

Traitement d'image avec commentaires

(compléments d'informations à destination du professeur)

Raoul HATTERER

7 février 2019

Table des matières

1	Installation de PIL	1
2	La logique RVB	1
3	Image de départ	2
4	Comment lire un pixel	2
5	Comment écrire un pixel	2
6	Que fait le programme suivant ?	3
7	Passage d'une image en niveau de gris	3
	Source : Traitement d'image de l'académie de grenoble	

1 Installation de PIL

À faire au préalable par le professeur.

```
pip3 install pillow
```

2 La logique RVB

Activité de découverte du RVB : [colorsRGB](#)

Pour un pixel, calculer la moyenne des niveaux puis mettre les 3 valeurs à cette moyenne. Essayer pour plusieurs couleurs.

3 Image de départ



FIGURE 1 – Image de départ (480px × 300px)

4 Comment lire un pixel

Après avoir fait quelques recherches sur ce qu'est un "pixel", voyons comment lire le pixel de coordonnées (100,250).

```
from PIL import Image
img = Image.open("pomme.jpg")
r,v,b=img.getpixel((100,250))
print("canal rouge : ",r,"canal vert : ",v,"canal bleu : ",b)

('canal rouge : ', 19, 'canal vert : ', 88, 'canal bleu : ', 192)
```

5 Comment écrire un pixel

```
from PIL import Image
img = Image.open("pomme.jpg")
img.putpixel((5,5),(255,0,0))
img.show()
```

6 Que fait le programme suivant ?

```
from PIL import Image # Importation de la librairie PILLOW (gestion image)
img = Image.open("pomme.jpg") # Mise en mémoire dans la variable "img" le fichier pomme.jpg
largeur_image,hauteur_image=img.size # Méthode size appliquée à la variable img qui renvoie la largeur et la hauteur de l'image
for y in range(hauteur_image): # Pour chaque ligne de 0 à hauteur de l'image
    for x in range(largeur_image): #Pour chaque colonne de 0 à la largeur de chaque ligne
        rouge,vert,bleu=img.getpixel((x,y)) #Méthode getpixels appliquée à la variable img
        nouveau_rouge=vert #le vert devient rouge
        nouveau_vert=bleu #le bleu devient vert
        nouveau_bleu=rouge #le rouge devient bleu
        img.putpixel((x,y),(nouveau_rouge,nouveau_vert,nouveau_bleu)) #Méthode putpixel appliquée à la variable img
img.show() #affichage de l'image
img.save("pommeMystere.jpg") #sauvegarde de l'image obtenue
```

On analyse le code ci-dessus qui servira de base pour le défi suivant.



FIGURE 2 – Résultat du programme mystère

7 Passage d'une image en niveau de gris

Après avoir fait quelques recherches sur les "images en niveau de gris", écrivez un programme qui transforme une "image couleur" en une "image en niveau de gris".

Petite astuce qui pourrait vous aider : en Python pour avoir une division entière (le résultat est un entier), il faut utiliser l'opérateur `//` à la place de l'opérateur `/`

Remarque : On donne l'algorithme aux élèves (ou on le construit avec eux); ils doivent alors programmer le passage d'une image couleur à une image en niveaux de gris.

```
from PIL import Image
img = Image.open("pomme.jpg")
largeur_image=480
hauteur_image=300
for y in range(hauteur_image):
    for x in range(largeur_image):
        rouge,vert,bleu=img.getpixel((x,y))
        nouveau_rouge=(vert+bleu+rouge)//3
        nouveau_vert=(vert+bleu+rouge)//3
        nouveau_bleu=(vert+bleu+rouge)//3
        img.putpixel((x,y),(nouveau_rouge,nouveau_vert,nouveau_bleu))
img.show()
img.save("pommegrise.jpg")
```



FIGURE 3 – Image en niveaux de gris