



Lycée Julien Wittmer

SNT

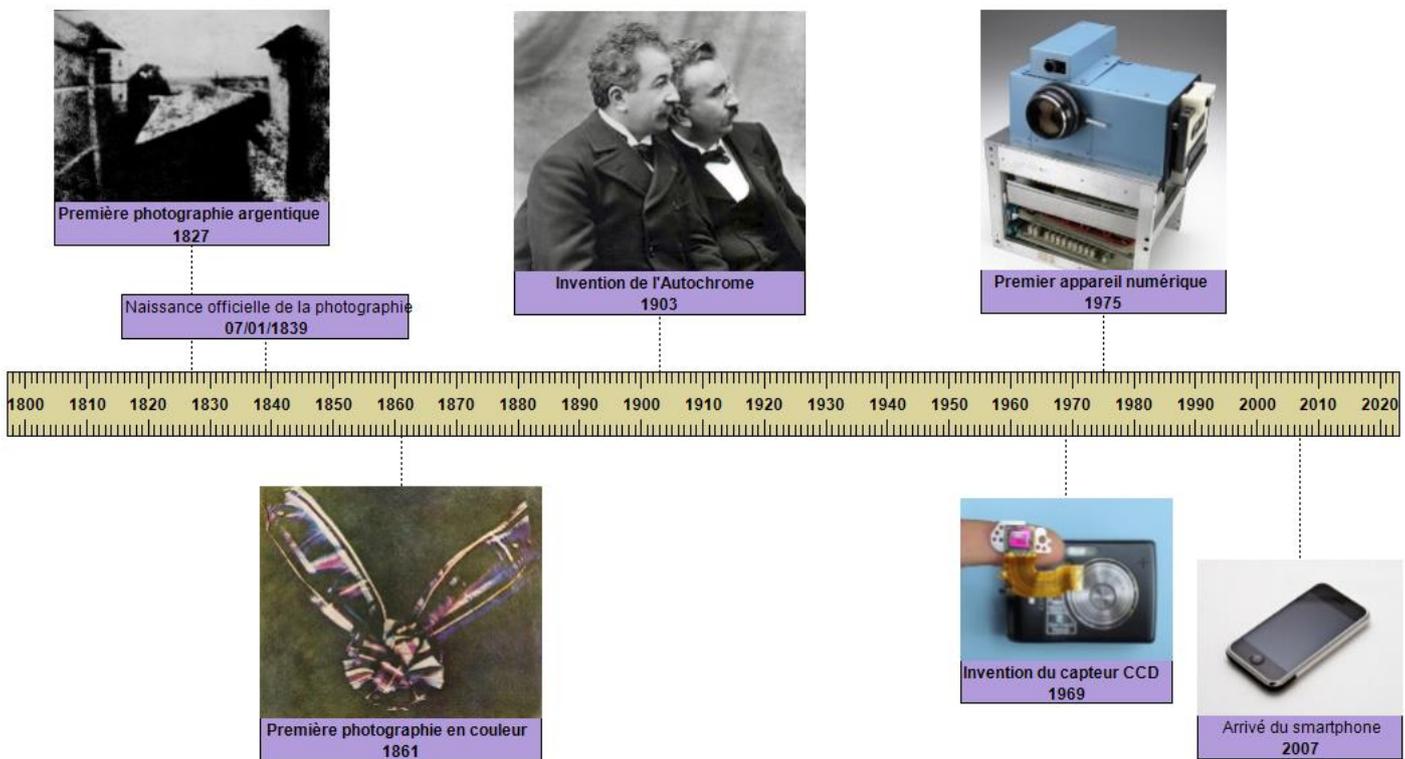
PHOTOGRAPHIE NUMÉRIQUE



- | | | |
|----------|-------------------------------|------|
| 1 | Repères historiques | p. 2 |
| 2 | Pixel, Définition, Résolution | p. 3 |
| 3 | Capteur photo | p. 4 |
| 4 | Métadonnées EXIF | p. 8 |
| 5 | Traitement d'image en Python | p. 8 |



Les repères historiques



1. Visionnez la vidéo : [Découvrons l'histoire de la photographie](#)
2. Complétez alors le cours suivant :

Cours : L'histoire de la photographie	
1827	fixe pour la première fois une image sur un support.
	Naissance officielle de la photographie, le jour de la présentation des travaux de Niépce et de son partenaire Louis Daguerre à l'académie des sciences.
1861	La première photographie en couleur est obtenue par l'anglais grâce à des prises de vue d'un ruban sous trois filtres différents : un filtre rouge, un vert et un bleu.
1903	Invention de l'autochrome par les frères Lumière, c'est la première technique industrielle de photographie couleur, elle produit des images sur plaques de verre.
1957	Première photo numérisée par .Cette technologie permet de transférer une photo argentique papier vers un ordinateur.
	Invention du capteur CCD (charge coupled device, en français « dispositif à transfert de charge) par le canadien Willard Boyle et l'américain George Smith. C'est un capteur qui transforme ce que vous voyez à travers votre viseur en une image numérique.
1975	Premier appareil photo numérique créé par pour Kodak.
	Arrivé du premier smartphone créé par Apple.



2

Cours : Pixel

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

A chaque pixel est associé 3 canaux : le rouge, le vert et le bleu. (on parle de système RVB ou RGB en anglais).

La valeur de l'intensité lumineuse associée à chaque canal est comprise entre 0 et 255

(Le nombre est codé sur 1 octet soit 8 bits ; il y a donc $2^8 = 256$ valeurs possibles).

Un pixel sera donc défini à l'aide d'un triplet de valeur (par exemple : 247, 56, 98).

La première valeur donnant l'intensité du canal rouge, la deuxième celle du canal vert et la troisième celle du canal bleu.



2

Activité 1 – Système RVB

NOVICE LEVEL

1. Combien de couleurs différentes est-il possible d'obtenir avec le système RVB ?
2. A l'aide de l'application proposée sur ce [site](#), faites varier les canaux rouge, vert et bleu (à l'aide des boutons + et -) afin d'obtenir différentes couleurs.
 - a. Comment obtenir du rouge ?
 - b. Comment obtenir du vert ?
 - c. Comment obtenir du bleu ?
 - d. Comment obtenir du blanc ?
 - e. Comment obtenir du noir ?
 - f. Que se passe-t-il quand les canaux ont la même valeur ?
3. En utilisant le même outil, donnez les couleurs correspondant aux codes suivants :
 - a. (0, 255, 255)
 - b. (255, 0, 255)
 - c. (255, 255, 0)
 - d. (255, 128, 0)



2

Cours : Définition, Taille et Résolution d'une image

Visualisez la [vidéo](#) suivante expliquant ce que sont la définition, la taille et la résolution d'une image.

La **définition** d'une image est le nombre de pixels qui composent l'image.

Une image de définition 800×600 signifie que cette image est composée de 800 pixels en largeur et de 600 pixels en hauteur, soit en tout $800 \times 600 = 480\,000$ pixels.

La **taille** correspond à la largeur et hauteur de votre image à l'impression.

Elle s'exprime en cm ou en pouces (inch en anglais).

La **résolution** est la quantité de pixels par unité de mesure. Cette valeur s'exprime en ppp (points par pouces) traduit en Anglais par dpi (dots per inch)

Ces trois informations sont liées. Si vous en connaissez deux, vous pouvez calculer la troisième.



2

Activité 2 – Résolution, Définition, Taille

INTERMEDIATE LEVEL

1. Soit une image de définition 800 x 533 que l'on imprime sur du papier photo de taille 15 x 10 (en cm), calculez la résolution de cette image en ppp (rappel : 1 pouce = 2,54 cm).
2. Sachant que l'on estime que pour avoir une impression de qualité il faut atteindre une résolution de 300 ppp, calculez la définition minimale d'une image dans le cas d'une impression sur papier photo 15 x 10.
3. L'écran d'un smartphone a une résolution de 458 ppp, il affiche des images de définition 2436 x 1125. Calculez la taille de cet écran (largeur, hauteur) en cm.

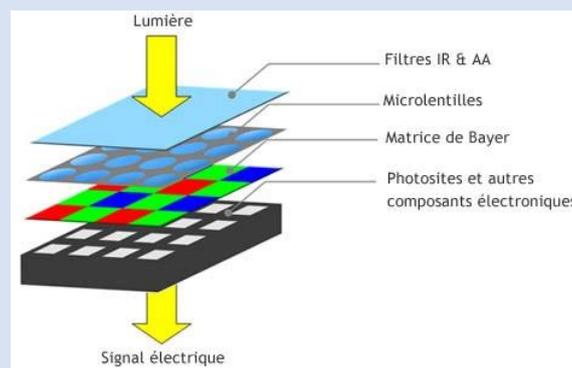


3

Cours : Capteur photo

Visualisez la [vidéo](#) suivante expliquant le fonctionnement d'un capteur photo.

Le **capteur photo** d'un appareil photo numérique remplace la pellicule d'un appareil photo argentique. Il est composé d'un grand nombre de **photosites**



Les **photosites** sont des petits éléments qui vont capter la lumière et convertir l'intensité lumineuse reçue en tension électrique. La tension électrique produite par un photosite sera ensuite "converti" en nombre (on parle de numérisation).

Si l'on se contentait de ce système, nous aurions uniquement des images en niveau de gris. Afin de pouvoir gérer les couleurs, on rajoute donc un filtre coloré (rouge, vert ou bleu) devant chaque photosite, afin de capter uniquement la couleur rouge, verte ou bleue. On obtient donc trois nombres qui correspondent à l'intensité de la lumière rouge, de la lumière verte et de la lumière bleue. En utilisant le système RVB, on peut recréer la couleur observée.



3

Activité 3 – Créer une image en Python

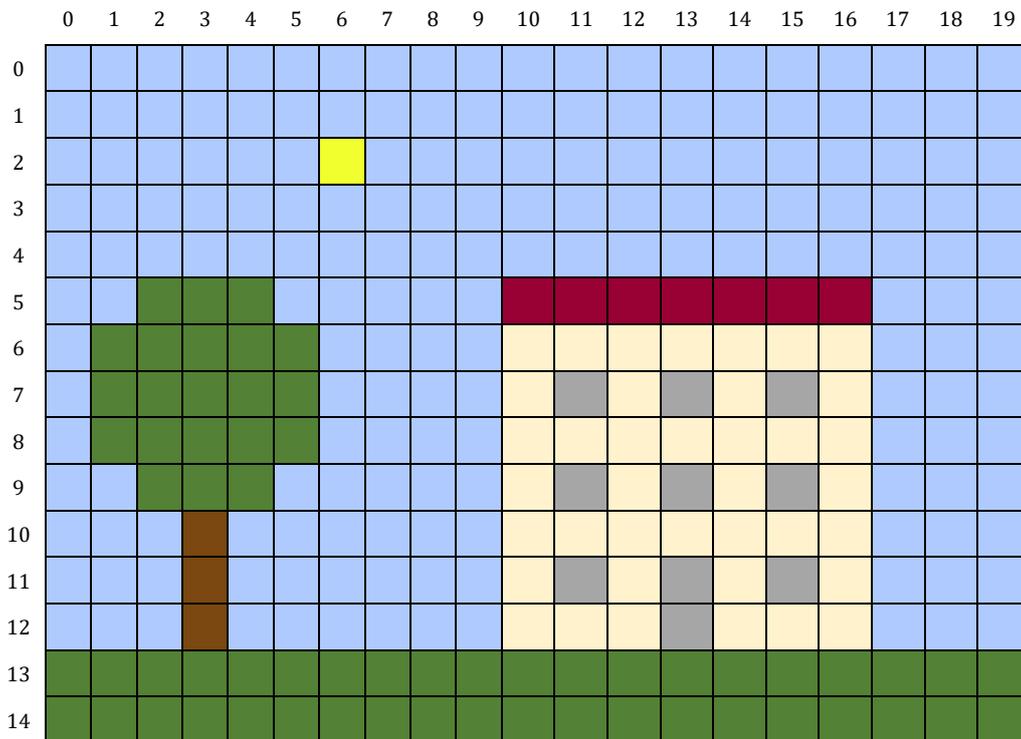
INTERMEDIATE LEVEL

Le but de cette activité est de créer en Python l'image ci-dessous dont la définition est de 20 × 15 pixels :

Deux choses sont à noter sur les coordonnées :

- Les coordonnées commencent à 0 (ce qui est courant en informatique) ;
- L'axe vertical est gradué de haut en bas (alors qu'habituellement en mathématiques, il est gradué de bas en haut).

Pour notre image de 20 × 15 pixels, les pixels ont donc une abscisse comprise entre 0 et 19 et une ordonnée comprise entre 0 et 14.



1. Récupérer les codes RVB des différentes couleurs qui composent l'image.

Utiliser l'outil capture de windows pour capturer l'image, enregistrer votre capture dans un dossier SNT_Photo dans vos documents puis ouvrez cette image avec PhotoFiltre. En utilisant l'outil pipette, compléter le tableau suivant :

Couleur	Code RVB
Ciel :	
Soleil :	
Herbe et feuilles de l'arbre :	
Tronc de l'arbre :	
Toit de l'immeuble :	
Murs de l'immeuble :	
Fenêtres et porte de l'immeuble :	

2. Programmation en Python.

Voici le programme python qui permet de commencer le dessin

```

1 from PIL import Image
2
3 img=Image.new('RGB',(20,15),(175,202,255))
4
5 img.putpixel((6,2),(240,255,45))
6
7 for x in range(10,17):
8     img.putpixel((x,5),(153,0,51))
9
10 for y in range(10,13):
11     img.putpixel((3,y),(122,72,16))
12
13 for x in range(20):
14     for y in range(13,15):
15         img.putpixel((x,y),(83,129,53))
16
17 img.save('immeuble.png')
    
```

Voici les explications pour chaque ligne :

<pre>1 from PIL import Image</pre>	Charge la fonction Image de la bibliothèque PIL (Pillow), cette ligne doit être présente au début de chaque programme.
<pre>3 img=Image.new('RGB',(20,15),(175,202,255))</pre>	Crée une nouvelle image, avec le système RGB, dont la définition est 20 pixels par 15 pixels, et dont le fond a pour code couleur RGB (175,202,255).
<pre>5 img.putpixel((6,2),(240,255,45))</pre>	<i>Cette ligne permet de dessiner le soleil</i> Colorie le pixel de coordonnées (6,2) avec le code couleur (240,255,45)
<pre>10 for y in range(10,13): 11 img.putpixel((3,y),(122,72,16))</pre>	<i>Ces lignes permettent de dessiner le toit de l'immeuble</i> Pour faire le toit de l'immeuble, il faut colorier les pixels dont l'abscisse x est comprise entre 10 et 16, et dont l'ordonnée est 5. Pour cela on utilise une boucle simple sur x. Attention : Lorsque x varie de 10 à 16 il faut taper : range(10,17) car le deuxième nombre est exclu.
<pre>10 for y in range(10,13): 11 img.putpixel((3,y),(122,72,16))</pre>	<i>Ces lignes permettent de dessiner le tronc de l'arbre</i> Cette fois ci, l'abscisse est toujours de 3 et c'est l'ordonnée qui varie entre 10 et 12, on fait donc une boucle simple sur y.
<pre>13 for x in range(20): 14 for y in range(13,15): 15 img.putpixel((x,y),(83,129,53))</pre>	<i>Ces lignes permettent de dessiner le sol vert</i> Pour dessiner le sol, on a besoin de faire varier à la fois x et y, on utilise donc une boucle double sur x et y.
<pre>17 img.save('immeuble.png')</pre>	A la fin du programme, on enregistre l'image obtenu sous le nom : immeuble.png. Le fichier sera placé dans le même dossier que celui où vous avez enregistré le programme.

- Ouvrir EduPython, copier le programme ci-dessous dans l'éditeur.
- Enregistrer votre programme dans votre dossier SNT_Photos sous le nom Immeuble.py.
- Compiler votre programme (en utilisant )
- Ouvrir le fichier Immeuble.png dans le dossier SNT_Photo puis terminer le dessin.

Faire valider par le professeur avant de passer à la suite.



3

Activité 4 – Les drapeaux

ADVANCED LEVEL

Écrivez des programmes permettant de créer les drapeaux suivants (Définition 300 x 200 pixels) :



Italie



Colombie



Bénin



Suède

On pourra utiliser ce [site](#) pour trouver le code RGB de la couleur souhaitée.

D'autres drapeaux, plus compliqués à créer (Pour les experts en Python)



Japon



Congo



République Tchèque



3

DM – PixelArt

INTERMEDIATE LEVEL

Demander la fiche à votre professeur pour réaliser le DM en respectant bien les consignes données.



4

Cours : Métadonnées EXIF

A chaque fois que vous prenez des photos avec un appareil photo numérique ou votre mobile Android, ce dernier enregistre tout un tas d'informations (**heure et lieu** de la prise de vue, **coordonnées GPS**, focale, temps d'exposition, type d'appareil etc...) et les ajoute automatiquement aux fichiers.

Ces informations sont les **Métadonnées EXIF**. (Exchangeable Image file Format)

Avec l'apparition des puces GPS dans les smartphones mais aussi dans les appareils photos numériques, il est aujourd'hui possible de localiser l'endroit précis où la photo a été prise. Si vous voyagez beaucoup, cela peut s'avérer très pratique pour se rappeler du lieu de la prise de vue. Par contre si vous avez l'habitude de partager vos photos sur internet, il est possible que vous ne souhaitiez pas divulguer l'endroit où celles-ci ont été prises.



4

Activité 5 - Métadonnées

NOVICE LEVEL

1. Téléchargez la photo pelican.jpg, puis ouvrez-la avec **photofiltre**.
2. Allez dans le menu : Fichier > Propriétés de l'image puis cliquer sur l'onglet EXIF.
En parcourant les données, retrouvez :
 - La date à laquelle a été prise cette photo :
 - La marque de l'appareil photo ;
 - Si le flash a été utilisé
3. On se demande où a été prise cette photo. Dans les métadonnées, on trouve les coordonnées GPS, la latitude est : 45 Degrees 46 Minutes 37 Seconds
Et la longitude : 4 Degrees 51 Minutes 18 Seconds
Ouvrez Google Maps et recherchez les coordonnées suivantes : 45°46'37"N 4°51'18"E
Quel est le lieu de la prise de vue ?
4. Déterminez où a été prise le photo statue-de-la-liberte.jpg ?



5

Activité 6 - Traitement d'image en Python

INTERMEDIATE LEVEL

1. Lecture des canaux (RVB) d'un pixel

Dans cette activité nous allons utiliser une photo de pomme de résolution 500 x 500.

Téléchargez le fichier pomme.jpg dans votre dossier **SNT_Photo**.

Après avoir ouvert **EduPython**, écrivez le programme suivant et l'enregistrez sous le nom pomme.py dans le même dossier que l'image pomme.jpg.



```

1 from PIL import Image
2 img = Image.open("pomme.jpg")
3 r,v,b=img.getpixel((1,1))
4 print("Rouge:",r,"Vert:",v,"Bleu:",b)
5

```

Voici l'explication de quelques lignes :

```
img = Image.open("pomme.jpg")
```

le programme ouvre le fichier pomme.jpg

```
r,v,b=img.getpixel((1,1))
```

Cette ligne récupère les valeurs du canal rouge (r), du canal vert (v) et du canal bleu(b) du pixel de coordonnées (1,1)

```
print("Rouge:",r,"Vert:",v,"Bleu:",b)
```

Cette ligne permet d'afficher le résultat

Exécutez le programme et affichez le résultat obtenu.

Modifiez le programme pour qu'il affiche les valeurs du canal rouge, du canal vert et du canal bleu du pixel de coordonnées (100,250).

2. Modification de la valeur des canaux d'un pixel

On considère le programme suivant :

```
1 from PIL import Image
2 img = Image.open("pomme.jpg")
3 (longueur,largeur)=img.size
4 for x in range(longueur):
5     for y in range(largeur):
6         r,v,b=img.getpixel((x,y))
7         img.putpixel((x,y),(v,b,r))
8 img.save("pomme2.jpg")
```

Voici l'explication de quelques lignes :

```
(longueur,largeur)=img.size
```

On récupère les dimensions de l'image que l'on stocke dans les variables longueur et largeur.

```
for x in range(longueur):
    for y in range(largeur):
```

Les boucles permettent de parcourir l'ensemble des pixels de l'image.

```
r,v,b=img.getpixel((x,y))
```

On récupère les valeurs des canaux rouge, vert et bleu du pixel de coordonnées (x,y)

```
img.putpixel((x,y),(v,b,r))
```

On remplace la couleur du pixel : on a mis la valeur du canal vert à la place de celle du canal rouge, celle du bleu à la place de celle du vert et celle du rouge à la place de celle du bleu

```
img.save("pomme2.jpg")
```

On enregistre le résultat dans un fichier nommé pomme2.jpg.

Recopiez et testez le programme précédent.

Modifiez le programme en réalisant d'autres inversions de couleur.



5

Activité 7 – Effet négatif d'une image

INTERMEDIATE LEVEL

Pour obtenir le négatif d'une image, il faut remplacer chacun des canaux par la valeur qu'il faudrait lui ajouter pour atteindre 255. Par exemple, si le canal rouge est 100, il faut remplacer par 255 – 100 soit la valeur 155.

1. Dans une image, un pixel a les valeurs suivantes : Rouge = 17 ; Vert = 127 ; Bleu = 255. Déterminez les valeurs des canaux Rouge, Vert et Bleu pour ce même pixel dans l'image négative.
2. Ecrivez un programme qui donne le négatif d'une image trouvée sur internet.



5

Activité 8 – Image en niveau de gris

INTERMEDIATE LEVEL

Pour qu'un pixel soit en niveau de gris, il doit avoir ses trois canaux RVB identiques.

Pour transformer une image en couleur en une image en niveau de gris, il faut donc attribuer aux trois canaux de chaque pixel la moyenne des canaux rouge, vert et bleu.

Par exemple, pour un pixel avec les canaux suivants : R = 97 ; V = 221 et B = 111

On attribue aux trois canaux R ; V et B la valeur : $\frac{97+221+111}{3} = 143$.

1. Déterminez la valeur qu'il faudra attribuer aux trois canaux d'un pixel vérifiant tel que : R = 157 ; V = 78 et B = 34 afin de le convertir en niveau de gris.
La valeur doit forcément être entière donc on arrondira le résultat à l'unité.
2. Ecrivez un programme qui transforme une image trouvée sur internet en niveau de gris.
Comme le nombre obtenu doit être entier, on utilisera la commande python `int()`.



Crédits

- Marie-Lorraine Lombard, Isabelle Noly, Josselin Darciaux, David Laurier, Patrick Meunier, Jean-Laurent Morati et Stéphane Moureaux
- Académie de Bordeaux
- Lycée des Flandres
- Louis Paternault