

C31_TP Images numériques

Objectif : Comprendre la structure de données d'un fichier image

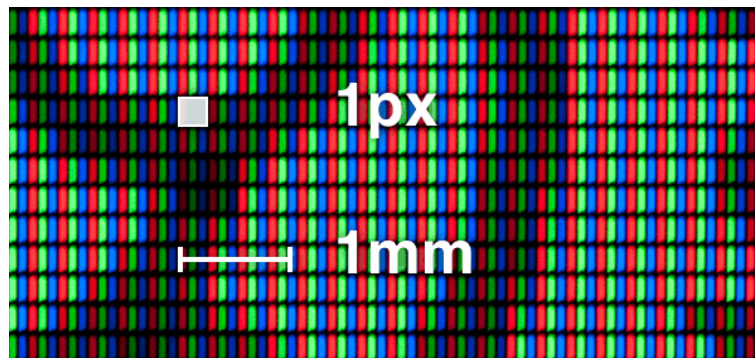
Partie 1 : Ecran numérique

Une image est composée de petits points appelés **pixels**.

Un **pixel** est composé de trois parties : une partie rouge, une partie verte et une partie bleue.

Ces sous-parties du pixel sont allumés ou éteint, avec une combinaison d'intensité relative qui détermine sa couleur.

La dimension d'un pixel : Elle dépend du terminal qui affiche l'image. On préfère alors donner l'information de la **densité** de pixels plutôt que la dimension d'un pixel :



Sur cet écran, 1 mm correspond à un peu moins de 4 pixels.

Cette densité est exprimée le plus souvent en pixels par pouce (en anglais pixels per inch, abrégé ppi).

Question a : Calculer la densité en pixels par pouces (ppi) à partir des informations de l'image ci-dessus.

Donnée : un pouce (ich) = 2,54 cm

Question b : Pourquoi l'œil ne parvient-il pas à distinguer les sous parties RVB d'un pixel ?

Partie 2 : synthèse additive des couleurs

À chaque pixel on associe donc 3 couleurs : le rouge, le vert et le bleu. On parle du canal rouge, du canal vert et du canal bleu d'un pixel (système RVB ou RGB en anglais).

La valeur de l'intensité lumineuse associée à chaque canal de chaque pixel d'une image est comprise entre 0 et 255 (256 valeurs possibles). On codera donc un pixel à l'aide d'un triplet de valeur (par exemple (247,56,98) en code décimal, ou son équivalent en hexadecimal : (f7,38,62)

Ouvrir le lien suivant : <http://www.proftnj.com/RGB3.htm>

Question c : à l'aide du logiciel :

Quelle est la couleur donnée par ce triplet de valeurs ?
(247,56,98)

Combien de couleurs différentes est-il possible d'obtenir avec ce système RVB ?

Comment obtenir du rouge ?

Comment obtenir du blanc ?

Comment obtenir du noir ?

Comment obtenir du jaune ?

Que se passe-t-il quand les trois canaux ont la même valeur (par exemple 125,125,125)

Partie 3 : Les données EXIF

Certaines informations, contenues dans le fichier image, donnent des informations sur les conditions de la prise de vue, le type d'appareil utilisé, la géolocalisation, les dimensions de l'image... Ces données suivent un format précis du standard EXIF.

Ouvrir la page : <http://exif.regex.info/exif.cgi>.

Uploader l'image fournie dans le dossier notebooks 'photo.jpg'

Le logiciel met ces données EXIF sous une forme lisible par un être humain :

Question c : compléter les informations contenues dans le fichier de cette photographie :

L'appareil photo utilisé

La lentille de l'objectif

Les coordonnées GPS du lieu où la photo a été prise

L'endroit où la photo a été prise (pays, region, commune) : utiliser le lien proposé vers openstreetmap par exemple.

La date de la prise de vue

Le nombre de pixels

Poids de la photo en ko Mo

Enfin, vous allez observer ces informations en chargeant le contenu des données EXIF dans une variable en python :

Ouvrir un notebook à partir du fichier ipython_notebook.ipynb du dossier TP_SNT_Exif

```
from PIL import Image
img = Image.open( 'NOM_DE_L' IMAGE' )
exif_data = img._getexif()
print(exif_data)
```

Les données EXIF sont stockées dans une variable d'un type particulier appelé « dictionnaire ».

Le dictionnaire rassemble la collection de donnée dans une même super-variable, un peu comme une liste. Mais à la différence de la liste, les valeurs sont associées à une clé. Cette clé peut être de type numérique, ou string. Pour y avoir accès, on n'utilise pas le rang, mais la clé : on fait : nom_de_la_variable[clé]

Dans l'exemple suivant, on définit un dictionnaire de nom « joueur ». Et on explore cette variable avec la clé « age » :

```
In : joueur = {'pseudo' : 'CR7', 'numero': 7, 'age' : 32, 'talent' :
'coups francs sur-puissants'}
In : joueur['age']
Out : 32
```

Question d : afficher la valeur associée à la clé 271 de la variable de type dictionnaire exif_data