

Ψηφιακή Επεξεργασία Εικόνας

-Εργασία 3-

Image Segmentation

Ομάδα Κατανόησης Πολυμέσων
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Η/Υ
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Άνοιξη 2018

Εισαγωγικά

Στην τρίτη και τελευταία εργασία του μαθήματος θα υλοποιήσετε τα ακόλουθα:

1. Αναπαράσταση εικόνων σαν γράφους
2. Image segmentation με τη μέθοδο *spectral clustering*
3. Image segmentation με τη μέθοδο *normalized cuts* ή αλλιώς *n-cuts* με την αναδρομική και τη μη-αναδρομική μέθοδο

Μαζί με την εκφώνηση θα βρείτε και το βοηθητικό MATLAB αρχείο `dip_hw_3.mat` το οποίο περιλαμβάνει τα δεδομένα που θα χρησιμοποιήσετε σε κάθε ερώτημα όπως εικόνες εισόδου και affinity πίνακες.

1 Εικόνες ως γράφοι

Κατασκευάστε την ρουτίνα `Image2Graph` η οποία δέχεται σαν είσοδο μια εικόνα με n κανάλια και επιστρέφει τον affinity πίνακα που περιγράφει ένα μη-κατευθυντικό γράφο $G = (V, E)$. Πιο συγκεκριμένα:

```
1 function myAffinityMat = Image2Graph(imIn)
```

όπου

`imIn`: Η $M \times N$ εικόνα εισόδου με n κανάλια.

`myAffinityMat`: Ο τετράγωνος (και συμμετρικός) affinity πίνακας που περιγράφει τον γράφο, με διαστάσεις $(M \times N) \times (M \times N)$.

Θεωρήστε πως κάθε pixel της εικόνας αποτελεί ένα *vertice* (ή *node*) του τελικού γράφου. Επιπλέον, οι τιμές των βαρών των ακμών του γράφου θα υπολογίζονται ως $A(i, j) = \frac{1}{e^{d(i, j)}}$, όπου $d(i, j)$ η Ευκλείδεια απόσταση της φωτεινότητας των καναλιών μεταξύ του i -οστού και του j -οστού pixel.

Ο γράφος που θα παράγει η ρουτίνα `Image2Graph` θα πρέπει να είναι *fully-connected*, δηλαδή για κάθε ζεύγος κορυφών i, j θα πρέπει να υπάρχει η αντίστοιχη ακμή $e_{i, j}$ με μη-μηδενικό βάρος.

2 Spectral Clustering

Στη δεύτερη ενότητα της εργασίας θα υλοποιήσετε τη μέθοδο *Spectral Clustering*. Τα βήματα της μεθόδου περιγράφονται παρακάτω:

1. Δεδομένης μιας εικόνας εισόδου, κατασκευάστε ένα μη κατευθυντικό γράφο σύμφωνα με τις προδιαγραφές της Ενότητας 1. Έστω \mathbf{W} ο affinity πίνακας που περιγράφει τον γράφο.
2. Υπολογίστε το μη-κανονικοποιημένο Λαπλασιανό πίνακα \mathbf{L} ως $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Ο διαγώνιος πίνακας \mathbf{D} ορίζεται ως: $\mathbf{D}(i, i) = \sum_j \mathbf{W}(i, j)$.
3. Λύστε το μη-γενικευμένο πρόβλημα ιδιοτιμών $\mathbf{L}x = \lambda x$, και υπολογίστε τις k μικρότερες ιδιοτιμές καθώς και τα k ιδιοδιανύσματα που αντιστοιχούν σε αυτές τις ιδιοτιμές.
4. Σχηματίστε τον πίνακα $\mathbf{U} \in \mathbb{R}^{n \times k}$ που περιέχει τα ιδιοδιανύσματα u_1, \dots, u_k σαν στήλες. Για $i = 1, \dots, n$, έστω $y_i \in \mathbb{R}^k$ το διάνυσμα που αντιστοιχεί στην i -οστή γραμμή του \mathbf{U} .
5. Ομαδοποιήστε τα σημεία $(y_i)_{i=1, \dots, n}$ με τον αλγόριθμο k -means στα clusters C_1, \dots, C_k .

Κατασκευάστε λοιπόν την ρουτίνα `mySpectralClustering` η οποία δέχεται σαν είσοδο έναν affinity πίνακα που περιγράφει έναν μη-κατευθυντικό γράφο $G = (V, E)$ και τον αριθμό των clusters και επιστρέφει τις ετικέτες των clusters στις οποίες ανήκουν οι κορυφές του γράφου. Πιο συγκεκριμένα:

```
1 function clusterIdx = mySpectralClustering(anAffinityMat, k)
```

όπου

`anAffinityMat`: Ο τετράγωνος (και συμμετρικός) affinity πίνακας που περιγράφει τον γράφο.

`k`: Ο αριθμός των clusters που θέλουμε να σχηματιστούν.

`clusterIdx`: Οι ετικέτες που δείχνουν σε πιο cluster ανήκει ο κάθε κόμβος του γράφου.

Η μη-επιβλεπόμενη διαδικασία ομαδοποίησης k -means θα πρέπει να γίνει με τη χρήση της MATLAB ρουτίνας `labels = kmeans(X, k)`, όπου \mathbf{X} ο $n \times k$ πίνακας εισόδου (ουσιαστικά n δείγματα διάστασης k) και k ο αριθμός των cluster που θέλουμε να σχηματιστούν. Το $n \times 1$ διάνυσμα `labels` που επιστρέφει η ρουτίνα περιέχει τις ετικέτες των clusters στις οποίες ανήκουν τα δείγματα εισόδου. Ο υπολογισμός των ιδιοδιανυσμάτων θα πρέπει να γίνεται με την χρήση της MATLAB ρουτίνας `eigs`.

TIP: μελετήστε καλά το documentation της ρουτίνας `eigs` (`help eigs`). Οι affinity πίνακες που θα κατασκευαστούν θα έχουν μεγάλες διαστάσεις, με αποτέλεσμα ο υπολογισμός όλων των ιδιοδιανυσμάτων θα είναι από **πολύ** χρονοβόρος έως απαγορευτικός.

3 Normalized-cuts

Για το τελευταίο κομμάτι της εργασίας θα κατασκευάσετε 2 εκδοχές (αναδρομική και μη-αναδρομική) της state-of-the-art μεθόδου για image segmentation, με την ονομασία **normalized-cuts** ή **n-cuts**.

Αναφορικά με τη μη-αναδρομική εκδοχή, τα βήματα της μεθόδου είναι τα παρακάτω:

1. Δεδομένης μιας εικόνας εισόδου, κατασκευάστε ένα μη κατευθυντικό γράφο σύμφωνα με τις προδιαγραφές της Ενότητας 1. Έστω \mathbf{W} ο affinity πίνακας που περιγράφει τον γράφο.
2. Υπολογίστε το μη-κανονικοποιημένο Λαπλασιανό πίνακα \mathbf{L} ως $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Ο διαγώνιος πίνακας \mathbf{D} ορίζεται ως: $\mathbf{D}(i, i) = \sum_j \mathbf{W}(i, j)$.
3. Λύστε το γενικευμένο πρόβλημα ιδιοτιμών $\mathbf{L}x = \lambda \mathbf{D}x$, και υπολογίστε τις k μικρότερες ιδιοτιμές καθώς και τα k ιδιοδιανύσματα που αντιστοιχούν σε αυτές τις ιδιοτιμές.
4. Σχηματίστε τον πίνακα $\mathbf{U} \in \mathbb{R}^{n \times k}$ που περιέχει τα ιδιοδιανύσματα u_1, \dots, u_k σαν στήλες. Για $i = 1, \dots, n$, έστω $y_i \in \mathbb{R}^k$ το διάνυσμα που αντιστοιχεί στην i -οστή γραμμή του \mathbf{U} .
5. Ομαδοποιήστε τα σημεία $(y_i)_{i=1, \dots, n}$ με τον αλγόριθμο k -means στα clusters C_1, \dots, C_k .

Στην συνέχεια, σαν βήμα 6, η μη-αναδρομική εκδοχή *ενώνει* σταδιακά τα k clusters που δημιουργήθηκαν. Στα πλαίσια αυτής της εργασίας όμως θα θεωρήσουμε πως η μη-αναδρομική εκδοχή της μεθόδου τελειώνει στη δημιουργία των k clusters (βήμα 5). Όπως μπορείτε να δείτε, η μόνη διαφορά μεταξύ της μεθόδου Spectral Clustering της Ενότητας 2 και της μη-αναδρομικής εκδοχής *ncuts* είναι το βήμα 3 και συγκεκριμένα το πρόβλημα των ιδιοτιμών που λύνουμε κάθε φορά.

Η αναδρομική εκδοχή της μεθόδου είναι μια υποπερίπτωση της μη-αναδρομικής για $k = 2$ ¹ (δείτε το βήμα 3 παραπάνω), δηλαδή κάθε φορά χωρίζουμε τον γράφο σε 2 κομμάτια. Μετά από κάθε διχοτόμηση (βήματα 1 μέχρι και 5 με $k = 2$), αποφασίζουμε αν θα συνεχιστεί η διχοτόμηση των συγκεκριμένων κομματιών που προέκυψαν από την διαδικασία ομαδοποίησης (k -means συγκεκριμένα). Πιο συγκεκριμένα μπορείτε να πάρετε την απόφαση με τον παρακάτω τρόπο: Αν ο αριθμός των κόμβων είτε με ετικέτα 1 είτε 2 που προκύπτουν είναι μικρότερος από ένα κατώφλι T^1 ή αν η τιμή $Ncut(A, B)$ είναι μεγαλύτερη από ένα κατώφλι T^2 , τότε η διχοτόμηση των συγκεκριμένων κομματιών που προέκυψαν σταματά. Σε διαφορετική περίπτωση κάθε ένα από τα δύο κομμάτια που προέκυψαν από την διαδικασία ομαδοποίησης χωρίζεται στα 2. Η διαδικασία συνεχίζει αναδρομικά και τερματίζει όταν κανένα κομμάτι δεν μπορεί να διχοτομηθεί για τους λόγους που αναφέρθηκαν παραπάνω.

Η μετρική $Ncut(A, B)$ για 2 ομάδες κόμβων με ετικέτες “Α” και “Β” (ή 1 και 0) ορίζεται ως εξής:

$$Ncut(A, B) = 2 - Nassoc(A, B) \quad (1)$$

με

$$Nassoc(A, B) = \frac{assoc(A, A)}{assoc(A, V)} + \frac{assoc(B, B)}{assoc(B, V)} \quad (2)$$

και

$$assoc(A, V) = \sum_{u \in A, t \in V} \mathbf{W}(u, t) \quad (3)$$

Ουσιαστικά η μετρική $assoc(A, V)$ είναι το άθροισμα όλων των βαρών μεταξύ των κόμβων που ανήκουν στην ομάδα Α (ή έχουν ετικέτα 1) προς όλους τους κόμβους του γράφου (V). Οι μετρικές $assoc(A, A)$, $assoc(B, V)$ και $assoc(B, B)$ ορίζονται αντίστοιχα. Με \mathbf{W} συμβολίζουμε τον affinity πίνακα.

Η δημοσίευση με την περιγραφή σε βάθος της μεθόδου *ncuts* σας δίνεται μαζί με την εκφώνηση της εργασίας.

Κατασκευάστε λοιπόν την ρουτίνα `myNCuts` η οποία υλοποιεί τη βασικό/κοινό κομμάτι της μεθόδου *ncuts*. Πιο συγκεκριμένα:

```
1 function clusterIdx = myNCuts(anAffinityMat, k)
```

όπου

`anAffinityMat`: Ο τετράγωνος (και συμμετρικός) affinity πίνακας που περιγράφει τον γράφο.

`k`: Ο αριθμός των clusters που θέλουμε να σχηματιστούν. Στην αναδρομική περίπτωση πρέπει $k = 2$ πάντα.

`clusterIdx`: Οι ετικέτες που δείχνουν σε πιο cluster ανήκει ο κάθε κόμβος του γράφου.

Επιπλέον, κατασκευάστε την ρουτίνα `calculateNcut` η οποία υπολογίζει την σχετική μετρική της εξίσωσης 1 για τις 2 ομάδες (clusters) που προκύπτουν από το βήμα 3 της περιγραφής της μεθόδου. Πιο συγκεκριμένα:

```
1 function nCutValue = calculateNcut(anAffinityMat, clusterIdx)
```

όπου

`anAffinityMat`: Ο τετράγωνος (και συμμετρικός) affinity πίνακας που περιγράφει τον γράφο.

`clusterIdx`: Οι ετικέτες που δείχνουν σε πιο από τα 2 cluster ανήκει ο κάθε κόμβος του γράφου.

`nCutValue`: Η τιμή της μετρικής για τις 2 ομάδες κόμβων.

¹Στην πραγματικότητα το ιδιοδιάνυσμα που αντιστοιχεί στην **μικρότερη** ιδιοτιμή δεν μας προσφέρει καμία πληροφορία (είναι σχεδόν constant) και θα μπορούσαμε να το παραλείψουμε εξ' ολοκλήρου χωρίς να δούμε διαφορά στα αποτελέσματα. Για πρακτικούς όμως λόγους, στα πλαίσια της εργασίας θα το χρησιμοποιήσουμε.

Αξιολόγηση & παραδοτέα

Κατά την υποβολή της εργασίας θα πρέπει να παραδώσετε τα αρχεία με τις συναρτήσεις:

- `Image2Graph.m`
- `mySpectralClustering.m`
- `myNCuts.m`
- `calculateNcut.m`

καθώς και μία αναφορά. **Επιπλέον**, ανά ενότητα θα πρέπει να παραδώσετε και ένα script με όνομα `demo1.m`, `demo2.m`, και `demo3a.m`, `demo3b.m` και `demo3c.m` αντίστοιχα, το οποίο θα εκτελείται **χωρίς** ορίσματα και θα παρουσιάζει τα ζητούμενα της κάθε ενότητας (δείτε τις παρακάτω παραγράφους). Στην αναφορά θα πρέπει επίσης να παρουσιάσετε όποιες σχεδιαστικές επιλογές έχετε κάνει.

Ενότητα 1

Για το demo της ενότητας 1 καλείστε να παρουσιάσετε την λειτουργία της ρουτίνας `mySpectralClustering`. Για τους σκοπούς του demo θα σας δίνονται ένας κατασκευασμένος από πριν affinity πίνακας (μεταβλητή `d1a` του βοηθητικού αρχείου `dip_hw_3.mat`). Επιπλέον, για τον λόγο του ότι ο αλγόριθμος k -means έχει ως πρώτο βήμα την τυχαία αρχικοποίηση των k κέντρων του, χρησιμοποιείτε την εντολή `rng(1)` στην αρχή του script σας έτσι ώστε να μπορείτε να επαναλάβετε το πείραμα με το ίδιο random seed και κατ' επέκταση να έχετε τα ίδια αποτελέσματα ταξινόμησης στα k clusters. Παρουσιάστε τις ετικέτες που προκύπτουν για παραμέτρους $k = 2$, $k = 3$ και $k = 4$ (σύνολο 3 πειράματα). Σχολιάστε τα αποτελέσματα.

Οι παραπάνω λειτουργίες θα πρέπει να βρίσκονται μέσα στο script `demo1.m`.

Ενότητα 2

Για το demo της ενότητας 2 καλείστε να παρουσιάσετε την λειτουργία της `mySpectralClustering` σε συνδυασμό με την ρουτίνα `Image2Graph`. Πιο συγκεκριμένα στο βοηθητικό `mat` αρχείο σας δίνονται 2 RGB εικόνες εισόδου με ονόματα μεταβλητών `d2a` και `d2b`. Μετατρέψτε κάθε εικόνα εισόδου στον αντίστοιχο γράφο (δηλαδή στον αντίστοιχο affinity πίνακα) με την χρήση της `Image2Graph` και στην συνέχεια πραγματοποιήστε την διαδικασία `spectral clustering`. Επαναλάβετε το πείραμα για αριθμούς κέντρων $k = 2$, $k = 3$ και $k = 4$ και για τις 2 εικόνες (σύνολο 6 πειράματα).

Δείξτε τα αποτελέσματα της διαδικασίας clustering πάνω στην κάθε εικόνα εισόδου. Όπως και στο `demo1`, χρησιμοποιήστε την εντολή `rng(1)` για να ελέγξετε την τυχαιότητα της αρχικοποίησης του k -means. Σχολιάστε τα αποτελέσματα.

Οι παραπάνω λειτουργίες θα πρέπει να βρίσκονται μέσα στο script `demo2.m`.

Ενότητα 3

Για το τελευταίο demo καλείστε να παρουσιάσετε την λειτουργία των ρουτινών `myNCuts` και `calculateNcut`. Για ακόμη μια φορά σαν εικόνες εισόδου χρησιμοποιήστε τις εικόνες του `demo2` (μεταβλητές `d2a` και `d2b`). Σε κάθε περίπτωση χρησιμοποιήστε την εντολή `rng(1)` για να ελέγξετε την τυχαιότητα των πειραμάτων.

1. Αρχικά κατασκευάστε τους αντίστοιχους γράφους και ομαδοποιήστε τα nodes τους με την μη-αναδρομική εκδοχή της μεθόδου `ncuts`. Επαναλάβετε το πείραμα για αριθμούς κέντρων $k = 2$, $k = 3$ και $k = 4$ και για τις 2 εικόνες (σύνολο 6 πειράματα).

Δείξτε τα αποτελέσματα της διαδικασίας πάνω στην κάθε εικόνα εισόδου. Συγκρίνετε και σχολιάστε τα αποτελέσματα της μη-αναδρομικής μεθόδου `ncuts` σε σχέση με τα αποτελέσματα που προκύπτουν από την διαδικασία `spectral`

clustering.

Οι παραπάνω λειτουργίες θα πρέπει να βρίσκονται μέσα στο script `demo3a.m`.

2. Στη συνέχεια, και αφού κατασκευάσετε τους αντίστοιχους γράφους για τις 2 εικόνες εισόδου, εκτελέστε τη αναδρομική μέθοδο n -cuts για **ένα βήμα**, δηλαδή σπάστε τον αρχικό γράφο σε 2 κομμάτια (σύνολο 2 πειράματα).

Δείξτε τα αποτελέσματα της διαδικασίας n cuts για ένα βήμα πάνω στην κάθε εικόνα εισόδου, καθώς και την τιμές των μετρικών n cut σε κάθε περίπτωση. Συγκρίνετε και σχολιάστε τα αποτελέσματα της μεθόδου n -cuts σε σχέση με τα αποτελέσματα που προκύπτουν από την διαδικασία spectral clustering για $k = 2$.

Οι παραπάνω λειτουργίες θα πρέπει να βρίσκονται μέσα στο script `demo3b.m`.

3. Για το τελευταίο κομμάτι του demo καλείστε να παρουσιάσετε την **ολοκληρωμένη** (i.e. αναδρομική) εκτέλεση της μεθόδου n cuts. Για κατώφλια T^1 και T^2 χρησιμοποιήστε τις τιμές και 5 και 0.20 αντίστοιχα. Σκεφτείτε την διαδικασία διαχωρισμού σε 2 clusters σαν την δημιουργία ενός δυαδικού (unbalanced ενδεχομένως) δέντρου, όπου κάθε κόμβος του δέντρου κρατάει την πληροφορία των ετικετών σε σχέση με τον γονέα του. Σαν εικόνες εισόδου χρησιμοποιήστε τις `d2a` και `d2b`.

Δείξτε τα αποτελέσματα της ολοκληρωμένης διαδικασίας n cuts πάνω στην κάθε εικόνα εισόδου. Συγκρίνετε και σχολιάστε τα αποτελέσματα της αναδρομικής μεθόδου n cuts σε σχέση με τα αποτελέσματα που προκύπτουν από την διαδικασία spectral clustering και τα αποτελέσματα της μη-αναδρομικής μεθόδου n cuts για $k = 2$ και $k = 3$.

Οι παραπάνω λειτουργίες θα πρέπει να βρίσκονται μέσα στο script `demo3c.m`.

Σχετικά με την υποβολή της εργασίας

Παραδώστε μία αναφορά με τις περιγραφές και τα συμπεράσματα που σας ζητούνται στην εκφώνηση. Η αναφορά θα πρέπει να επιδεικνύει την ορθή λειτουργία του κώδικά σας στις εικόνες που σας δίνονται.

Ο κώδικας θα πρέπει να είναι σχολιασμένος ώστε να είναι κατανοητό τι ακριβώς λειτουργία επιτελεί (σε θεωρητικό επίπεδο, όχι σε επίπεδο κλήσης συναρτήσεων). Επίσης, ο κώδικας θα πρέπει να εκτελείται και να υπολογίζει τα σωστά αποτελέσματα για *οποιαδήποτε* είσοδο πληροί τις υποθέσεις της εκφώνησης, και όχι μόνο για τις εικόνες που σας δίνονται.

Απαραίτητες προϋποθέσεις για την βαθμολόγηση της εργασίας σας είναι ο κώδικας να εκτελείται χωρίς σφάλμα, καθώς και να τηρούνται τα ακόλουθα:

- Υποβάλετε ένα και μόνο αρχείο, τύπου `zip`.
- Το όνομα του αρχείου πρέπει να είναι `AEM.zip`, όπου `AEM` είναι τα τέσσερα ψηφία του Α.Ε.Μ. του φοιτητή της ομάδας.
- Το προς υποβολή αρχείο πρέπει να περιέχει τα αρχεία κώδικα Matlab και το αρχείο `report.pdf` το οποίο θα είναι η αναφορά της εργασίας.
- Η αναφορά πρέπει να είναι ένα αρχείο τύπου PDF, και να έχει όνομα `report.pdf`.
- Όλα τα αρχεία κώδικα πρέπει να είναι αρχεία κειμένου τύπου UTF-8, και να έχουν κατάληξη `m`.
- Το αρχείο τύπου `zip` που θα υποβάλετε δεν πρέπει να περιέχει κανέναν φάκελο.
- Μην υποβάλετε τις εικόνες που σας δίνονται για πειραματισμό.
- Μην υποβάλετε αρχεία που δεν χρειάζονται για την λειτουργία του κώδικά σας, ή φακέλους/αρχεία που δημιουργεί το λειτουργικό σας, πχ `"Thumbs.db"`, `"DS_Store"`, `"directory"`.
- Για την ονομασία των αρχείων που περιέχονται στο προς υποβολή αρχείο, χρησιμοποιείτε μόνο αγγλικούς χαρακτήρες, και όχι ελληνικούς ή άλλα σύμβολα, πχ `"#"`, `"$"`, `"%"` κλπ.