

Affichage d'une image

Une image est un tableau bidimensionnel dont chaque élément (le pixel) représente une surface élémentaire de l'image (le pavé). La disposition des pixels (le maillage) est généralement en ligne et colonne (maillage rectangulaire); quelques applications peuvent présenter un maillage différent (utilisation de pavés hexagonaux par exemple).

Image noir&blanc

Chaque élément du tableau représente un *niveau de gris* ; la grandeur associée est un *nombre* représentant une échelle de gris . L'échelle démarre à zéro (le noir) et fini à un maximum (le blanc); ce maximum est soit normalisé à l'unité ("1" représente le blanc) soit fixé à une valeur (par exemple "100" ou "255").

Pour stocker le niveau de gris, on utilise généralement un nombre entier (type uint8 sur 8 bits ou uint16 sur 16 bits) plutôt qu'un scalaire Matlab (type double qui utilise 64 bits). Matlab gère ces formats de données directement.

Les images binaires sont une classe particulière dans laquelle il n'y a que deux niveaux : le noir "0" et le blanc "1" .

L'affichage de l'image à niveau de gris est direct pour Matlab, qui propose par défaut un affichage sur 64 niveaux (256 sur écran 24 bits) du blanc au noir

```
imshow(image)
```

Pour préciser l'échelle de gris, il suffit de donner les bornes d'affichage (qui peuvent être différentes des bornes de l'image).

```
imshow(image, [niv_min niv_max])
```

Ajustement automatique sur les niveaux extrêmes : `imshow(image, [])`

```
imshow(I1)
```



```
imshow(I1,[0 100])
```



```
imshow(im2bw(I1))
```

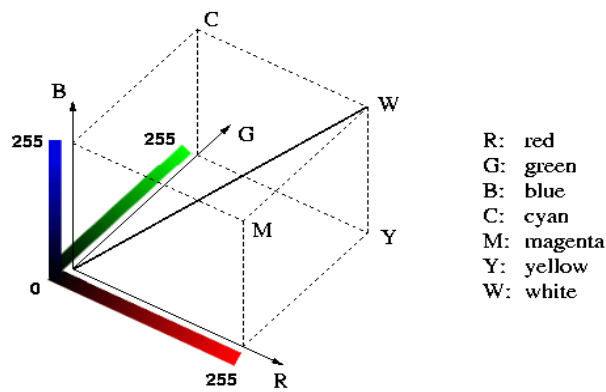


Image couleur

Chaque élément du tableau représente une couleur. Il existe deux modes essentiels de représentation:

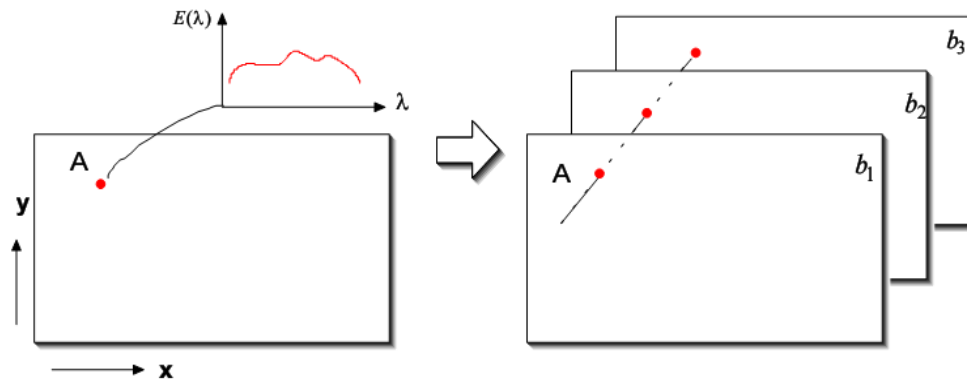
1) le mode *couleur vraie*: la couleur est représentée par un triplet (R,V,B) ou (R,G,B) en anglais, chaque composante représentant un niveau de couleur primaire, dans un espace [0..1] ou [0 255].

La somme vectorielle des trois composantes primaires donne une couleur unique dans l'espace des couleurs.



Pour des composantes 8bits (256 valeurs possibles), il existe $2^{3 \times 8} = 16 \times 10^6$ couleurs accessibles.

Une image couleur RGB est donc un tableau multidimensionnel à 3 dimensions.



Exemple:

soit `Im1` une image couleur

`Im1(10,150,2)` désigne la composante verte du pixel de coordonnées (10,150) ;

`coul = Im(230,192, :)` désigne le triplet couleur du pixel de coordonnées (230,192).

Les écrans d'affichage étant formés de 3 luminophores (R,V,B), il y a correspondance directe entre le contenu de l'image et son affichage. L'affichage d'une image RGB est donc direct par la fonction `imshow` .

Note importante : il existe d'autres espaces de représentation des couleurs; ces espaces comprennent toujours 3 composantes indépendantes (espaces TLS, HSV, Luv, Lab...) .Matlab propose un certain nombre de fonction de conversion d'un espace vers un autre. La fonction d'affichage `imshow` ne gère pas les autres types.

2) le mode **couleur indexée**: pour diminuer la taille informatique des images couleurs, le nombre des couleurs accessibles est limité à une *palette* (256 couleurs RGB par exemple). L'information de chaque pixel (un entier sur 8 ou 16 bits) renvoie à l'une des couleurs de la *palette*; ce contenu du pixel est donc le numéro d'index de la couleur sur la *palette*.

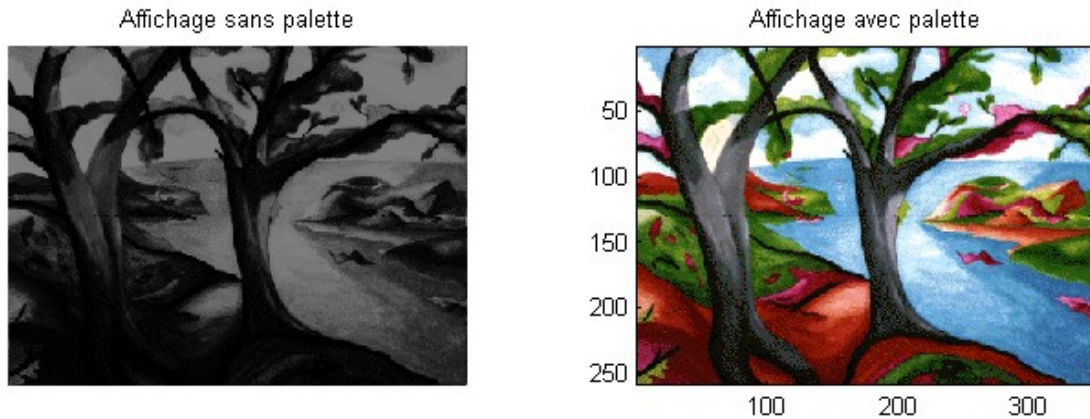
La *palette* est un élément essentiel de l'information puisque sans elle, l'image n'est qu'un tableau de nombre sans relation avec une couleur ou un niveau de gris. Il est donc indispensable de récupérer la palette lors de la lecture d'un fichier image.

```
[Im1,Palette1] = imread('trees.tif') ;
imshow(Im1,Palette1)
```

Attention : la palette affecte toute la figure (y compris les autres images en cas de subplot)

Affectation d'une palette différente à chaque sous-image :

```
[Im1,Palette1] = imread('trees.tif') ;
subplot(1,2,1), imshow(Im1) ; title('Affichage sans palette')
subplot(1,2,2), subimage(Im1,Palette1); title('Affichage avec palette')
```

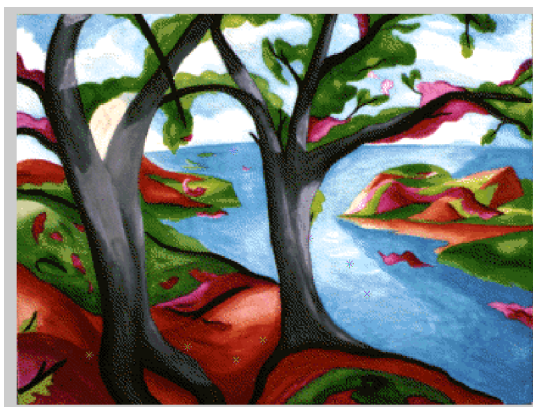


Il est possible de définir une palette d'affichage directement sous forme d'un tableau de 256x3 données (256 triplets RGB) pour un affichage particulier.

Utilisation de la souris dans une image:

Il est très simple de récupérer les niveaux de gris, index ou couleurs RGB de pixels sélectionnés interactivement à la souris. La fonction `impixel` retourne sous forme d'un tableau de n ligne la liste des informations concernant les n pixels sélectionnés.

```
[Im1,Palette1] = imread('forest.tif') ;
imshow(Im1,Palette1)
P = impixel(Im1) %retourne les index des pixels sélectionnés
P = impixel(Im1, Palette1) %retourne les triplets RGB des pixels sélectionnés
```



```
P =
0.4196 0.4196 0.4196
0.7412 0.9059 1.0000
0.2588 0.5176 0.7098
0.0941 0.2588 0.0314
0.1608 0.3529 0.0627
0.2235 0.4510 0.6784
0.6784 0.1608 0.0941
0.2588 0.4196 0.2588
```

```
[num_ligne,num_colonne,P] = impixel(Im1,Palette1) %retourne les coordonnées et la
couleur des pixels.
```