

TP 1 : Traitement d'image

Le but de ce TP est de s'initier en traitement d'image avec l'outil MATLAB : l'affichage d'histogramme et seuillage d'image ainsi que l'échantillonnage et quantification

1. Histogramme et seuillage d'une image

Indication :

L'histogramme d'une image donne la répartition de ses niveaux de gris. Ainsi pour une image qui possède 256 niveaux de gris, l'histogramme représente le niveau de gris en fonction du nombre de pixels à ce niveau de gris dans l'image.

*C'est connu que les niveaux de gris à zéro correspondent au noir et que les niveaux de gris à 1 indiquent le blanc. L'histogramme donne donc une excellente idée de la séparation entre ce qui est clair et ce qui est foncé dans l'image. Dans ce cas, une utilisation de ce fait est le **seuillage** d'une image, ce terme désigne la définition d'un seuil au-dessus ou en-dessous duquel nous allons garder certaines valeurs de niveaux de gris*

1. Lire l'image «circuit.tif»
2. Récupérer son histogramme à l'aide de la fonction **imhist**
» `histo = imhist(img,256);`
3. Afficher l'histogramme obtenu à l'aide de la commande suivante :
» `figure;plot(histo);`
4. Pour égaliser et recadrer la dynamique de l'image, l'exploitation de l'histogramme nous permet d'effectuer plusieurs opérations en utilisant les fonctions suivantes :

imadjust: recadrage de la dynamique de l'image,

histeq : égalisation d'histogrammes,

adapthisteq: égalisation adaptative d'histogrammes,

5. Etape de seuillage de l'histogramme : effectuer le seuillage en utilisant toujours l'image « circuit.tif » (ou une autre image de votre choix)
6. Afficher une autre fois l'image résultante
» `figure;imshow(img);`
7. Afficher les images résultantes (ci-dessus) en subdivisant la figure à l'aide de la fonction **subplot**(m,n,p) avec m le nombre de lignes, n le nombre de colonnes et p représente le p^{ème} axe

8. En procédant au seuillage de l'image, nous allons utiliser la fonction **im2bw**(img) qui permet la binarisation de l'image

» result=im2bw(img);

» subplot(m,n,p);imshow(result); % le m, n et p à fixer
9. Dans le même cadre de seuillage, créer une fonction qui permet d'obtenir une nouvelle image en fixant un seuil de votre choix

2. L'échantillonnage et quantification

1. Effet de la quantification :

Écrire un programme qui fait la quantification uniforme (sur l'intervalle [0, 255]) d'une image sur plusieurs niveaux de gris. Voir les résultats et commenter.

2. Effet de l'échantillonnage:

Faire un sous-échantillonnage de l'image mire radiale (« vertigo»). Commenter votre réponse en fonction du pas du sous-échantillonnage : $n = 256/m$ avec m prend les valeurs : 8, 16, 64, 256