

Traitement d'images : TP1

Si vous utilisez Octave, les commandes restent les mêmes. Il faut juste mettre la commande `pkg load image` au début de votre code pour charger le package image.

Certaines fonctions de traitement d'images Matlab ne peuvent être exécutées que sur des images dont les valeurs sont des réels entre 0 et 1. Avant d'appliquer une telle fonction, il faut donc vérifier que la matrice représentant l'image contient bien de telles valeurs et, si besoin, modifier le codage.

Certaines fonctions ne peuvent être exécutées que sur des images en niveaux de gris. Avant d'appliquer une telle fonction, il faut donc vérifier que la matrice représente bien une image en niveaux de gris et, au besoin, effectuer une conversion.

Exercice 1 Mise en place

- Lisez le document présentant les principales fonctionnalités de Matlab
- Récupérez les images (blobs.png et peppers.png) sur la plateforme et enregistrez-les dans votre dossier
- Lancez Matlab (sous Linux, ouvrez un terminal et tapez la commande `matlab`) et placez-vous dans votre répertoire
- Créez la matrice suivante : $M1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ et affichez l'image associée.

Exercice 2 Image en noir et blanc

- Chargez l'image blobs.png
- Extrayez de cette image un carré de 10 * 10 pixels (le coin en haut à gauche par exemple).
- Affichez la matrice associée à cette imagerie.

Exercice 3 Image en vraies couleurs

- Créez les trois matrices suivantes :

$$R = \begin{bmatrix} 0.5 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0.5 \end{bmatrix} \quad V = \begin{bmatrix} 0.5 & 0.6 & 0 \\ 0.6 & 0 & 0.6 \\ 0 & 0.6 & 0.5 \end{bmatrix} \quad B = \begin{bmatrix} 0.5 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0.5 \end{bmatrix}$$

- A l'aide de ces 3 matrices, construisez la matrice 3D dans laquelle la première couche sera R, la seconde sera V, et la troisième B.
- Créez et affichez l'image couleur associée à cette nouvelle matrice.
- Enregistrez cette image dans un fichier de nom `im_coull.bmp`.

Exercice 4 Image en couleurs indexées

- Créez les matrices : $X = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 2 \\ 3 & 2 & 1 \end{bmatrix}$ $P = \begin{bmatrix} 0.5 & 0.5 & 0.5 \\ 0 & 0.6 & 1 \\ 1 & 0 & 0 \end{bmatrix}$

P représente une palette de couleurs et X le codage d'une image indexée associée à cette palette.

— Affichez l'image représentée par ces deux matrices.

— Affichez la palette à l'aide de l'éditeur de variables et modifiez les valeurs en
$$\begin{bmatrix} 0.2 & 0.2 & 0.2 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

— Affichez l'image représentée par les deux matrices dans une nouvelle fenêtre

Exercice 5 Ecrivez le programme `ex5.m` qui :

- charge l'image couleur `peppers.png`,
- affiche la taille de l'image,
- affiche le nombre de pixels de l'image,
- extrait les matrices R, V, B, des composantes rouge, verte et bleue de cette image,
- affiche les images représentées par chaque matrice R, V et B (de quelle couleur est affichée chaque image?),
- crée une image en niveaux de gris à partir de l'image originale,
- enregistre cette nouvelle image dans un fichier,
- affiche le nombre de pixels de niveau de gris 150 dans l'image d'intensités,
- ajoute un carré rouge de taille 10*10 dans l'image originale à la position (50,50) (position du coin supérieur gauche du carré),
- enregistre l'image modifiée dans un fichier.

Exercice 6 Histogramme

Ecrivez une fonction matlab `histogramme` qui admet en paramètre d'entrée le nom d'un fichier image en vraies couleurs et qui effectue les traitements suivants :

- Chargement de l'image dont le nom est donné en paramètre d'entrée et affichage de celle-ci,
- Conversion de cette image en image d'intensités et affichage de celle-ci,
- Affichage du nombre de lignes et de colonnes de l'image,
- Calcul de l'histogramme dans une variable tableau H (utilisez la fonction `imhist`),
- Affichage des valeurs minimale et maximale de l'histogramme,
- Affichage graphique de l'histogramme avec la fonction `bar` (utilisez l'aide de matlab).

Testez votre fonction sur l'image `image1.jpg`.