



HMIN322 - Compression Multimédia



Compression basée sur les Super Pixels avec SLIC

Présenté par :

Thibault Odorico

2019-2020

Sommaire

Introduction

1. Mise en contexte et définitions

1. Différentes méthodes de compression
2. Comparatif des algorithmes
3. SLIC

2. Implémentation

1. Etape par etape
2. Résultats

Conclusion

Introduction

Super Pixel

- Empaquetage de pixels cohérents
 - Position
 - Couleur
- Enlève les redondances
- Réduit les éléments à traiter
- Pratique pour la segmentation
- Pratique pour la compression



Méthodes de compression

Méthode par graphe

- Pixel : Noeud du graphe
- Arêtes : Pondération selon la similarité des pixels
- Arêtes : Extraction en minimisant une fonction de coût sur le graphe

Méthode par gradient

- Initialisation uniforme des super pixels dans l'image (Comme une grille)
- Amélioration des super pixels de manière itérative (selon un critère de similarité)

Remarque : SLIC utilise une méthode par gradient

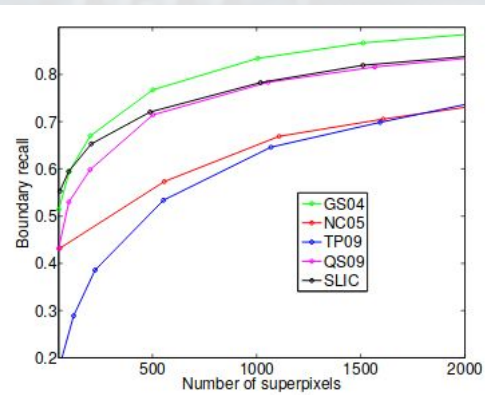
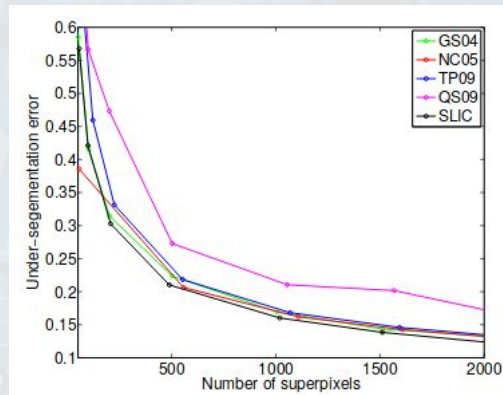
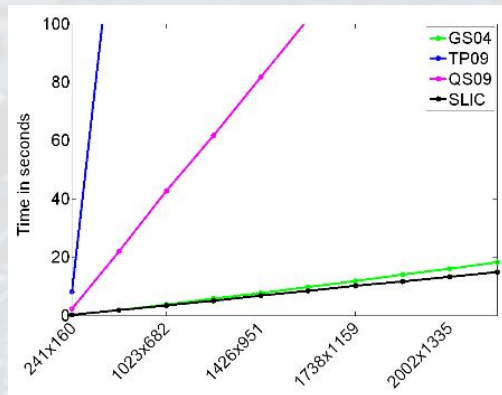
Comparatif des algorithmes

Properties	Graph-based			Gradient-ascent-based				
	GS04	NC05	SL08	WS91	MS02	TP09	QS09	SLIC
Supapixel no. ctrl.	No	Yes	Yes	No	No	Yes	No	Yes
Compactness ctrl.	No	Yes	Yes	No	No	Yes	No	Yes
Complexity $O(.)$	$N \log N$	$N^{3/2}$	$N^2 \log N$	$N \log N$	N^2	N	dN^2	N

Algorithme SLIC

- Complexité linéaire
- Applicable aux algorithmes de vision par ordinateur
- Fourni-t-il une bonne segmentation des pixels ?

Comparatif des algorithmes



Algorithme SLIC

- Le taux d'erreurs sur la bordure des objets est le plus bas

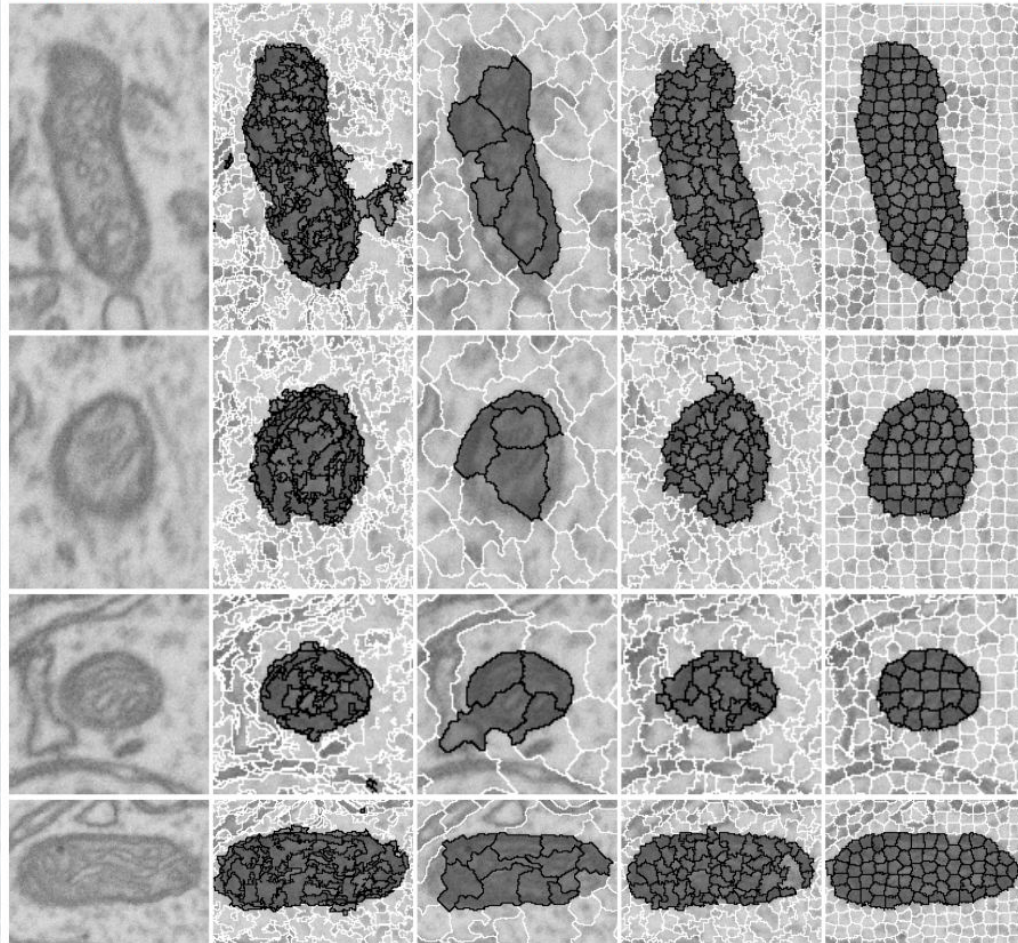
EM Image

GS04

TP09

QS09

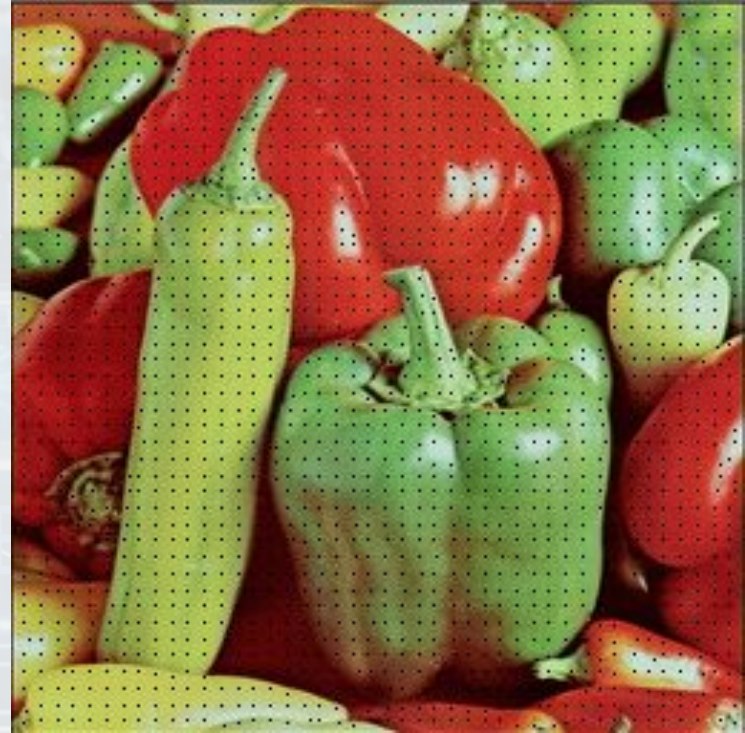
SLIC



Etape par etape : Initialisation uniforme des Super Pixels



Etape par etape : Éloignement des bords



Critère de similarité

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}$$

$$D_s = d_{lab} + \frac{m}{S} d_{xy} ,$$

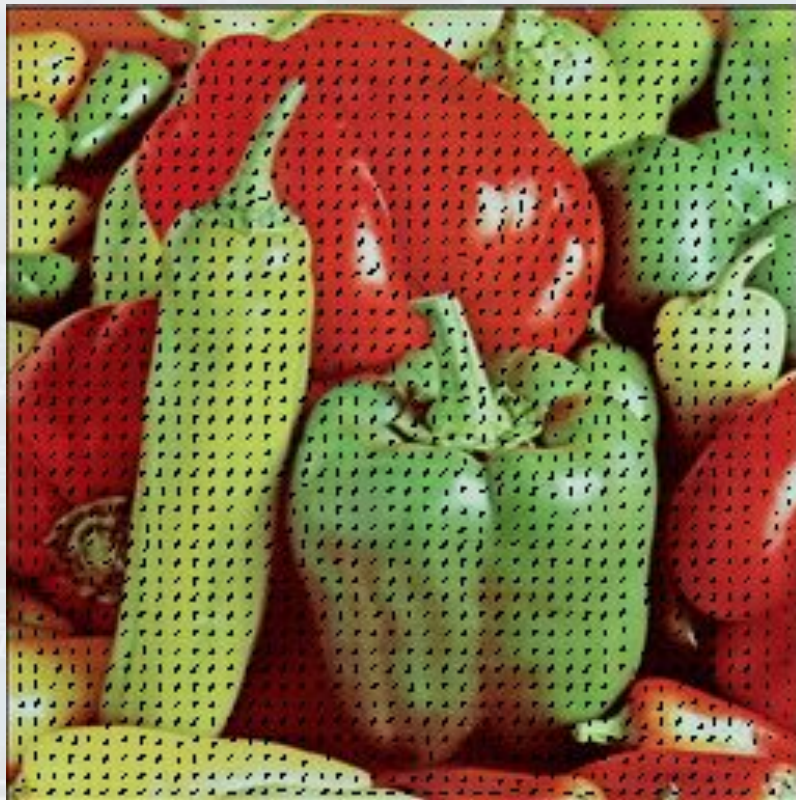
- Distance de position du pixel dans une zone 2S x 2S
- Distance de couleur convertie en LAB dans cette même zone

Critère de similarité

Algorithm 1 Efficient superpixel segmentation

- 1: Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S .
 - 2: Perturb cluster centers in an $n \times n$ neighborhood, to the lowest gradient position.
 - 3: **repeat**
 - 4: **for** each cluster center C_k **do**
 - 5: Assign the best matching pixels from a $2S \times 2S$ square neighborhood around the cluster center according to the distance measure (Eq. 1).
 - 6: **end for**
 - 7: Compute new cluster centers and residual error E {L1 distance between previous centers and recomputed centers}
 - 8: **until** $E \leq \text{threshold}$
 - 9: Enforce connectivity.
-

Etape par etape : Trajectoire des Super Pixels (10 iterations)



Résultats



Conclusion

- SLIC : Une des méthodes la plus efficace pour la vision par ordinateur
- SNIC : Une méthodes encore plus efficace par les mêmes créateurs
- Quelques problèmes de segmentation quand les images contiennent de l'herbe ou des poils
- Implementation : Problème à régler au moment d'assigner un pixel à son super pixel

Bibliographie

- http://openaccess.thecvf.com/content_cvpr_2017/papers/Achanta_Superpixels_and_Polygons_CVPR_2017_paper.pdf
- http://www.kev-smith.com/papers/SLIC_Superpixels.pdf
- <https://makina-corpus.com/blog/metier/2017/localisation-dun-objet-par-classification-de-superpixels>
- <http://vision.gel.ulaval.ca/~jflalonde/cours/4105/h17/tps/results/projet/111063028/index.html>
- <https://medium.com/@darshita1405/superpixels-and-slic-6b2d8a6e4f08>
- https://en.wikipedia.org/wiki/CIELAB_color_space#Forward_transformation



Merci pour votre attention !

