
Ψηφιακή Επεξεργασία Εικόνας - Εργασία 2

Ευθύμιος Γρηγοράκης AEM: 9694

1 Image2Graph

Η πρώτη συνάρτηση που κληθήκαμε να υλοποιήσουμε, δέχεται ως είσοδο μια $N \times M$ εικόνα και έχει ως έξοδο έναν affinity matrix. Αφού ανακατασκευάσουμε την εικόνα, για να είναι πιο εύκολη η πρόσβαση στα pixel γεμίζουμε το πάνω μέρος του πίνακα και βάσει αυτού δημιουργούμε τον υπόλοιπο πίνακα, αφού είναι συμμετρικός.

2 myGraphSpectralClustering

Σε αυτή την συνάρτησ ακολουθώντας τα βήματα το αλγορίθμου που μας δώθηκε στην εργασία, βρίσκουμε τις k μικρότερες ιδιοτιμές και ιδιοδιανύσματα με την συνάρτηση `eigs(L,D,k,'smallestabs')` και με την βοήθεια της συνάρτησης `kmeans(V,k)` βρίσκουμε σε ποιά ζλυστερ ανήκει κάθε κορυφή.

2.1 demo1

Στο demo1 αφού φορτώσουμε τον affinity matrix d1a βρίσκουμε με την συνάρτηση που υλοποιήσαμε παραπάνω τα clusters. Παρακάτω παρουσιάζονται τα αποτελέσματα των πειραμάτων.



```
Command Window
For k = 2 cluster indexies are:

clusterIdx =

     2
     2
     2
     2
     1
     1
     1
     1
     2
     2
     2
     2
```

```
Command Window
For k = 3 cluster indexies are:

clusterIdx =

     1
     1
     1
     1
     2
     2
     2
     2
     3
     3
     3
     3
```

```
Command Window
For k = 4 cluster indexies are:

clusterIdx =

     1
     1
     1
     1
     2
     3
     3
     3
     3
     4
     4
     4
     4
```

Ένα σημαντικό συμπέρασμα το οποίο θα χρησιμοποιηθεί και παρακάτω στην εργασία είναι ότι όσο αυξάνουμε τον αριθμό των clusters οι κορυφές δεν μπορούν να "μεταπηδήσουν" σε ένα cluster που δεν άνηκαν πριν, αλλά μόνο να "σπάσουν" σε κανούργια cluster.

2.2 demo2

Στο demo2 κάνουμε την ίδια δουλεία με το demo1, με την μόνη διαφορά ότι δεν έχουμε έτοιμο τον affinity matrix, αλλά τον δημιουργούμε εμείς από τις εικόνες d2a, d2b και την βοήθεια της συνάρτησης Image2Graph που υλοποιήσαμε.

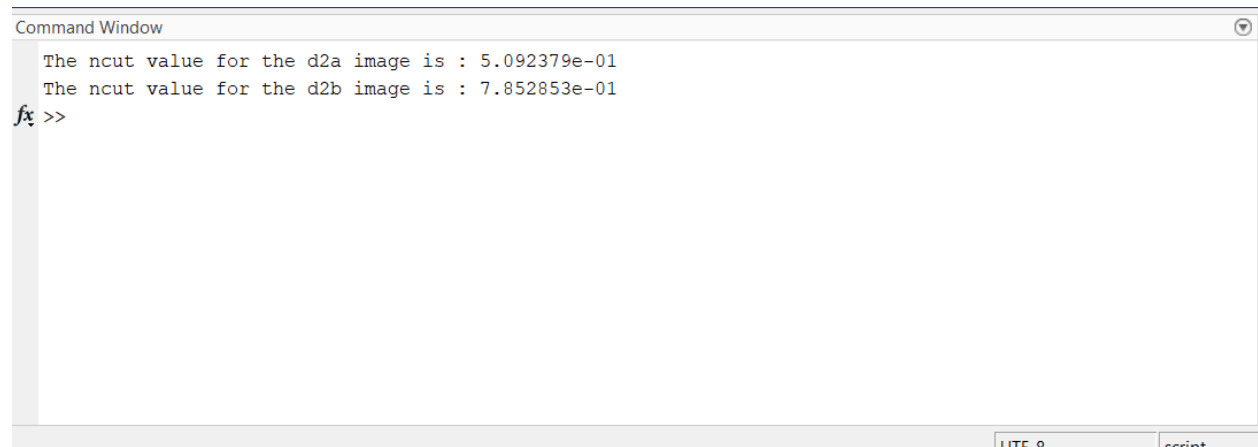
3 calculateNcut

Σε αυτή την συνάρτηση, αρχικά βρίσκουμε τον αριθμό των clusters που έχουμε και δημιουργούμε έναν πίνακα με διπλάσιο μέγεθος, αφού για κάθε cluster έχουμε 2 μετρικές. Στην συνέχεια, γεμίζουμε για κάθε cluster το πρώτο μισό του πίνακα με την 1 μετρική και το αντίστοιχο δεύτερο μισό

του πίνακα, το οποίο απεχει απο το πρώτο μισό μέγεθος πίνακα, με την δεύτερη μετρική. Αφού το κάνουμε αυτο για κάθε cluster υπολογίζουμε το Nassoc βάσει το δωσμένου τύπου.

3.1 demo3a

Τα αποτελέσματα της μη αναδρομικής μεθόδου για $K=2$ παρουσιάζονται παρακάτω.



```
Command Window
The ncut value for the d2a image is : 5.092379e-01
The ncut value for the d2b image is : 7.852853e-01
fx >>
```

3.2 demo3b

Σε αυτό το demo δημιουργούμε την αναδρομική μέθοδο. Για να συμβεί αυτό, ελέγχουμε την $ncut$ -value και τον αριθμό κορυφών κάθε cluster. Για τον αριθμό κορυφών κάθε cluster δημιουργούμε έναν πίνακα ο οποίος τις μετράει και ελέγχουμε αν οι κορυφές του cluster με τις λιγότερες είναι περισσότερες απο το δωσμένο όριο για να συνεχίσουμε. Όπως αναφέρθηκε και προηγουμένως, επειδή οι κορυφές δεν 'μεταπηδουν' σε cluster που δεν άνηκαν προηγουμένως, αρκεί σε κάθε επανάληψη να αυξάνουμε τον αριθμό των cluster κατά 1 για την νέα διχοτόμηση.

4 superpixelDescriptor

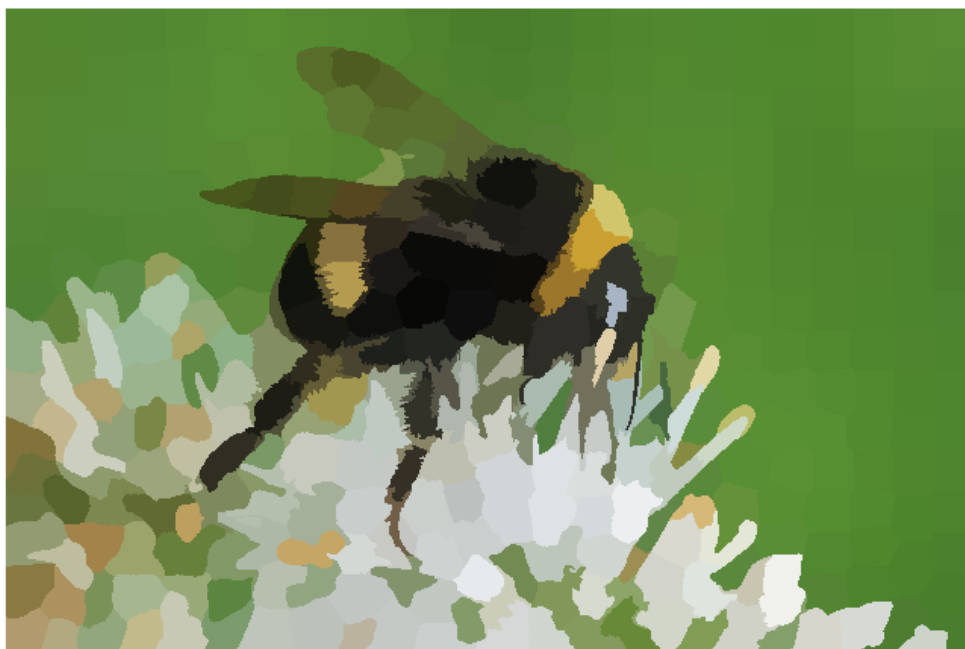
Αρχικά, βρίσκουμε πόσα superpixels υπάρχουν. Για κάθε ένα απο αυτά, βρίσκουμε απο ποια pixel της αρχικής εικόνας απαρτίζονται και βρίσκουμε τον μέσο όρο για κάθε κανάλι τους.

4.1 demo4

Τα αποτελέσματα της παραπάνω συνάρτησης φαίνονται παρακάτω:



original image



superpixelated image

Για να βρούμε τα clusters για την εικόνα με τα superpixels δημιουργούμε έναν πίνακα που περιέχει τα superpixels και το περνάμε ως είσοδο στην `myGraphSpectralClustering` και στην μη αναδρομική `normalized-cuts`. Για την αρχική εικόνα δεν ήταν δυνατόν να βρεθούν τα clusters

λογώ μεγέθους του affinity matrix.