Guénon Marie et Favreau Jean-Dominique

VIM / MASTER SSTIM

INTRODUCTION AUX PROBLEMES INVERSES EN TRAITEMENT D’IMAGE

dédruitage

Table des matières

[Méthode 2](#_Toc372643966)

[Paramètres 2](#_Toc372643967)

# Méthode

Pour chaque pixel de l’image , on prend un patch autour de lui (en excluant la bande de bord pour éviter d’avoir à gérer les problèmes).  
Pour une grande zone autour du pixel (fenêtre de recherche), on extrait tous les patches.

On calcule la distance entre et tous les autres patches et on garde les *k* patches les plus proches puis on fait la moyenne pondérée et on normalise et on obtient le patch :

Où est le patch i et

Et finalement, on assigne au pixel la valeur au centre du correspondant.

# Paramètres

Pour pouvoir appliquer notre algorithme, nous avons dû fixer un certain nombre des paramètres du modèle :

* Taille du patch :

A partir de cette taille, nous avons créé un patch de taille nxn.

Nous avons choisi de fixer la taille du patch à .

* Taille de la fenêtre de recherche :

Nous avons fixé la taille de la fenêtre de recherche à et on obtient alors une fenêtre de wxw.

* Paramètre de poids :

Nous avons testé plusieurs valeurs de et obtenus les résultats suivants :

Pour

Pour

Pour

Analyse (attention, lié au nombre de pixels dans l’image)

* Nombre de patches les plus similaires :

Nous avons testé plusieurs valeurs de et obtenus les résultats suivants : (avec )

Pour

Pour

Pour

analyse entre bruit et du flouttage :

Si trop petit reste trop de bruit, si trop grand on perd des détails et on introduit du flou)

# Commentaires

Attention ! Très lent (normal c’est Matlab => ce serait mieux de codé en parallèle GPU)

On voit bien qu’il y a un bord non traité en périphérie de l’image, ce qui est normal car nous n’appliquons pas notre algorithme sur une bande de taille n au bord de notre image initiale.

(vendredi à midi)