



Compression basée super pixels

Groupe 4.3 - Projet Image

Auteurs :

AMSALHEM Oren - 22206713

COMBOT Evan - 22315218

Enseignants :

PUECH William

REINDERS Erwan

Algorithme SLIC terminé.

Nous avons terminé l'algorithme SLIC qui est un algorithme basé sur un gradient. Pour ce faire, nous avons suivis ce pseudo-code. Comparé à notre algorithme SLIC précédent, celui-ci est beaucoup plus rapide et produit de meilleurs résultats.

Algorithm 1 SLIC superpixel segmentation

```
/* Initialization */
Initialize cluster centers  $C_k = [l_k, a_k, b_k, x_k, y_k]^T$  by
sampling pixels at regular grid steps  $S$ .
Move cluster centers to the lowest gradient position in a
 $3 \times 3$  neighborhood.
Set label  $l(i) = -1$  for each pixel  $i$ .
Set distance  $d(i) = \infty$  for each pixel  $i$ .

repeat
  /* Assignment */
  for each cluster center  $C_k$  do
    for each pixel  $i$  in a  $2S \times 2S$  region around  $C_k$  do
      Compute the distance  $D$  between  $C_k$  and  $i$ .
      if  $D < d(i)$  then
        set  $d(i) = D$ 
        set  $l(i) = k$ 
      end if
    end for
  end for
  /* Update */
  Compute new cluster centers.
  Compute residual error  $E$ .
until  $E \leq \text{threshold}$ 
```

FIGURE 1 – <https://infoscience.epfl.ch/record/177415>

Voici les étapes de l'algorithme :

- Initialisation de K (nombre de superpixels) centres de clusters séparés d'une distance S chacun.
- Perturbation des centres des clusters à l'aide d'un gradient. Dans notre cas, on utilise le gradient de Sobel car ce gradient nous permet de faire en sorte que les centres des clusters ne se retrouvent pas sur un contour, sur quelque chose de bruité mais sur une zone lisse. Ce gradient est meilleur que le gradient (norme L2). On cherche la valeur minimale du gradient parmi un n voisinage de chaque centre des clusters.
- Mise à jour des centres des clusters et attribution des couleurs (moyenne) pour un nombre d'itérations tant qu'il n'y a pas convergence.

Des soucis se posent sur les zones très texturés (à voir ajout TASP prochainement).

Nous avons la possibilité d'afficher les contours des superpixels déterminés par un gradient ainsi que les centres des clusters finaux. Nous calculons aussi le taux de compression après avoir effectué une compression par palette.

Voici les données d'entrées pour les exemples ci-dessous :

Supers Pixels : 1500

Compacité : 75

Voisinage : 3

Nombre d'itérations : 3

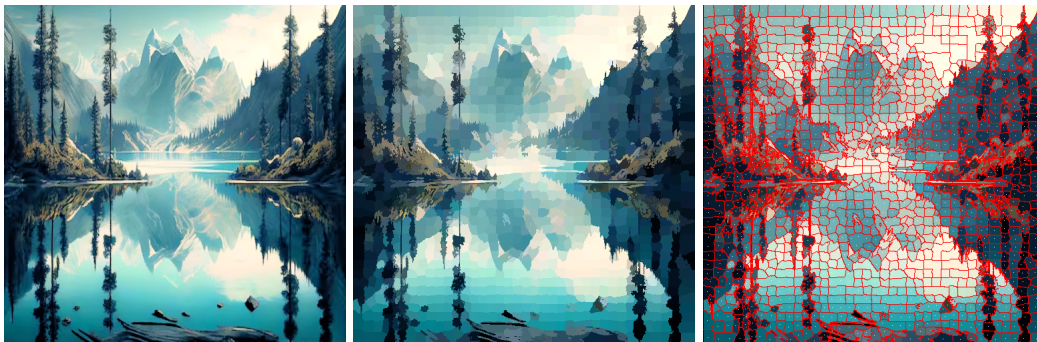


FIGURE 2 – nbr de couleurs : 1444, taux de compression : 55.53, PSNR : 23.73



FIGURE 3 – nbr de couleurs : 1681, taux de compression : 27.66, PSNR : 23.19

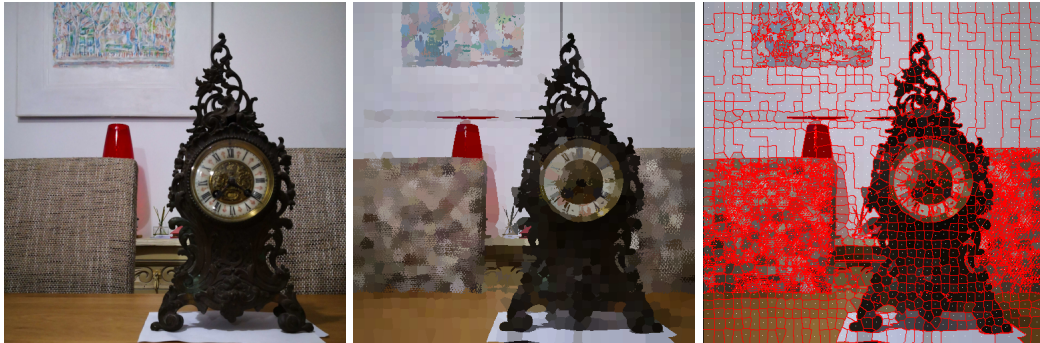


FIGURE 4 – nbr de couleurs : 1444, taux de compression : 55.7, PSNR : 26.23

Nous avons remarqué que plus le nombre de superpixels augmente, plus le taux de compression est faible et plus le PSNR est élevé.

On peut remarquer que certains superpixels connexes n'ont pas de contours car les couleurs moyennes de ces superpixels sont les mêmes. Nous prévoyons d'ajouter une version modifiée de l'algorithme SLIC que nous avons nommé : SLIC&Merge et qui permettra de fusionner ces superpixels en un seul. Le centre de ce nouveau superpixel sera alors recalculé.

2 - Amélioration de l'interface

Parallèlement, nous avons continué à améliorer l'interface. Nous l'avons grandement simplifié :

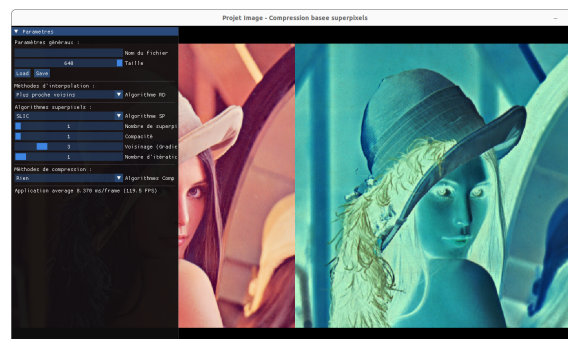


FIGURE 5 – Exemple de l'interface

Malheureusement, nous n'avons pas intégré l'algorithme SLIC car nous rencontrons de gros problèmes de performances. Nous nous sommes donc lancé dans de l'optimisation et plutôt que d'utiliser la librairie d'image des TPs, nous réfléchissons à utiliser d'autres librairies comme

OpenMP ou OpenCV (en adaptant nos fonctions de SLIC) ou autres pour paralléliser au maximum le code et avoir une interface en temps réel efficace.