



Α.Ε.Ι ΠΕΙΡΑΙΑ ΤΕΧΝΟΛΟΓΙΚΟΥ ΤΟΜΕΑ

«Μηχανική Όραση»

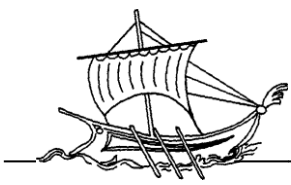
ΜΠΕΛΕΡΗΣ ΒΥΡΩΝ – ΣΤΑΥΡΟΣ

Επιβλέπων Καθηγητής: Δρ. Γρ. Νικολάου

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΑΥΤΟΜΑΤΙΣΜΟΥ

Οκτώβριος 2016



ΠΕΡΙΛΗΨΗ

Σκοπός της παρούσας πτυχιακής εργασίας είναι η μελέτη μεθόδων και τεχνικών, που σχετίζονται με τη Μηχανική Όραση. Η μηχανική όραση είναι συνδυασμός της τεχνητής νοημοσύνης και της επεξεργασίας εικόνας, όπου αναπαράγεται αλγοριθμικά η αίσθηση της όρασης, και σχετίζεται με τη θεωρία και την τεχνολογία, που αφορά τη σχεδίαση και κατασκευή συστημάτων, τα οποία λαμβάνουν, επεξεργάζονται και αναλύουν δεδομένα από ψηφιακές εικόνες. Στα παρακάτω κεφάλαια θα περιγραφούν και θα αναλυθούν διάφορες τεχνικές, που εφαρμόζονται για την επεξεργασία εικόνας, καθώς και θα υλοποιηθούν παραδείγματα μέσω Matlab.

ABSTRACT

The aim of this final paper is the study of methods associated with Computer Vision. The Computer vision is a combination of artificial intelligence and image processing, in which the sense of sight algorithmically reproduced and is combined with the theory and the technology on the design and manufacturing systems, which receive, process and analyze data for digital images. In the following sections will be described and analyzed various techniques applied to image processing and examples will be implemented via Matlab.

Keywords: Mechanical Vision, Computer Vision, Matlab, Segmentation, object detection



ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο / Η κάτωθι υπογεγραμμένος / α Μπελέρης Βίρων-Σταύρος
του Βαϊκωλάφ, με αριθμό μητρώου 39545 φοιτητής / τρια του
Τμήματος Μηχανικών Αυτοματισμού Τ.Ε. του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την
εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του
συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και
πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται
αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη
αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα
του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος
φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα
του έχει απονεμίσει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η
Συνέλευση του Τμήματος με νέα απόφαση της, μετά από αίτηση του ενδιαφερόμενου, του
αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα
καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός
ημερολογιακού βμήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα
προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Ο Δηλών

Βίρων-Σταύρος
Μπελέρης

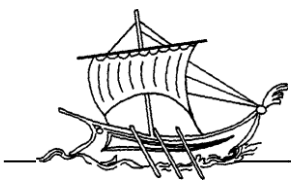
Ημερομηνία

14/11/2016



ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ.....	2
ΠΕΡΙΕΧΟΜΕΝΑ.....	3
1. ΕΙΣΑΓΩΓΗ.....	6
2. ΕΥΡΕΣΗ ΑΚΜΩΝ.....	6
2.1 Ανίχνευση Ακμών με αλγόριθμο Canny.....	7
2.2 Παράδειγμα Αλγόριθμου Canny.....	9
2.3 Κώδικας MatLab Παραδείγματος Αλγόριθμου Canny.....	11
3. ΧΡΩΜΑΤΙΚΟΙ ΧΩΡΟΙ.....	14
3.1 Μοντέλο RGB.....	15
3.2 Παράδειγμα Χρωματικού Μοντέλου RGB στο MATLAB.....	16
3.3 Μοντέλα HIS και HSV.....	17
3.4 Παράδειγμα Μοντέλων HSI και HSV στο MATLAB.....	19
3.5 Ο Χρωματικός χώρος YCbCr.....	20
3.6 Παράδειγμα μοντέλου YCbCr στο MATLAB.....	21
3.7 Μοντέλο CieLab.....	22
3.8 Παράδειγμα μοντέλου CieLab στο MATLAB.....	26
4. ΜΑΘΗΜΑΤΙΚΗ ΜΟΡΦΟΛΟΓΙΑ.....	27
4.1 Μορφολογικοί Τελεστές Συστολής και Διαστολής.....	28
4.2 Παραδείγματα Συστολής και Διαστολής.....	30
4.3 Μορφολογικοί Τελεστές Άνοιγμα και Κλείσιμο.....	31
4.4 Παραδείγματα Ανοίγματος και κλεισίματος.....	32
4.5 Κύριες Μορφολογικές Συναρτήσεις του MatLab.....	33
4.6 Μορφολογική Ανακατασκευή.....	36
4.6.1 Άνοιγμα με Ανακατασκευή.....	37
4.6.2 Κλείσιμο με Ανακατασκευή.....	37
4.6.3 Παραδείγματα Ανοίγματος και Κλεισίματος με Ανακατασκευή.....	38
5. ΤΜΗΜΑΤΟΠΟΙΗΣΗ Ή ΚΑΤΑΤΜΗΣΗ ΕΙΚΟΝΑΣ.....	40
5.1 Κατωφλίωση Ιστογράμματος.....	41
5.2 Παράδειγμα τμηματοποίηση με κατωφλίωση ιστογράμματος.....	41
5.3 Αλγόριθμος Μετασχηματισμού Κλιμακωτών Αναλλοίωτων Χαρακτηριστικών..	43
5.4 Παράδειγμα κατάτμησης και εύρεσης αντικειμένου με αλγόριθμο SIFT.....	45
5.5 Αλγόριθμος επιταχυνθέντων ισχυρών χαρακτηριστικών (SURF).....	52
5.6 Παράδειγμα εύρεσης αντικειμένου με τη χρήση του αλγόριθμου SURF.....	54
5.7 Αλγόριθμος κατάτμησης της εικόνας με βάση το χρώμα.....	58
5.8 Μέθοδος Κατάτμησης με Υφή.....	58
5.9 Κατάτμηση βάση περιοχών.....	59
5.10 Τοπολογικά Χαρακτηριστικά.....	59
6. ΠΕΡΙΓΡΑΦΗ ΣΧΗΜΑΤΟΣ.....	60
6.1 Chain Code.....	61
6.2 Γεωμετρικές ροπές (Image Moments).....	63
6.3 Είδη γεωμετρικών ροπών.....	64



6.4	Εξαγωγή χαρακτηριστικών με βάση τις ροπές	65
6.5	Γεωμετρικά χαρακτηριστικά.....	68
7.	ΣΥΜΠΕΡΑΣΜΑΤΑ	71
8.	ΒΙΒΛΙΟΓΡΑΦΙΑ	72



1. ΕΙΣΑΓΩΓΗ

Η μηχανική όραση είναι η επιστήμη όπου αναπαράγεται η όραση μέσω αλγορίθμων. Η αναπαραγωγή των αλγορίθμων γίνεται μέσω ηλεκτρονικού υπολογιστή ή ακόμα και ρομπότ. Μέσω ψηφιακών εικόνων λαμβάνονται δεδομένα (φωτογραφίες, βίντεο κ.τ.λ.) και σύμφωνα με την τεχνολογία που υπάρχει, δημιουργούνται συστήματα μηχανικής όρασης [1][6].

Κάποιες από τις θεωρίες και κάποια από τα μοντέλα που εφαρμόζονται στη μηχανική όραση είναι:

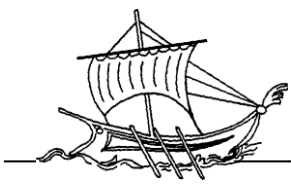
1. Ο έλεγχος των διεργασιών
2. Παρακολούθηση-Επιτήρηση
3. Αλληλεπίδραση συστήματος-ανθρώπου

2. ΕΥΡΕΣΗ ΑΚΜΩΝ

Η εύρεση ακμών θεωρείται πως αποτελεί βασικό εργαλείο, όσον αφορά τους τομείς ανάλυσης ψηφιακής εικόνας και τους τομείς της μηχανικής όρασης. Οι ακμές μπορούν να προσδιορίσουν, να αναγνωρίσουν και να περιγράψουν αντικείμενα, τα όποια περιέχονται σε ψηφιακές εικόνες. Με τον όρο ακμή προσδιορίζεται το σύνορο εκείνο μεταξύ δύο ομογενών περιοχών της εικόνας. Οι περιοχές αυτές χαρακτηρίζονται από εντάσεις φωτεινότητας διαφορετικές μεταξύ τους. Η ακμή αποτελεί την τοπική μεταβολή της φωτεινότητας. Ασυνέχειες ή απότομες μεταβολές στην ένταση της φωτεινότητας συνήθως οφείλονται σε:

- Ασυνέχειες στο βάθος ή στον προσανατολισμό επιφανειών
- Αλλαγές στις ιδιότητες του υλικού
- Μεταβολές στον φωτισμό

Το γεγονός πως δεν υφίσταται ευρέος αποδεκτός μαθηματικός όρος των ακμών οφείλεται στην πολυπλοκότητα του περιεχομένου της ψηφιακής εικόνας, αλλά και στην παρεμβολή



υψηλού επιπέδου μηχανισμών όρασης του ορίου ενός αντικειμένου στην ανθρώπινη αντίληψη.

Με δεδομένο πως οι περισσότερες μέθοδοι, που εφαρμόζονται για την εύρεση και την ανίχνευση ακμών βασίζονται στο ότι μια ακμή παρουσιάζεται εκεί όπου υπάρχει ασυνέχεια στη συνάρτηση έντασης ή απότομη κλίση, οπότε με την παράγωγο των τιμών της έντασης μπορούν να εντοπιστούν τα σημεία των ακμών, στα όποια η παράγωγος αυτή παρουσιάζει τοπικό μέγιστο.

2.1 Ανίχνευση Ακμών με αλγόριθμο Canny

Θεωρώντας μια συνεχή μονοδιάστατη συνάρτηση $f(x)$, για τον εντοπισμό τοπικών μεγίστων γίνεται χρήση της πρώτης και δεύτερης παραγώγου. Απαιτείται να βρεθεί η θέση ακρότατου σημείου, καθώς και τοπικού ελάχιστου ή μέγιστου. Οι διάφοροι αλγόριθμοι, που έχουν αναπτυχθεί βασίζονται στις χαρακτηριστικές ιδιότητες των $f'(x)$ και $f''(x)$.

Ο πιο ευρέως διαδομένος αλγόριθμος εύρεσης ακμών είναι ο ανιχνευτής ακμών Canny. Η ανάπτυξη του αλγόριθμού βασίζεται στα παρακάτω κριτήρια [2]:

- Ιδιαίτερα σημαντικό κρίνεται να ανιχνεύονται όλες οι πραγματικές τιμές και μην ανιχνεύονται ακμές που δεν υφίστανται.
- Απαιτείται ελαχιστοποίηση μεταξύ της απόστασης της πραγματικής ακμής και της ακμής που εντοπίζει ο αλγόριθμος. Τα όρια της ακμής χρειάζεται να είναι σαφή.
- Μια πραγματική τιμή δε θα πρέπει να δίνει περισσότερες από μία απόκρισεις.

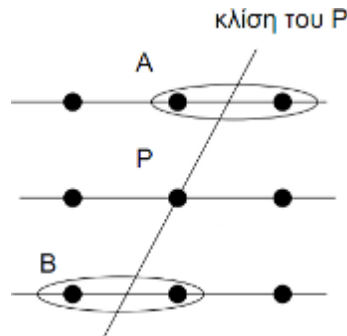
Εφαρμόζοντας τον αλγόριθμο Canny ακολουθούνται τα παρακάτω βήματα:

- ❖ Φιλτράρεται και λειαινεται η εικόνα μέσω ενός φίλτρου Gaussian καθορισμένης τυπικής απόκλισης σ και μηδενικής μέσης τιμής.



- ❖ Προσδιορίζεται η κλίση της φιλτραρισμένης εικόνας με μάσκες παραγωγίσης, όπως Sobel, Prewitt, Roberts, κτλ.
- ❖ Γίνεται καταστολή των μη μέγιστων τιμών, απαλείφοντας ως ακμές εκείνες που έχουν τη μικρότερη τιμή τοπικά. Το κάθε pixel P ορίζεται ως pixel ακμής, όταν το μέτρο της κλίσης του είναι μεγαλύτερο από το μέσο πλάτος των κλίσεων στις περιοχές A και B.

Στο σχήμα 2.1 παρουσιάζεται το μέσο πλάτος κλίσης των περιοχών A και B.



Σχήμα 2.1: Μέσο πλάτος κλίσης των περιοχών A και B

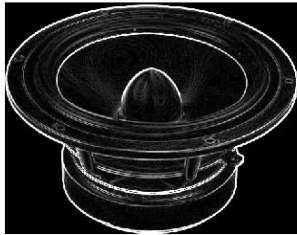
- ❖ Στη συνέχεια χρησιμοποιούνται δύο κατώφλια T_1 και T_2 , όπου $T_1 < T_2$. Εντοπίζεται το πρώτο pixel που έχει κλίση μεγαλύτερη του T_2 και εξετάζονται διαδοχικά τα γειτονικά pixels, μέχρι να βρεθεί κάποιο με μικρότερη κλίση από αυτή του T_2 . Για την καταφλίωση σε ένα δεδομένο pixel, με χρήση ελάχιστης μέγιστης φωτεινότητας ακολουθούνται τα παρακάτω:
 - Όταν το μέγεθος της κλίσης είναι μικρότερο του T_1 , η τιμή της κλίσης γίνεται μηδέν.
 - Όταν η κλίση είναι T_2 , το pixel παραμένει ως έχει.
 - Όταν η κλίση βρίσκεται μεταξύ T_1 και T_2 , η τιμή κλίσης γίνεται μηδέν.
 - Pixels με κλίση τουλάχιστον ίση με T τα οποία να αποτελούν την διαδρομή



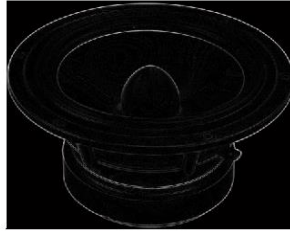
α) Αρχική εικόνα



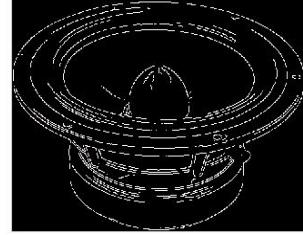
β) Εφαρμόζοντας Gaussian φίλτρο



γ) Το πλάτος της κλίσης



δ) Μετά την καταστολή

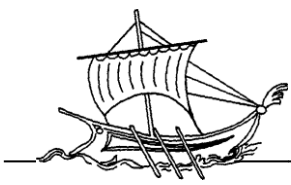


ε) Τελική εικόνα

Σχήμα 2.2: Στάδια εφαρμογής αλγόριθμου Canny

2.2 Παράδειγμα Αλγόριθμου Canny

Οι παρακάτω εικόνες απεικονίζουν τα στάδια εφαρμογής του αλγόριθμου Canny. Για την υλοποίησή τους χρησιμοποιήθηκε κώδικας σε γλώσσα MATLAB.



Α) Αρχική εικόνα



Β) Εικόνα οριζόντιων Ακμών Sobel



Γ) Εικόνα κατακόρυφων Ακμών Sobel



Δ) Εικόνα Ακμών Sobel



Ε) Εικόνα οριζόντιων Ακμών Prewitt



Ζ) Εικόνα κατακόρυφων Ακμών Prewitt



Η) Εικόνα Ακμών Prewitt



Θ) Εικόνα οριζόντιων Ακμών Roberts

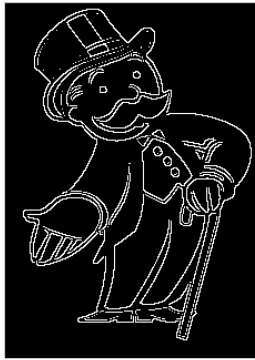


Ι) Εικόνα κατακόρυφων Ακμών Roberts



Κ) Εικόνα Ακμών Roberts

Σχήμα 2.3: Εικόνες ανιχνευτών Ακμής



Α) Εικόνα Ακμών
Roberts με κατώφλι
 $T=0.1$



Β) Εικόνα Ακμών
Prewitt με κατώφλι
 $T=0.1$



Γ) Εικόνα Ακμών
Sobel με κατώφλι
 $T=0.1$



Δ) Εικόνα Ακμών
Canny με κατώφλι
 $T=0.1$ και $\sigma=1$



Ε) Εικόνα Ακμών
Canny με κατώφλι
 $T=0.1$ και $\sigma=1.5$



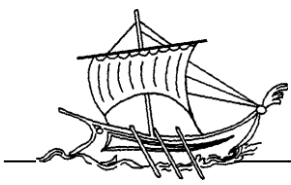
Ζ) Εικόνα Ακμών
Canny με κατώφλι
 $T=0.1$ και $\sigma=2$

Σχήμα 2.3: Εικόνες ανιχνευτών ακμής με μάσκες παραγωγίσης Sobel, Prewitt, Roberts

2.3 Κώδικας MatLab Παραδείγματος Αλγόριθμου Canny

Παρακάτω φαίνεται ο κώδικας γραμμένος στο MATLAB, ο οποίος δημιουργεί τις εικόνες στα σχήματα 2.2 και 2.3.

```
[cdata, colormap]=imread('man','gif');  
man = ind2gray(cdata, colormap);  
man = im2double(man);  
Sv = [-1 0 1; -2 0 2; -1 0 1];  
Sh = [-1 -2 -1; 0 0 0; 1 2 1];
```



```
Pv = [-1 0 1; -1 0 1; -1 0 1];
Ph = [-1 -1 -1; 0 0 0; 1 1 1];
Rv = [0 0 -1; 0 1 0; 0 0 0];
Rh = [-1 0 0; 0 1 0; 0 0 0];
Sv=im2double(Sv);
Sh=im2double(Sh);
Pv=im2double(Pv);
Ph=im2double(Ph);
Rv=im2double(Rv);
Rh=im2double(Rh);
Isv = conv2(man,Sv);
Ish = conv2(man,Sh);
Ipv = conv2(man,Pv);
Iph = conv2(man,Ph);
Irv = conv2(man,Rv);
Irh = conv2(man,Rh);
Is = sqrt( Isv.^2 + Ish.^2 );
Ip = sqrt( Ipv.^2 + Iph.^2 );
Ir = sqrt( Irv.^2 + Irh.^2 );
figure, imshow(Isv);
figure, imshow(Ish);
figure, imshow(Is);
figure, imshow(Ipv);
figure, imshow(Iph);
figure, imshow(Ip);
figure, imshow(Irv);
figure, imshow(Irh);
figure, imshow(Ir);
BW_sobel = edge(man,'sobel', 0.1);
BW_prewitt = edge(man,'prewitt', 0.1);
BW_roberts = edge(man,'roberts', 0.1);
BW_canny_1 = edge(man,'canny', 0.1, 1);
```



```
BW_canny_2 = edge(man, 'canny', 0.1, 1.5);  
BW_canny_3 = edge(man, 'canny', 0.1, 2);  
  
figure, imshow(BW_sobel);  
figure, imshow(BW_prewitt);  
figure, imshow(BW_roberts);  
figure, imshow(BW_canny_1);  
figure, imshow(BW_canny_2);  
figure, imshow(BW_canny_3);
```



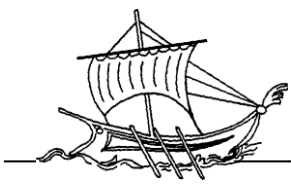
3. ΧΡΩΜΑΤΙΚΟΙ ΧΩΡΟΙ

Τα χρώματα, ουσιαστικά, είναι η κωδικοποίηση που υπάρχει στο νευρικό σύστημα του ανθρώπου, διακρίνοντας μήκη κύματος του φωτός, τα οποία δέχονται τα ανθρώπινα μάτια. Το ανθρώπινο μάτι διεγείρεται από μήκη κύματος που κυμαίνονται από 400nm έως 700nm περίπου. Στον πίνακα 3.1 φαίνεται η χρωματική κωδικοποίηση αυτή. Το ανθρώπινο μάτι αντιστοιχεί σε κάθε μήκος κύματος και ένα χρώμα.

Πίνακας 3.1: Χρώματα ορατού φάσματος

Χρώμα	Περιοχή μηκών κύματος (nm)	Περιοχή συχνοτήτων (Hz)
Ερυθρό	~ 630–700 nm	~ 476–429 $\times 10^{12}$ Hz
Πορτοκαλί	~ 590–630 nm	~ 510–476 $\times 10^{12}$ Hz
Κίτρινο	~ 560–590 nm	~ 535–510 $\times 10^{12}$ Hz
Πράσινο	~ 500–560 nm	~ 600–535 $\times 10^{12}$ Hz
Κυανό	~ 440–500 nm	~ 680–600 $\times 10^{12}$ Hz
Ιώδες	~ 400–440 nm	~ 750–680 $\times 10^{12}$ Hz

Όταν στον ανθρώπινο οφθαλμό προσπίπτουν δύο ακτινοβολίες με διαφορετικό μήκος κύματος, τότε τα χρώματα συνθέτονται δημιουργώντας καινούργια. Αν για παράδειγμα, μια φωτεινή πηγή φαίνεται πως εκπέμπει κίτρινο χρώμα, τότε πιθανόν να έχει μήκος κύματος από 560nm έως 590nm ή εκπέμπει πράσινες και κόκκινες ακτινοβολίες ταυτόχρονα, οι οποίες όταν συνδυάζονται δημιουργούν κίτρινο χρώμα. Συνεπώς, τα υπόλοιπα χρώματα μπορούν να συνθεθούν από αυτά που ονομάζουμε βασικά χρώματα. Τα βασικά χρώματα, που δημιουργούν τη σύνθεση χρωμάτων διαφέρουν από εφαρμογή σε εφαρμογή. Οι διαφοροποιήσεις προκύπτουν από τον τρόπο με τον οποίο φτάνει το φως στο μάτι. Το φως μπορεί να προέρχεται είτε από απευθείας εκπομπή, είτε από ανάκλαση, είτε να διέρχεται από ημιδιαφανή χρωματιστά υλικά. Κατά την εκπομπή τα μήκη κύματος αθροίζονται και δημιουργούν ένα χρωματικό αποτέλεσμα. Ενώ, στη διεργασία της απορρόφησης του φωτός, γίνεται αφαίρεση των υλικών για τη δημιουργία του αποτελέσματος του χρώματος. Επιπλέον, το ανθρώπινο μάτι αντιλαμβάνεται και άλλα χαρακτηριστικά όπως τη λαμπρότητα, τη χρωματική καθαρότητα, κ.α. Γίνεται αντιληπτό, πως η αίσθηση του χρώματος αποτελεί μια



πολύπλοκη διαδικασία. Τα χρωματικά μοντέλα βοηθούν στην περιγραφή και στην αναπαραγωγή των χρωμάτων. Κάποια συγκεκριμένα βασικά χρώματα είναι η βάση των χρωματικών μοντέλων. Σπάνια συμπίπτει το ίδιο χρωματικό αποτέλεσμα από δύο εφαρμογές, ακόμα και αν εφαρμόζεται ίδιο, ακριβώς, χρωματικό μοντέλο.

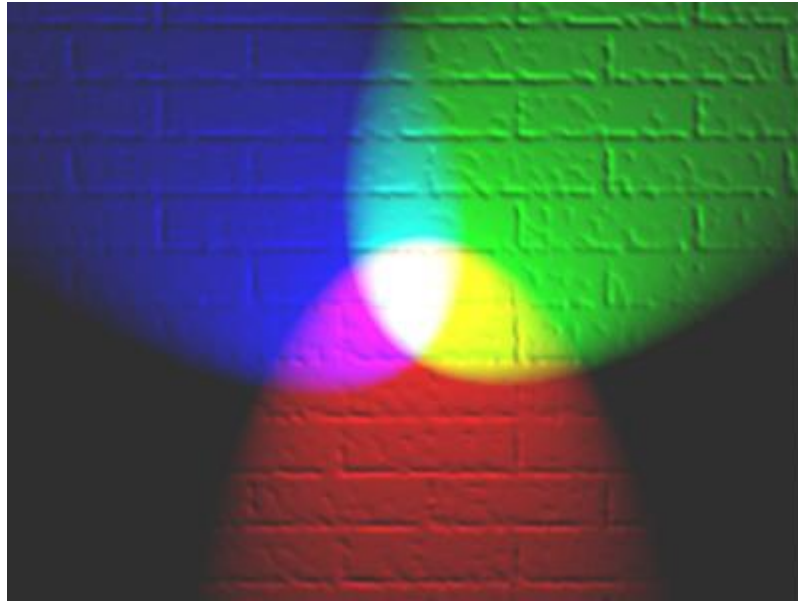
Κάθε χρωματικό μοντέλο αποτελεί μαθηματικό μοντέλο περιγράφοντας τον τρόπο που τα χρώματα απεικονίζονται ως σημεία σε ένα σύστημα συντεταγμένων. Τα πιο γνωστά χρωματικά μοντέλα είναι το RGB, το HIS και το HSV. Στην επεξεργασία μίας εικόνας η χρήση του χρώματος λαμβάνει χώρα για δύο κυρίως λόγους. Αρχικά, το χρώμα είναι ισχυρό χαρακτηριστικό για την περιγραφή και συνήθως απλοποιεί την αναγνώριση αντικειμένων. Και δεύτερον, επειδή ο ανθρώπινος οφθαλμός είναι ικανός να διακρίνει χιλιάδες έγχρωμες αποχρώσεις, έναντι των 24 αποχρώσεων περίπου του γκρι. Ο διαχωρισμός μεταξύ των χρωμάτων γίνεται με βάση τα παρακάτω χαρακτηριστικά:

- Την λαμπρότητα
- Την απόχρωση
- Τη χρωματική καθαρότητα

3.1 Μοντέλο RGB

Σε ένα χρωματικό μοντέλο RGB, κάθε χρώμα αποτελεί συνάρτηση των χρωμάτων κόκκινο, μπλε και πράσινο, βασιζόμενο σε ένα καρτεσιανό σύστημα συντεταγμένων. Ο ανθρώπινος οφθαλμός έχει τρεις δέκτες, που διεγείρονται από ερεθίσματα που δέχονται, και κάθε δέκτης έχει ευαισθησία σε συγκεκριμένη περιοχή μήκους κύματος.

Η δημιουργία ενός χρώματος στο μοντέλο RGB πραγματοποιείται αν τρεις μονοχρωματικές ακτίνες φωτός των βασικών χρωμάτων υπερτεθούν. Κάθε μια από αυτές, μπορεί να έχει διαφορετική ένταση από το 0 έως το 1. Το άθροισμα των μηκών κύματος αποτελεί το μήκος κύματος του τελικού χρώματος. Η ανάλυση σε συνιστώσες μιας RGB έγχρωμης εικόνας αναπαριστάται στο σχήμα 3.2.



Σχήμα 3.2: Αθροιστική μίξη χρωμάτων

3.2 Παράδειγμα Χρωματικού Μοντέλου RGB στο MATLAB

Στο σχήμα 3.3 διακρίνονται οι χρωματικές συνιστώσες ενός χρωματικού μοντέλου RGB



Α) Αρχική εικόνα



Β) Η κόκκινη χρωματική συνιστώσα



Γ) Η πράσινη χρωματική συνιστώσα



Δ) Η μπλε χρωματική συνιστώσα

Σχήμα 3.3: Χρωματικές συνιστώσες μοντέλου RGB



Ο κώδικας που χρησιμοποιήθηκε για την υλοποίηση.

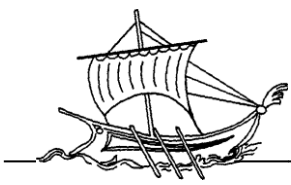
```
flowers=imread('flowers','jpg');  
  
red = flowers(:,:,1);  
green = flowers(:,:,2);  
blue = flowers(:,:,3);  
figure, imshow(flowers);  
figure, imshow(red);  
figure, imshow(green);  
figure, imshow(blue);
```

3.3 Μοντέλα HIS και HSV

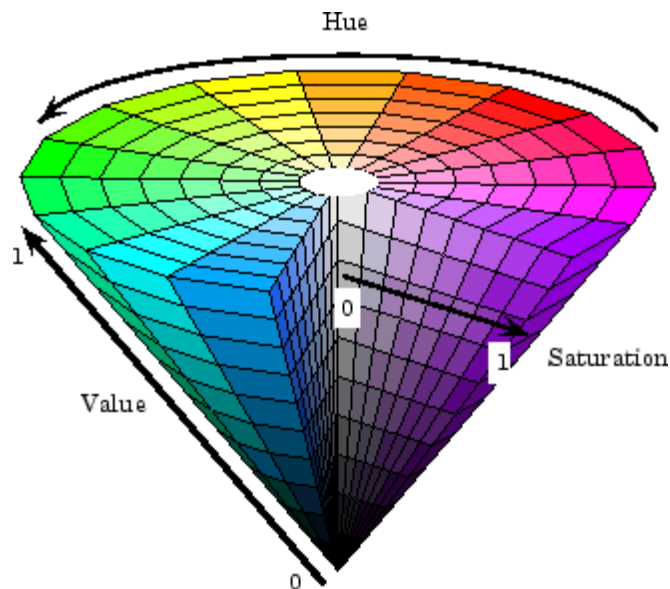
Τα μοντέλα αυτά έχουν σχεδιαστεί με σκοπό την προσέγγιση της ανθρώπινης αντίληψης. Οι παράμετροι που χρησιμοποιούν είναι:

- Η συνιστώσα απόχρωσης (Hue), η οποία κυμαίνεται από 0° έως 360° .
- Η συνιστώσα χρωματικής καθαρότητας (Saturation), η οποία αποτελεί το βαθμό μίξης ενός βασικού χρώματος με το άσπρο.
- Η συνιστώσα τιμής (Value), η οποία αναφέρεται στο βαθμό μίξης με το μαύρο.
- Η συνιστώσα της έντασης φωτός (Intensity), η οποία εκφράζει την ένταση και δεν παρέχει καμία χρωματική πληροφορία.

Τα δύο μοντέλα διαφέρουν μεταξύ τους ως προς τον υπολογισμό της συνιστώσας της λαμπρότητας (brightness). Για κλασικές συναρτήσεις ψηφιακής εργασίας, όπως η εξισορρόπηση ιστογράμματος χρησιμοποιείται συνήθως το μοντέλο HIS. Ενώ, το μοντέλο HSV χρησιμοποιείται κυρίως για τη διαχείριση των συνιστωσών της απόχρωσης και της χρωματικής καθαρότητας.

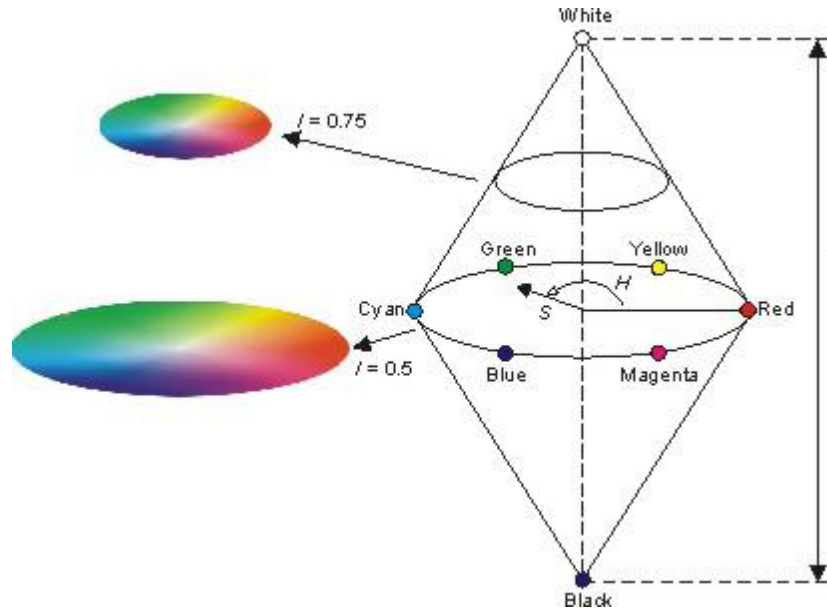


Στο παρακάτω σχήμα (Σχήμα 3.4) απεικονίζεται ο χρωματικός χώρος ενός μοντέλου HSV. Στην κορυφή του κώνου το χρώμα είναι μαύρο, αφού η συνιστώσα τιμής V είναι μηδέν. Ενώ στη βάση που η συνιστώσα τιμή είναι 1 το χρώμα είναι άσπρο. Το κόκκινο χρώμα έχει 0° , ενώ τα συμπληρωματικά διαφέρουν κατά 180° σε σχέση με τη συνιστώσα απόχρωσης H. Η συνιστώσα χρωματικής καθαρότητας S καθορίζεται από την απόσταση του κέντρου της βάσης του κώνου. Τα σημεία που βρίσκονται στον κάθετο άξονα της συνιστώσας τιμής V αποτελούν διαβαθμίσεις του γκρι. Αν στο εξωτερικό σημείο της βάσης προστεθεί μαύρο χρώμα μειώνεται η συνιστώσα τιμής V και διατηρείται σταθερή η συνιστώσα χρωματικής καθαρότητας S. Ενώ, αν προστεθεί άσπρο μειώνεται η συνιστώσα χρωματικής καθαρότητας S.



Σχήμα 3.4: Χρωματικός χώρος μοντέλου HSV

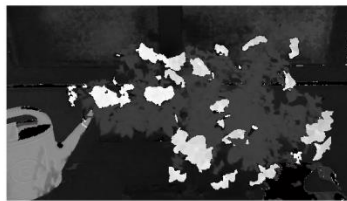
Στο παρακάτω σχήμα (σχήμα 3.5) απεικονίζεται ο διπλός κώνος του μοντέλου HIS. Η άνω κορυφή αντιστοιχεί σε $I=1$, δηλαδή άσπρο χρώμα. Ενώ, η κάτω κορυφή αντιστοιχεί σε μαύρο χρώμα, δηλαδή $I=0$. Η μέγιστη καθαρότητα $\Sigma=1$ εμφανίζεται στο σημείο όπου $I=0.5$.



Σχήμα 3.5: Χρωματικός χώρος μοντέλου HIS

3.4 Παράδειγμα Μοντέλων HSI και HSV στο MATLAB

Στις παρακάτω εικόνες (σχήμα 3.6) παρουσιάζεται η ανάλυση μια έγχρωμης εικόνας HSV στις συνιστώσες της.



Α) Η χρωματική συνιστώσα Hue

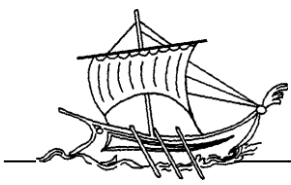


Β) Η χρωματική συνιστώσα Saturation



Γ) Η χρωματική συνιστώσα Value

Σχήμα 3.6: Χρωματικές Συνιστώσες μοντέλου HSV



Η υλοποίηση πραγματοποιήθηκε με τον παρακάτω κώδικα σε γλώσσα MATLAB.

```
flowers=imread('flowers','jpg');  
hsv = rgb2hsv(flowers);  
hue = hsv(:,:,1);  
sat = hsv(:,:,2);  
val = hsv(:,:,3);  
figure, imshow(hue)  
figure, imshow(sat)  
figure, imshow(val)
```

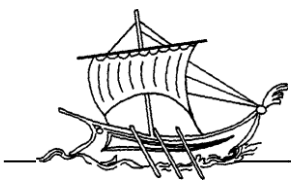
3.5 Ο Χρωματικός χώρος YCbCr

Η δημιουργία του χρωματικού χώρου YCbCr πρόσθεσε την πληροφορία του χρώματος στο ήδη υπάρχον ασπρόμαυρο τηλεοπτικό σήμα, δηλαδή καλύπτει τις ανάγκες της ψηφιακής επεξεργασίας σήματος. Χρησιμεύει σε εφαρμογές συμπίεσης, όμως τα χρώματα δεν απεικονίζονται σύμφωνα με τον τρόπο αντίληψης των ανθρώπων. Η συνιστώσα Y υποδηλώνει την φωτεινότητα και οι Cb, Cr την χρωματική διαφορά ως προς τα μπλε και κόκκινο, αντίστοιχα [3].

Οι λόγοι που χρησιμοποιείται αυτός ο μετασχηματισμός είναι:

- α) Η όραση του ανθρώπου είναι πιο ευαίσθητη στη φωτεινότητα παρά στο χρώμα
- β) Η ταχύτητα μετάδοσης της εικόνας βελτιστοποιείται μετασχηματίζοντας το σήμα RGB σε σήμα φωτεινότητας και 2 σήματα μεταφοράς χρωματικής πληροφορίας. Τα 2 σήματα χρώματος χρειάζονται μικρότερη ακρίβεια για να αναπαρασταθούν και να μεταδοθούν από την ακρίβεια ενός σήματος φωτεινότητας.

Η σχέση μετατροπής του RGB σε YCbCr είναι:



$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Με την προϋπόθεση ότι οι τιμές του RGB βρίσκονται στο διάστημα [0,1], η τιμή της μεταβλητής Y βρίσκεται στο διάστημα [16,235], και οι Cb, Cr στο διάστημα [16,240].

Η σχέση του αντίστροφου μετασχηματισμού με R,G,B ∈ [0,1] είναι:

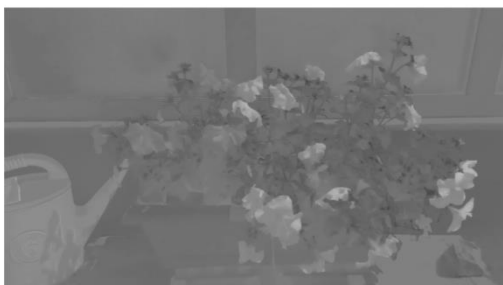
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 0.00456621 & 0 & 0.00625893 \\ 0.00456621 & -0.00153632 & -0.00318811 \\ 0.00456621 & 0.00791071 & 0 \end{bmatrix} \left(\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} - \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \right)$$

3.6 Παράδειγμα μοντέλου YCbCr στο MATLAB

Στο σχήμα 3.7 διακρίνονται οι χρωματικές συνιστώσες ενός χρωματικού μοντέλου YCbCr.



Α) Η χρωματική συνιστώσα Y



Β) Η χρωματική συνιστώσα Cb



Γ) Η χρωματική συνιστώσα Cr

Σχήμα 3.7: Χρωματικές Συνιστώσες μοντέλου YCbCr

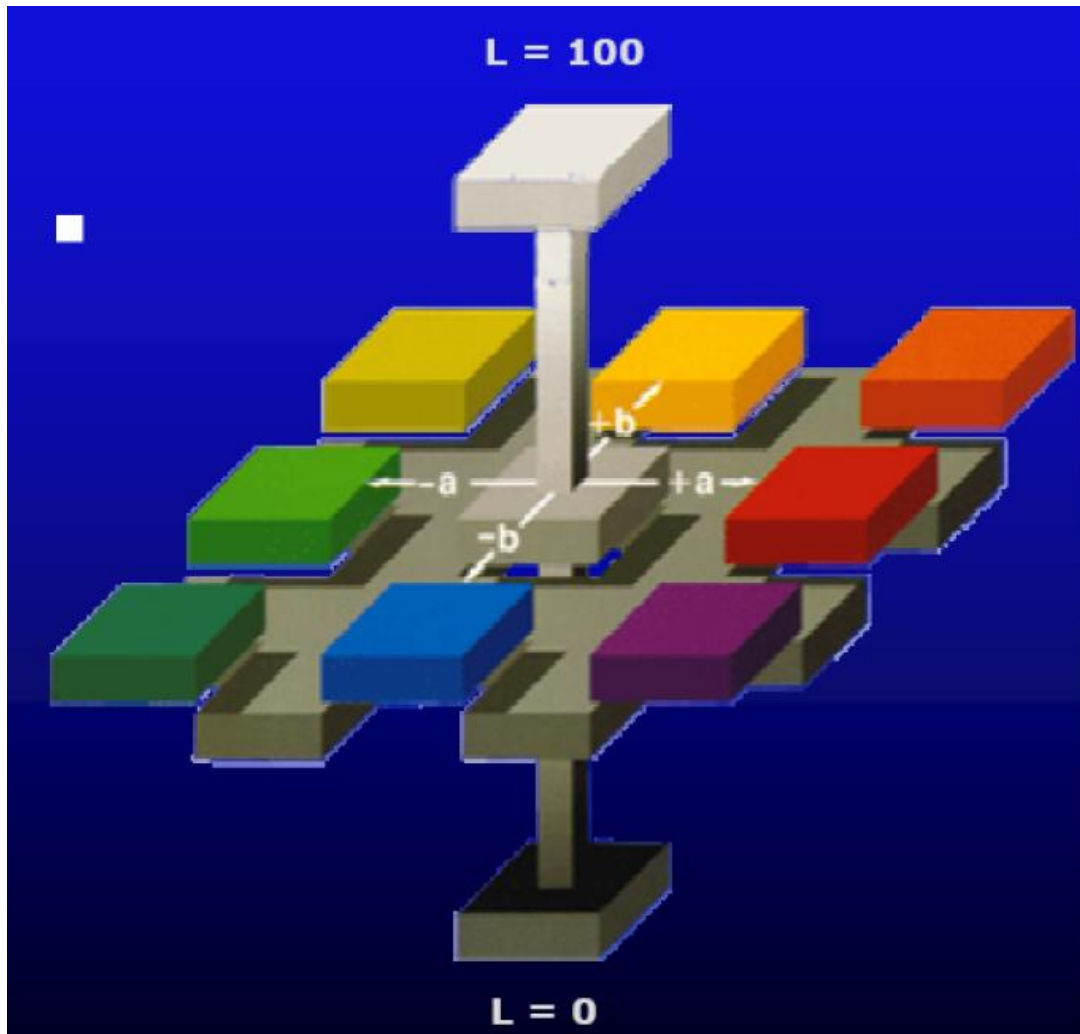
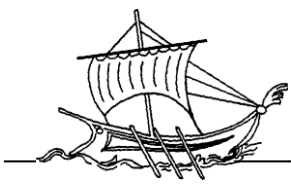


Η υλοποίηση πραγματοποιήθηκε με τον παρακάτω κώδικα σε γλώσσα MATLAB.

```
flowers =imread('flowers','jpg');  
ycbcr = rgb2ycbcr(flowers);  
y = ycbcr(:, :, 1);  
cb = ycbcr(:, :, 2);  
cr = ycbcr(:, :, 3);  
figure, imshow(y)  
figure, imshow(cb)  
figure, imshow(cr)
```

3.7 Μοντέλο CieLab

Η Διεθνής Επιτροπή Φωτισμού CIE παρουσίασε το χρωματικό μοντέλο CieLab (αλλιώς L^*, a^*, b^*) το 1976. Είναι ένας χρωματικός χώρος ο οποίος είναι ομοιόμορφος οπτικά. Όπως και στον RGB έτσι και στο χρωματικό μοντέλο CieLab, τα χρώματα περιγράφονται από 3 παράγοντες, τους L^* , a^* και b^* . Αυτοί, μπορούν να απεικονιστούν σε ένα τρισδιάστατο καρτεσιανό σύστημα συντεταγμένων. Στον παράγοντα L^* (Lightness) αποθηκεύεται όλη η πληροφορία της φωτεινότητας και παίρνει τιμές 0 (μαύρο) έως 100 (άσπρο) και οι a^* και b^* παράγοντες αποθηκεύουν την πληροφορία του χρώματος χωρίς να έχουν κάποιο όριο. Οι αρνητικές τιμές του a^* αντιστοιχούν στις αποχρώσεις του πράσινου ενώ, οι θετικές στις αποχρώσεις του κόκκινου. Οι αρνητικές τιμές του b^* αντιστοιχούν στις αποχρώσεις του μπλε, ενώ οι θετικές στις αποχρώσεις του κίτρινου.



Σχήμα 3.7 Χρωματικό Μοντέλο L^* , a^* , b^*

Το πλέον σημαντικό χαρακτηριστικό, και αυτό που μας δίνει αρκετές δυνατότητες είναι η ανεξαρτησία που υπάρχει μεταξύ φωτεινότητας και χρώματος. Δηλαδή, η συντεταγμένη L^* περιγράφει την ασπρόμαυρη εκδοχή του με το φάσμα του να είναι συνήθως διευρυμένο.

Το CIE Lab ουσιαστικά είναι υπερσύνολο όλων των χρωματικών μοντέλων RGB, YCbCr, κ.α. Φτάνοντας στα όρια του, διαμορφώνονται χρώματα τόσο πρωτόγνωρα και εξωπραγματικά που πολλές συσκευές απεικόνισης δεν μπορούν να τα απεικονίσουν. Πολύ σημαντικό να αναφερθεί είναι ότι το CIE Lab μοντέλο είναι ανεξάρτητο από τις ιδιότητες των συσκευών καταγραφής και απεικόνισης.



Το χρωματικό μοντέλο CieLab μπορεί και μέσω του μοντέλου CieL*,C*,h να αναπαρασταθεί σε κυλινδρικό σύστημα πολικών συντεταγμένων. Όπου:

- L*, η φωτεινότητα
- C*(Chroma), η χρωματική πυκνότητα. Είναι η σχέση μεταξύ έντασης και φωτεινότητας της απόχρωσης που μελετάμε
- H, η χροιά. Προσδιορίζει την απόχρωση και μετριέται σε μοίρες.

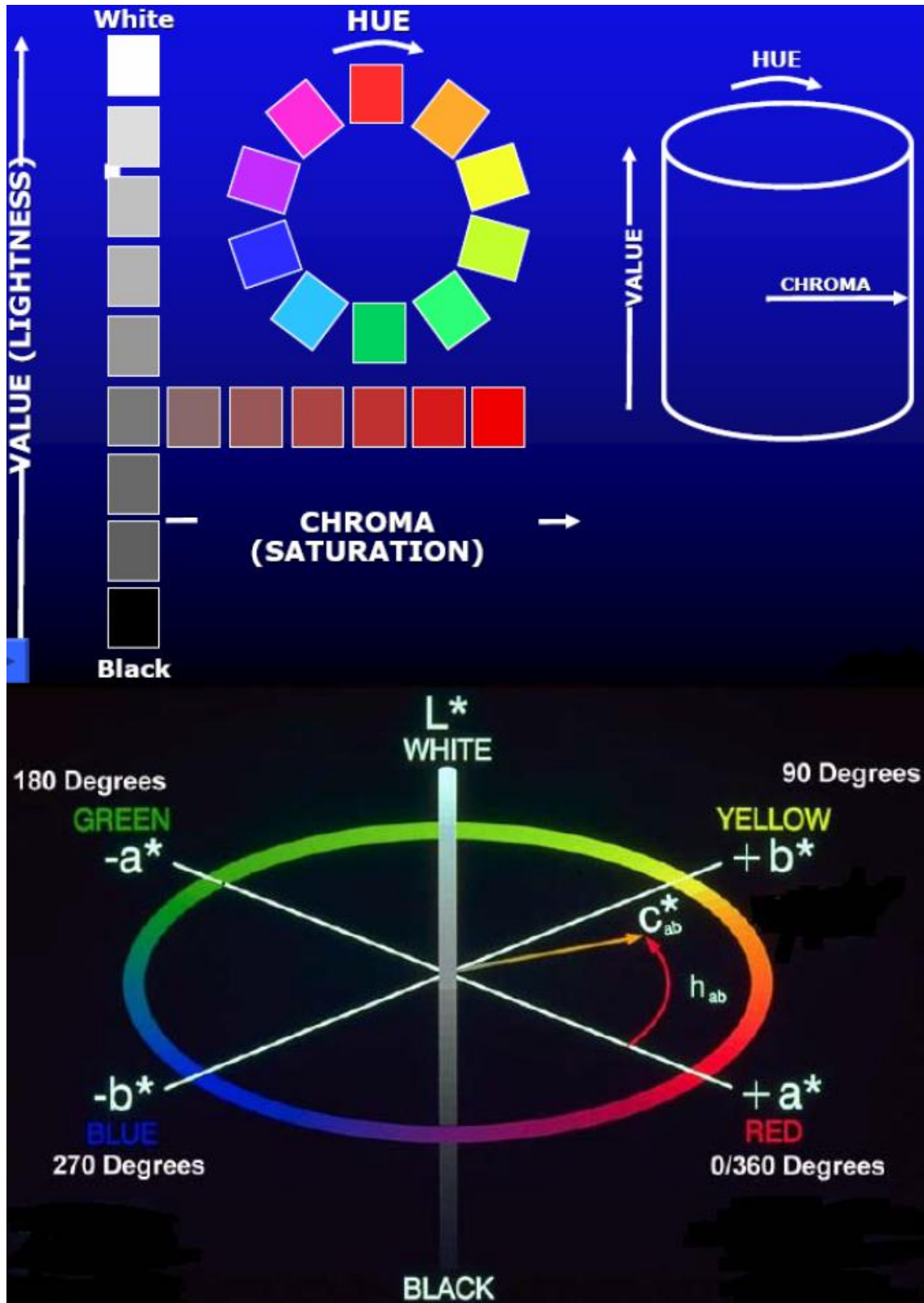
Η χρωματική πυκνότητα C* υπολογίζεται από την μαθηματική εξίσωση:

$$C^* = \sqrt{(a^*)^2 + (b^*)^2}$$

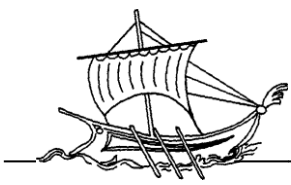
Ο υπολογισμός του παράγοντα h φαίνεται παρακάτω:

- $h = \tan^{-1}\left(\frac{b^*}{a^*}\right)$ όταν $a^* > 0$ και $b^* \geq 0$
- $h = 0^\circ$ όταν $a^* = 0$ και $b^* = 0$
- $h = 90^\circ$ όταν $a^* = 0$ και $b^* > 0$
- $h = 180^\circ + \tan^{-1}\left(\frac{b^*}{a^*}\right)$ όταν $a^* < 0$
- $h = 270^\circ$ όταν $a^* = 0$ και $b^* < 0$

$$h = 360^\circ + \tan^{-1}\left(\frac{b^*}{a^*}\right) \text{ όταν } a^* > 0 \text{ και } b^* < 0$$



Σχήμα 3.8 Απεικόνιση του χρωματικού μοντέλου CieLab σε κυλινδρικό σύστημα συντεταγμένων L^* , C^* και h



3.8 Παράδειγμα μοντέλου CieLab στο MATLAB

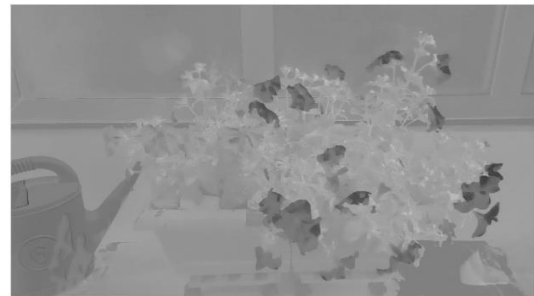
Στο σχήμα 3.9 διακρίνονται οι χρωματικές συνιστώσες ενός χρωματικού μοντέλου YCbCr



A) Η χρωματική συνιστώσα L^*



B) Η χρωματική συνιστώσα a^*



Γ) Η χρωματική συνιστώσα b^*

Σχήμα 3.9 Χρωματικές συνιστώσες του CieLab

Η υλοποίηση πραγματοποιήθηκε με τον παρακάτω κώδικα σε γλώσσα MATLAB.

```
flowers=imread('flowers','jpg');  
colorTransform = makecform('srgb2lab');  
lab = applycform(flowers, colorTransform);  
L = lab(:, :, 1);  
A = lab(:, :, 2);  
B = lab(:, :, 3);  
figure,imshow(L);  
figure,imshow(A);  
figure,imshow(B);
```



4. ΜΑΘΗΜΑΤΙΚΗ ΜΟΡΦΟΛΟΓΙΑ

Η μαθηματική μορφολογία είναι ο τομέας που έχει προσφέρει το μέρος των τελεστών στην επεξεργασία εικόνας, οι οποίοι είναι σύμφωνοι με μερικές βασικές αρχές της θεωρίας των συνόλων. Οι τελεστές χρησιμεύουν στην ανάλυση εικόνων στο δυαδικό σύστημα. Οι πιο συνηθισμένες εφαρμογές τους είναι η ανίχνευση ακμών, η απομάκρυνση θορύβου, η ενίσχυση και τμηματοποίηση μιας εικόνας

Οι ιδιότητες των μορφολογικών μετασχηματισμών είναι οι παρακάτω:

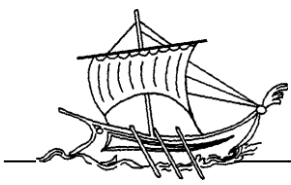
- Σταθερότητα στη μετατόπιση
- Σταθερότητα στην κλιμάκωση
- Τοπική γνώση (παίρνουν πληροφορίες από την τοπική γειτονιά τους)
- Ημισυνέχεια, ικανοποιούνται μόνο συγκεκριμένες ιδιότητες της συνέχειας

Οι βασικότεροι τελεστές της μαθηματικής μορφολογίας είναι η διάβρωση ή συστολή (erosion) και η διαστολή (dilation). Οι τελεστές αυτοί παίρνουν σαν είσοδο δύο ορίσματα:

- Μια εικόνα, στην οποία θα εφαρμοστεί συστολή ή διαστολή
- Ένας πυρήνας ή μια μήτρα, τα οποία είναι δομικά στοιχεία

Τα δομικά στοιχεία, γενικώς, μπορεί να έχουν διάφορα σχήματα και μεγέθη. Όμως πρέπει πάντα να εμπεριέχουν ένα σημείο αναφοράς. Ένα δομικό στοιχείο, τις περισσότερες φορές, είναι ένα ενιαίο τετράγωνο με διαστάσεις 3×3 και το σημείο αναφοράς του είναι στο κέντρο του.

Με βάση τη συστολή και τη διαστολή και συνδυάζοντας διάφορους τελεστές από τη θεωρία των συνόλων (ένωση, τομή κ.α.) μπορούν να οριστούν όλοι οι μορφολογικοί τελεστές. Κάποιοι σημαντικοί τελεστές είναι το Κλείσιμο και το Άνοιγμα.

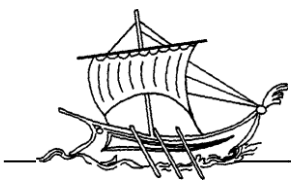


4.1 Μορφολογικοί Τελεστές Συστολής και Διαστολής

Η έννοια της διαστολής είναι η συνέλιξη της επιλεγμένης εικόνας ή ενός μέρους της με ένα δομικό στοιχείο. Θεωρούμε δομικό στοιχείο μια μάσκα, και το αποτέλεσμα ενός τελεστή εύρεσης τοπικών μεγίστων είναι το αποτέλεσμα της Διαστολής. Κατά τη διάρκεια της σάρωσης της εικόνας από τον πυρήνα, υπολογίζεται η μέγιστη τιμή των επικαλυπτόμενων από τον πυρήνα pixel. Έπειτα, αντικαθίσταται η υπολογισμένη μέγιστη τιμή στο pixel το οποίο βρίσκεται στον πυρήνα, κάτω από το σημείο αναφοράς. Το αποτέλεσμα της διαδικασίας αυτής είναι η διόγκωση των φωτεινών περιοχών. Ο τελεστής Διαστολής είναι ουσιαστικά αυτός που προκαλεί αυτή την διόγκωση.

Η ακριβώς αντίθετη διαδικασία είναι η Συστολή. Το αποτέλεσμα της εφαρμογής της Συστολής είναι αντίστοιχο με το αποτέλεσμα ενός τελεστή εύρεσης τοπικών ελαχίστων. Η διαδικασία παραγωγής μιας νέας εικόνας από τη Συστολή είναι παρόμοια με αυτή της Διαστολής. Δηλαδή, κατά την διάρκεια της σάρωσης της εικόνας από τον πυρήνα, υπολογίζεται η ελάχιστη τιμή των επικαλυπτόμενων από τον πυρήνα pixel. Στη συνέχεια αντικαθίσταται η υπολογισμένη ελάχιστη τιμή στο pixel, το οποίο βρίσκεται στον πυρήνα κάτω από το σημείο αναφοράς.

Γενικώς, η Συστολή έχει την τάση να σμικρύνει την περιοχή όπου εφαρμόζεται και η Διαστολή έχει την τάση να την μεγενθύνει. Ακόμα, η Συστολή εξαφανίζει τις προεξοχές της εικόνας, ενώ η Διαστολή εξομαλύνει τις κοιλότητες της εικόνας. Η ακρίβεια των αποτελεσμάτων είναι πάντα εξαρτώμενη από τον πυρήνα που έχει επιλεγθεί, αλλά σε γενικό πλαίσιο οι παραπάνω ιδιότητες ισχύουν στα συχνότερα χρησιμοποιημένα δομικά στοιχεία.



α) Αρχική εικόνα



β) Εικόνα μετά από 2 εφαρμογές Διαστολής



γ) Εικόνα μετά από 2 εφαρμογές Συστολής

Σχήμα 4.1: Αποτελέσματα Διαστολής και Συστολής

Η ιδέα πίσω από την Συστολή είναι να απομακρυνθεί ο θόρυβος. Ο θόρυβος είναι συνήθως μικρά στίγματα σε μια εικόνα. Μέσω της Συστολής συστέλλονται τα στίγματα αυτά, και σταματάνε να υπάρχουν έτσι οι μεγαλύτερες περιοχές παραμένουν ανέπαφες. Χρησιμοποιούμε τον τελεστή Συστολής για τον εντοπισμό συνδεδεμένων περιοχών, δηλαδή περιοχών της εικόνας οι οποίες είναι μεγάλες, διακριτές και έχουν την ίδια φωτεινότητα ή το ίδιο χρώμα. Η Διαστολή χρησιμοποιείται σε μεγάλες περιοχές οι οποίες είναι κατατετημημένες λόγω του θορύβου, των σκιάσεων και άλλων αιτιών. Το αποτέλεσμα της εφαρμογής του τελεστή είναι η ελαφρά διόγκωση των τμημάτων και σύνδεση τους, έτσι ώστε να σχηματίσουν μια μεγάλη περιοχή. Τα αποτελέσματα της χρήσης Συστολής και Διαστολής φαίνονται παρακάτω.



4.2 Παραδείγματα Συστολής και Διαστολής



α) Αρχική εικόνα



β) Η εικόνα μετά από Διαστολή



γ) Η εικόνα μετά από 2 Διαστολές

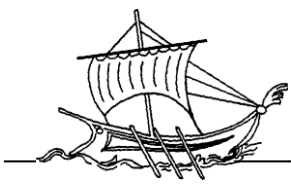


δ) Η εικόνα μετά από Συστολή

Σχήμα 4.2: Αποτελέσματα Διαστολής και Συστολής

Ο κώδικας που χρησιμοποιήθηκε για την υλοποίηση είναι:

```
noir= imread('noir','jpg');  
se=strel('square',3);  
er=imerode(noir,se);  
er=imerode(er,se);  
di=imdilate(noir,se);  
di=imdilate(di,se);  
figure,imshow(noir);  
figure,imshow(er);  
figure, imshow(di);  
  
renoir=imread('renoir','jpg');  
se=strel('square',3);  
renoir_er=imerode(renoir,se);
```



```
renoir_dil=imdilate(renoir,se);  
renoir_dil2=imdilate(renoir_dil,se);  
figure, imshow(renoir);  
figure, imshow(renoir_er);  
figure, imshow(renoir_dil);  
figure, imshow(renoir_dil2);
```

4.3 Μορφολογικοί Τελεστές Άνοιγμα και Κλείσιμο

Όπως αναφέρθηκε και στην αρχή του κεφαλαίου, συνδυάζοντας τους τελεστές Συστολής και Διαστολής ορίζονται οι μορφολογικοί τελεστές Άνοιγμα και Κλείσιμο. Στο Άνοιγμα, πρώτα εφαρμόζεται η Συστολή της εικόνας και έπειτα η Διαστολή της. Η χρησιμότητα του Ανοίγματος είναι η απαρίθμηση κάποιων περιοχών σε μια δυαδική εικόνα. Στο κλείσιμο, πρώτα εφαρμόζεται η Διαστολή και μετά η Συστολή της εικόνας. Ο τελεστής Κλείσιμο χρησιμοποιείται σε αλγόριθμους τμηματοποίησης εικόνας και σε αλγόριθμους εξαγωγής αντικειμένων έτσι ώστε να μειωθούν τα τμήματα που δημιουργούνται λόγω του θορύβου.

Η συχνότερη χρήση των παραπάνω τελεστών γίνεται σε σειρά. Πρώτα για τη μείωση ή εξάλειψη του θορύβου χρησιμοποιείται το Κλείσιμο, και έπειτα για να ενωθούν οι μεγάλες γειτονικές περιοχές χρησιμοποιείται το Άνοιγμα. Η ουσιαστική διαφορά των τελεστών αυτών από του τελεστές Συστολή και Διαστολή είναι ότι στους τελεστές Άνοιγμα και Κλείσιμο υπάρχει η τάση να διατηρείται με μεγαλύτερη ακρίβεια η αρχική επιφάνεια των εφαρμοζόμενων περιοχών.



4.4 Παραδείγματα Ανοίγματος και κλεισίματος



α) Αρχική εικόνα



β) Η εικόνα μετά από Άνοιγμα



γ) Η εικόνα μετά από Κλείσιμο



δ) Η εικόνα μετά από Άνοιγμα, Κλείσιμο, 2 Διαστολές και 2 Συστολές

Σχήμα 4.3 Αποτελέσματα εφαρμογής Ανοίγματος, Κλεισίματος, Διαστολής και Συστολής

Ο κώδικας που χρησιμοποιήθηκε για την υλοποίηση είναι:

```
renoir_open = imopen(renoir, se);  
renoir_close = imclose(renoir, se);  
renoir_connected = imclose(renoir_open, se);  
renoir_connected = imdilate(renoir_connected, se);  
renoir_connected = imdilate(renoir_connected, se);  
renoir_connected = imerode(renoir_connected, se);  
renoir_connected = imerode(renoir_connected, se);  
figure, imshow(renoir_open);  
figure, imshow(renoir_close);  
figure, imshow(renoir_connected);
```




4.5 Κύριες Μορφολογικές Συναρτήσεις του MatLab

4.5.1 Ο μετασχηματισμός Hit-or-Miss

Πολλές φορές είναι χρήσιμο να μπορούν να αναγνωριστούν οι διάφορες τοπολογίες των pixel, όπως για παράδειγμα κάποια απομονωμένα pixel που μπορεί να υπάρχουν στο φόντο. Για αυτό το λόγο εφαρμόζεται ο μετασχηματισμός hit-or-miss. Για να υλοποιηθεί χρησιμοποιείται ένα διπλό δομικό στοιχείο $s=(s_1, s_2)$, και ο ορισμός του είναι:

$$f \otimes s = (f \ominus s_1) \cap (f \ominus s_2)$$

Υλοποιείται στο MatLab με τη χρήση της συνάρτησης bwhitmiss

```
>>X=bwhitmiss(f,s1,s2)
```

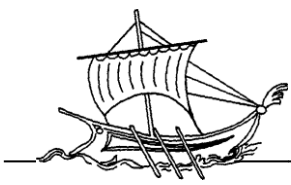
4.5.2 Η συνάρτηση bwmorph

Η συνάρτηση δυαδικών εικόνων bwmorph έχει μια ποικιλία από χρήσιμες εφαρμογές, οι οποίες είναι βασισμένες σε συνδυασμούς συστολών και διαστολών. Υλοποιείται ως εξής:

```
>>X=bwmorph(f,operation,n)
```

Όπου:

- F η δυαδική εικόνα
- Operation η επιθυμητή λειτουργία (πίνακας 4.1)
- N οι φορές που θα εφαρμοστεί η λειτουργία



ΠΙΝΑΚΑΣ 4.1 Εντολές MatLab

Operation	Description
bothat	Εφαρμόζει τη μορφολογική λειτουργία "bottom hat" (διαστολή και μετά συστολή)
branchpoints	$\begin{matrix} 0 & 0 & 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & \rightarrow & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \end{matrix}$
bridge	$\begin{matrix} 1 & 0 & 0 & & 1 & 1 & 0 \\ 1 & 0 & 1 & \rightarrow & 1 & 1 & 1 \\ 0 & 0 & 1 & & 0 & 1 & 1 \end{matrix}$
clean	<p style="text-align: center;">Αφαιρεί 1 που περικυκλώνεται από 0</p> $\begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$
close	Εφαρμόζει το Κλείσιμο
diag	$\begin{matrix} 0 & 1 & 0 & & 0 & 1 & 0 \\ 1 & 0 & 0 & \rightarrow & 1 & 1 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 \end{matrix}$
endpoints	$\begin{matrix} 1 & 0 & 0 & 0 & & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \rightarrow & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \end{matrix}$
fill	<p style="text-align: center;">Αλλάζει το 0 σε 1 που περικυκλώνεται από 1</p> $\begin{matrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{matrix}$
hbreak	$\begin{matrix} 1 & 1 & 1 & & 1 & 1 & 1 \\ 0 & 1 & 0 & \rightarrow & 0 & 0 & 0 \\ 1 & 1 & 1 & & 1 & 1 & 1 \end{matrix}$



majority	Διαμορφώνει ένα pixel ως 1 αν στην 3x3 γειτονιά του υπάρχουν 5 ή παραπάνω 1, αλλιώς τα διαμορφώνει ως 0
open	Εφαρμόζει το Άνοιγμα
remove	Αφαιρεί απομονωμένα pixel
shrink	Μικραίνει τα αντικείμενα που δεν έχουν οπή σε ένα σημείο και μικραίνει τα αντικείμενα που έχουν οπή σε μια δακτύλιο
skel	Αφαιρεί τα pixels στα όρια των αντικειμένων, αλλά δεν επιτρέπει τα αντικείμενα να σπάσουν. Τα pixels που παραμένουν συνθέτουν την εικόνα σκελετό
spur	$\begin{array}{cc} 0 & 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \rightarrow & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & & 1 & 1 & 0 & 0 \end{array}$
thicken	Πυκνώνει τα αντικείμενα χωρίς να τα συνδέσει
thin	Αφαιρεί pixels από ένα αντικείμενο χωρίς οπές έτσι ώστε να γίνει μια γραμμή
tophat	Εφαρμόζει τη μορφολογική λειτουργία "top hat" (συστολή και μετά διαστολή)

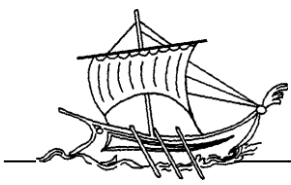
4.5.3 Η συνάρτηση bwlabel

Η συνάρτηση bwlabel βάζει ετικέτες σε κάθε αντικείμενο ξεχωριστά σε μια εικόνα 2-D. Συντάσσεται ως εξής:

```
>>[L,num]=bwlabel(f,conn)
```

Όπου:

- f η δυαδική εικόνα
- conn ο τρόπος σύνδεσης (4 ή 8)
- L ο πίνακας των ετικετών
- Num ο αριθμός των αναγνωρισμένων αντικειμένων



4.5.4 Top-Hat & Bot-Hat

A) ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ TOP-HAT

Αν εφαρμόσουμε Άνοιγμα σε μια εικόνα και έπειτα αυτή αφαιρεθεί από την αρχική της, τότε η διαδικασία αυτή λέγεται μετασχηματισμός top-hat. Η top-hat συντάσσεται ως εξής:

$$\gg F = \text{imtophat}(f_0, se)$$

B) ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ BOT-HAT

Η αντίστοιχη διαδικασία, με τη μόνη διαφορά ο τελεστής να είναι Κλείσιμο αντί Άνοιγμα, είναι ο μετασχηματισμός bot-hat και συντάσσεται ως εξής:

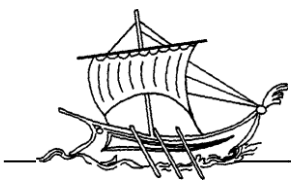
$$\gg F = \text{imbothat}(f_0, se)$$

4.6 Μορφολογική Ανακατασκευή

Μορφολογική ανακατασκευή θεωρείται μια διαδικασία, η οποία αποτελείται από διαδοχικές Διαστολές μιας περιοχής-εικόνας (εικόνα δείκτης) ώσπου να ταιριάζει το περίγραμμά της κάτω από μια άλλη εικόνα (εικόνα μάσκα). Οι κορυφές της εικόνας δείκτη, κατά τη διάρκεια της μορφολογικής ανακατασκευής ή θα διαστέλλονται ή θα γίνονται επίπεδες.

Η διαδικασία ανακατασκευής είναι η ακόλουθη:

- i. Αρχικοποίηση της h_1 ως εικόνα-δείκτης f
- ii. Δημιουργία ενός δομικού στοιχείου: $s = \text{ones}(3)$
- iii. Επανάληψη του παρακάτω:



$$h_{k+1} = (h_k \oplus s) \cap g$$

Μέχρι $h_{k+1}=h_k$

Πρέπει η f να είναι υποσύνολο της g : $f \subseteq g$

Η ανακατασκευή υλοποιείται στο MatLab με τη συνάρτηση:

```
>>out=imreconstruct(marker,mask)
```

4.6.1 Άνοιγμα με Ανακατασκευή

Στην διαδικασία του μορφολογικού ανοίγματος, η συστολή απαλείφει μικρά αντικείμενα και η ακολουθούμενη διαστολή επαναφέρει το σχήμα των εναπομεινάντων αντικειμένων. Η ομοιότητα μεταξύ των σχημάτων και του πυρήνα είναι ο σημαντικότερος παράγοντας για την ακριβή επαναφορά των αντικειμένων. Για να γίνει η ακριβής επαναφορά των αντικειμένων που απομένουν μετά την συστολή χρησιμοποιείται η μέθοδος άνοιγμα με ανακατασκευή. Το Άνοιγμα με ανακατασκευή ορίζεται ως:

$R_f(f \ominus s)$

Όπου s , το δομικό στοιχείο

4.6.2 Κλείσιμο με Ανακατασκευή

Αντίστοιχα με το Άνοιγμα με Ανακατασκευή, το κλείσιμο με Ανακατασκευή είναι μια μέθοδος που πρώτα γίνεται μια Διαστολή και έπειτα ακολουθείται μορφολογική ανακατασκευή.



4.6.3 Παραδείγματα Ανοίγματος και Κλείσιματος με Ανακατασκευή



α) Αρχική εικόνα



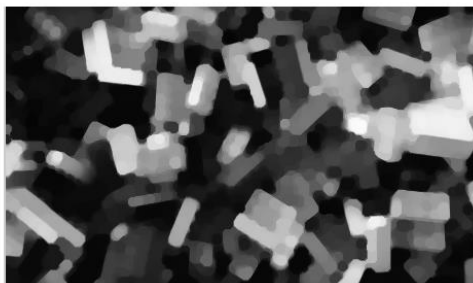
β) Η εικόνα σε αποχρώσεις του γκρι



γ) Η εικόνα μετά από Άνοιγμα



δ) Η εικόνα μετά από Άνοιγμα με Ανακατασκευή

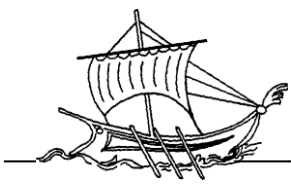


ε) Η εικόνα μετά από κλείσιμο



στ) Η εικόνα μετά από Κλείσιμο με Ανακατασκευή

Σχήμα 4.4 Αποτελέσματα Ανοίγματος και Κλείσιματος με Ανακατασκευή



Ο κώδικας που χρησιμοποιήθηκε για την υλοποίηση είναι:

```
lego = imread('lego','jpg');
figure,imshow(lego);
lego = rgb2gray(lego);
se2 = strel('disk', 5);
Io = imopen(lego, se2);
Ie = imerode(lego, se2);
Iobr = imreconstruct(Ie, lego);
figure, imshow(lego);
figure, imshow(Io);
figure, imshow(Iobr);
Ioc = imclose(Io, se2);
Iobrd = imdilate(Iobr, se2);
Iobrcbr = imreconstruct(imcomplement(Iobrd),
imcomplement(Iobr));
Iobrcbr = imcomplement(Iobrcbr);
figure, imshow(Ioc);
figure, imshow(Iobrcbr);
```



5. ΤΜΗΜΑΤΟΠΟΙΗΣΗ Ή ΚΑΤΑΤΜΗΣΗ ΕΙΚΟΝΑΣ

Στον τομέα της μηχανικής όρασης, ο όρος τμηματοποίηση ή κατάτμηση μιας εικόνας (image segmentation) αναφέρεται στην τμηματοποίηση μιας ψηφιακής εικόνας με στόχο την απλοποίηση και την ανάλυσή της.

Γενικά σε απλές εφαρμογές μηχανικής όρασης μπορούμε να συλλέξουμε πληροφορίες μετά από την επεξεργασία της εικόνας μέσα από κάποια φίλτρα αλλά και με μετατροπή του χρωματικού χώρου. Επίσης, σε αυτή την διαδικασία μπορούμε να χρησιμοποιήσουμε και άλλες μεθόδους επεξεργασίας, όπως ένα φίλτρο υψηλής αντίθεσης, ώστε να καταλήξουμε στη διαδικασία λογικής αποφλοίωσης των pixel, όπου μας δίνει μια δυαδική εικόνα που απεικονίζει, σύμφωνα με τα χαρακτηριστικά κατωφλίου, τις περιοχές που μας ενδιαφέρουν. Μετά από αυτό το στάδιο γίνεται ο διαχωρισμός της εικόνας, δηλαδή η κατάτμησή της.

Στα υψηλά επίπεδα Μηχανικής Όρασης, η διαδικασία της κατάτμησης χρησιμεύει στην αναγνώριση αντικειμένων αλλά και στην παρακολούθηση εικόνων σαν δομικό στοιχείο των διάφορων αλγορίθμων. Μια δυαδική εικόνα είναι το μοναδικό στοιχείο ενδιαφέροντος μας, διότι μας δίνει ένα σύνολο συμπαγών σχημάτων, όπου μέσω της μαθηματικής μορφολογίας μπορούμε να αναλύσουμε τα διαφορετικά αντικείμενα. Με πολύ απλά βήματα γίνεται η εξαγωγή μιας δυαδικής εικόνας, θέτοντάς την σε κατώφλι μιας παραμέτρου της εικόνας. Η φωτεινότητα των pixel αλλά και ο υπολογισμός γειτονικών pixel, με την εφαρμογή προϋποθέσεων υστέρησης μας βοηθούν στην εξαγωγή της εικόνας.

Διάφορα χαρακτηριστικά όπως η υφή αλλά και το χρώμα των αντικειμένων μιας εικόνας μας βοηθούν ώστε να την τμηματοποιήσουμε. Υπάρχουν πολλές τεχνικές κατάτμησης που χωρίζονται σε αυτές, που βασίζονται σε μονόχρωμη επεξεργασία και σε αυτές με έγχρωμη επεξεργασία της εικόνας, όπου βοηθούν να γίνεται η αναγνώριση αντικειμένων, προσώπων κλπ.



Τα αρχικά στάδια της τμηματοποίησης χρώματος μιας εικόνας γίνεται σε αποχρώσεις του μαύρου και του γκρι, και μας δίνει μια προσέγγιση στην εύρεση γωνιών, ακμών και άλλων περιοχών [4].

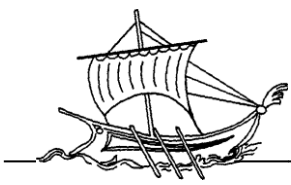
5.1 Κατωφλίωση Ιστογράμματος

Σε αυτή την τεχνική, η τμηματοποίηση βασίζεται στο ιστόγραμμα, το οποίο είναι μονοδιάστατο για μονόχρωμες εικόνες και πολυδιάστατο για έγχρωμες. Η γενικότερη λογική είναι ότι η ανίχνευση κορυφών ή κοιλάδων στο ιστόγραμμα γίνεται από το διαχωρισμό του φόντου της. Τα ιστογράμματα επεξεργάζονται ξεχωριστά και προκύπτει ένα ολοκληρωμένο αποτέλεσμα. Συνηθίζεται, οι αλγόριθμοι που εφαρμόζουν κατωφλίωση ιστογράμματος, να μετασχηματίζουν τις εικόνες στα χρωματικά μοντέλα HSV, HSI ή HSL. Οι μετασχηματισμοί αυτοί δίνουν την ευκολία του διαχωρισμού των χρωμάτων τα οποία είναι όμοια σε μια ομάδα χρωμάτων. Το πρόβλημα του θορύβου είναι το μόνο αξιοσημείωτο, δηλαδή οι κορυφές που δημιουργούνται στο ιστόγραμμα που πολλές φορές μπορούν να επιφέρουν ακόμα και την αποτυχία του προγράμματος. Εδώ χρησιμοποιείται η ομαλοποίηση (smoothing) του ιστογράμματος ώστε να μειωθεί η επίδραση του θορύβου.

5.2 Παράδειγμα τμηματοποίηση με κατωφλίωση ιστογράμματος

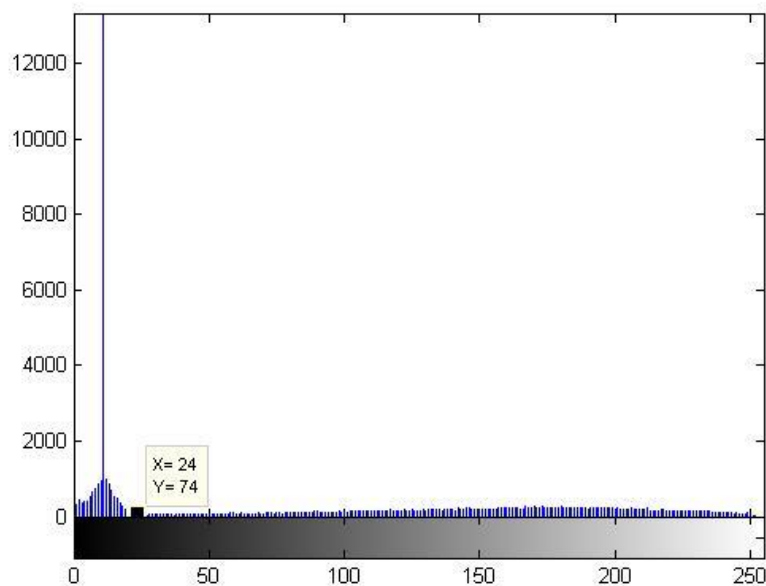
Παρακάτω είναι τα βήματα τμηματοποίησης με τη χρήση κατωφλίωσης του ιστογράμματος μιας εικόνας.

- 1) Στο MatLab διαβάζουμε την εικόνα που θέλουμε να τμηματοποιήσουμε, και την μετατρέπουμε σε gray.



Σχήμα 5.1: Εικόνα μετασηματισμένη σε gray

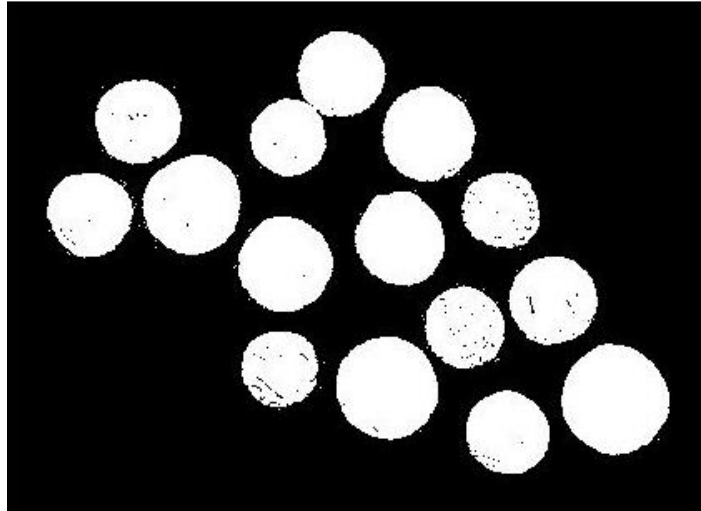
- 2) Στη συνέχεια εμφανίζουμε το ιστόγραμμα της εικόνας. Αναγνωρίζουμε τις τιμές έντασης που έχει και διακρίνουμε το φόντο από τα αντικείμενα που θέλουμε, και τις λεπτομέρειες πάνω στα κέρματα. Για να εφαρμοστεί ο κώδικας επιλέγεται χειροκίνητα το σημείο που χωρίζει τις τιμές.



Σχήμα 5.2: Ιστόγραμμα (Η τιμή x είναι η τιμή έντασης που χωρίζει την εικόνα)



- 3) Χρησιμοποιώντας τα παραπάνω προκύπτει η τελική εικόνα, με τα ζητούμενα τα αντικείμενα.



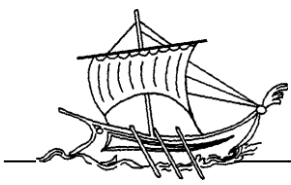
Σχήμα 5.3: Τελικό αποτέλεσμα

Παρακάτω φαίνεται ο κώδικας γραμμένος στο MATLAB.

```
x=imread('coin.jpg');  
im=rgb2gray(x);  
figure,imshow(im);  
figure, imhist(im);  
T=24;  
It=im2bw(im,T/255);  
figure,imshow(It);
```

5.3 Αλγόριθμος Μετασχηματισμού Κλιμακωτών Αναλλοίωτων Χαρακτηριστικών

Ο αλγόριθμος εφαρμόζεται στην περίπτωση, όπου απαιτείται να γίνει περιγραφή των χαρακτηριστικών των αντικειμένων μιας εικόνας εξάγοντας τα σημαντικά σημεία ή σημεία ενδιαφέροντος των αντικειμένων (feature description). Σαν σημαντικό σημείο ή σημείο

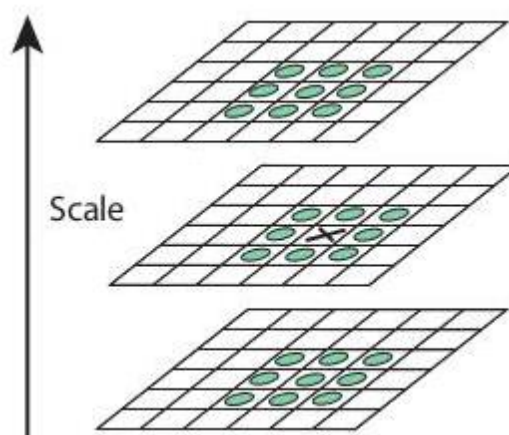


ενδιαφέροντος ορίζεται το pixel το οποίο είναι σε μια θέση, συγκεκριμένη, εύκολα ανιχνεύσιμη και περιεκτική σε τοπικές πληροφορίες. Από την εικόνα εκπαίδευσης (η εικόνα που παρήγαγε τα χαρακτηριστικά και περιέχει μόνο το αντικείμενο που ζητείται) εντοπίζεται το αντικείμενο το οποίο βρίσκεται μέσα σε μια δεύτερη εικόνα με διάφορα άλλα αντικείμενα. Έτσι εξάγονται οι ακμές των αντικείμεμων και έπειτα αποθηκεύονται στη βάση δεδομένων. Μέσω της σύγκρισης της μιας εικόνας με διάφορα αντικείμενα και της βάση δεδομένων ανιχνεύονται τα αντικείμενα που ψάχνουμε. Στο τέλος διάφορα φίλτρα ελέγχουν τις ακραίες τιμές και διατηρούνται τα ταιριάσματα που διαπιστώσαμε ότι είναι αξιόπιστα [4].

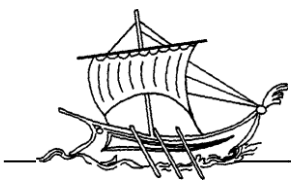
Κλιμακωτή ανίχνευση ακρότατων στοιχείων στο χώρο

Η διαδικασία της κλιμακωτής ανίχνευσης ακρότατων στοιχείων στο χώρο είναι η πρώτη φάση της επεξεργασίας, όπου όλες οι τοποθεσίες και οι κλίμακες μιας εικόνας διερευνώνται. Η πρώτη φάση της επεξεργασίας είναι αρκετά χρήσιμη διότι δεν μπορεί να γίνει η χρήση του ίδιου παραθύρου για την ανίχνευση των σημείων-κλειδιών σε άλλη κλίμακα. Μέσω Λαπλασιανών των Γκαουσιανών φίλτρων πραγματοποιείται η φάση αυτή. Τα φίλτρα αυτά ανιχνεύουν τα πιθανά σημαντικά σημεία της εικόνας με διάφορες «σ» (σίγμα) τιμές, οι οποίες εκτελούνται ως συντελεστής κλίμακας με τη διαφορά των Γκαουσιανών σε μια σειρά από ομαλοποιήσεις και επαναδειγματοληφθέντες εικόνες.

Η Γκαουσιανή διαφορά είναι η σύγκριση μεταξύ του κάθε εικονοστοιχείου με τα γειτονικά του. Πάντα πρέπει το επιλεγόμενο pixel να έχει διαφορετική φωτεινότητα από τα γειτονικά.



Σχήμα 5.4 Υπολογισμός του μεγίστου και ελαχίστου της Γκαουσιανής διαφοράς σε ένα εικονοστοιχείο.



Ανίχνευση των σημείων κλειδιών:

Η δεύτερη φάση είναι η ανίχνευση των σημείων κλειδιών. Τα σημεία κλειδιά πρέπει να προσδιοριστούν, μετά από τον εντοπισμό των περιοχών της τοποθεσίας τους, έτσι ώστε να μπορούν να παρθούν ακριβή αποτελέσματα. Χρησιμοποιώντας τη σειρά Τέιλορ, επεκτείνουμε την κλίμακα του χώρου. Έπειτα λαμβάνεται η ακριβής θέση των ακρότατων των σημείων, και σε συνδυασμό με την έντασή τους, αν η τιμή της είναι μικρότερη από αυτή του κατωφλιού, τα απορρίπτουμε. Εν ολίγοις, στη δεύτερη φάση διαλέγουμε τα σημεία κλειδιά σύμφωνα με την τιμή της σταθερότητας τους.

Ανάθεση προσανατολισμού:

Στη φάση αυτή, σύμφωνα με τις τοπικές κατευθύνσεις και τις κλίσεις της εικόνας, όλα τα σημεία κλειδιά έχουν τον δικό τους προσανατολισμό. Τα σημεία κλειδιά παραμένουν αμετάβλητα σε οποιαδήποτε περιστροφή της εικόνας. Χρησιμοποιώντας την περιοχή που περιβάλλει το σημείο κλειδί, η οποία είναι ανάλογη της κλίμακας, μπορεί να υπολογιστεί το βαθμωτό μέγεθός του, όπως και η κατεύθυνσή του.

Περιγραφή των σημείων κλειδιών:

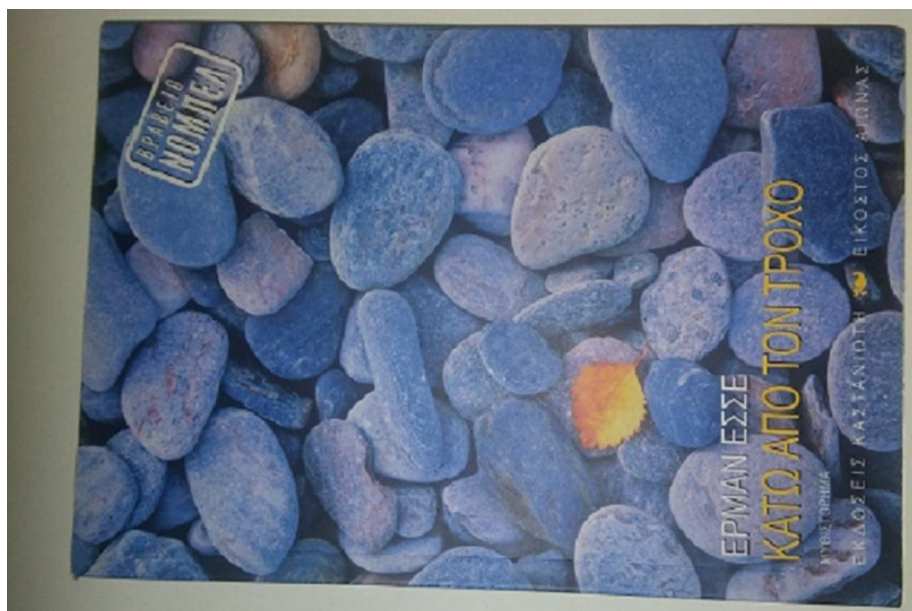
Στη φάση αυτή γίνεται η περιγραφή των σημείων κλειδιών. Δημιουργούνται αναπαραστάσεις, που έχουν συγκεκριμένη κλίμακα, από τις βαθμίδες των τοπικών εικόνων που περιβάλλουν το κάθε σημείο κλειδί. Οι αναπαραστάσεις αυτές έχουν την ιδιότητα να επιτρέπουν οποιαδήποτε αλλαγή της φωτεινότητας και διαστρέβλωση της τοπικής μορφής.

5.4 Παράδειγμα κατάτμισης και εύρεσης αντικειμένου με αλγόριθμο SIFT

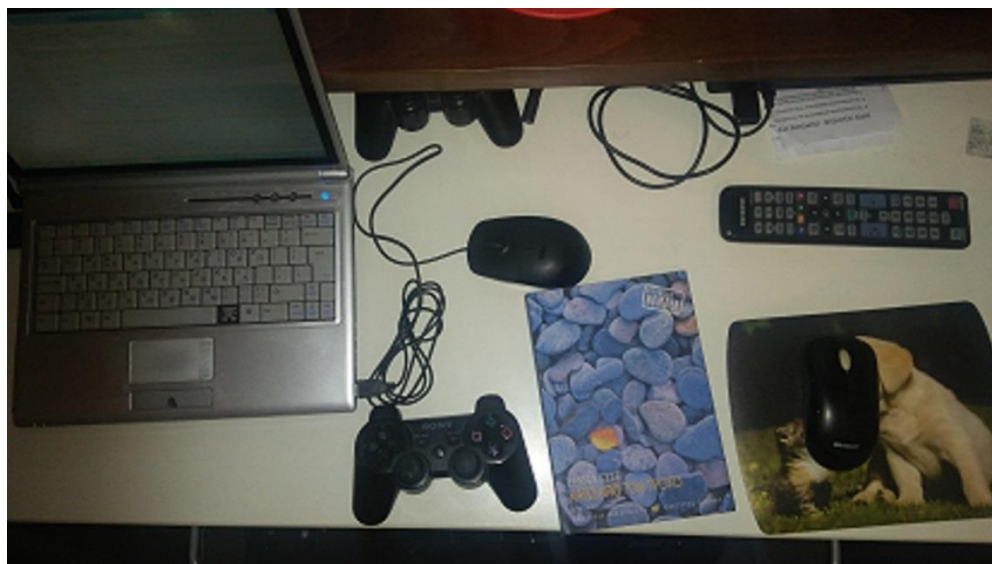
Για να υλοποιήσουμε τη διαδικασία χρησιμοποιήθηκε η συνάρτηση `sift` που υπάρχει στην βιβλιοθήκη του `MatLab`. Οι 4 φάσεις του αλγορίθμου SIFT φαίνονται παρακάτω. Καταρχάς



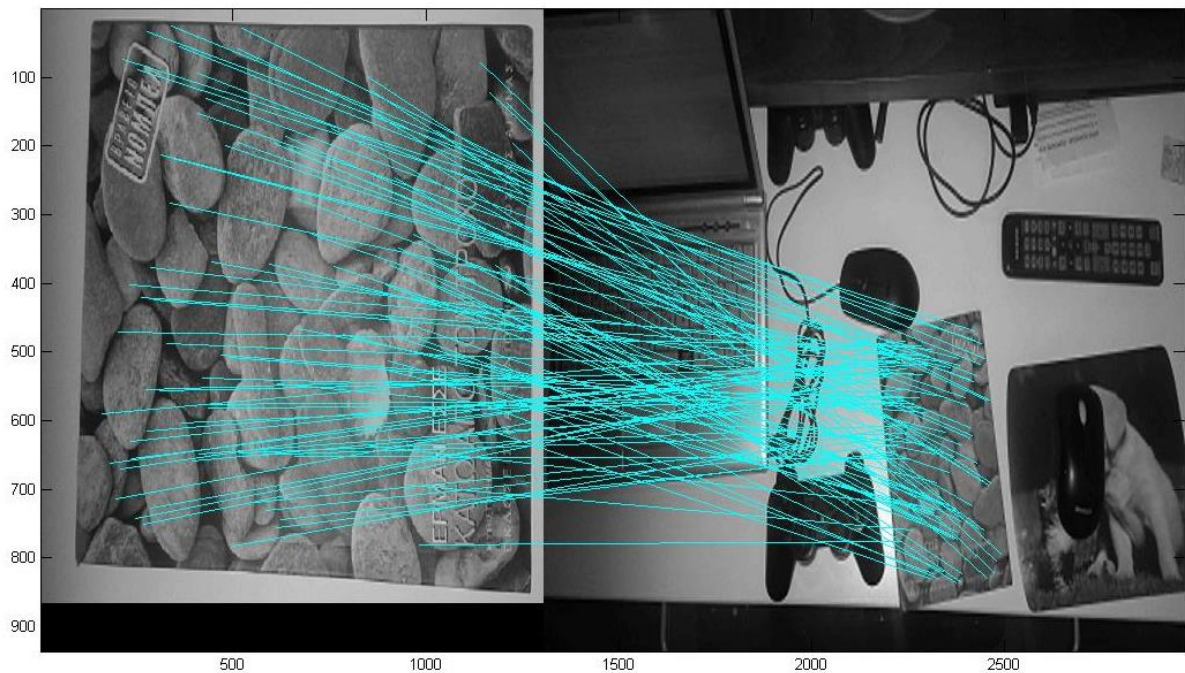
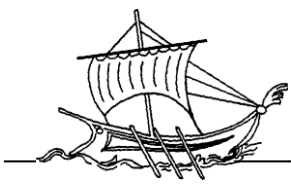
επιλέγονται 2 εικόνες. Η πρώτη έχει ένα βιβλίο και η δεύτερη έχει διάφορα αντικείμενα, μαζί με το βιβλίο της πρώτης εικόνας και προσπαθήσαμε να κάνουμε τα ταιριάσματα.



Σχήμα 5.5: Η εικόνα με το ζητούμενο αντικείμενο



Σχήμα 5.6: Η εικόνα με τα διάφορα αντικείμενα



Σχήμα 5.7: Η τελική εικόνα με τα ταιριάσματα

Όπως βλέπουμε παραπάνω, μπορούμε να διακρίνουμε τα σημεία κλειδιά που βρήκε το πρόγραμμα και έκανε το ταιρίασμα με την δεύτερη εικόνα

Παρακάτω αναγράφονται ο κώδικας και τα script που χρησιμοποιήθηκαν στο MatLab.

```
>> num=match('book.jpg','items.jpg');  
Finding keypoints...  
3819 keypoints found.  
Finding keypoints...  
2394 keypoints found.  
Found 99 matches.  
>>
```



Ο κώδικας του `match.m` για το ταίριασμα είναι [5]:

```
function num = match(image1, image2)

% Find SIFT keypoints for each image
[im1, des1, loc1] = sift(image1);
[im2, des2, loc2] = sift(image2);

% For efficiency in Matlab, it is cheaper to compute dot
products between
% unit vectors rather than Euclidean distances. Note that
the ratio of
% angles (acos of dot products of unit vectors) is a close
approximation
% to the ratio of Euclidean distances for small angles.
%
% distRatio: Only keep matches in which the ratio of vector
angles from the
% nearest to second nearest neighbor is less than distRatio.
distRatio = 0.6;

% For each descriptor in the first image, select its match to
second image.
des2t = des2'; % Precompute matrix transpose
for i = 1 : size(des1,1)
    dotprods = des1(i,:) * des2t; % Computes vector of
dot products
    [vals,indx] = sort(acos(dotprods)); % Take inverse cosine
and sort results

% Check if nearest neighbor has angle less than distRatio
times 2nd.
```




```
if (vals(1) < distRatio * vals(2))
    match(i) = indx(1);
else
    match(i) = 0;
end
end

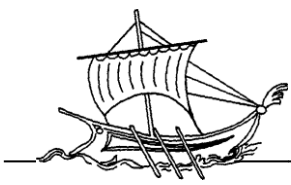
% Create a new image showing the two images side by side.
im3 = appendimages(im1,im2);

% Show a figure with lines joining the accepted matches.
figure('Position', [100 100 size(im3,2) size(im3,1)]);
colormap('gray');
imagesc(im3);
hold on;
cols1 = size(im1,2);
for i = 1: size(des1,1)
    if (match(i) > 0)
        line([loc1(i,2) loc2(match(i),2)+cols1], ...
            [loc1(i,1) loc2(match(i),1)], 'Color', 'c');
    end
end
hold off;
num = sum(match > 0);
fprintf('Found %d matches.\n', num);
```

Ο κώδικας του αλγόριθμου SIFT για το ταίριασμα είναι:

```
function [image, descriptors, locs] = sift(imageFile)

% Load image
image = imread(imageFile);
```



```
% If you have the Image Processing Toolbox, you can uncomment
the following
%     lines to allow input of color images, which will be
converted to grayscale.
if size(image,3)==3
    image = rgb2gray(image);
end

[rows, cols] = size(image);

% Convert into PGM imagefile, readable by "keypoints"
executable
f= fopen('tmp.pgm', 'w');
if f == -1
error('Could not create file tmp.pgm.');
```

```
end
fprintf(f, 'P5\n%d\n%d\n255\n', cols, rows);
imageT=image.';
fwrite(f, imageT, 'uint8');
fclose(f);

% Call keypoints executable
if isunix
    command = '!./sift ';
else
    command = '!siftWin32 ';
end
command = [command ' <tmp.pgm >tmp.key'];
eval(command);

% Open tmp.key and check its header
g = fopen('tmp.key', 'r');
```



```
if g == -1
error('Could not open file tmp.key.');
```

```
end

[header, count] = fscanf(g, '%d %d', [1 2]);
if count ~= 2
error('Invalid keypoint file beginning.');
```

```
end

num = header(1);
len = header(2);
if len ~= 128
error('Keypoint descriptor length invalid (should be 128).');
```

```
end

% Creates the two output matrices (use known size for
efficiency)
locs = double(zeros(num, 4));
descriptors = double(zeros(num, 128));

% Parse tmp.key
for i = 1:num
    [vector, count] = fscanf(g, '%f %f %f %f', [1 4]); %row
col scale ori
    if count ~= 4
error('Invalid keypoint file format');
```

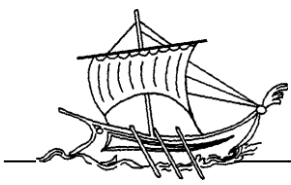
```
end

locs(i, :) = vector(1, :);

    [descrip, count] = fscanf(g, '%d', [1 len]);
if (count ~= 128)
error('Invalid keypoint file value.');
```

```
end

% Normalize each input vector to unit length
```

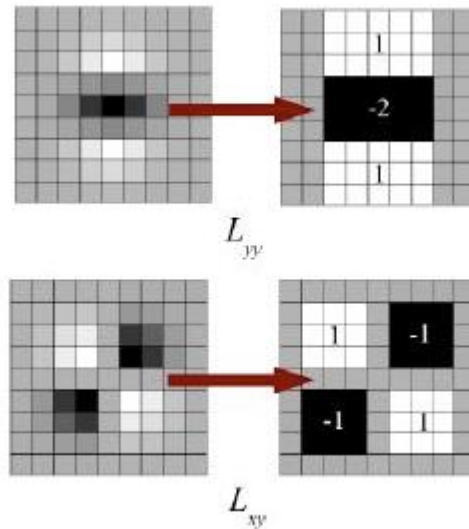
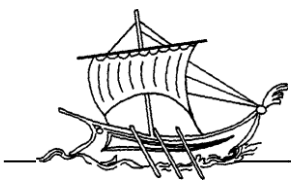


```
descrip = descrip / sqrt(sum(descrip.^2));  
descriptors(i, :) = descrip(1, :);  
end  
fclose(g);
```

5.5 Αλγόριθμος επιταχυνθέντων ισχυρών χαρακτηριστικών (SURF)

Προηγουμένως μελετήθηκε συνοπτικά ο αλγόριθμος SIFT για τον προσδιορισμό των σημείων κλειδιών. Αποδείχθηκε πως ο αλγόριθμος αυτός είναι σχετικά αργός. Εξ' αιτίας αυτού, οι προγραμματιστές χρειάζονται μια νέα έκδοση του αλγορίθμου που η διαδικασία της να είναι αρκετά γρηγορότερη. Οι Bay, Tuytelaars και Van Gool δημιούργησαν το 2006 μια νέα μέθοδο, τον αλγόριθμο των επιταχυνθέντων ισχυρών χαρακτηριστικών (Speeded Up Robust Features). Ως γρηγορότερη εκδοχή του προηγούμενου αλγορίθμου SIFT, ο οποίος λειτουργεί πιο έμπιστα στις διάφορες μεταβολές της εικόνας.

Στον αλγόριθμο χρησιμοποιείται ένα φίλτρο κουτιών για την προσέγγιση της Λαπλασιανής της Γκαουσιανής. Ο υπολογισμός της συνέλιξης με φίλτρο κουτιών υπολογίζεται μέσω των ενσωματωμένων εικόνων. Μπορεί, επίσης, παραλλήλως να γίνει για διάφορες κλίμακες. Αυτός ο υπολογισμός είναι ίσως, το μεγαλύτερο πλεονεκτήματα του αλγορίθμου SURF. Βάση της διακρίνουσας του πίνακα Hessian, καθορίζεται η κλίμακα και η τοποθεσία ενός αντικειμένου.



Σχήμα 5.8 Προσέγγιση της Λαπλασιανής της Γκαουσιανής με το φίλτρο κουτιών

Υπάρχουν τρία βασικά βήματα που πρέπει να ακολουθηθούν με σκοπό την αναζήτηση των ταιριασμάτων μεταξύ του ζητούμενου αντικειμένου και της εικόνας που έχουμε.

1. Εξέταση των σημείων ενδιαφέροντος στα διακριτά μέρη της εικόνας (π.χ. γωνίες). Επαναλαμβάνοντας, υπάρχει μεγαλύτερη αξιοπιστία στα σημεία που ανιχνεύονται, ανεξαρτήτως την οποιαδήποτε αλλαγή των συνθηκών που βρίσκεται το σημείο.
2. Ένα διάνυσμα χαρακτηριστικών περιγράφει τα σημεία που γειτονεύουν με το σημείο ενδιαφέροντος. Το σημείο ενδιαφέροντος παραμένει αναλλοίωτο σε οποιαδήποτε περίπτωση παραμόρφωσης ή θορύβου.
3. Εξέταση των διανυσματικών χαρακτηριστικών με σκοπό την ταύτιση ανάμεσα σε διάφορες εικόνες και την αντιστοιχία.

Πολλά χαρακτηριστικά που σκοπεύουν στην ταχύτερη απόδοση, προστίθενται μέσω του αλγορίθμου επιταχυνθέντων ισχυρών χαρακτηριστικών, στα προηγούμενα βήματα. Η απόδοση εξαγωγής χαρακτηριστικών του αλγορίθμου SURF είναι η ίδια με την απόδοση του αλγορίθμου SIFT, μόνο που ο αλγόριθμος SURF είναι τρεις φορές πιο γρήγορος. Τέλος, μέσω του αλγορίθμου SURF, κάποιος μπορεί να χειριστεί πάρα πολύ εύκολα τις εικόνες με περιστροφή ή θόλωση, αλλά δεν μπορεί να χειριστεί εύκολα οποιαδήποτε αλλαγή της οπτικής γωνίας ή του φωτισμού.

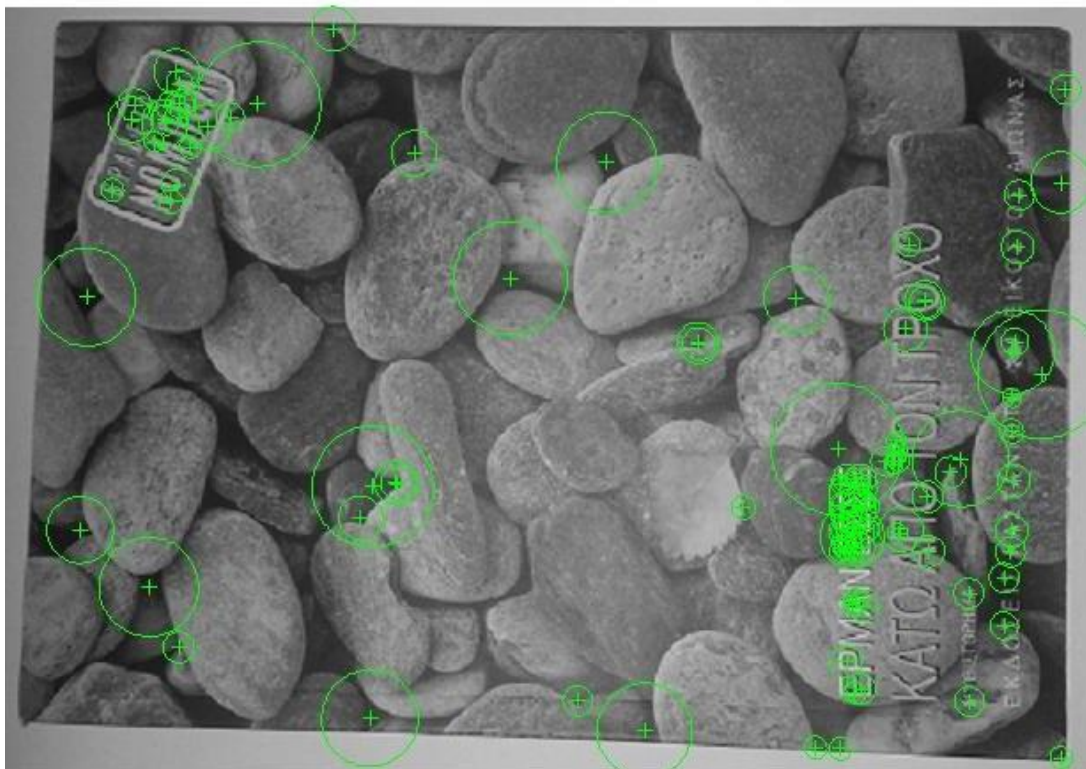


5.6 Παράδειγμα εύρεσης αντικειμένου με τη χρήση του αλγόριθμου SURF

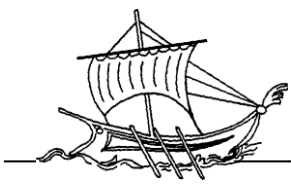
Στο παράδειγμα που ακολουθεί παρουσιάζεται ο εντοπισμός ενός αντικειμένου με τη χρήση του αλγορίθμου SURF. Οι εικόνες που χρησιμοποιήθηκαν είναι οι ίδιες με τις εικόνες του SIFT. Παρακάτω εξηγούνται και τα βήματα που ακολουθήθηκαν.

- 1) Διαβάσαμε τις εικόνες και με την χρήση του SURF, εντοπίζοντας τα πιο σημαντικά χαρακτηριστικά τους

Τα 100 πιο σημαντικά σημεία ενδιαφέροντος της εικόνας book



Σχήμα 5.8: Τα 100 πιο σημαντικά χαρακτηριστικά της πρώτης εικόνας



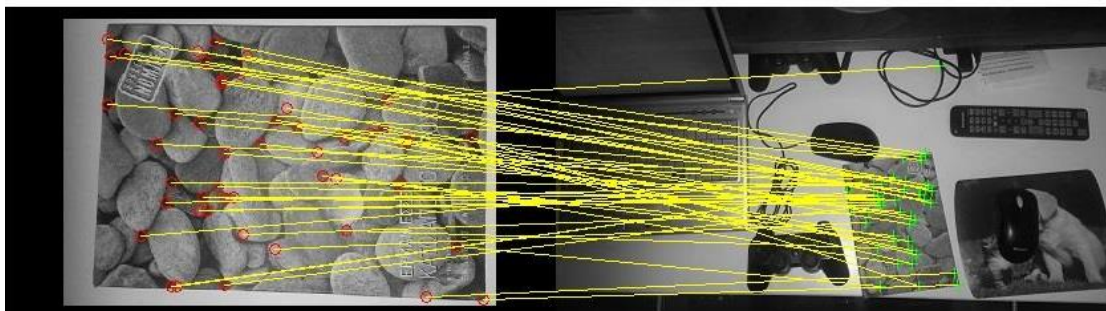
Τα 300 πιο σημαντικά σημεία ενδιαφέροντος της εικόνας items



Σχήμα 5.9: Τα 300 πιο σημαντικά χαρακτηριστικά της δεύτερης εικόνας

- 2) Έγινε εξαγωγή των χαρακτηριστικών περιγραφής στα σημεία ενδιαφέροντος των εικόνων και το ταίριασμα τους.

Υποθετικά σημεία ταιριάσματος(Συμπεριλαμβανομένων ακραίων τιμών)

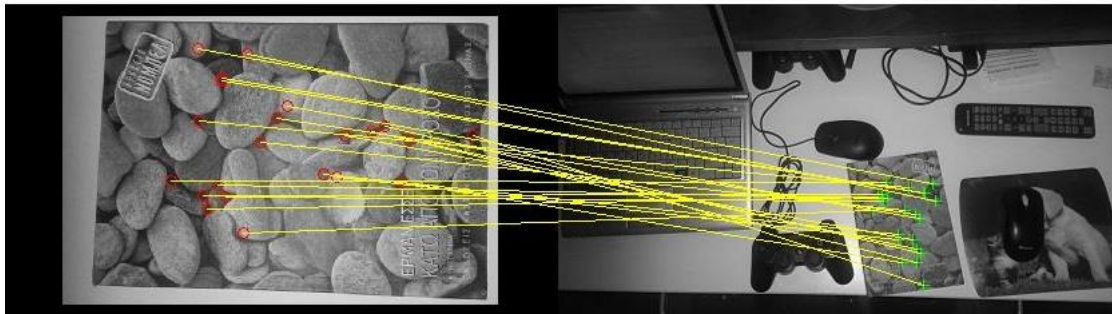


Σχήμα 5.10: Τα ταιριάσματα που έγιναν μαζί με τα ακραία σημεία

- 3) Κρατήσαμε μόνο τα πιο σχετικά σημεία και αποκλείσαμε τα ακραία σημεία. Έτσι παίρνουμε μόνο το αντικείμενο που θέλουμε.



Σημεία ταιριάσματος (Χωρίς τις ακραίες τιμές)



Σχήμα 5.11: Τα ταιριάσματα που έγιναν χωρίς τα ακραία σημεία

- 4) Πήραμε το πολύγωνο που οριοθετεί την εικόνα μεταφοράς και το μεταμορφώσαμε αναλόγως στη δεύτερη εικόνα. Το τελικό πολύγωνο υποδεικνύει την περιοχή που είναι το αντικείμενο που ζητάμε.

Ανιχνευμένο αντικείμενο



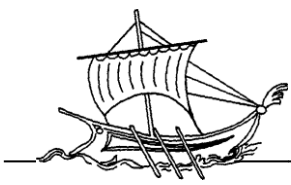
Σχήμα 5.12: Το ανιχνευμένο αντικείμενο

Παρακάτω φαίνεται ο κώδικας που υλοποιήθηκε στο MatLab.

```
bookImage = imread('book.jpg');  
bookImage = rgb2gray(bookImage);  
itemsImage = imread('items.jpg');  
itemsImage = rgb2gray(itemsImage);
```




```
bookPoints = detectSURFFeatures(bookImage);
itemsPoints = detectSURFFeatures(itemsImage);
figure;
imshow(bookImage);
title('Τα 100 πιο σημαντικά σημεία ενδιαφέροντος της εικόνας
book');
hold on;
plot(selectStrongest(bookPoints, 100));
figure;
imshow(itemsImage);
title('Τα 300 πιο σημαντικά σημεία ενδιαφέροντος της εικόνας
items');
hold on;
plot(selectStrongest(itemsPoints, 300));
[bookFeatures, bookPoints] = extractFeatures(bookImage,
bookPoints);
[itemsFeatures, itemsPoints] = extractFeatures(itemsImage,
itemsPoints);
bookPairs = matchFeatures(bookFeatures, itemsFeatures);
matchedBookPoints = bookPoints(bookPairs(:, 1), :);
matchedItemsPoints = itemsPoints(bookPairs(:, 2), :);
figure;
showMatchedFeatures(bookImage, itemsImage, matchedBookPoints,
matchedItemsPoints, 'montage');
title('Υποθετικά σημεία ταιριάσματος (Συμπεριλαμβανομένων
ακραίων τιμών)');
[tform, inlierBookPoints, inlierItemsPoints] =
estimateGeometricTransform(matchedBookPoints,
matchedItemsPoints, 'affine');
figure;
showMatchedFeatures(bookImage, itemsImage, inlierBookPoints,
inlierItemsPoints, 'montage');
title('Σημεία ταιριάσματος (Χωρίς τις ακραίες τιμές)');
bookPolygon = [1, 1; size(bookImage, 2), 1; size(bookImage,
2), size(bookImage, 1); 1, size(bookImage, 1); 1, 1];
newBookPolygon = transformPointsForward(tform, bookPolygon);
figure;
imshow(itemsImage);
hold on;
line(newBookPolygon(:, 1), newBookPolygon(:, 2), 'Color',
'y');
title('Ανιχνευμένο αντικείμενο');
```



5.7 Αλγόριθμος κατάτμησης της εικόνας με βάση το χρώμα

Σε αυτή την ενότητα παρουσιάζεται ο αλγόριθμος κατάτμησης με βάση το χρώμα. Η διαδικασία τμηματοποίησης αυτού του αλγορίθμου εμπεριέχει την ανάλυση των χρωμάτων των εικονοστοιχείων, έτσι ώστε να γίνει διαχωρισμός των διαφορετικών αντικειμένων που βρίσκονται σε μια εικόνα. Είναι η διαδικασία τμηματοποίησης εικόνας σε υποσύνολα των εικονοστοιχείων που έχουν όμοιες ιδιότητες χρώματος.

Σε αυτόν τον αλγόριθμο, η μέθοδος προσέγγισης που ακολουθείται είναι η συλλογή δειγμάτων των εικονοστοιχείων. Έπειτα ελέγχεται η χρωματική περιοχή στην οποία ανήκουν με σκοπό την ομαδοποίηση των εικονοστοιχείων. Το αποτέλεσμα αυτής της μεθόδου είναι η εξαγωγή του περιγράμματος του ζητούμενου αντικειμένου.

Λόγω της μεγάλης προόδου που έχει γίνει στη δυνατότητα επεξεργασίας των υπολογιστών, η τμηματοποίηση της εικόνας χρησιμοποιώντας τις πληροφορίες των χρωμάτων εφαρμόζεται σε πολλές εφαρμογές και διευρύνεται το ενδιαφέρον από την ερευνητική κοινότητα.

Το μεγαλύτερο πρόβλημα που παρουσιάζεται σε αυτήν την μέθοδο είναι η ευαισθησία σε μεταβολές συνθηκών φωτισμού, όπως και η περιπλοκότητα σε εφαρμογές που τμηματοποιούν εικόνες, οι οποίες εμπεριέχουν διάφορα αντικείμενα που έχουν ίδιο ή και διαφορετικό χρωματισμό.

5.8 Μέθοδος Κατάτμησης με Υφή

Η κατάτμηση των ιατρικών εικόνων μπορεί να γίνει με τη γνώση της υφής της εικόνας. Μέσω της υφής είναι δυνατόν να τμηματοποιηθεί η εικόνα. Η μέθοδος κατάτμησης με υφή γίνεται εφικτή με τη χρήση της εντροπίας (entropy), η οποία παίζει σημαντικό ρόλο για να δημιουργηθεί μια εικόνα υφής. Η εντροπία είναι ένα στατιστικό μέτρο της τυχειότητας που μπορεί να χρησιμοποιηθεί για τον χαρακτηρισμό της υφής. Άρα η τεχνική αυτή ανήκει στις στατιστικές μεθόδους ανίχνευσης και περιγραφής της υφής.



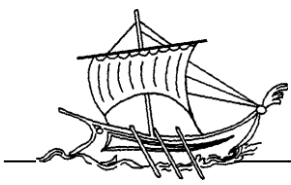
Επίσης η χρήση της εντροπίας είναι γνωστή στην ανάλυση υφής, μέσω της τεχνικής maximum entropy random field (MERF). Η τεχνική αυτή βασίζεται στην πολλαπλή ανάλυση φίλτρων με τη μέγιστη εντροπία, τα οποία μπορούν να χρησιμοποιηθούν για την εξαγωγή χαρακτηριστικών υφής σε διάφορες κατευθύνσεις και κλίμακες. Η κατανομή της πιθανότητάς της μπορεί να θεωρηθεί ως κατανομή Gibbs.

5.9 Κατάτμηση βάση περιοχών

Μια εξειδικευμένη κατηγορία κατάτμησης εικόνας είναι η κατάτμηση βάση περιοχών της εικόνας. Η ομοιομορφία των περιοχών εξαρτάται κάθε φορά από την εικόνα.

5.10 Τοπολογικά Χαρακτηριστικά

Ένα ακόμη χαρακτηριστικό που μας δίνει σημαντική πληροφορία σε μια εικόνα είναι η θέση και τοποθέτηση των αντικειμένων μέσα σε αυτή, δηλαδή τα τοπολογικά χαρακτηριστικά της. Για παράδειγμα ο ουρανός συνήθως βρίσκεται στο πάνω, ενώ η θάλασσα στο κάτω μέρος της εικόνας. Μας ενδιαφέρει όχι μόνο η χωρική θέση των αντικειμένων ή των περιοχών αλλά και οι σχέσεις μεταξύ των περιοχών της εικόνας (π.χ. το αντικείμενο α βρίσκεται στο πάνω αριστερό μέρος της εικόνας και είναι δεξιά του β και πάνω από το γ). Επιτρέπει το ταίριασμα εικόνων βασισμένη στις σχετικές θέσεις των αντικειμένων μέσα σε αυτές. Υπάρχουν πολύ τρόποι για την εξαγωγή των τοπολογικών χαρακτηριστικών. Ένας από αυτούς είναι να χωρίσουμε την εικόνα σε εννέα κελιά (3x3) και σε κάθε περιοχή ανάλογα με το τι καλύπτει μεγαλύτερο μέρος της επιφάνειάς της, δημιουργούμε μια τιμή δείκτη (μία για κάθε κελί) που έπειτα μπορεί να συνδυαστεί και με το χρώμα και το σχήμα για καλύτερα αποτελέσματα.



6. ΠΕΡΙΓΡΑΦΗ ΣΧΗΜΑΤΟΣ

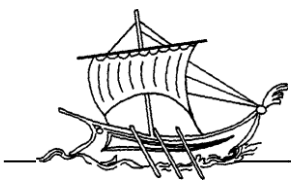
Ένα πολύ σημαντικό θέμα στη σύνθεση και στη ανάλυση μιας εικόνας η περιγραφή σχήματος. Δίνει λύση στο πρόβλημα που υπάρχει στην αντιστοίχιση αντικειμένων ή περιοχών ανάμεσα σε 2 ή παραπάνω εικόνες. Το πρόβλημα αυτό, της αντιστοίχισης των εικόνων (ή αλλιώς image registration), υπάρχει σε αρκετές εφαρμογές της Επεξεργασίας εικόνας και της Υπολογιστικής όρασης. Για παράδειγμα κάποιες από τις εφαρμογές αυτές είναι:

- Object Tracking
- Stereo Correspondence
- Optical Flow
- Motion Estimation
- Image Alignment

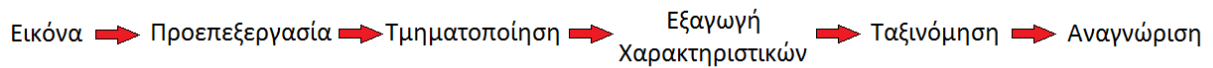
Υπάρχουν δύο τρόποι για την περιγραφή δισδιάστατων σχημάτων.

- I. Γίνεται χρήση του περιγράμματος (contour), έτσι ώστε να εξαχθούν τα χαρακτηριστικά για την περιγραφή του. Χαρακτηριστικό θεωρούμε κάποιο μετρήσιμο μέγεθος. Αυτή η μέθοδος συνδέεται άμεσα με τις τεχνικές της ανίχνευσης ακμών
- II. Γίνεται χρήση της επιφάνειας (region) που καταλαμβάνεται από το αντικείμενο.

Ο λόγος που η περιγραφή των αντικειμένων γίνεται με χαρακτηριστικά είναι επειδή τα αντικείμενα αναπαριστώνται στο χώρο των χαρακτηριστικών και έτσι καταφέρνουμε να τα αναγνωρίσουμε μέσω της σύγκρισης των χαρακτηριστικών τους. Γι' αυτό το λόγο τα αντικείμενα πρέπει να χωρίζονται και να περιγράφονται σε όσο το δυνατόν πιο συγκεκριμένα χαρακτηριστικά, τα οποία να έχουν όσο το δυνατόν λιγότερες ιδιότητες. Οπότε καταλαβαίνουμε ότι οι ιδιότητες αναπαράστασης σχήματος είναι:



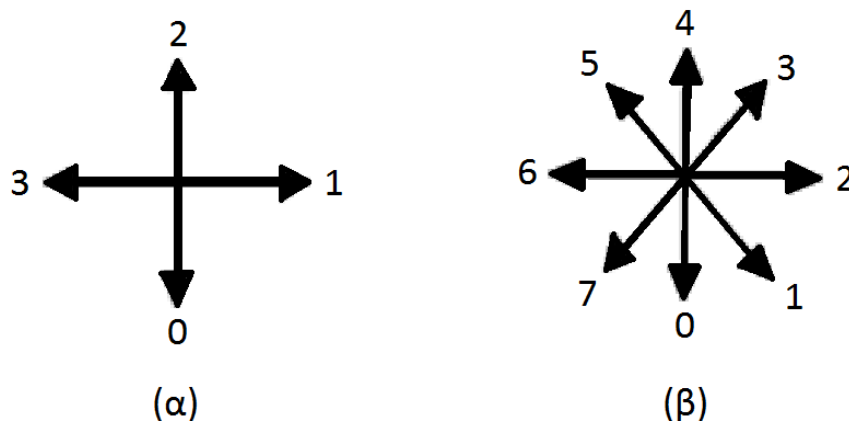
- i. Μοναδικότητα. Κρίσιμη ιδιότητα, διότι η αναπαράσταση κάθε αντικειμένου πρέπει να είναι μοναδική.
- ii. Πληρότητα. Η δυνατότητα να αναπαρασταθούν όλα τα πιθανά σχήματα.
- iii. Αμεταβλητότητα. Κατά την διάρκεια περιστροφής, ανάκλασης, μετατόπισης και κλιμάκωσης η αμεταβλητότητα αποτελεί σημαντικό παράγοντα.
- iv. Ευαισθησία. Η ευκολία της αντανάκλασης των διαφορών μεταξύ παραπλήσιων εικόνων.
- v. Απαλοιφή λεπτομερειών. Η ικανότητα αφαίρεσης των λεπτομερειών από ένα σχήμα έτσι ώστε να αναπαρασταθούν μόνο τα βασικά χαρακτηριστικά του.



Σχήμα 6.1: Διάταξη εξαγωγής και χρήσης των χαρακτηριστικών

6.1 Chain Code

Η κωδικοποίηση Chain Code (αλυσίδας) χρησιμεύει στην παράσταση των περιγραμμάτων των αντικειμένων χωρίς απώλειες. Με τη δημιουργία συνδεδεμένων γραμμικών τμημάτων, τα οποία έχουν προκαθορισμένα μήκη και διευθύνσεις, περιγράφονται τα περιγράμματα. Οι διευθύνσεις της παράστασης είναι συνήθως 4 ή 8 οι οποίες φαίνονται στο παρακάτω σχήμα



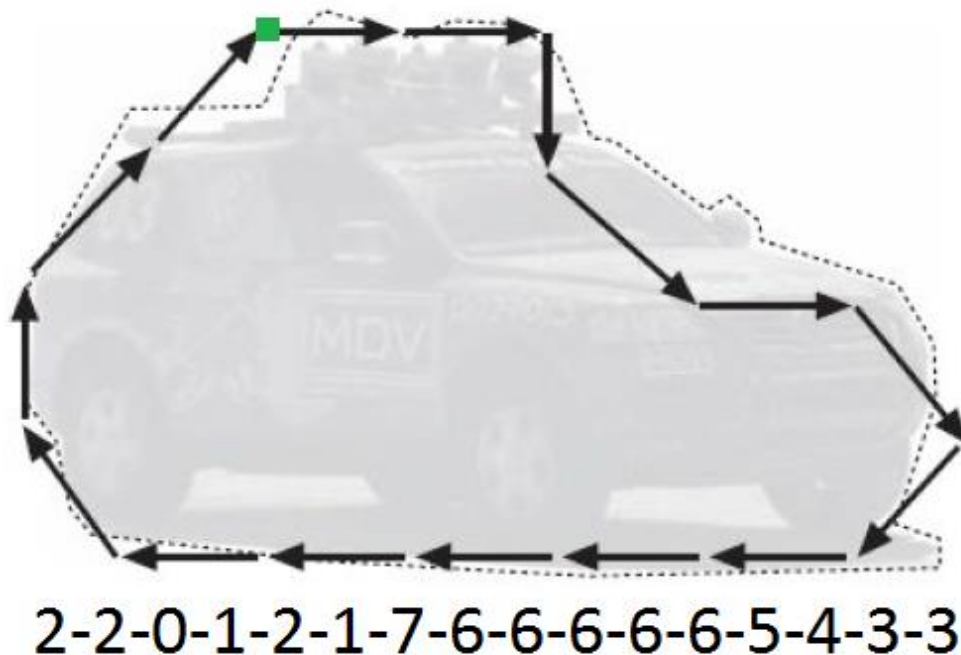
Σχήμα 6.2: Διευθύνσεις του Chain Code



Η διαδικασία της κωδικοποίησης chain code έχει 3 βήματα:

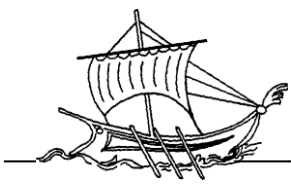
- I. Αρχικοποίηση σε ένα pixel του περιγράμματος
- II. Προσδιορισμός του κοντινότερου pixel
- III. Μετάβαση σε αυτό

Κάθε φορά που τελειώνουν αυτά τα 3 βήματα ο κωδικοποιητής ορίζει το σύμβολο-αριθμό που αντιπροσωπεύει την κίνηση που έγινε. Στο τέλος όλης της διαδικασίας, το αποτέλεσμα είναι μια σειρά από αριθμούς (αν έχουμε 8 διευθύνσεις, θα είναι από 0 έως 7, όπως φαίνεται στο Σχ.6.2). Δίνεται παρακάτω, στο Σχήμα 6.3, παράδειγμα της διαδικασίας. Κάποιοι γνωστοί κωδικοποιητές Chain Code είναι οι Three orthogonal symbol chain code (3OT), Freeman Chain Code of Eight Direction (FCCE), Directional Freeman Chain Code of Eight Directions (DFCCE) και Vertex Chain Code (VCC).

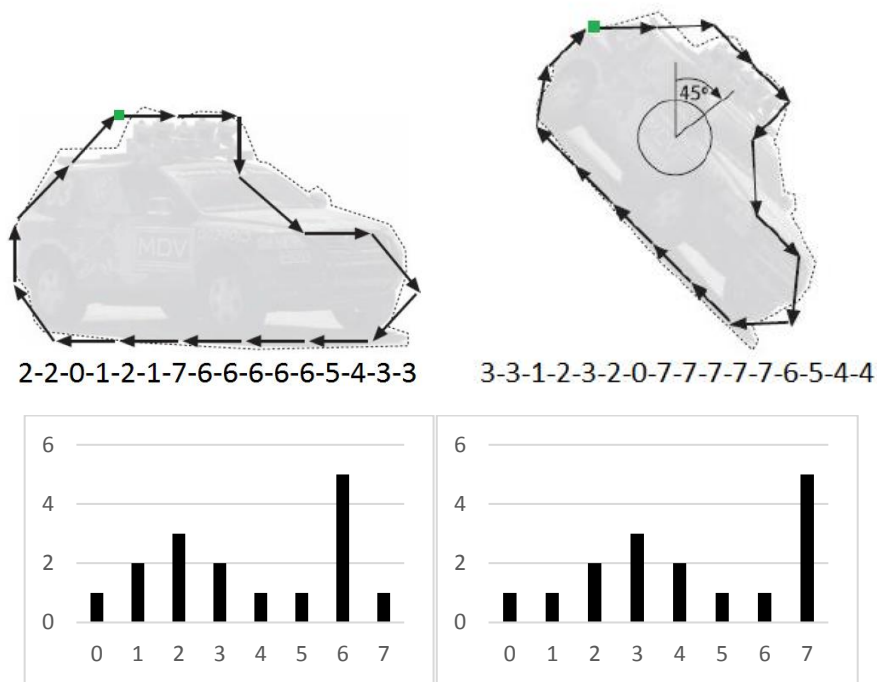


Σχήμα 6.3: Παράδειγμα της διαδικασίας Freeman Chain Code

Η κωδικοποίηση έχει αρκετές εφαρμογές, με την πλέον σημαντική εφαρμογή να είναι η δημιουργία CCH (Chain Code Histogram). Πρόκειται, για ένα ιστόγραμμα το οποίο



δημιουργείται από την καταμέτρηση των μεταβάσεων που έχουν ίδια κατεύθυνση στη διαδικασία του Chain Code. Αυτό το ιστόγραμμα έχει αρκετές σημαντικές ιδιότητες με την πιο ιδιαίτερη να είναι η αμεταβλητότητα που φέρει τις περιστροφές της περιοχής σε πολλαπλάσιες των 45° γωνίες. Μια περιστροφή 45° μετατοπίζει κυκλικά τις τιμές του ιστογράμματος. Αυτό μπορούμε να το παρατηρήσουμε στο σχήμα 6.4 Αυτή η ιδιότητα είναι αρκετά σημαντική, διότι σε τέτοιες περιστροφές δεν επηρεάζεται καθόλου η αναγνώριση αντικειμένων.



Σχήμα 6.4: Αναπαράσταση περιγραμμάτων και των ιστογραμμάτων τους.

6.2 Γεωμετρικές ροπές (Image Moments)

Τα αντικείμενα εμπεριέχουν ένα σύνολο από στατιστικά χωρικά χαρακτηριστικά, τα οποία είναι οι γεωμετρικές ροπές. Αυτά τα χαρακτηριστικά χρησιμεύουν σε αρκετές εφαρμογές ανάλυσης και ανάκτησης εικόνων.

Οι γεωμετρικές ροπές ενός διδιάστατης εικόνας $f(x,y)$ και δίνονται από τον τύπο:



$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy$$

Για να μη δημιουργηθεί ένα περιορισμένο σύνολο ροπών, τα p και q παίρνουν μόνο θετικές τιμές. Για την καλύτερη περιγραφή της πρότυπης εικόνας $f(x, y)$ πρέπει να υπάρχουν όσο περισσότερες ροπές γίνεται.

Οι γεωμετρικές ροπές μιας διακριτής εικόνας δίνονται από τον τύπο:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y)$$

6.3 Είδη γεωμετρικών ροπών

Υπάρχουν 3 είδη γεωμετρικών ροπών.

- i. Οι κεντρικές ροπές. Αυτές οι ροπές είναι ανεξάρτητες από τη μετατόπιση του αντικειμένου. Ορίζονται από τον τύπο:

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

- ii. Οι κανονικοποιημένες κεντρικές ροπές. Αυτές οι ροπές είναι ανεξάρτητες και από την μετατόπιση αλλά και από το μέγεθος του αντικειμένου. Ορίζονται από τον τύπο:

$$n_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, \quad \gamma = \frac{p+q}{2} + 1, \quad p+q = 2, 3, \dots$$



- iii. Οι αναλλοίωτες ροπές H_u . Αυτές οι ροπές είναι ανεξάρτητες από τη μετατόπιση, το μέγεθος και τη στροφή του αντικειμένου. Ορίζονται από τους τύπους:

$$\Phi_1 = n_{20} + n_{02}$$

$$\Phi_2 = (n_{20} - n_{02})^2 + 4n_{11}^2$$

6.4 Εξαγωγή χαρακτηριστικών με βάση τις ροπές

Οι γεωμετρικές ροπές, αν και δεν διαθέτουν κάποιο γεωμετρικό νόημα που να είναι άμεσα κατανοητό, χρησιμοποιούνται για να υπολογιστούν πολλές γεωμετρικές παράμετροι. Έτσι μπορεί να ανακατασκευαστεί κάποιο αντικείμενο το οποίο το περιγράφουν οι ροπές.

Για την εξαγωγή των χαρακτηριστικών, χρησιμοποιούνται ροπές μέχρι 2^{ης} τάξης. Αυτό γίνεται για να μην υπερβούν την πολυπλοκότητα που υπάρχει σε μια έλλειψη.

Επιφάνεια Α

Στη περίπτωση που έχουμε μια δυαδική εικόνα και μια συνάρτηση έντασης φωτεινότητας $f(x,y)$, η οποία παίρνει για το φόντο του αντικείμενο μηδενικές τιμές και για τα σημεία μέσα στην περίμετρο τιμή ένα, τότε υπάρχει αντιστοιχία μεταξύ της επιφάνειας του αντικειμένου με τη ροπή μηδενικής τάξης.

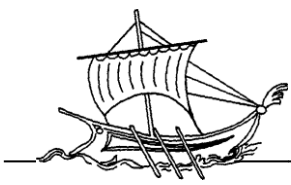
$$A = m_{00}$$

Κέντρο Βάρους

Η σχέση του κέντρου βάρους ενός αντικειμένου είναι η παρακάτω:

$$\bar{x} = \frac{m_{10}}{A} = \frac{m_{10}}{m_{00}}$$

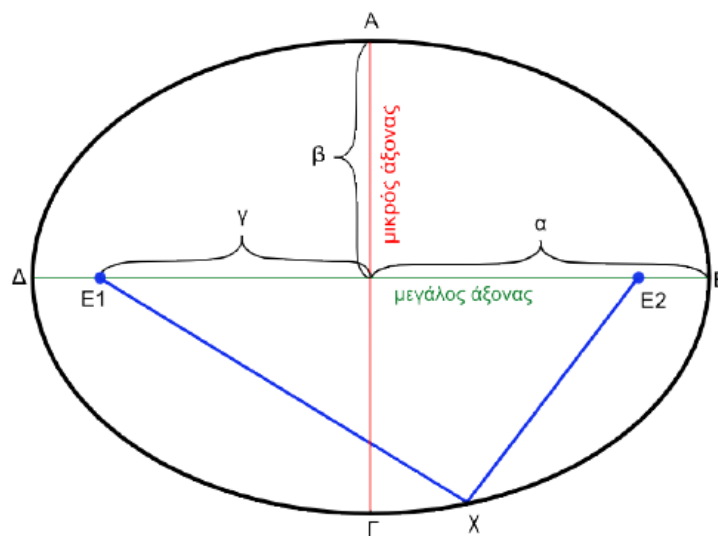
$$\bar{y} = \frac{m_{01}}{A} = \frac{m_{01}}{m_{00}}$$



Άξονες συμμετρίας έλλειψης

Σε μια έλλειψη υπάρχουν δύο άξονες συμμετρίας. Ουσιαστικά αυτές είναι η μεγαλύτερη διάμετρος της έλλειψης και η μικρότερη. Αυτές χρησιμοποιούνται στην προσέγγιση του αντικειμένου που περιγράφεται. Παρακάτω φαίνεται ο τύπος τους και η σχηματική αναπαράστασή τους.

$$\alpha_{1,2} = \sqrt{\frac{2 \cdot (\mu_{20} + \mu_{02}) \pm \sqrt{4 \cdot \mu_{11}^2 + (\mu_{20} - \mu_{02})^2}}{m_{00}}}$$



Σχήμα 6.5 Σχήμα έλλειψης και τα βασικά στοιχεία της

Όπου:

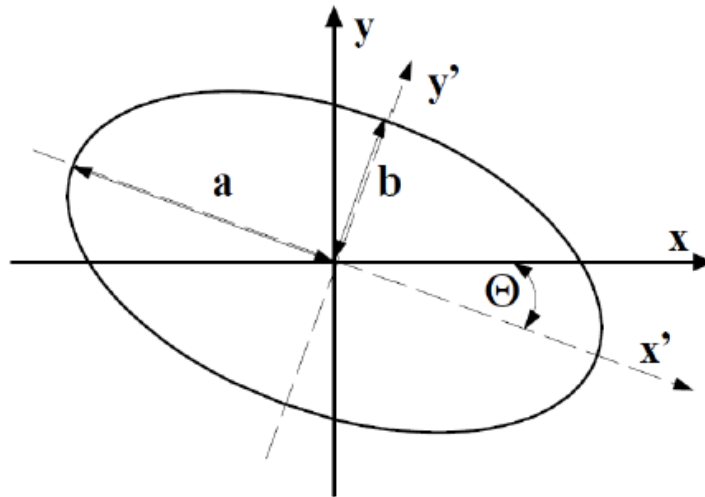
- E1-E2 οι εστίες της έλλειψης
- α ο μεγαλύτερος ημιάξονας
- β ο μικρότερος ημιάξονας
- γ η απόσταση της εστίας
- ΔΒ ο μεγαλύτερος άξονας
- ΑΓ ο μικρότερος άξονας



Προσανατολισμός Θ

Η γωνία μεταξύ του άξονα x σε ένα αντικείμενο και του μεγαλύτερου άξονα είναι ο προσανατολισμός Θ . Παρακάτω φαίνεται ο τύπος του και η σχηματική αναπαράστασή του.

$$\Theta = \frac{1}{2} \cdot \arctan \frac{2\mu_{11}}{\mu_{20} - \mu_{02}}$$



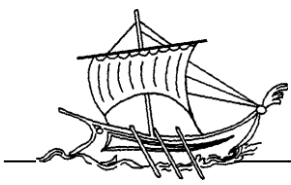
Σχήμα 6.6 Προσανατολισμός Θ

Εκκεντρότητα ϵ

Ο τύπος της εκκεντρότητας είναι ο παρακάτω:

$$\epsilon = \frac{\sqrt{\alpha_1^2 - \alpha_2^2}}{\alpha_1}$$

Η εκκεντρότητα συμβολίζει το «πάχος» της έλλειψης. Δηλαδή αν $\epsilon=0$ τότε η έλλειψη είναι κύκλος ενώ στην περίπτωση που $\epsilon=1$ τότε η έλλειψη είναι μια ευθεία γραμμή.



Επιφάνεια της έλλειψης

Η επιφάνεια της έλλειψης, η οποία περιγράφει ένα αντικείμενο καλύτερα δίνεται από τον τύπο:

$$A=\pi\alpha\beta$$

6.5 Γεωμετρικά χαρακτηριστικά

Η γεωμετρική μορφή ενός αντικειμένου μας δίνει τα γεωμετρικά του χαρακτηριστικά. Αυτά τα χαρακτηριστικά αναλύονται παρακάτω.

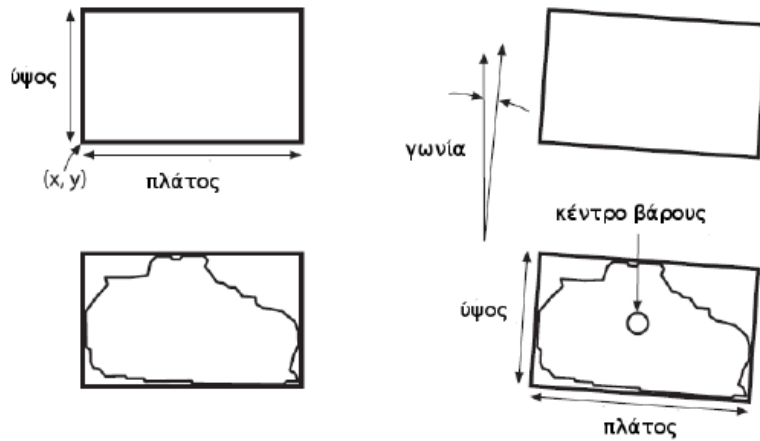
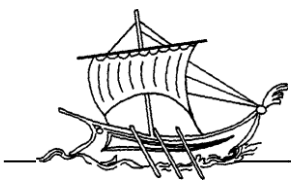
Περίμετρος

Η περίμετρος είναι το μήκος του περιγράμματος μια εικόνας. Στις ψηφιακές εικόνες η περίμετρος είναι ένα σύνολο από pixels τα οποία βρίσκονται στην περίμετρό της.

Περιγεγραμμένα Ορθογώνια

Τα περιγεγραμμένα ορθογώνια είναι σχηματικά πρότυπα, τα οποία βοηθούν στην περιγραφή και στην προσέγγιση των αντικειμένων, έτσι ώστε να εμφανιστούν περισσότερα στοιχεία τους.

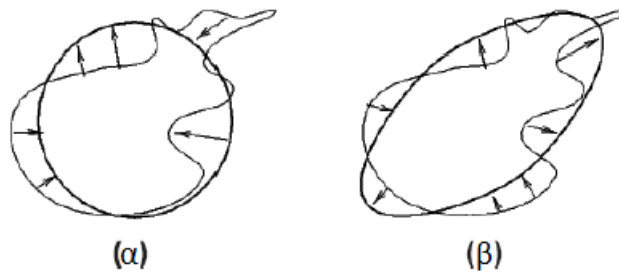
Παρακάτω φαίνονται κάποια από τα χαρακτηριστικά ενός περιγεγραμμένου ορθογώνιου. Αυτά είναι το πλάτος, το ύψος, το κέντρο βάρους και ο προσανατολισμός. Από αυτά τα χαρακτηριστικά προκύπτουν η αναλογία του ύψους προς το πλάτος, η περίμετρος του ορθογώνιου, η επιφάνεια του, η σχέση μεταξύ της επιφάνειας του ορθογώνιου με την επιφάνεια της εικόνας κ.α.



Σχήμα 6.7 Αριστερά το κατακόρυφο Π.Ο. και δεξιά το ελάχιστο Π.Ο.

Κυκλική Διακύμανση και Ελλειπτική Διακύμανση

Τα ιδανικότερα σχήματα για τη σύγκριση των αντικειμένων είναι η έλλειψη και ο κύκλος. Η κυκλική διακύμανση αναλογεί στο μέσο τετραγωνικό σφάλμα της σύγκρισης της εικόνας με τον κύκλο. Αν το αντικείμενο είναι ένας τέλειος κύκλος τότε η κυκλική διακύμανση θα ισούται με το μηδέν. Όσο πιο περίπλοκο το αντικείμενο τόσο αυξάνεται και η τιμή της. Τα αντίστοιχα ισχύουν και για την ελλειπτική διακύμανση.



Σχήμα 6.8 α) Κυκλική Διακύμανση β) Ελλειπτική Διακύμανση

MAX και MIN Ακτίνα

Ως ακτίνα R ορίζουμε την απόσταση του κέντρου βάρους του αντικειμένου από την περίμετρο του. R_{max} και R_{min} είναι η μεγαλύτερη και η μικρότερη απόσταση.



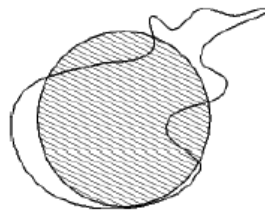
Καμπυλότητα και συντελεστές μορφής

Η σχέση της καμπυλότητας του αντικειμένου, είναι:

$$C = \frac{4\pi A}{P^2}$$

όπου A η επιφάνεια και P η περίμετρος του αντικειμένου.

Παρατηρούμε ότι για $C=1$ έχουμε ένα κυκλικό αντικείμενο και για $C=\pi/4$ έχουμε ένα τετράγωνο αντικείμενο.



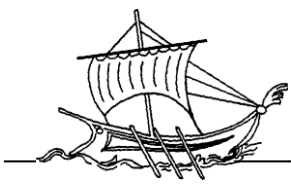
Σχήμα 6.9 Καμπυλότητα

Κυρτότητα

Η κυρτότητα είναι ο λόγος του μήκους του κυρτού περιγράμματος (το περίγραμμα που περιβάλλει την εικόνα, το οποίο έχει το μικρότερο μήκος) προς το μήκος περιμέτρου της εικόνας.



Σχήμα 6.10 Κυρτότητα



7. ΣΥΜΠΕΡΑΣΜΑΤΑ

Οι συνεχείς εξελίξεις στην τεχνολογία των υπολογιστών τις τελευταίες δεκαετίες σε συνδυασμό με τις αυξανόμενες απαιτήσεις είχαν ως αποτέλεσμα την εφαρμογή των νέων τεχνολογιών στην επεξεργασία εικόνας. Σε αυτό το πλαίσιο αναπτύχθηκαν μια σειρά τεχνικών και μεθόδων. Επιπρόσθετα, αναπτύχθηκε ένα ειδικός κλάδος λογισμικού, που υλοποιεί τις προηγούμενες τεχνικές και μεθόδους.

Στην παρούσα εργασία μελετήθηκαν μέθοδοι και τεχνικές που σχετίζονται με τη Μηχανική Όραση. Περιγράφηκαν τεχνικές όπως η ανίχνευση ακμών μέσω αλγορίθμου, ο οποίος προέκυψε πως είναι αρκετά αργός, καθώς επίσης, και ο θόρυβος που προκύπτει επηρεάζει την απόδοση του, σημαντικά.

Στη συνέχεια αναλύθηκαν τα χρωματικά μοντέλα RGB, HIS, HSV, YCbCr και CieLab, τα οποία χρησιμεύουν στην ανάλυση των εικόνων, προσδιορίζοντας πολύ καλύτερα τα χαρακτηριστικά τους.

Αναλύθηκαν οι σημαντικότεροι τελεστές της μαθηματικής μορφολογίας, που χρησιμοποιούνται για τη μείωση ή εξάλειψη του θορύβου, και συνεισφέρουν σε μεγάλο βαθμό στην επιτυχή κατάτμηση.

Στη συνέχεια αναλύθηκε η τεχνική τμηματοποίησης και κατάτμησης εικόνας μέσω διάφορων αλγορίθμων. Οι αλγόριθμοι κατάτμησης προσφέρουν πολλές δυνατότητες στην εύρεση και στον εντοπισμό αντικειμένου. Χρονικά και ποιοτικά παρατηρήθηκε πως είναι καλύτερος ο αλγόριθμος SURF, ο οποίος σε μελλοντική χρήση παρέχει τη δυνατότητα ανίχνευσης αντικειμένου σε πραγματικό χρόνο. Τέλος προσδιορίστηκαν μέθοδοι για την περιγραφή σχήματος, οι οποίοι βοηθούν στο βέλτιστο εντοπισμό του γεωμετρικού μεγέθους ενός αντικειμένου.

Με τη συνεχή αυξανόμενη βελτίωση της τεχνολογίας η Μηχανική Όραση θα αποτελέσει βασικό παράγοντα της επεξεργασίας εικόνας και οι μέθοδοι, καθώς και οι τεχνικές, αλλά κυρίως οι αλγόριθμοι που χρησιμοποιούνται απαιτούν συνεχή βελτίωση. Οι νέες απαιτήσεις των αναγκών των χρηστών, θα οδηγήσουν σε εξέλιξη νέων τεχνολογιών.



8. ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Ι. Πήτας, *Ψηφιακή Επεξεργασία Εικόνας*, Θεσσαλονίκη 2001.
- [2] John Canny, *A computational approach to edge detection. Pattern Analysis and Machine Intelligence*, IEEE Transactions on, PAMI-8(6):679–698, Nov. 1986.
- [3] [www.m3.tuc.gr/adcnano site/paradotea-pkg3.html](http://www.m3.tuc.gr/adcnano%20site/paradotea-pkg3.html) , προσπελάστηκε στις 9/6/2016.
- [4] https://arch.icte.uowm.gr/mdasyg/pdfs/object_quality_assurance_with_computer_vision_loizou_dasygenis.pdf, προσπελάστηκε στις 15/8/2016
- [5] <https://www.mathworks.com/matlabcentral/mlc-downloads/downloads/submissions/22212/versions/1/previews/myLabel.m/index.html>
προσπελάστηκε στις 26/8/2016
- [6] https://en.wikipedia.org/wiki/Machine_vision, προσπελάστηκε στις 5/3/2016