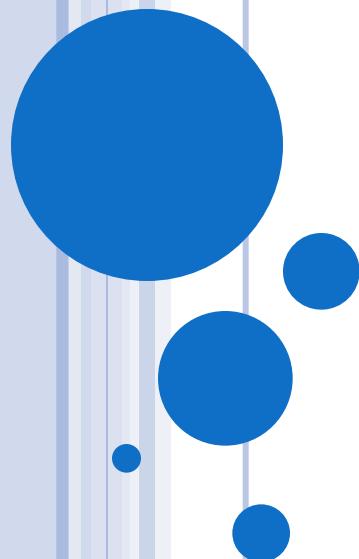


# **TRAITEMENT D'IMAGES & RECONNAISSANCE DE FORMES**



**Asma Ben Abdallah  
Pr. en Informatique  
ING 2 Informatique**

**AU 2025-2026**

## **Objectif du cours :**

- Présenter les concepts de base de l'analyse d'images et la reconnaissance des formes
- Préparer les prérequis pour assimiler les concepts liés à la vision d'ordinateur et à l'apprentissage machine.

# **PLAN DU COURS**

**Partie 1 : Bases du traitement d'image**

**Partie 2 : Amélioration d'image**

**Opérations ponctuelles sur l'image**

Contraste, brillance

Histogramme et traitement à base d'histogramme

Seuillage et inversion

**Opérations locales sur l'image**

Convolution numérique discrète

Filtres passe bas usuels et lissage

Contour et types de contours

Filtres Passe haut et détections de contours

**Partie 3 : Segmentation**

# PARTIE 1 : BASES DU TRAITEMENT D'IMAGES

# I. NOTIONS DE BASE

## Traitement d'images :

- Remplacer l'observateur humain par la machine en vue de traiter et analyser finement les phénomènes enregistrés, les archiver et les illustrer,
- Le but est l'amélioration des images et l'obtention d'une plus grande lisibilité,
- il n'y a pas création d'informations, mais mise en évidence de l'information pertinente déjà présente,
- il facilite les traitements ultérieurs.

# QU'EST CE QU'UNE IMAGE NUMÉRIQUE?

## Image numérique:

- C'est une représentation planaire d'une scène ou d'un objet situé généralement dans un espace 3D.
- Image numérique : acquise, créée, traitée ou stockée sous forme binaire
  - Acquise par suite à une conversion analogique-numérique (scanners, appareils photos, caméscopes numériques, cartes d'acquisition numérisant directement une source comme la télévision).
  - Crée par des outils informatiques, via la souris, les tablettes graphiques, ou par la modélisation 3D (images de synthèse).
  - Traitée par des outils informatiques (peut être transformée, nous pouvons appliquer dessus des filtres variés,...).
  - Sauvegardée sur un support informatique (magnétique, magnéto-optique, ...).

# FORMELLEMENT

- Suivant le domaine considéré et les techniques utilisées pour la traiter, l'image numérique peut être considérée comme :
  1. Un signal bidimensionnel à support et à valeurs bornés noté  $A[i,j]$  avec  $[i,j] \in NxM$  et  $0 \leq i \leq N-1 ; 0 \leq j \leq M-1$ .
- L'image est le résultat de l'échantillonnage du signal continu  $A(i,j)$ .
- $s=[i,j]$  désigne un site de coordonnées  $[i,j]$  dont la valeur notée  $A[s]$  ou  $A[i,j]$ . On appelle pixel le couple( $s, A[s]$ ).

*Exemple :*

- *Signal bidimensionnel* : une image possède deux dimensions: largeur et hauteur.
- *Signal fini* : une image possède des dimensions finies exemple :  $640x480, 800x600$  points (pixels)...
- *Signal échantillonné* : les pixels d'une image sont régulièrement espacés sur une grille carrée.
- *Valeurs quantifiées* : les valeurs des pixels appartiennent à un intervalle borné connu (par exemple  $[0,Ng]$  où  $Ng$  est la valeur maximale du niveau de gris, par exemple  $[0,255]$ ).

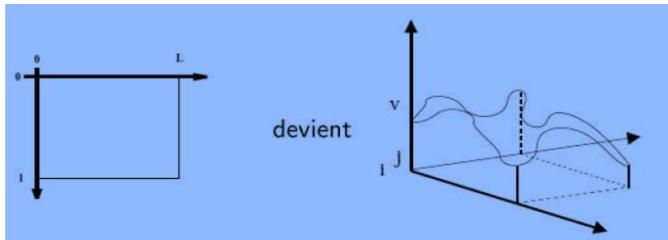
## FORMELLEMENT

L'image numérique peut être considérée aussi comme :

- Un processus stochastique  $A[s]$ :
  - Une **variable aléatoire** est une fonction définie sur l'ensemble des éventualités.
  - Un **processus stochastique** représente une évolution, généralement dans le temps, d'une variable aléatoire.
- Un vecteur aléatoire  $A=(A_1, \dots, A_s, \dots)$  où  $A_s$  est une variable aléatoire associée au site  $s$ .

# FORMELLEMENT

- Une surface  $(i,j, A[i,j])$  de l'espace  $(N^3)$ .



- L'image numérique peut être définie comme une fonction en deux dimensions  $f(x,y)$ .
  - ✓  $x$  et  $y$  : **coordonnées spatiales**
  - ✓ et la **valeur** prise par  $f$  : **niveau de gris ou couleur** .

# ANALYSE D'IMAGES (IMAGE ANALYSIS)

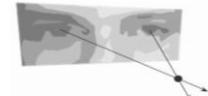
- Techniques de traitement d'images : toutes les techniques ayant pour but la modification des caractéristiques chromatiques des pixels des images bitmap
  - Entrée : image
  - Sortie : image
- Le terme générique *analyse d'images* désigne l'ensemble d'opérations allant : de l'extraction de l'information à partir des images numérique, au traitement, puis à l'interprétation
  - Entrée : image
  - Sortie : information ou description
- Exemple : *les zones d'une image aérienne peuvent être :*
  - *décris par leur forme (carré, linéique, ...)*
  - *étiquetées par les termes : villa, parc, route, ...*

## II. EXEMPLES DE CHAMPS D'APPLICATION

### ■ Imagerie en robotique :

- Robots autonomes
  - Intervention en milieu contaminé
  - Sonde d'exploration autonome
- Robotique médicale
  - Assistance au geste médical

Vision stéréoscopique



Reconnaissance de scène



Robocup



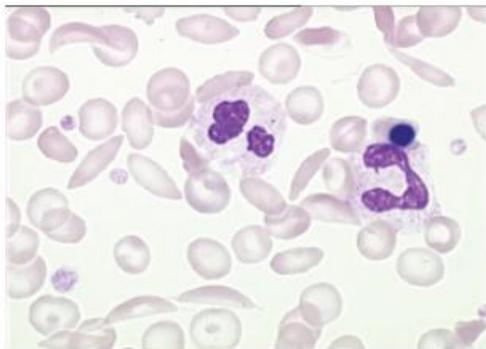
[http://www.tzb.fraunhofer.de/flag/fileadmin/Image\\_Archive/-Tag/Robocup-Web.jpg](http://www.tzb.fraunhofer.de/flag/fileadmin/Image_Archive/-Tag/Robocup-Web.jpg)



<http://zen.ece.ohio.edu/~robocup/robomovies/qualifying-movies-2004.html>

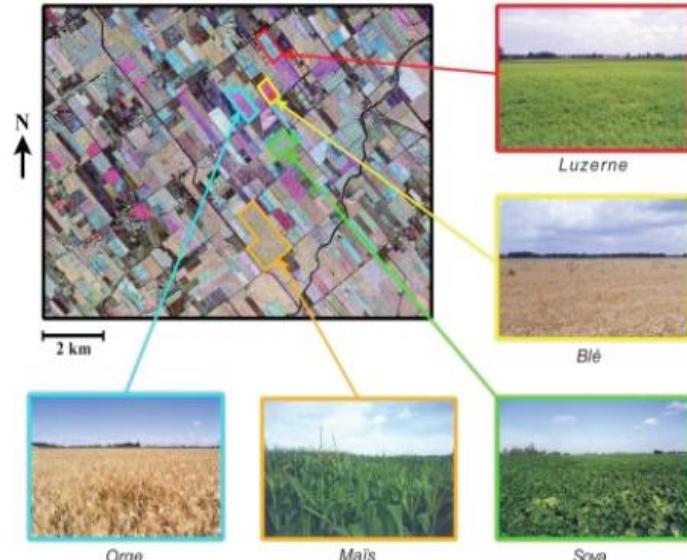
# II. Exemples de champs d'application

- Imagerie biologique :
  - Ex : comptage de cellules



[http://www.med.univ-angers.fr/disciplines/lab\\_bama/morphogweb/20.jpg](http://www.med.univ-angers.fr/disciplines/lab_bama/morphogweb/20.jpg)

- Imagerie aérienne et satellitaire :
  - Analyse des ressources terrestres
  - Météo
  - Cartographie
  - Astronomie

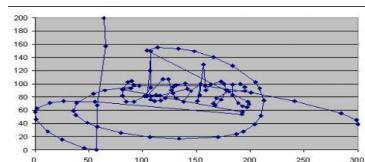
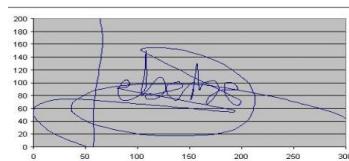


[http://ccrs.nrcan.gc.ca/radarign/crop\\_id/details\\_f.php](http://ccrs.nrcan.gc.ca/radarign/crop_id/details_f.php)

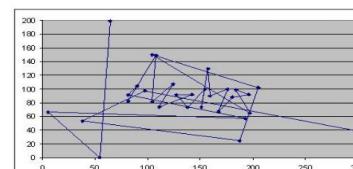
- Biométrie
  - Reconnaissance de visages
  - Reconnaissance d'iris
  - Reconnaissance d'empreintes digitales



[http://www.grinsle.com/public\\_download/grinsle\\_screenshot.jpg](http://www.grinsle.com/public_download/grinsle_screenshot.jpg)



- Reconnaissance de caractères Chèques et Courrier



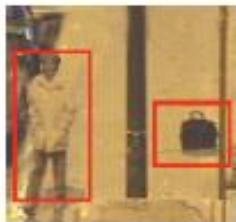
## II. Exemples de champs d'application

### Vidéo surveillance

- Détection de mouvements et reconnaissance de comportements



Détection de mouvement



Détection de bagages abandonné



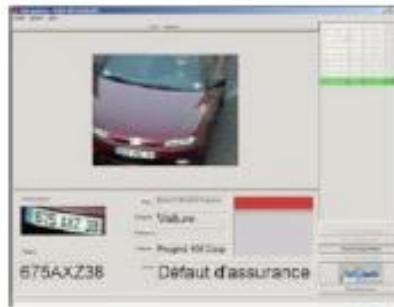
Comptage de personnes

[http://www.lcriq.com/tlproductique\\_tfp/tendances\\_electronique\\_2007-05-08.html](http://www.lcriq.com/tlproductique_tfp/tendances_electronique_2007-05-08.html)



Détection automatique d'incidents

[http://www.ttich-sa.ch/detection\\_automatique\\_dincident.html](http://www.ttich-sa.ch/detection_automatique_dincident.html)



Analyse de plaques d'immatriculation

<http://www.ttich-sa.ch>



M. Dahmane, 2004



Surveillance de piscines

<http://www.poseidon-tech.com>

## II. Exemples de champs d'application

### Reconnaissance de caractères (OCR):

- Traitement de texte
- Tri Postal
- Reconnaissance d'un matricule



### Détection & classification d'objets



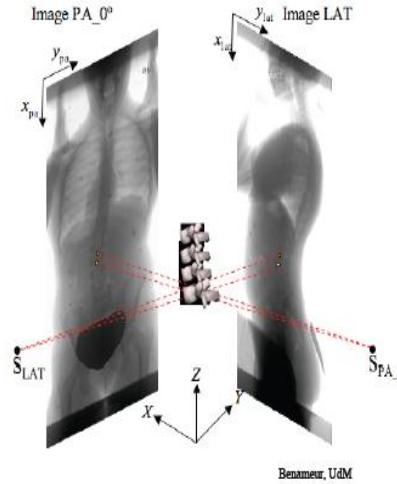
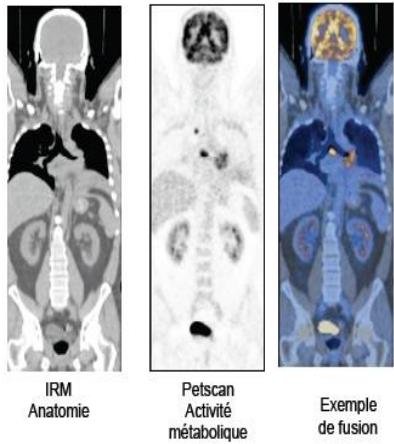
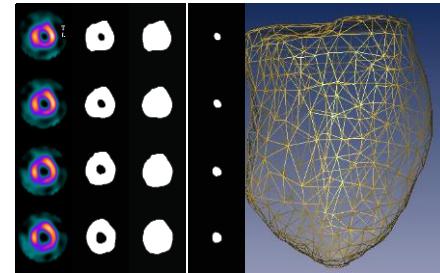
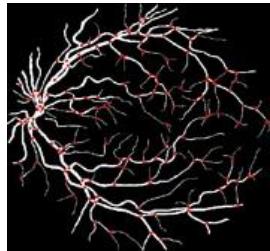
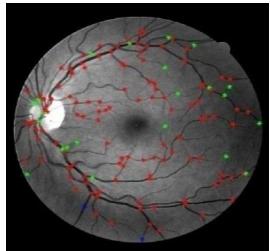
Panneau stop

### Reconnaissance faciale



## II. Exemples de champs d'application

Imagerie médicale : Aide au diagnostic, reconstruction  
3D, ...



Reconstruction 3D



## II. Exemples de champs d'application

- Réalité augmentée  
(analyse:  
traitement+synthèse)



- Infographie
- Logiciels des traitements  
d'images



### Bases de données : BDD image, BDD multimédia

 Similarity: 3.865388 Query Image <input type="button" value="top"/>	 Similarity: 1.078554 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.615849 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.443702 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.400244 <input type="button" value="neutral"/> <input type="button" value="top"/>
 Similarity: 0.379220 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.373110 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.353322 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.297391 <input type="button" value="neutral"/> <input type="button" value="top"/>	 Similarity: 0.269941 <input type="button" value="neutral"/> <input type="button" value="top"/>

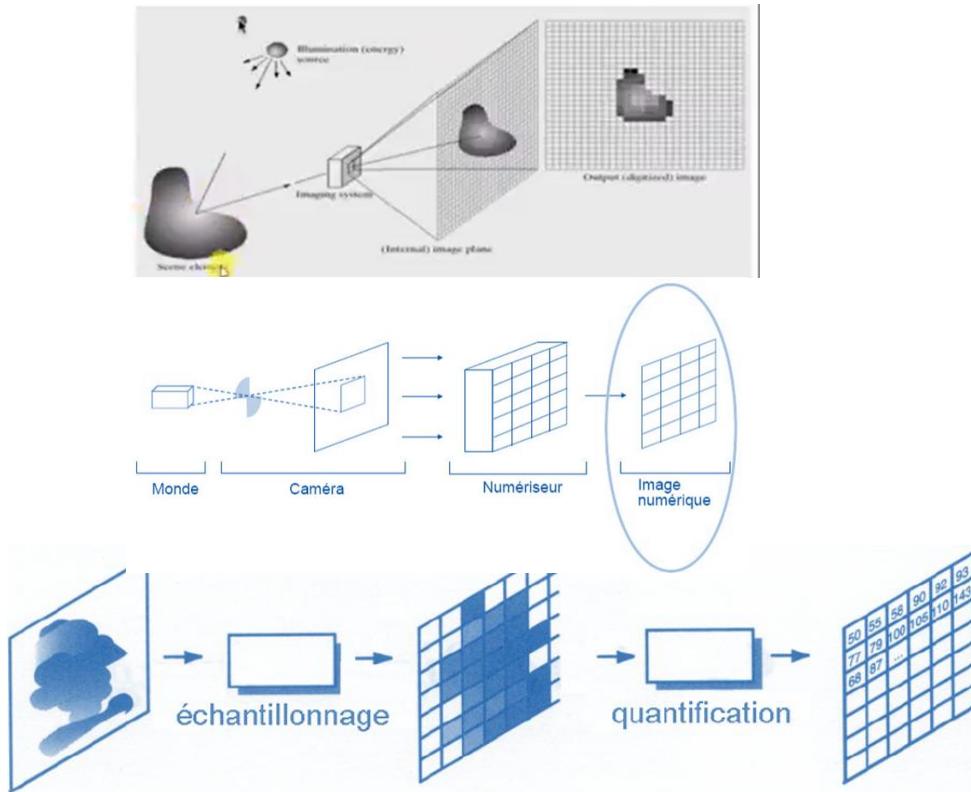
- Véhicules intelligents
  - Aide au conducteur
  - Conduite automatique



### III. NUMÉRISATION D'UNE IMAGE

- Numérisation d'une image: conversion de son état analogique en une image numérique.
- L'image numérique est représentée par un tableau bidimensionnel de valeurs numériques  $I(i,j)$ .  $i, j$  désignent les coordonnées cartésiennes d'un point de l'image et  $I(i, j)$  : désigne son niveau de gris.
- La numérisation d'une image est composée de deux étapes :
  - Numérisation des coordonnées : **Echantillonnage spatial** (sampling).
  - Numérisation des valeurs prises : **Quantification** (quantization).

# III. Numérisation d'une image



En 2D, une image numérique est une image échantillonnée et quantifiée

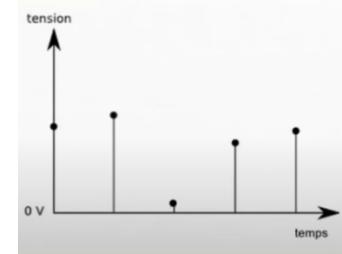
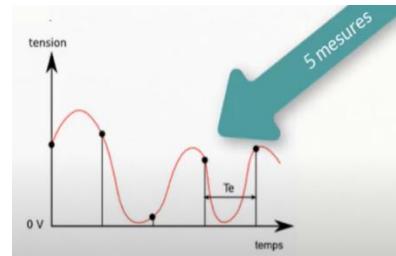
# III. Numérisation d'une image

## Numérisation - Echantillonnage d'un signal :

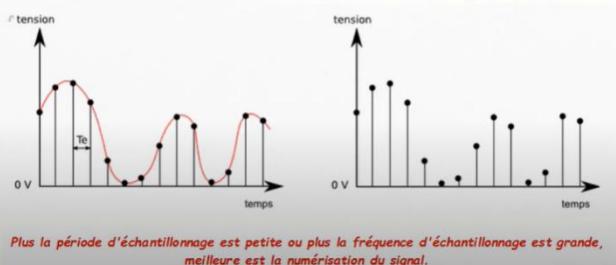
Pour stocker numériquement un signal (1D), l'échantillonnage va le réduire à une suite de points discrets. Par la suite :

- L'information présente sur le point de capture est la seule enregistrée,
- Une fréquence d'échantillonnage très faible inclut des acquisitions très espacées. Les détails entre deux positions de capture seront perdus.

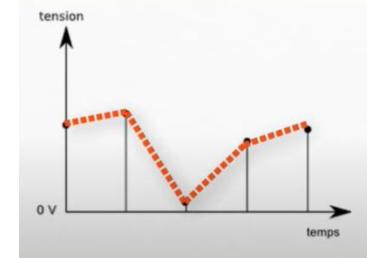
Sur cet exemple, la tension en rouge est mesurée 5 fois



Sur cet exemple, la tension en rouge est mesurée 14 fois



Le signal reconstruit ressemble très peu au signal analogique

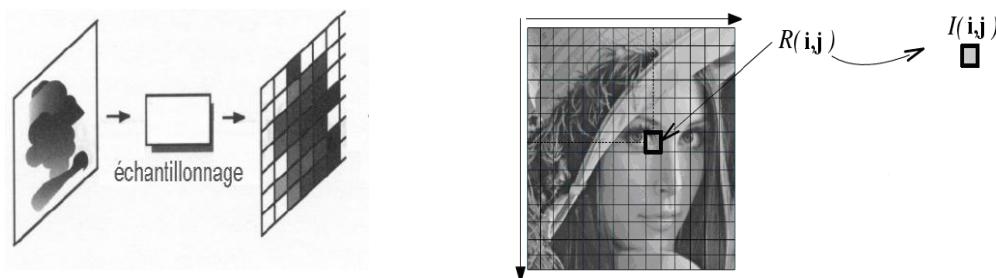


### III. NUMÉRISATION D'UNE IMAGE

- Etape-1 : Echantillonnage d'une image (résolution spatiale).

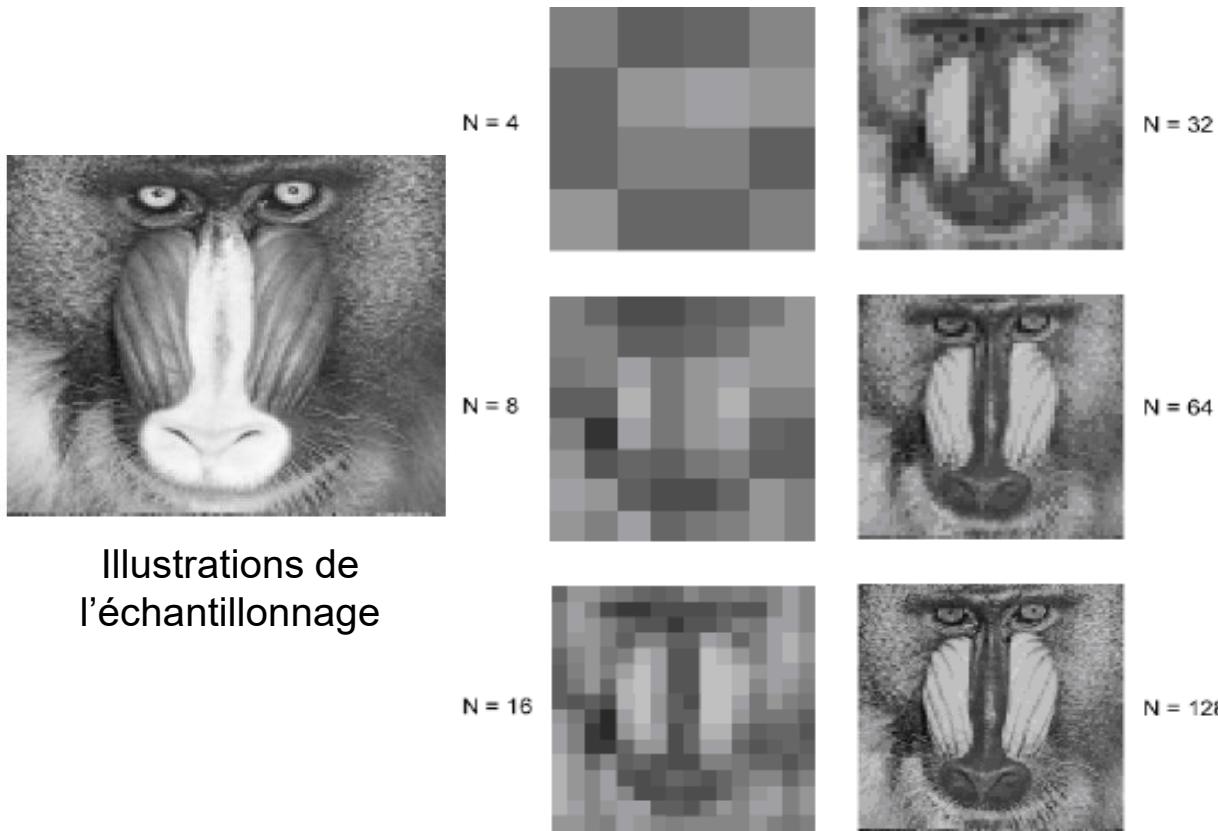
Discrétisation ***spatiale*** d'une image qui consiste à associer à chaque zone rectangulaire ou carrée  $R(i,j)$  d'une image continue une valeur  $I(i,j)$  unique.

- La résolution spatiale est le plus petit détail discernable.
- L'échantillonnage dépend de la capacité du capteur, (nombre de pixels disponible).
- Le sous-échantillonnage consiste à diminuer le nombre d'échantillons d'une image discrétisée.



# ECHANTILLONNAGE ET INFORMATION

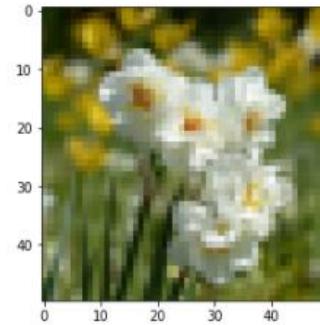
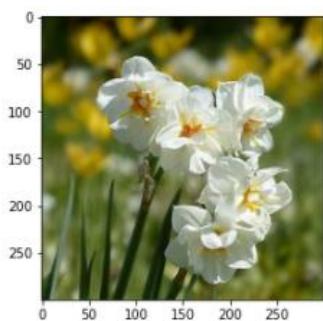
## III. Numérisation d'une image



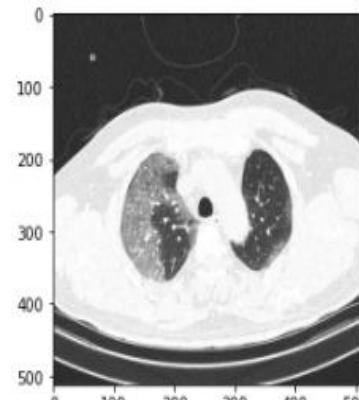
Illustrations de  
l'échantillonnage

# ECHANTILLONNAGE ET INFORMATION

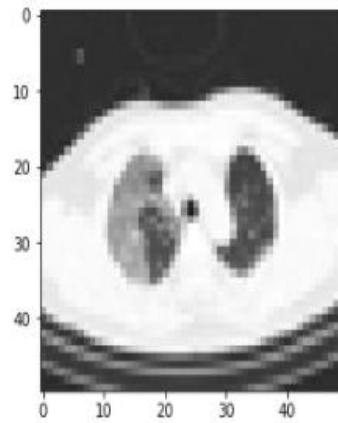
## III. Numérisation d'une image



512\*512



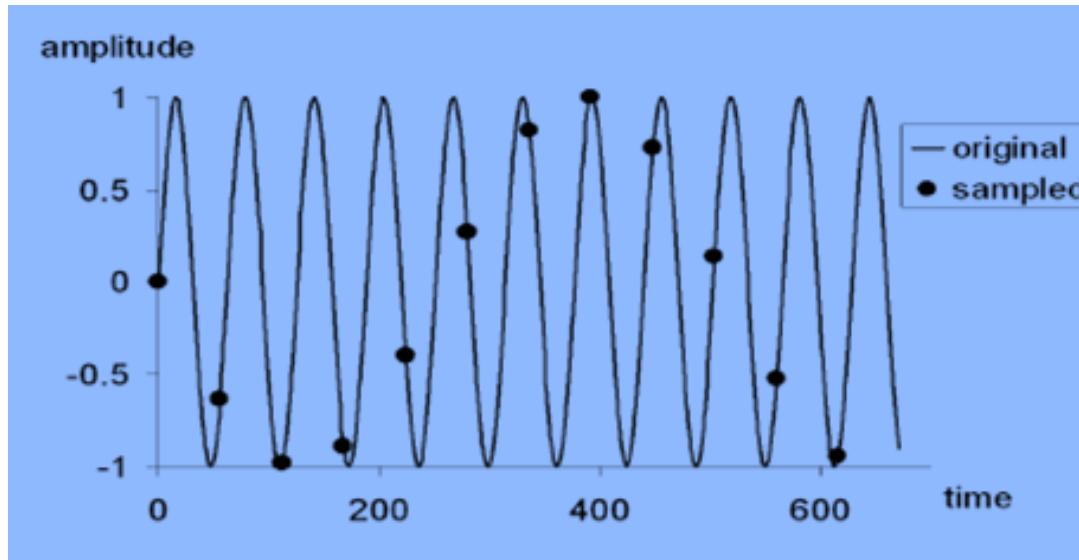
50 \* 50



# ECHANTILLONNAGE ET INFORMATION

## III. Numérisation d'une image

- Sur cet exemple (en 1d), le signal échantillonné « ressemble » à une sinusoïde de fréquence huit fois plus faible que l'originale.



# ECHANTILLONNAGE ET INFORMATION

## III. Numérisation d'une image

- *Aliasing* (2d) : il affecte *la fréquence et la direction* des structures périodiques.

sur une image naturelle

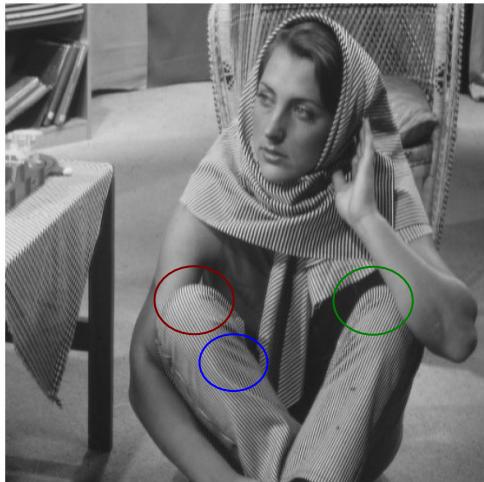


Image originale

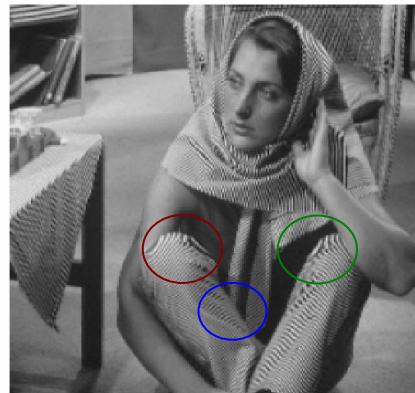


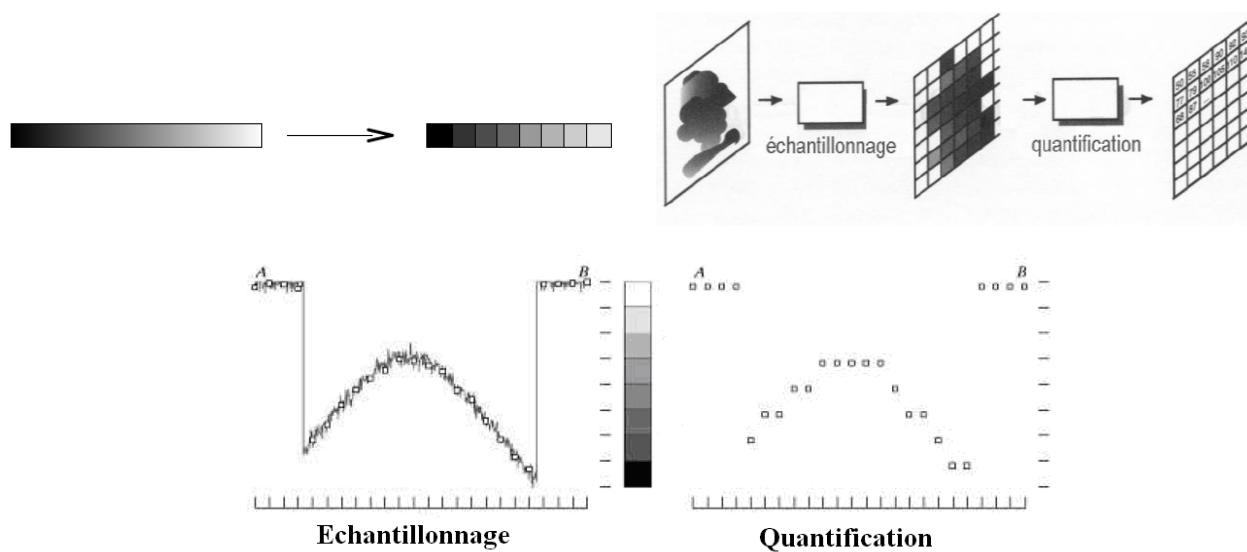
Image sous-échantillonnée

# ECHANTILLONNAGE ET INFORMATION

## III. Numérisation d'une image

### □ Etape-2 : Quantification

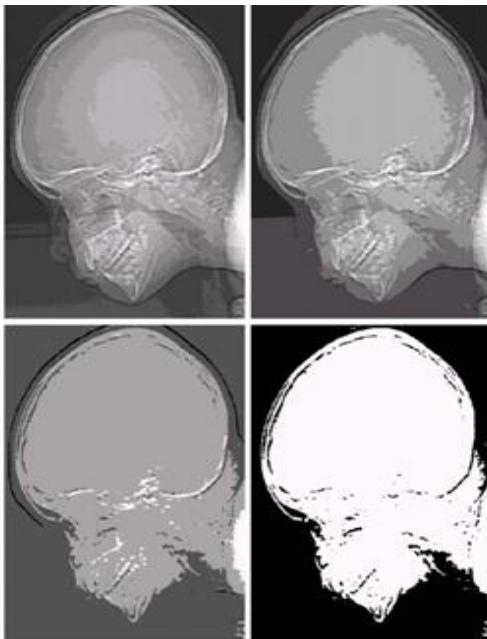
- La quantification se réfère au nombre de valeurs différentes disponible que peut prendre  $I(i,j)$ .
- La résolution tonale est le plus petit changement discernable.



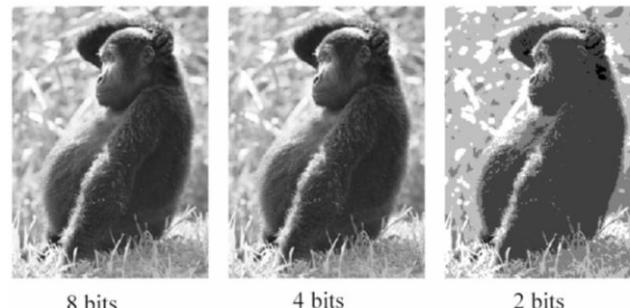
# QUANTIFICATION ET INFORMATION

## III. Numérisation d'une image

(Continued)  
(c)-(h) Image displayed in 16, 8, 4, and 2 gray levels. (Original courtesy of Dr. David R. Pickens, Department of Radiology & Radiological Sciences, Vanderbilt University Medical Center.)



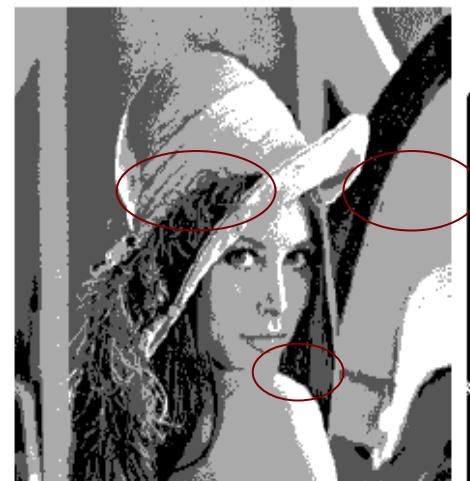
Illustrations de la quantification



# QUANTIFICATION ET INFORMATION

## III. Numérisation d'une image

- La quantification peut induire l'apparition des faux contours



## IV. REPRÉSENTATION DES IMAGES

- Point de vue informatique : L'image est un tableau de pixels à deux dimensions  $M \times N$  ( $M$  : hauteur,  $N$  : largeur).
- Un élément (pixel) est repéré par ses coordonnées  $(i,j)$ . L'origine du repère image se trouve en haut à gauche.
- Chaque élément a une valeur entière dans l'intervalle  $[L_{\min}, L_{\max}]$ . Cette valeur est codée sur un nombre de bits déterminant la couleur ou l'intensité du pixel, appelée **profondeur de codage** (parfois *profondeur de couleur*).
- Le nombre de bits requis pour représenter les niveaux de gris dans l'intervalle  $L$  est  $K$ . La relation entre  $K$  et  $L$  est :  $L=2^{k}$

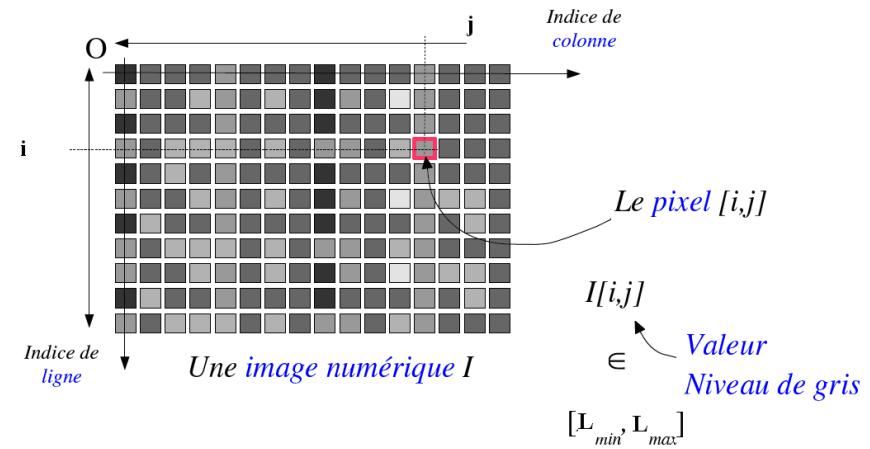
Par exemple, pour  $k=1$  bit,  $L=2^1=2$  (noir, blanc),

$k=8$ ,  $L=2^8= 256$  (256 niveaux de gris),...

- Pour une image ayant une résolution de  $M \times N$  pixels et une résolution de  $K$  bits (ou de  $L$  niveaux de tons), le nombre de bits pour stocker une image est:  
 **$b=M \times N \times K$**

# IV. REPRÉSENTATION DES IMAGES

- ❖ La surface de l'image numérique est divisée en éléments de tailles fixes : pixels
- ❖ Nous pouvons représenter une image par un tableau de pixels dont chaque case contient une valeur codée sur un nombre de bits déterminant la couleur ou l'intensité du pixel, appelée **profondeur de codage** (parfois *profondeur de couleur*).



x	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
41	210	209	204	202	197	247	143	71	64	86	94	54	54	57	56
42	205	196	203	197	195	210	207	56	65	58	53	53	61	62	51
43	201	207	192	201	198	213	156	69	65	57	55	52	53	60	50
44	216	206	211	193	202	207	208	57	69	80	55	77	49	62	51
45	224	206	211	194	196	197	226	56	60	66	55	46	67	58	108
46	209	214	224	198	194	193	204	173	84	90	59	51	82	58	48
47	204	212	213	208	191	190	191	214	60	82	68	76	51	49	55
48	214	215	215	207	208	190	172	188	88	72	55	49	56	52	56
49	209	205	214	201	204	196	187	190	86	82	65	67	57	60	46
50	208	200	205	203	202	198	174	185	149	71	63	56	55	45	56
51	207	210	211	199	217	194	183	177	209	96	82	64	32	93	32
52	208	203	200	209	197	194	193	197	167	209	98	68	61	51	56
53	204	208	203	209	196	203	188	185	183	221	75	81	58	80	60
54	200	203	199	206	188	197	183	190	163	156	122	63	58	94	66
55	205	210	202	203	199	197	196	191	173	186	115	62	57	94	63

## IV. REPRÉSENTATION DES IMAGES

- La résolution spatiale désigne le nombre de pixels d'une image (résolution spatiale en pixels)
- La **résolution en pixels par unité de longueur** d'une image désigne le nombre de pixel par unité de longueur décrivant la finesse des détails dans une image bitmap. La résolution se mesure principalement en :
  - dpi (ppi) : dot (pixel) per inch (2.54 cm)
  - ppm :pixel par millimètre
- Plus le nombre de pixels par inch (pouce) est grand, plus la **résolution est** élevée



## V.1 IMAGES BINAIRES

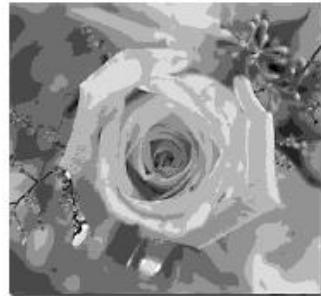
- **Images binaires** (noir ou blanc) : 1 bit



## V.2 IMAGES EN NIVEAUX DE GRIS

- **Images en niveaux de gris** : codés sur  $n$  ( $2 : 8$ ) bits.
- 4 niveaux = 2 bits
- 8 niveaux = 3 bits
- 16 niveaux = 4 bits (Bitmap 16 couleurs ou 16 niveaux de gris). En stockant 4 bits dans chaque case, il est possible de définir pour chaque pixel, c'est-à-dire 16 dégradés de gris allant du noir au blanc ou bien 16 couleurs différentes.
- 32 niveaux = 5 bits
- 64 niveaux = 6 bits
- 128 niveaux = 7 bits
- 256 niveaux = 8 bits. Images en teintes (ou niveaux) de gris. On ne code ici plus que le niveau de l'intensité lumineuse, généralement sur un octet (256 valeurs).  
➤ La valeur zéro représente le noir et la valeur 255 le blanc par convention

Illustrations d'images en niveau de gris: profondeur de codage variant de 2 à 8 bits (de gauche à droite, de haut en bas).

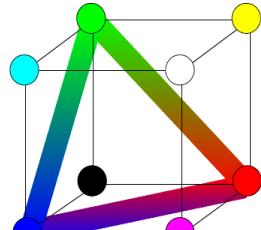


## V.3 IMAGES EN COULEUR

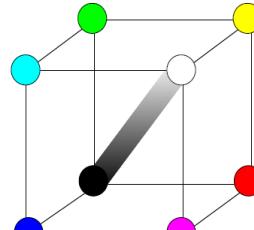
- Plusieurs espaces de codage de couleurs existent : CMJN, TSL, RVB, ...
- L'espace colorimétrique Rouge, Vert, Bleu RVB ou RGB (Red Green Blue) est très utilisé pour le maniement des images.
- Ce principe de synthèse additive de la couleur : se retrouve dans la plupart des dispositifs lumineux de restitution de la couleur : écrans cathodiques (CRT), écrans à cristaux liquides (LCD), Plasma.
- Les moniteurs d'ordinateur utilisent l'espace RVB pour l'affichage des couleurs

# IMAGES COULEURS: L'ESPACE RGB (RVB)

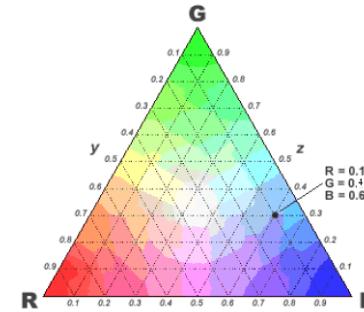
- L'espace RGB est l'espace vectoriel engendré par les 3 composantes primaires (Rouge, Vert, Bleu).
- Les primaires RVB sont des primaires additives, c'est-à-dire, que les contributions de chacun sont ajoutés pour créer le résultat. Toute couleur s'obtient par combinaison linéaire des trois primaires.
- Nuance de gris neutre : est obtenue si les trois composantes ont une valeur égale. Par exemple : s'ils ont la valeur 255, un blanc pur est obtenu, 0 pour chacune des composantes donne du noir.



Triangle chromatique



Axe achromatique



# IMAGES COULEURS : IMAGES 24 BITS

- 3 octets sont utilisés pour coder la couleur. La valeur de chaque composante couleur est un entier de 0 à 255.
- Le nombre de couleurs différentes dans l'espace RVB est de  $256 \times 256 \times 256$  : 16777216 de couleurs.

R	G	B	Couleur
0	0	0	noir
0	0	1	nuance de noir
255	0	0	rouge
0	255	0	vert
0	0	255	bleu
128	128	128	gris
255	255	255	blanc

# CONVERSION RVB EN NG

- La conversion entre les composantes de couleur RVB (Rouge, Vert, Bleu) et une intensité en niveau de gris peut se faire de différentes manières.
- **Calcul de l'intensité comme la moyenne des valeurs des composantes RVB:**

$$I = (R+V+B)/3$$

Où R, V et B sont les valeurs des composantes Rouge, Vert et Bleu respectivement pour un pixel donné.

- **Calcul de l'intensité (niveau de gris) avec des pondérations :**

Des pondérations différentes pour chaque composante RVB en fonction de leur importance relative dans la perception humaine de la luminosité sont utilisées.

Par exemple, les composantes vertes sont souvent perçues comme plus lumineuses que les composantes rouges et bleues.

Une formule couramment utilisée utilise ces pondérations :

$$I = 0.299 R + 0.587 V + 0.114 B$$

## V.4 IMAGES À PALETTES, IMAGES EN 256 COULEURS (8 BITS)

- But : réduire la place occupée par l'information de couleur,
- Principe : utiliser une *palette de couleurs* (ou tables d'index) « attachée » à l'image : on parle de couleurs indexées.
- La valeur de chaque pixel renvoie à l'entrée correspondante dans la palette de couleurs appelée *look-up table* (LUT).
- Dans la palette, les couleurs sont complètement spécifiées.
- La palette ou *look-up* (LUT) ne permet de sélectionner qu'un nombre limité de couleurs (256, par exemple).

## V.4 IMAGES À PALETTES, IMAGES EN 256 COULEURS (8 BITS)

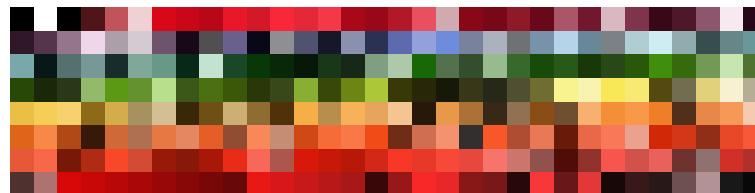
- La conversion d'une image RVB (Rouge, Vert, Bleu) en une image à couleurs indexées permet de réduire le nombre de couleurs utilisées dans l'image à un certain nombre prédéfini de couleurs.
- Gain de place: si on considère l'exemple de 256 couleurs
  - Pour les coder : la palette occupera 24 bits x 256 entrées. Les pixels de l'image seront associés à des index codés sur un octet.
  - ✓ Taille de l'image : 1 octet par pixel + taille de la LUT (un peu plus du tiers de la place occupée par une image en couleurs 24 bits).
  - ✓ Il faut noter ici que plus l'image contient de pixels, plus le gain de place sera plus élevé.
- On peut changer seulement la LUT d'une image en conservant les valeurs des index de l'image

## V.4 IMAGES À PALETTES, IMAGES EN 256 COULEURS (8 BITS)



Mode couleurs indexées: permet d'obtenir jusqu'à 256 couleurs fixes, définies à l'avance dans une palette. Il utilise qu'une seule couche.

- Codage en 8 bits par pixel (bpp) =>  $2^8 = 256$  possibilités  
Chaque pixel peut avoir jusqu'à 256 couleurs fixes possibles.

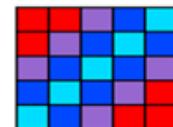


Index

0	0	1	2	3
0	1	2	3	2
1	2	3	2	1
2	3	2	1	0
3	2	1	0	0

Palette (Cr,Cv,Cb)

220	0	0
120	20	30
0	0	200
0	20	210



0 =   
1 =   
2 =   
3 =

## V.5 IMAGES AVEC GESTION DE LA TRANSLUCIDITÉ

- Canal alpha : un canal supplémentaire définissant le degré de transparence de l'image. Il est codé sur un nombre fixe de bits par pixel (8 ou 16 bits ).
- La translucidité d'un pixel est échelonné linéairement de l'opacité complète à la transparence.
- Une valeur de 0 signifie que le pixel est totalement transparent (invisible), tandis qu'une valeur de 255 signifie que le pixel est totalement opaque (entièrement visible).
- utile dans des domaines tels que la conception graphique, les jeux vidéo, les effets spéciaux et la réalité augmentée.

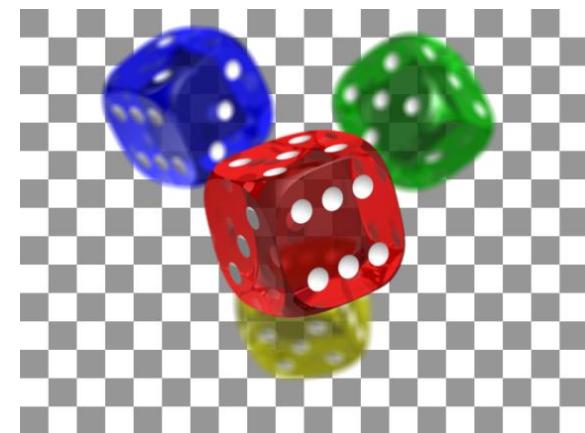


Image sur un fond en damier permet d'observer la transparence.

## VI. TYPES ET FORMATS D'IMAGES

- Une **image matricielle** (« bitmap » ou « raster ») : est une image numérique dans un format de données composé d'un tableau de pixels qui peut se visualiser sur un moniteur d'ordinateur, tout autre dispositif d'affichage.
- Exemple : images en niveaux de gris ou couleurs de type photo : GIF, PCX, BMP, JPEG, PPM, PGM...



# IMAGES VECTORIELLES

- Une **image vectorielle**, est une image numérique composée d'objets géométriques individuels (segments de droite, polygones, arcs de cercle, etc.) caractérisés par différents attributs: de couleur, de forme, de position, etc.
- Une image vectorielle d'un cercle : peut être définie par les attributs : position du centre, rayon...
- Un dessin vectoriel est dessiné à chaque visualisation, induisant des calculs sur la machine.
- L'avantage de ce type d'image est qu'elles occupent peu de place en mémoire. De plus, elles peuvent être redimensionnées tout en conservant les informations et sans effet d'escalier (crénelage).
- Ce type d'images est très utilisé dans la génération de schémas avec certains logiciels de DAO (Dessin Assisté par Ordinateur) comme AutoCAD ou CATIA
- Exemple de formats d'images : SVG, flash, AI, Eps, DXF.

Image Matricielle

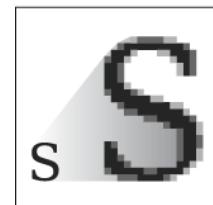


Image Vectorielle



# FORMATS D'IMAGES

- La notion de formats d'images est très importante pour le concepteur et l'utilisateur.
- De point de vue concepteur c'est la capacité de son outil à gérer le maximum de format et à stocker ses résultats sous différents formats lisibles. De côté utilisateur, c'est son besoin de lire des images quelques soit le format proposé.
- Dans des domaines particuliers les formats sont généralement spécifiques aux concepteurs pour avoir main mise sur le marché.
- Sur un support de masses (disques ...) les images sont stockées sous la forme de fichiers qui contiennent :
  - Une entête ( Hauteur, Largeur, LUT)
  - La valeur des pixels

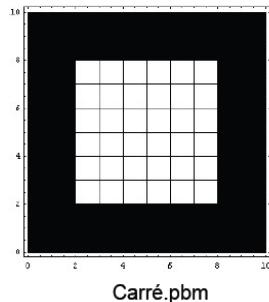
# FORMATS D'IMAGES

Il existe deux grandes catégories de format :

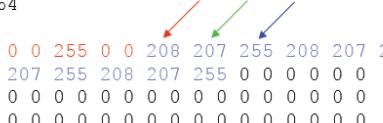
- Sans compression :
  - PBM Noir et Blanc
  - PGM Niveaux de gris
  - PPM Couleurs
  - BMP format Windows
- Avec compression
  - Sans perte (TIFF, PNG, GIF ...)
  - Avec perte (JPEG ...)

## Illustrations de Formats d'Images : PBM, PGM et PPM

## Format PBM (Portable Bit Map)



## Format PPM (Portable Pixel Map)

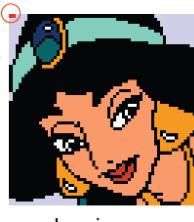
```
P3  
# Created by Paint Shop Pro 5  
64 64  
255  


|     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 255 | 0   | 0   | 255 | 0   | 0   | 208 | 207 | 255 | 208 | 207 | 255 |
| 208 | 207 | 255 | 208 | 207 | 255 | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 208 | 207 | 255 | 208 | 207 | 255 | 208 | 207 | 255 | 208 | 207 | 255 |



...


```



## Format PGM (Portable Grey Map)

```
P2  
# Created by Paint Shop Pro 5  
64 64  
255 ← Image 8 bits  
212 212 212 212 212 212 212 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 212 212 212 212 212  
212 212 212 212 212 212 212 212 212  
212 212 212 212 212 212 212 212 212  
0 0 0 0 0 0 0 0 0 167 167 0 0 0  
0 0 204 204 204 204 204 204 204 204  
204 204 204 204 204 204 204 204 204
```



Jasmine.pgm

"P3" désigne la variante qui stocke les couleurs en texte brut, (P6" stocke les couleurs en binaire).

64 64 dimension de l'image

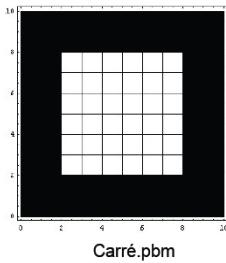
255 : valeur maximale de l'image

## Illustrations de Formats d'Images : PBM, PGM et PPM

- Le format PBM (Portable Bitmap) est un format d'image permettant de représenter des images en noir et blanc (monochromes) en utilisant uniquement des pixels noirs et blancs (ou 0 et 1).
- Variantes du PBM
  - Image noir et blanc en format text, chaque pixel est représenté par un caractère '0' pour noir et '1' pour blanc.
  - Image noir et blanc en format binaire : PBM P4 est binaire.

### Format PBM (Portable Bit Map)

```
P1
# Created by Paint Shop Pro 5
10 10 ← nbcol nblig
1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
```



- "P1" format PBM, variante textuelle
- 10 10 dimension de l'image

## Illustrations de Formats d'Images : PBM, PGM et PPM

- Le format PGM est utilisé pour représenter des images en niveaux de gris.
  - Chaque pixel est représenté par un nombre (généralement entre 0 et 255) indiquant l'intensité du gris

## Format PGM (Portable Grey Map)

```
P2
# Created by Paint Shop Pro 5
64 64
255 ← Image 8 bits
entête
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
212 212 212 212 212 212 212 212 212 212 212 212 212 212 212 212
212 212 212 212 212 212 212 212 212 212 212 212 212 212 212 212
212 212 212 212 212 212 212 212 212 212 212 212 212 212 212 212
0 0 0 0 0 0 0 0 0 0 0 0 167 167 0 0 0
0 0 204 204 204 204 204 204 204 204 204 204 204 204 204 204
204 204 204 204 204 204 204 204 204 204 204 204 204 204 204 204

Jasmine.pgm
```

"P3" désigne la variante qui stocke les couleurs en texte brut,

(P6" stocke les couleurs en binaire) •

## 64 64 dimension de l'image

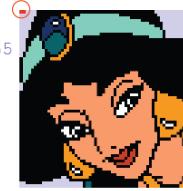
## 255 : valeur maximale de l'image

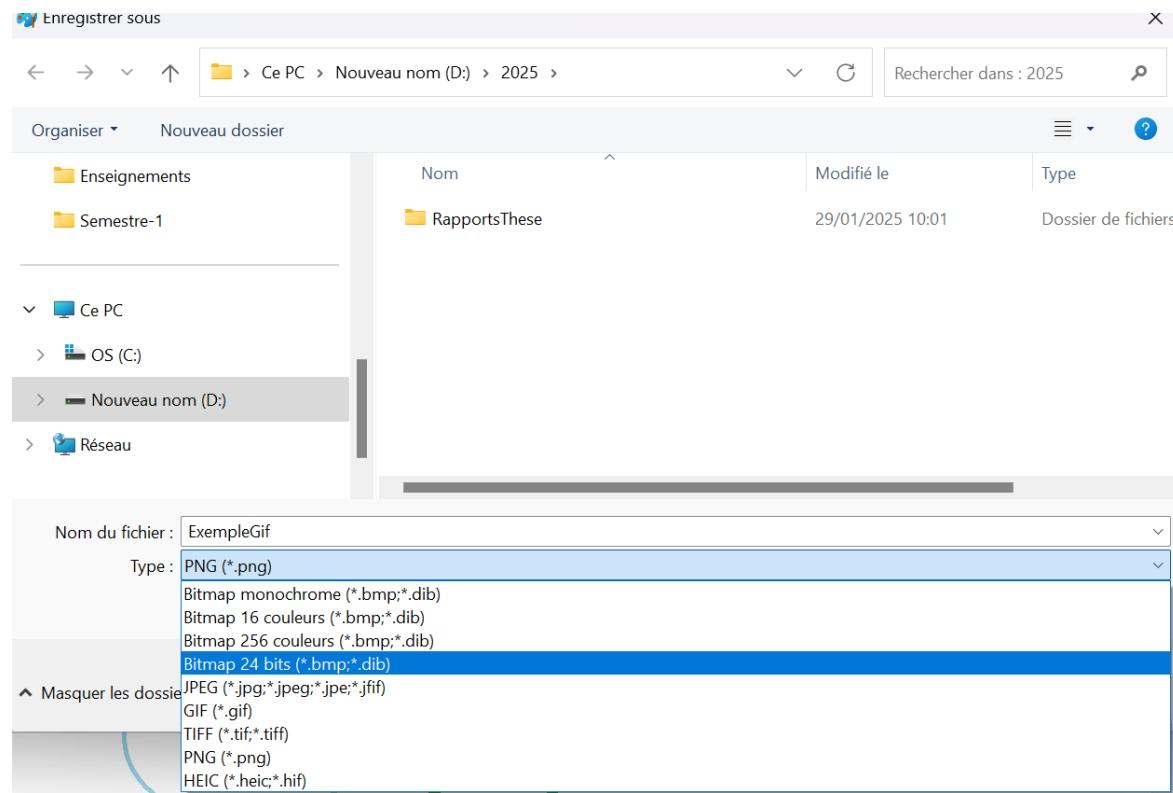
## Format PPM (Portable Pixel Map)

```
P3  
# Created by Paint Shop Pro 5  
64 64  
255  


|     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 255 | 0   | 0   | 255 | 0   | 0   | 208 | 207 | 255 | 208 | 207 | 255 |
| 208 | 207 | 255 | 208 | 207 | 255 | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |
| 208 | 207 | 255 | 208 | 207 | 255 | 208 | 207 | 255 | 208 | 207 | 255 |


```





Extension(s) de fichier(s)	.bmp		
Spécification	Aucune spécification ; toutefois, Microsoft fournit une documentation générale sur le format à l'adresse <a href="https://docs.microsoft.com/fr/windows/desktop/gdi/bitmap-storage">docs.microsoft.com/fr/windows/desktop/gdi/bitmap-storage</a>		
Compatibilité des navigateurs	Toutes versions de Chrome, Edge, Firefox, Internet Explorer, Opera, et Safari		
Dimensions maximales	Soit $32\ 767 \times 32\ 767$ ou $2\ 147\ 483\ 647 \times 2\ 147\ 483\ 647$ pixels, selon la version du format		
Modes de couleur supportés	Mode de couleur	Bits par composante ( $D$ )	Description
	Niveau de gris	1	Chaque bit représente un seul pixel, qui peut être noir ou blanc.
	Vraie couleur	8 et 16	Chaque pixel est représenté par trois valeurs représentant les composantes de couleur rouge, verte et bleue ; chacune d'entre elles est constituée de $D$ bits.
	Couleur indexée	2, 4, et 8	Chaque pixel est représenté par une valeur de 2, 4 ou 8 bits, qui sert d'index dans la table des couleurs.
	Niveau de gris avec alpha	n/a	Le BMP n'a pas de format d'échelle de gris distinct.
	Vraie couleur avec alpha	8 et 16	Chaque pixel est représenté par quatre valeurs représentant les composantes de couleur rouge, verte, bleue et alpha ; chacune d'entre elles est constituée de $D$ bits.
Compression	Plusieurs méthodes de compression sont prises en charge, notamment les algorithmes avec ou sans perte		

Extention(s) du fichier(s)	.gif		
Spécification	<a href="#">Spécification GIF87a</a> <a href="#">Spécification GIF89a</a>		
Compatibilité des navigateurs	Toutes les versions de Chrome, Edge, Firefox, Internet Explorer, Opera, and Safari		
Dimensions maximales	65 536×65 536 pixels		
Modes de couleur pris en charge	Mode couleur	Bits par composante ( $D$ )	Description
	Niveau de gris	n/a	Le GIF n'inclut pas un format de niveaux de gris spécifique.
	Vraie couleur	n/a	Le GIF ne prend pas en charge les pixels en couleurs réelles.
	Couleur indexée	8	Chaque couleur d'une palette GIF est définie par 8 bits de rouge, de vert et de bleu (24 bits au total par pixel).
	Niveau de gris avec alpha	n/a	Le GIF ne fournit pas de format de niveaux de gris spécifique.
	Vraie couleur avec alpha	n/a	Le GIF ne prend pas en charge les pixels en couleurs réelles.
Compression	Sans perte (LZW)		

Extentions(s) de fichier(s)	. jpg , . jpeg , . jpe , . jif , . jfif		
Spécification	<a href="http://jpeg.org/jpeg/">jpeg.org/jpeg/</a>		
Compatibilité des navigateurs	Toutes les versions de Chrome, Edge, Firefox, Internet Explorer, Opera et Safari		
Dimensions maximales	65 535×65 535 pixels		
Modes de couleur pris en charge	Mode de couleur	Bits par composante (D)	Description
	Niveau de gris	n/a	Le JPEG n'a pas de mode d'échelle de gris distinct.
	Vraie couleur	8	Chaque pixel est décrit par les composantes de couleur rouge, bleue et verte, chacune d'entre elles étant composée de 8 bits.
	Couleur indexée	n/a	Le JPEG n'offre pas de mode couleur indexé.
	Niveau de gris avec alpha	n/a	Le JPEG ne prend pas en charge un canal alpha.
Compression	Vraie couleur avec alpha	n/a	Le JPEG ne prend pas en charge un canal alpha.
	Sans perte; sur la base de la <a href="#">transformée en cosinus discrète</a>		

## VII. Opérations entre images

- Chaque pixel de l'image est caractérisé par sa position  $(i, j)$  et son intensité  $k$ .
- On peut citer deux types de transformations sur les pixels de l'image :
  - ✓ les **transformations géométriques** modifiant les positions des pixels,
  - ✓ les **transformations radiométriques** modifiant les intensités des pixels.
- Il est possible d'effectuer sur l'image des opérations utilisant ces deux types de transformation simultanément.

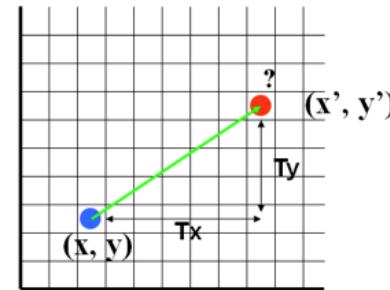
# VII. Opérations entre images

## A. Exemples de Transformations Géométriques

Comme exemple de transformations géométriques, nous pouvons citer la transformation affine.

- Dans le plan 2D, la translation de vecteur  $(t_x, t_y)^t$ , transforme le point  $P(x, y)$  en  $P'(x', y')$  tel que :

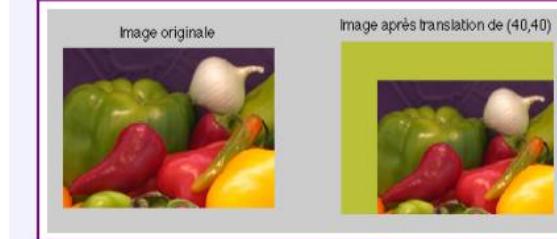
$$\begin{cases} x' = x + tx \\ y' = y + ty \end{cases}$$



$$\begin{pmatrix} i' \\ j' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} t_i \\ t_j \end{pmatrix}$$

- On peut alors définir une matrice de translation T

et l'écriture matricielle de la translation devient :

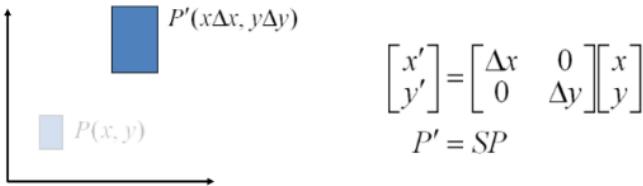
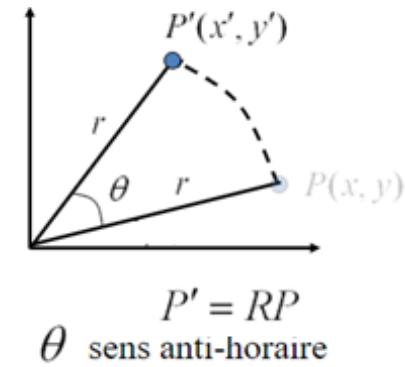


# VII. Opérations entre images

## A. Exemples de Transformations Géométriques

- La rotation d'un pixel  $(i, j)$  d'angle  $\theta$  (dans un repère au centre de l'image) s'exprime :

$$\begin{pmatrix} i' \\ j' \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix}$$



- Le changement d'échelle d'un pixel  $(i, j)$  de coefficients  $\alpha_i$  et  $\alpha_j$  s'exprime :

$$\begin{pmatrix} i' \\ j' \end{pmatrix} = \begin{pmatrix} \alpha_i & 0 \\ 0 & \alpha_j \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix}$$



# VII. Opérations entre images

## B. Exemples de Transformations

### Radiométriques

#### Opérations Logiques :

- Opérations réalisées bit par bit sur les images
- Appliqués à des images en niveaux de gris, les opérations logiques s'effectuent sur des chaînes de bits:

$$a = 131 \rightarrow 10000011$$
$$\bar{a} \rightarrow 01111100 \rightarrow 124$$
$$124 + 131 = 255$$

$$a = 109 \rightarrow 01101101, b = 89 \rightarrow 01011001$$
$$a \& \& b \rightarrow 01001001 \rightarrow 73$$

et	0	1
0	0	0
1	0	0

o	0	1
u	0	1
1	1	1

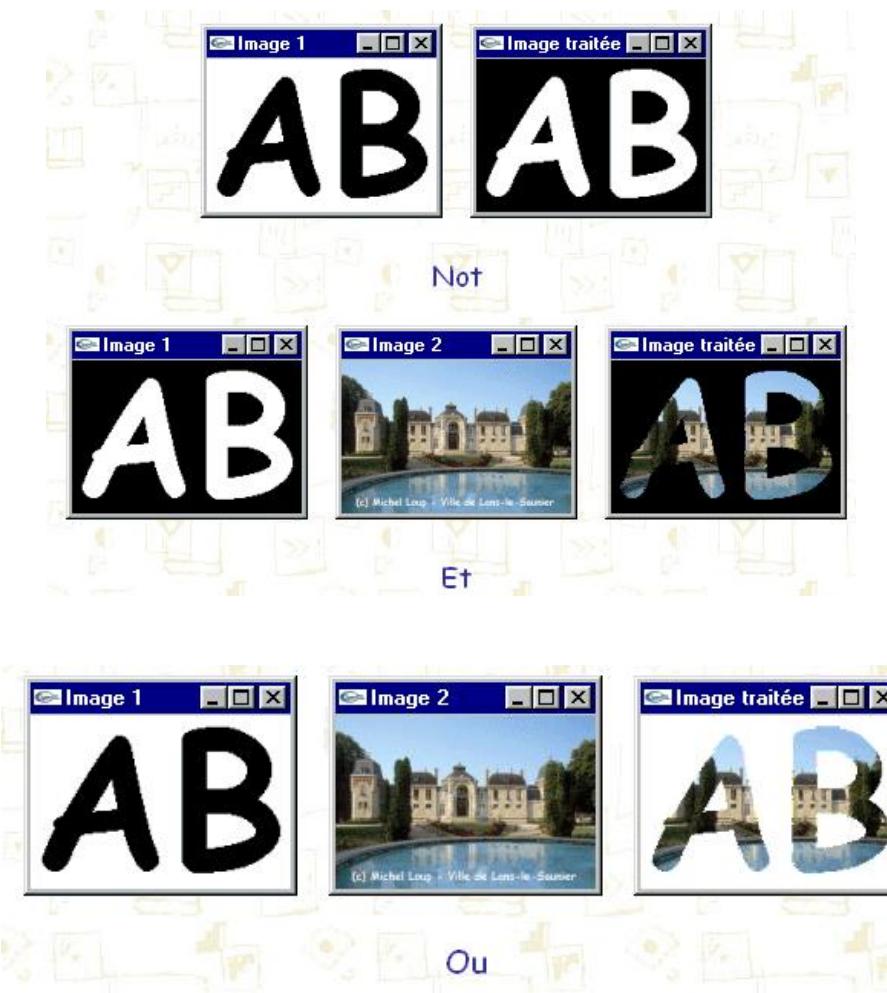
xo r	0	1
0	0	1
1	1	0

-	no t
0	1
1	0

# VII. Opérations entre images

## B. Exemples de Transformations

### Radiométriques



# VII. Opérations entre images

## B. Exemples de Transformations

### Radiométriques

#### Opérations Arithmétiques: Addition

- On peut définir la l'addition A pixel à pixel de deux images f et g par :

$$A(x,y)=\text{Min}(f(x,y)+g(x,y);255)$$



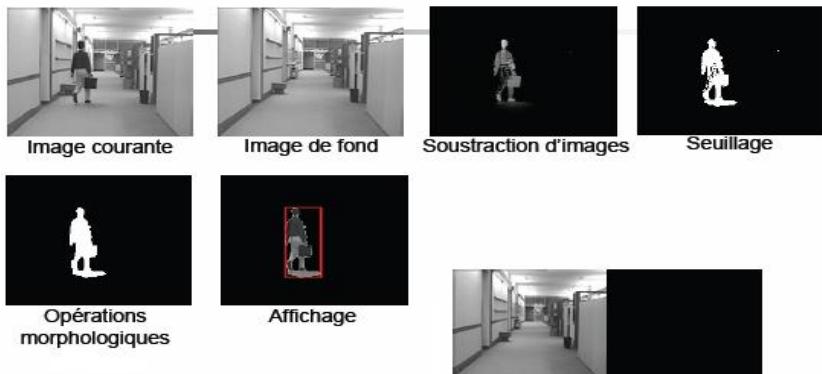
# VII. Opérations Entre images

## B. Exemples de Transformations Radiométriques

### Opérations Arithmétiques: Soustraction

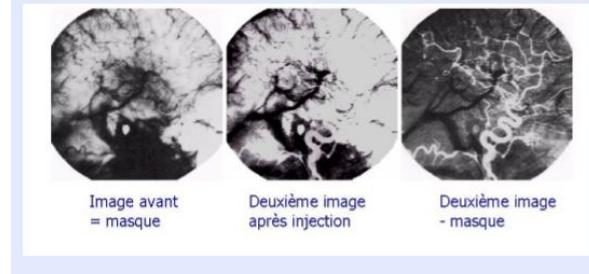
□ On peut définir la soustraction S pixel à pixel de deux images f et g par :

$$S(x,y) = \text{Max}(f(x,y) - g(x,y); 0)$$



Exemple : vidéo surveillance

Soustraction ⇒ : visualisation du changement (images superposables)

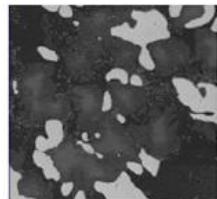


# VII. Opérations Entre images

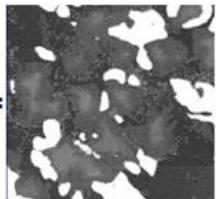
## B. Exemples de Transformations Radiométriques

### Opérations arithmétiques: Multiplication

- On peut définir la multiplication M d'une images f par un ration par :  
 $M(x,y)=\text{Min}(f(x,y)*\text{ration};255)$
- La multiplication d'images peut améliorer le contraste ou la luminosité



$\times 1,5 =$



$\times 1,2 =$



Image originale   Masque multiplicateur   Image résultat

# EXERCICES

## Exercice-1 : Calcul de la résolution d'une image

Pour calculer la résolution d'une image, deux aspects sont considérés :

- la résolution spatiale (nombre de pixels)
- et la résolution en pixels par unité de longueur (pixels par pouce ou pixels par centimètre) :

Soit une image I de dimensions 1920 pixels en largeur et 1080 pixels en hauteur.

La taille physique de l'image en pouces (ou centimètres) est de 10 pouces en largeur et 5.625 pouces en hauteur (1 pouce = 2,54 cm).

**Calculer la résolution spatiale (en pixels) et la résolution en pixels par pouce (PPP).**

## Exercice-2 : Conversion Couleur en NG

Supposons que nous avons un pixel dans une image RVB avec les valeurs de composantes suivantes : Rouge (R) : 120 , Vert (G) : 200, Bleu (B) : 50

**Convertir ce pixel en niveaux de gris en utilisant différentes méthodes de pondération des composantes RVB.**

**PS: pour la seconde méthode, utilisez les coefficients de pondération suivants : 0.3 pour le rouge, 0.59 pour le vert et 0.11 pour le bleu**

### Exercice-3

Soient les valeurs de composantes RVB suivantes pour trois pixels d'une image RVB:  
Pixel 1 : (200, 100, 50), Pixel 2 : (50, 150, 200) et Pixel 3 : (100, 200, 150)

Convertir ces pixels en couleurs indexées en utilisant une palette de couleurs avec les couleurs suivantes :

Couleur 1 : (255, 0, 0) (rouge), Couleur 2 : (0, 255, 0) (vert) et  
Couleur 3 : (0, 0, 255) (bleu)

### Exercice-4

1. Calcul de l'intensité en utilisant la moyenne simple :

$$Imoyenne = \frac{120+200+50}{3} = 123.33 = 123$$

L'intensité en niveaux de gris est environ 123 (arrondi)

2. Calcul de l'intensité en utilisant la méthode avec pondération :  
 $I_{pondéré} = 0.3 \times 120 + 0.59 \times 200 + 0.11 \times 50$   
 $= 36 + 118 + 5.5 = 159.5 \approx 159$  (arrondi)

## RÉFÉRENCES

1. Introduction au traitement d'images. Diane Lingrand. Édition Vuibert, 2008.
2. Le traitement des images. Henri Maître. Hermès Science Publications 2002.
3. Analyse d'images : filtrage et segmentation. J.P. Cocquerez et S. Philipp. Masson 1995.
4. Le traitement des images. I. Bloch, Y. Gousseau, H. Maître, D. Matignon, B. Pesquet-Popescu, F. Schmitt, M. Sigelle, F. Tupin, tome 2. Version 5.0, 2005. (sur internet <https://perso.telecom-paristech.fr/bloch/ANIM/poly-animate2.pdf>).
5. <http://perso.ensta-paristech.fr/~manzaner/Cours>
6. [https://www.isima.fr/~vbarra/IMG/pdf/diapos\\_cours.pdf](https://www.isima.fr/~vbarra/IMG/pdf/diapos_cours.pdf)
7. [https://perso.limsi.fr/vezien/pdf\\_cours\\_ima\\_jmv.pdf](https://perso.limsi.fr/vezien/pdf_cours_ima_jmv.pdf)