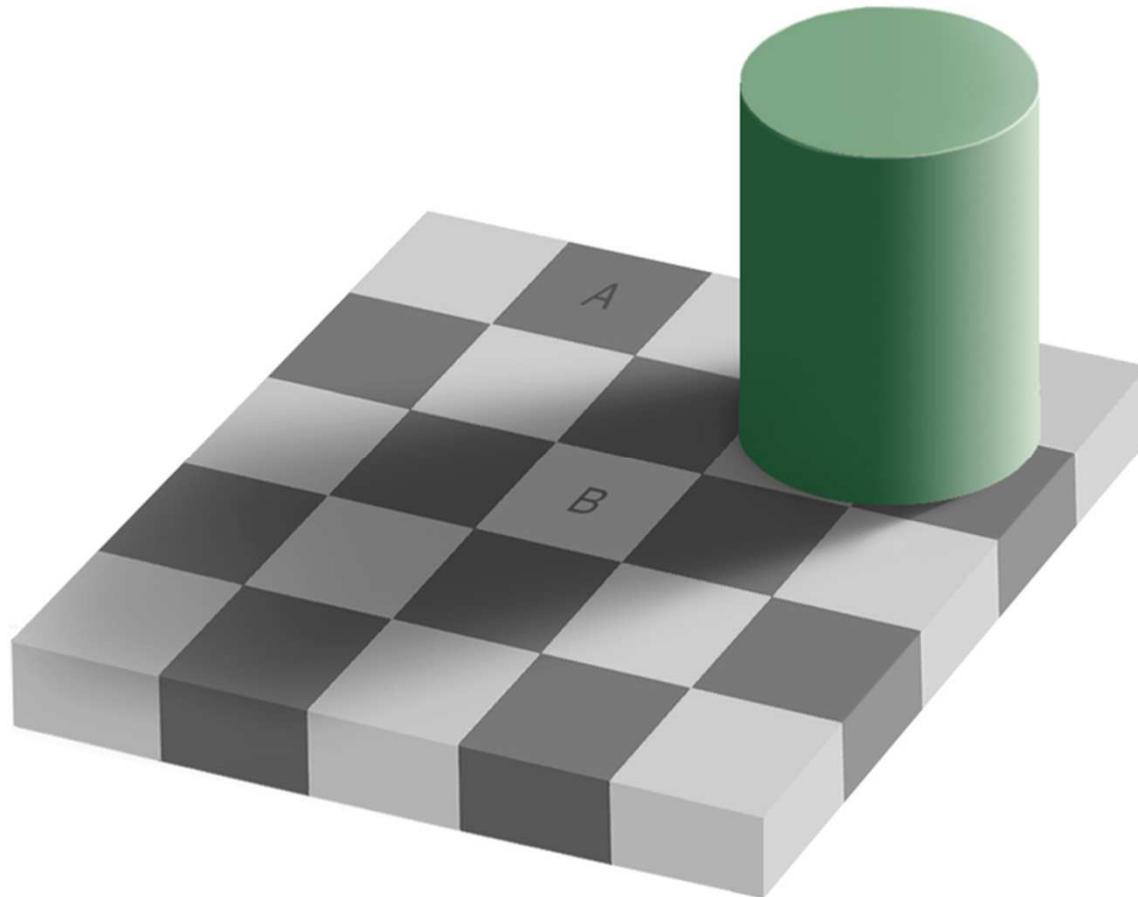


Lumière du jour

Lumière artificielle

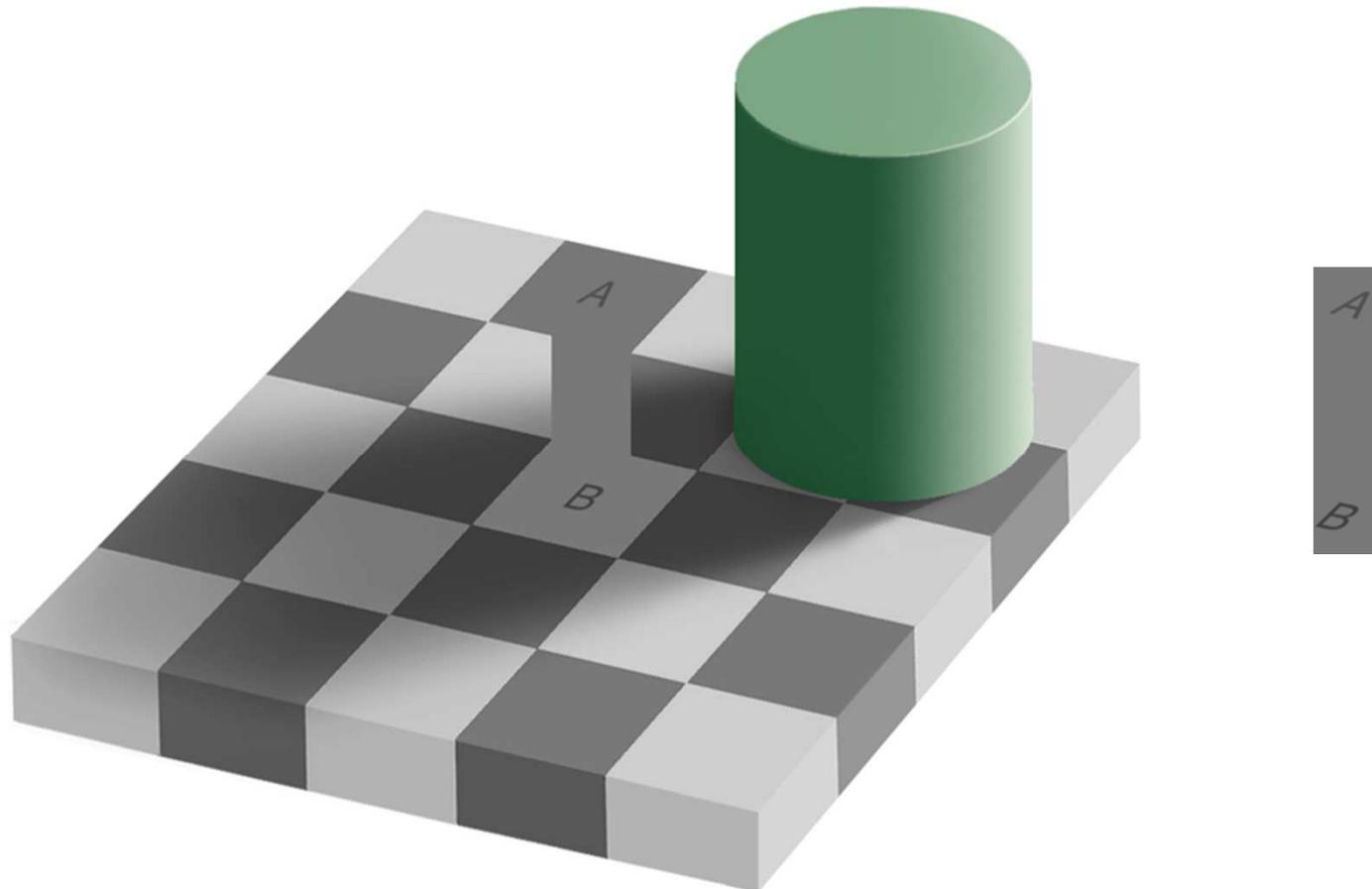
# La vision humaine -> approche psychophysique

- Perception et interprétation



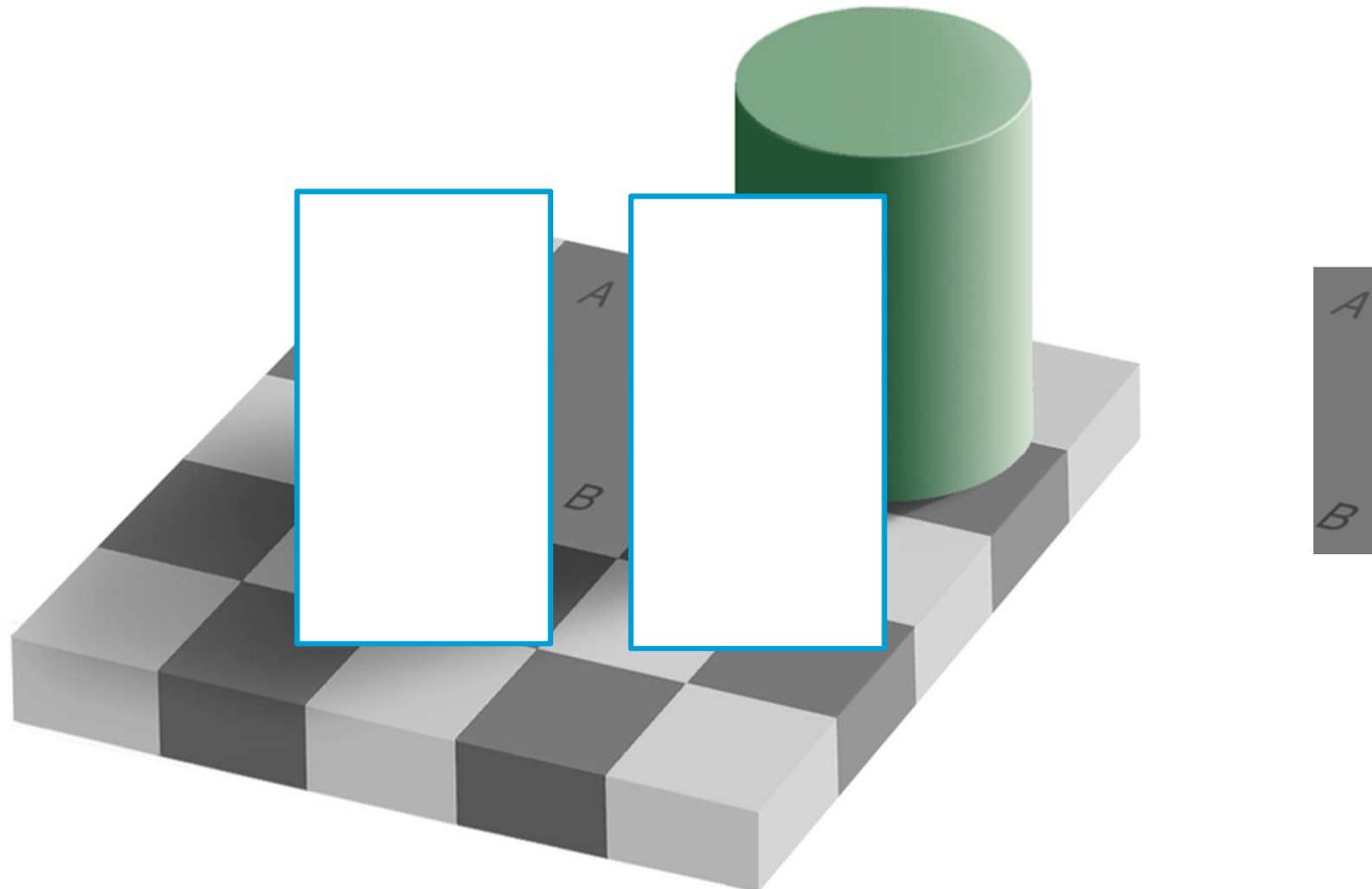
# La vision humaine -> approche psychophysique

- Perception et interprétation



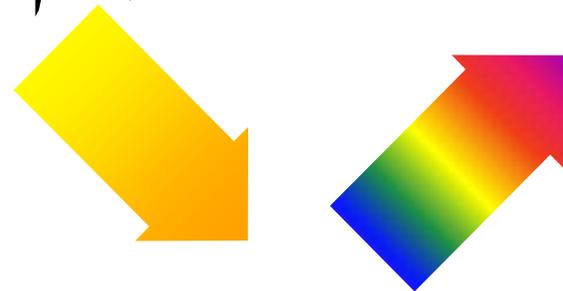
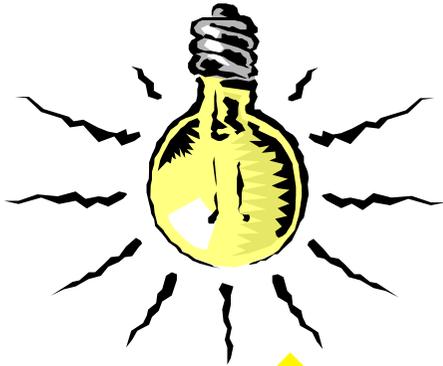
# La vision humaine -> approche psychophysique

- Perception et interprétation

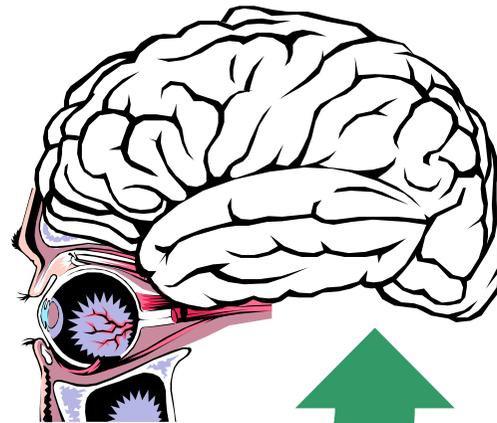


# De la vision humaine à la vision par ordinateur

Éclairage



Objet



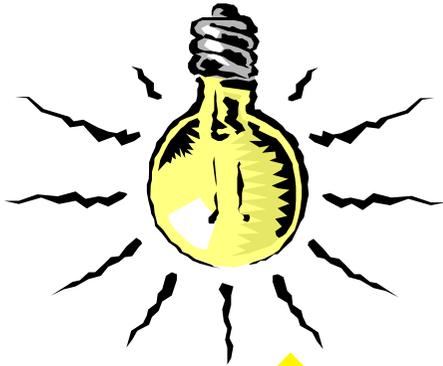
Cerveau

Oeil

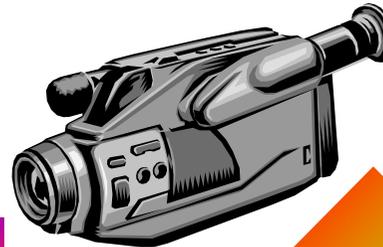


# De la vision humaine à la vision par ordinateur

Éclairage



Objet



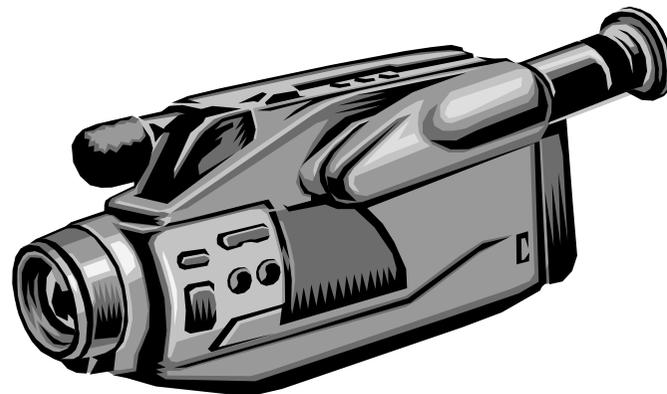
Caméra  
+ optique

Unité de  
traitement

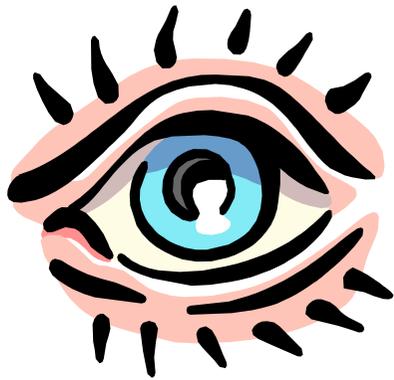


Décision  
Affichage

## Représentation numérique et dispositifs d'acquisition d'images



# Vision par ordinateur : pourquoi ?

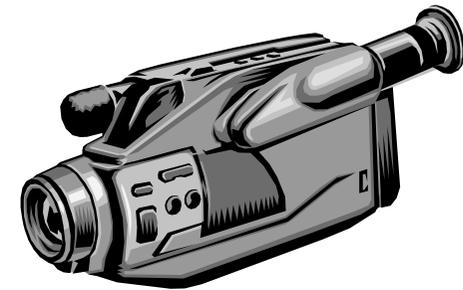


## Vision humaine

Fastidieux, fatigant,  
peu rapide, peu  
robuste, peu précis,  
dangereux, impossible

Pourquoi  
?

Remplacer l'observateur  
humain par une machine  
ayant la capacité de voir  
afin d'**automatiser** les  
tâches de contrôle



## Vision industrielle

Performance, fiabilité,  
qualité, traçabilité,  
sécurité

# Vision par ordinateur : qui ?



Industrie  
Contrôle qualité



Verre sans défaut



Verre avec un défaut  
de tâché



Verre avec un défaut  
de nuance

# Vision par ordinateur : qui ?



## Industrie

Contrôle qualité

Tri automatique

Pilotage de robots

...

Qui  
?



# Vision par ordinateur : qui ?

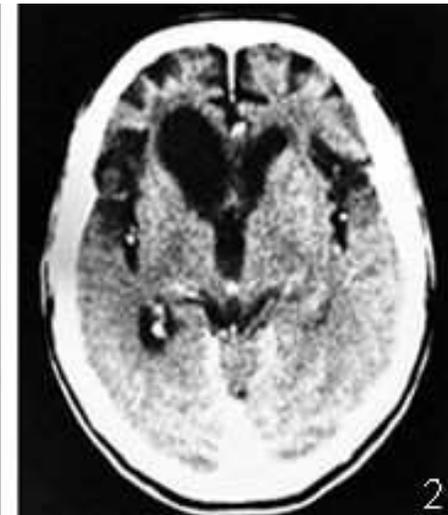


## Médecine

Aide au diagnostic

Chirurgie assistée par ordinateur

Suivi médical



# Vision par ordinateur : qui ?



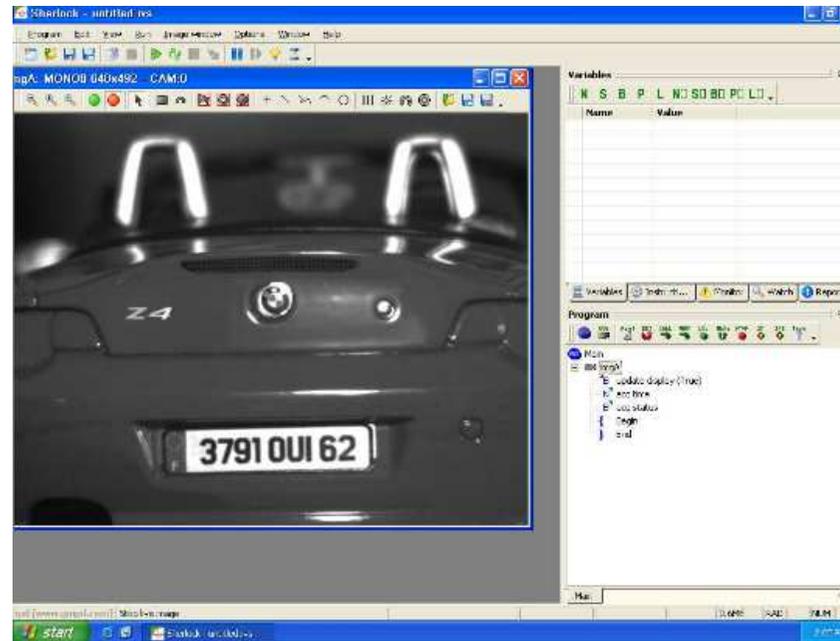
## Transport

Radar automatique

Suivi de véhicules

Détection d'obstacles

...



# Vision par ordinateur : qui ?

Qui  
?

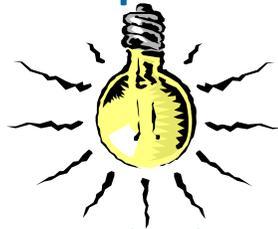
Et aussi multimédia, militaire,  
spatial, environnement,  
communication, culture,  
sécurité, sport, ...



# Vision par ordinateur : comment ?

- Un dispositif acquisition est composé de :

- Eclairage



- Caméra, associée à une optique



- Unité de traitement contenant une carte d'acquisition

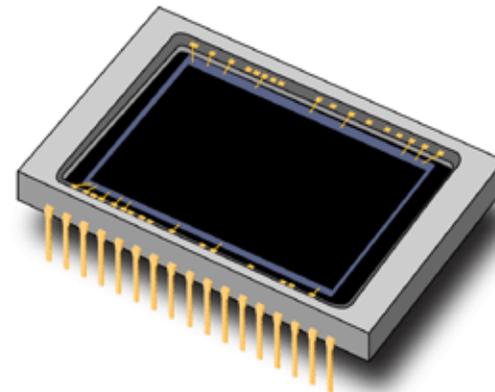


- Ecran permettant de visualiser le résultat, associé à une carte vidéo

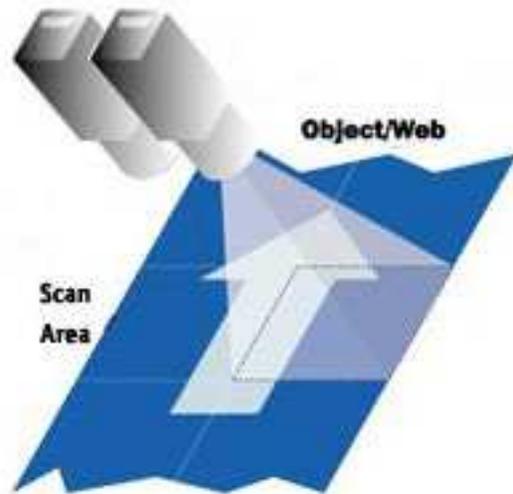


# Vision par ordinateur : comment ?

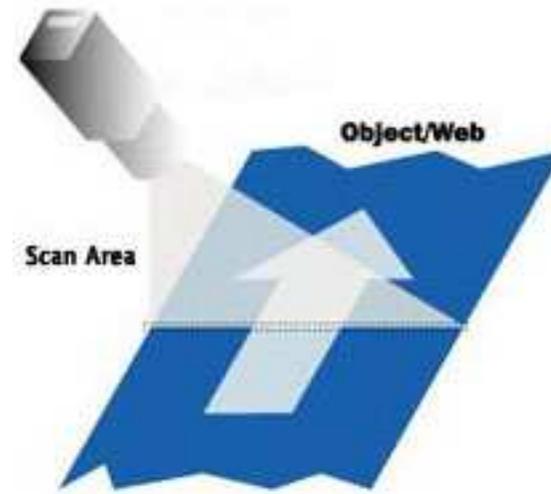
- Définitions :
  - Le **dispositif d'acquisition** capture une information lumineuse et la convertit en signal électrique analogique
  - La conversion en signal électrique est réalisé grâce aux **photodétecteurs**
  - L'acquisition d'une image est réalisée grâce à une suite de **photoéléments** organisés en barette (caméra linéaire) ou en matrice (caméra matricielle)



# Caméras matricielles et linéaires

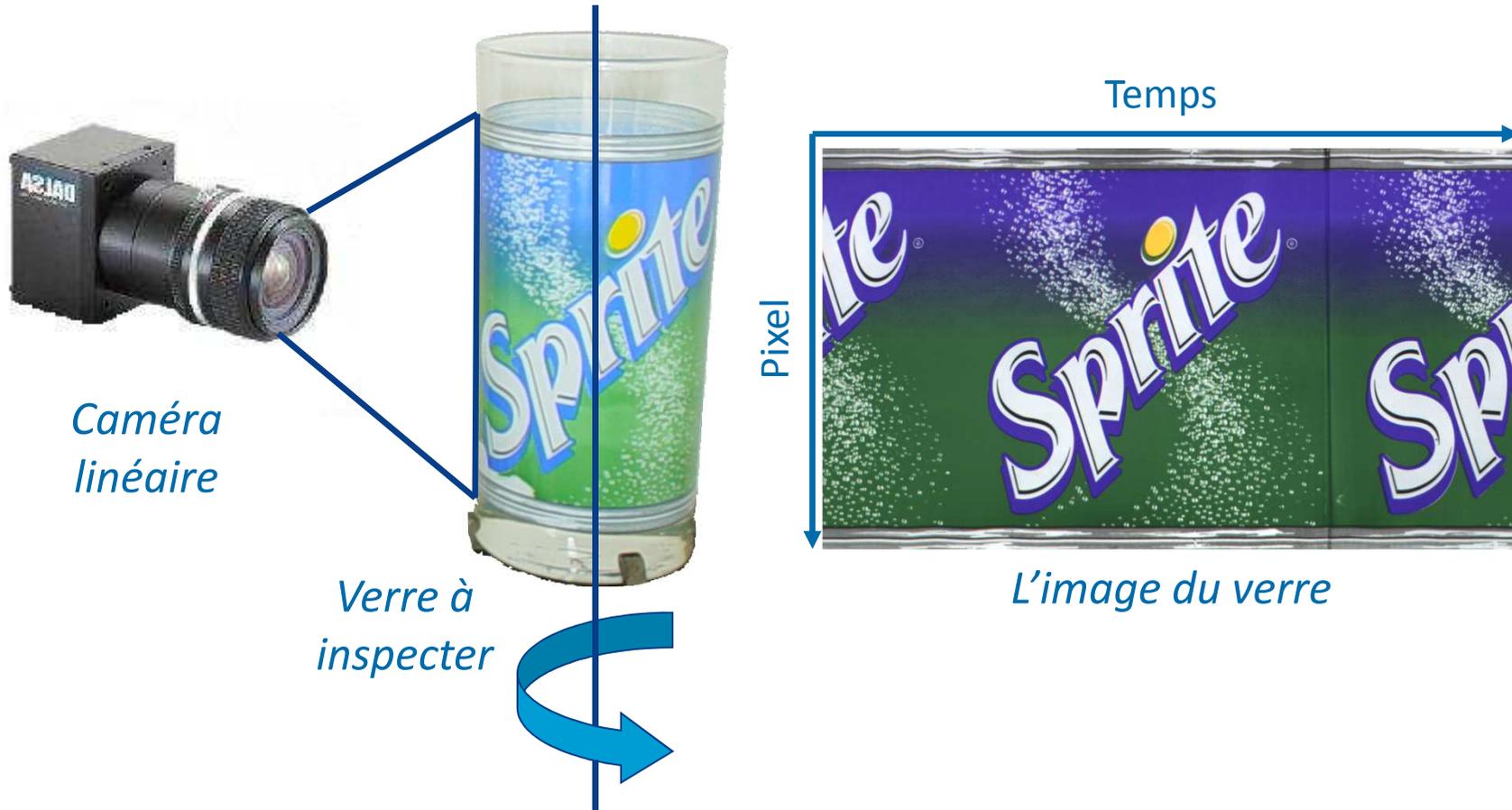


Principe de la caméra matricielle



Principe de la caméra linéaire

# Caméras matricielles et linéaires

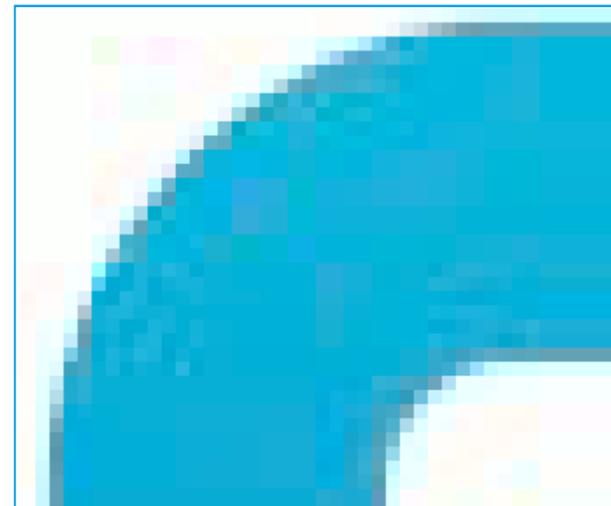
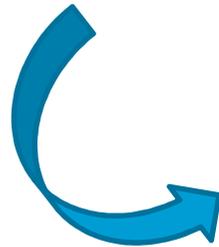


# Vision par ordinateur : comment ?

**EiL** Côte  
d'Opale  
ÉCOLE D'INGÉNIEURS DU LITTORAL

**EiL** Côte  
d'Opale  
ÉCOLE D'INGÉNIEURS DU LITTORAL

L'image ainsi obtenue est constituée d'un ensemble de pixels qui correspondent aux photoéléments

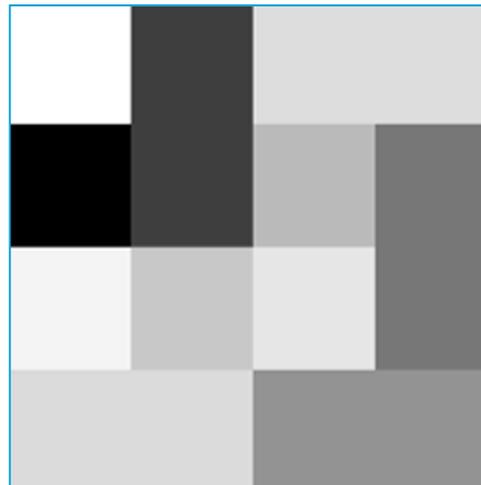


# Vision par ordinateur : comment ?

- Remarques :
  - La résolution de la caméra dépend du nombre de photoéléments constituant le capteur
  - La technologie la plus répandue pour la réalisation des capteurs d'images est actuellement la technologie CCD (Charged Coupled Device) dont les photorécepteurs produisent une tension croissante en fonction de l'intensité lumineuse qu'ils reçoivent

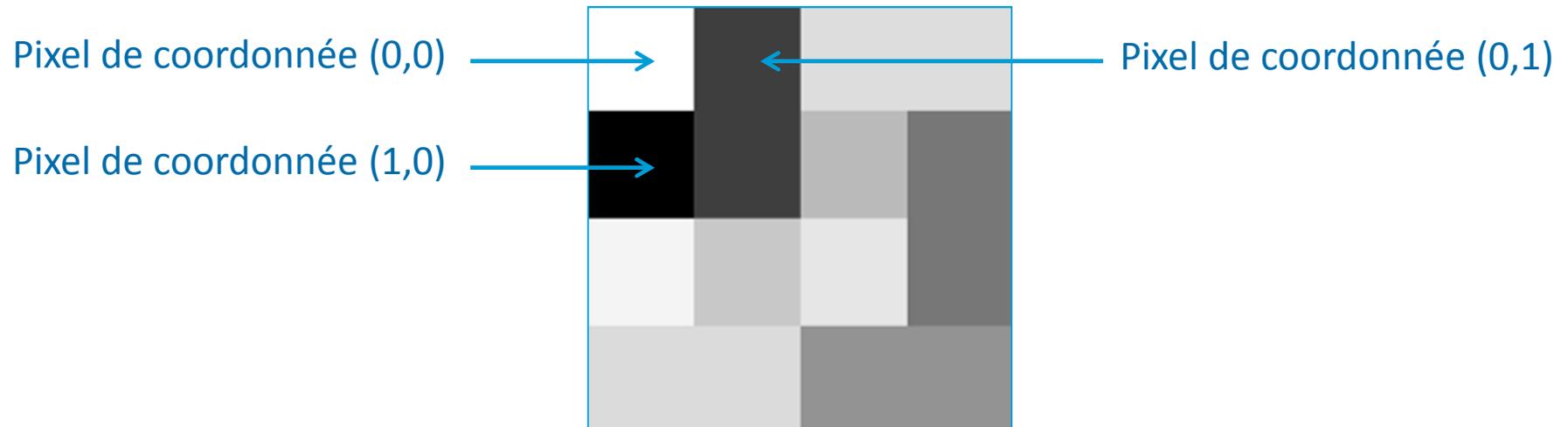
# Représentation numérique d'une image

- Définitions :
  - Une **image numérique** est une matrice de pixels
  - Le **pixel** est l'unité de base permettant de mesurer la définition d'une image. Son nom provient de l'anglais « **picture element** » qui signifie « élément d'image »



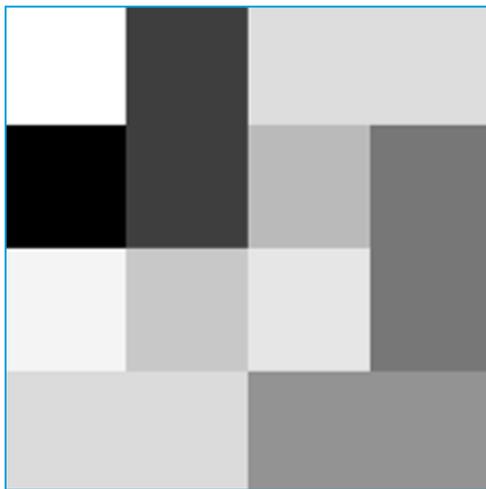
# Représentation numérique d'une image

- Définition :
  - Chaque pixel est localisé par 2 coordonnées  $x$  et  $y$  dans le repère image



# Représentation numérique -> niveaux de gris

- Représentation numérique d'une image :
  - Chaque pixel est associé à un niveau de gris (ou de couleur) généralement codé sur 8 bits (de 0 à 255)
  - Image en niveau de gris



255	83	205	205
0	83	147	88
232	179	209	88
200	200	111	111

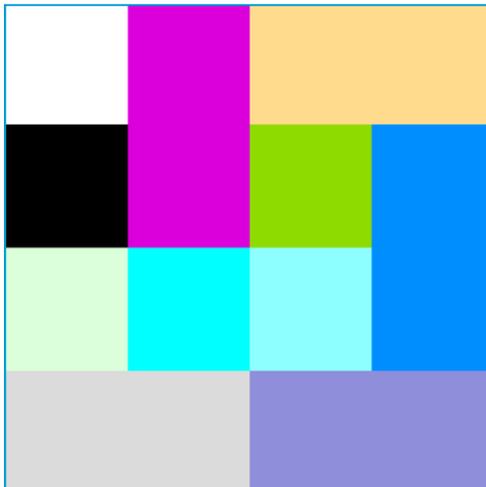
# Représentation numérique -> niveaux de gris

- Représentation numérique d'une image :
  - Image en niveau de gris
    - Niveau = 0 -> Noir
    - Niveau = 255 -> Blanc
  - Analyse des images en niveaux de gris :
    - Rapide
    - Parfois insuffisante -> utilisation de la couleur



# Représentation numérique -> couleur (R,G,B)

- Représentation numérique d'une image :
  - Image couleur (R,G,B)

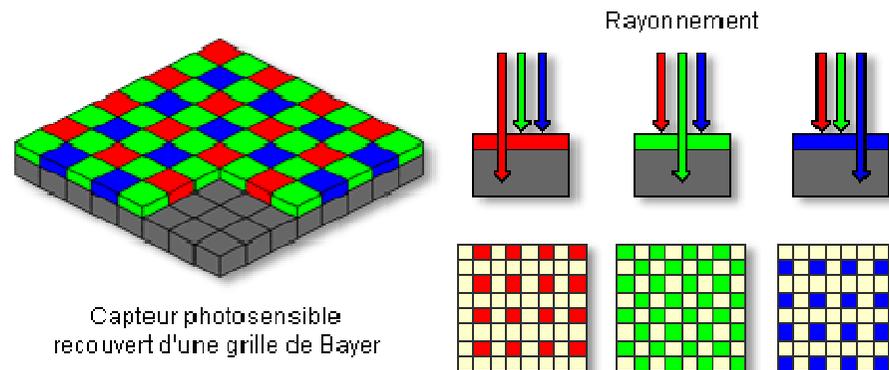


255	200	255	255
255	0	200	200
255	200	100	100
0	200	100	0
0	0	200	100
0	200	0	255
200	0	100	0
255	255	255	100
200	255	255	255
200	200	100	100
200	200	100	100
200	200	200	200



# Acquisition d'une image couleur

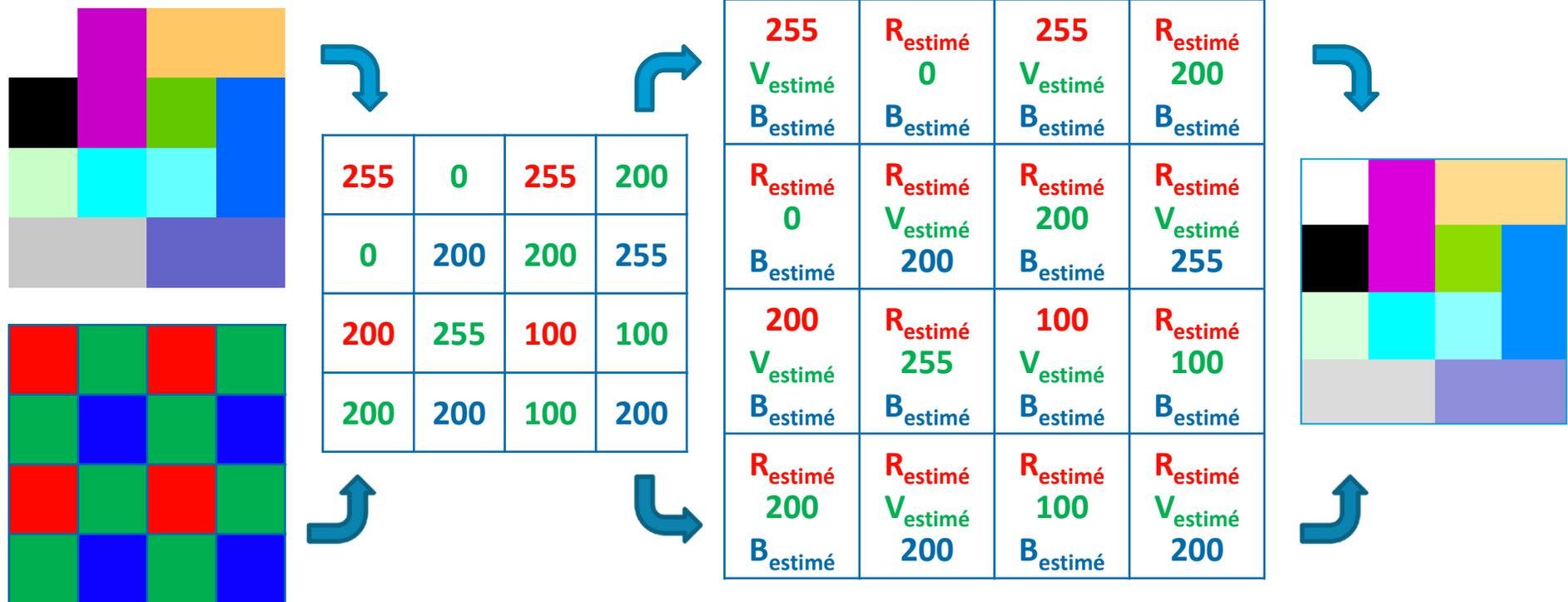
- Il existe essentiellement deux types de caméras couleur CCD :
- Les caméras mono-CCD : elles sont équipées d'un seul capteur CCD et de trois filtres colorés (rouge, vert et bleu) entrelacés



- Les caméras tri-CCD : elles sont équipées de trois capteurs CCD montés sur un système optique à base de prismes. Chacun des trois capteurs reçoit respectivement les composantes rouge, vert et bleu de l'image

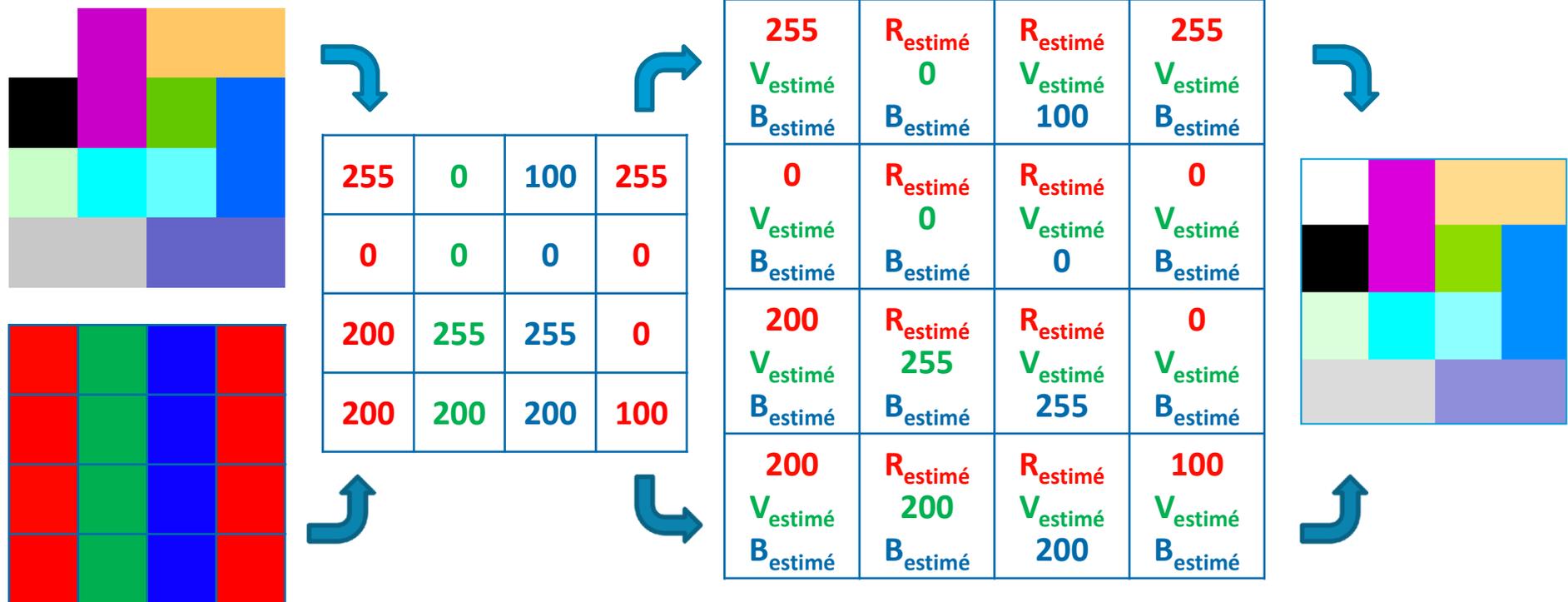
# Acquisition d'une image couleur

- Il existe essentiellement deux types de caméras couleur CCD :
  - Acquisition par caméras mono-CCD : exemple 1



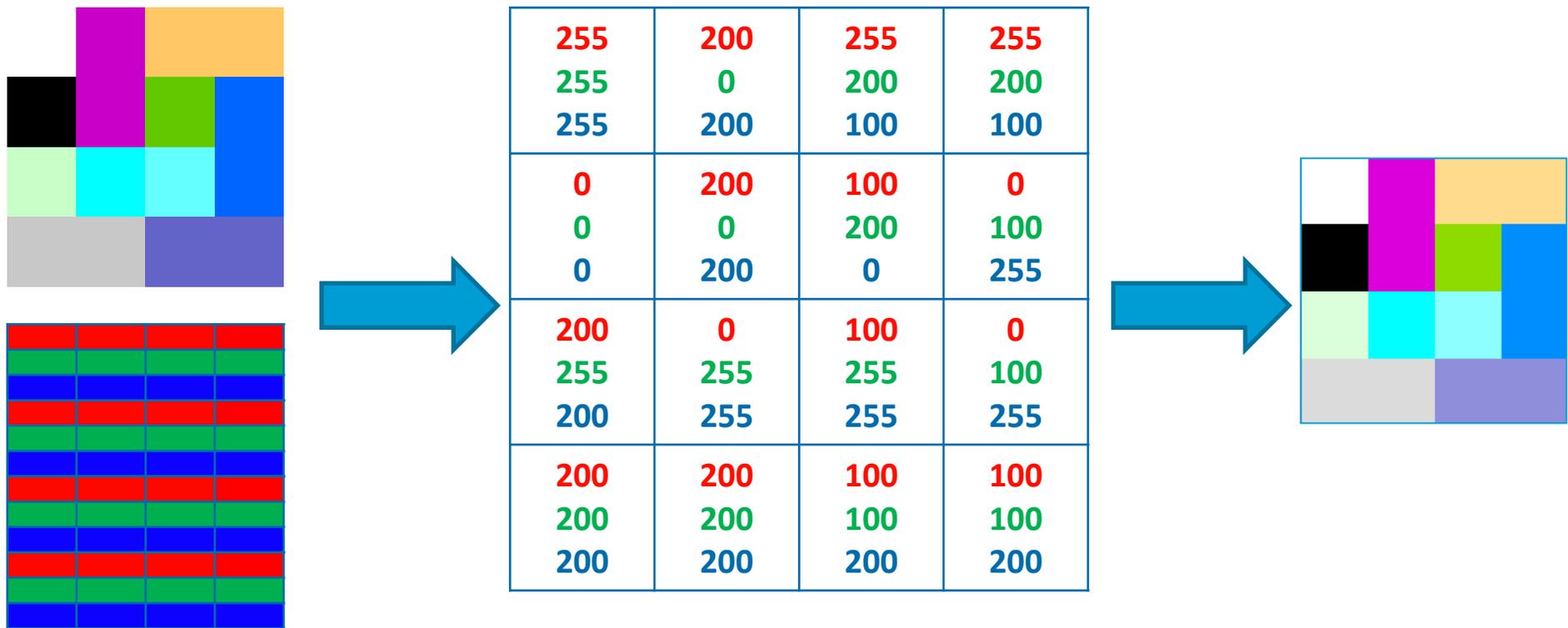
# Acquisition d'une image couleur

- Il existe essentiellement deux types de caméras couleur CCD :
  - Acquisition par caméras mono-CCD : exemple 2



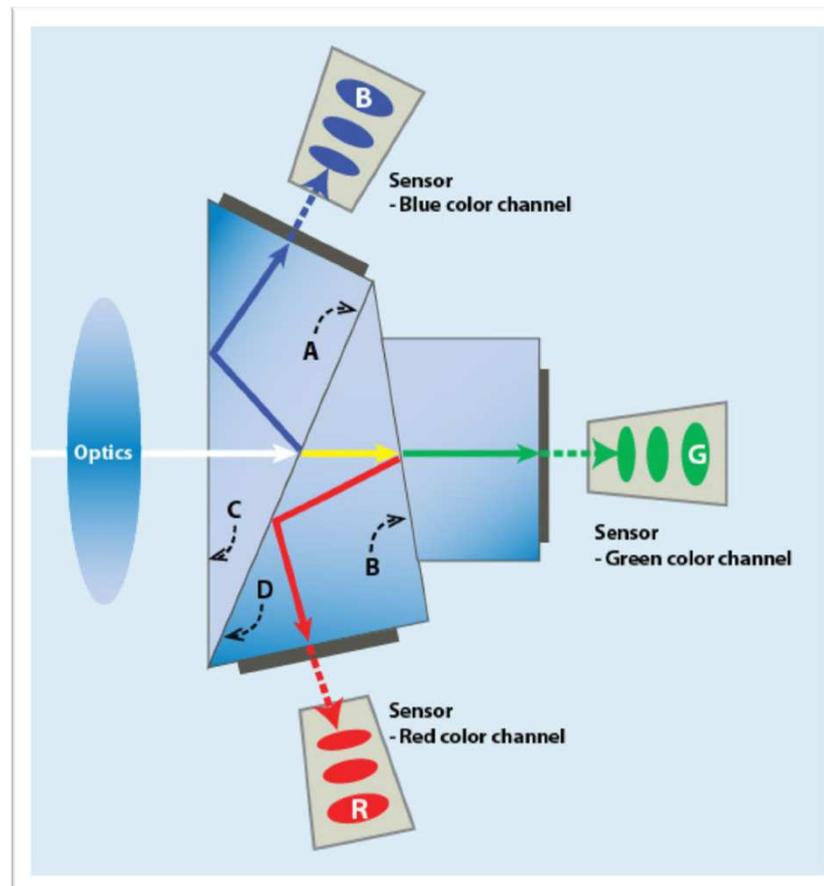
# Acquisition d'une image couleur

- Il existe essentiellement deux types de caméras couleur CCD :
  - Acquisition par caméras tri-CCD :



# Acquisition d'une image couleur

- Il existe essentiellement deux types de caméras couleur CCD :
  - Acquisition par caméras tri-CCD :



# Représentation numérique -> couleur (R,G,B)

- Représentation numérique d'une image :
  - Image couleur (R,G,B)

0
0
0

-> Noir

255
255
255

->



255
0
0

-> Rouge

0
255
0

-> Vert

0
0
255

-> Bleu

255
255
0

-> Jaune

255
128
0

-> Orange

128
0
255

-> Violet

# Représentation numérique

- Question : comment convertit-on une image couleur en une image en niveau de gris ?

$$NdG = \frac{R + G + B}{3}$$

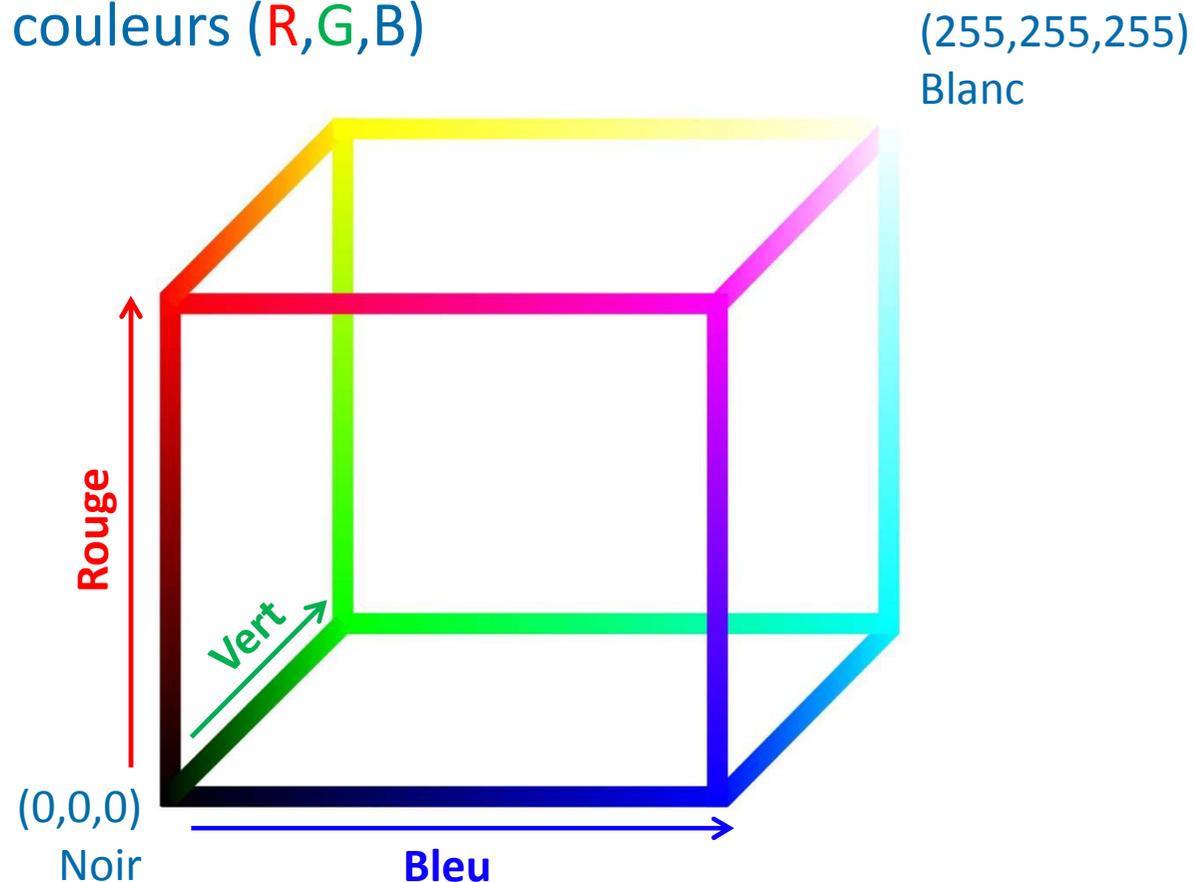
$$NdG = \frac{\log(R) + \log(G) + \log(B)}{3}$$

$$NdG = 0.30R + 0.59G + 0.11B$$

# Représentation numérique -> couleur (R,G,B)

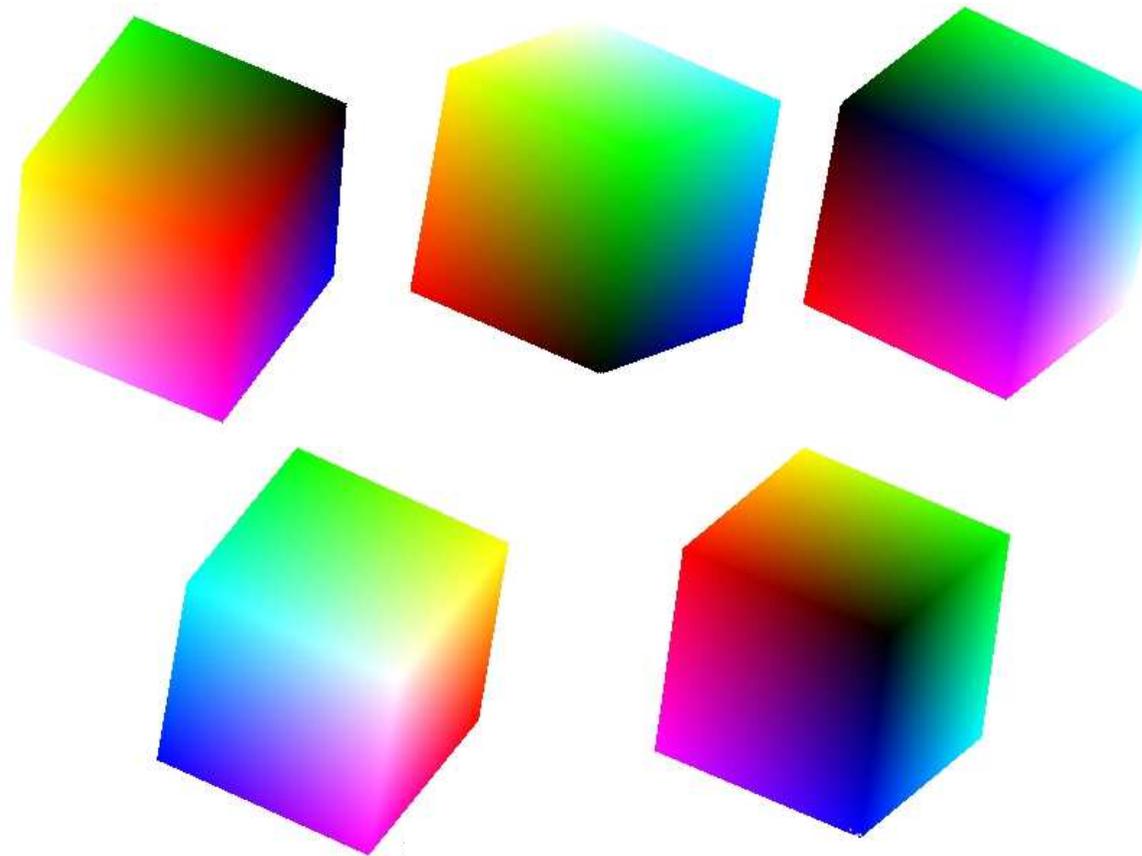
- Représentation numérique d'une image :

- Cube des couleurs (R,G,B)



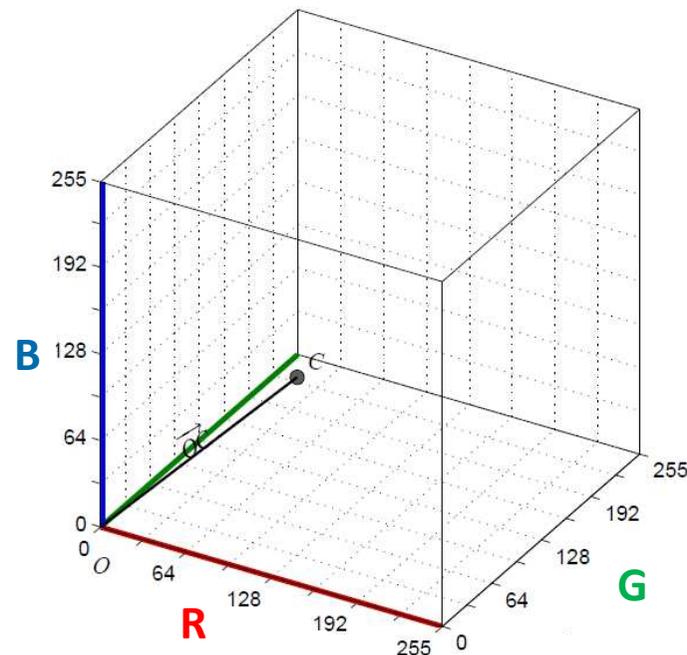
# Représentation numérique -> couleur (R,G,B)

- Représentation numérique d'une image :
  - Cube des couleurs (R,G,B)



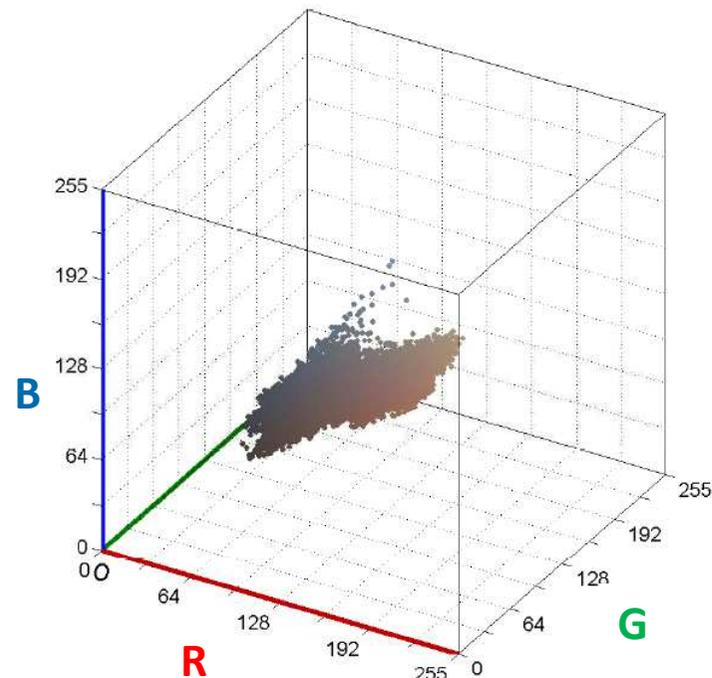
# Représentation numérique -> couleur (R,G,B)

- Représentation numérique d'une image :
  - La couleur d'un pixel est représentée par trois composantes **R**, **G** et **B**
  - Dans l'espace vectoriel (**R,G,B**), la couleur d'un pixel donne naissance à un point



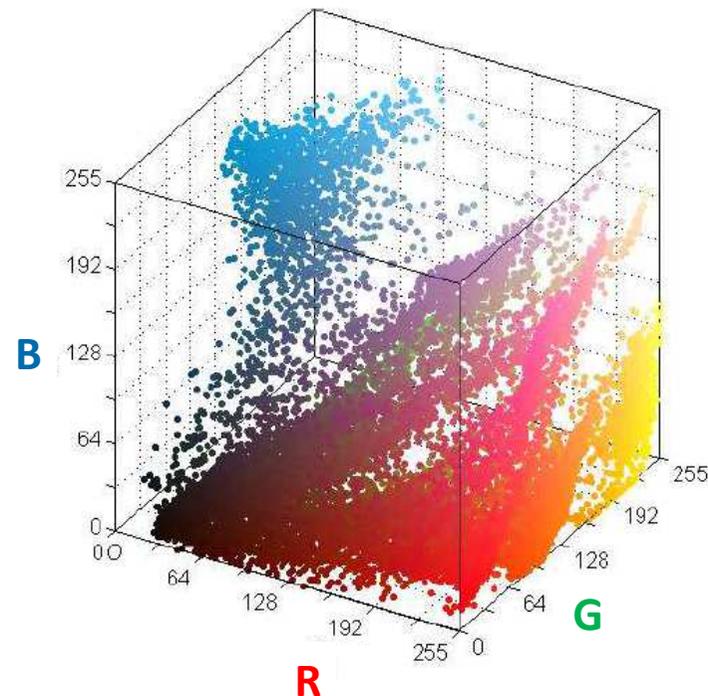
# Représentation numérique -> couleur (R,G,B)

- Représentation numérique d'une image :
  - Dans l'espace vectoriel (R,G,B), l'image est représentée par un nuage de points



# Représentation numérique -> couleur (R,G,B)

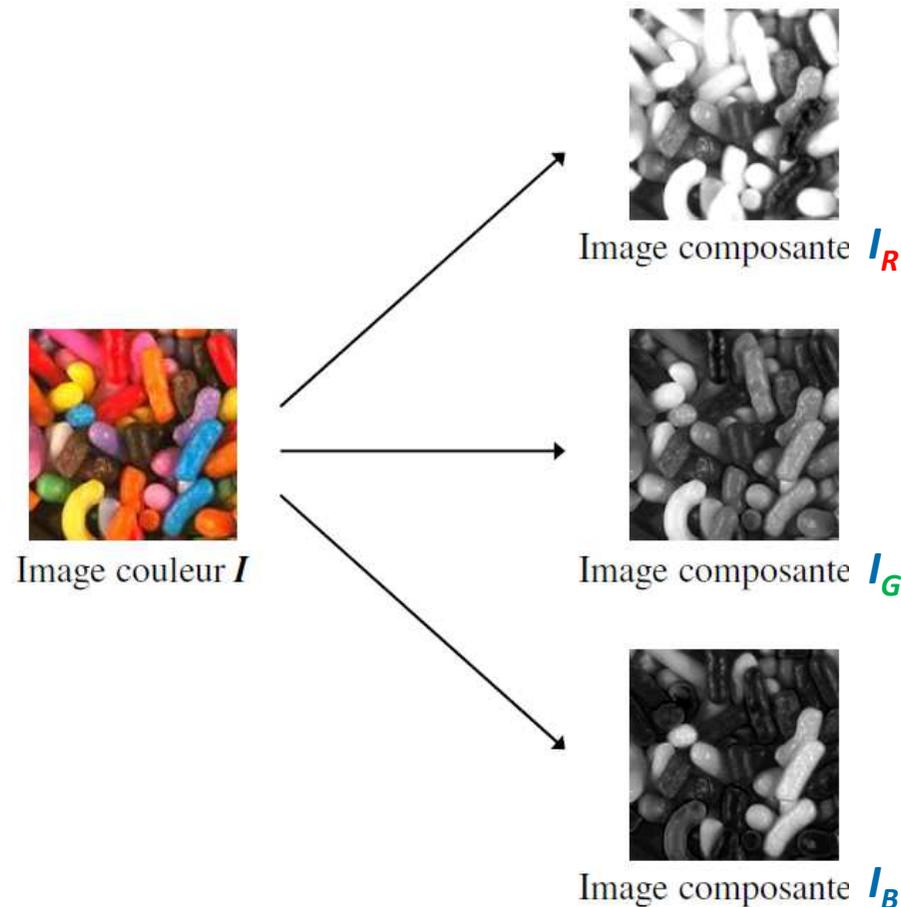
- Représentation numérique d'une image :
  - Représentation dans l'espace vectoriel (R,G,B)



# Représentation numérique -> couleur (R,G,B)

- Décomposition d'une image en images-composantes :

- A partir de l'image couleur  $I$  codée dans l'espace (R,G,B), on peut extraire trois images-composantes  $I_R$ ,  $I_G$  et  $I_B$ , où chaque pixel est caractérisé par une des composantes couleur R, G ou B



# Représentation numérique -> couleur (R,G,B)

- Décomposition d'une image en images-composantes :
  - Les trois images-composantes  $I_R$ ,  $I_G$  et  $I_B$  sont des images en niveaux de gris

255	200	255	255
255	0	200	200
255	200	100	100
0	200	100	0
0	0	200	100
0	200	0	255
200	0	100	0
255	255	255	100
200	255	255	255
200	200	100	100
200	200	100	100
200	200	200	200

255	200	255	255
0	200	100	0
200	0	100	0
200	200	100	100

255	0	200	200
0	0	200	100
255	255	255	100
200	200	100	100

255	200	100	100
0	200	0	255
200	255	255	255
200	200	200	200

# Les systèmes de représentation de la couleur

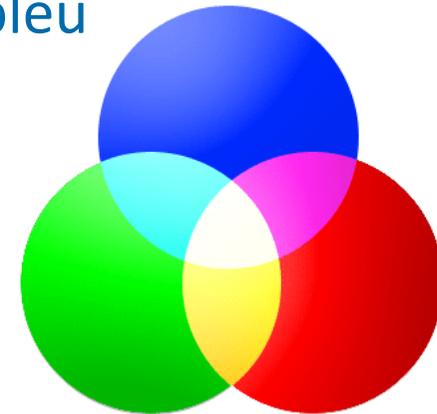
- Les autres systèmes de représentation de la couleur :
  - Espace couleur (R,G,B) :
    - système de représentation de la couleur le plus connu
    - permet d'acquérir les images par la plus grande majorité des caméras
  - Il existe de nombreux autres espaces couleurs qui ont été construits en se basant sur différentes propriétés physiques, physiologiques ou psychologiques de la perception de la couleur
  - Le choix d'un espace couleur est une étape importante en analyse d'images car il influence les résultats de l'analyse

# Les systèmes de représentation de la couleur

- Les systèmes de représentation de la couleur peuvent être regroupés en 4 familles :
  - les espaces de primaires,
  - les espaces luminance-chrominance,
  - les espaces perceptuels,
  - les espaces d'axes indépendants.

# Les systèmes de représentation de la couleur

- Les espaces de primaires :
  - Les espaces de primaires se basent sur le fait que tout stimulus de couleur peut être reproduit par le mélange de trois autres stimuli : le rouge, le vert et le bleu
  - C'est le principe de synthèse additive
  - Il existe une multitude d'espaces (R,G,B) : la définition de ces espaces dépend de la définition des composantes R,G et B utilisées pour reproduire la couleur



# Les systèmes de représentation de la couleur

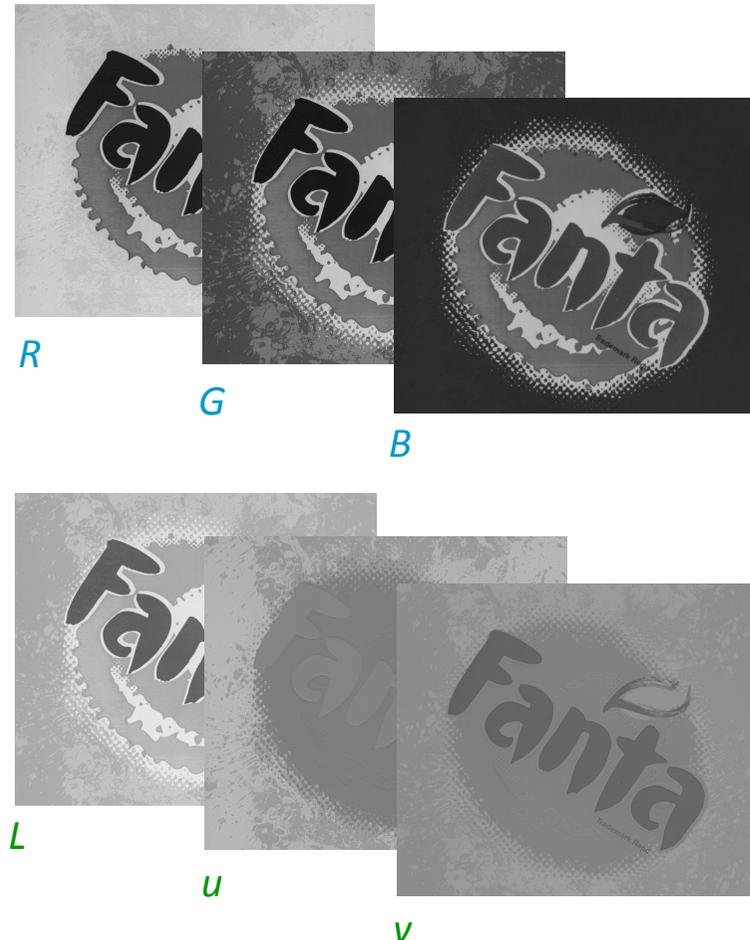
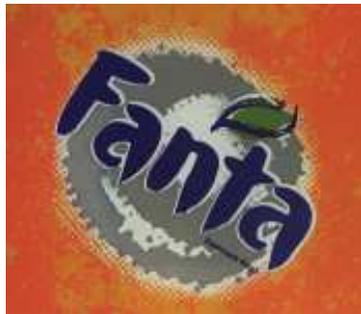
- Les espaces de primaires :
  - Exemples :
    - $(R_C, G_C, B_C)$  : les stimuli de couleur monochromatiques  $R_C$ ,  $G_C$  et  $B_C$  correspondent respectivement aux longueurs d'onde 700,0 nm, 546,1 nm et 435,8 nm -> convention de la CIE (Commission Internationale de l'Éclairage)
    - Les téléviseurs couleur analogiques utilisent des tubes cathodiques dont les luminophores émettent dans des longueurs d'onde différentes de celles définies par la CIE
      - norme NTSC :  $(R_F, G_F, B_F)$
      - normes PAL et SECAM :  $(R_E, G_E, B_E)$

# Les systèmes de représentation de la couleur

- Les espaces luminance-chrominance :
  - Les espaces luminance-chrominance possèdent :
    - une composante dite de "luminance", qui permet de quantifier le caractère achromatique d'un stimuli de couleur
    - deux autres composantes de chrominance permettant de quantifier son caractère chromatique
  - Exemples :
    - (L,a,b) - > L représente la luminance, a & b la chrominance
    - (L,u,v) - > L représente la luminance, u & v la chrominance
    - (Y,I,Q) - > Y représente la luminance, I & Q la chrominance
    - (Y,U,V) - > Y représente la luminance, U & V la chrominance

# Les systèmes de représentation de la couleur

- Les espaces luminance-chrominance :
  - Illustration :



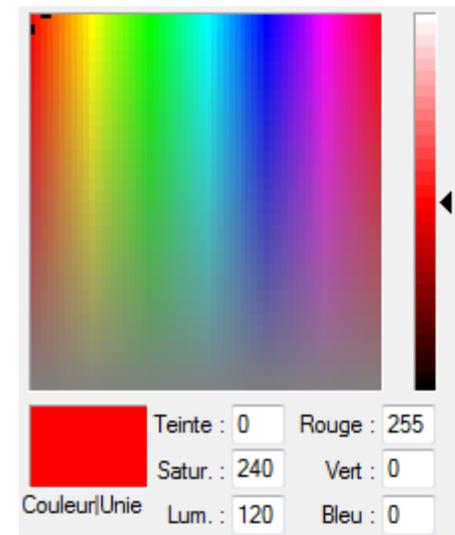
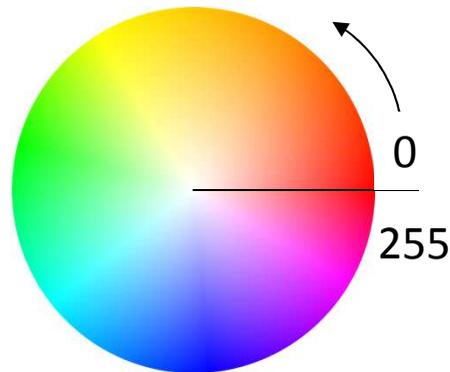
# Les systèmes de représentation de la couleur

- Les espaces perceptuels :
  - L'homme ne perçoit pas la couleur comme une combinaison de composantes trichromatiques liées à des primaires mais selon des entités plus subjectives liées à la luminosité, la teinte et la saturation
  - Les espaces perceptuels permettent de quantifier ces informations
  - Exemples : (I,S,H), (H,S,L), (H,S,V), (T,L,S), (L,C,H), ...



# Les systèmes de représentation de la couleur

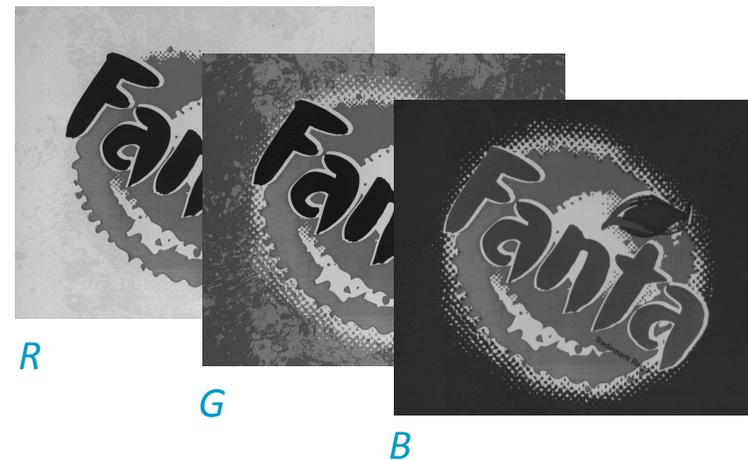
- Les espaces perceptuels :
  - Luminance : pourcentage de noir ou de blanc dans la couleur considérée, brillance de la couleur, aspect clair ou sombre
  - Teinte : correspond à la couleur souhaitée à partir des couleurs à disposition



- Saturation : mesure l'intensité ou la pureté d'une couleur, caractère vif ou terne

# Les systèmes de représentation de la couleur

- Les espaces perceptuels :
  - Illustration :

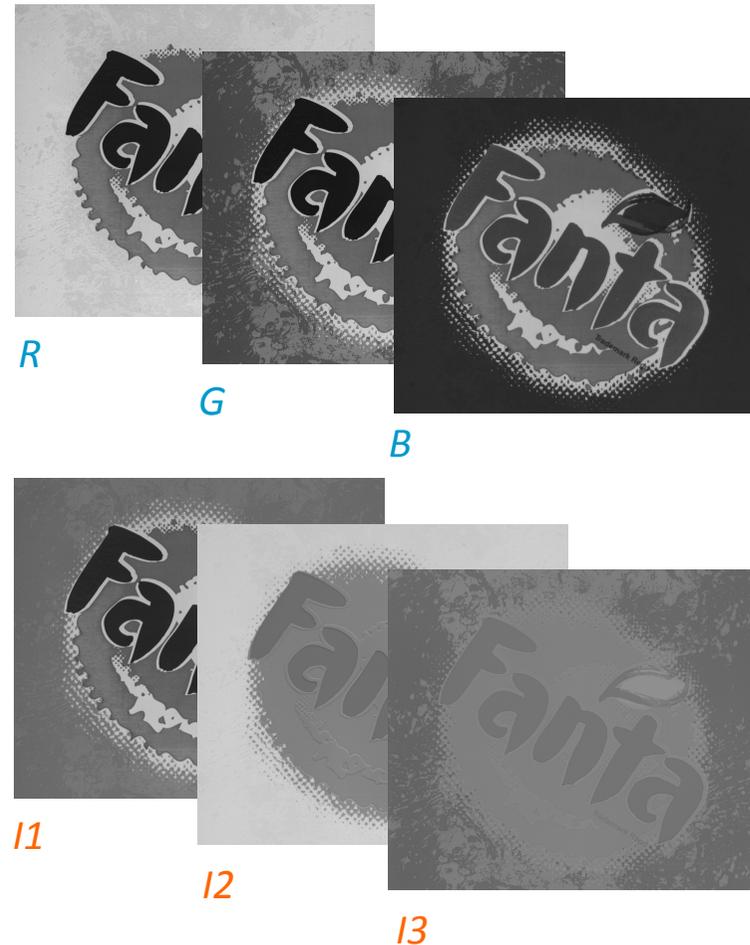
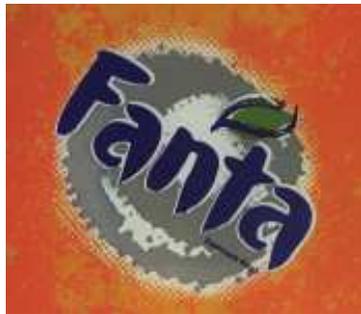


# Les systèmes de représentation de la couleur

- Les espaces d'axes indépendants :
  - Un des inconvénients des espaces (R,G,B) est que les trois composantes sont fortement corrélées. En effet, elles possèdent un fort facteur de luminance réparti sur chacune d'entre elles
  - Ainsi, de nombreux auteurs ont tenté de déterminer des systèmes de représentation de la couleur dont les composantes sont indépendantes, c'est-à-dire des composantes qui portent des informations différentes
  - Exemple : (I1,I2,I3)

# Les systèmes de représentation de la couleur

- Les espaces d'axes indépendants :
  - Illustration :



# Les systèmes de représentation de la couleur

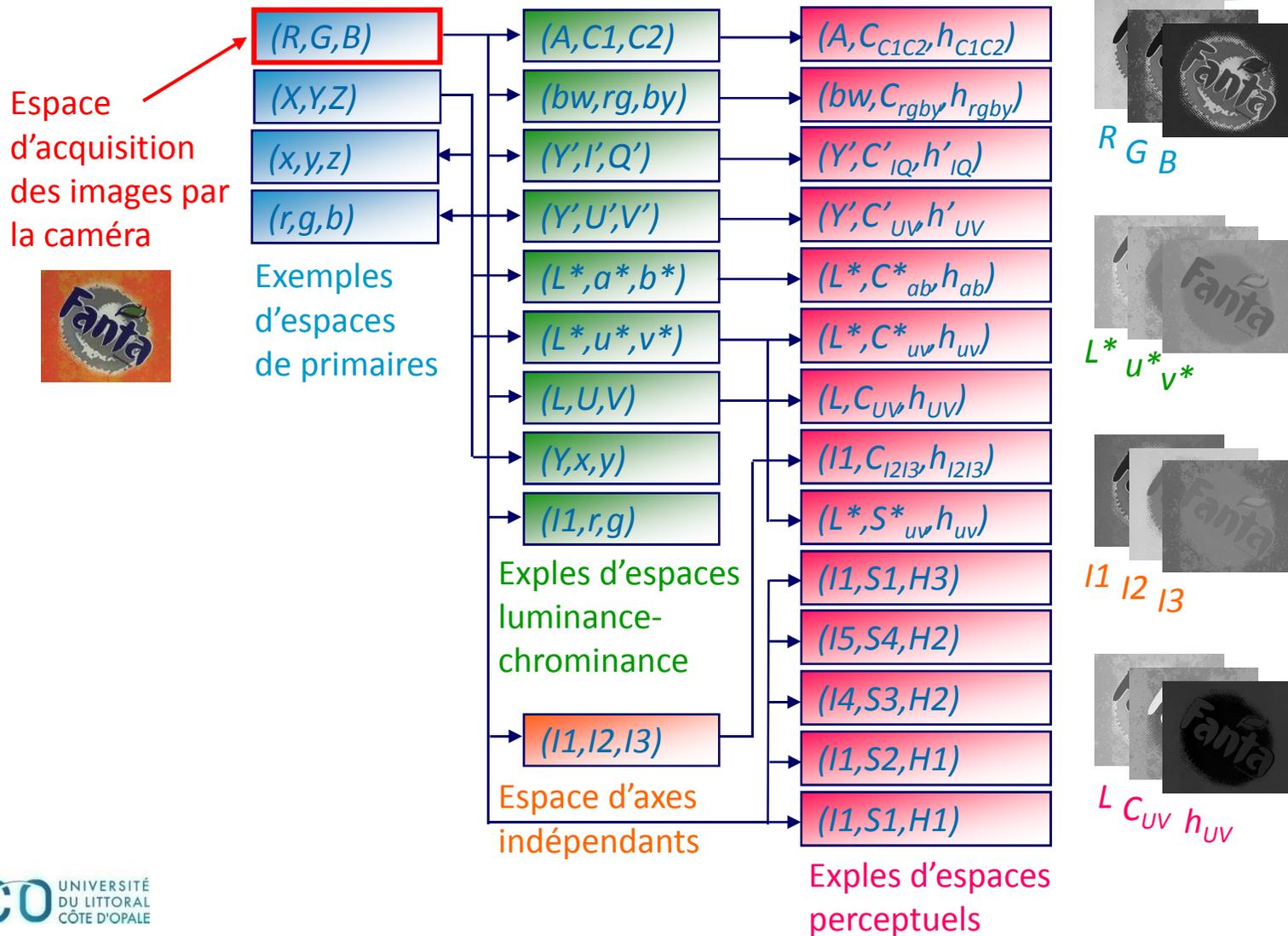
- Passage d'un espace couleur à un autre espace couleur :
  - Pour chacun des espaces couleur, il existe une relation, linéaire ou non, permettant de transformer la couleur codée dans l'espace d'acquisition (R,G,B) vers un codage dans un autre espace de représentation de la couleur
  - Exemple : (R,G,B) -> (I1,I2,I3)

$$I1 = \frac{R+G+B}{3},$$

$$I2 = \frac{R-B}{2},$$

$$I3 = \frac{2G-R-B}{4}.$$

# Les systèmes de représentation de la couleur

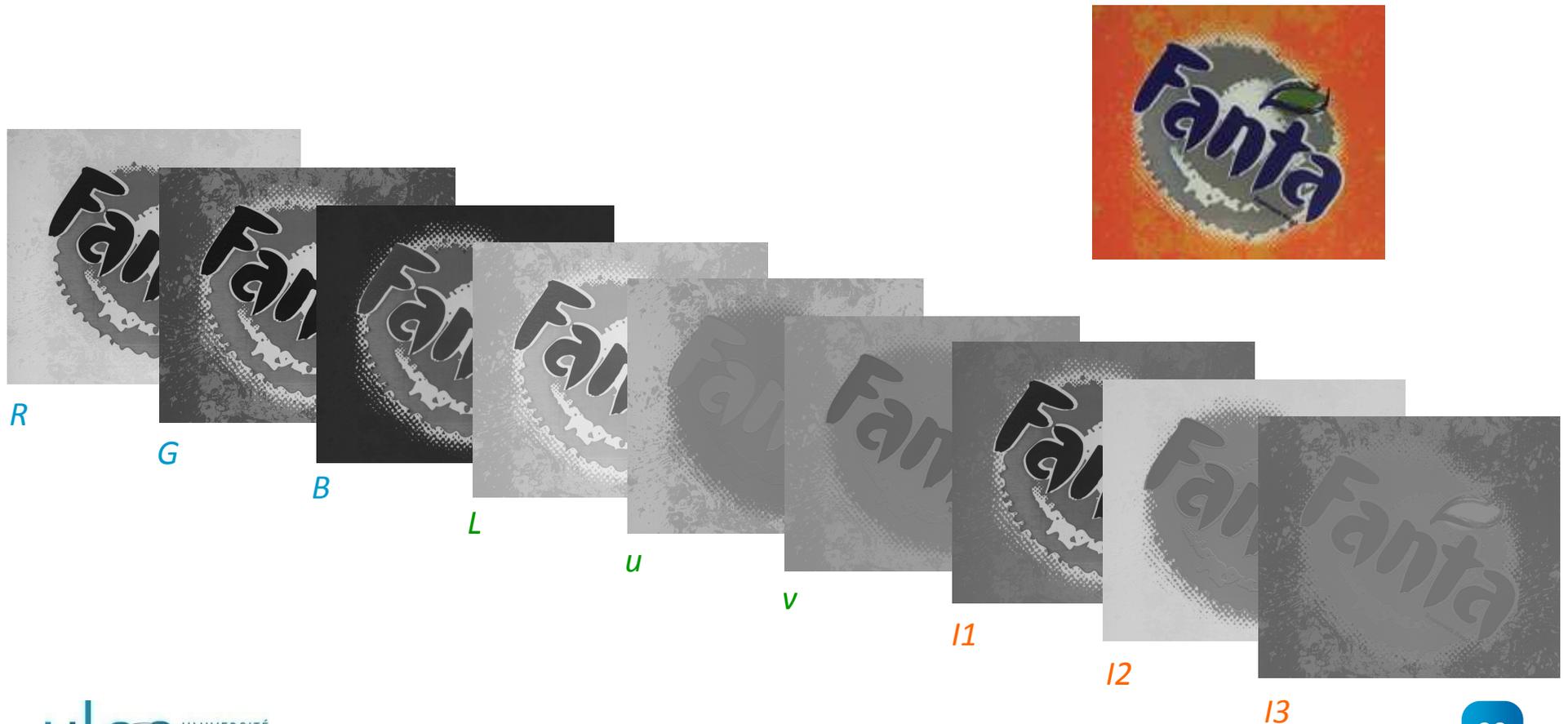


# Les systèmes de représentation de la couleur

- Importance du choix de l'espace couleur :
  - De nombreux auteurs ont comparé les résultats de méthodes d'analyse appliquées à des images dont la couleur est codée selon différents espaces couleur.
  - Ils en ont conclu que :
    - il n'existe pas d'espace couleur adapté à tous les problèmes d'analyse d'images couleur
    - les résultats obtenus dépendent du choix de l'espace couleur

# Les systèmes de représentation de la couleur :

- Importance du choix de l'espace couleur :



# Les systèmes de représentation de la couleur

- Importance du choix de l'espace couleur :
  - Solutions :
    - Étude préliminaire sur le meilleur espace couleur en fonction de l'application considérée
    - Au lieu de chercher l'espace couleur le mieux adapté à un problème d'analyse d'images, il peut être judicieux de construire un espace couleur hybride, composé de composantes couleur n'appartenant pas forcément toutes au même espace couleur mais pouvant être issues de différents espaces  
Exemple :  $(R, I3, a)$
    - Considérer plusieurs espaces couleur simultanément

# Prétraitement d'images

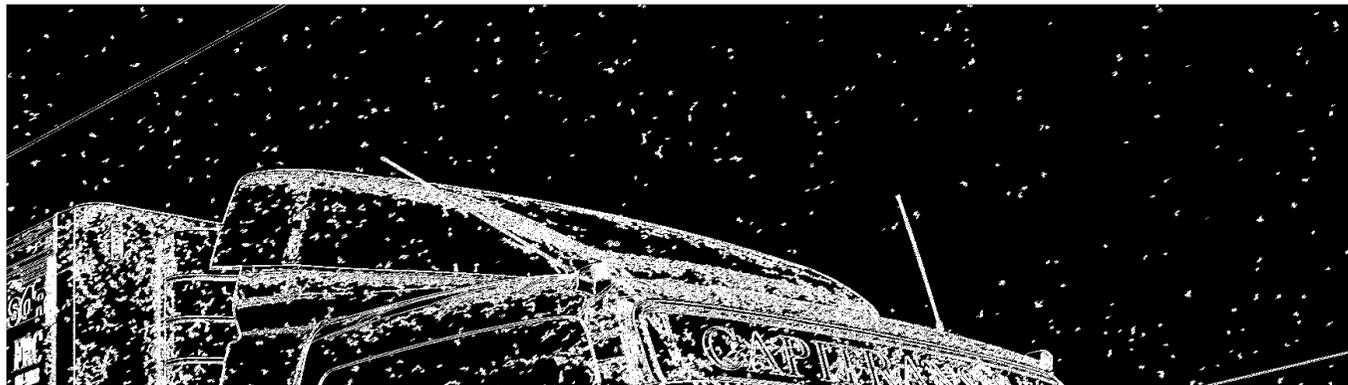
## Prétraitement d'images



# Prétraitement d'images

- Objectif du prétraitement : préparer l'image au traitement et à l'analyse

Sans  
prétraitement



Avec  
prétraitement

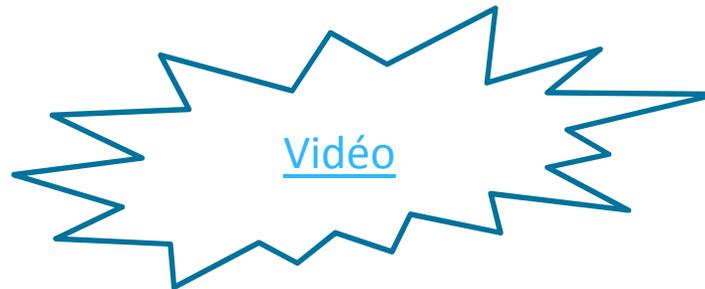


# Prétraitement d'images

- Objectif du prétraitement :
  - Mettre en relief l'information utile contenue dans l'image
  - Atténuer, voire éliminer, l'information inutile
  - Dépend de la finalité à atteindre
- Il existe différents types de prétraitement :
  - Les opérations « ponctuelles » : elles modifient la valeur de chaque pixel de l'image sans tenir compte des pixels voisins
  - La réduction du bruit
  - Le rehaussement de contraste
  - La compression

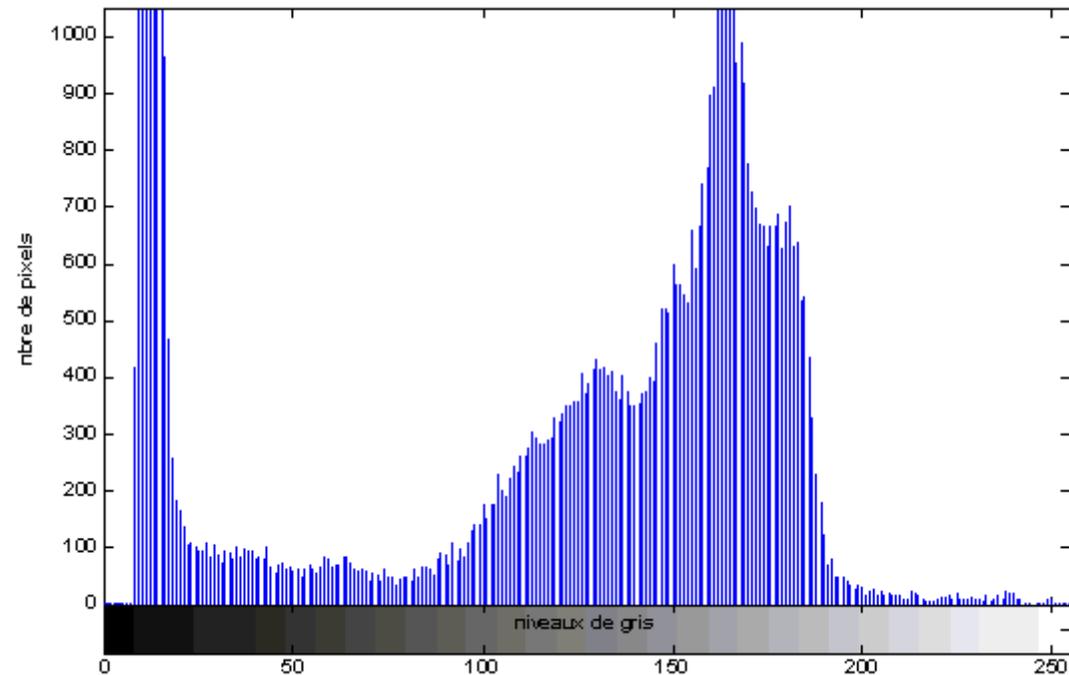
# Histogramme de l'image

- Les opérations ponctuelles : elles correspondent à des modifications de l'histogramme de l'image
  - Définition :  
L'**histogramme** représente la distribution des intensités (ou des couleurs) de l'image. C'est une fonction discrète qui représente le nombre de pixels en fonction du niveau de gris.



# Histogramme de l'image

- Exemple : image en niveau de gris et histogramme associé



# Opérations ponctuelles



- Les opérations ponctuelles :
  - Expansion de dynamique
  - Egalisation d'histogramme
  - Spécification d'histogramme

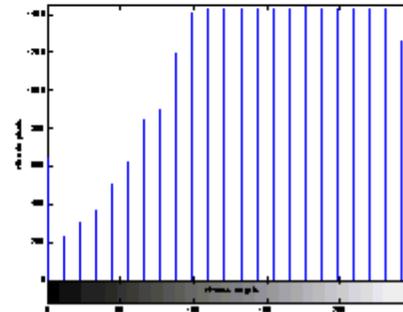
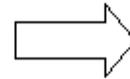
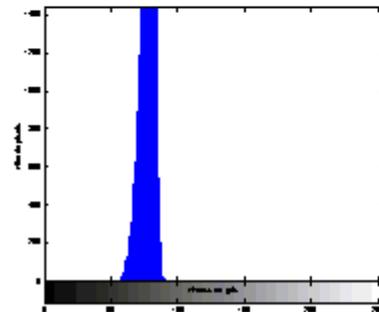
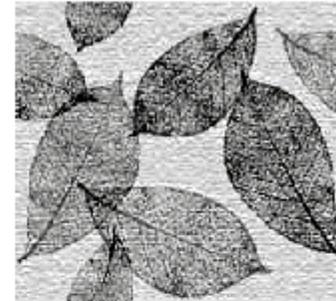
# Expansion de dynamique

- Expansion de dynamique
  - Objectif : utiliser au mieux l'échelle de niveaux de gris disponible
  - Améliore le contraste
  - Transformation linéaire des niveaux de gris
  - Utilisation d'une table de conversion -> LUT (Look Up Table) :  
Soit  $N$ , le niveau de quantification (en général  $N=256$ ) :  
 $n \in [n_{\min}; n_{\max}] \rightarrow n' \in [0; 255]$

$$n' = E \left( (N - 1) \frac{(n - n_{\min})}{n_{\max} - n_{\min}} \right) \quad H(n') = H(n)$$

# Expansion de dynamique

- Exemple :



# Egalisation d'histogramme

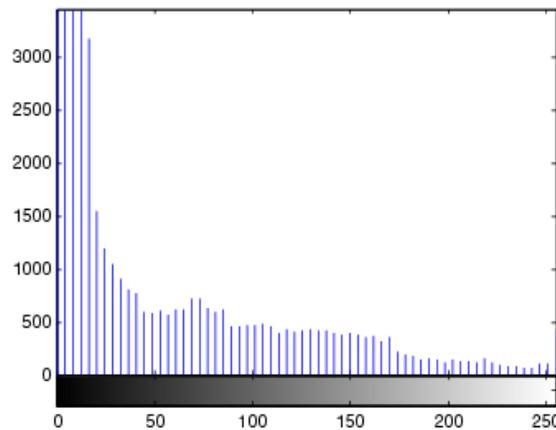
- Egalisation d'histogramme
  - Définition : opération consistant à rendre l'histogramme le plus « plat » possible
  - Améliore le contraste
  - Utilisation d'une LUT :

$$n' = E \left( (N - 1) \sum_{j=0}^n H_{norm}(j) \right) \quad H_{norm}(j) = \frac{H(j)}{\sum_{i=0}^{N-1} H(i)}$$

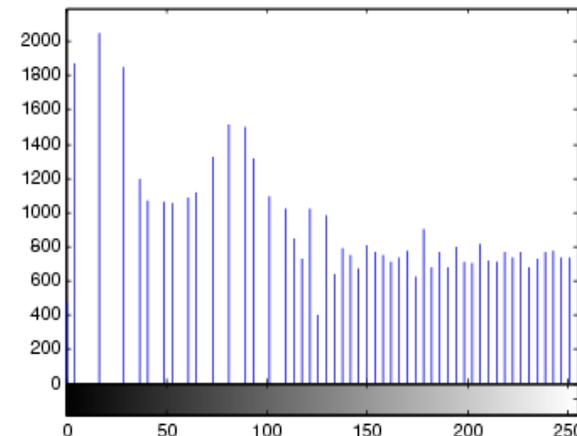
# Egalisation d'histogramme

- Exemple :

Sans égalisation



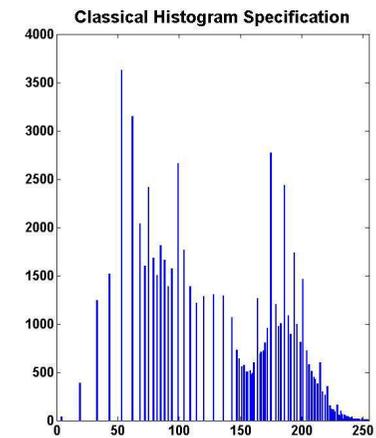
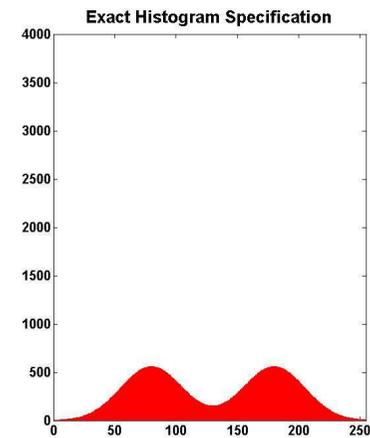
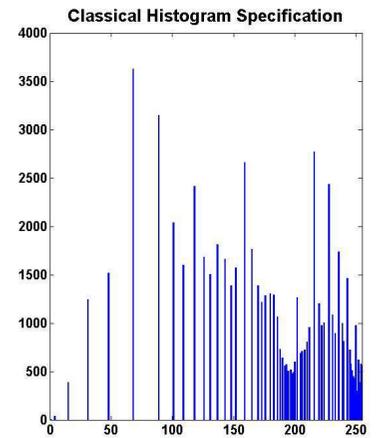
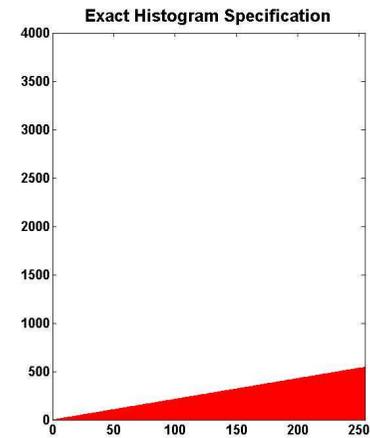
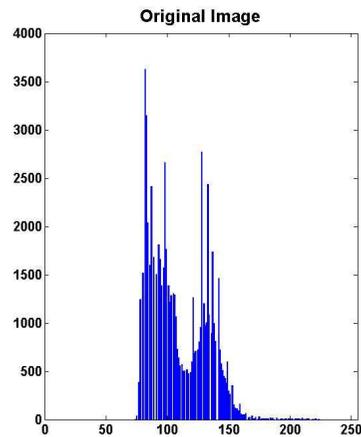
Avec égalisation



# Spécification d'histogramme

- Spécification d'histogramme
  - Cette transformation consiste à rendre la distribution de niveaux de gris voisine d'une distribution spécifiée à l'avance

# Spécification d'histogramme



# Réduction du bruit

- La réduction du bruit
  - Définition :

Le **bruit** correspond à des perturbations provenant soit du dispositif d'acquisition, soit de la scène observée elle-même



Image non bruitée



Image bruitée

# Réduction du bruit

- La réduction du bruit
  - Les filtres linéaires :
    - Filtre moyenneur
    - Filtre gaussien
    - ...
  - Les filtres non linéaires :
    - Filtre médian
    - Filtres morphologiques
    - ...

# Filtrage

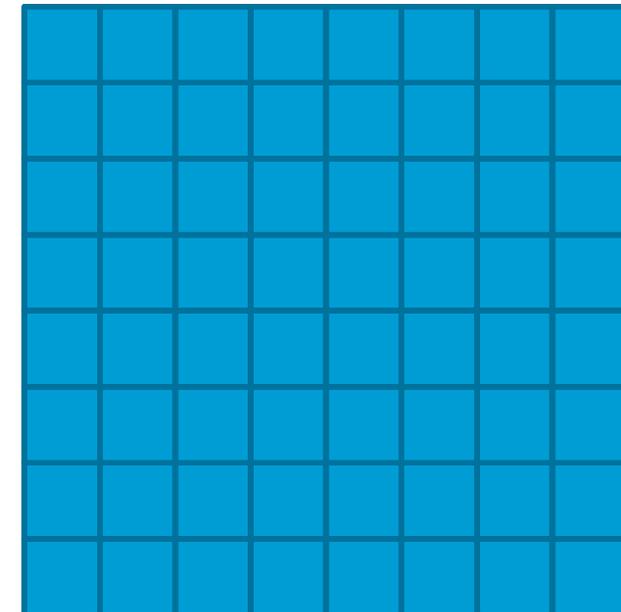
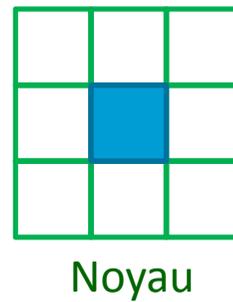
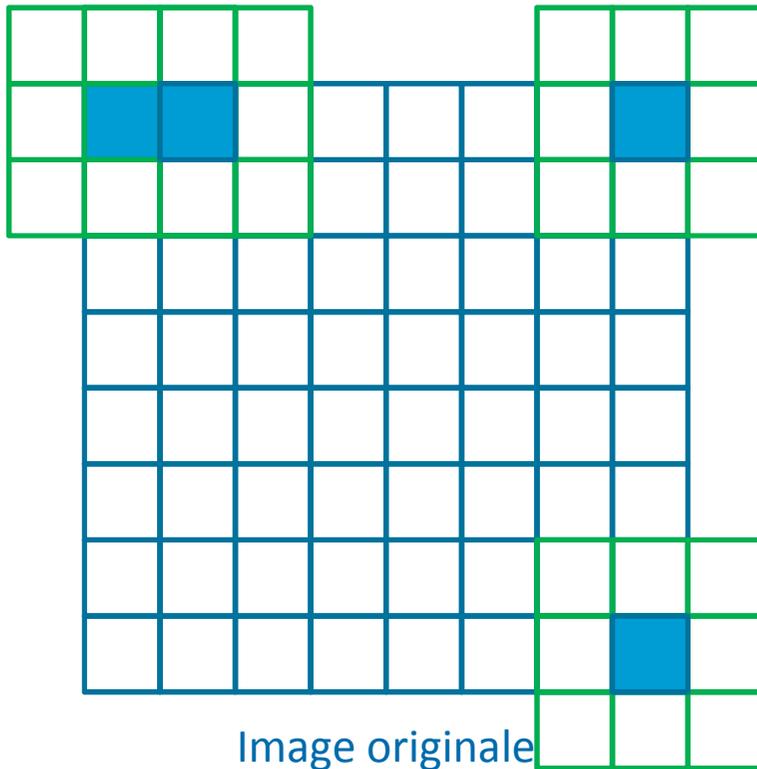
- Principe du filtrage

$h(-1,-1)$	$h(0,-1)$	$h(1,-1)$
$h(-1,0)$	$h(0,0)$	$h(1,0)$
$h(-1,1)$	$h(0,1)$	$h(1,1)$

Filtre  $h$  de taille  $3 \times 3$

# Filtrage

- Principe du filtrage



# Réduction du bruit

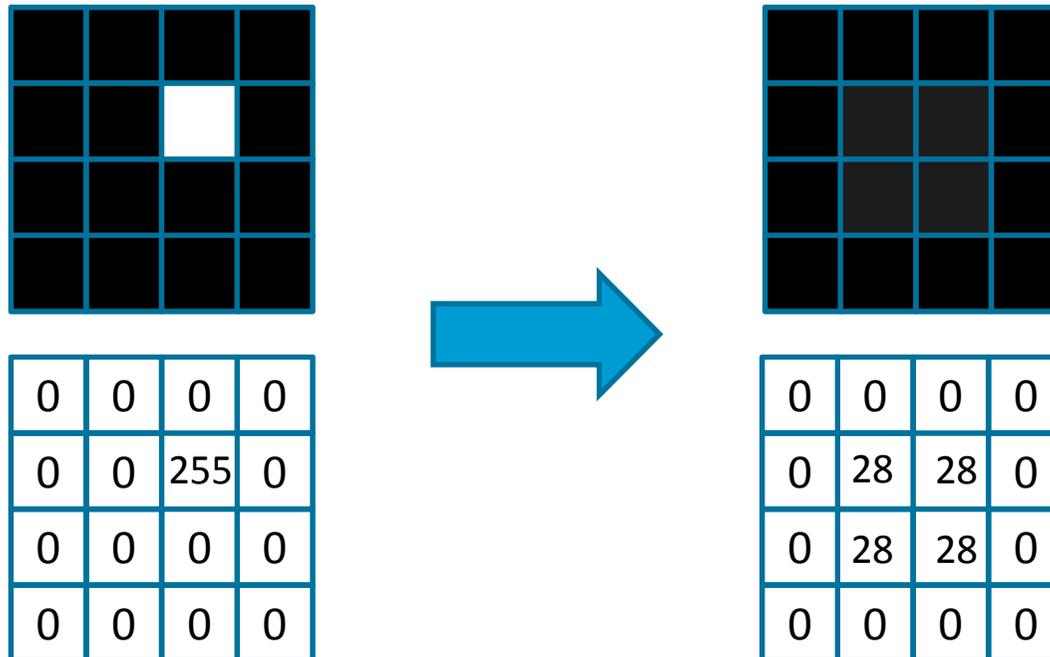
- Filtre moyennneur

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

$1/10$	$1/10$	$1/10$
$1/10$	$2/10$	$1/10$
$1/10$	$1/10$	$1/10$

# Réduction du bruit

- Filtre moyenneur -> Exemple : élimination du bruit



# Réduction du bruit

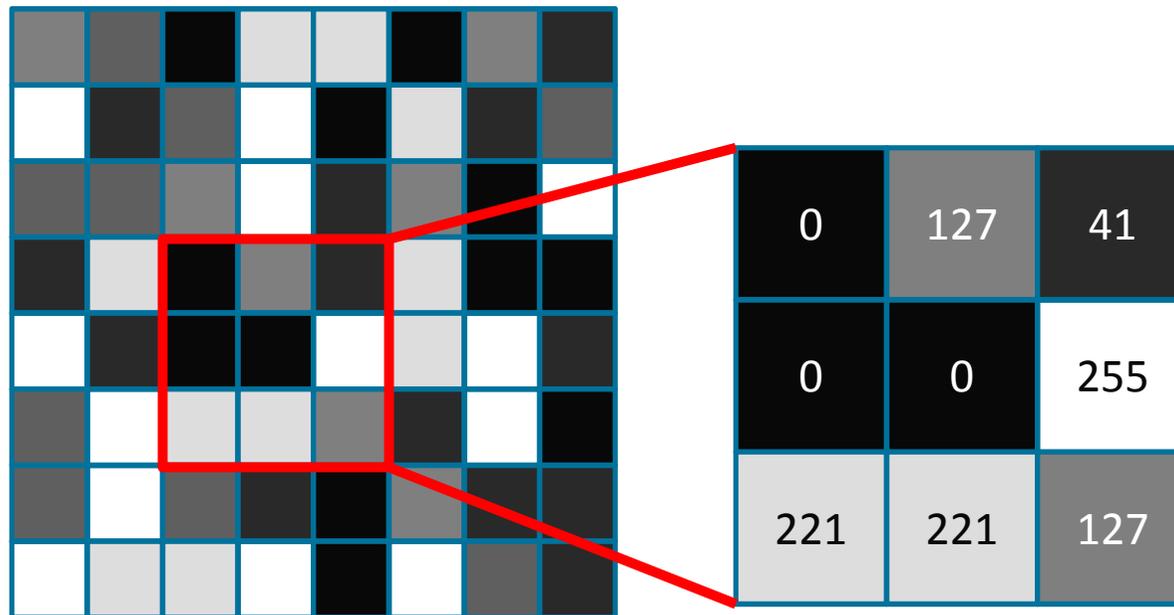
- Filtre gaussien

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

0,011	0,084	0,011
0,084	0,619	0,084
0,011	0,084	0,011

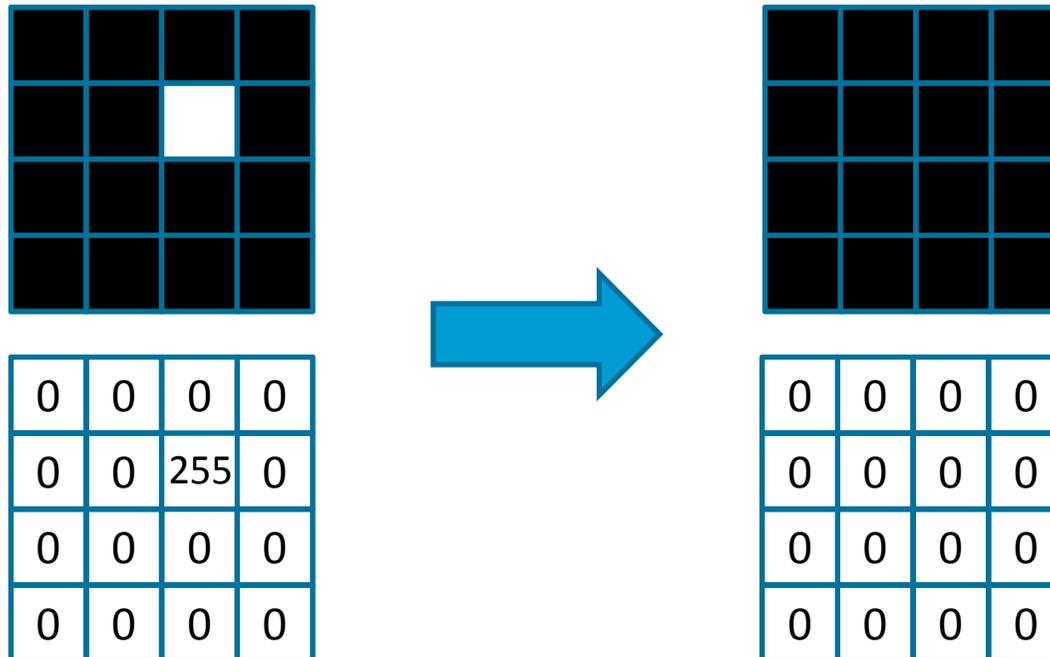
# Réduction du bruit

- Filtre médian



# Réduction du bruit

- Filtre médian -> Exemple : élimination du bruit

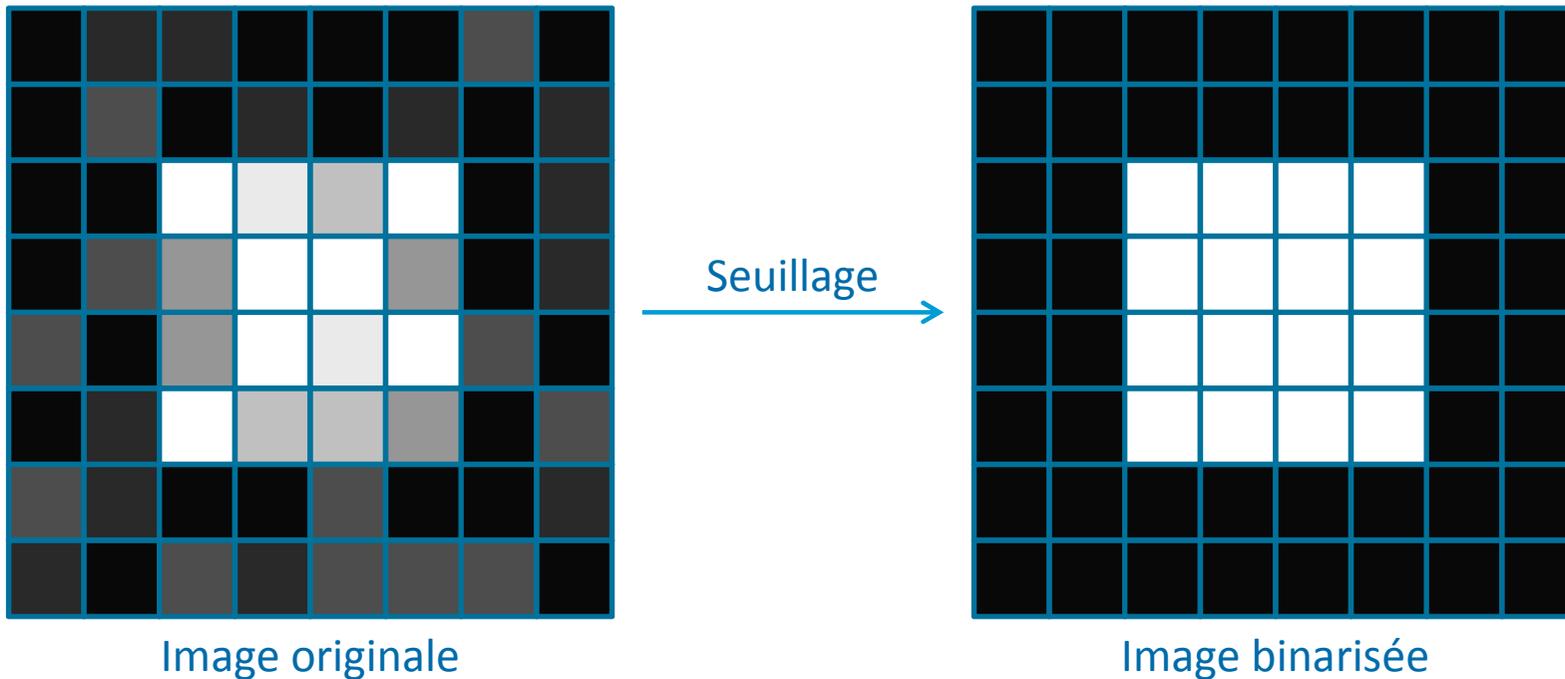


# Réduction du bruit

- Filtres morphologiques
  - Appliqués sur une image binaire
  - Utilisation d'un élément structurant
  - Opérations :
    - Érosion
    - Dilation
  - Ouverture
  - Fermeture

# Binarisation

- Binarisation d'une image



**-> tous les niveaux de gris inférieurs ou égaux au seuil prennent la valeur 0 sinon 1**

# Réduction du bruit

- Filtres morphologiques

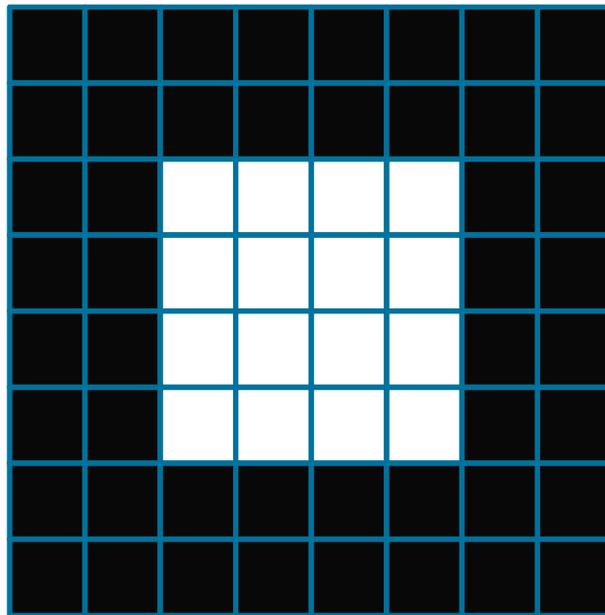
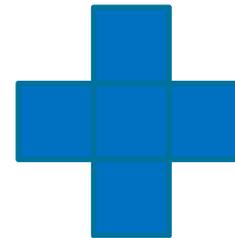
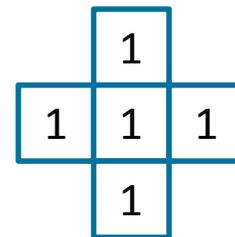


Image binarisée



Exemple  
d'élément  
structurant



# Réduction du bruit

- Erosion : min

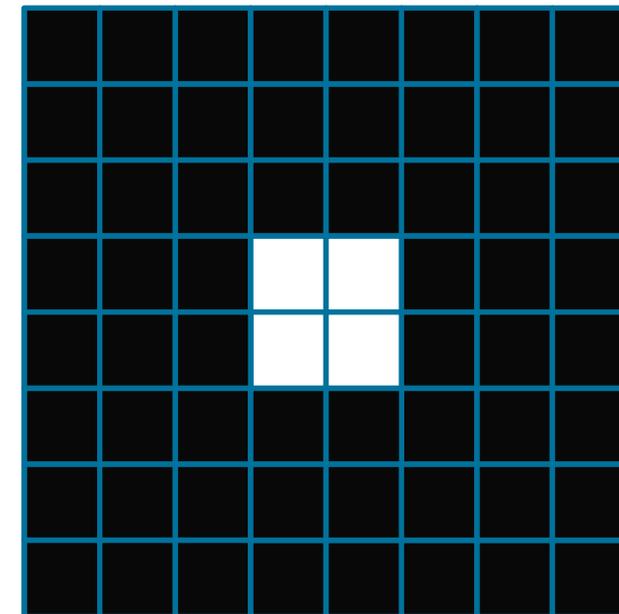
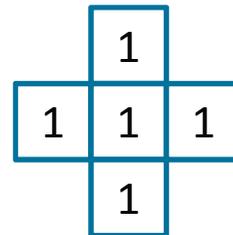
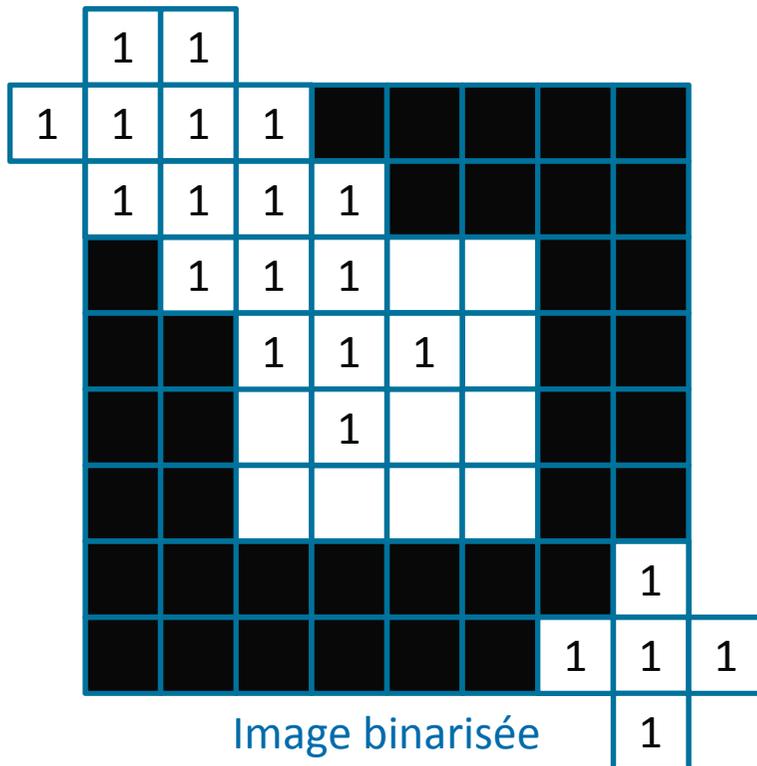
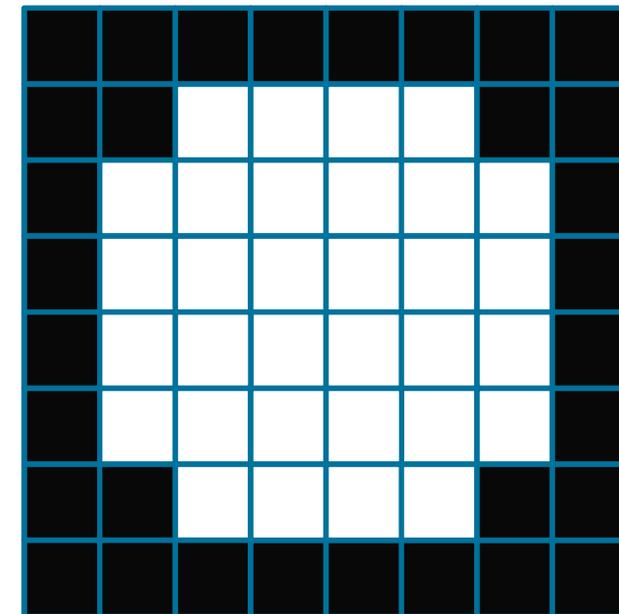
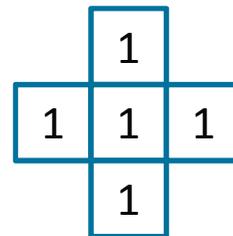
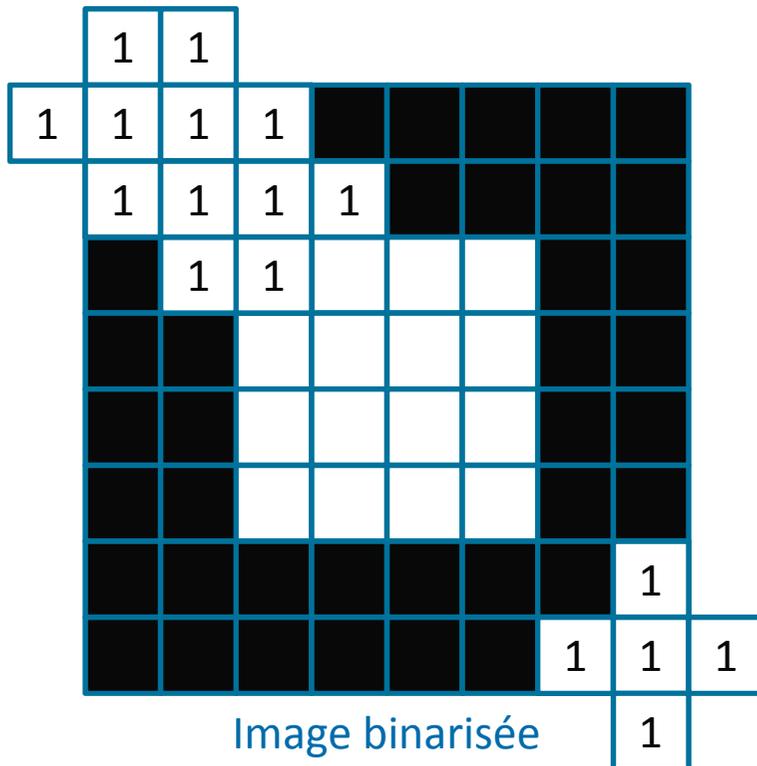


Image après érosion

# Réduction du bruit

- Dilatation : max



# Réduction du bruit

- Filtres morphologiques
  - Ouverture : érosion suivie d'une dilatation
  - Fermeture : dilatation suivie d'une érosion
  - Illustration de la réduction du bruit :

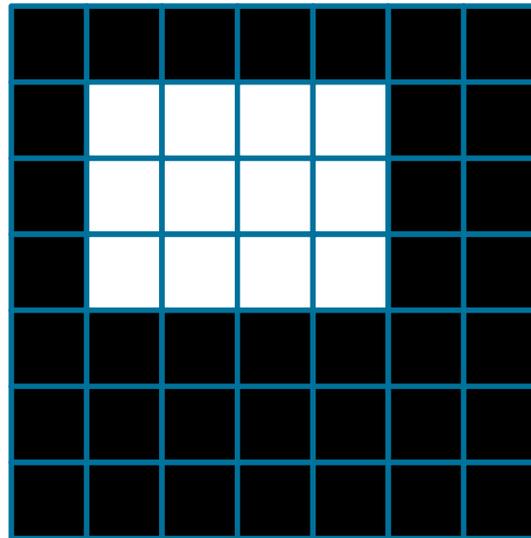


Image originale



Élément  
structurant

# Réduction du bruit

- Ouverture :

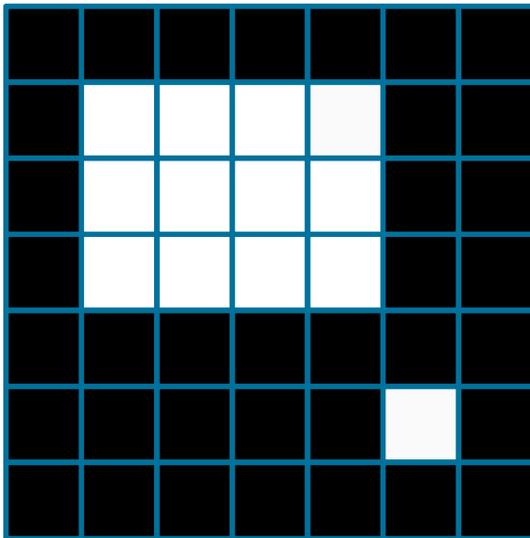


Image bruitée

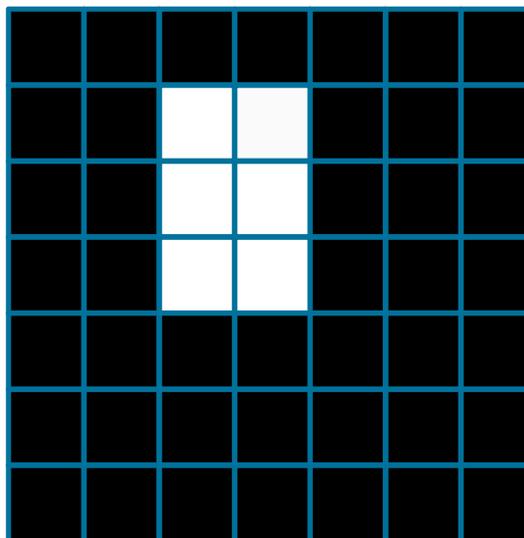


Image après  
érosion

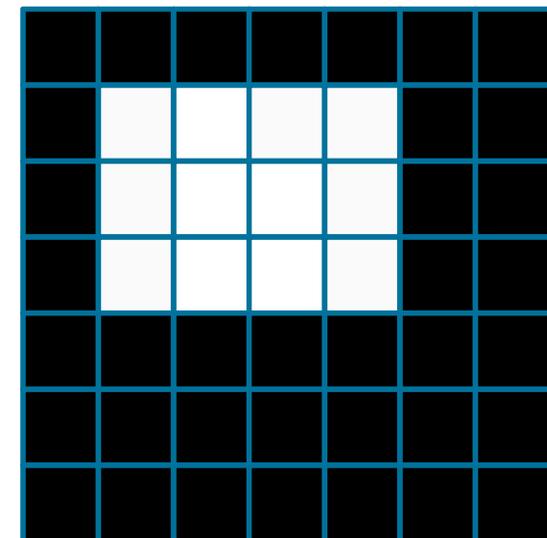


Image érodée  
ayant subi une  
dilatation

# Réduction du bruit

- Fermeture :

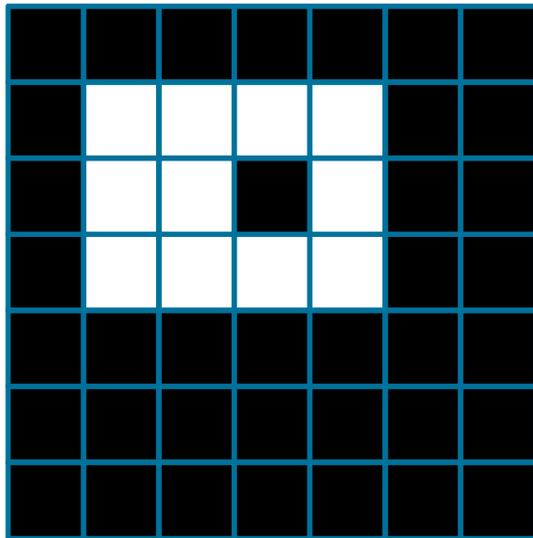


Image bruitée

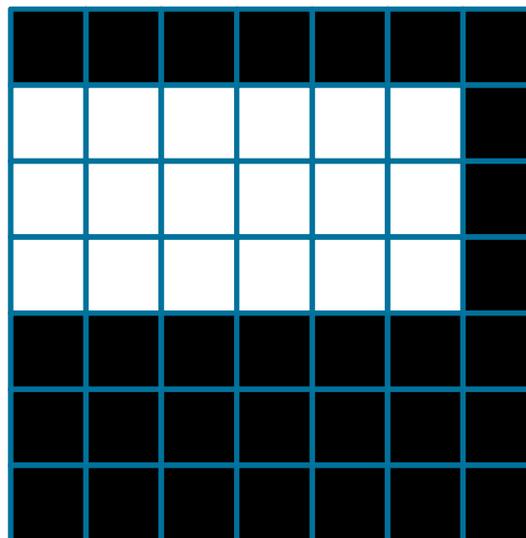


Image après  
dilatation

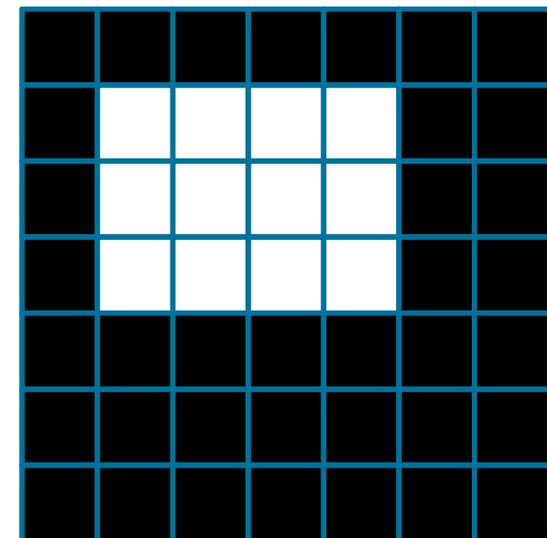


Image dilatée  
ayant subi une  
érosion

# Rehaussement de contraste

- Le rehaussement de contraste
  - Filtre rehausseur

-1	-1	-1
-1	9	-1
-1	-1	-1

# Rehaussement de contraste

- Le rehaussement de contraste
  - Filtre Laplacien

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

# Compression d'images

- Les formats d'images non compressées :
  - Bmp :
    - Pas de compression = pas de perte de qualité
    - Mais fichiers de grande taille, impossibles à afficher sur internet pour un utilisateur ayant une connexion bas débit
  - Tiff :
    - Format à destination des professionnels (imprimeurs, publicitaires...)
    - Reconnu sur tout type de système d'exploitation : Windows, Linux, ...
    - Image de très bonne qualité, mais taille volumineuse, même si elle est inférieure à celle des fichier BMP

# Compression d'images

- Les formats d'images compressées

-> Compression avec perte

- Jpeg :

- Taux de compression inégalés, même si la qualité de l'image s'en ressent au fur et à mesure que la compression est augmentée
- Fichiers images de petite taille -> standard des formats d'image sur internet

- Gif :

- Ce format est l'autre standard d'internet. Les fichiers gif sont de petites tailles, ce qui est dû au fait que ces images ne peuvent enregistrer que 256 couleurs
- Le format gif permet également la création d'animations



# Compression d'images

- Compression avec perte :
  - Par transformée en cosinus discrète : JPEG
    - Décomposition de l'image en blocs (en général 8x8)
    - Application de la transformée en cosinus discrète (DCT) sur chaque bloc
    - La DCT est une transformation permettant de passer du domaine spatial au domaine fréquentiel (similaire à la transformée de Fourier)

**Basses fréquences** : variations lentes et faibles / zones presque uniformes



**Hautes fréquences** : variations rapides et contrastées / contours / bruit / détails

# Compression d'images

- Compression avec perte :
  - Par transformée en cosinus discrète : JPEG
    - L'œil humain est plus sensible aux basses fréquences -> pour la compression JPEG, les composantes fréquentielles de faible amplitude et les hautes fréquences sont supprimées
    - Un codage à longueur variable de type Huffman est enfin utilisé (cf. compression sans perte)

100% fidelity  
Image is 725kB



90%  
250kB



10%  
37kB



1%  
20kB



# Compression d'images

- Compression avec perte :
  - Par transformée en cosinus discrète : JPEG
  - Par transformée en ondelettes : JPEG 2000
  - Par détection de motifs images redondants (fractale)
  - Par quantification

0	1	2	...	15	16	...																254	255	
0					1					...					...					15				



N=256

N=128

N=64

N=32

N=16

N=8

N=4

N=2

# Compression d'images

- Evaluation de la qualité de la compression avec perte
  - Erreur quadratique moyenne :

$$E = \frac{1}{N_P} \sum_{i,j} (I(i, j) - I'(i, j))^2$$

$N_P$  : nb de pixels de l'image

- Rapport signal sur bruit :

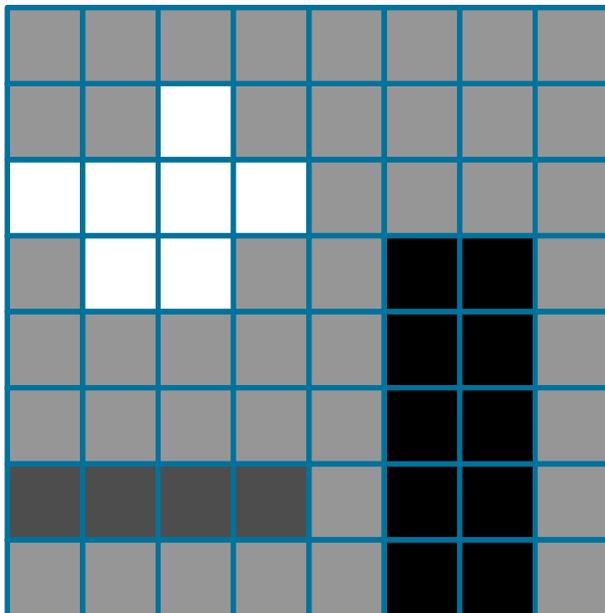
$$R = 10 \log_{10} \left( \frac{V}{E} \right) \quad \text{avec} \quad V = \frac{1}{N_P} \sum_{i,j} I(i, j)^2$$

# Compression d'images

- Les formats d'images compressées
  - > Compression sans perte
    - Png :
      - Futur standard internet
      - Taille des fichiers raisonnable
      - Permet la compression sans perte de donnée

# Compression d'images

- Compression sans perte :
  - RLE (Running Length Encoding ou codage par plages)

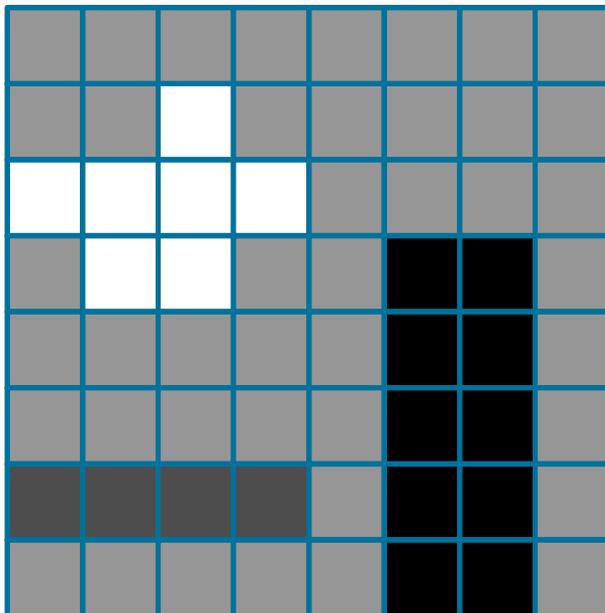


- Chaque niveau de gris est codé sur 1 octet
- Taille de l'image : 8 x 8 x 1 octet = 64 octets
- RLE : on analyse les pixels consécutifs -> **un octet pour indiquer le niveau de gris** et **un octet pour indiquer le nombre de pixels ayant cette valeur**
- Taille de l'image après compression : 38 octets
- Adapté aux images simples

150 – 10 – 255 – 1 – 150 – 5 – 255 – 4 – 150 – 5 – 255 – 2 – 150 – 2 – 0 – 2 – 150 – 6 – ...

# Compression d'images

- Compression sans perte :
  - Codage de Huffman



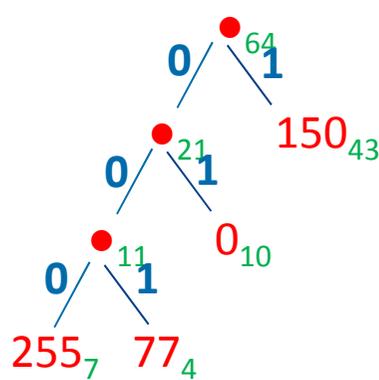
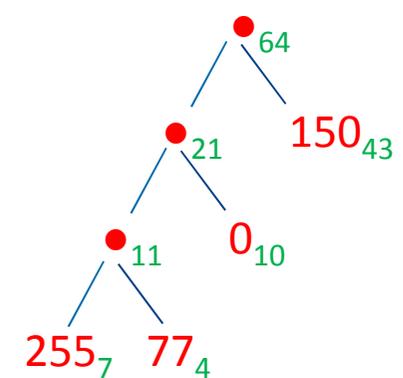
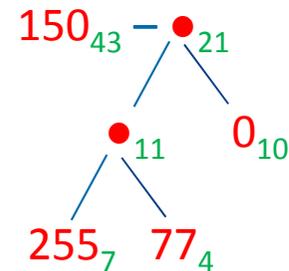
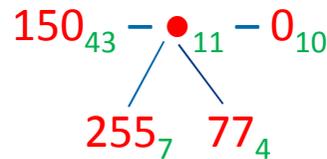
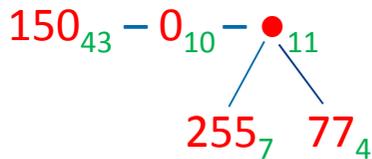
1. Trier les différentes valeurs par ordre décroissant de fréquence d'apparition  
 $150_{43} - 0_{10} - 255_{7} - 77_{4}$
2. Fusionner les deux poids minimaux
3. Réordonner
4. Recommencer l'étape 2 autant de fois que possible
5. Affecter des valeurs binaires

# Compression d'images

- Compression sans perte :

- Codage de Huffman

$$150_{43} - 0_{10} - 255_7 - 77_4$$



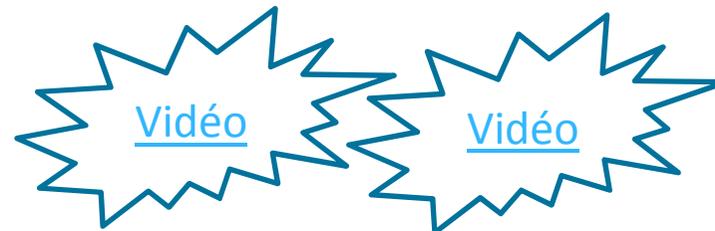
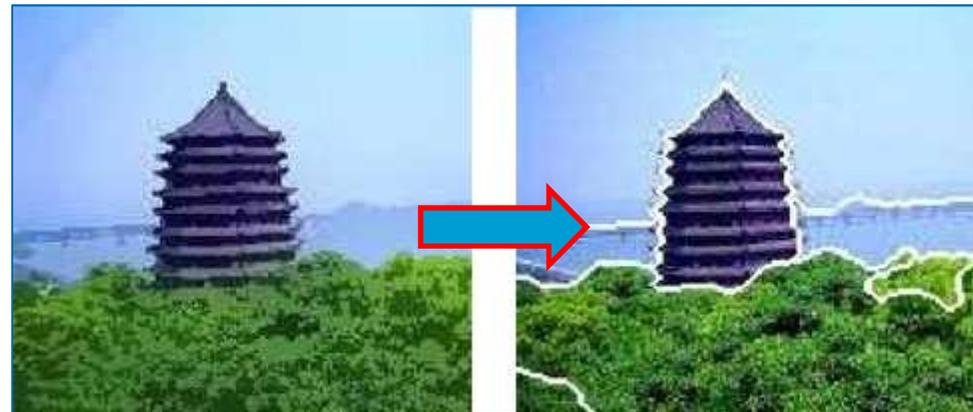
1	150
01	0
001	77
000	255

- Niveau de gris codé sur 1 bit au lieu de 8
- Niveau de gris codé sur 2 bits au lieu de 8
- Niveau de gris codé sur 3 bits au lieu de 8
- Niveau de gris codé sur 3 bits au lieu de 8

➔ Taille de l'image : 96 bits (12 octets)

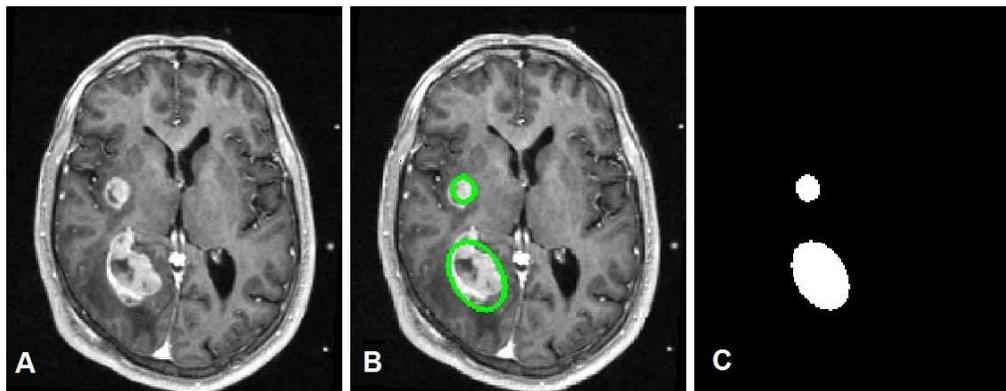
# Segmentation d'images

## Segmentation d'images



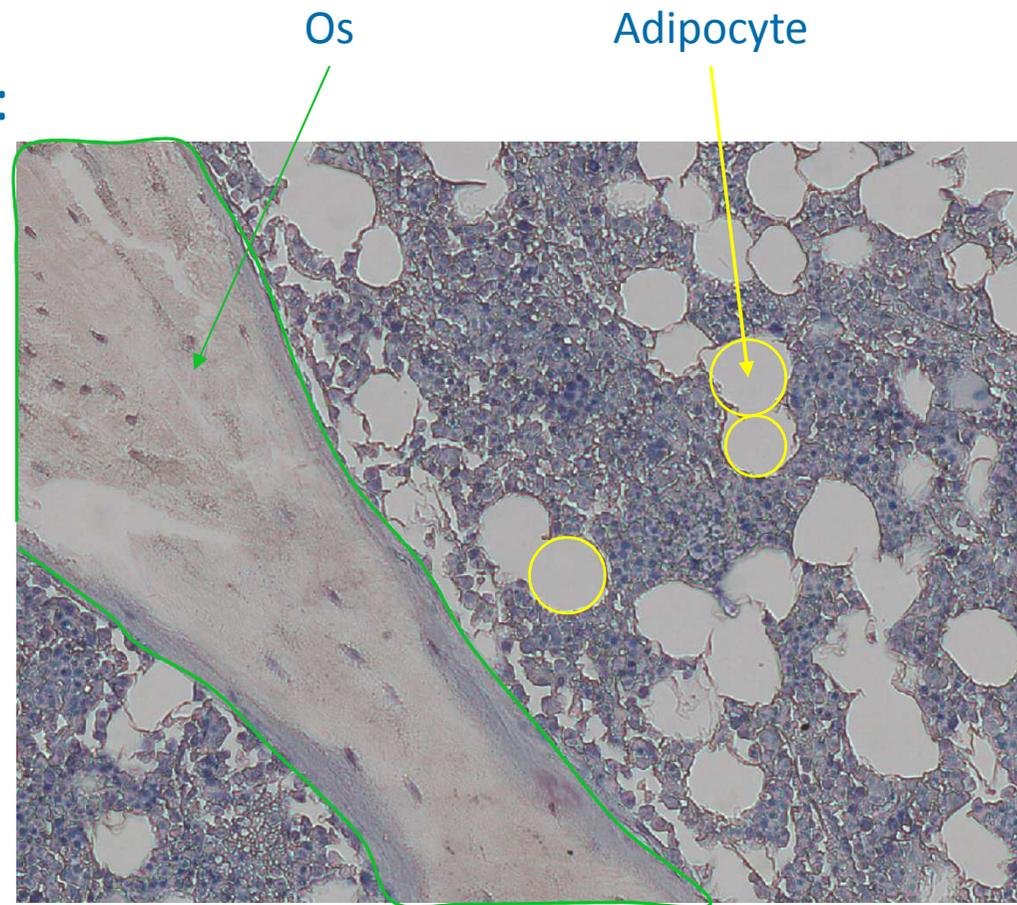
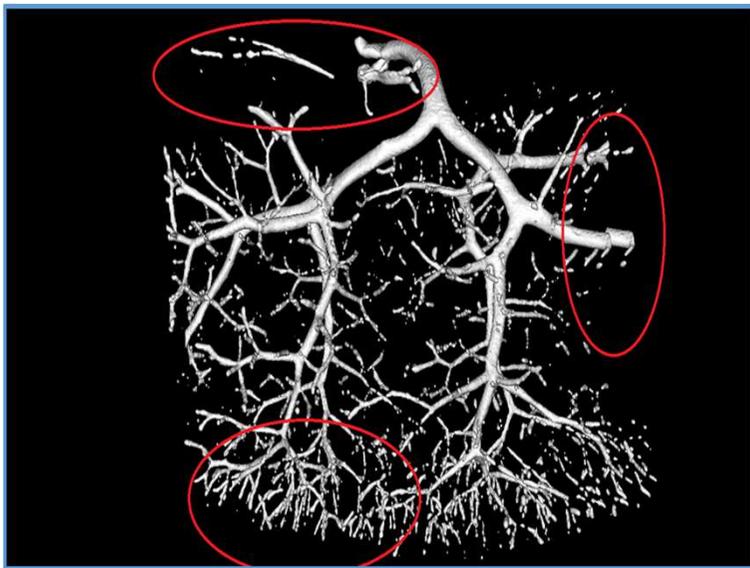
# Segmentation d'images

- Définition :
  - Principal outil de traitement bas-niveau
  - Consiste à créer une partition de l'image en sous-ensembles appelés **régions**
  - Exemples d'application :



# Segmentation d'images

- Définition :
- Exemples d'application :



# Segmentation d'images

- Définition :
    - 2 approches :
      - Approche « région » : fait référence à des groupements de pixels ayant des propriétés communes
      - Approche « frontière » ou « contours » : fait référence à une variation locale du niveau de gris ou de la couleur.
- Une fois les frontières détectées, on procède à une fermeture des contours pour obtenir une partition complète de l'image et donc les régions.

# Segmentation - Approche « région »

- Segmentation par **agrégation de pixels** :
  - Procédure itérative
  - Initialisation : chaque pixel forme une région
  - Itération :
    - on choisit une région et on fait croître cette région tant que des pixels de son voisinage vérifient le critère de similarité
    - lorsqu'il n'y a plus de pixel candidat dans le voisinage, on choisit une nouvelle région et on itère le processus
  - On regroupe les régions avec un double critère de similarité des niveaux de gris et d'adjacence
  - Le critère de similarité peut par exemple être : la variance des niveaux de gris de la région est inférieure à un seuil

# Segmentation - Approche « région »

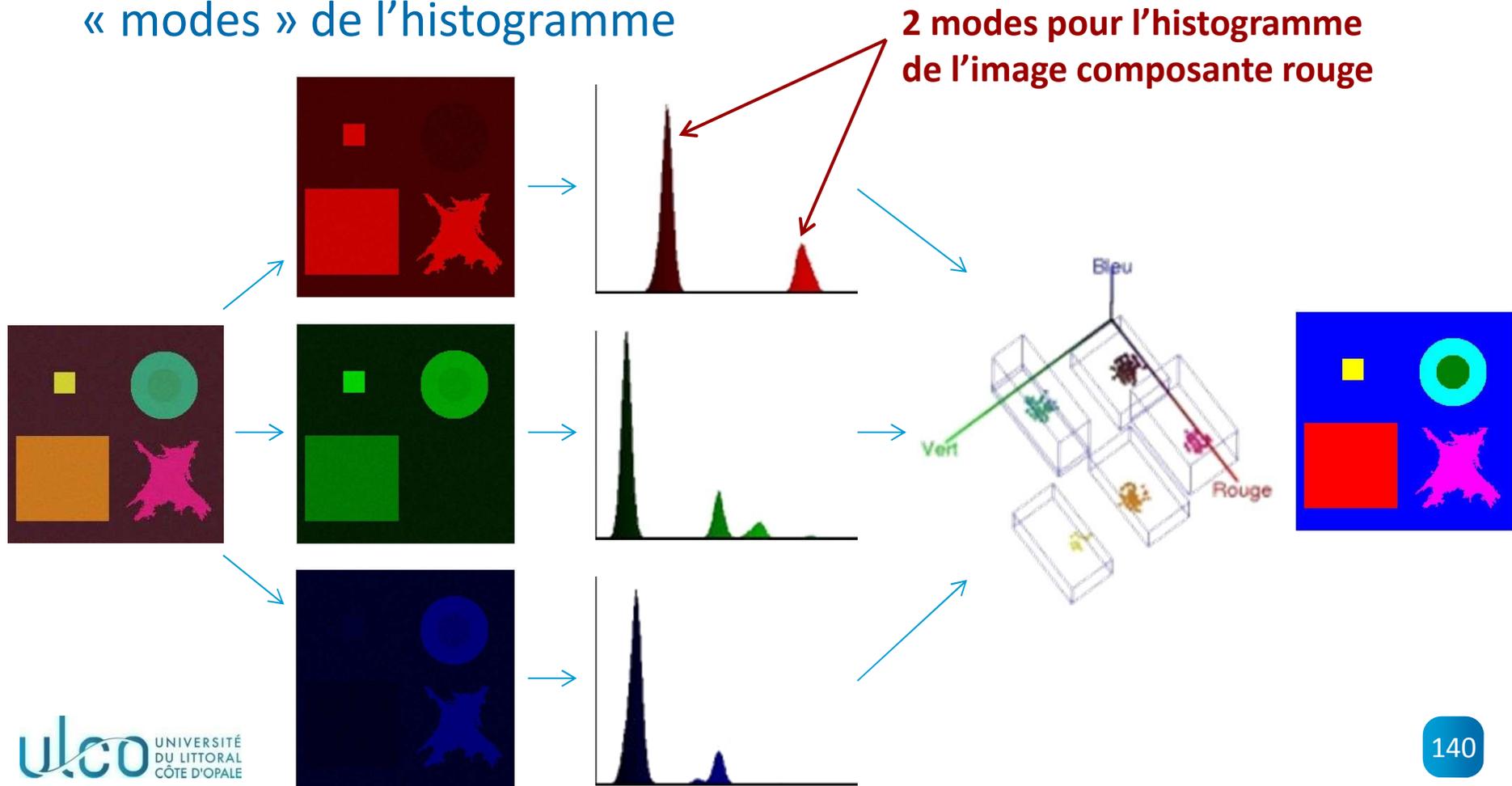
- Segmentation par **agrégation de pixels** :
  - Paramètres :
    - Point de départ
    - Sens de parcours de l'image
    - Connexité du voisinage
    - Ordre d'analyse du voisinage

# Segmentation - Approche « région »

- Segmentation par **division** :
  - Procédure itérative
  - Initialisation : image = une seule région
  - Itération :
    - Si la région n'est pas homogène (critère d'homogénéité), elle est divisée en plusieurs régions, sinon, le processus se termine
    - Chaque région nouvellement créée est potentiellement redivisée en plusieurs régions si elle n'est pas homogène
  - Lors de la division, on peut rajouter un critère d'arrêt sur la taille des régions par exemple
  - La morphologie mathématique peut également être utilisée pour la reconstruction des régions

# Segmentation - Approche « région »

- Exemple de segmentation par division grâce à la détection des « modes » de l'histogramme



# Segmentation - Approche « région »

- Segmentation par **fusion/séparation** :
  - Séparation (arbre quaternaire (quadtree)) :
    - L'image est divisée en 4 si les caractéristiques des pixels sont différentes
    - L'opération est réitérée sur les zones divisées
    - Sous Matlab : qtdemo
  - Séparation et fusion (split and merge) :
    - A la fin de l'étape de division, les régions voisines ayant des caractéristiques communes sont fusionnées

# Segmentation - Approche « région »

- Segmentation par **fusion/séparation** :
  - Paramètres :
    - Point de départ
    - Sens de parcours de l'image
    - Connexité du voisinage
    - Ordre d'analyse du voisinage

# Segmentation - Approche « frontière »

- Approche « frontière » :
  - Elimination éventuelle du bruit
  - Filtrage :
    - Estimation de la dérivée première :
      - Gradient
      - Sobel
      - Prewitt
      - Roberts
    - Estimation de la dérivée seconde :
      - Laplacien
  - Fermeture de contours
  - Suivi et localisation des contours

# Segmentation - Approche « frontière »

- Filtrage :
  - Gradient

0	0	0
-1	0	1
0	0	0

Gradient horizontal

0	-1	0
0	0	0
0	1	0

Gradient vertical

-1	0	0
0	0	0
0	0	1

Gradient 1<sup>ère</sup> diagonale

0	0	1
0	0	0
-1	0	0

Gradient 2<sup>ème</sup> diagonale

# Segmentation - Approche « frontière »

- Filtrage :
  - Sobel

-1	0	1
-2	0	2
-1	0	1

Sobel horizontal

-1	-2	-1
0	0	0
1	2	1

Sobel vertical

-2	-1	0
-1	0	1
0	1	2

Sobel 1<sup>ère</sup> diagonale

0	1	2
-1	0	1
-2	-1	0

Sobel 2<sup>ème</sup> diagonale

# Segmentation - Approche « frontière »

- Filtrage :
  - Prewitt

-1	0	1
-1	0	1
-1	0	1

Prewitt horizontal

-1	-1	-1
0	0	0
1	1	1

Prewitt vertical

-1	-1	0
-1	0	1
0	1	1

Prewitt 1<sup>ère</sup> diagonale

0	1	1
-1	0	1
-1	-1	0

Prewitt 2<sup>ème</sup> diagonale

# Segmentation - Approche « frontière »

- Filtrage :
  - Roberts
    - Application de 2 masques :

0	0	0
0	1	0
0	0	-1

0	0	0
0	0	1
0	-1	0

- $Roberts[x][y] = \text{abs}(I[x][y]-I[x+1][y+1]) + \text{abs}(I[x+1][y]-I[x][y+1])$

# Segmentation - Approche « frontière »

- Filtrage :
  - Laplacien

0	0	0
1	-2	1
0	0	0

Dérivée seconde  
horizontale

0	1	0
0	-2	0
0	1	0

Dérivée seconde  
verticale

# Segmentation - Approche « frontière »

- Filtrage :
  - Estimation de la dérivée première :
    - Contour = maximum de la dérivée première
  - Estimation de la dérivée seconde :
    - Contour = passage par zéro de la dérivée seconde

# Traitement haut-niveau

- > Classification de pixels (segmentation)
- > Classification d'images
- > Recherche par le contenu
- > Reconnaissance de formes
- > ...

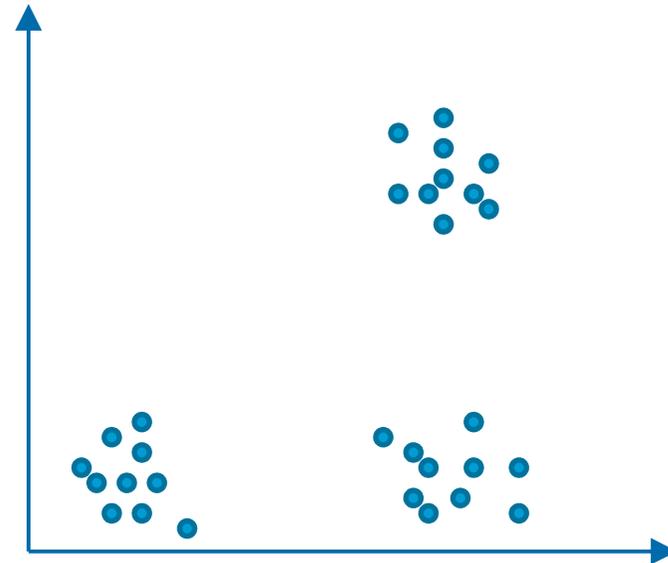
**=> CLASSIFICATION DE DONNEES**

DONNEES = pixels, images, caractéristiques, objet, ...

# Classification

- Classification de données :

- Données à classer

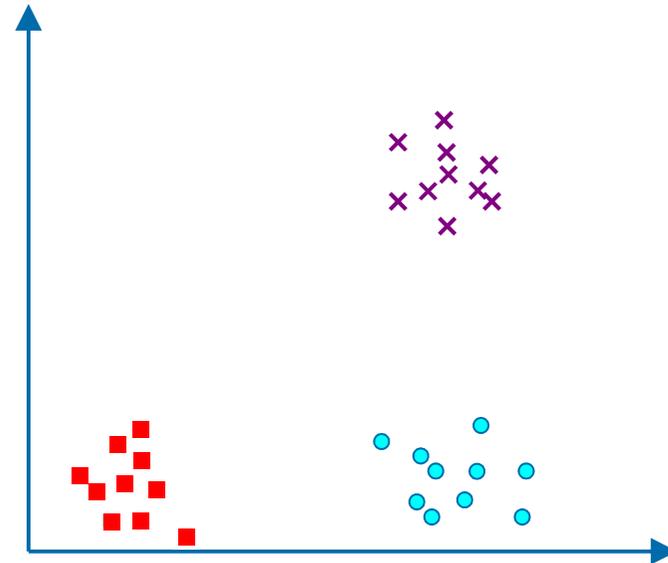


Exemple : représentation des données dans un espace de dimension 2

# Classification

- Classification de données :

- Classe 1
- × Classe 2
- Classe 3



Exemple : représentation des données  
dans un espace de dimension 2

# Classification



- Classification de données :
  - Définition : partitionner un ensemble de données en groupes ou classes, grâce à l'exploitation de leur similarité
  - 3 approches :
    - Supervisée : caractérisation des différentes classes entièrement basée sur des prototypes, qui sont des échantillons de classe connue
    - Non supervisée : aucune connaissance a priori sur les classes
    - Semi supervisée : connaissance partielle

# Classification



- Classification de données :
  - Etapes :
    - Représentation des données : extraction de caractéristiques (également appelées attributs) + éventuellement sélection des plus pertinentes
    - Définition d'une mesure de similarité
    - Classification
    - Eventuellement feedback sur la représentation et/ou la mesure de similarité

# Attributs

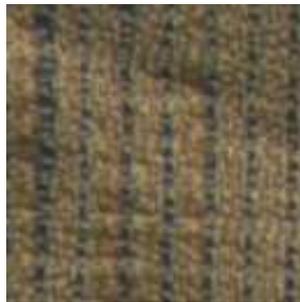
- Attributs :
  - Quantitatif :
    - Valeurs continues (ex : périmètre ou surface d'une région)
    - Valeurs discrètes (ex : niveau de gris)
    - Intervalles de valeurs
  - Qualitatif (ex : nom de couleur)

# Attributs

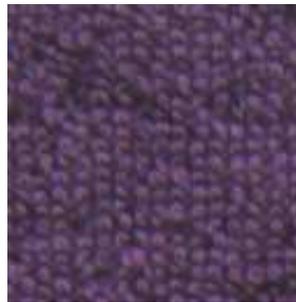
- Attributs propres au pixel :
  - Obtenus à partir des niveaux de gris ou de composantes couleur
  - Statistiques d'images :
    - Moyenne des niveaux,
    - Variance ou Ecart-type des niveaux,
    - ...
  - Histogramme d'image 1D ou 3D
  - Statistiques d'histogramme :
    - Médiane : valeur pour laquelle l'histogramme est partitionné en deux parties d'égales populations,
    - Mode : correspond au maximum de l'histogramme,
    - ...

# Attributs

- Attributs propres au pixel :
  - Cas où l'information provenant des niveaux de composantes couleur suffit pour discriminer les classes :
    - Exemple sur la classification d'images



Exemple d'image  
de la classe 1



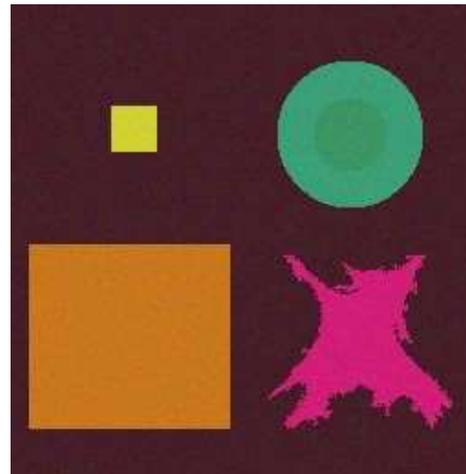
Exemple d'image  
de la classe 2



Exemple d'image  
de la classe 3

# Attributs

- Attributs propres au pixel :
  - Cas où l'information provenant des niveaux de composantes couleur suffit pour discriminer les classes :
    - Exemple sur la classification de pixels (segmentation)



# Attributs

- Attributs propres au pixel :
  - Cas où l'information provenant des niveaux de composantes couleur ne suffit pas pour discriminer les classes :
    - Exemple sur la classification d'images



Exemple  
d'image de  
la classe 1



Exemple  
d'image de  
la classe 2



Exemple  
d'image de  
la classe 3



Exemple  
d'image de  
la classe 4



Exemple  
d'image de  
la classe 5

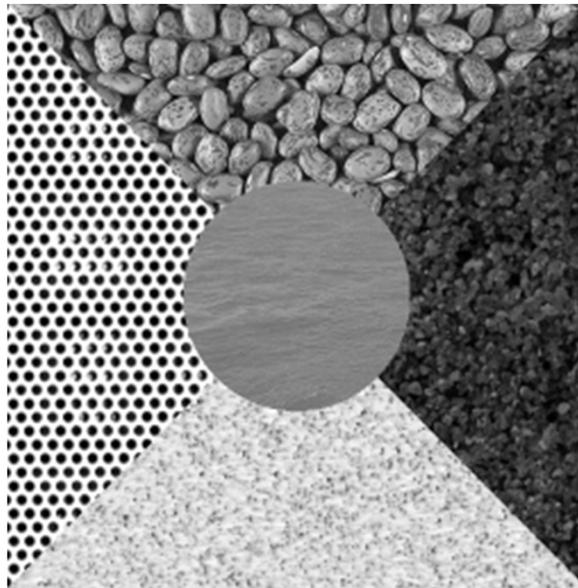


Exemple  
d'image de  
la classe 6

-> L'information couleur ne suffit pas. Nécessité de  
quantifier les relations spatiales entre les pixels

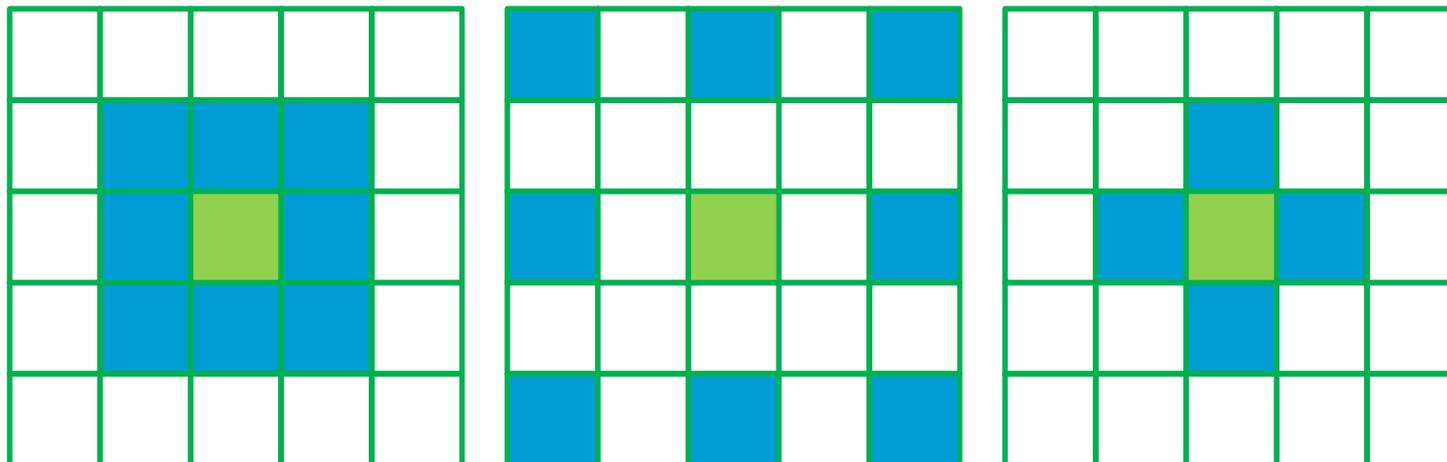
# Attributs

- Attributs propres au pixel :
  - Cas où l'information provenant des niveaux de gris ou de composantes couleur ne suffit pas pour discriminer les classes :
    - Exemple sur la classification de pixels (segmentation)



# Attributs

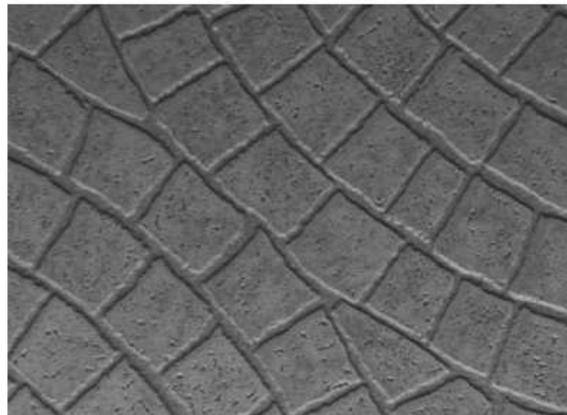
- Attributs propres au pixel
- Attributs du pixel et de son **voisinage** 
  - Prise en compte de l'information des niveaux de gris ou de la couleur + **l'interaction spatiale entre les pixels**
  - Utile pour classer ou segmenter des images contenant des **textures**



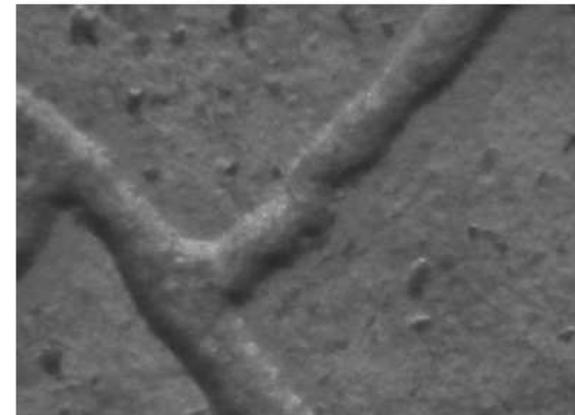
Exemples de voisinage

# Attributs

- Attributs du pixel et de son **voisinage**
- Définition de la texture :
  - Agencement spatial plus ou moins régulier d'éléments qui constituent un ensemble
  - 2 points de vue :



(a) Observation macroscopique d'une texture (image de taille  $746 \times 538$  extraite de la base OuTex).



(b) Observation microscopique d'une texture (image de taille  $746 \times 538$  extraite de la base OuTex).

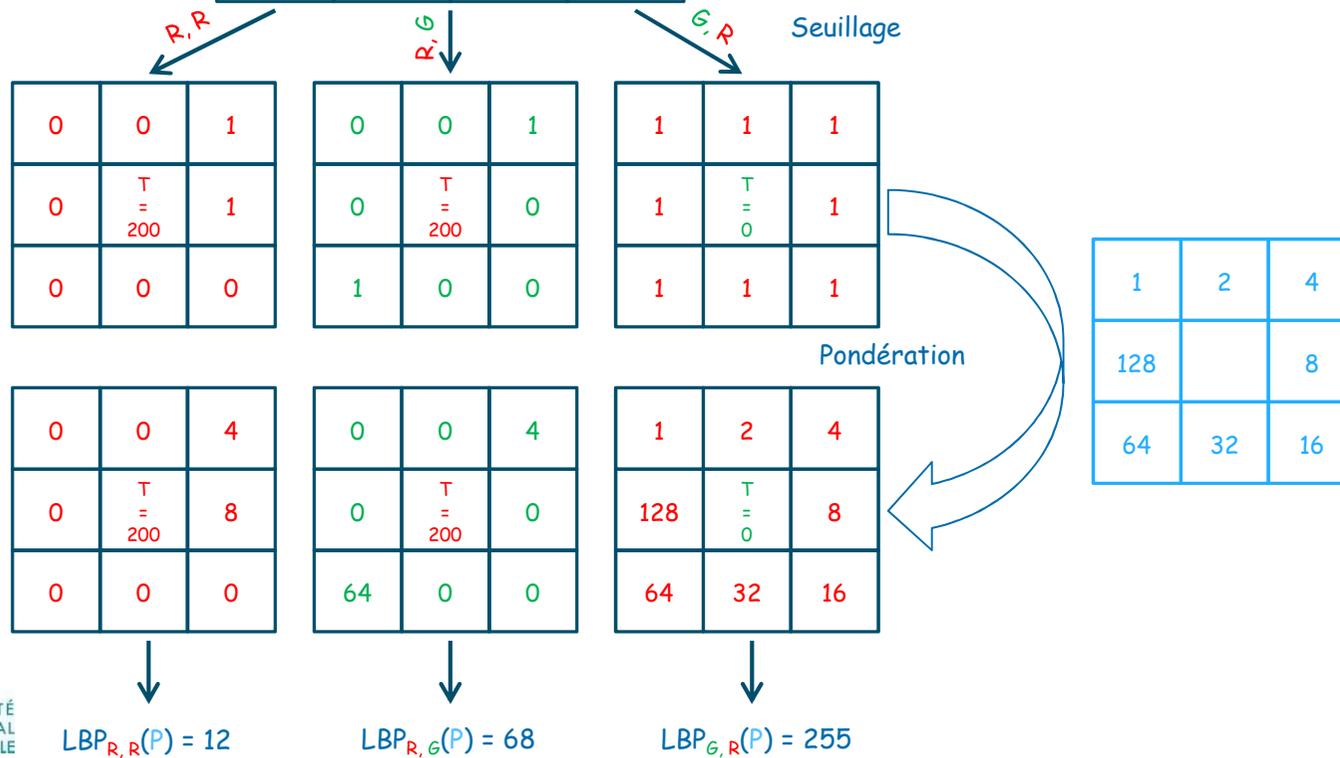
# Attributs

- Attributs du pixel et de son **voisinage** :
  - Il existe une multitude d'attributs permettant de caractériser les relations spatiales entre les pixels et donc la texture
  - Exemple 1 : les motifs locaux binaires (LBP : Local Binary Patterns)
    - > [Vidéo - Applications LBP](#)

# Attributs

Image couleur I codée dans l'espace (R, G, B)

0	100	255	200
0	0	200	50
100	200	0	100
100	200	200	50
100	0	100	50
0	0	0	50
150	0	0	250
200	0	0	100
200	0	200	100
200	255	0	100
0	255	200	255
150	255	50	200



- Motifs locaux binaires

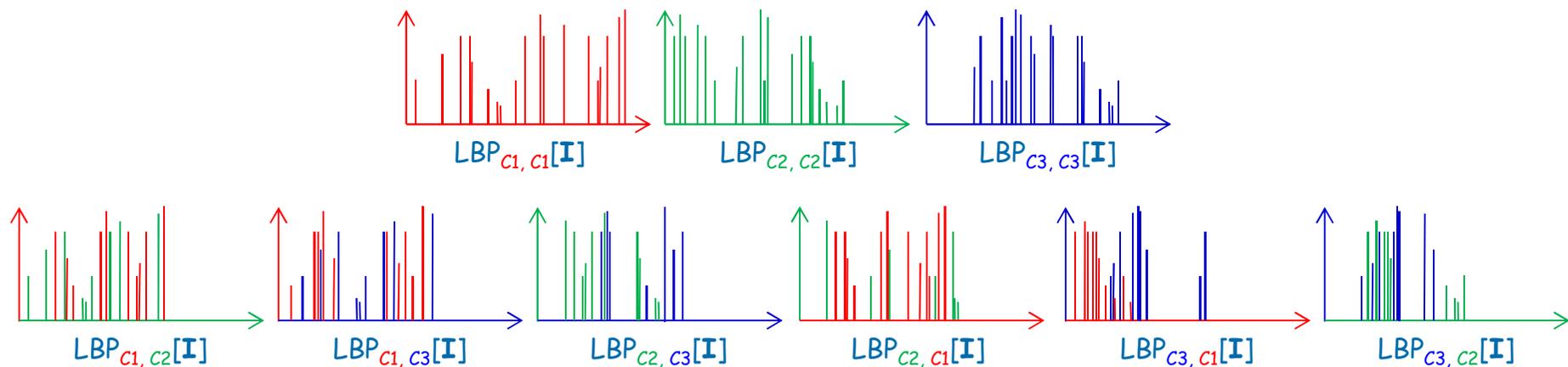
- Pour caractériser chaque pixel  $P$  (et son voisinage) d'une image couleur  $I$ , 9 LBP sont donc nécessaires :
  - 3 LBP caractérisant les relations intra-composantes :  $LBP_{R,R}(P)$ ,  $LBP_{G,G}(P)$ ,  $LBP_{B,B}(P)$ ,
  - 6 LBP caractérisant les relations inter-composantes :  $LBP_{R,G}(P)$ ,  $LBP_{G,R}(P)$ ,  $LBP_{R,B}(P)$ ,  $LBP_{B,R}(P)$ ,  $LBP_{G,B}(P)$ ,  $LBP_{B,G}(P)$ .

# Attributs

- Motifs locaux binaires

- Pour caractériser la texture complète à analyser :

- Les LBP de chaque pixel de l'image  $I$  sont calculés,
- Les distributions correspondant aux 9 LBP sont présentées dans 9 histogrammes différents

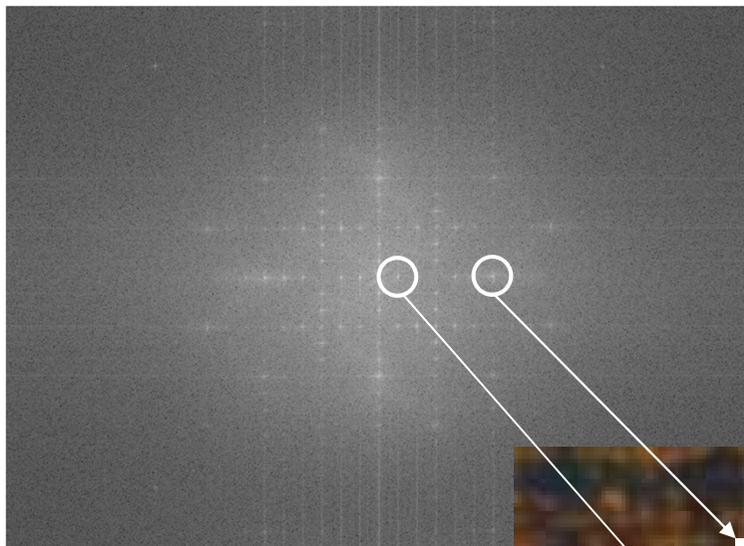


# Attributs

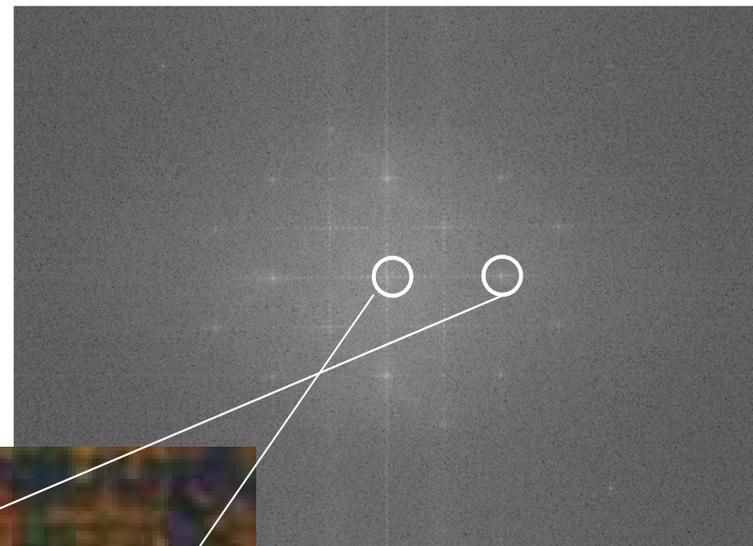
- Attributs du pixel et de son **voisinage** :
  - Exemple 2 : la transformée de Fourier
    - Fournit une représentation de la texture dans le domaine des fréquences
    - Hautes fréquences : caractérisent des changements locaux de niveaux de gris ou de composantes couleur
    - Basses fréquences : caractérisent les grandes continuités entre les valeurs de chaque composante couleur des pixels (textures grossières)
    - Un ou plusieurs attributs statistiques (comme l'énergie, la variance, ...) sont ensuite calculés à partir des images filtrées

# Attributs

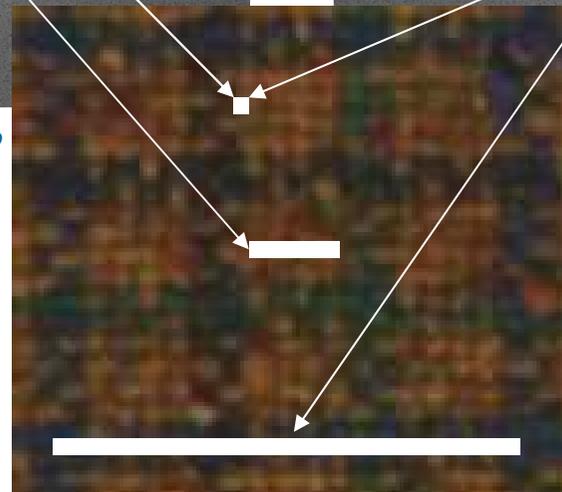
- Transformée de Fourier



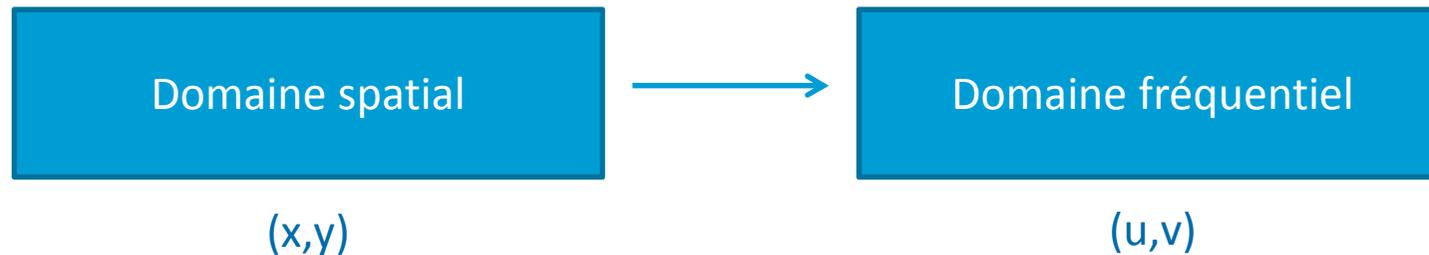
*Spectre de la composante R*



*Spectre de la composante B*



- Transformée de Fourier



- Soit  $I$  une image de taille  $X \times Y$ , sa transformée de Fourier est donnée par :

$$F(u, v) = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I(x, y) \exp^{-j2\pi \left( \frac{ux}{X} + \frac{vy}{Y} \right)}$$

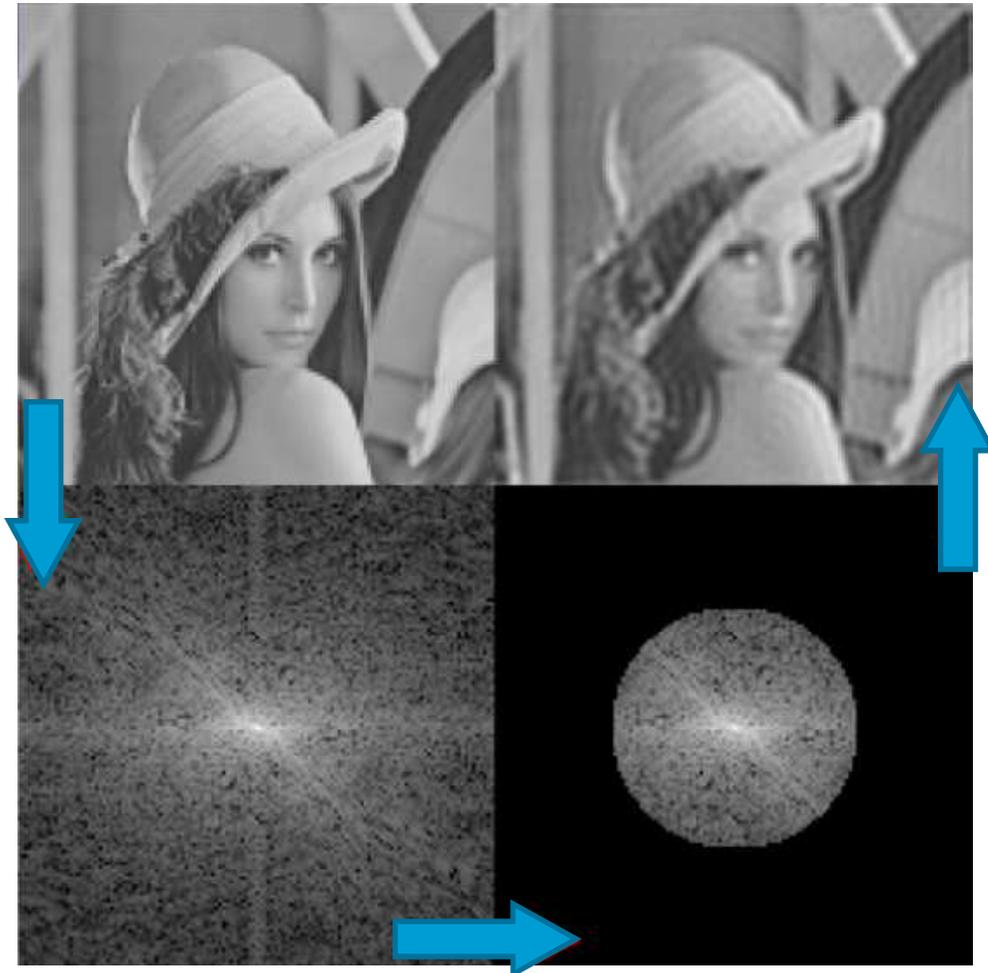
# Attributs



- Transformée de Fourier
  - Domaines d'application très variés :
    - Classification
    - Restauration
    - Compression
    - Filtrage

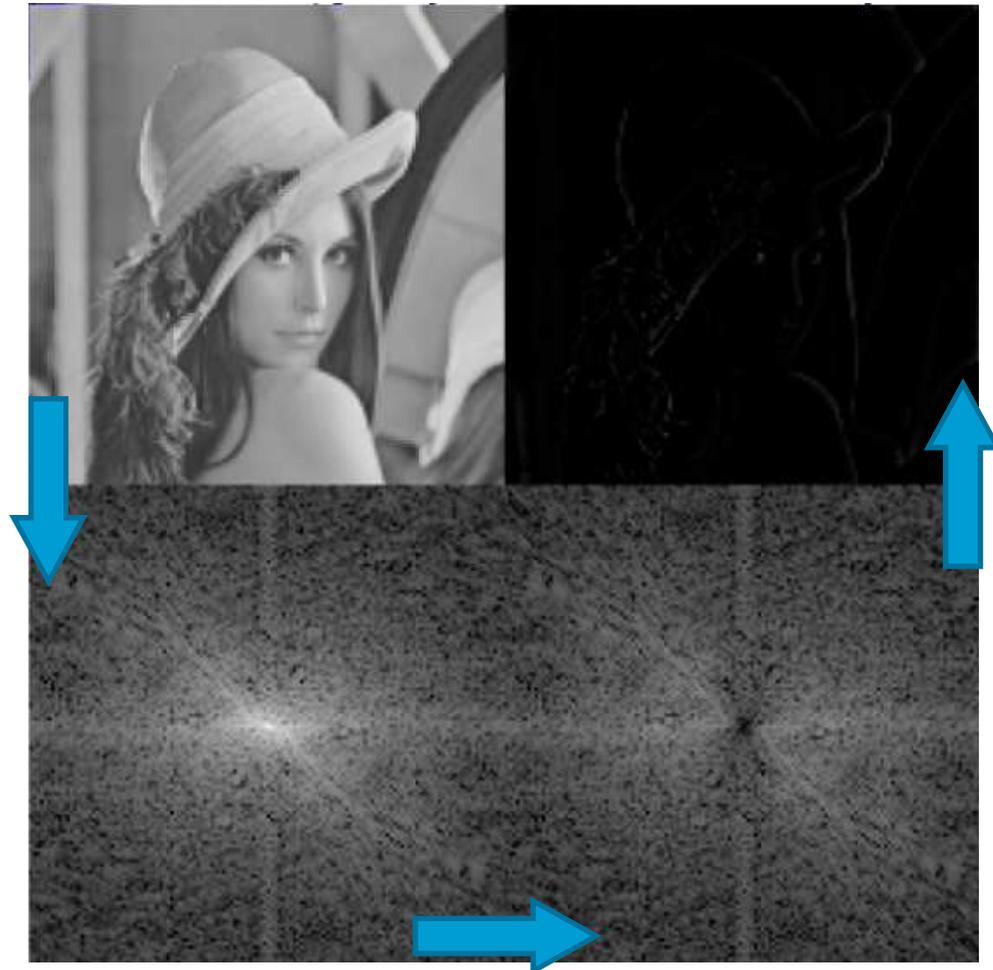
# Attributs

- Transformée de Fourier
  - Filtrage passe-bas :



# Attributs

- Transformée de Fourier
  - Filtrage passe-haut :





# Attributs



- Attributs de reconnaissance de forme :
  - Obtenus à partir d'une image étiquetée
    - Grandeurs géométriques
    - Indices de forme
    - Signatures du contour
    - Descripteurs de Fourier
    - Cavités
  - Obtenus à partir de l'image en niveau de gris et de son image étiquetée correspondante
    - Moments

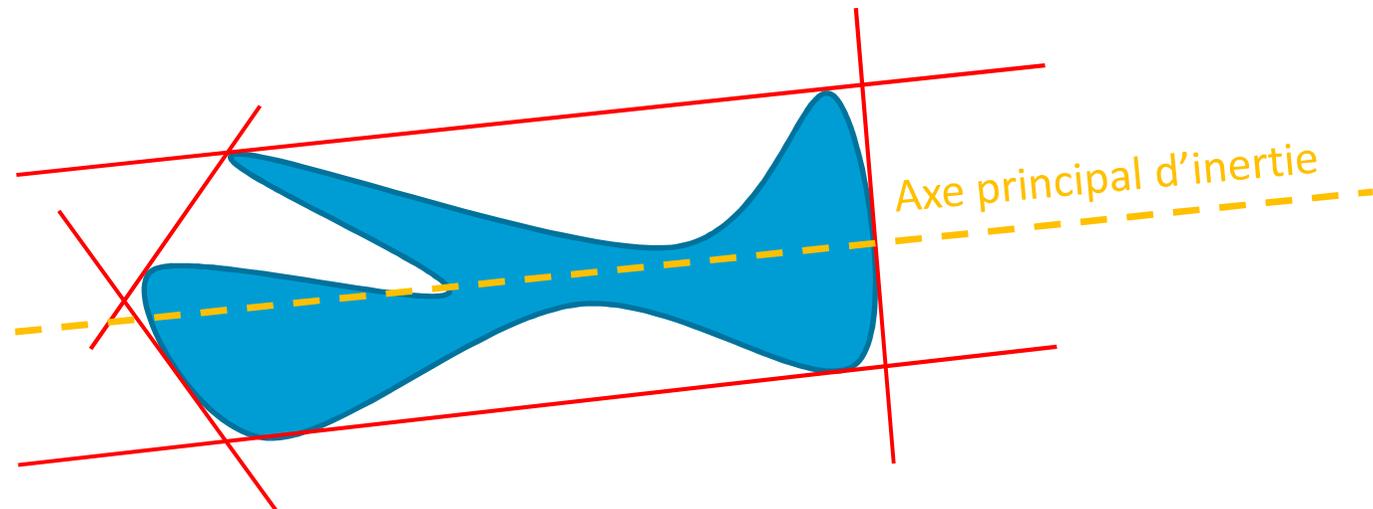
# Attributs

- Attributs de reconnaissance de forme :
  - Grandeurs géométriques :
    - Centre / barycentre de la forme
    - Aire : nombre de pixels de la forme
    - Périmètre : nombre de pixels de la forme voisins d'un pixel du fond
    - Diamètre : plus grande distance entre deux points de la forme (distance euclidienne ou géodésique)



# Attributs

- Attributs de reconnaissance de forme :
  - Grandeurs géométriques :
    - Enveloppe convexe : intersection de tous les demi-plans contenant la forme
    - Orientation



# Attributs

- Attributs de reconnaissance de forme :
  - Indices de forme :

- Rapport isopérimétrique :  $r = \frac{\text{périmètre}^2}{4\pi \times \text{aire}}$

- Concavité :  $c = \frac{\text{périmètre de l'enveloppe convexe}}{\text{périmètre de la forme}}$

# Attributs

- Attributs de reconnaissance de forme :
  - Moments :

- Moment d'ordre 0 : 
$$M_{00} = \sum_x \sum_y F_I(x, y)$$

- Moments d'ordres supérieurs : 
$$M_{ij} = \sum_x \sum_y x^i y^j F_I(x, y)$$

- Ces formules s'appliquent sur les pixels composant chaque forme

# Attributs

- Attributs de reconnaissance de forme :
  - Moments :

- Moments centrés : 
$$\mu_{ij} = \sum_x \sum_y (x - \bar{x})^i (y - \bar{y})^j F_I(x, y)$$

Attributs invariants à la translation

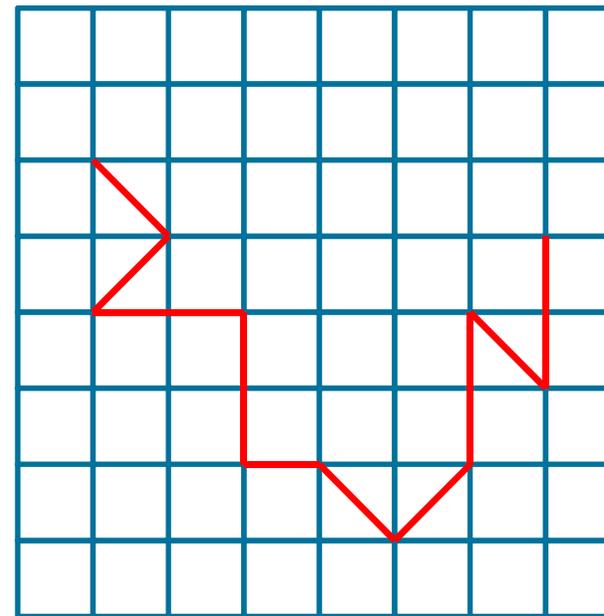
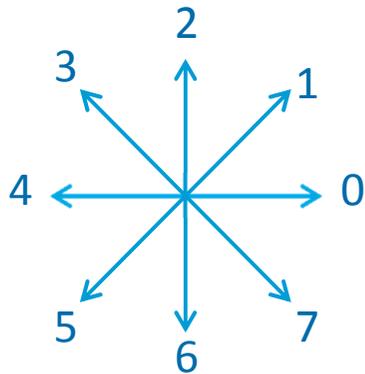
- Moments centrés normalisés : 
$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{1+(i+j)/2}}$$

Attributs invariants au changement d'échelle

- Moments de Hu : invariants en translation, au changement d'échelle et à la rotation

# Attributs

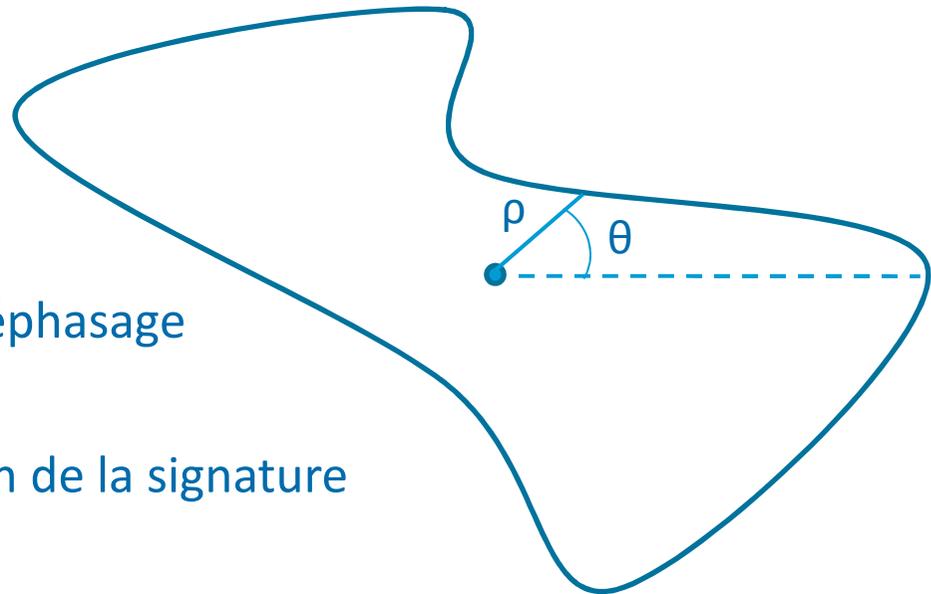
- Attributs de reconnaissance de forme :
  - Signature du contour :
    - Codage de Freeman
    - Propriétés :
      - Invariant par translation
      - Rotation = addition



**75006607122722**

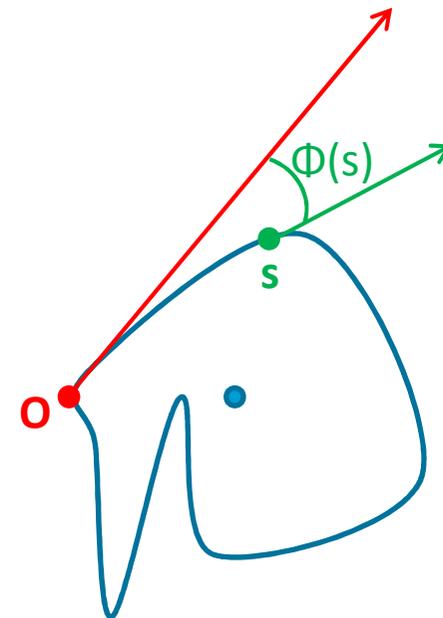
# Attributs

- Attributs de reconnaissance de forme :
  - Signature du contour par coordonnées polaires :
    - Déterminer la fonction  $\rho(\theta)$
    - Propriétés :
      - Fonction périodique
      - Invariance en translation
      - Rotation de la forme = déphasage de sa signature
      - Dilatation = multiplication de la signature par le coefficient
    - Inconvénient : nécessite que la forme soit convexe



# Attributs

- Attributs de reconnaissance de forme :
  - Signature du contour par abscisse curviligne :
    - On ne mesure plus le rayon, mais l'angle entre la tangente en  $s$  et la droite issue du point origine
    - Déterminer la fonction  $\Phi(\theta)$
    - $\Phi(\theta) = \Phi(s) - \theta$   
où  $s = P\theta/2\pi$   
et  $P$  est le périmètre de la forme
    - Propriétés : identiques à la signature du contour par coordonnées polaires
    - Peut être appliquée aux formes concaves et aux contours non fermés



# Attributs

- Attributs de reconnaissance de forme :
  - Descripteurs de Fourier :
    - Signature du contour par abscisse curviligne = fonction périodique
    - Décomposition en série de Fourier :  
permet de coder la signature de manière plus compacte
    - Coefficients de Fourier = attributs de reconnaissance de forme
    - Comparaison des coefficients jusqu'à une valeur fixant le niveau de détails

# Attributs

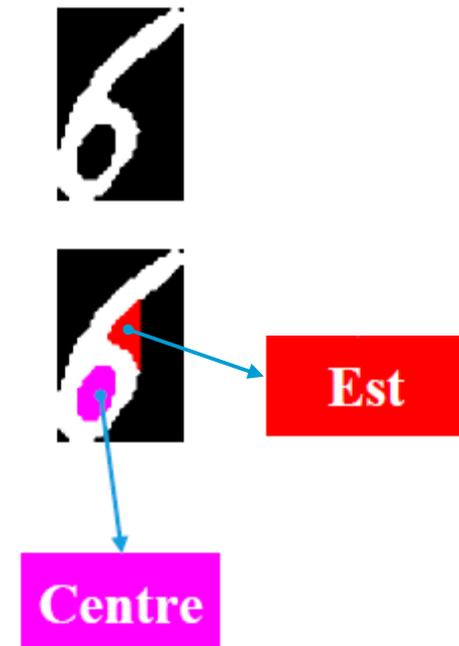
- Attributs de reconnaissance de forme :
  - Cavités :

- Objectif :

Extraire les cavités nord, sud, est, ouest et centre d'une forme

- Définition :

Un pixel P appartient à une cavité de direction  $\theta$  (est, ouest, sud, nord ou centre) si P n'appartient pas au tracé et si, en partant de P, on ne rencontre pas le tracé dans la direction  $\theta$  mais dans les autres directions



# Attributs

- Attributs de reconnaissance de forme :
  - Cavités :
    - Calcul des attributs :

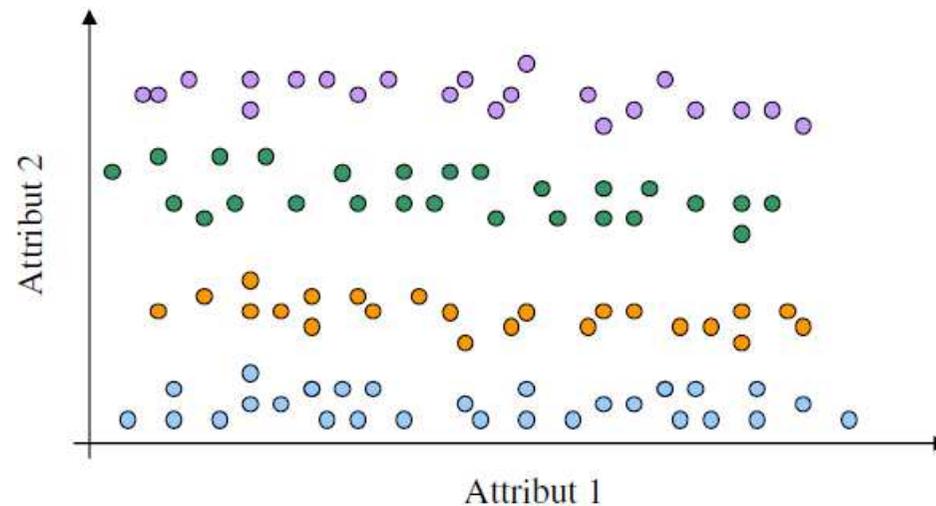
$$\theta_{\%} = \frac{\Sigma_{\theta}}{\Sigma_{Centre} + \Sigma_{Nord} + \Sigma_{Sud} + \Sigma_{Est} + \Sigma_{Ouest}}$$

- Vecteurs d'attributs :

$$V = [\theta_{Est} \theta_{Ouest} \theta_{Sud} \theta_{Nord} \theta_{Centre}]$$

# Sélection

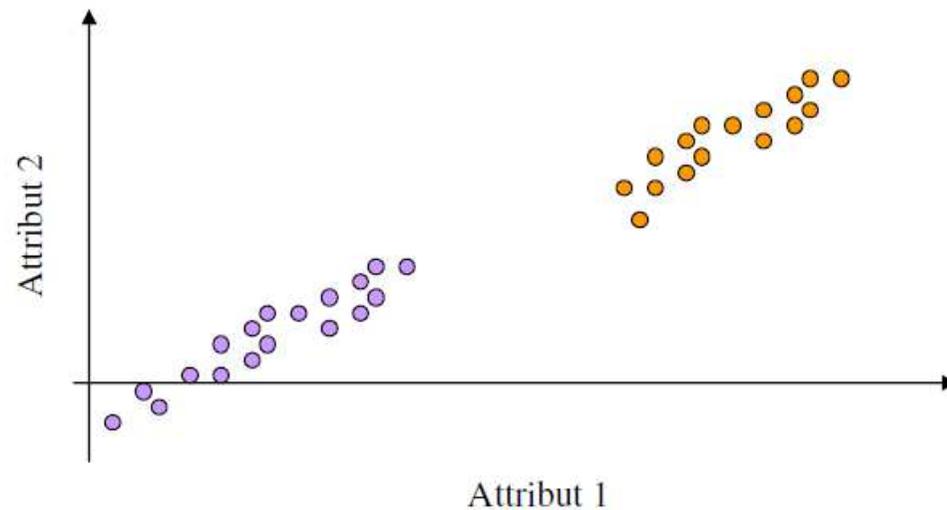
- Classification de données :
  - Sélection :
    - Intérêt de la sélection :



- Point-image appartenant à la classe 1
- Point-image appartenant à la classe 2
- Point-image appartenant à la classe 3
- Point-image appartenant à la classe 4

# Sélection

- Classification de données :
  - Sélection :
    - Intérêt de la sélection :



● Point-image appartenant à la classe 1   ● Point-image appartenant à la classe 2

- Intérêt de la sélection :
  - Caractérisation des images par de nombreux attributs :
    - coût important en stockage mémoire et en temps de calcul
    - augmente le risque de considérer des attributs non pertinents
  - > Mesure de la pertinence des attributs pris individuellement ou d'un sous-ensemble d'attributs (ex : mesure du taux d'erreur de classification sur les prototypes dans le cas supervisé)
  - La prise en compte d'attributs corrélés par un algorithme de classification majore le poids de ces derniers lors de la décision
  - > Mesure de la corrélation entre attributs

# Mesures de similarité

- Classification de données :
  - Mesures de similarité :
    - Exemple 1 : distance de Minkowski
      - Soit deux images I et J caractérisées par un unique attribut x, la similarité entre I et J est donnée par :

$$S = \sqrt[p]{|x_I - x_J|^p}$$

- Distance également appelée :
  - Distance de Manhattan quand p=1
  - Distance euclidienne quand p=2

# Mesures de similarité

- Mesures de similarité :
  - Exemple 1 : distance de Minkowski
    - Soit deux images I et J caractérisées par un ensemble attributs  $x_1, \dots, x_n$ , la similarité entre I et J est donnée par :

$$S = \sqrt[p]{\sum_{k=1}^n |x_{k,I} - x_{k,J}|^p}$$

- Plus la distance est proche de 0, plus les images sont similaires

# Mesures de similarité

- Mesures de similarité :
  - Exemple 2 : intersection d'histogrammes
    - Soit deux images I et J caractérisées par un histogramme H, la similarité entre I et J est donnée par :

$$S = \sum_{b=0}^{N-1} \min(H_I(b), H_J(b))$$

Avec N, le niveau de quantification (en général N=256)

- Plus la distance est grande, plus les images sont similaires

# Mesures de similarité

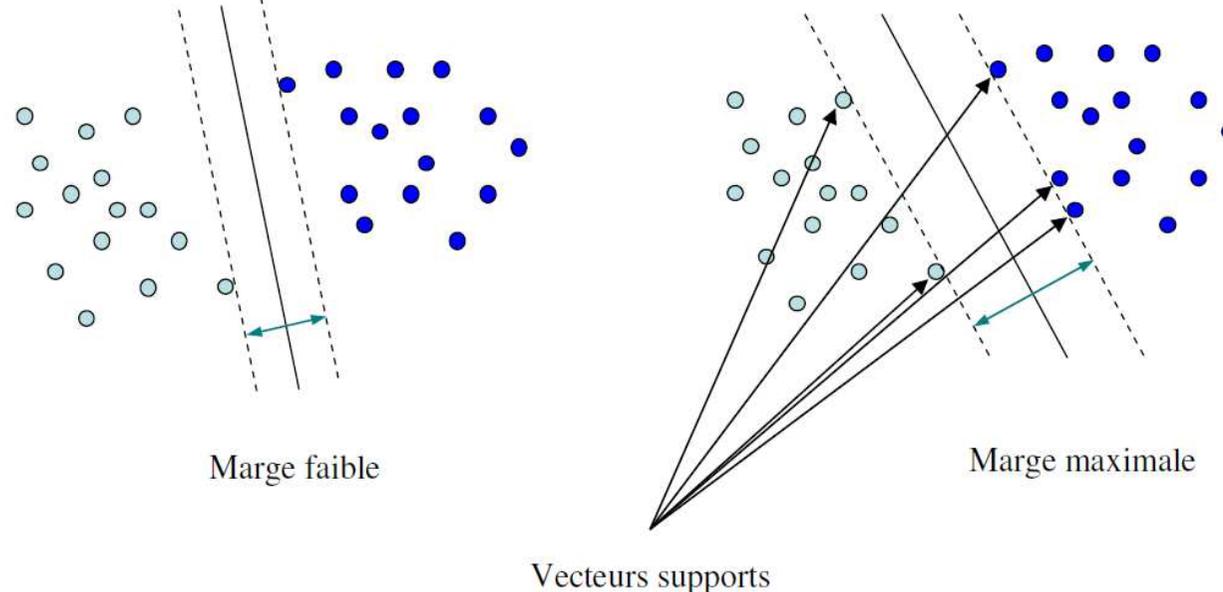
- Mesures de similarité :
  - Exemple 3 : corrélation

$$S = \frac{1}{\sigma_I \sigma_J} \sum_{x=1}^X \sum_{y=1}^Y (I(x, y) - \mu_I)(J(x, y) - \mu_J)$$

$$\mu_I = \frac{\sum_{x=1}^X \sum_{y=1}^Y I(x, y)}{XY} \quad \sigma_I^2 = \sum_{x=1}^X \sum_{y=1}^Y (I(x, y) - \mu_I)^2$$

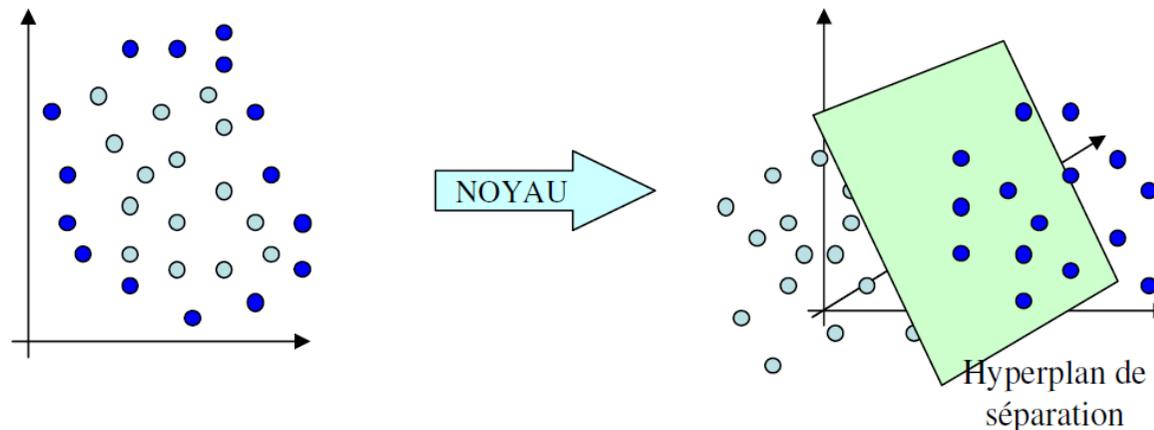
# Classifieurs

- Classification de données :
  - Classifieur :
    - Exemple 1 : machine à vecteurs supports
      - Objectif : déterminer, dans l'espace des attributs, des hyperplans séparant les points-image représentatifs des prototypes de chacune des classes



# Classifieurs

- Classifieur :
  - Exemple 1 : machine à vecteurs supports
    - Lorsque les données ne sont pas linéairement séparables :
      - Transformation de l'espace des attributs en un espace de dimension plus grande dans lequel une frontière séparatrice linéaire existe
      - Cette transformation non linéaire est réalisée via une fonction dite noyau

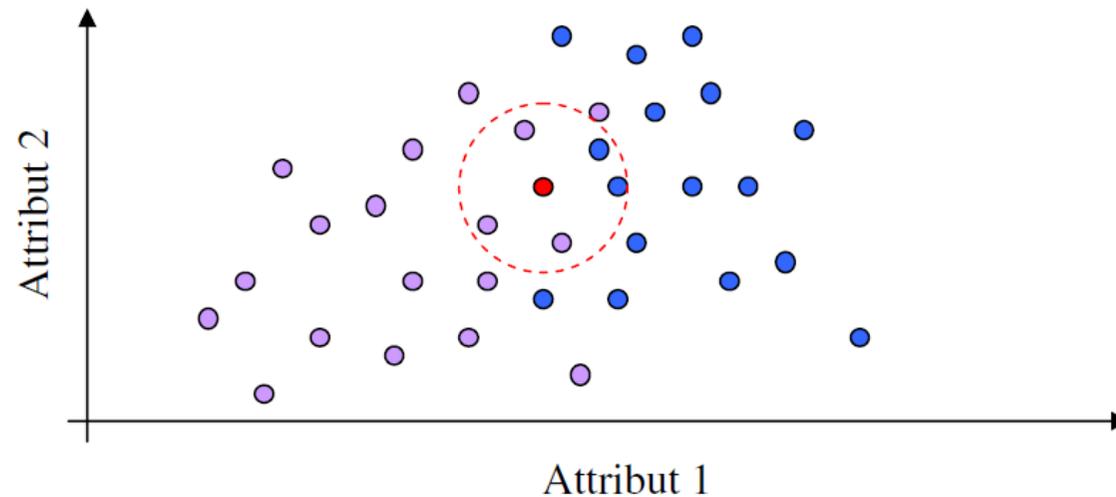


# Classifieurs

- Classifieur :
  - Exemple 1 : machine à vecteurs supports
    - Classification d'une image requête : comparaison la position du point représentatif de l'image avec la position des hyperplans dans l'espace des attributs

# Classifieurs

- Classifieur :
  - Exemple 2 : k-plus proches voisins
    - k=5 :



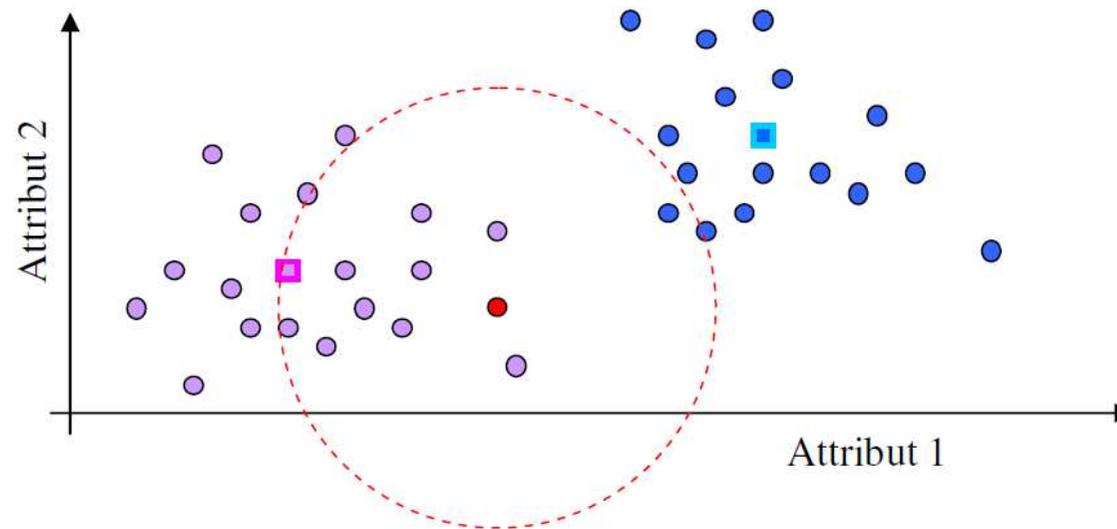
● Point-prototype  
appartenant à la classe 1

● Point-prototype  
appartenant à la classe 2

● Point-image test

# Classifieurs

- Classifieur :
  - Exemple 3 : plus proche barycentre



- Prototype appartenant à la classe 1
- Prototype appartenant à la classe 2
- Barycentre de la classe 1
- Barycentre de la classe 2
- Point-image test