

# Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Ψηφιακή επεξεργασία εικόνας

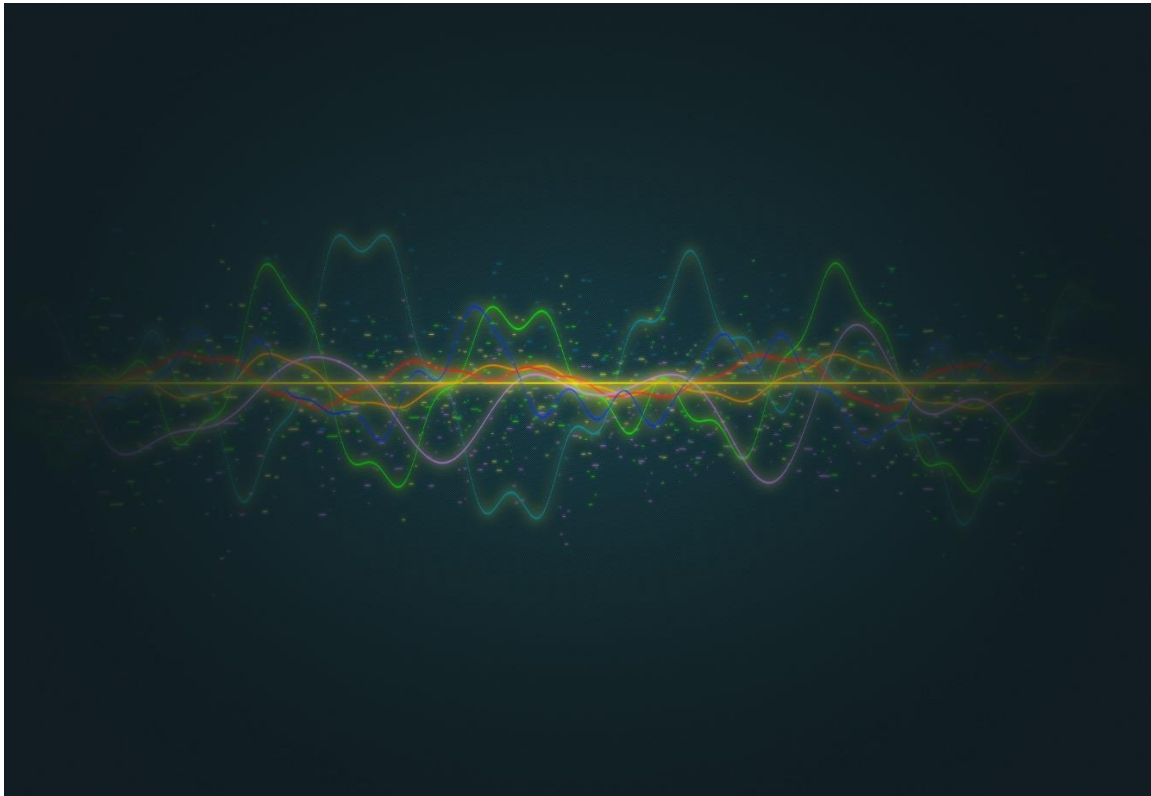
Εργασία 2

2019



Ευάγγελος Τσιμπούρης ΑΕΜ 8257

# Image segmentation



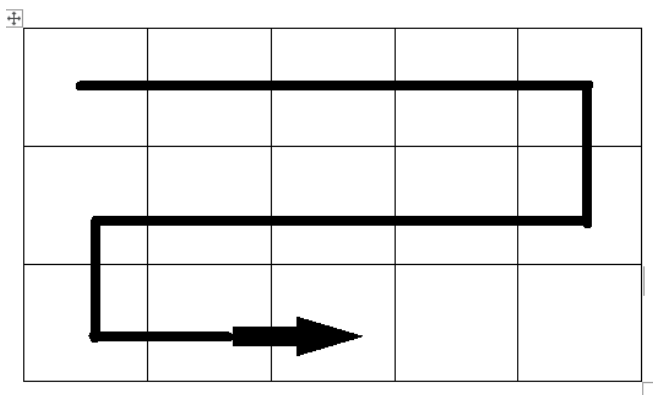
## Περιεχόμενα

Εικόνες ως γράφοι .....	3
Spectral Clustering .....	4
N-Cuts .....	7
Τελική υλοποίηση Ncuts .....	11

## Εικόνες ως γράφοι

Στην ενότητα αυτή θα αναφερθούν οι βασικοί τύποι και οι παραδοχές που έχουν γίνει για τον υπολογισμό του Affinity Matrix. Κατόπιν θα παρουσιαστούν τα αποτελέσματα του demo.

Ως πρώτο βήμα έχουμε την μετατροπή μιας εικόνας διαστάσεων (M,N) σε ένα διάνυσμα διαστάσεων (M\*N,1) . Στη πιο κάτω εικόνα φαίνεται η σειρά με την οποία σαρώνουμε την εικόνα στις 2 διαστάσεις.



Στην συνέχεια αφού δημιουργήσουμε το διάνυσμα της εικόνας για κάθε ένα από τα κανάλια της χρειάζεται να ορίσουμε την ευκλείδεια απόσταση μεταξύ των pixel της. Έστω  $i$  ο δείκτης ενός pixel του διανύσματος :  $i \in [1, M*N]$  και ομοίως ο δείκτης  $j \in [1, M*N]$  . Θεωρώ την ευκλείδεια απόσταση ως την απόσταση σε φωτεινότητα των 3 καναλιών όπως φαίνεται στην (1).

$$d_{ij}^2 = (r_i - r_j)^2 + (g_i - g_j)^2 + (b_i - b_j)^2 \quad (1)$$

Στην συνέχεια υπολογίζουμε τον Affinity Matrix με στοιχεία τα βάρη των ακμών μεταξύ των pixel. Ο τύπος που χρησιμοποιήθηκε είναι της εξίσωσης (11) του paper και όχι αυτός που δίνει η εκφώνηση. Στην ουσία η μόνη διαφορά είναι η διαίρεση με την standard deviation όπως φαίνεται στην (2).

$$W_{ij} = A_{ij} = e^{-\frac{d_{ij}^2}{\sigma_x^2}} \quad (2)$$

# Spectral Clustering

Η συνάρτηση αυτή έχει στόχο την ομαδοποίηση των pixels βάση της ομοιότητας τους σε χρώμα. Οι είσοδοι αυτής της συνάρτησης είναι ο Affinity πίνακας που υπολογίστηκε στην προηγούμενη ενότητα και ο αριθμός των  $k$  clusters επιθυμούμε στην έξοδο. Συνοπτικά αναφέρονται τα βήματα του αλγορίθμου:

Υπολογισμός του λαπλασιανού πίνακα  $L$  βάση της εξίσωσης (3).

$$L = D - W \quad (3)$$

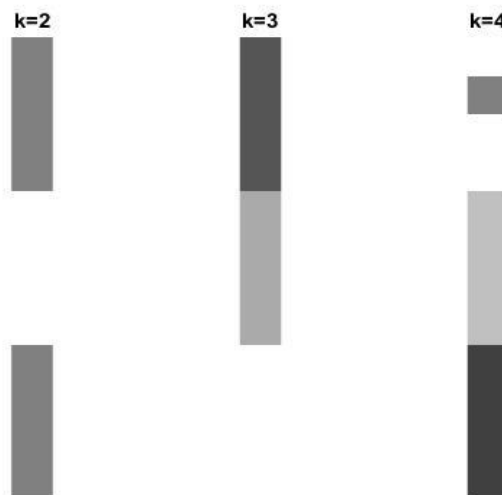
Όπου  $W$  ο Affinity πίνακας και  $D$  ο διαγώνιος πίνακας που ορίζεται από την (4).

$$D_{ii} = \sum_j W_{ij} \quad (4)$$

Στην συνέχεια βρίσκουμε τις  $k$  μικρότερες ιδιοτιμές του πίνακα  $L$  και δημιουργούμε τον πίνακα  $U$  με στοιχεία τα  $k$  ιδιοδύνασματα του πίνακα  $L$ .

Στο τελευταίο βήμα ομαδοποιούμε τα στοιχεία του  $U$  με τον αλγόριθμο  $k$ -means σε  $k$  κέντρα. Ο αριθμός  $i$  της γραμμής του  $U$  αντιστοιχεί στο  $i$  pixel της vectorized εικόνας. Επομένως και το label που θα προκύψει από τον  $k$ -means θα αντιστοιχεί στο cluster στο οποίο ομαδοποιήθηκε το pixel  $i$ .

Παρακάτω βλέπουμε τα αποτελέσματα της συνάρτησης `mySpectralClustering` στον προϋπολογισμένο Affinity πίνακα `d1a` του `demo1`:



Εικόνα (1)

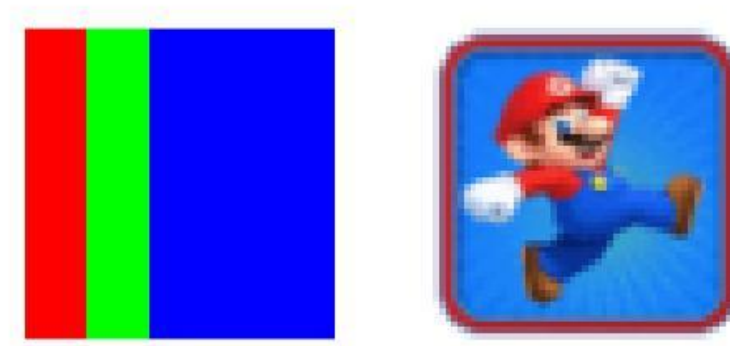
Για να ελέγξουμε τα αποτελέσματα της ομαδοποίησης παρακάτω παραθέτουμε την πρώτη γραμμή του πίνακα affinity.

1	2	3	4	5	6	7	8	9	10	11	12
0.9000	0.9000	0.9000	0.9000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000

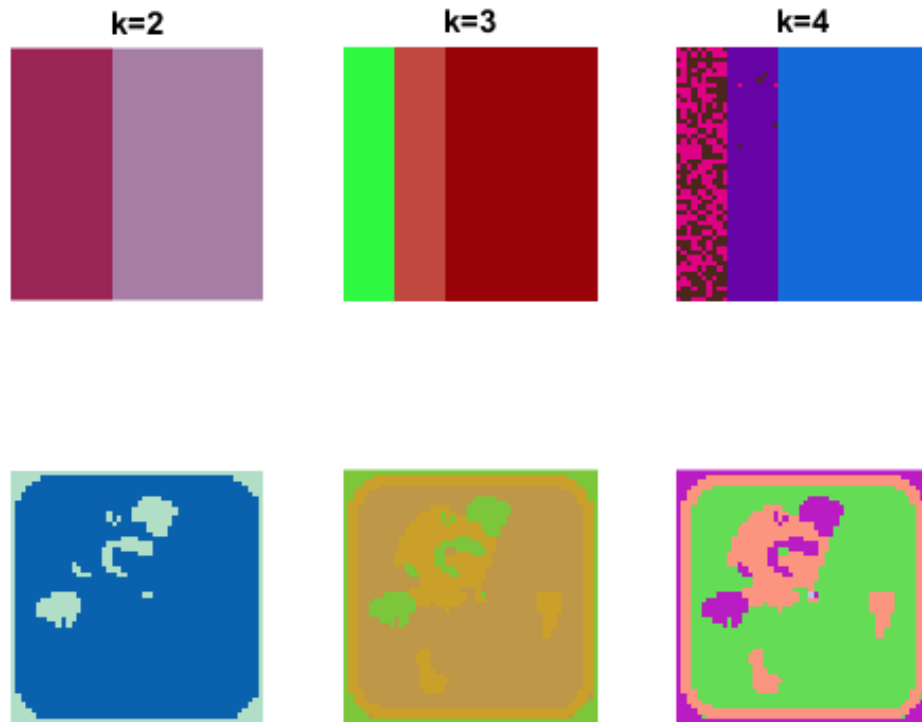
Παρατηρούμε ότι το πρώτο στοιχείο έχει μεγαλύτερη ομοιότητα με τα πρώτα 4 στοιχεία. Με βάση τα αποτελέσματα της ομαδοποίησης της εικόνας (1) έχουμε σωστό clustering μόνο στο 2<sup>ο</sup> πείραμα ( $k=3$ ). Αυτό συμβαίνει γιατί «ωθήσαμε» τον kmeans στο 1<sup>ο</sup> πείραμα να δημιουργήσει 2 ακριβώς ομάδες και στο 3<sup>ο</sup> πείραμα 4.

Για τον έλεγχο της ομαδοποίησης μπορούμε να ελέγξουμε κάθε μια από τις γραμμές του πίνακα affinity για να εντοπίσουμε με ποια στοιχεία παρουσιάζεται η μεγαλύτερη ομοιότητα. Σε μεγάλες εικόνες όμως αυτό είναι αδύνατο γιατί πχ σε μια εικόνα μεγέθους 100x100 ο πίνακας affinity έχει μέγεθος 10000x10000. Σε αυτές τις περιπτώσεις υπάρχουν μετρικές όπως η separation και η cohesion που μας δείχνουν πόσο καλή ήταν η ομαδοποίηση.

Σε επόμενο βήμα το demo2 υπολογίζει διάφορες ομαδοποιήσεις για τις παρακάτω αρχικές εικόνες.



Το demo2 αφού υπολογίσει τα labels της κάθε ομάδας, αναθέτει ένα τυχαίο χρώμα σε κάθε ομάδα. Στη συνέχεια προβάλλει τις ομάδες που βρέθηκαν στις αρχικές διαστάσεις των εικόνων. Αυτό γίνεται για να μπορούμε να ελέγξουμε αν η ομαδοποιήσεις είχαν κάποιο «νόημα».



Αποτελέσματα demo2.

#### Εικόνα (2)

Για την πρώτη εικόνα φαίνεται πως για  $k=3$  έχουν βρεθεί σωστά οι 3 ομάδες-χρώματα, ενώ για  $k=4$  η ομαδοποίηση δεν είναι καλή αφού ζητάμε 4 ομάδες σε μια τραχωματική εικόνα. Το «σπάσιμο» του πρώτου χρώματος είναι τυχαίο και έχει να κάνει με την αρχική κατανομή των κέντρων.

Για τη δεύτερη εικόνα έχουμε μεγαλύτερη ποικιλία στην παλέτα με αποτέλεσμα ακόμη και οι 4 ομάδες να είναι ίσως λίγες. Για  $k=2$  φαίνεται να γίνεται απλά ένας διαχωρισμός μεταξύ του άσπρου της εικόνας και των υπόλοιπων χρωμάτων. Ως προς τα πίξελ των ομάδων για  $k=3,4$  φαίνεται ότι το παντελόνι του Mario «μπερδεύεται» με το background γιατί και τα 2 είναι περίπου μπλε. Τα γάντια που είναι άσπρα σωστά έχουν ομαδοποιηθεί με το επίσης άσπρο περίγραμμα και

το άσπρο των ματιών. Το καπέλο και τα χέρια που είναι κόκκινα βρίσκονται στην ίδια ομάδα με το κόκκινο περίγραμμα αλλά και τις καφέ μπότες. Αυτό συμβαίνει γιατί το καφέ βρίσκεται πιο «κοντά» στο κόκκινο από ότι στο μπλε ή το άσπρο.

## N-Cuts

Ο Ncuts είναι και αυτός ένας αλγόριθμος image segmentation που βασίζεται στη μετρική normalized cuts. Έστω ότι σπάμε τον γράφο  $V$  σε 2 υπογράφους  $A, B$ , η  $Ncut(A, B)$  δίνεται από τις εξισώσεις (5), (6) και (7).

$$Ncut(A, B) = 2 - Nassoc(A, B) \quad (5)$$

$$Nassoc(A, B) = \frac{assoc(A, A)}{assoc(A, V)} + \frac{assoc(B, B)}{assoc(B, V)} \quad (6)$$

$$assoc(A, V) = \sum_{u \in A, t \in V} w(u, t) \quad (7)$$

Στην ουσία η (6) υπολογίζεται από 4 αθροίσματα στοιχείων του affinity πίνακα.

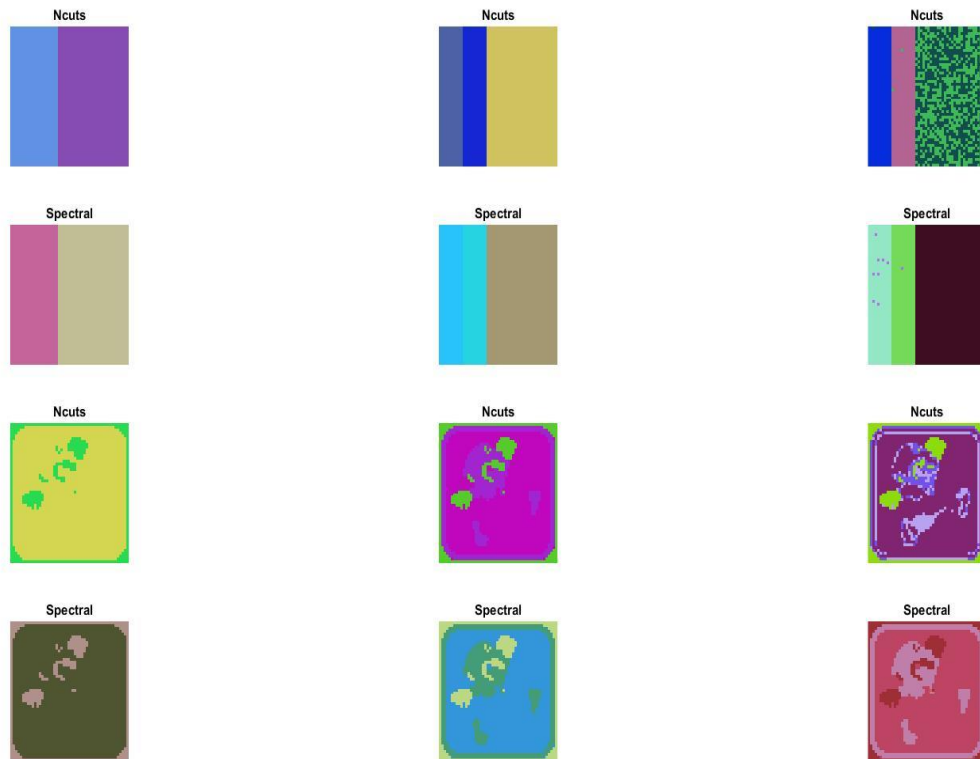
Σύμφωνα με το paper στόχος είναι να ελαχιστοποιηθεί η μετρική  $Ncut$  της (5). Μετά από πράξεις καταλήγουμε η σύνθηκη που ελαχιστοποιεί την (5) δίνεται από την (8).

$$(D - W)y = \lambda Dy \quad (8)$$

Όπου ο  $D$  ορίστηκε στην (4) και  $W$  ο affinity πίνακας. Στην ουσία λύνουμε το πρόβλημα ιδιοτιμών της (8) και ομαδοποιούμε το αποτέλεσμα. Παρόμοια λογική ακολουθήσαμε και στην ενότητα του Spectral Clustering.

Στα πλαίσια της εργασίας έγιναν 2 υλοποιήσεις. Η πρώτη λύνει το πρόβλημα ιδιοτιμών της εξίσωσης (8) και παίρνει σαν είσοδο τον αριθμό των κέντρων για την ομαδοποίηση του kmeans.

Παρακάτω φαίνονται τα αποτελέσματα του demo3a :



Εικόνα (3)

Αριστερά clustering για  $k = 2$ , κεντρικά για  $k = 3$ , δεξιά για  $k = 4$ .

Παραπάνω φαίνονται τόσο τα αποτελέσματα της μέθοδου Ncuts όσο και της Spectral clustering ώστε να γίνουν πιο εμφανείς οι διαφορές τους.

Στην πρώτη τριχρωματική εικόνα δεν γίνεται αντιληπτή κάποια διαφορά εκτός από ότι δουλεύουν σωστά μόνο για  $k=3$  όπως αναμένονταν.

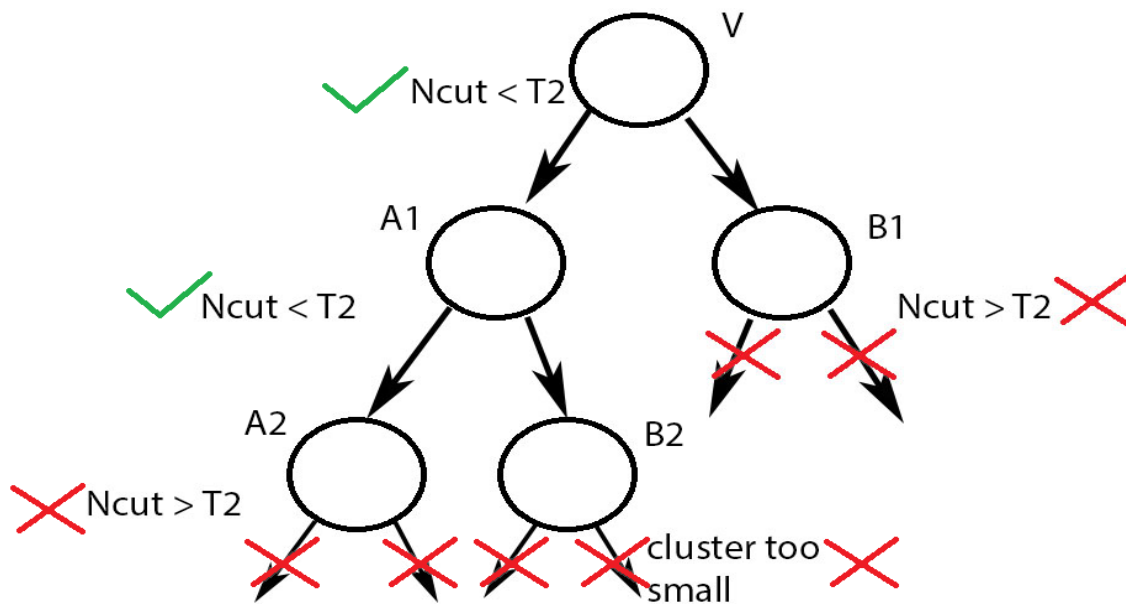
Στην δεύτερη εικόνα οι ομαδοποιήσεις των 2 μεθόδων μοιάζουν αρκετά για  $k=2,3$  αλλά για  $k=4$  παρουσιάζουν διαφορές. Η Ncut έχει ομαδοποιήσει το καπέλο του Μάριο με το background ενώ μια πιο σκοτεινή απόχρωση του παντελονιού έχει



ομαδοποιηθεί μαζί με το καφέ των παπουτσιών του. Εκ πρώτης όψης φαίνεται ότι η Ncut είναι πιο ευαίσθητη για διάφορες αποχρώσεις από την Spectral.

Η δεύτερη υλοποίηση αφού λύσει το πρόβλημα ιδιοτιμών σπάει σε 2 ομάδες τον πίνακα  $U$  προς ομαδοποίηση. Στην συνέχεια υπολογίζει την μετρική Ncut και αποφασίζει αν η ομαδοποίηση ήταν καλή. Αν η τιμή της Ncut δεν ξεπεράσει ένα threshold τότε η ομαδοποίηση αποθηκεύεται και δημιουργούνται αναδρομικά 2 νέοι κόμβοι οι οποίοι εξετάζουν αν κάθε μια από τις 2 προηγούμενες ομάδες επιδέχεται περαιτέρω ομαδοποίηση. Μια ομαδοποίηση μπορεί επίσης να σταματήσει αν κάποιο cluster που δημιουργήθηκε ήταν πολύ μικρό.

Παρακάτω φαίνεται ένα παράδειγμα δένδρου της δεύτερης υλοποίησης:

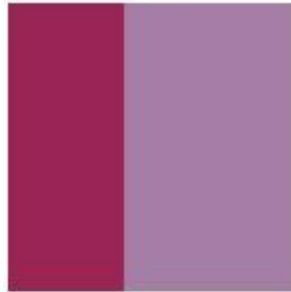


Εικόνα (4)

Αναδρομική Ncut

Το παραπάνω δένδρο θα μπορούσαμε να πούμε ότι κάνει το segmentation της πρώτης τριχρωματικής εικόνας αφού δημιουργεί 3 ομάδες.

Σαν πρώτη δοκιμή της αναδρομικής μεθόδου δημιουργήσαμε ακριβώς 2 ομάδες για τις εικόνες μας και μετά διακόψαμε την αναδρομή. Παρακάτω φαίνονται τα αποτελέσματα του demo3b:



Τα αποτελέσματα είναι ακριβώς τα ίδια με αυτά της μη αναδρομικής Ncut για  $k=2$ .

## Τελική υλοποίηση Ncuts

Σαν τελική υλοποίηση της Ncut έχουμε την πλήρη αναδρομική εκδοχή της. Στα πλαίσια αυτής της υλοποίησης έγιναν 3 πειράματα. T1 είναι το threshold για τη μικρότερη επιτρεπτή ομάδα και T2 είναι το threshold για τη μέγιστη επιτρεπτή τιμή της Ncut.

- ➔ Clustering στην εικόνα του Μάριο για  $T1 = 0.2$ ,  $T2 = 5$
- ➔ Clustering στην εικόνα του Μάριο για  $T1 = 1$ ,  $T2 = 50$ . Σκοπός αυτού του πειράματος είναι να δούμε πόσα και τι clusters θα δημιουργηθούν για λιγότερο αυστηρά κριτήρια ως προς την Ncut.
- ➔ Clustering στην τριχρωματική εικόνα για  $T1=0.2$ ,  $T2 = 5$  με θόρυβο στο 10% της εικόνας. Σκοπός αυτού του πειράματος είναι να δούμε κατά πόσο θα επηρεαστεί η ομαδοποίηση από τον θόρυβο . Επίσης τα clusters που θα δημιουργηθούν θα πάρουν χρώμα ενός δείγματος της αρχικής εικόνας και όχι ένα τυχαίο χρώμα. Έτσι θα δείξουμε καλύτερα το denoising effect της ομαδοποίησης μας.



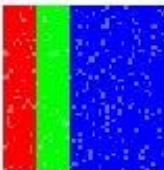
**T1 = 0.2 T2 = 5**



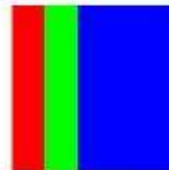
**T1 = 1 T2 = 50**



**10% noise**



**T1 = 0.2 T2 = 5**



Παραπάνω φαίνονται τα αποτελέσματα των πειραμάτων.

Στο πρώτο πείραμα βρέθηκαν 3 ομάδες . Ίσως να είναι λίγες σε αριθμό αλλά αυτό δικαιολογείτε από το γεγονός ότι το threshold για την τιμή της  $N_{cut}$  είναι αρκετά αυστηρό στα 0.2. Οι ομάδες που βρέθηκαν φαίνεται να είναι ίδιες με αυτές για  $k = 3$ .

Στο δεύτερο πείραμα άλλαξε η τιμή του threshold της  $N_{cut}$  και αυτό στην ουσία οδήγησε στο να γίνουν επιτρεπτά λιγότερα καλά «σπασίματα». Πράγματι βρέθηκαν 10 ομάδες και εύκολα παρατηρούμε ότι κάποιες από αυτές προήλθαν από μικρές μεταβολές στις αποχρώσεις του ίδιου χρώματος.

Στην τρίτη εικόνα φαίνεται πως ο θόρυβος δεν επηρέασε το αποτέλεσμα και μάλιστα η μέθοδος εντόπισε αυτόματα 3 ομάδες που είναι και το σωστό. Η τελική εικόνα είναι πλήρης ανακατασκευή της πρώτης χωρίς θόρυβο.

Από τα παραπάνω πειράματα φαίνεται πως η μέθοδος  $N_{cut}$  μπορεί να μας εντοπίσει αυτόματα τον βέλτιστο αριθμό ομάδων και να αποφύγουμε προβλήματα κακής ομαδοποίησης όπως αυτά της τραχωματικής εικόνας για  $k=2,4$  στο spectral clustering και στην μη αναδρομική  $N_{cuts}$ .