

Traitement d'images

TP N°2 : Opérations ponctuelles et transformations géométriques

Manipulation 1

- 1- Lire et afficher l'image 'flower.bmp' .
- 2- Convertir l'image en niveaux de gris
- 3- Calculer et afficher l'histogramme et l'histogramme normalisé. Commenter
- 4- Calculer et afficher l'histogramme cumulé. Commenter
- 5- Réaliser une égalisation d'histogramme, afficher l'histogramme égalisé et commenter
- 6- Comparer avec le résultat obtenu par la fonction [cv.equalizeHist](#)
- 7- Egaliser les 3 composantes couleurs indépendamment, puis visualiser l'image résultante et calculer son entropie.
- 8- Egaliser les 3 composantes couleurs par rapport à l'histogramme cumulé de la question 4 et visualiser l'image correspondante. Calculer son entropie.
- 9- Comparer les entropies et les histogrammes. Commenter.
- 10- Lire et visualiser l'image 'pentagon.tif', égaliser son histogramme. Commenter.
- 11- Réaliser une amélioration de contraste de l'image 'cameraman.jpg' à l'aide de l'histogramme matching de l'image 'barbara.png' à l'aide de la fonction [match_histograms](#).

NB : il faut rajouter les commandes suivantes :

```
from skimage import exposure  
  
from skimage.exposure import match_histograms
```

Manipulation 2

Créer 2 images binaires A et B de taille 256x256 pixels telles que ;

- l'image A possède un rectangle noir aux coordonnées $90 \leq x \leq 120$ et $90 \leq y \leq 180$
 - l'image B possède un carré noir aux coordonnées $110 \leq x \leq 150$ et $110 \leq y \leq 150$
- 1- Réaliser les opérations logiques NOT, AND, OR et XOR à l'aide des instructions ; [cv.bitwise_not](#), [cv.bitwise_and](#), [cv.bitwise_or](#), [cv.bitwise_xor](#). Analyser et interpréter.
 - 2- Générer les images suivantes : $A_i = \text{not } A$ et $B_i = \text{Not } B$. Réaliser les opérations logiques $\text{AND}(A_i, B_i)$, $\text{OR}(A_i, B_i)$ et $\text{XOR}(A_i, B_i)$. Comparer avec les résultats obtenus précédemment.
 - 3- Comment doivent être les images à l'entrée de ces opérateurs logiques afin d'avoir un résultat cohérent lors de leur utilisation ?

Manipulation 3

- 1- Lire et afficher l'image 'BoatsColor.bmp' .
- 2- pour chacune des transformations suivantes, comparer les différentes méthodes d'interpolation (plus proche voisin, bilineaire et bicubic)
 - a. Translater l'image de 35 pixels vers le bas et 10 pixels vers la droite

Traitement d'images

- b. Redimensionner la taille de l'image en divisant sa largeur par un facteur de 2 et sa hauteur par un facteur de 1,5
- c. Pivoter l'image d'un angle de 60°

Pour les instructions, consulter :

https://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_geometric_transformations/py_geometric_transformations.html