

TP : Élimination du bruit

Jonathan Fabrizio
<http://jo.fabrizio.free.fr/>

1 Objectif

L'objectif de ce T.P. est de survoler des techniques d'élimination du bruit. Pour y parvenir vous pouvez reprendre les classes images `tifo::gray8_image` et `tifo::rgb24_image` du TP1 ainsi que les fonctions pour lire ou écrire des images : `tifo::load_image` et `tifo::save_image`.

2 Algorithmes

2.1 Flou Gaussien

Écrivez une fonction qui réalise un flou Gaussien sur une image couleur. Vous pourrez traiter indépendamment chaque canal et reprendre vos fonctions du TP1 pour séparer/fusionner les différents canaux ainsi que la fonction de convolution.

2.2 NLMeans

Écrivez une fonction qui applique l'algorithme NLMeans sur une image couleur. Vous pouvez préalablement lire l'article "Non-Local Means Denoising" écrit par Antoni Buades, Bartomeu Coll et Jean-Michel Morel.

2.3 Flou Gaussien Sélectif

Écrivez une fonction qui réalise un flou Gaussien sur une image couleur. Comparativement au flou Gaussien classique, la contribution d'un pixel dépend non seulement de la distance au pixel central mais aussi de la distance entre la couleur du pixel central et le pixel considéré. On restera dans l'espace RGB (même si les distances entre couleurs dans l'espace RGB ne sont pas proportionnelles à la perception).

3 Analyse

3.1 Fixer les paramètres

Ces algorithmes dépendent de paramètres. Réfléchissez pour chacun d'entre eux, en fonction de quoi et comment fixer les paramètres des algorithmes. Essayer d'estimer grossièrement l'écart type du bruit (en vous aidant de GIMP par exemple) puis choisissez les paramètres de vos algorithmes.

3.2 Comparaison des résultats

Comparer les résultats des différentes méthodes.

4 Pour aller plus loin

4.1 Comparaisons d'images

Pour pouvoir analyser correctement les résultats, il est nécessaire de pouvoir soustraire deux images. Écrivez rapidement une fonction qui réalise cette soustraction. Vérifiez si possible que cette fonction a bien été vectorisée par le compilateur.

Si vous le souhaitez, essayez d'écrire vous même une version vectorisée. Pour cela regardez quels sont les jeux d'instructions disponibles sur votre machine (dans `/proc/cpuinfo`). Vous avez besoin des fonctions `load` et `store` et d'une fonction pour réaliser la soustraction (comme `__m256i _mm256_subs_epu8 (__m256i a, __m256i b)`);



sub : soustraction, s : saturée, ep u :unsigned). Vous pouvez trouver la liste des instructions à l'adresse suivante : <https://software.intel.com/sites/landingpage/IntrinsicsGuide/>.

4.2 Combinaison du renforcement de la netteté et élimination du bruit.

Reprenez votre TP précédent. Essayez d'améliorer la netteté puis d'enlever le bruit. Essayez dans l'autre sens. Quel est la séquence qui donne le meilleur résultat ? Pourquoi ?

Il reste toutefois des artefacts (par exemple sur les branches dans l'image 20140326_1244555), à quoi sont-ils dus ?

4.3 Correction gamma

Dans l'ensemble de nos programmes, nous n'avons pas tenu compte de la correction gamma. Évidemment cela introduit des fautes. Pour faire bien, il serait nécessaire de tenir compte du gamma (en supposant que les images soient encodée en utilisant sRGB par exemple).