

Rapport du TME 10: Segmentation d'images

Antunes Costa Gonçalves Daniel
Wolfrom Matthieu

1 Exercice I

1.1 Question 1

Pour écrire cette fonction permettant de trouver la plus petite valeur de n telle que 2^n soit plus grand que la largeur et la longueur, il suffit de prendre la plus grande des deux dimensions, `dimMax`, et calculer $n = \text{ceil}(\log_2(\text{dimMax}))$.

On obtient ainsi n et il suffit de créer une image carrée de taille $2^n * 2^n$ et d'y recopier l'image d'origine.

1.2 Question 2

En utilisant $n = \text{size}(\text{find}(S > 0), 1)$; on obtient simplement le nombre de régions dans un découpage Quadtree S , vu que S est une matrice creuse.

1.3 Question 3

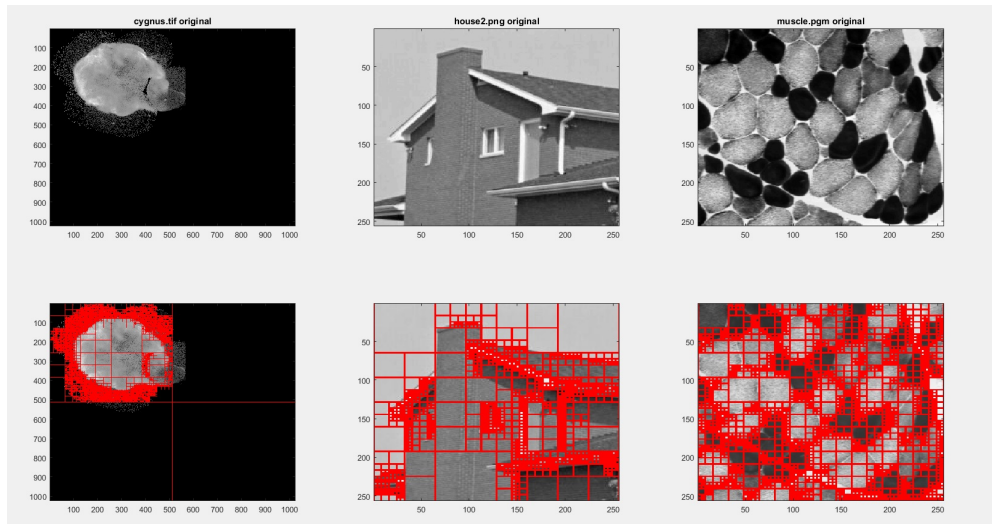
Pour écrire la fonction $Q = \text{quaddraw}(I, S)$, on stocke tout d'abord la taille de I dans les variables n et m . On initialise ensuite Q avec I . (Q sera notre image résultat)

On initialise ensuite la variable $kmax$ qui nous indique la taille du plus grand bloc de la matrice creuse S . On boucle ensuite sur toutes la valeurs k possibles pour la taille d'un bloc.

Pour chaque k possible on extrait les informations sur ces blocs grâce à $[vals, i, j] = \text{qtgetblk}(I, S, k)$; On parcourt ensuite tous les blocs extraits et on dessine un carré rouge sur l'image résultante grâce à la variable k qui nous donne la taille des carrés à dessiner.

1.4 Question 4

Pour créer le script demandé on utilise les fonctions créées précédemment et on obtient le résultat suivant:



2 Exercice II

2.1 Question 1

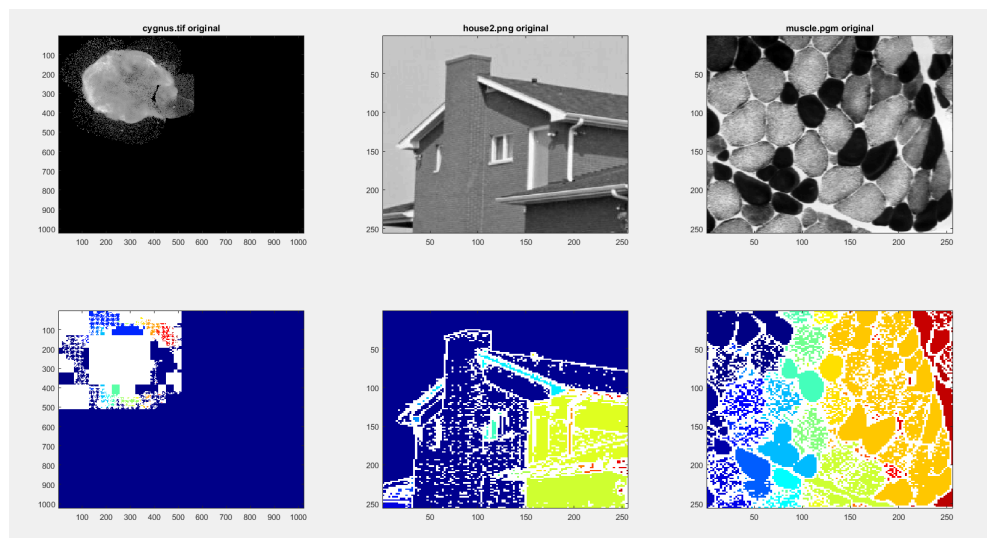
On réalise la fonction de fusion selon l'algorithme vu en TD, en utilisant la fonction matlab `bwlabel` pour attribuer un indice à chaque composante connexe de l'image obtenue par la fusion.

Ensuite pour chaque image de test, nous avons fait varier les 3 paramètres suivants:

- Le seuil (`thresh`) et la dimension minimale des blocs (`mindim`) pour la décomposition en quadtree
- Le seuil (`thresh2`) pour la fusion des blocs

Nous avons trouvé les valeurs suivantes pour les images `cygnus.tif`, `house2.png` et `muscle.pgm` :

```
thresh = [30 3 5];  
mindim = [2 2 2];  
thresh2 = [17 7 8.5];
```



2.2 Question 3

Nous avons réalisé ensuite la seconde fonction de fusion, cette fois globale, et nous avons cherché à obtenir des tests concluants à l'aide de cette fonction également.

Nous avons trouvé les valeurs suivantes pour les images `cygnus.tif`, `house2.png` et `muscle.pgm` :

```
thresh = [5 3 75];  
mindim = [2 4 2];  
thresh2 = [10 35 30];
```

Les résultats sont plus intéressants pour la première image où la fusion des

blocs est très précise, en revanche sur la seconde image nous n'avons pas réussi à obtenir un résultat plus convaincant à l'aide de la fusion globale. Pour la troisième image, aucune des deux fonctions ne donne un résultat plus lisible que l'autre, nous n'avons pas trouvé de paramétrage correct pour cette image.

