

TP1 texture

Florent Diet

Novembre 2022

1 P1 : Image Warping

1.1 rendu

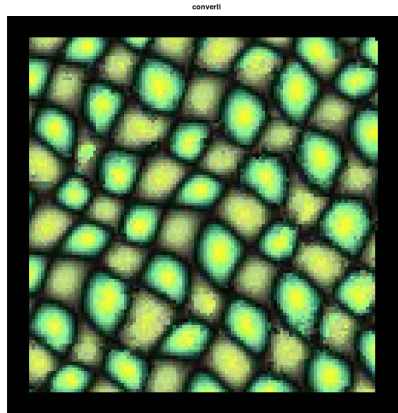


Figure 1: patch 13x13 avec un epsilon de 0.1

Cette image est le résultat de mon travail et correspond à l'application d'un patch de 13x13 qui vient d'une image de texture en 64x64 dans une image 130x130 et un epsilon de 0.1. J'ai placé mon patch au centre et arrivé aux bords de la texture originel (64x64) la distance euclidienne minimum entre les patches A et le patch B ne me donne plus 0 vu que je ne peux pas retrouver le meme pattern que sur la texture originel.

C'est là on peut commencer à voir des éléments un peu perturbants même si l'image créée me semble correcte. Ici j'ai généré 13 000 points à partir du patch et cela m'a pris 16 minutes.

1.2 Changer la taille du patch

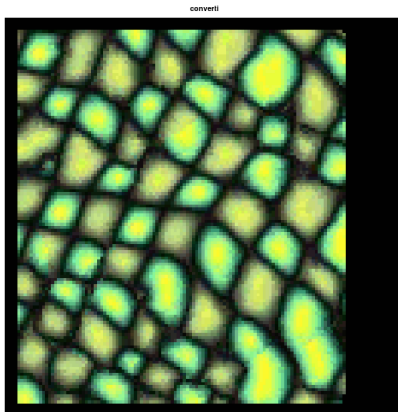


Figure 2: patch de 7x7 avec un epsilon de 0.1

En changeant la taille de mon patch de 13x13 à 7x7 avec comme epsilon 0.1 j'obtiens cette image et cela a pris 19 min pour générer 13 000 points. Le resultat me semble plus correct que quand j'avais un patch plus grand cela doit etre du au fait que le calcul de la distance ne distingue pas entre les pixel les plus proches du pixel au centre du patch donc il prend potentiellement des pixel qui ne correspondent pas forcément au cas actuel.

On peut donc s'imaginer que quand le patch est trop petit on a un manque de donnée donc on peut arriver a un moment ou l'algo prend toujours la meme couleur. Et si le patch est trop grand il ne sera pas précis donc on va avoir du bruit.

1.3 Changer epsilon

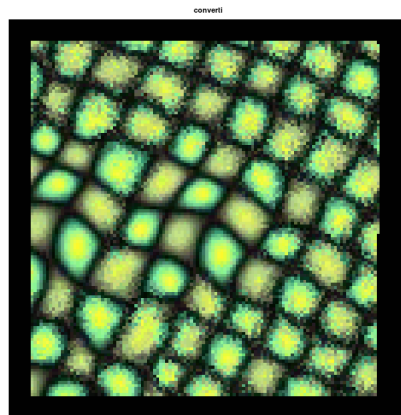


Figure 3: patch de 13x13 avec un epsilon de 0.5

Pour un patch de 13x13 et un epsilon de 0.5 j'obtiens une image qui est nettement moins précise que mon image genere avec un patch 13x13 et un epsilon de 0.1.

Cela me semble logique car il prend en compte plus de patch de ressemblance (entre 100% et 150% par rapport au meilleur patch de ressemblance donc 50% de marge contre 10% pour la premiere image genere).Donc la perte de précision est justifié.

1.4 implementation plus rapide

Comme Nous avons vu l'image généré a partir d'un patch 13x13 a mis 3 min de plus a se faire que l'image généré à partir du patch 7x7. Nous savons que quand le patch est pls grand le calcul de distance est plus conséquent puisqu'il y a plus de valeur mais il passe moins de fois dans les boucles. C'est l'inverse pour les patch plus petits.

A partir de cette observation nous pouvons en déduire que octave met plus de temps à passer dans les boucles que à calculer les distances. Un bon point d'amélioration de mon code pour accélérer le rendu serait donc de remedier a ces boucles.

En ayant essayé de limiter le plus possible le nombre de boucle je me retrouve tout de meme avec 3 boucles. L'une me semble essentiel c'est celle qui va itéré sur mes points de mon image ou j'ai collé un patch et ou je veux generer une nouvelle texture. Les 2 autres me permettent de parcourir l'image de texture originel afin de trouver le patch ressemblant le plus au patch de mon pixel actuel. Il est peut etre possible de réduire ces 2 boucles par un raccourci que peut proposer octave cependant je n'ai pas la connaissance nécessaire pour pouvoir le faire actuellement.