

TP 5 : Traitement d'images en python Partie 2

Une image est un ensemble de pixels qu'on peut manipuler en python sous forme d'un tableau numpy. Lorsque l'image est en niveaux de gris, le tableau est bidimensionnel et contient des entiers entre 0 (Noir) et 255 (Blanc) donnant ainsi 256 intensités de niveaux de gris du pixel de coordonnée (i,j) où i est l'indice ligne et j est l'indice colonne. Cependant, les niveaux de gris peuvent être représentés par des valeurs réelles de 0 (Noir) à 1 (Blanc), ceci dépend du format de l'image ainsi que du module utilisé.

Lorsque l'image est en couleur, le tableau est à 3 dimensions. La troisième dimension contient une entrée à 3 valeurs [r,g,b] représentant le codage de la couleur (l'intensité des 3 couleurs : rouge, vert, bleu). Ce codage s'appelle RGB pour Red Green Blue.

Manipulation d'une image couleur :

1. Écrire une fonction python qui permet de charger l'image 'image.png' (fourni avec ce TP) et de retourner un tableau numpy I représentant cette image.

NB : on donne la fonction *imread* du sous module image de matplotlib qui permet de lire une image donnée en paramètre sous forme de fichier et retourne un tableau numpy représentant l'image.

2. Testez votre fonction en affichant le type, la dimension du tableau I ainsi que le dtype des éléments du tableau.
3. Écrire une fonction python qui permet d'afficher à l'écran une image donnée en paramètre.

NB : on donne les fonctions suivantes du sous module pyplot de matplotlib :

- la fonction *imshow* permet de charger en mémoire une image donnée sous forme d'un tableau numpy.
- la fonction *show* permet l'affichage à l'écran.

4. Écrire trois fonctions qui permettent de retourner un tableau numpy représentant l'image donnée en paramètre en considérant à chaque fois une seule composante de couleur (bleue, vert et rouge).
5. Écrire une fonction qui permet d'afficher quatre images dans un même graphique en associant à chaque image une étiquette avec *xlabel('etiquette')*. Testez votre fonction en affichant l'image d'origine et celles de la composantes blue, verte et rouge.

NB : on donne la fonction *subplot* du sous module pyplot de matplotlib qui permet d'afficher plusieurs figures sur un même graphique. La syntaxe de subplot est la suivante : *subplot(l,c,n)* où l est le nombre de lignes, c le nombre de colonnes et n un compteur.

6. Écrire une fonction python qui permet de retourner une image en niveaux de gris à partir d'une image couleur (donnée sous forme de tableau numpy) sachant que le niveau de gris d'un pixel de l'image retournée est la moyenne des valeurs des composantes RGB de l'image couleur.

Manipulation d'une image en niveaux de gris :

Dans toutes les fonctions qui vont suivre, nous allons travailler avec une image en niveaux de gris. On considérera l'image résultat de la fonction de la question 6.

1. Écrire deux fonctions python qui permettent de construire respectivement l'image symétrique d'une image par rapport à l'axe vertical et celle symétrique par rapport à l'axe horizontal (effet miroir).
2. Écrire une fonction python qui permet de construire le négatif d'une image en niveaux de gris passée en paramètre. Le négatif d'une image I est obtenu en inversant l'échelle de niveaux de gris. Vous pouvez définir une fonction lambda qui retourne l'inverse d'un niveau de gris.
3. Pour améliorer le contraste, on peut transformer le niveaux de gris d'entrée en un nouveau niveau dans le but d'écarter les différences entre les niveaux. Pour cela, on applique une fonction bien choisie. Définir une fonction python qui permet d'appliquer une fonction sur une image en niveaux de gris (Testez avec les deux fonctions suivantes et interprétez) :

— $f(x) = \sqrt{x}$

— $g(x) = \frac{1}{2} + \frac{1}{2} \times \sin(\pi(x - \frac{1}{2}))$

NB : définir f et g comme deux fonctions lambda.