

M1 – Informatique

Introduction au traitement du signal, aux signaux sonores et aux images

TP5 : À la découverte des images RGB (Red Green Blue)

I. Rehaussement par Laplacien

1. Charger l'image `peppers.png` et afficher la.



2. Créer le masque correspondant à l'opérateur Laplacien avec un centre négatif. Filtrer l'image en utilisant la fonction de la déconvolution `scipy.signal.convolve2d` sur chaque composante de l'image. Afficher le résultat.
3. À partir de l'image originale et l'image obtenue en 2., comment pouvons-nous obtenir une image rehaussée ? Appliquer cette méthode. Afficher le résultat et commenter.

II. Manipulation des composantes RGB

1. Affichage des composantes de l'image de la voiture : `2cv.jpg`



- a. Construire l'image en intensité de la composante bleue uniquement. Pour cela :
- créer une image `img_bleue` de la même taille que `img` et contenant que des 0,
 - copier les valeurs de la composante bleue de `img` dans `img_bleue`,
 - afficher l'image : que remarquez-vous ?



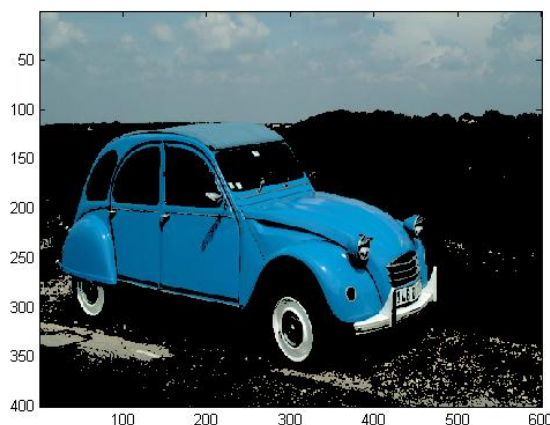
- b. Construire de même l'image de la composante verte et l'image de la composante rouge.
- c. Commenter l'évolution de l'intensité des pixels correspondant à la carrosserie de la voiture.

2. Filtrage en fonction des composantes

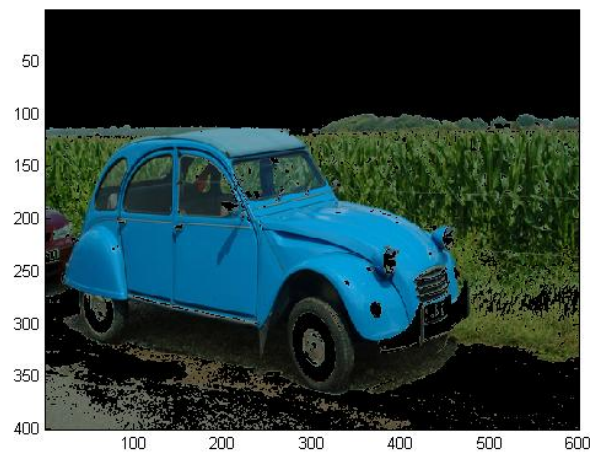
- a. Construire l'image filtrée où ne sont retenus que les pixels dont l'intensité en bleu est supérieure à un seuil donné :

```
dimensions=np.shape(img)
seuil = 200
masque = (img[:, :, 2] > seuil)
imgf = np.zeros(dimensions, dtype=np.uint8)
imgf[masque, 0] = img[masque, 0]
imgf[masque, 1] = img[masque, 1]
imgf[masque, 2] = img[masque, 2]
```

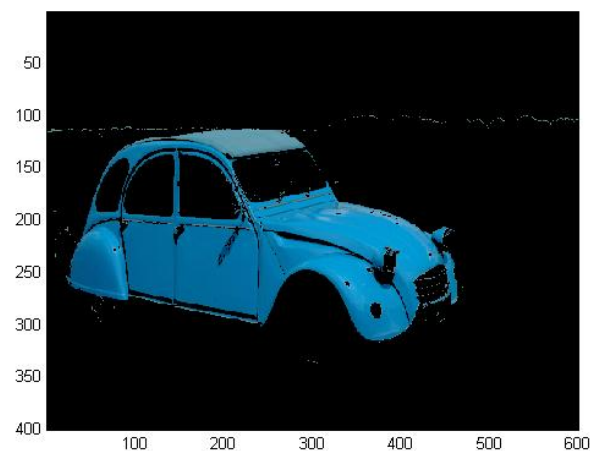
Donner une valeur de la variable `seuil` sur la composante bleue qui permet d'obtenir la carrosserie complète de la voiture tout en éliminant un maximum des autres pixels de l'image. Identifier les principales zones ayant été retenues après filtrage et expliquer la raison de leur présence à l'image.



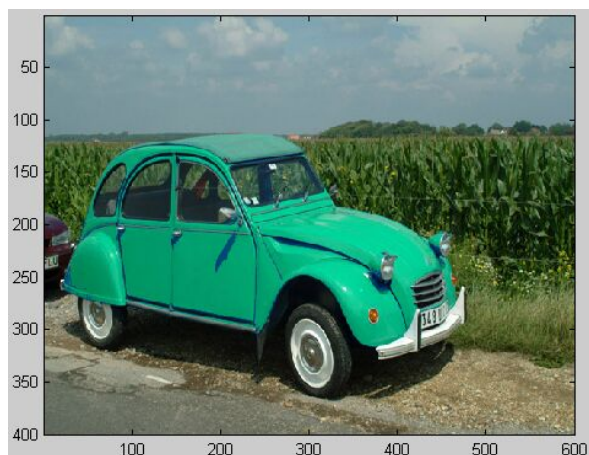
- b. Sur le même principe, proposer un seuillage sur la composante rouge pour essayer de conserver l'intégralité de la carrosserie de la voiture tout en éliminant un maximum de pixels. Produire le même type d'analyse que pour le résultat précédent.



- c. Proposer une combinaison des deux seuils précédents (sur la composante bleue et sur la composante rouge) pour cerner au mieux la carrosserie. Les seuils permettent ici de déterminer une région de l'image. Nous avons « masqué » les pixels n'appartenant pas à la voiture. D'une manière générale, en traitement d'image, un « masque » désigne une image qui identifie un sous-ensemble des pixels d'une image de départ.



- d. À l'aide de ces informations, intervertissez les valeurs des composantes verte et bleue pour transformer la couleur de la voiture.



III. Conversion d'image

1. Image en niveaux de gris et image binaire

- a. Convertir l'image de départ en niveaux de gris et afficher le résultat obtenu, en utilisant la palette graphique composée de 256 niveaux de gris.
- b. Binariser l'image en niveaux de gris avec le seuil par défaut (128). Afficher les résultats obtenus.

2. Histogramme

- a. Calculer l'histogramme de l'image en niveaux de gris. Afficher le résultat.
- b. À l'aide de l'histogramme, est-il possible de binariser l'image en niveaux de gris de manière plus intelligente (afin que l'image binaire soit plus lisible) ?