

**Chapitre 2. . Capteurs d'images et dispositifs d'acquisition numérique
(2Semaines)**

- Schéma de principe d'une chaîne de traitement d'images.
- Principe des capteurs CCD et CMOS
- Spécifications des capteurs couleur
- Numérisation d'une image
- Notions de définition, résolution et quantification d'une image numérique (taille, dpi, ppi, bpp ...etc)
- Exemples de formats d'images numériques (BMP, TIFF, JPG, GIF et PNG)

2.1 Schéma de Principe d'une Chaîne de Traitement d'Images

L'allure générale d'un système d'analyse d'images est ainsi résumée sur la Figure

1.21.

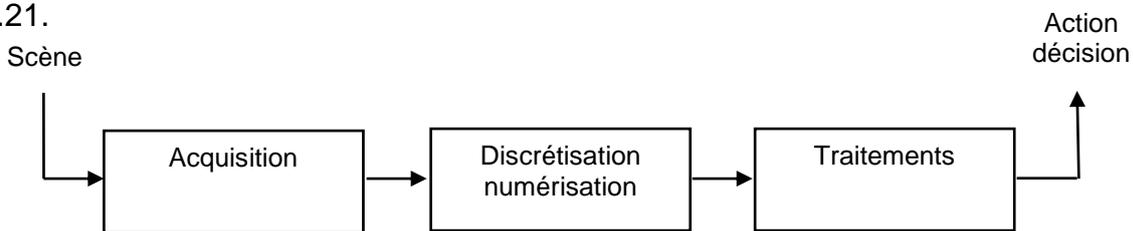


Fig.1.21 Allure générale d'un système d'analyse d'images

1.5 METHODOLOGIE EN ANALYSE D'IMAGES

La plupart des systèmes d'analyse d'images adoptent la même méthodologie : en suivant Marr, une succession de représentations intermédiaires est construite par un enchaînement de traitement conduisant à une interprétation des images. Nous distinguons catégories de transformation en traitement des images :

- Le traitement des images :
image → *image*, « Image Processing »
- L'analyse d'image :
image → *mesures*, « Image Analysis »
- L'analyse sémantique :
image → *description de haut niveau* « Understanding Image »

La Figure 1.22 résume cette méthodologie et permet d'identifier les différentes phases détaillées au cours des paragraphes suivants.

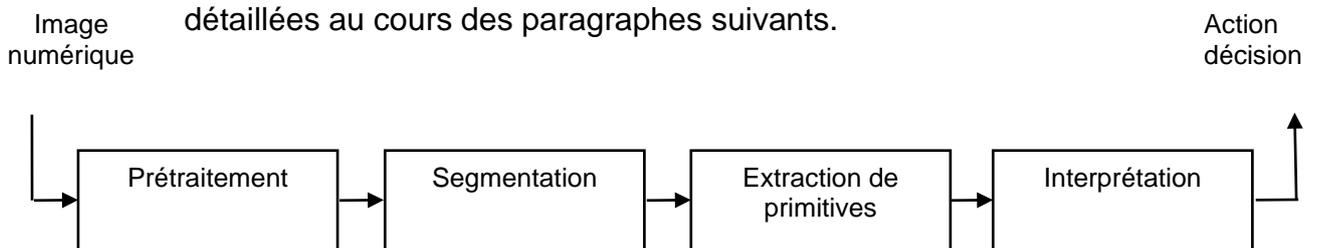
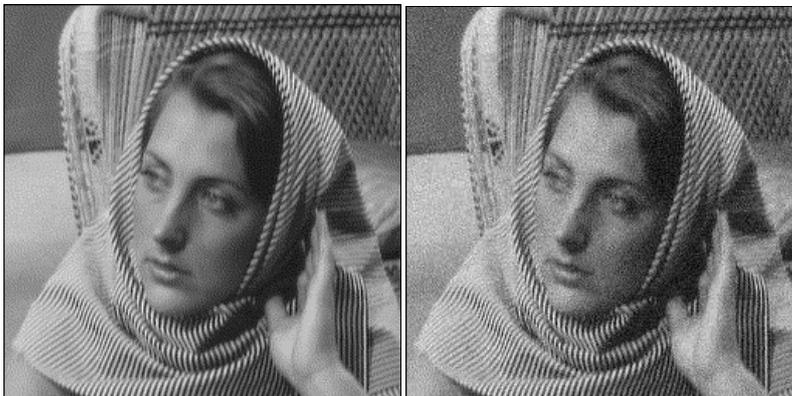


Fig. 1.22 Enchaînement des traitements

1.5.1 Prétraitement

Ces transformations correspondent essentiellement à des méthodes d'amélioration et de restauration d'images. Situés en amont des autres phases d'extraction de l'information. Les prétraitements permettent de limiter les distorsions et les fluctuations du système d'acquisition. Sans constituer véritablement une étape complète dans l'enchaînement de traitements, cette phase de prétraitements se révèle pourtant fondamentale ; la fiabilité des techniques d'analyse d'images dépend pour une grande partie des performances et de la qualité des prétraitements effectués. Un exemple de prétraitement est l'application du filtre passe-bas, filtre gaussien, filtre médian et le filtre Wiener. Le résultat du filtrage est montré sur la Figure 1.24. L'importance du prétraitement est rarement démentie dans les applications industrielles mais nous pouvons néanmoins considérer que l'extraction de l'information utile débute avec la phase de **segmentation**.

Exemple (sous Matlab)



(a) Image originale (b) Image bruitée (bruit blanc gaussien $\sigma = 25$)



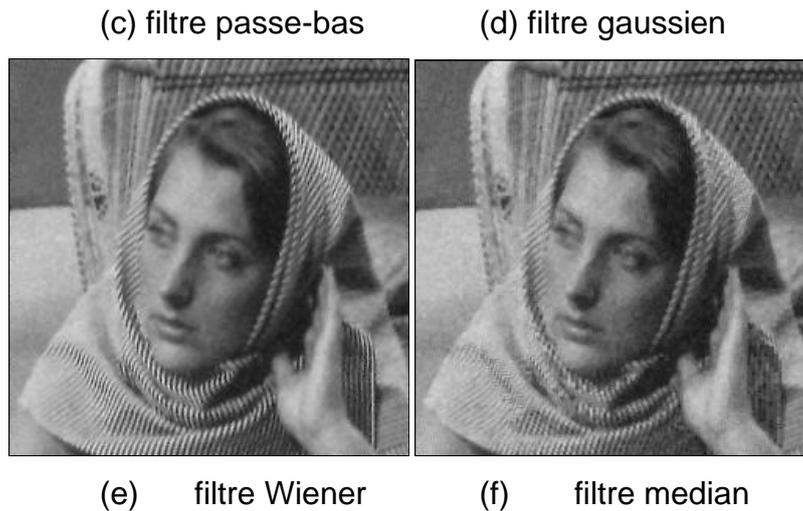


Fig. 1.23 Exemples de résultats de prétraitement par différents filtres.

```
>> load woman
>> x=double(X);
>> I_n=im_noise(x,25);           % fonction créée en matlab pour générer une image bruitée
>> h = fspecial('gaussian');     % filtre gaussien
>> G = imfilter(I_n,h,'same','conv');% filtrage par le filter gaussien
>> figure('name','Gaussian'),imshow(uint8(G),'true size');           % visualiser l'image
```

1.5.2 Segmentation

Lorsque nous regardons une image, nous pouvons analyser son contenu grâce à nos connaissances. En effet, si on nous présente une image de paysage de montagne, nous allons reconnaître immédiatement la montagne, les arbres, le chalet. Maintenant, si on nous présente la photo de l'oursin représentée dans la Figure 2.5, nous verrons une forme sphérique (car nous percevrons le relief) sans pour autant savoir de quoi il s'agit. On pourrait penser à une vue grossie d'un virus ou d'une bactérie ou encore à la vue de dessus d'un champignon.

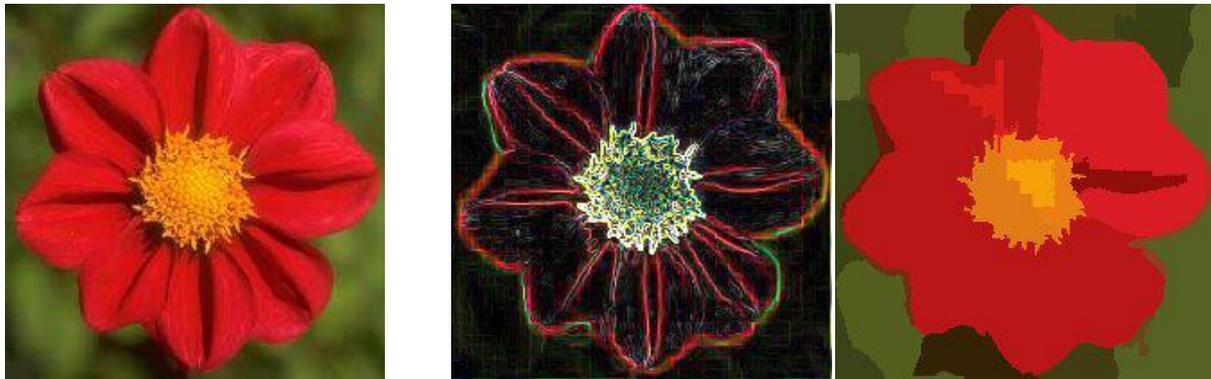
Afin d'analyser la scène contenue dans une image, une méthode classique consiste en la recherche des différentes régions qui composent l'image. Dans la plupart des cas, ces régions correspondent aux objets présents dans la scène. La recherche des régions qui composent une image est la segmentation.

La segmentation représente une étape essentielle dans la recherche d'informations pertinentes sur l'environnement. Cette étape peut être décrite comme une phase assez générale de réduction de l'information qui vise à extraire des indices et des structures significatives des images. Il existe deux approches de segmentation :

- **Détection de Contours** : on cherche à déterminer les frontières des régions de l'image

- **Segmentation par Région** : on cherche directement les régions présentes dans l'image.

Ces deux approches ne donnent pas forcément la même partition de l'image et les résultats dépendent très largement de la méthode utilisée. La Figure 1.24 donne un exemple de segmentation d'une image.



(a) Image originale

(b) détection de contours

(c) segmentation par région

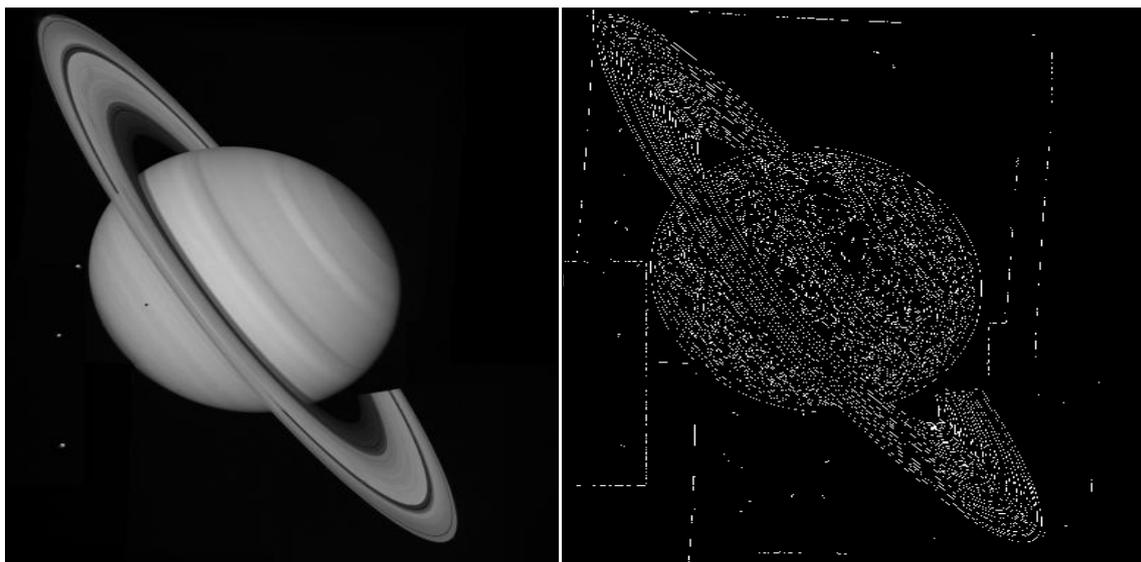
Fig. 1.24 Exemple de segmentation

Exemple (Sous Matlab) : Détection De Contours

```

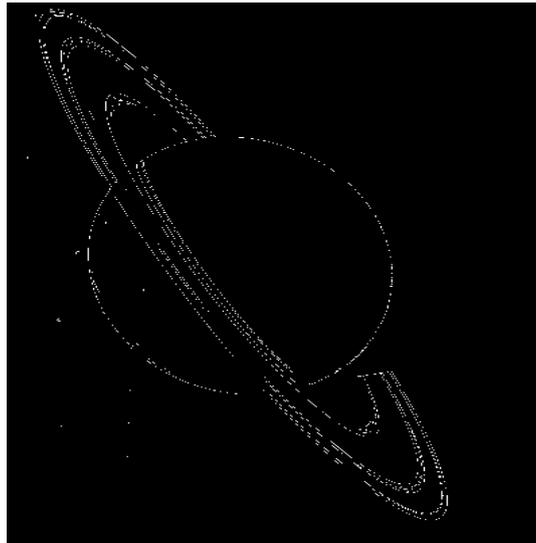
I = imread('saturn.pgm'); % lire l'image
BW1 = edge(I,'prewitt'); % l'instruction edge avec le masque "Prewitt"
BW2 = edge(I,'canny'); % l'instruction edge avec le détecteur "Canny"
figure, imshow(BW1); % afficher l'image
figure, imshow(BW2); % afficher l'image

```



(a) Image originale

(b) détection de contours par « Canny »



(c) détection de contours « masque de prewitt »

Fig. 1.24 Exemples de détection de contours par le détecteur « **Canny** » et le masque « **Prewitt** »

1.5.3 Extraction de Primitives

La phase de segmentation regroupe les pixels en régions homogènes et organise les données. L'extraction de primitives peut être vue comme la suite logique de ce traitement à un niveau conceptuel plus élevé : les résultats produits à l'issue de cette étape ne correspondent en effet plus à des images, mais une liste d'attributs ou de primitives liées à des notions perceptives telles que la forme, la texture, etc Des structures significatives pourront ainsi être extraites en regroupant des entités qui n'étaient pas reliées à l'issue de la segmentation. L'extraction de primitives précise de plus également les caractéristiques des régions détectées. Les primitives seront associées à des propriétés géométriques (composantes connexes, régions adjacentes,...), statistiques (texture,...). Un nombre restreint de primitives suffit alors à identifier et décrire chacune des régions et des objets observées, facilitant en cela la phase d'interprétation.

1.5.4 Interprétation

Cette étape dépend du contexte dans lequel s'inscrit le système d'analyse d'image : détection, identification, action sur l'environnement, etc. En fonction des objectifs fixés, les caractéristiques pertinentes des scènes observées sont synthétisées sous la forme d'un modèle. Ce dernier résume l'ensemble des connaissances sur le domaine d'application et intègre les contraintes imposées.

L'interprétation, phase ultime de l'analyse d'image, procède à une mise en correspondance des primitives d'origine sensorielle avec des instances du modèle.

On peut interpréter une image en utilisant son histogramme. En fait, Les histogrammes [31] sont des indicateurs de répartition de niveaux de gris (ou de couleurs) dans une image. Ils sont très utilisés en recherche par le contenu car l'histogramme d'une image est presque invariant en rotation, translation et changement d'échelle de cette image.

2.2 Principe des capteurs CCD et CMOS

Fichier Capteur CCD

Les Capteurs CMOS

les capteurs CMOS (complementary metal oxyde semiconductor) sont néanmoins basés sur le même principe d'absorption de photons et de génération de courant proportionnel au flux incident sur chaque pixel. La différence provient de la façon dont les charges sont lues.

Contrairement aux capteurs CCD, les charges sont cette fois directement converties au niveau du photosite de génération par le biais d'un amplificateur sur chaque pixel (figure 4). Cette particularité permet de supprimer un grand nombre de transferts comme dans le cas des registres à décalage du CCD et donc d'augmenter la cadence de lecture. La taille des capteurs est voisine de celle des CCD mais la résolution spatiale est moindre, du fait de la présence des amplificateurs associés à chaque pixel (un peu comme dans le cas des CCD interlignes). On peut là encore adjoindre une matrice de microlentilles pour compenser en partie cet effet.

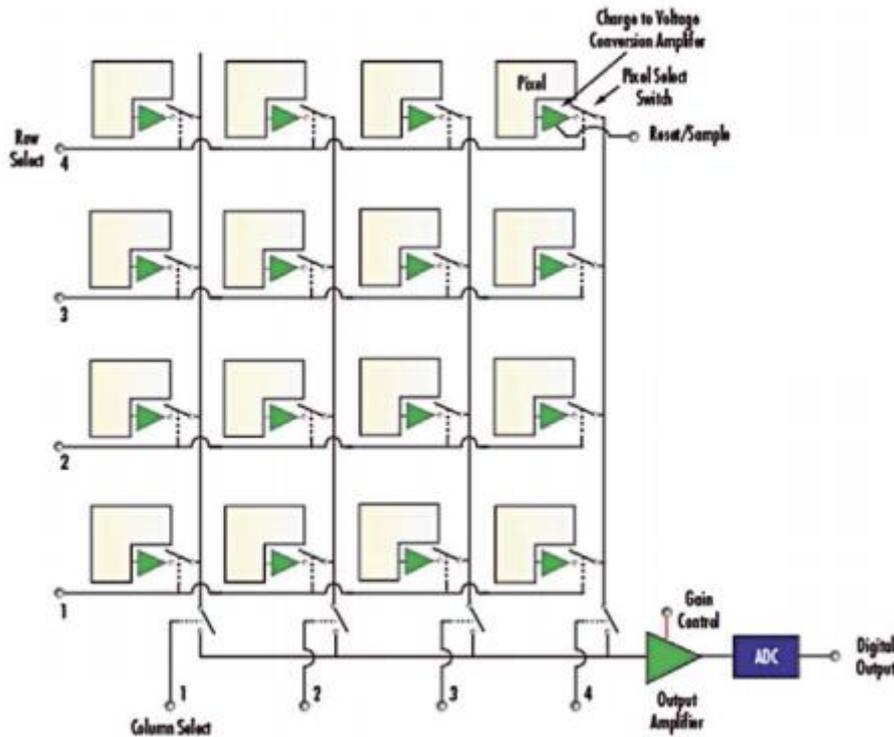
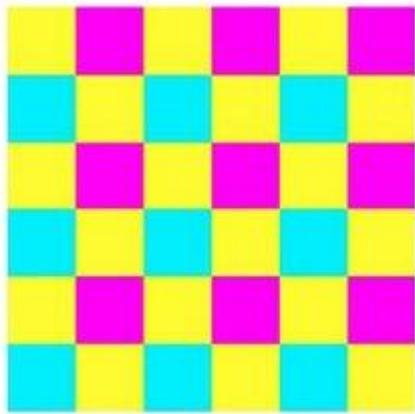


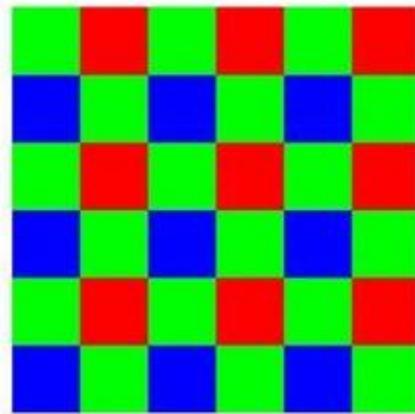
Schéma de l'architecture d'un capteur CMOS. Chaque photosite dispose de son propre amplificateur. Chaque pixel peut donc être adressé individuellement et possède une réponse propre indépendante de ses voisins.

2.3 Spécifications des capteurs couleur

La capture d'image en couleurs est parfois requise. Principalement, deux techniques sont privilégiées pour numériser la couleur : soit on décime la résolution d'un seul capteur par un réseau de filtre couleur (filtre de Bayer, 3 couleurs (Fig.2) ou plus (Fig.3), soit on décompose la lumière en plusieurs composantes avec un système de prismes qui fabriquent chacun l'image de la composante couleur de l'objet sur un détecteur dédié à cette couleur (Fig. 4).

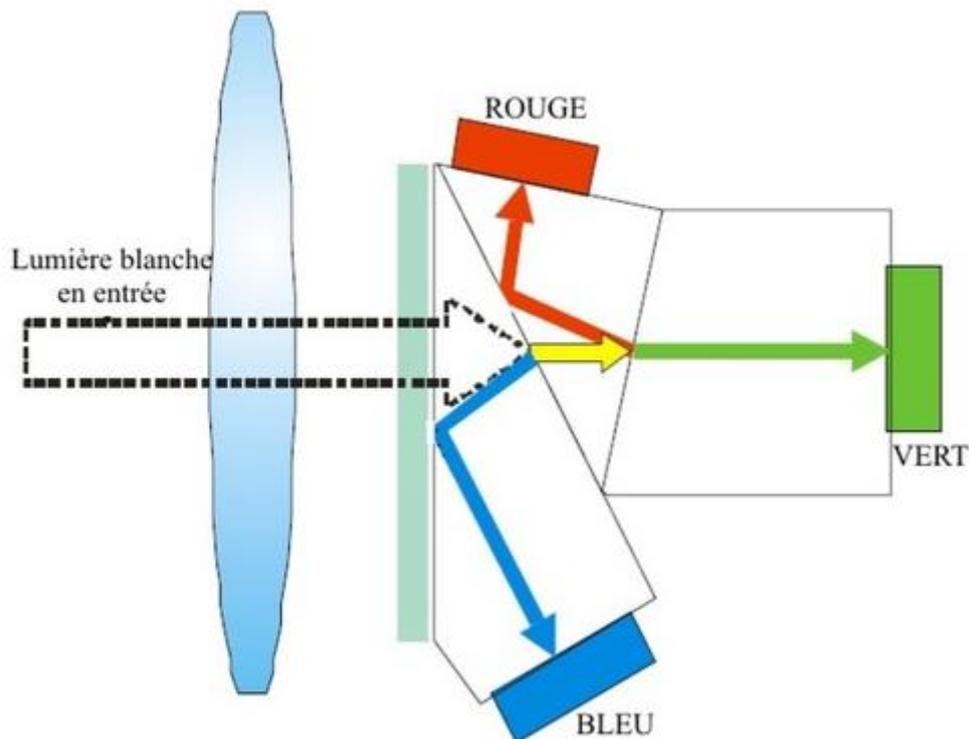


CMY Bayer Pattern
DCS 620x

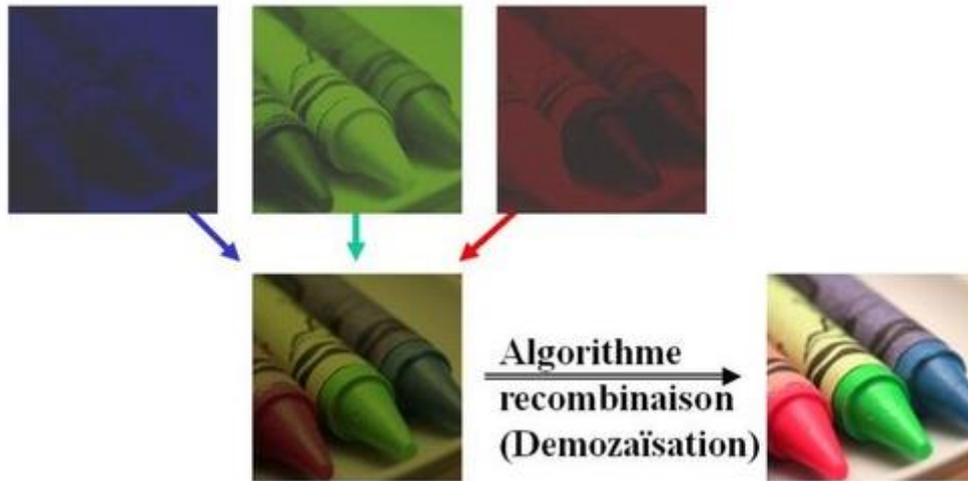


RGB Bayer Pattern
DCS 620

L'image finale est ensuite reconstruite par des algorithmes d'association des composantes, qui gèrent la sensibilité chromatique, le gain, et la décomposition spectrale du signal lumineux. Ces algorithmes assurent la « démozaïsation » de l'image.

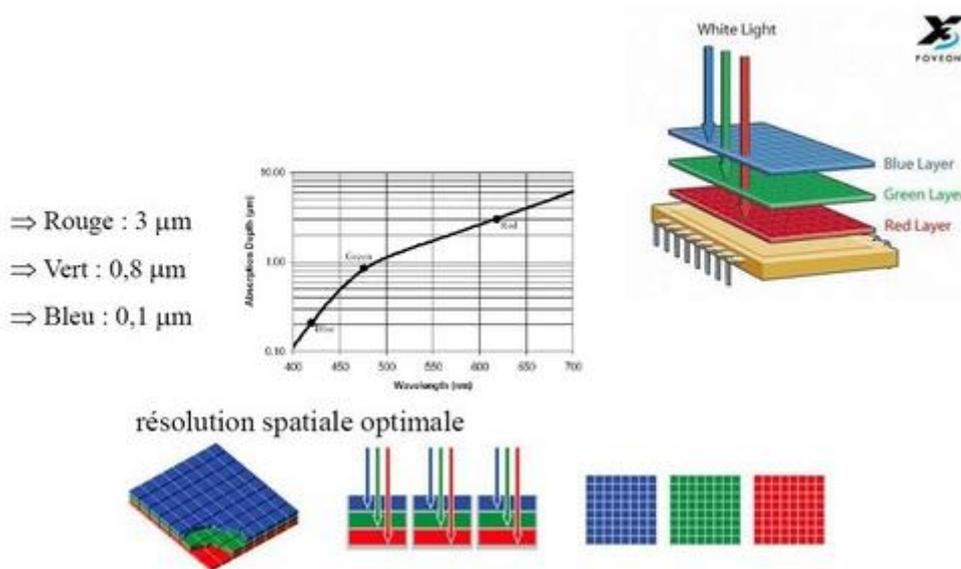


2.4 Séparation chromatique par prismes



Reconstruction d'une image en couleurs

Une dernière possibilité est d'empiler les diodes photosensibles et de réaliser le filtrage par une sélection de la longueur d'onde sur la profondeur du capteur.

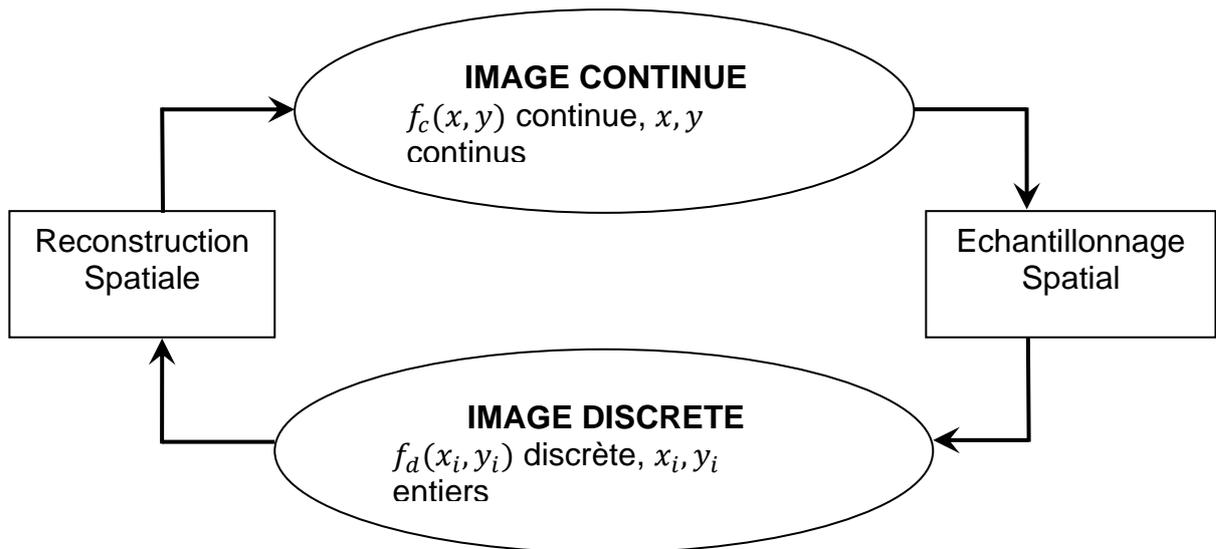


Principe du capteur Foveon

2.5 Numérisation d'une image

NUMERISATION D'IMAGES : ECHANTILLONNAGE ET QUANTIFICATION

L'image est physiquement continue. Pour avoir une image numérique, il faut échantillonner puis quantifier l'image continue. En respectant le théorème de Shannon, on peut reconstruire l'image originale à partir de l'image échantillonnée. La Figure 2.9 illustre cette réciprocity.



Echantillonnage d'une image

Rappel sur Le Peigne de Dirac

Cas continu

$$\delta_0(x) = \begin{cases} \infty & \text{si } x = 0, \\ 0 & \text{sinon} \end{cases}$$

et $\int_{-\infty}^{+\infty} \delta_0(x) dx = 1$

Cas Discret

$\delta(0) = 1$

$\delta(n) = 0$ pour $n \neq 0$

et $\sum_n \delta(n) = 1$

Le peigne de Dirac est donné par :

$$S(x, y) = \sum_i \sum_j \delta(x + i\Delta x, y + j\Delta y)$$

Dans le domaine de Fourier :

$$S(u, v) = \sum_i \sum_j \delta(u + i\frac{2\pi}{\Delta x}, v + j\frac{2\pi}{\Delta y}) = \sum_i \sum_j \delta(u + i\Delta u, v + j\Delta v)$$

avec $\Delta u = \frac{2\pi}{\Delta x}$ et $\Delta v = \frac{2\pi}{\Delta y}$

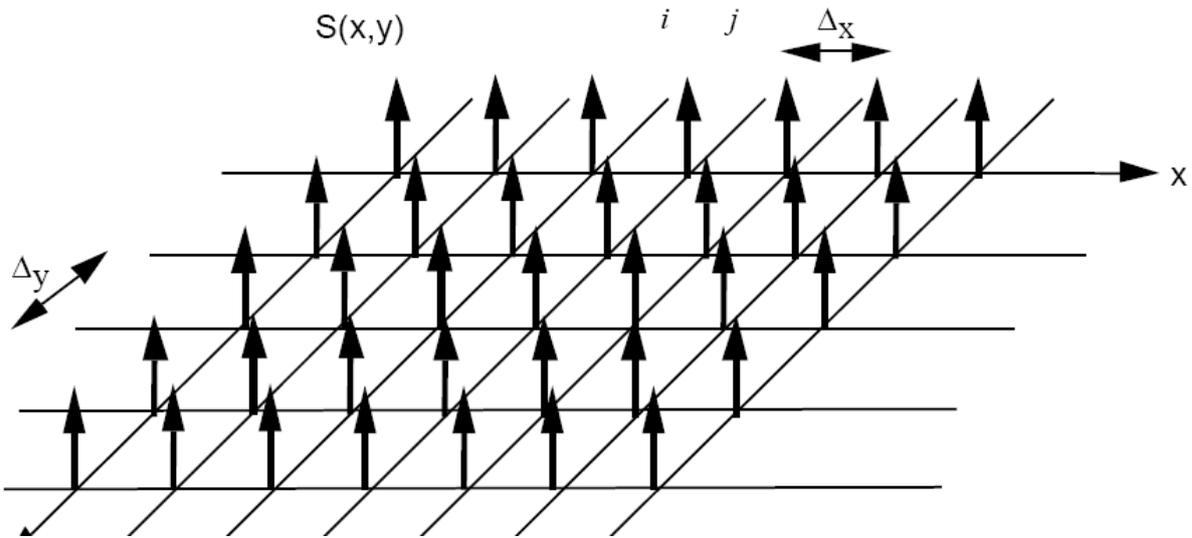


Fig. 2.10 peigne de Dirac

2.3.2 Echantillonnage

Rappelons pour mémoire que l'échantillonnage est un prélèvement périodique des échantillons d'un signal analogique comme c'est illustré dans la Figure 2.11.

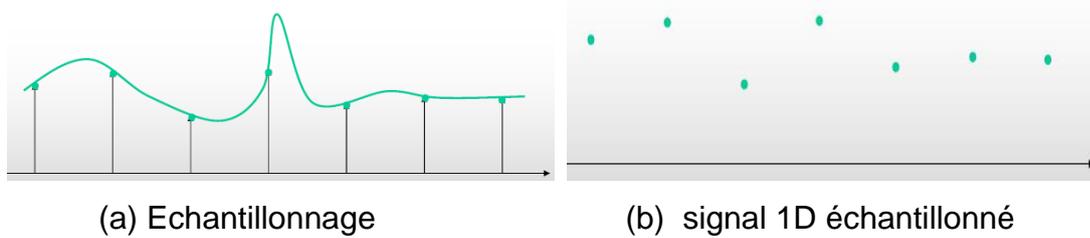


Fig.2.11 Exemple d'échantillonnage d'un signal 1D

Dans le cas 2D tel que les images, l'échantillonnage s'effectue par le produit de la fonction continue $f_c(x,y)$ et le peigne de Dirac 2D $S(x,y)$. L'image échantillonnée $f_d(x,y)$ est donc obtenue par

$$f_d(x,y) = f_c(x,y) \cdot S(x,y)$$

Dans le domaine de Fourier le produit classique devient une convolution :

$$F_d(u,v) = F_c(u,v) * S(u,v)$$

C'est donc un repliement du spectre $F_c(u,v)$.

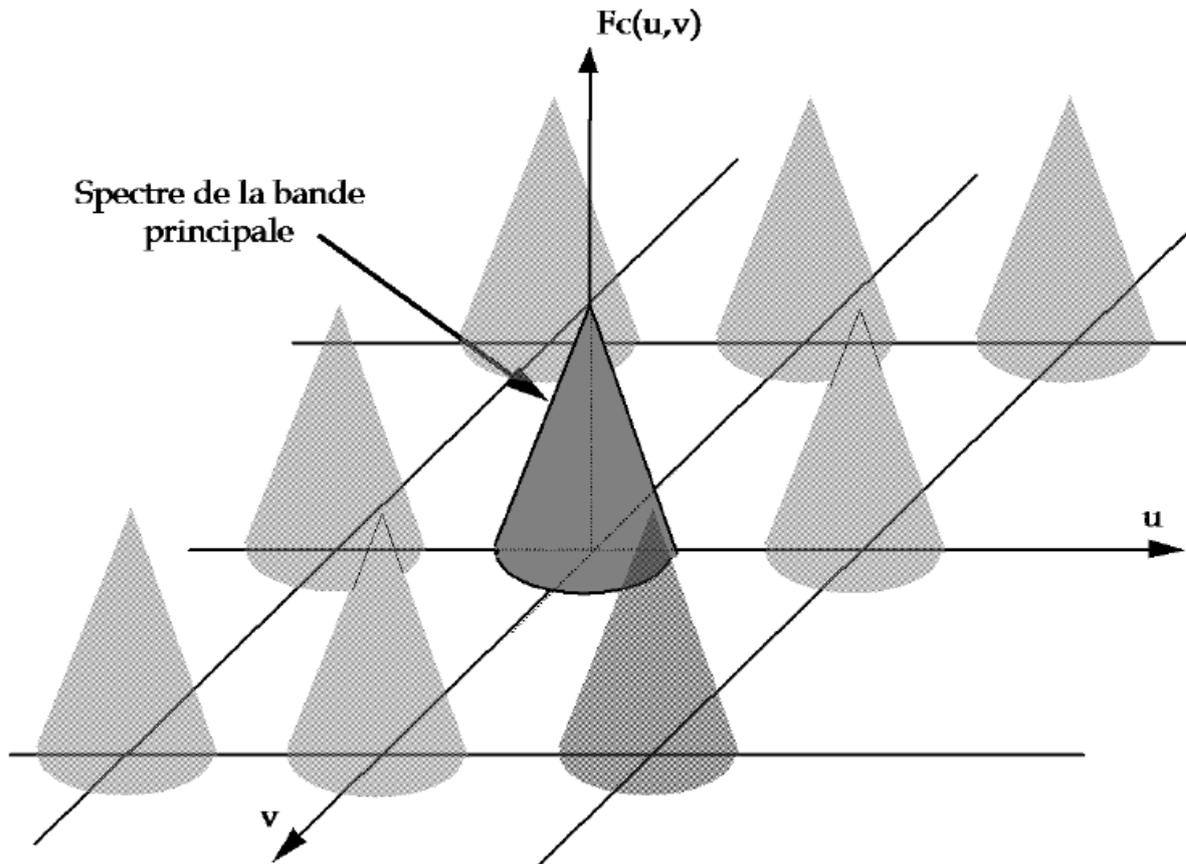


Fig.2.12 repliement du spectre $F_c(u, v)$.

Reconstruction

Théorème de Nyquist-Shannon

Soit un signal 1D, de support infini, à bande limitée (fréquence maximale finie f_{max}). Ce signal analogique peut entièrement reconstruit à partir de ses échantillons pour autant que les fréquences d'échantillonnage f_e soient au moins deux fois plus grandes que les fréquences maximales du signal.

$$f_e \geq 2f_{max}$$

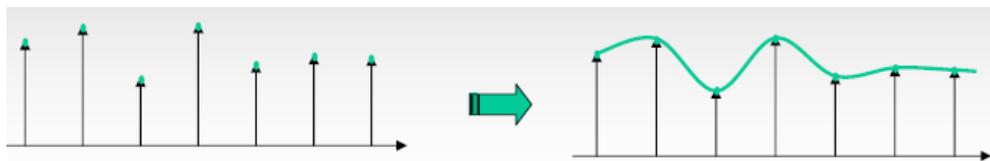


Fig.2.12 reconstruction du signal 1D à partir de ses échantillons

Le signal $f(x)$ est reconstruit à partir de ses échantillons, notés f_i dans l'intervalle Δx , (en respectant la condition de Shannon $f_{max} < 1/(2\Delta x)$ avec $f_e = \frac{1}{\Delta x}$) :

$$f(x) = \sum_{i=-\infty}^{+\infty} f_i \frac{\sin \left[\pi \frac{(x - i\Delta x)}{\Delta x} \right]}{\pi \frac{(x - i\Delta x)}{\Delta x}} = \sum_{i=-\infty}^{+\infty} f_i \text{sinc} \left(\pi \frac{(x - i\Delta x)}{\Delta x} \right)$$

Pour une image, cas 2D, $f(x, y)$ est reconstruite à partir des échantillons $f_{i,j}$

$$f(x, y) = \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} f_{i,j} \frac{\sin[\pi(x-i)]}{\pi(x-i)} \times \frac{\sin[\pi(y-j)]}{\pi(y-j)}$$

avec $\Delta x = \Delta y = 1$.

Numérisation d'une Image

Une image est un signal bidimensionnel. Une image analogique est par exemple celle formée sur la rétine de l'œil ou l'image obtenue par la photographie argentique classique. Une image numérique est un signal numérique composé d'unités élémentaires (appelées pixels) qui représentent chacun une portion de l'image.

Une image digitale (numérique) = Description dans un espace discret de R^2 d'une image continue $I(x, y)$.

Un pixel est l'unité indivisible permettant de coder l'information relative à la luminosité en une certaine position. Pixel vient de « picture element ».

Contrairement au cas unidimensionnel, nous étudierons uniquement les images numériques (discrètes). Une image numérique est définie par :

- le nombre de pixels qui la composent en largeur et en hauteur (qui peut varier presque à l'infini).
- l'étendue des teintes de gris ou des couleurs que peut prendre chaque pixel (on parle de dynamique de l'image). Un pixel est donc composé :
 - de coordonnées (i, j) permettant de le situer
 - d'une valeur $l = p(i, j)$ représentant sa couleur ou son niveau de gris.

La Figure 1.4 illustre les pixels et les niveaux de gris. I est l'image numérique

$I[i, j] = l$ est la valeur du niveau de gris.

Lorsque $n \in [l_{min}, l_{max}]$, $l_{max} - l_{min}$ est le nombre de niveaux de gris.

Dynamique de l'Image

La dynamique de l'image est donnée par :

$$D = \log_2(l_{max} - l_{min})$$

Les amplitudes peuvent être :

- Réelles (voir complexes)

- Entiers représentant la quantification des valeurs sur un ensemble discret (exemple: 24 bits).

Si on a une image de N lignes et M colonnes (de taille $N \times M$) et de plus $M = N = 2^B$, on aura L niveaux de gris, tel que $L = 2^B$

$B = \log_2 L$ est le nombre de bits. On dit que chaque niveau de gris est codé sur B bits.

Si $B = 1$, on aura $L = 2$ niveaux de gris ; c'est une image binaire.

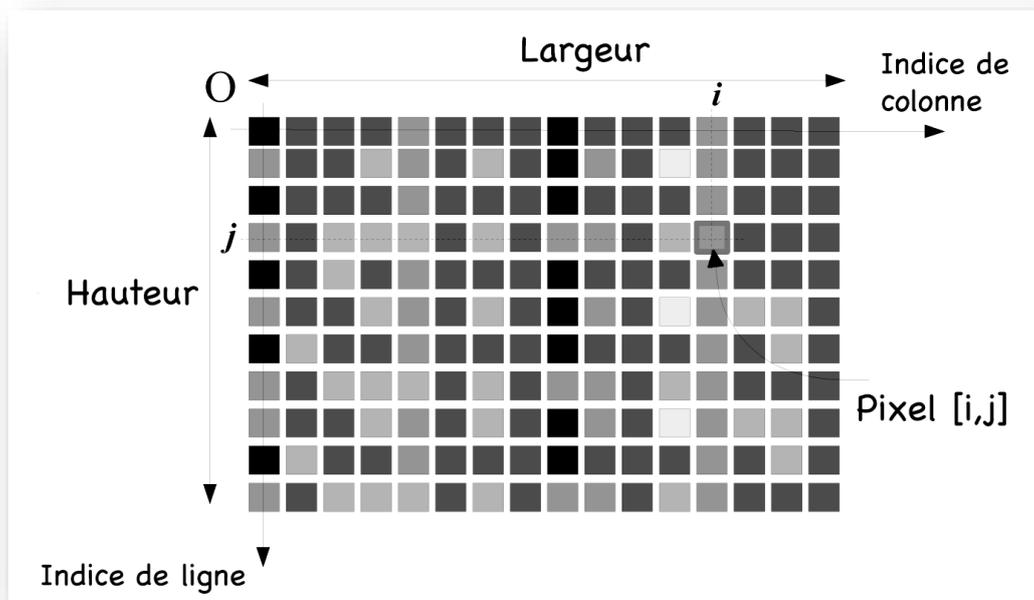


Fig. 1.4 Pixels et niveaux de gris

1.3.1 Images Binaires (Noir ou Blanc)

Dans une image binaire, un pixel peut prendre uniquement les valeurs noir ou blanc. C'est typiquement le type d'image que l'on utilise pour scanner du texte quand celui-ci est composé d'une seule couleur.

1.3.2 Images En Niveaux de Gris

En général, les images en niveaux de gris renferment (contiennent) 256 teintes de gris. Par convention la valeur zéro représente le noir (intensité lumineuse nulle) et la valeur 255 le blanc (intensité lumineuse maximale). Le nombre 256 est lié à la quantification de l'image. En effet chaque entier représentant un niveau de gris est codé sur 8 bits. Il est donc compris entre 0 et $2^8-1=255$. C'est la quantification la plus courante. On peut coder une image en niveaux de gris sur 16 bits ($0 \leq n \leq 2^{16} - 1$)

ou sur 2 bits : dans ce dernier cas « le niveau de gris » vaut 0 ou 1 il s'agit alors d'une image binaire (Noir et Blanc) comme c'est illustré dans la Figure 1.5.

1.3.3 Images Couleurs

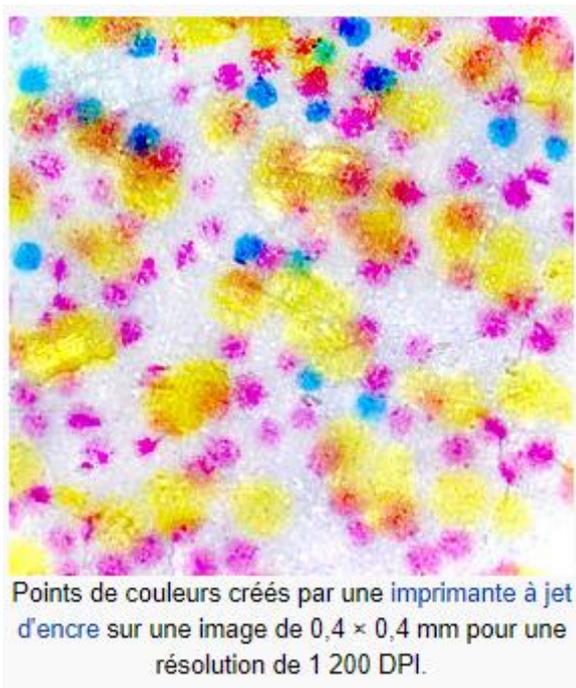
Une image est représentée comme la superposition de trois images N&B. L'espace couleur est basé sur la synthèse additive des couleurs, c'est-à-dire que le mélange de trois composantes (par exemple R,V,B) donne une couleur. On garde l'information couleur, ou intensité lumineuse et chromaticité. Un pixel est codé par trois valeurs numériques. La signification de ces valeurs dépend du type de codage choisi. Le plus utilisé pour le maniement des images numériques est l'espace couleur Rouge Vert, Bleu (R,V,B) (RGB en anglais).cet espace est représenté sur la Figure 1.6. La restitution des couleurs sur écran utilise cette représentation (synthèse additive).

Notions de définition, résolution et quantification d'une image numérique (taille, dpi, ppi, bpp ...etc)

Dpi=dot per inch

Ppp=point par pouce

Inch=2.54 cm



Nous avons 1200 points par pouce dans 0.16mm²

Pour 0.4 mm=0.04cm-→0.04/2.54=0.015 inches

Donc on a 0.015*0.015 inches=225*10⁻⁶ inches

On a 1200*225*10⁻⁶ par cm=0.27 p par cm

On a (1200/2.54)=472 ppcm (points par cm)

Définition = (Nb pixels sur la Longueur) × (Nb pixels sur la Hauteur)
 soit
 Définition = (Longueur de l'image x Résolution) × (Hauteur de l'image x Résolution)

Exemple :

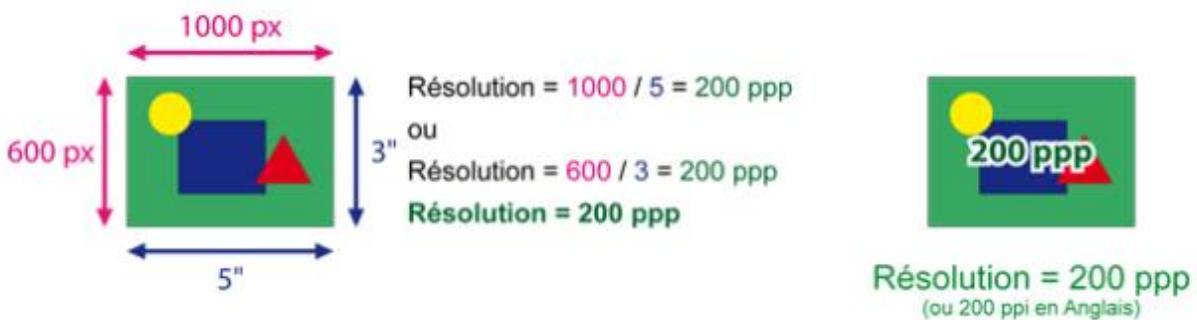


Calcul de la définition d'une image en fonction de sa résolution et de sa taille

Résolution de l'image

Résolution = définition sur la Longueur / Longueur de l'image
 ou
 Résolution = Définition sur la Hauteur / Hauteur de l'image

Exemple



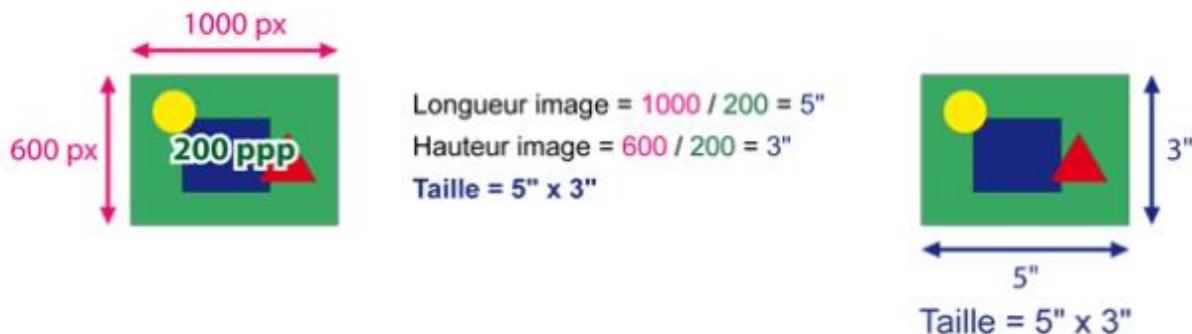
Taille de l'Image

Taille en Longueur de l'image = Définition sur la Longueur / Résolution

Taille en Hauteur de l'image = Définition sur la Hauteur / Résolution

Taille de l'image = (Définition sur la Longueur / Résolution) x (Définition sur la Hauteur / Résolution)

Exemple



Calcul de la taille d'une image en fonction de sa définition et de sa résolution

Notion de poids d'une image

Poids d'une image = nombre de bits (en ko, Mo, Go).

Exemple :

Si le pixel est codé sur 8 bits d'image de 512×512 pixels ----> le poids de l'image est $512 \times 512 \times 8$.

Exemples de formats d'images numériques (BMP, TIFF, JPG, GIF et PNG)

Il existe une centaine

- Le **BMP** : bitmap, une matrice de bits, codé en couleur jusqu'à 24 bits par pixel li uniquement sur des logiciel Windows
- Le **GIF** : adapté au transfert par le réseau Compuserve format compressé avec codage couleur sur 8 bits/pixel bien adapté aux images graphiques (les logos).
- Le **JPG** (jpeg) : compression d'images photographiques compression DCT.
- L'**EPS** : format postscript avec codage couleur sur 8, 16, 24 ou 32 bits.
- Le **PSD** : format correspondant au logiciel Adobe Photoshop format très répandu, peut coder la couleur sur 8, 16, 24 ou 32 bits.

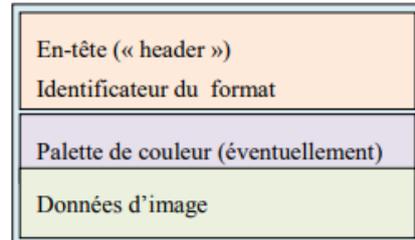
Exemple :

« 800 × 600 » signifie une largeur de 800 et une hauteur de 600 pixels.

« 600 dpi » (« ppp » : point par pouce, « dpi » : dots per inch) signifie 600 pixels par pouce (1 pouce = 25,4 mm).

En connaissant le nombre de pixels d'une image et la mémoire nécessaire à l'affichage d'un pixel, il est alors possible de définir exactement la taille qu'occupe le fichier.

Une définition du type 800 × 600 avec un codage sur 24 bits (3 octets) des couleurs, donne un fichier image de taille 1,44 Mo.

**Le modèle matriciel « bitmap »**

Une matrice de pixels (« picture elements ») est constituée d'un ensemble de points pour former une image. Le pixel représente ainsi le plus petit élément constitutif d'une image numérique.

- Formats limités à 256 couleurs : GIF, PCX, PGM, etc.
- Formats acceptant différentes quantifications BMP, TIFF, TGA, PNG, etc.
- Formats limités à 16 millions de couleurs JPEG, etc.

Le fichier cours-codage des images