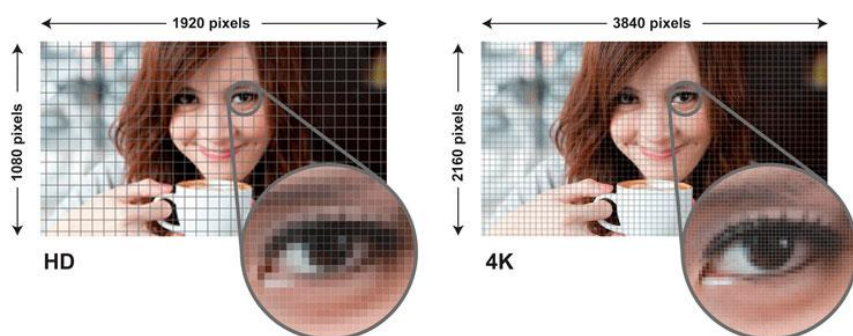


Activité

Codage des images
Fichier numérique d'une photographie

I – Rappels sur le codage des photographies numériques

Une image est caractérisée par le nombre de points, appelés pixels, qui la compose. Le pixel est donc l'élément le plus petit qui constitue une image. La **définition** est le nombre total de pixels constituant une image. Sur les deux exemples ci-dessous la définition est donc de 2 073 600 pixels (2 Mpx) en HD et de 8 294 400 pixels (8 Mpx) en 4K.



La **résolution** est la quantité de pixels par unité de longueur. Elle s'exprime en général en pixels ou points par pouce (ppi ou dpi, avec 1 pouce = 2,54 mm). Elle est utilisée pour connaître la qualité d'une image affichée sur un écran ou imprimée.

La profondeur de couleur :

Il s'agit de définir le **nombre de bits utilisé par pixel (bpp)**. Plus il est important, plus l'image peut afficher des couleurs différentes. Elle est comprise entre 1 et 32 bits.

Codage en noir ou blanc utilisant 1 bit/px :

Pour ce type de codage, chaque pixel est soit noir, soit blanc. Il faut donc un bit pour coder un pixel : 0 pour noir, 1 pour blanc. L'image codée de 10 000 pixels occupe 10 000 bits en mémoire, soit 1 250 octets. Ce type de codage est économe en mémoire.



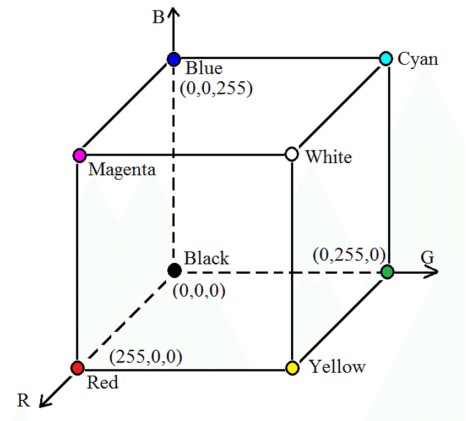
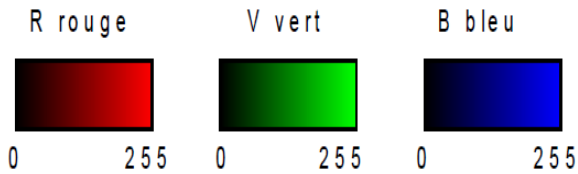
Codage en niveau de gris :

En général on code chaque pixel sur 8 bits (1 octet) pour ce réglage, on obtient donc 256 nuances de luminosité par pixel.



Codage en « couleurs vraies » utilisant 24 bits/px :

Il s'agit du codage le plus utilisé. On procède par mélange des trois couleurs fondamentales ; le rouge, le vert et le bleu (RVB). Chacune de ces couleurs sera codée sur 8 bits, soit **une consommation de 3 octets par pixel**. Par conséquent, une image codée de 10 000 pixels occupe 30 000 octets en mémoire.



II - Codage et gestion des couleurs des images numériques

- Ouvrir le fichier **Hermione.bmp** avec un éditeur hexadécimal puis écraser 3 octets de données par des 0xFF à partir de l'offset 0x36 (le codage des pixels débutant à cet endroit du fichier). Ouvrez l'image sous Photofiltre puis recherchez l'emplacement de ce pixel blanc sur l'un des angles de l'image. En déduire le mode de remplissage du fichier numérique permettant de coder l'image.

| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00000000 | 42 | 4D | 36 | F2 | 2C | 02 | 00 | 00 | 00 | 00 | 36 | 00 | 00 | 00 | 28 | 00 |
| 00000010 | 00 | 00 | B0 | 10 | 00 | 00 | 20 | 0B | 00 | 00 | 01 | 00 | 18 | 00 | 00 | 00 |
| 00000020 | 00 | 00 | 00 | F2 | 2C | 02 | 12 | 0B | 00 | 00 | 12 | 0B | 00 | 00 | 00 | 00 |
| 00000030 | 00 | 00 | 00 | 00 | 00 | 00 | FF | FF | FF | 46 | 44 | 4A | 42 | 40 | 46 | 3F |
| 00000040 | 3B | 41 | 3B | 37 | 3D | 34 | 2C | 33 | 28 | 21 | 26 | 24 | 18 | 1E | 1E | 13 |
| 00000050 | 15 | 1F | 12 | 14 | 1F | 11 | 12 | 1C | 0E | 0F | 1A | 0E | 0E | 1B | 0F | 0F |
| 00000060 | 1A | 10 | 10 | 18 | 0E | 0E | 1B | 0F | 0F | 1A | 0E | 0E | 1A | 13 | 10 | 1B |

- Donner la définition de l'image numérique (nombre total de pixels). En déduire la taille du fichier en nombre octets, vérifier cette valeur.
- Ouvrez l'image couleurs.jpg » puis utilisez l'outil Pipette de couleur disponible à droite de votre fenêtre pour relever le niveau RVB des objets suivants :

(Donner les valeurs en hexadécimal et en décimal)

- Le rouge de la boule n°23
- Le jaune de la boule n°62
- Le fond blanc de l'image
- Le centre du « 0 » de la boule n°90



- Avec le menu Filtre > Couleur > Niveau de gris, transformer l'image en niveaux de gris (cocher option luminosité).
- Reprenez la pipette pour relever le niveau RVB des objets suivants :
 - Le centre du « 0 » de la boule n°90
 - Le fond blanc de l'image
- La formule standard donnant le niveau de gris en fonction des trois composantes RVB est :

$$\text{Gris} = 0,3 \times R + 0,59 \times V + 0,11 \times B$$

Vérifier que cette formule a bien été utilisée par Photofiltre

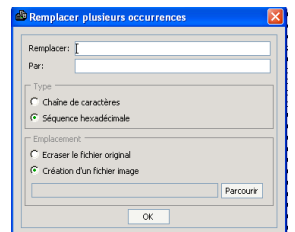
- 7- Annulez l'opération précédente (Ctrl-Z), allez dans le menu Réglage puis négatif. Relevez le niveau RVB du centre du « 0 » de la balle n°90, faites le lien avec les valeurs précédentes. Expliquez comment la conversion a eu lieu.
- 8- Exécuter le script « Négatif.py » sur une distribution Python puis observer le résultat de la transformation de l'image utilisée.
- 9- Adapter le code du script précédent afin de convertir une image couleur vers 256 niveaux de gris.

Principe de la Stéganographie

Le mot stéganographie vient du grec 'steganos' (caché ou secret). La majeure partie des fichiers informatiques (images, textes, sons, vidéos...) contiennent des zones de données non utilisées ou insignifiantes. Il est possible d'envoyer une image, un texte, un son, une vidéo ou un fichier quelconque à un correspondant sans qu'il sache ce qui se cache réellement à l'intérieur.

Nous prendrons l'exemple du fichier « pavillon.bmp » dans lequel un message est inscrit sur ce principe.

- 10- A l'aide de la commande « Remplacer plusieurs occurrences » disponible sous l'éditeur hexadécimal, échanger tous les codes « FE » par « 00 ». Sauver votre fichier puis observer cette nouvelle image. Commenter l'opération réalisée.



- 11- Proposer un script python capable de réaliser la même opération sur cette image.