

Exercices sur les images

Exercice 1. Transformation d'une image couleur en niveaux de gris(*)

L'œil est plus sensible à certaines couleurs qu'à d'autres. Le vert (pur), par exemple, paraît plus clair que le bleu (pur). Pour tenir compte de cette sensibilité dans la transformation d'une image couleur en une image en niveaux de gris, on ne prend généralement pas la moyenne arithmétique des intensités de couleurs fondamentales, mais une moyenne pondérée. La formule standard donnant le niveau de gris en fonction des trois composantes est :

$$\text{gris} = \lfloor 0.299 * \text{rouge} + 0.587 * \text{vert} + 0.114 * \text{bleu} \rfloor$$

Écrire une fonction `niveau_gris` ayant pour argument une chaîne de caractères étant le nom du fichier de l'image initiale. Cette fonction doit créer une nouvelle image en niveau de gris à partir d'une matrice ne contenant que le niveau de gris et pas les niveaux R, V, B. Le nom du fichier résultant sera *fichier_gris.png*

Exercice 2. Effet Pop-Art(**)

Écrire une fonction permettant de réaliser un effet Pop-Art sur une image. Pour cela appliquer un filtre de couleur à l'image (ce filtre peut être un filtre rouge, vert, bleu ou de tout autre couleur mélange de ces trois couleurs primaires).

Le résultat attendu est une image composée dont les dimensions sont les doubles de l'image de départ et constituée de quatre images de départ filtrées. En haut à gauche, l'image est obtenue avec un filtre jaune (R,V,B)=(255,255,0); en haut à droite avec un filtre rouge (R,V,B)=(255,0,0); en bas à gauche avec un filtre magenta (R,V,B)=(255,0,255) et en bas à droite avec un filtre vert (R,V,B)=(0,255,0).

L'image sera supposée stockée dans un tableau `numpy.array`

Exercice 3. Réduction d'une image (**)

Les pixels d'une image sont supposées codées dans un tableau `numpy.array`. On supposera également dans un premiers temps que l'on travaille avec des images codées en niveaux de gris.

1. On suppose que l'image est de taille $2n \times 2m$. Écrire une fonction `reduction` qui prend en argument une telle image, et qui renvoie une image de taille $n \times m$ où les pixels de la nouvelle image sont obtenus en moyennant des groupes de 4 pixels, de sorte à ce que les 4 pixels en haut à gauche donnent le pixel en haut à gauche de la nouvelle image; les 4 pixels à droite de ceux-ci donnent le pixel à droite du premier dans la nouvelle image etc..

2. Généraliser à un facteur de réduction de p pour une image de taille $pn \times pm$

3. Généraliser à des images en couleur, codées en RGB.