

Traitement d'image : Convolution

1 Première étape : implémentation java

Le but de ce TP est d'implémenter la convolution. Il faut donc appliquer un noyau (typiquement carré de largeur impaire) au voisinage de chaque pixel de l'image pour convoluer les valeurs image.

Le travail minimal demandé consiste à :

- fournir un calcul de convolution pour une taille de fenêtre fixée (par exemple 3×3)
- implémenter plusieurs opérateurs basés sur une convolution. Ces opérateurs peuvent nécessiter d'appliquer un seul noyau (par exemple pour faire un filtre Moyenneur) ou plusieurs (par exemple pour faire un Extracteur de contour).

2 Renderscript

Implémentez les opérateurs basés sur une convolution en renderscript et comparez les performances avec la version java.

Vous pouvez ensuite comparer votre version à une version utilisant les scripts intrinsic (<https://developer.android.com/reference/android/renderscript/ScriptIntrinsic.html>) qui permettent par exemple de faire une convolution 3×3 , 5×5 ou un filtre gaussien sur une image.

3 Généricité

La convolution pose plusieurs questions au niveau de l'architecture logicielle, afin d'éviter autant que possible de dupliquer du code. Même s'il n'est pas obligatoire d'y réfléchir pour le rendu de TP, ce sera nécessaire pour le projet.

Plusieurs améliorations peuvent également être envisagées :

- Permettre d'appliquer des filtres carrés de taille quelconque.
- Permettre d'appliquer des filtres de taille quelconque (utile par exemple pour les filtres séparables).
- Séparer l'itérateur sur le voisinage d'un pixel du calcul qui y est associé (ce qui permet de coder d'autres opérateurs définis sur un voisinage, tels que les filtres de rang).