

Codage numérique : du nombre au pixel - Cours 5

Codage des images

L1 – Université de Lorraine
B. Girau et N. de Rugy Altherre

Transparents disponibles sur la plateforme de cours en ligne

Pixel

Le *pixel* (*P*icture *E*lement) est l'unité de base d'une image numérique matricielle.

Un pixel ne contient qu'une couleur.

des images

Pixel

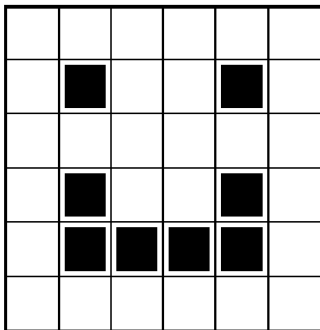
Le *pixel* (*P*icture *E*lement) est l'unité de base d'une image numérique matricielle.

Un pixel ne contient qu'une couleur.

La résolution d'une image est le nombre de pixels de l'image. Un écran de résolution 1280×1024 affichera 1280 lignes et 1024 colonnes, soit 1 310 720 pixels.

Bitmap

Une image peut être découpée en pixels et codée par une matrice donnant à chaque pixel sa couleur. On parle alors d'image *matricielle* ou bitmap (carte de points).



1	1	1	1	1	1
1	0	1	1	0	1
1	1	1	1	1	1
1	0	1	1	0	1
1	0	0	0	0	1
1	1	1	1	1	1

source : thimbleprojects.org

Comment coder la couleur d'un pixel ? Si on associe 1bit par pixel, ils ne pourront avoir que 2 couleurs, noir ou blanc. Avec 2 bits les pixels peuvent avoir 4 couleurs, avec 3 bits 8 couleurs etc... Par exemple sur des nuances de gris :



1 bit



2 bits



4 bits

source : wikipedia

Image et couleurs à l'écran

Chaque couleur reproduite à l'écran (LCD, OLED, etc.) est obtenue par le mélange de trois couleurs primaires (Rouge, Vert, Bleu). On parle de modèle de couleur par synthèse additive des couleurs primaires.

Fondamentalement, la "quantité" de chaque couleur est une valeur réelle comprise entre 0 et la quantité maximale.

L'être humain ne sait distinguer que quelques dizaines de milliers à quelques centaines de milliers de couleurs différentes.

L'expression de la quantité de chaque couleur a été réduite à une valeur entière comprise entre 0 et 255. Cela permet de définir plus de 16 millions de couleurs, ce qui est largement suffisant.

N.B. : certaines technologies échantillonnent chaque couleur entre 0 et 65535, d'où presque 300 millions de milliards de couleurs.
Quel véritable intérêt ?

Image et couleurs stockées

Les images stockées sous forme de fichiers contiennent l'information des couleurs des pixels sous une forme encodée et potentiellement compressée.

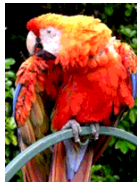
Encodage RGB

Le codage *RGB* code les couleurs par synthèse additive des trois couleurs primaires (Red, Green, Blue).

En fonction du nombre de bits utilisé pour chaque couleur, la quantité associée est plus ou moins échantillonnée entre 0 et la valeur maximale.



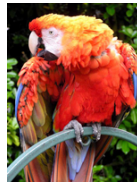
RGB 3 bits



9 bits



15 bits



24 bits

source : wikipedia

Où est l'encodage RGB tel que directement utilisé par les écrans ?

Autres encodages

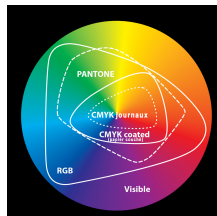
D'autres encodages de la couleur existent.

Chacun est associé à un modèle de couleur, plus ou moins proche :

- des mécanismes physiques (ondulatoires) générant les couleurs,
- de la technologie des media de visualisation,
- ou de la perception visuelle que nous avons des teintes.

Par exemple, CMY(K) est un encodage soustractif des couleurs, en lien direct avec les technologies d'impression (cf TD).

Attention, le choix d'un modèle de couleur ne permet pas toujours de représenter toutes les couleurs possibles.



Encodage "perceptif"

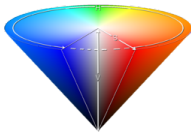
Il n'est pas toujours facile de faire le lien entre le code RGB d'une couleur et son "rendu". Un modèle souvent utilisé pour demander à un utilisateur de choisir une couleur est le modèle TSB (teinte, saturation, brillance, aussi appelé TSV, V pour valeur). Ce modèle est basé sur les caractéristiques les plus perçues par l'oeil humain.

Encodage "perceptif"

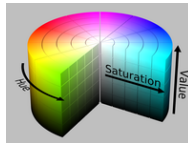
Le modèle TSB (en anglais HSB, Hue, Saturation, Brightness) définit les couleurs dans un cône d'axe z, avec la luminosité sur l'axe z, la saturation en rayon autour de l'axe z, et la couleur perçue répartie sur les angles 0 à 360 degrés.

Le noir est à la pointe du cône (i.e. en $(0,0,0)$), le blanc au milieu de la base (i.e. en $(0,0,1)$), et les couleurs "pures" situées sur le cercle autour de la base (i.e. en $(\cos a, \sin a, 1)$).

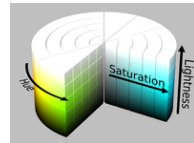
Un modèle proche est TSL/HSL (L : luminosité/lightness).



cône TSB



vue cylindre TSB

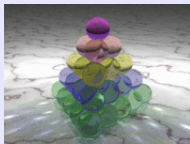


vue cylindre TSL

RGB α : gestion de la transparence

Dans de nombreux logiciels, une image peut être composée de couches, représentant par exemple un arrière-plan et des objets au premier plan.

Les couleurs des pixels des différentes couches sont alors munies d'un coefficient de transparence α .



Le rendu final au niveau d'un pixel de l'image est la somme de la couleur RGB de ce pixel dans la couche supérieure multipliée par $(1 - \alpha)$ et de la couleur RGB de ce pixel dans les couches inférieures multipliée par α .

Taille d'une image

$\text{taille image (octets)} = \text{nbPixels} * \text{nbOctetsParPixel}$

C'est le cas des images de format BMP (par exemple).

Par exemple, une image de 1280×1024 pixels codée en RGB 24 bits aura une taille de :

$$1280 \times 1024 \times 3\text{o} = 3932160\text{o} \approx 4\text{MO}$$

Plutôt que d'échantillonner régulièrement et uniformément les trois couleurs primaires, un encodage peut consister à définir un "catalogue" contenant un nombre réduit de couleurs, appelé palette.

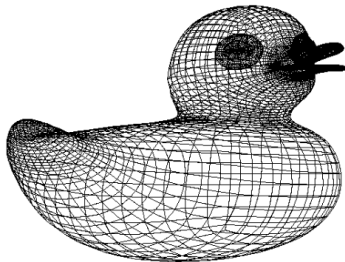
Palette de couleurs

Le fichier image contient alors la palette de 2^b couleurs, ou chaque couleur est définie en RGB, d'où 3×2^b octets, puis l'indication de la couleur de chaque pixel par son numéro dans la palette, codé sur b bits. La valeur $b = 8$ (256 couleurs) est souvent utilisée. On a alors : taille image (octets) = $3 \times 2^b + nbPixels * \frac{b}{8}$

Dans certains cas, une palette par défaut de 256 couleurs est proposée sur ordinateur. Il n'est plus utile de l'insérer dans le fichier image.

Image vectorielle

Une image vectorielle est décrite comme une juxtaposition d'objets géométriques individuels (arcs de cercle, segments de droite, courbe de Bézier, polygône etc).



Le logiciel d'affichage de l'image transforme alors la description mathématique dans une bitmap, aussi précise que l'on souhaite.

En résumé, pour coder une image, plusieurs choix doivent être fait, aboutissant à plusieurs formats d'image :

- Matricielle ou vectorielle ?
- Quel encodage pour chaque pixel (RGB 3 octets, 256 couleurs, etc.) ?
- Quelle compression ?

Compression

Compresser une image consiste à ne pas enregistrer les couleurs de tous les pixels séparément, en se basant sur la redondance des données.

Taux de compression : ratio entre la taille de l'image matricielle et la taille du fichier image compressé.

Exemple : un pixel bleu sur une image est souvent entouré d'autres pixels bleus (le ciel).

Compression sans perte

Une image peut être compressée sans perte : toutes les informations, sans être directement enregistrées, peuvent être retrouvées à partir de l'image compressée. Les redondances sont réduites. Le taux de compression dépend fortement de l'image elle-même.

Exemple : pour un bloc de 10×12 pixels d'une même couleur, au lieu de sauvegarder 120 couleurs, on sauvegarde les bords de ce bloc et 1 couleur.

Compression avec perte

Une image peut être compressée avec perte : des détails sont supprimés pour garder les informations essentielles de l'image. On choisit soit le taux de compression visé, soit la "qualité" de l'image (différence entre l'image d'origine et l'image compressée puis décompressée).

Exemple : un pixel de couleur (200,150,210) entouré de pixels de couleurs (200,150,200) peut être considéré comme un détail.