



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ

ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ – ΕΡΓΑΣΤΗΡΙΟ ΤΗΛΕΠΙΣΚΟΠΗΣΗΣ

**ΑΝΑΠΤΥΞΗ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΓΙΑ ΤΟΝ ΑΥΤΟΜΑΤΟ
ΕΝΤΟΠΙΣΜΟ ΣΤΑΘΜΕΥΜΕΝΩΝ ΑΕΡΟΣΚΑΦΩΝ ΣΕ ΔΙΑΧΡΟΝΙΚΑ
ΔΟΡΥΦΟΡΙΚΑ ΔΕΔΟΜΕΝΑ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Δημήτριος Α. Πούλος

Επιβλέπων Καθηγητής: Καράντζαλος Κωνσταντίνος

Λέκτορας Ε.Μ.Π

Αθήνα, Οκτώβριος 2013



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ

ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ – ΕΡΓΑΣΤΗΡΙΟ ΤΗΛΕΠΙΣΚΟΠΗΣΗΣ

**ΑΝΑΠΤΥΞΗ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΓΙΑ ΤΟΝ ΑΥΤΟΜΑΤΟ
ΕΝΤΟΠΙΣΜΟ ΣΤΑΘΜΕΥΜΕΝΩΝ ΑΕΡΟΣΚΑΦΩΝ ΣΕ ΔΙΑΧΡΟΝΙΚΑ
ΔΟΡΥΦΟΡΙΚΑ ΔΕΔΟΜΕΝΑ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Δημήτριος Α. Πούλος

Επιβλέπων Καθηγητής: Καράντζαλος Κωνσταντίνος

Λέκτορας Ε.Μ.Π

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 2013

.....
.....

Κ. Καράντζαλος

Λέκτορας Ε.Μ.Π

Δ. Αργιαλάς

Καθηγητής Ε.Μ.Π

Α. Γεωργόπουλος

Καθηγητής Ε.Μ.Π

Αθήνα, Οκτώβριος 2013

Δημήτριος Α. Πούλος

Διπλωματούχος Αγρονόμος και Τοπογράφος Μηχανικός Ε.Μ.Π.

Copyright © Δημήτριος Α. Πούλος, 2013

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς το συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά, τον επιβλέποντα καθηγητή μου, Κωνσταντίνο Καράντζαλο, για την πολύτιμη βοήθεια και συμπαράστασή του σε όλη την διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας καθώς επίσης και όλους τους καθηγητές, υποψήφιους διδάκτορες, συμφοιτητές και συναδέλφους που με τις διορθώσεις στάθηκαν πολύτιμοι αρωγοί στην ολοκλήρωση της διπλωματικής μου. Ένα μεγάλο ευχαριστώ οφείλω να πω στην οικογένειά μου και ειδικά στη γιαγιά μου, που με υποστήριξε με κάθε τρόπο, όλα αυτά τα 5 εξάμηνα των σπουδών μου στο Ε.Μ.Π.

Τέλος, θα ήθελα να ευχαριστήσω την Γεωγραφική Υπηρεσία Στρατού, που μου έδωσε την δυνατότητα να αποκτήσω νέες και πολύ σημαντικές γνώσεις που μου παρέχουν μεγαλύτερη σιγουριά για την μελλοντική σταδιοδρομία στον χώρο του Γεωγραφικού Σώματος.

Δημήτριος Αν. Πούλος

Οκτώβριος 2013

Περιεχόμενα:

Πίνακας Αγγλικών Όρων	9
1. Εισαγωγή.....	11
1.1. Ανίχνευση στρατιωτικών στόχων.....	11
1.2. Αντικείμενο και Στόχοι της Εργασίας	15
1.3. Κίνητρο και Συνεισφορά της Εργασίας	16
1.4. Δομή Εργασίας	16
2. Ανασκόπηση Βιβλιογραφίας	18
2.1 Ανασκόπηση Γενικής Βιβλιογραφίας	18
2.2. Ανασκόπηση Ειδικότερης Βιβλιογραφίας	20
3. Ανάπτυξη αλγορίθμων για τον αυτόματο εντοπισμό αεροσκαφών σε διαχρονικά δορυφορικά δεδομένα.	24
3.1. Ανάπτυξη αλγορίθμου με βάση τεχνικές αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας.	24
3.1.1. Προσδοκώμενα αποτελέσματα.....	24
3.1.2. Αλγοριθμικά επιμέρους βήματα.....	24
3.1.3.1. Αντιστοίχιση εικόνων (Registration).....	26
3.1.3.2. Αποκατάσταση και διόρθωση των διαστάσεων.....	30
3.1.3.3. Ραδιομετρική διόρθωση εικόνων.....	30
3.1.3.4. Ομαλοποίηση εικόνων.....	32
3.1.3.5. Αναζήτηση εντοπισμών σε όλους τους δυνατούς συνδυασμούς.....	34
3.1.3.6. Αφαίρεση & Κατωφλίωση εικόνων	35
3.1.3.7. Μαθηματική μορφολογία και ανάλυση ιδιοτήτων.....	36
3.1.3.8. Υπολογισμοί MAX (μέγιστο), MIN (ελάχιστο), M.O (Μέσου Όρου) για κάθε κανάλι και για κάθε οντότητα.....	40
3.1.3.9. Δημιουργία κατωφλίων με βάση τους κανονικοποιημένους M.O για την εξαγωγή των επιθυμητών στόχων.....	43
3.2. Ανάπτυξη αλγορίθμου με βάση τεχνικές αναγνώρισης προτύπων και διανυσμάτων υποστήριξης.....	45

3.2.1.	Εισαγωγή και Θεωρητικό Υπόβαθρο.....	45
3.2.2.	Μηχανές διανυσματικής Υποστήριξης	45
3.2.3.	Μετασχηματισμός Fourier	62
3.2.4.	Συνέλιξη (Convolution) – Συσχέτιση (Correlation).....	65
3.2.5.	Φίλτρα Gabor	68
3.2.6.	Ανθρώπινο Σύστημα Όρασης	69
3.2.7.	Συναρτήσεις Gabor	70
3.2.8.	Δημιουργία Φίλτρων Gabor	72
3.2.9.	Αναλυτική Περιγραφή Αναπτυγμένου Αλγορίθμου και Συναρτήσεων .	76
3.3.	Script Αρχεία	113
4.	Πειραματικά αποτελέσματα, Ποσοτική και Ποιοτική Αξιολόγηση	115
4.1.	Αξιολόγηση με βάση τεχνικές αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας.....	115
	Παρατηρήσεις	122
4.2.	Αξιολόγηση με βάση τεχνικές αναγνώρισης και διανύσματα υποστήριξης.....	125
	Παρατηρήσεις	130
5.	Συμπεράσματα και Προτάσεις για περαιτέρω έρευνα	133
5.1.	Γενικά Συμπεράσματα.....	133
5.2.	Ειδικότερα Συμπεράσματα	134
5.3.	Προτάσεις	134
6.	Βιβλιογραφία.....	136

Πίνακας Αγγλικών Όρων

A/A	Αγγλικός Όρος	Ελληνικός Όρος
1	Artificial Intelligence	Τεχνητή Νοημοσύνη
2	Autocorrelation	Αυτοσυσχέτιση
3	Bias - Threshold	Κατώφλι
4	Binary	Δυαδικός
5	Brightness	Φωτεινότητα
6	Classification	Ταξινόμηση
7	Classifier	Ταξινομητής
8	Completeness	Πληρότητα
9	Computer Vision	Όραση Υπολογιστών
10	Contrast	Αντίθεση
11	Convolution	Συνέλιξη
12	Correctness	Ορθότητα
13	Database	Βάση Δεδομένων
14	Descriptors	Περιγραφίτες
15	DFT (Discrete Fourier Transform)	Διακριτός Μετασχηματισμός Fourier
16	Doughnut shapes	Σχήμα Κουλουριού
17	Grayscale - BW	Ασπρόμαυρος
18	Holes	Τρύπες
19	Illumination	Φωτισμός
20	Image Registration	Αντιστοίχιση εικόνων
21	Inverse	Αντίστροφος
22	Linear	Γραμμικός
23	Margin	Περιθώριο
24	Orientation	Διεύθυνση
25	Patch	Μπάλωμα
26	Pattern	Πρότυπο
27	Perceptron	Είδος Νευρώνα
28	Pixel	Εικονοστοιχείο
29	RGB (Red – Green – Blue)	Κόκκινο πράσινο Μπλε
30	SVM (Support Vector Machine)	Μηχανές Υποστήριξης Διανυσμάτων
31	Tie points	Σημεία σύνδεσης
32	Training	Εκπαίδευση
33	Vector	Διάνυσμα
34	Wavelets	Κυματοπακέτα

1. Εισαγωγή

Στο εισαγωγικό Κεφάλαιο 1, παρατίθενται γενικά στοιχεία για τον εντοπισμό στρατιωτικών στόχων και ακολούθως περιγράφονται οι αντικειμενικοί στόχοι, το κίνητρο και η συνεισφορά της εργασίας. Τέλος, πραγματοποιείται μια συνοπτική περιγραφή της δομής και της διάρθρωσης των κεφαλαίων της εργασίας.

1.1. Ανίχνευση στρατιωτικών στόχων

Πρόσφατες επιχειρήσεις σε εμπόλεμες ζώνες διαφόρων περιοχών του κόσμου έχουν καταδείξει την ανάγκη για αυτόματο, σε πραγματικό χρόνο και με κατάλληλη χωρική ακρίβεια, εντοπισμό στρατιωτικών στόχων. Για παράδειγμα, το 1999 στο Κόσσοβο, οι Νατοϊκές δυνάμεις είχαν εντοπίσει και βομβαρδίσει φάλαγγα προσφύγων. Η φάλαγγα αποτελούνταν από Αλβανούς του Κοσσόβου οι οποίοι προσπαθούσαν να επιστρέψουν στα σπίτια τους. Όχημα της φάλαγγας αναγνωρίστηκε ως στρατιωτικός στόχος με την διαφορά ότι ήταν ένα απλό τρακτέρ. Απολογισμός του λανθασμένου αυτού εντοπισμού ήταν 87 νεκροί και 60 τραυματίες εκ των οποίων πολλά παιδιά.



Εικόνα 1 – Γελοιογραφία της εποχής για τον λανθασμένο βομβαρδισμό φάλαγγας πολιτών στην Κορσα του Κοσσόβου.

Επομένως, υπάρχει ανάγκη για πιο γρήγορα και πιο «έξυπνα» συστήματα αναγνώρισης και εντοπισμού στρατιωτικών στόχων.

Ο όρος του «στρατιωτικού στόχου» χρησιμοποιείται συχνά για να περιγράψει το γενικό σχέδιο μιας δεδομένης αποστολής. Για παράδειγμα, την κατάληψη ενός υψώματος, της καταστροφής μιας γέφυρας, την διάβαση ενός ποταμού κ.ά. Υπό την στενότερη έννοια, ένας στρατιωτικός στόχος μπορεί να αναφέρεται σε έναν συγκεκριμένο στόχο για την ουδετεροποίηση ή την καταστροφή του.











Οι νόμοι του Πολέμου χρησιμοποιούν τον όρο κυρίως με την δεύτερη έννοια, για να προσδιορίσουν τοποθεσίες, οχήματα, εξοπλισμούς, εγκαταστάσεις ή ακόμα και εχθρικό προσωπικό το οποίο, κάτω από ορισμένες συνθήκες, αποτελεί έναν νόμιμο στρατιωτικό στόχο. Προφανώς και ορισμένα αντικείμενα ή άνθρωποι αποτελούν παράνομους στόχους και οποιαδήποτε προσβολή τους ισοδυναμεί με την διάπραξη εγκλήματος πολέμου. Για παράδειγμα, η άμεση επίθεση σε άμαχο πληθυσμό, σε αντικείμενα που χρησιμοποιούνται για ανθρωπιστικούς, θρησκευτικούς ή πολιτιστικούς λόγους, όπως νοσοκομεία, σχολεία, εκκλησίες, μουσουλμανικά τεμνία και μουσεία, αποτελούν μερικά από τα χαρακτηριστικότερα παραδείγματα μη στρατιωτικών στόχων.

Στον επόμενο πίνακα παρατίθενται μερικοί από τους χαρακτηριστικότερους στρατιωτικούς στόχους.

ΣΤΡΑΤΙΩΤΙΚΟΙ ΣΤΟΧΟΙ								
A/A	Κινούμενοι - Σταθεροί	Κατηγορία	Φωτογραφία	Ενδεικτικός Τύπος	Τυπικές Διαστάσεις Μήκος(m)	Τυπικές Διαστάσεις Πλάτος(m)	Τυπικές Διαστάσεις Ύψος(m)	Παρ/σεις
 ΣΤΡΑΤΟΣ ΞΗΡΑΣ								
1	Κινούμενοι	Άρματα		Leo2 A6	9,970	3,759	3,000	-
2	Κινούμενοι	Άρματα Περισυλλογής		Buffel (BPz3)	9,070	3,540	2,735	-
3	Κινούμενοι	Αυτοκινούμενα Πυροβόλα		Pantzerhaubitze 2000	11.7	3.6	3.1	-
4	Κινούμενοι	Ρυμουλκούμενα Πυροβόλα		M777	10,700	-	-	-
5	Κινούμενοι	Συστήματα Πολλαπλών Εκτοξευτών Πυραύλων		M270 MLRS	6,850	2,970	2,590	-
6	Κινούμενοι	UAVs		RQ-1/MQ-1 Predator	8,220	14,800	2,100	-

A/A	Κινούμενοι - Σταθεροί	Κατηγορία	Φωτογραφία	Ενδεικτικός Τύπος	Τυπικές Διαστάσεις Μήκος(m)	Τυπικές Διαστάσεις Πλάτος(m)	Τυπικές Διαστάσεις Ύψος(m)	Παρ/σεις
7	Κινούμενοι	Τεθωρακισμένο Οχημα Μάχης (ΤΟΜΑ)		Marder (IV)	6,790	3,240	2,980	-
8	Κινούμενοι	Τεθωρακισμένο Οχημα Μεταφοράς Προσωπικού (ΤΟΜΠ)		M113	4,863	2,686	2,500	-
9	Κινούμενοι	Οχήματα Γενικής Χρήσεως		M/S 240 GD	4,010	1,700	2,100	-
10	Κινούμενοι	Αρματοφορείς		M1070 tractor+M1000 trailer	24,990	3,700	3,680	-
11	Κινούμενοι	Ελικόπτερα Μεταφοράς Προσωπικού		Boeing CH-47 Chinook	30,180	4,800	5,490	*Ακτίνα κύριου στροφείου
12	Κινούμενοι	Επιθετικά Ελικόπτερα		Boeing AH-64 Apache	17,700	4,900	3,870	*Ακτίνα κύριου στροφείου
13	Κινούμενοι	Γεφυροφόρα		Biber	11,820	4,000	3,550	-
14	Κινούμενοι	Τροποποιημένα άρματα άρσης ναρκοπεδίων		Leo C2 Mine plow	9,970	3,759	3,000	-
 ΠΟΛΕΜΙΚΗ ΑΕΡΟΠΟΡΙΑ								
15	Κινούμενοι	Αεροπλάνα (Μαχητικά)		F-16 Fighting Falcon	14,800	9,400	4,800	* Έκταση φτερών
16	Κινούμενοι	Ελικόπτερα Αεροπορίας		Eurocopter AS 332 Super Puma	16,300	15,080	4,600	*Διάμετρος κύριου στροφείου
17	Κινούμενοι	Αεροπλάνα (Μεταγωγικά)		Lockheed C- 130 Hercules	29,790	40,410	11,840	* Έκταση φτερών
18	Κινούμενοι	Αεροπλάνα (Επιτήρησης- Ηλεκτρονικού Πολέμου)		Boeing E-3 Sentry	46,610	44,420	12,730	* Έκταση φτερών
19	Κινούμενοι	Αεροπλάνα (Ανεφοδιασμού)		Boeing KC-46	48,500	47,600	15,800	* Έκταση φτερών
20	Κινούμενοι	Αντιαεροπορικά		MIM-23 Hawk	5,080	-	-	-
 ΠΟΛΕΜΙΚΟ ΝΑΥΤΙΚΟ								
21	Κινούμενοι	Φρεγάτες		MEKO 200HN	117,000	14,800	4,100	*Βύθισμα
22	Κινούμενοι	Αντιπτορπικά		Arleigh Burke- class destroyer	154,000	20,000	9,300	*Βύθισμα
23	Κινούμενοι	Πυραυλάκατοι		Super Vita	62,000	9,500	2,600	*Βύθισμα
24	Κινούμενοι	Υποβρύχια		Type 214	65,300	6,300	13,000	*Βύθισμα

A/A	Κινούμενοι - Σταθεροί	Κατηγορία	Φωτογραφία	Ενδεικτικός Τύπος	Τυπικές Διαστάσεις Μήκος(m)	Τυπικές Διαστάσεις Πλάτος(m)	Τυπικές Διαστάσεις Ύψος(m)	Πα/σεις
25	Κινούμενοι	Κορβέτες		B-class corvette	80,520	10,200	3,420	*Βύθισμα
26	Κινούμενοι	Αποβατικά		Klasse 520	39,900	8,800	1,800	*Βύθισμα
27	Κινούμενοι	Γενικής Υποστήριξης / Ναρκαλιευτικά		Hunt	60,300	10,400	3,200	*Βύθισμα
28	Κινούμενοι	Κανονιοφόροι		Osprey HSY- 56A	56,500	10,000	2,550	*Βύθισμα
29	Κινούμενοι	Αρματαγωγά		L-173	116,000	15,300	4,600	*Βύθισμα
30	Κινούμενοι	Ταχείας Μεταφοράς		Zubr	57,300	25,600	21,900	*Βύθισμα
31	Κινούμενοι	Περιπολικά Νέου Τύπου		YTKB-400	55,750	8,900	2,650	*Βύθισμα
32	Κινούμενοι	Αεροπλανοφόρα		USS John F. Kennedy (CV- 67)	321,000	77,000	58,520	*Βύθισμα
33	Κινούμενοι	Ελικοφόρα		HMS Ocean	203,400	35,000	6,500	*Βύθισμα
34	Κινούμενοι	Ελικόπτερα Ναυτικού		S-70 Aegean Hawk	15,260	1,750	3,790	-
35	Κινούμενοι	Αεροπλάνα Ναυτικής Συνεργασίας		Lockheed P-3 Orion	25,600	30,400	10,300	* Έκταση φτερών
 ΣΤΡΑΤΟΣ ΞΗΡΑΣ								
36	Σταθεροί	Στρατόπεδα		-	-	-	-	-
37	Σταθεροί	Αποθήκη Πυρομαχικών (ΠΑΠ)		-	-	-	-	-
 ΠΟΛΕΜΙΚΗ ΑΕΡΟΠΟΡΙΑ								
38	Σταθεροί	Ραντάρ Αεράμυνας		-	-	-	-	-
39	Σταθεροί	Αεροδρόμια		-	-	-	-	-
40		Χώροι Προστασίας Αεροσκαφών		-	-	-	-	-
41	Σταθεροί	Βαλλιστικοί Πύραυλοι		-	-	-	-	-
 ΠΟΛΕΜΙΚΟ ΝΑΥΤΙΚΟ								
42	Σταθεροί	Ναύσταθμοι		-	-	-	-	-

A/A	Κινούμενοι - Σταθεροί	Κατηγορία	Φωτογραφία	Ενδεικτικός Τύπος	Τυπικές Διαστάσεις Μήκος(m)	Τυπικές Διαστάσεις Πλάτος(m)	Τυπικές Διαστάσεις Ύψος(m)	Παρ/σεις
 ΣΤΡΑΤΗΓΙΚΟΙ ΣΤΟΧΟΙ								
43	Σταθεροί	Αεροδρόμια		-	-	-	-	-
44	Σταθεροί	Λιμάνια		-	-	-	-	-
45	Σταθεροί	Γέφυρες		-	-	-	-	-
46	Σταθεροί	Διυλιστήρια		-	-	-	-	-
47	Σταθεροί	Φράγματα		-	-	-	-	-
48	Σταθεροί	Θερμοηλεκτρικό Εργοστάσιο		-	-	-	-	-
49	Σταθεροί	Σήραγγες		-	-	-	-	-
50	Σταθεροί	Βιομηχανικές Ζώνες		-	-	-	-	-
51	Σταθεροί	Πυρηνικός Σταθμός		-	-	-	-	-
52	Σταθεροί	Σιδηροδρομικό Δίκτυο		-	-	-	-	-

Πίνακας 1 – Κατηγορίες Στρατιωτικών Στόχων

1.2. Αντικείμενο και Στόχοι της Εργασίας

Βασικό αντικείμενο της παρούσας διπλωματικής εργασίας αποτέλεσε η διερεύνηση, ανάπτυξη και αξιολόγηση μεθοδολογίας για τον αυτόματο εντοπισμό στρατιωτικών στόχων. Έμφαση δόθηκε σε στόχους όπως μαχητικά αεροπλάνα, αεροδρόμια και ελικόπτερα. Επίσης, βασική επιδίωξη ήταν τα υψηλά επίπεδα τυποποίησης των παραμέτρων και των επιμέρους βημάτων στους αλγόριθμους που αναπτύχθηκαν. Θα πρέπει να τονιστεί η ιδιαίτερη προσοχή και επιμέλεια που δόθηκε στην εκτέλεση όλων των δυνατών δοκιμών στους αναπτυγμένους αλγόριθμους για τον αποτελεσματικό εντοπισμό των παραμέτρων, αλλά και του τρόπου με τον οποίο αυτοί επηρεάζουν την ποιότητα των αποτελεσμάτων.

-

1.3. Κίνητρο και Συνεισφορά της Εργασίας

Κίνητρο για τη διερεύνηση μεθοδολογιών αυτόματου εντοπισμού στρατιωτικών στόχων, αποτέλεσαν οι συνεχείς υπηρεσιακές ανάγκες των Ενόπλων Δυνάμεων για τη διατήρηση του υψηλού επιπέδου των στελεχών της, πάνω σε θέματα αυτόματου εντοπισμού στρατιωτικών στόχων. Δεν πρέπει να ξεχνάμε ότι πλέον στις μέρες μας η κύρια επιδίωξη δεν είναι τόσο η απόκτηση και συλλογή νέας πληροφορίας όσο η αξιοποίηση και ανάλυση της υπάρχουσας.

Η συνεισφορά της παρούσας εργασίας συνοψίζεται στα παρακάτω σημεία:

- Ανάπτυξη και αξιολόγηση αλγορίθμου για τον αυτόματο εντοπισμό σταθμευμένων αεροσκαφών σε στρατιωτικά αεροδρόμια από διαχρονικά δορυφορικά δεδομένα με βάση τεχνικές αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας.
- Ανάπτυξη και αξιολόγηση αλγορίθμου για τον αυτόματο εντοπισμό σταθμευμένων αεροσκαφών σε στρατιωτικά αεροδρόμια από διαχρονικά δορυφορικά δεδομένα με βάση τεχνικές αναγνώρισης προτύπων και διανυσμάτων υποστήριξης.

1.4. Δομή Εργασίας

Αρχικά, όπως προαναφέραμε στην εισαγωγή, έχουμε το **Κεφάλαιο 1**, που αποτελεί την εισαγωγή στο αντικείμενο, τον στόχο και την συνεισφορά της εργασίας. Στην συνέχεια ακολουθούν άλλα τέσσερα (4) κεφάλαια.

Στο **Κεφάλαιο 2** πραγματοποιείται μια ανασκόπηση της υπάρχουσας βιβλιογραφίας. Αρχικά, αναλύονται οι σημαντικότερες εργασίες που έχουν πραγματοποιηθεί γενικότερα στην ανίχνευση – εντοπισμό στρατιωτικών στόχων και στη συνέχεια αναλύονται οι εργασίες που συνδέονται με τους στρατιωτικούς στόχους της παρούσας εργασίας, δηλαδή τα σταθμευμένα στρατιωτικά αεροσκάφη.

Στη συνέχεια στο **Κεφάλαιο 3** έχουμε την ανάλυση των δύο μεθοδολογιών που αναπτύχθηκαν για τον αυτόματο εντοπισμό των αεροσκαφών. Η ανάλυση περιλαμβάνει αρχικά το θεωρητικό υπόβαθρο πίσω από την κάθε μεθοδολογία και ακολούθως την ανάλυση του κώδικα που χρησιμοποιήθηκε γραμμή – γραμμή.

Στο **Κεφάλαιο 4** παρουσιάζονται τα σημαντικότερα αποτελέσματα από τους εντοπισμούς που πραγματοποιήθηκαν και για τις δύο προαναφερθείσες μεθοδολογίες. Ακολουθεί η ποιοτική και ποσοτική αξιολόγησή με τους αντίστοιχους δείκτες ποιότητας και παρατίθενται τα κυριότερα προβλήματα τα οποία εντοπίστηκαν από ολόκληρη την διαδικασία των μεθοδολογιών που χρησιμοποιήθηκαν.

Τέλος, στο **Κεφάλαιο 5** περιλαμβάνονται τα τελικά συμπεράσματα από το σύνολο της διπλωματικής εργασίας, καθώς επίσης και προτάσεις για μελλοντική βελτίωση και επέκταση πάνω στο συγκεκριμένο αντικείμενο. Το τεύχος ολοκληρώνεται με την παράθεση της βιβλιογραφίας στο **Κεφάλαιο 6** που χρησιμοποιήθηκε για την σύνταξη της παρούσας εργασίας

2. Ανασκόπηση Βιβλιογραφίας

Στο Κεφάλαιο 2 πραγματοποιείται ανασκόπηση της σημαντικότερης βιβλιογραφίας για το αντικείμενο της συγκεκριμένης διπλωματικής εργασίας. Αρχικά, γίνεται μια ανασκόπηση των σημαντικότερων εργασιών που έχουν υλοποιηθεί με κοινό παρονομαστή το ευρύτερο αντικείμενο του εντοπισμού στρατιωτικών στόχων και στη συνέχεια ακολουθεί μια συνοπτική ανάλυση των εργασιών που έχουν στενή εφαρμογή με τις μεθοδολογίες που χρησιμοποιήθηκαν στην παρούσα διπλωματική.

2.1 Ανασκόπηση Γενικής Βιβλιογραφίας

Η αυτόματη ανίχνευση στόχων γενικά προβληματίζει τους επιστήμονες εδώ και δεκαετίες. Οι κυριότερες εφαρμογές που έχουν υλοποιηθεί τα τελευταία χρόνια με αρκετά ικανοποιητικά αποτελέσματα έχουν να κάνουν με τον εντοπισμό ναρκοπεδίων – ναρκών αλλά και κυρίως με αεροσκάφη. Το 1999 οι S.M. Khanna, F. Raquet, R. Apps and J.S. Seregelyi, από την Οττάβα του Καναδά [45] ανέπτυξαν μια νέα τηλεπισκοπική μέθοδο για τον εντοπισμό ναρκών χρησιμοποιώντας HPM Irradiation και IR Detection. Ακολούθησε ο Cheolha Pedro Lee από το Πανεπιστήμιο του Tennessee [44] ο οποίος δούλεψε πάνω στην ανίχνευση ναρκών με την χρήση πολλαπλών αισθητήρων. Τέλος, το 2003 ο B.H.P. Maathuis από την Ολλανδία [46], ασχολήθηκε με τον εντοπισμό ναρκοπεδίων στις διάφορες περιοχές της Ζιμπάμπουε χρησιμοποιώντας δορυφορικές εικόνες.

Αξιοσημείωτο είναι το γεγονός ότι η παγκόσμια ακαδημαϊκή κοινότητα δεν έπαψε ποτέ να προβληματίζεται πάνω στο θέμα του εντοπισμού – αναγνώρισης στρατιωτικών στόχων με την ευρύτερη έννοια. Συγκεκριμένα το 1997, οι L.J Denes, M. Gottlieb, B. Kaminsky στο Pittsburg των ΗΠΑ [51], δούλεψαν πάνω στην αναγνώριση στρατιωτικών στόχων. Η κεντρική ιδέα της εργασίας τους βασίστηκε στην κατασκευή μιας πλήρως ηλεκτρονικής φασματικής-πολωτικής (μετρητικής) φωτογραφικής μηχανής, η οποία λειτουργούσε με ρυθμό λήψης εικόνων 30/sec. Η φασματική ικανότητα της συγκεκριμένης κάμερας αλλά και η δυνατότητα ηλεκτρονικής πόλωσης της παρείχε αρκετά μεγάλη ευκολία στην επιλεκτικότητα συγκεκριμένων φασματικών υπογραφών – στόχων. (Διέθετε ικανότητα να πραγματοποιεί πόλωση τόσο γραμμικά όσο και ελλειπτικά).

Το 2000 στη γείτονα χώρα Τουρκία, ο A. Khashman [48], χρησιμοποίησε τον συνδυασμό νευρωνικών δικτύων και χωρικής ανάλυσης κλίμακας για να επιτύχει την αυτόματη επιλογή της βέλτιστης κλίμακας για ολόκληρη την εικόνα. Ακολούθως,

εφάρμοσε αλγορίθμους εντοπισμού ακμών για να επιτύχει τον εντοπισμό των στρατιωτικών στόχων.

Στην συνέχεια, οι Kailash Chandra Tiwari, Manoj K. Arora, Dharmendra Singh, Deepti Yadav το 2011 [54], παρουσίασαν την εργασία τους πάνω στον εντοπισμό στρατιωτικών στόχων σε υπερφασματικά δεδομένα με βάση τον εντοπισμό της αντίστοιχης φασματικής υπογραφής. Χρησιμοποιήθηκε Ανάλυση Ανεξαρτήτων Στοιχείων (ICA~Independent component analysis) η οποία έχει ως σκοπό να εντοπίσει τα τμήματα του φάσματος τα οποία είναι στατιστικά ανεξάρτητα(ή όσο το δυνατόν ανεξάρτητα μπορεί να είναι). Το πλεονέκτημα του ICA είναι το γεγονός ότι δεν χρειάζεται a priori πληροφορίες για τον στόχο που θέλουμε να εντοπίσουμε.

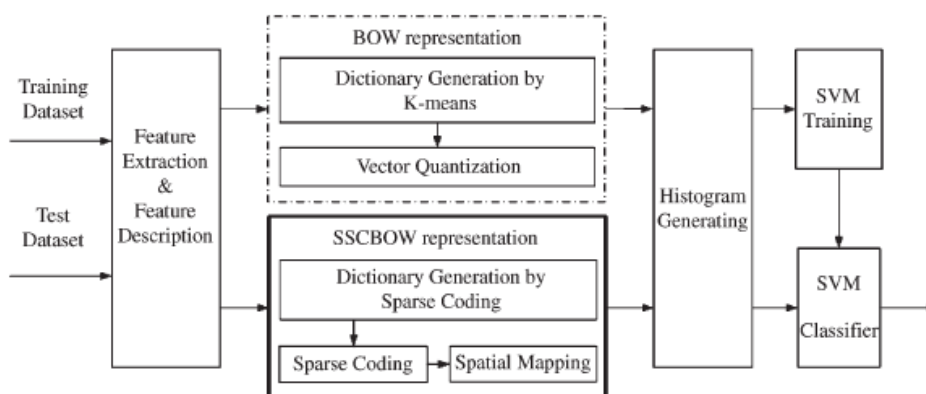
Οι υπάρχοντες αλγόριθμοι εντοπισμού πολλαπλών στόχων χρησιμοποιούσαν στατιστικά μοντέλα τα οποία θα μπορούσαν να διαχειριστούν γκαουσιανά δεδομένα αρκετά καλά. Στις πραγματικές υπερφασματικές εικόνες όμως, οι φασματικές υπογραφές των στόχων δεν ακολουθούν την κανονική κατανομή. Το 2012 οι Zhenwei Shi, Shuo Yang, Zhen Qin από την Κίνα [52], πρότειναν τον αλγόριθμο RNGMD ο οποίος χρησιμοποιεί μη συμβατά με την κανονική κατανομή στατιστικά μοντέλα για να χαρακτηρίσει τις φασματικές υπογραφές των στόχων. Το μοντέλο που προτάθηκε ανάγει τον εντοπισμό των στόχων σε ένα περιορισμένο πρόβλημα βελτιστοποίησης και χρησιμοποιεί την μέθοδο καθόδου των κλίσεων για να λύσει το πρόβλημα της βελτιστοποίησης. Τέλος, το 2012, οι Hao Sun, Xian Sun, Hongqi Wang, Yu Li, και Xiangjuan Li [47], εργάστηκαν πάνω στον αυτόματο εντοπισμό στρατιωτικών στόχων μέσω της χρησιμοποίησης ανιχνευτών περιγραφικών και γραμμικών SVM. Συνοπτικά, η συγκεκριμένη μεθοδολογία, αφού επιλέξει την μονάδα επεξεργασίας από το κινούμενο παράθυρο (αναζήτησης) και πραγματοποιήσει την εξαγωγή των χαρακτηριστικών σημείων, ακολουθεί μια νέα στρατηγική χωρικής χαρτογράφησης, η οποία επιτυγχάνει την κωδικοποίηση της γεωμετρικής πληροφορίας, η οποία δεν αναπαριστά μόνο την σχετική θέση των στοιχείων ενός στόχου αλλά και την δυνατότητα να αντιμετωπίσει περιπτώσεις περιστροφής αυτού. Σε αντίθεση με τον αλγόριθμο των K-κέντρων στο παραδοσιακό μοντέλο BOW, η αραιή κωδικοποίηση εισάγεται για να επιτύχει ανακατασκευή με πολύ μικρότερες πιθανότητες λάθους.

2.2. Ανασκόπηση Ειδικότερης Βιβλιογραφίας

Η ανασκόπηση της ειδικής βιβλιογραφίας, αναλύει διεξοδικά την εργασία των Hao Sun, Xian Sun, Hongqi Wang, Yu Li, και Xiangjuan Li [47]. Το 2012 παρουσίασαν το άρθρο **“Automatic Target Detection in High-Resolution Remote Sensing Images Using Spatial Sparse Coding Bag-of-Words Model”** (IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, VOL. 9, NO. 1, JANUARY 2012). Το άρθρο είχε ως αντικειμενικό σκοπό την ανάπτυξη μεθοδολογίας αυτόματου εντοπισμού στρατιωτικών στόχων με σύνθετο σχήμα σε εικόνες υψηλής ανάλυσης.

Συγκεκριμένα προτάθηκε ένα νέο πλαίσιο ανίχνευσης βασισμένο στο μοντέλο SSCBOW Spatial Sparse Coding Bag of Words, το οποίο παρουσιάζει καλύτερα αποτελέσματα από τον προκάτοχό του (BOW). Το μοντέλο BOW αρχικά είχε προταθεί για την κατηγοριοποίηση κειμένου. Ακολούθως χρησιμοποιήθηκε και στην ταξινόμηση αντικειμένων σε τηλεπισκοπικές εφαρμογές. Ο Xu [3] το 2012 χρησιμοποίησε τον BOW για ταξινόμηση σε εφαρμογές χρήσεων γης σε αεροφωτογραφίες υψηλής ανάλυσης.

Το BOW μοντέλο, όπως προαναφέραμε, αρχικά χρησιμοποιήθηκε για την στατιστική διαμόρφωση κειμένων σε μεγάλες συλλογές εγγράφων. Πρόσφατα χρησιμοποιήθηκε επιτυχώς για φυσικό σχολιασμό εικόνας, για αναγνώριση αντικειμένου, για ανάκτηση εικόνας κ.ά. Το μοντέλο διαχειρίζεται την εικόνα ως μια συλλογή άτακτης εμφάνισης περιγραφικών (descriptors) οι οποίοι έχουν εξαχθεί από τοπικά «μπαλώματα». Ακολούθως τα κβαντοποιεί και υπολογίζει μια συμπαγή αντιπροσώπηση ιστογράμματος.



Σχήμα 1 - Αναπαράσταση της διαδικασίας υλοποίησης του αλγορίθμου είτε με την χρήση του μοντέλου BOW είτε με το προτεινόμενο SSCBOW.

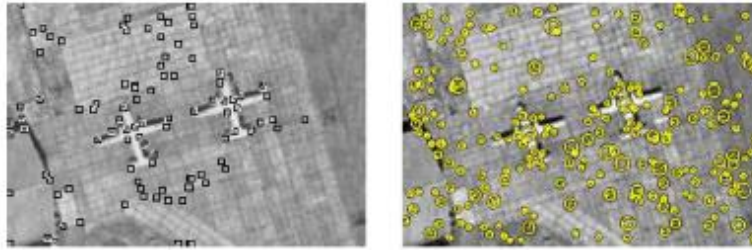
Στην φάση της εκπαίδευσης, ένα σύνολο από περιγραφικές λαμβάνεται από εξαγωγή χαρακτηριστικών και μια κωδική-οπτική λέξη διαμορφώνεται μέσω του αλγορίθμου K-κέντρων. Με την κωδική-οπτική λέξη αυτή οι περιγραφικές για κάθε εικόνα κωδικοποιούνται σε ένα γραμμικοποιημένο ιστόγραμμα, το οποίο και θα αποτελέσει στοιχείο τροφοδοσίας της εκπαίδευσης του SVM. Στο στάδιο της ταξινόμησης τα γραμμικοποιημένα ιστογράμματα της εικόνας δοκιμής τροφοδοτούνται για να ταξινομηθούν.

Για να ξεπεραστούν οι περιορισμοί του BOW μοντέλου προτείνεται το SSCBOW μοντέλο για τον χωρικό μετασχηματισμό των πληροφοριών δομής των στόχων. Αρχικά, δίνοντας μια εικόνα δοκιμής στον αλγόριθμο πρέπει να εξάγουμε διάφορα μπαλώματα ή ακόμα και ολόκληρες περιοχές οι οποίες θα αποτελέσουν υποψήφια προς ανίχνευση αντικείμενα. Οι περιοχές αυτές σαρώνονται από αμετάβλητους (κατά κλίμακα, στροφή ή ακόμα και αφινικούς μετασχηματισμούς) τοπικούς ανιχνευτές. Μερικοί από αυτούς τους ανιχνευτές είναι οι κάτωθι:

- Harris affine region Detector
- Dog (Lowe difference of Gaussians Detector)
- Kadir-Brady saliency detector
- SIFT scale invariant feature transform

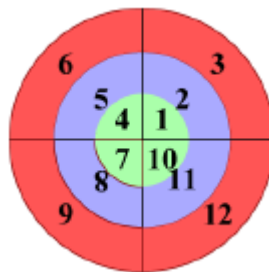
Μετά την εξαγωγή των χαρακτηριστικών, η περιγραφή των γνωρισμάτων υλοποιείται από περιγραφικές γνωρισμάτων οι οποίοι αντιπροσωπεύουν τα μπαλώματα ως αριθμητικά διανύσματα. Ένας καλός περιγραφιτής θα πρέπει να δύναται να αντιμετωπίσει παραλλαγές εντάσεως, κλίμακας, στροφής και αφινικούς μετασχηματισμούς.

Ο SIFT παρουσιάζει τα καλύτερα αποτελέσματα σε σχέση με τους ανωτέρω ανταγωνιστές του. Στην εν λόγω εργασία έγινε χρήση του CSS corner detector, ο οποίος είναι ένας ανιχνευτής γωνιών σε πολλαπλές κλίμακες με δυναμική περιοχή υποστήριξης και χρησιμοποιεί τον DOG για την εξαγωγή των πιθανών περιοχών (μπαλωμάτων). Τονίζεται ότι για την εξαγωγή αρκετών σημείων, αυξάνεται η ανάλυση των εικόνων πέντε φορές με αναδειγματοληψία (διγραμμική παρεμβολή).



Εικόνα 2 - Εξαγόμενα σημεία ενδιαφέροντος με τον αλγόριθμο CSS corner detector (AP), Εξαγόμενα σημεία ενδιαφέροντος με τον αλγόριθμο DoG detector (ΔΕΞ),

Για να αντιμετωπιστεί το πρόβλημα αταξίας του μοντέλου BOW, υιοθετείται στρατηγική χωρικής χαρτογράφησης για να αναπαρασταθεί η δομή των στόχων. Με δεδομένο ότι οι στόχοι μπορούν να βρεθούν σε οποιαδήποτε κατεύθυνση, η αντιπροσώπευση της δομής πρέπει να είναι σταθερή σε μετασχηματισμούς περιστροφής. Εμπνευσμένοι από τους αλγορίθμους ανάκτησης εικόνας, χαρτογραφούνται όλα τα μέρη του στόχου σε ένα πολικό σύστημα συντεταγμένων και χρησιμοποιούμε ένα κυκλικό παράθυρο το οποίο κινείται για να ανιχνεύσει τις περιοχές ενδιαφέροντος. Αφού καθοριστεί ένα κέντρο κύκλου ακολούθως ορίζονται τομείς του κύκλου με την ίδια ακτίνα (τεταρτημόρια) και το ίδιο διάστημα, όπως φαίνεται στην παρακάτω εικόνα:

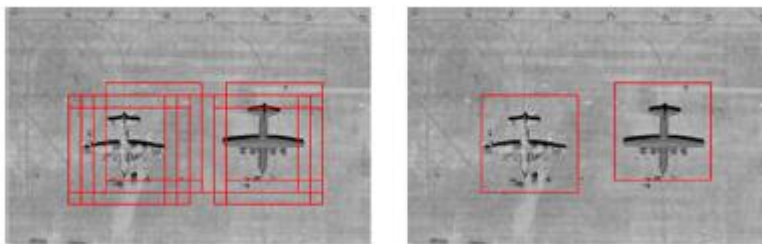


Σχήμα 2 - Αναπαράσταση της διαμερισματοποίησης του κυκλικού παραθύρου αναζήτησης.

Κάθε τομέας θεωρείται ως ένα δοχείο και το αντίστοιχο ιστόγραμμα υπολογίζεται για να χαρακτηρίζει την αντίστοιχη περιοχή. Στην περίπτωση που ένας στόχος βρίσκεται με διαφορετικούς προσανατολισμούς σε διαφορετικά δοχεία, τότε αυτό θα οδηγήσει σε λανθασμένο συνδυασμό των δύο ιστογραμμάτων. Ως εκ τούτου η κανονικοποίηση του ιστογράμματος χρησιμοποιείται για να λύσει το παραπάνω πρόβλημα.

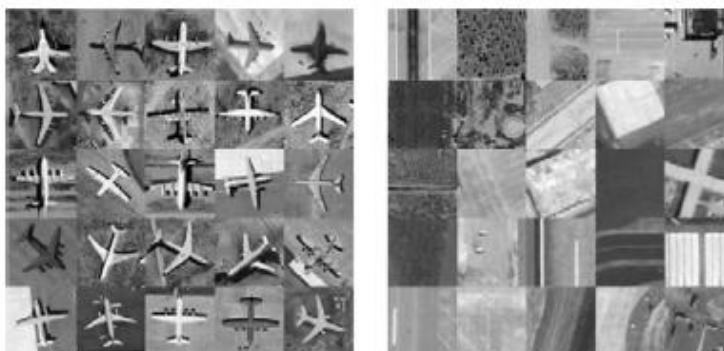
Κατά την διάρκεια της φάσης της ανίχνευσης ο δυαδικός ταξινομητής παραθύρων κινείται μέσα στην εικόνα με κάποιο βήμα για να εντοπίσει τους στόχους. Οι

περιπτώσεις των πολλαπλών εντοπισμών διορθώνονται με την υιοθέτηση στρατηγικής καταστολής του ελαχίστου.

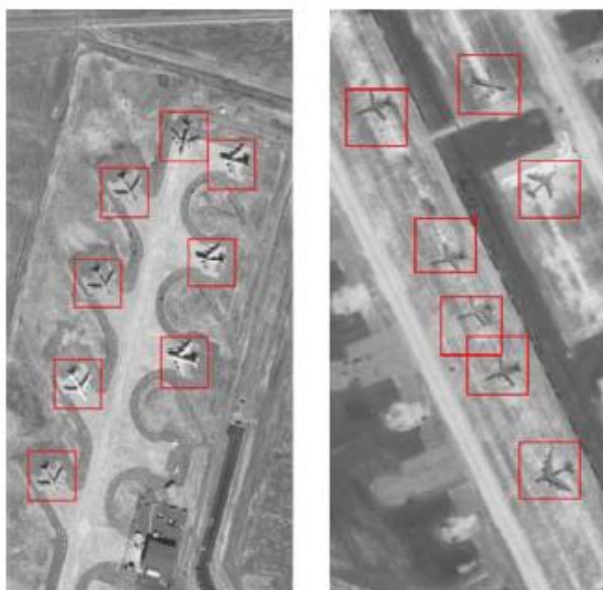


Εικόνα 3Α - Παράδειγμα πολλαπλού εντοπισμού στόχου.

Στις παρακάτω εικόνες εμφανίζονται ενδεικτικά παραδείγματα εικόνων που χρησιμοποιήθηκαν για την διαδικασία εκπαίδευσης του αλγόριθμο καθώς επίσης και τα τελικά αποτελέσματα ανίχνευσης:



Εικόνα 4B - Παράδειγμα εικόνων εκπαίδευσης. Θετικές εικόνες εκπαίδ. (ΑΡ) – Αρνητικές εικόνες εκπαίδ. (ΔΕΞ)



Εικόνα 5Γ Παράδειγμα εντοπισμού χρησιμοποιώντας την μέθοδο SSCBOW.

3. Ανάπτυξη αλγορίθμων για τον αυτόματο εντοπισμό αεροσκαφών σε διαχρονικά δορυφορικά δεδομένα.

3.1. Ανάπτυξη αλγορίθμου με βάση τεχνικές αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας.

Στο κεφάλαιο αυτό παρουσιάζεται μια συνοπτική περιγραφή των βασικών τεχνικών επεξεργασίας που πραγματοποιήθηκαν στις ψηφιακές εικόνες στα πλαίσια του εντοπισμού των Αεροσκαφών. Θα υλοποιηθεί μια σειρά από βήματα προεπεξεργασίας των ψηφιακών εικόνων με κύριο σκοπό την εξαγωγή των αεροσκαφών μέσα από μετρήσεις που θα πραγματοποιηθούν στις δυαδικές εικόνες που θα έχουν προκύψει από την αφαίρεση και κατωφλίωση των εικόνων.

3.1.1. Προσδοκώμενα αποτελέσματα.

Όταν ολοκληρωθεί η μελέτη του παρόντος κεφαλαίου θα πρέπει να έχουμε επιτύχει:

1. Αντιστοίχιση των εικόνων με τη χρήση ArcGis10
2. Αποκατάσταση και διόρθωση των διαστάσεων των εικόνων.
3. Ραδιομετρική διόρθωση.
4. Ομαλοποίηση εικόνων.
5. Εντοπισμός αντικειμένων σε binary εικόνες.
6. Πραγματοποίηση μετρήσεων (γεωμετρικά & φασματικά) επί των οντοτήτων που θα έχουν εντοπιστεί και επιλογή μόνο των αντικειμένων ενδιαφέροντος.

3.1.2. Αλγοριθμικά επιμέρους βήματα.

Η ψηφιακή εικόνα αποτελεί στην εποχή μας μια από τις σημαντικότερες πηγές πληροφορίας. Σημαντικό πρόβλημα της ψηφιακής εικόνας είναι η διαχείριση του τεράστιου διαθέσιμου όγκου δεδομένων καθώς και η εξαγωγή από αυτόν, των επιθυμητών πληροφοριών, στον επιθυμητό χρόνο. Στην συνέχεια, θα αναλυθούν διεξοδικά τα βήματα επεξεργασίας που πραγματοποιήθηκαν για την εξαγωγή των αεροσκαφών.






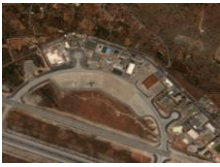

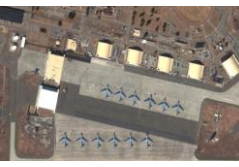

















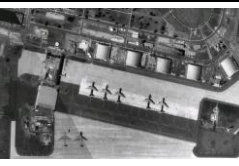

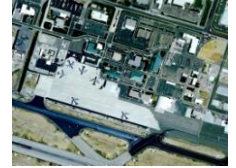
Τα βήματα επεξεργασίας των εικόνων που κόπηκαν από το Google Earth παρουσιάζονται στο παρακάτω διάγραμμα ροής:

Βήματα επεξεργασίας	
1	Αντιστοίχιση εικόνων (Registration).
2	Αποκατάσταση και διόρθωση των διαστάσεων.
3	Ραδιομετρική διόρθωση εικόνων.
4	Ομαλοποίηση εικόνων.
5	Αναζήτηση εντοπισμών σε όλους τους δυνατούς συνδυασμούς
6	Αφαίρεση & Κατωφλίωση εικόνων.
7	Μαθηματική μορφολογία και ανάλυση ιδιοτήτων.
7α	#Απόρριψη "holes" που βρίσκονται στο περίγραμμα.
7β	#Εντοπισμός αντικειμένων (holes) και υλοποίηση labeling αυτών.
7γ	#Υπολογισμός των ιδιοτήτων των holes.
7δ	#Ανάλυση και τυποποίηση ιδιοτήτων για την εξαγωγή των επιθυμητών στόχων.
8	Υπολογισμοί MAX, MIN, M.O για κάθε κανάλι και για κάθε οντότητα.
9	Δημιουργία κατωφλίων με βάση τους κανονικοποιημένους M.O για την εξαγωγή των επιθυμητών στόχων.

Πίνακας 2 – Βήματα Επεξεργασίας 1^{ης} Μεθοδολογίας.

Τα δεδομένα που χρησιμοποιήθηκαν προήλθαν από δορυφορικές εικόνες της Google Earth. Οι περιοχές μελέτης αφορούσαν στρατιωτικά και πολιτικά αεροδρόμια τόσο στην Ελλάδα όσο και στις Ηνωμένες Πολιτείες Αμερικής. Τα αεροδρόμια που χρησιμοποιήθηκαν είναι τα κάτωθι:

1. Σούδα (Αμερικάνικη Αεροπορική Βάση)
2. Holladay USAF (Στρατιωτικό & Πολιτικό)
3. Watover Air Reserve Base (Στρατιωτικό)
4. Tyndal Air Force Base (Στρατιωτικό)
5. Fort Riley Kansas (Στρατιωτικό)

1 Σούδα	2 Holladay USAF	3 Watover Air R. Base	4 Tyndal Air Force Base	5 Fort Riley Kansas
				
				
				
				
				
				
				

Πίνακας 3 – Δορυφορικές εικόνες που χρησιμοποιήθηκαν ανά αεροδρόμιο για την 1^η Μεθοδολογία.

3.1.3.1. Αντιστοίχιση εικόνων (Registration)

Το registration των εικόνων υλοποιήθηκε με το λογισμικό ArcGis10. Ταυτόχρονα δοκιμάστηκε και ο αλγόριθμος Open Surf που προσφέρεται από βιβλιοθήκη του Matlab 2013a. Το στάδιο του registration δεν ήταν ο κύριος στόχος της παρούσας διπλωματικής για αυτό και δεν δόθηκε μεγάλη βαρύτητα στη χρήση αλγορίθμων όπως οι Sift, Surf, OpenSurf κ.α.

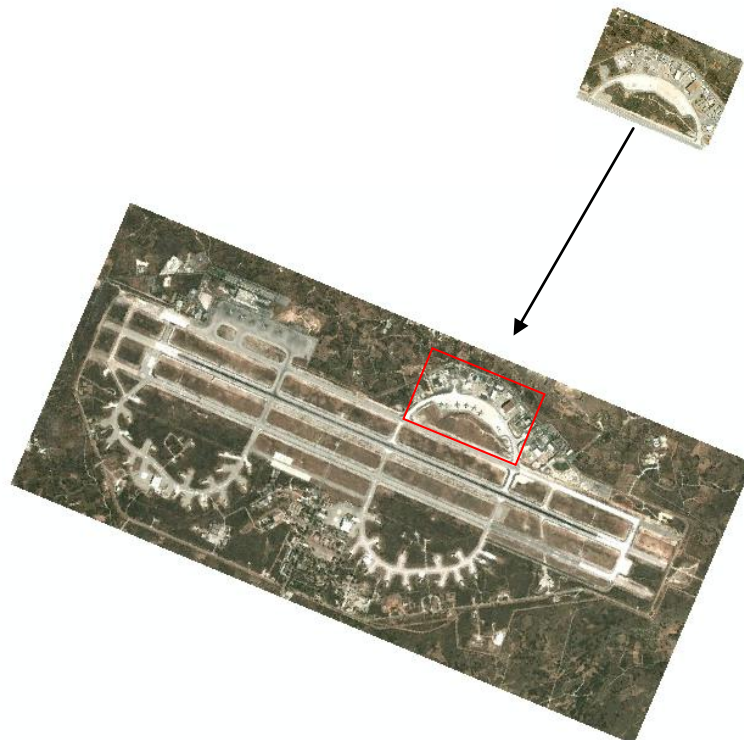
Τα βήματα που ακολουθήθηκαν για το registration είναι τα κάτωθι:

1. Γεωαναφορά της εικόνας ολόκληρου του Α/Δ με βάση τις συντεταγμένες της Google Earth.



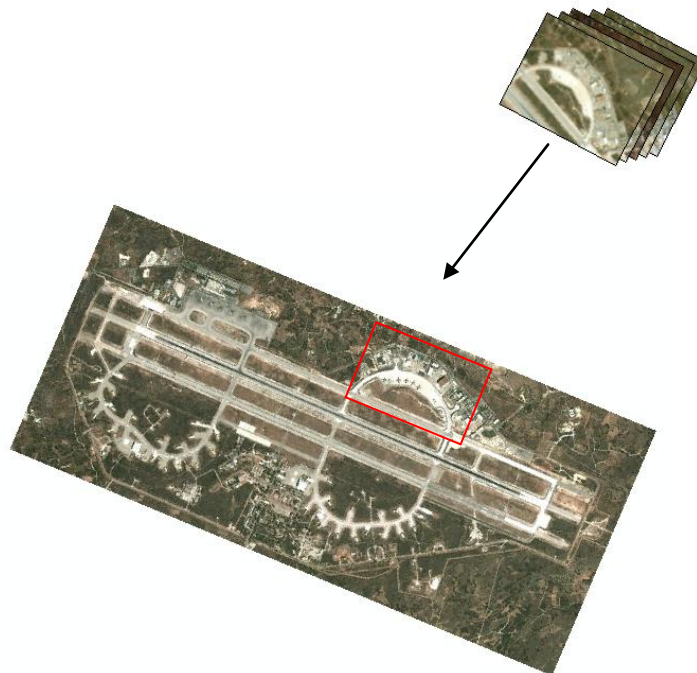
Εικόνα 6 – Εικόνα αεροδρομίου Σούδας που έχει γεωαναφερθεί με βάση τις συντεταγμένες του Google Earth .

2. Γεωαναφορά με βάση την προηγούμενη του Α/Δ, μιας εικόνας που συνήθως έχει και την καλύτερη ανάλυση.



Εικόνα 7– Γεωαναφορά της εικόνας με την καλύτερη ανάλυση με βάση την εικόνα του προηγούμενου βήματος.

3. Γεωαναφορά όλων των υπολοίπων εικόνων με βάση την προηγούμενη εικόνα.



Εικόνα 8 – Γεωαναφορά όλων των υπολοίπων εικόνων με βάση την εικόνα του προηγούμενου βήματος.

4. Αποκοπή της περιοχής που θα εξεταστεί με το toolbox clip.



Εικόνα 9– Αποκοπή της περιοχής που θα εξεταστεί .

Παρατήρηση 1^η:

1. Για την «σωστή» υλοποίηση της γεωαναφοράς απαιτήθηκε η εφαρμογή πολυωνύμου 2^{ου} ή 3^{ου} βαθμού. Με τον όρο σωστά εννοούμε την μεγαλύτερη δυνατή ταύτιση των δύο εικόνων ή το δυνατόν στο pixel, έτσι ώστε κατά την διαδικασία αφαίρεσης των εικόνων, να προκύπτουν τα αντικείμενα αναζήτησης. Αυτό σημαίνει ότι έπρεπε να επιλεγούν τουλάχιστον 6 και 10 tie points αντίστοιχα (Στην πράξη χρησιμοποιήθηκαν πολύ περισσότερα). Στην αντίθετη περίπτωση, της εφαρμογής του αφινικού μετασχηματισμού (απαραίτητα κατ'ελάχιστον 3 σημεία), το αποτέλεσμα δεν ήταν το επιθυμητό. Ταυτόχρονα με την εντολή clip η ανάλυση χειροτέρευε διότι η τομή ανάγκαζε την εικόνα σε μια νέα ανακατανομή των pixels με βάση το νέο πλαίσιο αποκοπής.

Παρατήρηση 2^η:

Πραγματοποιήθηκε έλεγχος του registration των παραπάνω εικόνων με βάση τον αλγόριθμο OpenSurf, χρησιμοποιώντας αυτόματη ταύτιση χαρακτηριστικών σημείων. Ο συγκεκριμένος αλγόριθμος κάνει χρήση και του αλγορίθμου RANSAQ για το έλεγχο και την επιλογή των καλύτερων δυνατών εξαγόμενων σημείων.

3.1.3.2. Αποκατάσταση και διόρθωση των διαστάσεων

Πριν την οποιαδήποτε αναδιαστασιολόγηση, εντοπίστηκαν οι διαστάσεις όλων των εικόνων από κάθε dataset. Καταχωρήθηκαν τα αποτελέσματα κατά x και κατά y σε νέο πίνακα και μετά βρέθηκαν οι μέγιστες τιμές. Ακολούθως έγινε το resizing και αποθηκεύτηκαν οι νέες εικόνες σε ξεχωριστό φάκελο.

```
%Edopismos ton diastaseon gia to resizing
imagefiles = dir(fullfile(f1, '*.jpg'));
nfiles = length(imagefiles);    % Number of files found
BB=zeros(nfiles,2);
for ii=1:nfiles
    currentfilename = imagefiles(ii).name;
    currentimage = imread(currentfilename);
    [BB(ii,1),BB(ii,2)]=size(currentimage);
end
gr = max (BB(:,1));%GrammesRe
st = max (BB(:,2)/3);%Stiles

%Resizing
currentimage = imresize(currentimage, [gr st]);%Pragmatopoeitai resize tis eikonas
baseFileName = sprintf('R%d.jpg', ii); %Kathorismos onomatos eikonas (resized)
%figure(ii);imshow(image(ii));impixelinfo;
fullFileName = fullfile(f2, baseFileName);%Olokliro to path os sinthesi tou fi kai
του neou onomatos
imwrite(currentimage, fullFileName); %Grapsimo tis neas eikonas
%figure(ii);imshow(currentimage);impixelinfo;hold on;
```

3.1.3.3. Ραδιομετρική διόρθωση εικόνων

Στη σελίδα 24 μπορούμε εύκολα να παρατηρήσουμε ότι οι κομμένες εικόνες έχουν διαφορετικά ραδιομετρικά χαρακτηριστικά. Η λήψη τους έχει γίνει σε διαφορετική χρονική περίοδο και προφανώς με διαφορετικές συνθήκες φωτισμού και σκίασης. (Πολύ πιθανόν να έχουν γίνει και με διαφορετικούς δέκτες). Για τους παραπάνω λόγους οφείλουμε να τις διορθώσουμε ραδιομετρικά.

Για την εκτέλεση της ραδιομετρικής διόρθωσης κατασκευάστηκε η συνάρτηση “Radiocorr.m”. Η συνάρτηση παίρνει ως όρισμα μόνο την εικόνα που θέλουμε να διορθώσουμε ραδιομετρικά.

```
function [correctedRgbImage] = Radiocorr(uncorrectedRgbImage)
```

Μετά σπάμε την εικόνα στα τρία διαφορετικά κανάλια (R-G-B) και υπολογίζουμε τον Μ.Ο για κάθε πίνακα-κανάλι. Με τους τρεις αυτούς Μ.Ο για κάθε κανάλι υπολογίζουμε τον επιθυμητό Μ.Ο και των τριών καναλιών. Ο επιθυμητός Μ.Ο θα χρησιμοποιηθεί για τον υπολογισμό του συντελεστή διόρθωσης του κάθε καναλιού ξεχωριστά.

Για να υπολογίσουμε τον συντελεστή διόρθωσης του κάθε καναλιού διαιρούμε τον επιθυμητό Μ.Ο με τον Μ.Ο του αντίστοιχου καναλιού. Τέλος, πολλαπλασιάζουμε το κάθε κανάλι με τον συντελεστή διόρθωσης (correctionFactorR) και επανασυνθέτουμε την εικόνα (με την εντολή cat). Όπως και παραπάνω, κάθε στάδιο επεξεργασίας αποθηκεύεται σε ξεχωριστό φάκελο.

```
% Extract the individual red, green, and blue color channels.
redChannel = uncorrectedRgbImage(:, :, 1);
greenChannel = uncorrectedRgbImage(:, :, 2);
blueChannel = uncorrectedRgbImage(:, :, 3);
% Get the means of each color channel

meanR = mean2(redChannel);
meanG = mean2(greenChannel);
meanB = mean2(blueChannel);
% specify the desired mean.

desiredMean = mean([meanR, meanG, meanB]);
% Linearly scale the image in the cropped ROI.

correctionFactorR = desiredMean / meanR;
correctionFactorG = desiredMean / meanG;
correctionFactorB = desiredMean / meanB;

...

currentimage = Radiocorr(currentimage);
    baseFileName = sprintf('C%d.jpg', ii); %Kathorismos onomatos eikonas (resized)
    %figure(ii);imshow(image(ii));impixelinfo;
    fullFileName = fullfile(f3, baseFileName);%Olokliro to path os sinthesi tou fi kai
του neou onomatos
    imwrite(currentimage, fullFileName); %Grapsimo tis neas eikonas
```

Ακολούθως η μετατροπή των εικόνων σε grayscale γίνεται με την έτοιμη συνάρτηση “rgb2gray”.

```
%Metatropi se grayscale olon ton eikonon
currentimage = rgb2gray(currentimage);
baseFileName = sprintf('G%d.jpg', ii); %Kathorismos onomatos eikonas (resized)
%figure(ii);imshow(image(ii));impixelinfo;
fullFileName = fullfile(f4, baseFileName);
imwrite(currentimage, fullFileName); %Grapsimo tis neas eikonas
```

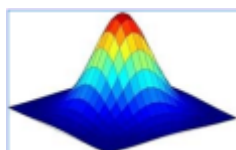
3.1.3.4. Ομαλοποίηση εικόνων.

Για την ομαλοποίηση των εικόνων χρησιμοποιήθηκε το φίλτρο Gauss. Το φίλτρο Gauss είναι ένα χαμηλοπερατό φίλτρο το οποίο απομακρύνει τον θόρυβο και αμβλύνει τις ακμές (θόλωμα της εικόνας). Χρησιμοποιήθηκε η έτοιμη συνάρτηση “imfilter” και δοκιμάστηκαν οι παρακάτω συνδυασμοί παραθύρου συνέλιξης και σ:

3*3 & σ=0.5	5*5 & σ=0.5	7*7 & σ=0.5
3*3 & σ=1.0	5*5 & σ=1.0	7*7 & σ=1.0
3*3 & σ=1.5	5*5 & σ=1.5	7*7 & σ=1.5
Παράθυρο 3*3	Παράθυρο 5*5	Παράθυρο 7*7

Πίνακας 4 – Περιπτώσεις Γκαουσιανών φίλτρων που χρησιμοποιήθηκαν, στο στάδιο της απομάκρυνσης θορύβου των εικόνων.

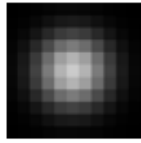
Το εν λόγω φίλτρο έχει κατανομή gauss περί του κεντρικού pixel της μάσκας και προφανώς έχει μηδενική επίδραση στα σημεία τα οποία βρίσκονται σε απόσταση >3σ από αυτό. Η σχέση που το ορίζει σε δύο διαστάσεις είναι :



$$g(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$










Σχήμα 3– Αναπαράσταση Γκαουσιανού φίλτρου και εξίσωση αυτού για πίνακα δύο διαστάσεων.

Παράδειγμα φίλτρου Gauss με 5*5 παράθυρο και $\sigma=1$:

$$\sigma=1 \rightarrow a_{\text{Gauss}} = \frac{1}{6.17} \begin{bmatrix} 0.02 & 0.08 & 0.14 & 0.08 & 0.02 \\ 0.08 & 0.37 & 0.61 & 0.37 & 0.08 \\ 0.14 & 0.61 & 1 & 0.61 & 0.14 \\ 0.08 & 0.37 & 0.61 & 0.37 & 0.08 \\ 0.02 & 0.08 & 0.14 & 0.08 & 0.02 \end{bmatrix}$$


Σχήμα 4 – Παράδειγμα φίλτρου Gauss 5-5 και $\sigma=1$.

Στον παρακάτω πίνακα φαίνονται όλα τα αποτελέσματα από τα φίλτρα gauss που χρησιμοποιήθηκαν για μια εικόνα δοκιμής:

$\sigma=0,5$			
$\sigma=1,0$			
$\sigma=1,5$			
	3*3	5*5	7*7

Πίνακας 5 –Απεικονίσεις όλων των φίλτρων Gauss που χρησιμοποιήθηκαν σε εικόνα του Α/Δ Σούδας

Παρατήρηση 3^η: Ολόκληρος ο κώδικας από το βήμα 6^ο και μετά, υλοποιήθηκε για κάθε μια από τις παραπάνω περιπτώσεις γκαουσιανών φίλτρων.

```

%Gaussian filter
%Apothikeuo tin eikona gia na min alloiothei apo ta Gaussiana filtra
Baseimage = currentimage;
%figure(ii);imshow(image(ii));impixelinfo;

%Oi periptoseis pou exo epilekseis
for r=1:sin2
    m1 = Sineliksi(1,r);
    for t=1:sig2
        s1 = Sigma(1,t);
        G = fspecial('gaussian',[m1 m1],s1);
        Gaussianimage = imfilter(Baseimage,G,'same');
        %figure(ii);imshow(image(ii));impixelinfo;hold on;
        str = ['G_',num2str(ii),'_',num2str(m1),'_',num2str(s1),'.jpg'];
        %baseFileName = sprintf('G_%d_%d_%d.jpg',ii,stm1,sts1); %Kathorismos
onomatos eikonas (resized)
        strfolder = [num2str(m1),'_',num2str(s1)];% Gia tin apothikeusi se kathe
fakelo ksexorista
        fullFileName = fullfile(f5,strfolder, str);%Olokliro to path os sinthesi
tou fi kai tou neou onomatos
        imwrite(Gaussianimage, fullFileName); %Grapsimo tis neas eikonas
    end
end
end

```

3.1.3.5. Αναζήτηση εντοπισμών σε όλους τους δυνατούς συνδυασμούς.

Ο στόχος μας είναι η εξαγωγή των αεροσκαφών μέσω της πράξης της αφαίρεσης μεταξύ δύο εικόνων. Επειδή οι εικόνες που διαθέτουμε είναι περισσότερες από δύο σε κάθε αεροδρόμιο χρησιμοποιούμε την έτοιμη συνάρτηση “combntns” για υπολογίσουμε όλους τους δυνατούς συνδυασμούς μεταξύ των διατιθέμενων εικόνων προς επεξεργασία. Η παραπάνω συνάρτηση υπολογίζει όλους τους δυνατούς συνδυασμούς ανά δύο εικόνες. Με αυτόν τον τρόπο εντοπίστηκαν και τα ζεύγη εικόνων τα οποία είχαν τα καλύτερα αποτελέσματα.

Παρατήρηση: Η παρακάτω γραμμή κώδικα βρίσκεται μέσα στη συνάρτηση “Afaresi_Katoflios”.

```

combos = combntns(1:nfiles,2);

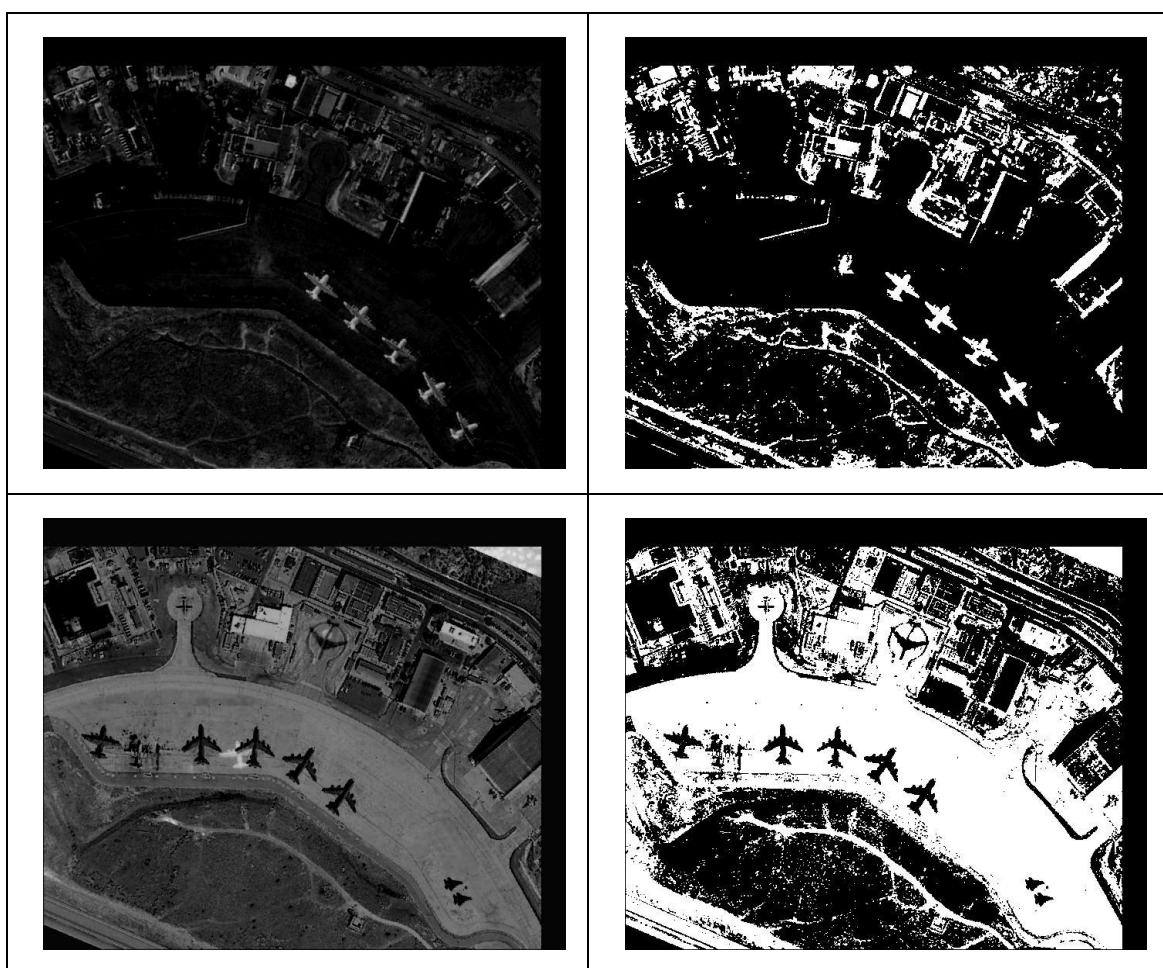
```

3.1.3.6. Αφαίρεση & Κατωφλίωση εικόνων

Όπως προαναφέραμε παραπάνω, το παρόν στάδιο πραγματοποιήθηκε και για τα 9 gauss-ιανά φίλτρα που υπολογίσαμε. Δημιουργήθηκε η συνάρτηση “Afaireisi_Katofliosi”. Η συνάρτηση είναι κατασκευασμένη έτσι ώστε να τρέχει σε loop με ορίσματα τα χαρακτηριστικά των φίλτρων gauss.

Ουσιαστικά υπολογίζει σε κάθε επανάληψη τη διαφορά των εικόνων (για όλους τους συνδυασμούς) και μετά κατωφλιώνει την εικόνα (BW) μέσω της εντολής “im2bw” και του αντίστοιχου επιπέδου κατωφλίωσης που έχει εξαχθεί αυτόματα.

```
ImageNew1 = imread(imagefiles(combos(ii,1),1).name) -  
imread(imagefiles(combos(ii,2),1).name);%Eikona afaireisis  
level = (graythresh(ImageNew1));  
ImageNew2 = im2bw(ImageNew1, level);
```

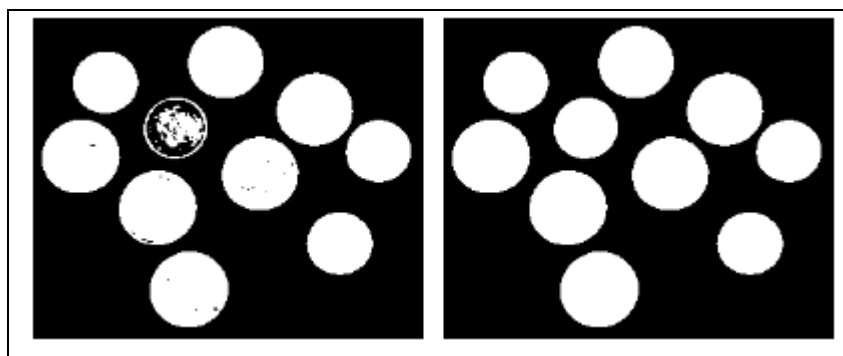


Πίνακας 6 –Παραδείγματα Αφαίρεσης (AP) και Κατωφλίωσης (ΔΕΞ)

3.1.3.7. Μαθηματική μορφολογία και ανάλυση ιδιοτήτων.

Από το 9^ο στάδιο ξεκινάει η επεξεργασία δυαδικών εικόνων έτσι ώστε να βελτιστοποιήσουμε το επιθυμητό αποτέλεσμα (εντοπισμός αεροπλάνων). Η πρώτη ενέργεια που οφείλουμε να κάνουμε στις δυαδικές μας εικόνες είναι να γεμίσουμε τις «λευκές» τρύπες, έτσι ώστε να μην εμφανίζονται “doughnut shapes”. Αυτό επιτυγχάνεται με την εντολή “imfill”, όπως φαίνεται παρακάτω:

```
ImageNew2 = imfill(ImageNew2,'holes');
```

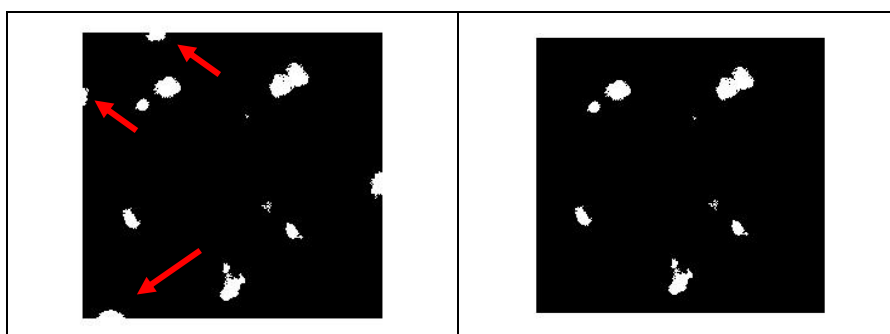


Εικόνα 10 – Παράδειγμα εφαρμογής της εντολής imfill

Απόρριψη “holes” που βρίσκονται στο περίγραμμα.

Σε αυτό το στάδιο αφαιρούνται όλες οι «τρύπες» οι οποίες εφάπτονται με τα όρια της εικόνας. Με αυτόν τον τρόπο ελαχιστοποιούνται τόσο οι πιθανές «τρύπες» που δύναται να αποτελούν «σκουπίδια» στις τελικές εικόνες όσο και οι ιδιαίτερα δύσκολες περιπτώσεις εντοπισμού αεροσκαφών που όντως είναι στα όρια και είναι πρακτικά αδύνατον να εντοπιστούν.

```
F=imclearborder(bw); % Katharisma orion
```



Εικόνα 11–Παράδειγμα απόρριψης οντοτήτων που εφάπτονται με το περίγραμμα της εικόνας

Εντοπισμός αντικειμένων (holes) και υλοποίηση labeling αυτών.

Το στάδιο αυτό είναι το πιο σημαντικό από όλη την διαδικασία, διότι μέσω της εντολής «bwlablel» εντοπίζονται στις δυαδικές μας εικόνες οι «τρύπες». Ο εντοπισμός που πραγματοποιείται περιλαμβάνει τον καθορισμό ενός μοναδικού «ID» για κάθε τρύπα (label/ετικέτα). Ουσιαστικά δημιουργείται ένας νέος πίνακας ίδιων διαστάσεων με την αρχική εικόνα ο οποίος έχει παντού μηδέν και, όπου υπάρχει οντότητα-τρύπα, αναγράφεται ο αντίστοιχος αύξοντας αριθμός ID.

```
L = bwlablel(F);
```

Στις παρακάτω εικόνες φαίνεται ένα αρκετά απλοποιημένο παράδειγμα για την κατανόηση της εντολής.

```
BW = logical ([1 1 1 0 0 0 0 0
               1 1 1 0 1 1 0 0
               1 1 1 0 1 1 0 0
               1 1 1 0 0 0 1 0
               1 1 1 0 0 0 1 0
               1 1 1 0 0 0 1 0
               1 1 1 0 0 1 1 0
               1 1 1 0 0 0 0 0]);
```

```
L = bwlablel(BW,4)
```

```
L =
```

```
1 1 1 0 0 0 0 0
1 1 1 0 2 2 0 0
1 1 1 0 2 2 0 0
1 1 1 0 0 0 3 0
1 1 1 0 0 0 3 0
1 1 1 0 0 0 3 0
1 1 1 0 0 3 3 0
1 1 1 0 0 0 0 0
```

Σχήμα 5–Παράδειγμα υλοποίησης του labeling στις δυαδικές εικόνες

Υπολογισμός των ιδιοτήτων των holes

Αφού έχουν υπολογιστεί τα παραπάνω labels, πλέον μπορούμε να υπολογίσουμε διάφορα γεωμετρικά χαρακτηριστικά τα οποία θα μας χρησιμεύσουν στο επόμενο στάδιο για την απομόνωση των διαφόρων τύπων αεροσκαφών. Για κάθε οντότητα υπολογίστηκαν όλα τα διατιθέμενα χαρακτηριστικά, αλλά χρησιμοποιήθηκαν μόνο τα κάτωθι:

Field	Value	Min	Max
Area	215	215	215
Centroid	[243.9953,111.2744]	111.2...	243.99...
BoundingBox	[229.5000,99.5000,29,...	26	229.50...
SubarrayIdx	<1x2 cell>		
MajorAxisLength	23.8642	23.8642	23.8642
MinorAxisLength	22.2610	22.2610	22.2610
Eccentricity	0.3603	0.3603	0.3603
Orientation	-59.7670	-59.7...	-59.76...
ConvexHull	<18x2 double>	99.5000	258.50...
ConvexImage	<26x29 logical>		
ConvexArea	473	473	473
Image	<26x29 logical>		
FilledImage	<26x29 logical>		
FilledArea	215	215	215
EulerNumber	1	1	1
Extrema	<8x2 double>	99.5000	258.50...
EquivDiameter	16.5453	16.5453	16.5453
Solidity	0.4545	0.4545	0.4545
Extent	0.2851	0.2851	0.2851
PixelIdxList	<215x1 double>	127668	143264
PixelList	<215x2 double>	100	258
Perimeter	152.8112	152.8...	152.81...

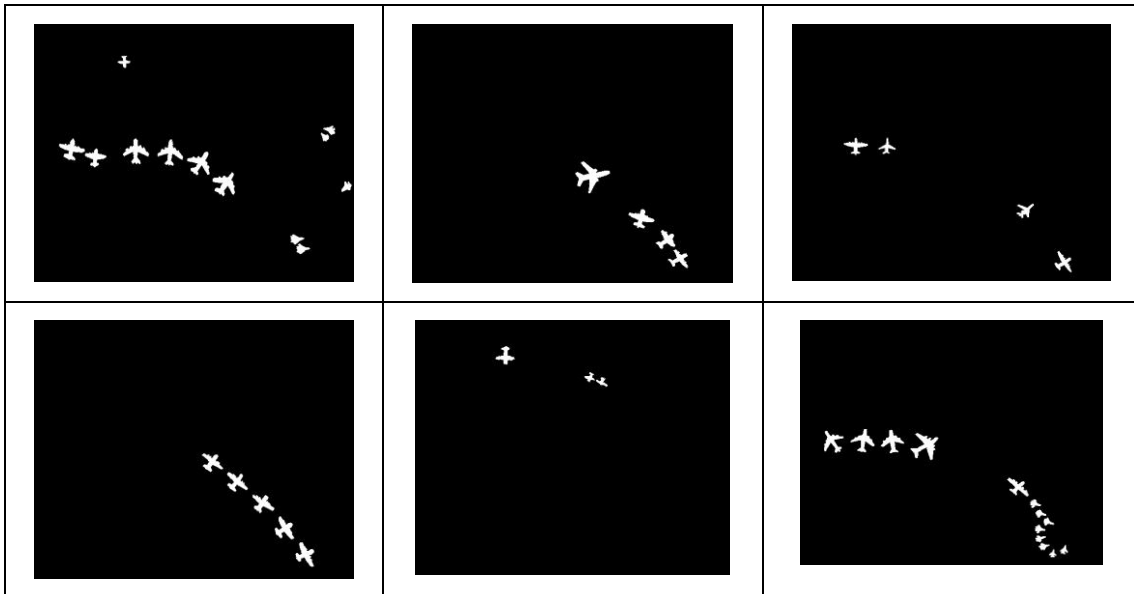
- Εμβαδόν
- Περίμετρος
- Μεγάλος Άξονας
- Μικρός Άξονας
- Εκκεντρότητα
- Ισοδύναμη διάμετρος

Σχήμα 6 – Στιγμιότυπο εξαγωγής όλων των γεωμετρικών ιδιοτήτων (AP) καθώς και εκείνων που χρησιμοποιήθηκαν (ΔΕΞ)

Ανάλυση και τυποποίηση ιδιοτήτων για την εξαγωγή των επιθυμητών στόχων

Αφού υπολογίστηκαν τα παραπάνω χαρακτηριστικά, πλέον μπορούμε να διαμορφώσουμε τα κατάλληλα φίλτρα τα οποία θα απομονώνουν τον κάθε επιθυμητό τύπο αεροσκάφους.

Για τον υπολογισμό των παραπάνω χαρακτηριστικών κατασκευάστηκαν εικόνες “ground-truth” στις οποίες ψηφιοποιήθηκαν τα αντικείμενα αναζήτησης και μετά καταγράφηκαν τα εύρη στα οποία κυμαίνονταν τα γεωμετρικά χαρακτηριστικά για κάθε τύπο αεροσκάφους.



Πίνακας 7 – Εικόνες Ground Truth για το Α/Δ της Σούδας

Παρακάτω φαίνεται η εντολή με την οποία υπολογίστηκαν αυτόματα τα χαρακτηριστικά για την κάθε οντότητα καθώς και εκείνα που τελικά αξιοποιήθηκαν. Τονίζεται ότι τα εύρη για το κάθε χαρακτηριστικό επαυξήθηκαν ανάλογα, γιατί κατά κανόνα οι αξιοποιήσιμες binary εικόνες με αεροσκάφη είχαν οντότητες με μεγαλύτερη επιφάνεια, πράγμα που επηρέαζε όλα τα κριτήρια.

```
stats = regionprops(L,'all');

allBlobAreas = [stats.Area];
allBlobPerimeter = [stats.Perimeter];
allBlobMajorAxisLength=[stats.MajorAxisLength];
allBlobMinorAxisLength=[stats.MinorAxisLength];
allBlobEccentricity=[stats.Eccentricity];
allBlobEquivDiameter=[stats.EquivDiameter];

%0) Big-big c130
allowableAreaIndexes0 = (allBlobAreas > 1700) & (allBlobAreas < 2350) &
(allBlobPerimeter > 300) & (allBlobPerimeter < 450) & (allBlobMajorAxisLength<110) &
(allBlobMinorAxisLength>35) & (allBlobEccentricity>0.5) & (allBlobEccentricity<0.95) &
(allBlobEquivDiameter < 55) & (allBlobEquivDiameter > 30);

%1) >c130
allowableAreaIndexes1 = (allBlobAreas > 900) & (allBlobAreas < 1750) &
(allBlobPerimeter > 215) & (allBlobPerimeter < 300) & (allBlobMajorAxisLength<65) &
(allBlobMinorAxisLength>35) & (allBlobEccentricity>0.2) & (allBlobEccentricity<0.78) &
(allBlobEquivDiameter < 48) & (allBlobEquivDiameter > 30);

%2) <c130
allowableAreaIndexes2 = (allBlobAreas > 500) & (allBlobAreas < 900) &
(allBlobPerimeter > 110) & (allBlobPerimeter < 250) & (allBlobMajorAxisLength<59) &
```

```

(allBlobMinorAxisLength>12) & (allBlobEccentricity>0.2) & (allBlobEccentricity<0.98) &
(allBlobEquivDiameter < 40)& (allBlobEquivDiameter > 20);

%3) =c130
allowableAreaIndexes3 = (allBlobAreas > 580)& (allBlobAreas < 1300)&
(allBlobPerimeter > 180)&(allBlobPerimeter < 280) & (allBlobMajorAxisLength<60) &
(allBlobMinorAxisLength>23) & (allBlobEccentricity>0.3) & (allBlobEccentricity<0.9) &
(allBlobEquivDiameter < 50)& (allBlobEquivDiameter > 15);

%4) f15
allowableAreaIndexes4 = (allBlobAreas > 130)& (allBlobAreas < 380) &
(allBlobPerimeter > 50)&(allBlobPerimeter < 100) & (allBlobMajorAxisLength<38) &
(allBlobMinorAxisLength>8) & (allBlobEccentricity>0.5) & (allBlobEccentricity<0.95)&
(allBlobEquivDiameter < 25)& (allBlobEquivDiameter > 10);

%5) chessna
allowableAreaIndexes5 = (allBlobAreas > 190)& (allBlobAreas < 510) &
(allBlobPerimeter > 75)&(allBlobPerimeter < 290) & (allBlobMajorAxisLength<41) &
(allBlobMinorAxisLength>14) & (allBlobEccentricity>0.3) & (allBlobEccentricity<0.78) &
(allBlobEquivDiameter < 27)& (allBlobEquivDiameter > 15);

%6) president
allowableAreaIndexes6 = (allBlobAreas > 300)& (allBlobAreas < 550) &
(allBlobPerimeter > 120)&(allBlobPerimeter < 200) & (allBlobMajorAxisLength<40) &
(allBlobMinorAxisLength>20) & (allBlobEccentricity>0.3) & (allBlobEccentricity<0.8) &
(allBlobEquivDiameter < 30)& (allBlobEquivDiameter > 18);

%7) awacs
allowableAreaIndexes7 = (allBlobAreas > 900)& (allBlobAreas < 1400) &
(allBlobPerimeter > 250)&(allBlobPerimeter < 330) & (allBlobMajorAxisLength<55) &
(allBlobMinorAxisLength>45) & (allBlobEccentricity>0.2) & (allBlobEccentricity<0.45) &
(allBlobEquivDiameter < 45)& (allBlobEquivDiameter > 30);

```

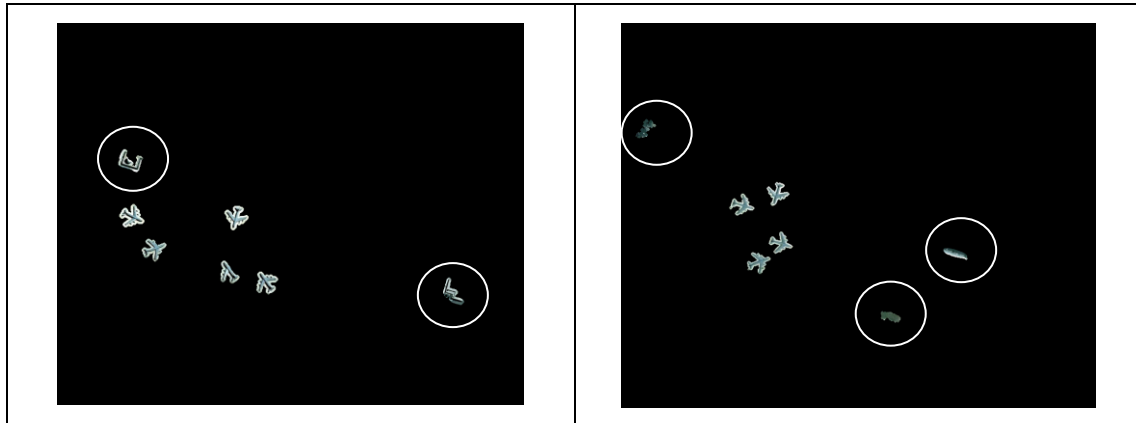
3.1.3.8. Υπολογισμοί MAX (μέγιστο), MIN (ελάχιστο), Μ.Ο (Μέσου Όρου) για κάθε κανάλι και για κάθε οντότητα.

Στο στάδιο αυτό προσπαθούμε να βελτιώσουμε τα αντικείμενα τα οποία έχουν εντοπιστεί (απόρριψη επιπλέον «σκουπιδιών»), με την χρήση των τιμών φωτεινότητας για κάθε κανάλι, από τις αρχικές RGB εικόνες.

Στη σελίδα 38 και συγκεκριμένα στο σχήμα 6 παρουσιάζονται όλα τα χαρακτηριστικά που υπολογίστηκαν, για κάθε μια οντότητα ξεχωριστά. Αυτό που δεν είχαμε χρησιμοποιήσει και θα αξιοποιήσουμε σε αυτό το βήμα είναι ο υπολογισμένος πίνακας "PixeiList". Ο πίνακας αυτός αναπαριστά τις συντεταγμένες όλων των Pixel για την αντίστοιχη οντότητα κατά x και y. Αυτός ο πίνακας μας βοηθάει για να

χρησιμοποιήσουμε τις συγκεκριμένες συντεταγμένες και στην αρχική RGB εικόνα και να υπολογίσουμε την τιμή φωτεινότητας για το κάθε κανάλι.

Η ιδέα που κρύβεται πίσω από αυτό το βήμα έγκειται στην υπόθεση ότι τα αεροπλάνα θα χαρακτηρίζονται από διαφορετικές τιμές φωτεινότητας σε σχέση με τα «σκουπίδια» που, μέχρι τώρα, έχουν περάσει από τους διάφορους ελέγχους.



Πίνακας 8 – Αποτέλεσμα εντοπισμού από το 13^ο βήμα, στο οποίο έχουν προστεθεί και οι τιμές των πολυφασματικών εικόνων(Ζεύγος εικόνων για το Αεροδρόμιο Holladay)

Όπως βλέπουμε και στις παραπάνω εικόνες, υπάρχουν πέντε αντικείμενα τα οποία δεν είναι αεροπλάνα. Τα βήματα για την επεξεργασία των φασματικών ιδιοτήτων των αντικειμένων του Σχήματος 8 είναι τα κάτωθι:

- Υπολογίζονται συνολικά 9 τιμές για κάθε οντότητα. Αυτές είναι η μέγιστη τιμή, η ελάχιστη και ο Μ.Ο, για κάθε κανάλι ξεχωριστά. Εξού και το γεγονός ότι ο πίνακας “Results” που θα αποθηκεύονται έχει διαστάσεις [9*(Σύνολο οντοτήτων = k1a)].

```
Results = zeros(9,k1a);
```

- Στον “Pinaka sidetagmenon” καταχωρούνται για κάθε επανάληψη οι τιμές του πίνακα “PixelList” που εμπεριέχει, όπως προαναφέραμε, τις συντεταγμένες όλων των pixels για κάθε οντότητα.

```
Pinakas_sidetagmenon = (stats1(d).PixelList);  
[k1aa,k2aa] = size(Pinakas_sidetagmenon);  
Synolo_pixels = k1aa;
```

- Μέσω των διαστάσεων του πίνακα “Pinaka sidetagmenon”, εντοπίζεται και το σύνολο των pixels από τα οποία αποτελείται η κάθε οντότητα (k1aa).
- Δημιουργούνται τρεις πίνακες-στήλη, για κάθε χρώμα και με μέγεθος το εκάστοτε (k1aa) σύνολο pixels που αποτελεί την κάθε οντότητα.

```
Red = zeros(k1aa,1);
Green = zeros(k1aa,1);
Blue = zeros(k1aa,1);
```

- Έπειτα καταχωρούνται οι τιμές για κάθε κανάλι στους αντίστοιχους πίνακες, υπολογίζονται οι τιμές που προαναφέραμε μέσω έτοιμων συναρτήσεων και αποθηκεύονται στον πίνακα “Results” στην κατάλληλη θέση για κάθε οντότητα με βάση τον αύξοντα αριθμό που είχε ήδη.

Ακολούθως υπολογίζονται οι συγκεντρωτικές τιμές για κάθε κανάλι, προερχόμενες από τα δεδομένα όλων των οντοτήτων.

```
for a_a = 1 : Synolo_pixels

    j = Pinakas_sidetagmenon(a_a,1);
    i = Pinakas_sidetagmenon(a_a,2);
    %-----
    % Megista gia kathe kanali

    Red(a_a,1)= A0(i, j, 1);
    Green(a_a,1)= A0(i, j, 2);
    Blue(a_a,1)= A0(i, j, 3);
    MaxR = max(Red);
    MaxG = max(Green);
    MaxB = max(Blue);
    MinR = min(Red);
    MinG = min(Green);
    MinB = min(Blue);
    M_O_R = mean(Red);
    M_O_G = mean(Green);
    M_O_B = mean(Blue);

End

Results(1,d) = MaxR;
Results(2,d) = MaxG;
Results(3,d) = MaxB;
Results(4,d) = MinR;
Results(5,d) = MinG;
Results(6,d) = MinB;
Results(7,d) = M_O_R;
Results(8,d) = M_O_G;
Results(9,d) = M_O_B;
% Ypologismos megiston kai elaxiston gia kathe kanali
max_R=max(Results(7,:));
max_G=max(Results(8,:));
max_B=max(Results(9,:));
min_R=min(Results(7,:));
min_G=min(Results(8,:));
min_B=min(Results(9,:));
```

3.1.3.9. Δημιουργία κατωφλίων με βάση τους κανονικοποιημένους Μ.Ο για την εξαγωγή των επιθυμητών στόχων.

Η κανονικοποίηση δεδομένων είναι ένα χρήσιμο μέτρο πρόληψης, που προηγείται της σύγκρισης τους, όταν το εύρος τιμών διαφέρει σημαντικά μεταξύ των οντοτήτων. Μέσω της κανονικοποίησης, οι οντότητες που λαμβάνουν μεγάλες τιμές περιορίζονται σε προκαθορισμένα όρια. Η κανονικοποίηση που χρησιμοποιήθηκε είναι η $[0,2]$.

```
for ii = 1:kla
    Results(7,ii) = ((Results(7,ii)-min_R)/(max_R-min_R) - 0.5) * 2 +1;
    Results(8,ii) = ((Results(8,ii)-min_G)/(max_G-min_G) - 0.5) * 2 +1;
    Results(9,ii) = ((Results(9,ii)-min_B)/(max_R-min_B) - 0.5) * 2 +1;
end
```

Από τις κανονικοποιημένες τιμές εντοπίζονται τα κατώφλια τα οποία δύναται να μας βοηθήσουν στην απόρριψη των αντικειμένων που δεν θέλουμε. Αυτό το επιτυγχάνουμε με την δημιουργία ενός πίνακα στήλη με διαστάσεις ίσες με το μέγεθος των αρχικών οντοτήτων. Στην περίπτωση, που κάποια οντότητα ικανοποιεί και τις τρεις συνθήκες (Μ.Ο κόκκινο κανάλι, Μ.Ο πράσινο κανάλι, Μ.Ο μπλε κανάλι) Τότε καταχωρείται η τιμή “1” στον πίνακα “ID” ενώ στην αντίθετη περίπτωση η τιμή “0”.

Πλέον με την χρήση του πίνακα “ID” μπορούμε να κατασκευάσουμε μια νέα binary εικόνα. Όποια pixels της προηγούμενης binary εικόνας έχουν τιμή διαφορετική από μηδέν (που σημαίνει ότι ήταν εντοπισμένο αντικείμενο) ελέγχονται με βάση τον αύξοντα αριθμό σε συνάρτηση με την τιμή που έχουν στον πίνακα “ID”. Αν έχουν και στον πίνακα “ID” τιμή διάφορη του μηδενός σημαίνει ότι περνούν τον έλεγχο και καταχωρείται στην νέα εικόνα η ποσότητα 255-άσπρο.

```
ID = zeros(kla,1);
% Anazitisi
for i = 1 : kla
    if (Results(7,i)> f11)
        if (Results(8,i) > f22)
            if (Results(9,i)> f33)
                ID(i,1) = 1;
            else
                ID(i,1) = 0;
            end
        else
            ID(i,1) = 0;
        end
    end
end
```

```

else
    ID(i,1) = 0;
end
end
%create new bw L1
for i = 1 : k1
    for j = 1 : k2
        if (L1(i,j) ~= 0)
            if (ID(L1(i,j),1) ~= 0)
                L1(i,j) = 255;
            else
                L1(i,j) = 0;
            end
        else
            L1(i,j) = 0;
        end
    end
end
end
end
end

```

Το τελικό αποτέλεσμα έχει απορρίψει το μεγαλύτερο ποσοστό των αντικειμένων που δεν αποτελούσαν αεροπλάνα. Στις παρακάτω εικόνες φαίνονται σε αντιπαραβολή τα τελικά αποτελέσματα εντοπισμού τόσο από την κατωφλίωση με βάση γεωμετρικά χαρακτηριστικά όσο και με την συνέχιση της κατωφλίωσης με χρήση της αρχικής φασματικής πληροφορίας.



Πίνακας 9–Αντιπαραθεση αποτελεσμάτων εντοπισμού μεταξύ χρήσης γεωμετρικής πληροφορίας (ΑΡ) και γεωμετρικής –φασματικής πληροφορίας (ΔΕΞ) στα οποία έχουν προστεθεί και οι τιμές των αρχικών πολυφασματικών εικόνων(Αεροδρόμιο Holladay)

3.2. Ανάπτυξη αλγορίθμου με βάση τεχνικές αναγνώρισης προτύπων και διανυσμάτων υποστήριξης.

3.2.1. Εισαγωγή και Θεωρητικό Υπόβαθρο

Στο δεύτερο μέρος της διπλωματικής εργασίας, χρησιμοποιείται ένα νευρωνικό δίκτυο τύπου SVM (Support Vector Machine) για τον εντοπισμό των αεροσκαφών. Θα γίνει αναφορά στα βασικά χαρακτηριστικά των νευρωνικών δικτύων καθώς επίσης και στις αρχές πάνω στις οποίες βασίζονται οι «Μηχανές Υποστήριξης Διανυσμάτων» ενώ θα αναλυθεί σε βάθος ο κώδικας και η γενικότερη φιλοσοφία που ακολουθήθηκε για τον εντοπισμό των αεροσκαφών. Τέλος, θα παρουσιαστούν τα αποτελέσματα του κώδικα καθώς επίσης και η ακρίβεια αυτών με βασικούς δείκτες απόδοσης.

3.2.2. Μηχανές διανυσματικής Υποστήριξης

Τεχνητή Νοημοσύνη

Ο όρος τεχνητή νοημοσύνη (Artificial Intelligence) αναφέρεται στον κλάδο της επιστήμης υπολογιστών ο οποίος ασχολείται με τη σχεδίαση και την υλοποίηση υπολογιστικών συστημάτων που μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς τα οποία υπονοούν έστω και στοιχειώδη ευφυΐα: μάθηση, προσαρμοστικότητα, εξαγωγή συμπερασμάτων, κατανόηση από συμφραζόμενα, επίλυση προβλημάτων κ.ά.

Η τεχνητή νοημοσύνη αποτελεί χώρο δράσης πολλών επιστημονικών πεδίων, όπως της επιστήμης υπολογιστών, της ψυχολογίας, της φιλοσοφίας, της νευρολογίας, της γλωσσολογίας και της επιστήμης μηχανικών, με στόχο τη σύνθεση ευφυούς συμπεριφοράς, με στοιχεία συλλογιστικής μάθησης και προσαρμογής στο περιβάλλον. Συνήθως εφαρμόζεται σε μηχανές ή υπολογιστές ενώ διαιρείται στη συμβολική τεχνητή νοημοσύνη και στην υποσυμβολική τεχνητή νοημοσύνη. Η συμβολική επιχειρεί να εξομοιώσει την ανθρώπινη νοημοσύνη αλγοριθμικά χρησιμοποιώντας σύμβολα και λογικούς κανόνες υψηλού επιπέδου ενώ η υποσυμβολική προσπαθεί να αναπαράγει την ανθρώπινη ευφυΐα χρησιμοποιώντας στοιχειώδη αριθμητικά μοντέλα που συνθέτουν επαγωγικά νοήμονες συμπεριφορές με τη διαδοχική αυτοοργάνωση απλούστερων δομικών συστατικών (συμπεριφορική τεχνητή νοημοσύνη) που προσομοιώνουν πραγματικές βιολογικές διαδικασίες ή αποτελούν εφαρμογή στατιστικών μεθοδολογιών σε προβλήματα TN.

Η διάκριση σε συμβολικές και υποσυμβολικές προσεγγίσεις αφορά τον χαρακτήρα των χρησιμοποιούμενων εργαλείων, ενώ δεν είναι σπάνια η σύζευξη πολλαπλών προσεγγίσεων (διαφορετικών συμβολικών, υποσυμβολικών, ή ακόμα συμβολικών και υποσυμβολικών μεθόδων) κατά την προσπάθεια αντιμετώπισης ενός προβλήματος. Με βάση τον επιθυμητό επιστημονικό στόχο, η ΤΝ κατηγοριοποιείται σε άλλου τύπου ευρείς τομείς, όπως επίλυση προβλημάτων, μηχανική μάθηση, ανακάλυψη γνώσης, συστήματα γνώσης κλπ. Επίσης, υπάρχει επικάλυψη με συναφή επιστημονικά πεδία όπως η μηχανική όραση, η επεξεργασία φυσικής γλώσσας, η ρομποτική κλπ.

Νευρωνικό Δίκτυο

Νευρωνικό δίκτυο ονομάζεται ένα κύκλωμα διασυνδεδεμένων νευρώνων. Στην περίπτωση βιολογικών νευρώνων, πρόκειται για ένα τμήμα νευρικού ιστού. Πάνω στην ίδια φιλοσοφία η περίπτωση των τεχνητών νευρώνων, αφορά ένα αφηρημένο αλγοριθμικό κατασκεύασμα το οποίο εμπίπτει στον τομέα της υπολογιστικής νοημοσύνης (Artificial Intelligence), όταν στόχος του νευρωνικού δικτύου είναι η επίλυση κάποιου υπολογιστικού προβλήματος με βάση πάντα κάποιο μαθηματικό μοντέλο.

Ουσιαστικά αποτελεί ένα δίκτυο από απλούς υπολογιστικούς κόμβους (νευρώνες, νευρώνια) διασυνδεδεμένους μεταξύ τους και είναι εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα (ΚΝΣ), το οποίο προσπαθεί να προσομοιώσει.

Νευρώνας

Οι νευρώνες είναι τα δομικά στοιχεία του νευρωνικού δικτύου. Κάθε τέτοιος κόμβος δέχεται ένα σύνολο αριθμητικών εισόδων από διαφορετικές πηγές (είτε από άλλους νευρώνες, είτε από το περιβάλλον), επιτελεί έναν υπολογισμό με βάση αυτές τις εισόδους και παράγει μία έξοδο. Η εν λόγω έξοδος είτε κατευθύνεται στο περιβάλλον (χωρίς ανατροφοδότηση), είτε τροφοδοτείται ως είσοδος σε άλλους νευρώνες του δικτύου (ανατροφοδότηση). Υπάρχουν τρεις τύποι νευρώνων: οι νευρώνες εισόδου⁽¹⁾, οι νευρώνες εξόδου⁽²⁾ και οι υπολογιστικοί νευρώνες ή κρυμμένοι νευρώνες⁽³⁾. Οι νευρώνες εισόδου δεν επιτελούν κανέναν υπολογισμό, μεσολαβούν απλώς ανάμεσα στις περιβαλλοντικές εισόδους του δικτύου και στους υπολογιστικούς νευρώνες. Οι νευρώνες εξόδου διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου. Οι υπολογιστικοί νευρώνες

πολλαπλασιάζουν κάθε είσοδό τους με το αντίστοιχο βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων. Το άθροισμα αυτό τροφοδοτείται ως όρισμα στη συνάρτηση ενεργοποίησης, την οποία υλοποιεί εσωτερικά κάθε κόμβος. Η τιμή που λαμβάνει η συνάρτηση για το εν λόγω όρισμα είναι και η έξοδος του νευρώνα για τις τρέχουσες εισόδους και βάρη.

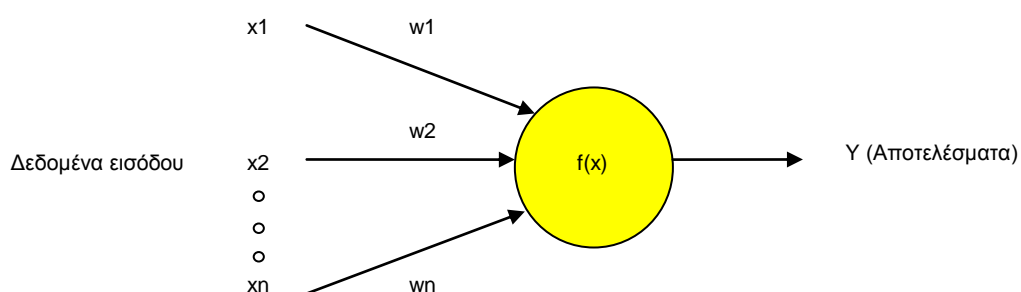
Ο νευρώνας "Perceptron» είναι ένα είδος τεχνητού νευρωνικού δικτύου που εφευρέθηκε το 1957 στο Αεροναυτικό Εργαστήριο του Κορνέλ (Cornell Aeronautical Laboratory) από τον Φρανκ Ρόζενμπλαττ (Frank Rosenblatt). Δύναται να χαρακτηριστεί ως ένα απλό είδος ενός εμπροσθοτροφοδοτούμενου (feed - forward) νευρωνικού δικτύου. Ο αλγόριθμος perceptron είναι κατάλληλος στην περίπτωση προβλήματος δύο κλάσεων που είναι γραμμικώς διαχωρίσιμες. (linear classifier).

Οι νευρώνες χρησιμοποιούν τα ιδιοδιανύσματα πίνακα, για να αναπαραστήσουν εμπροσθο-τροφοδοτούμενα νευρωνικά δίκτυα και αποτελούν ουσιαστικά ταξινομητές τρίτου βαθμού οι οποίοι απεικονίζουν την είσοδο x σε μία μοναδική δυαδική τιμή εξόδου $f(x)$:

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0 \\ 0 & \text{else} \end{cases}$$

Εικόνα 12 - Παράδειγμα συνάρτησης ενεργοποίησης

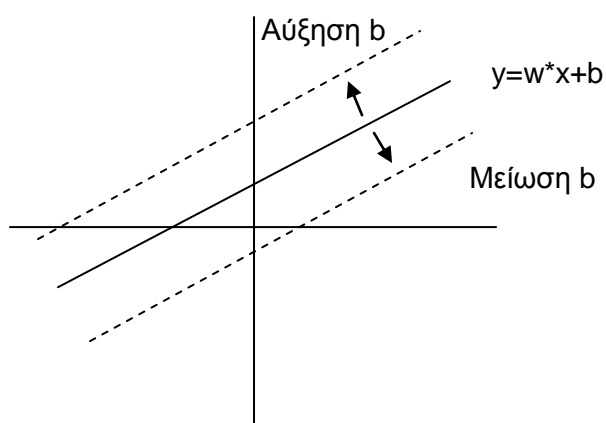
όπου "w" είναι ένα διάνυσμα από βάρη με πραγματικές τιμές και "w*x" είναι το εσωτερικό γινόμενο μεταξύ των διανυσμάτων. Το "b" είναι ένας σταθερός όρος (κατώφλι), ο οποίος δεν εξαρτάται από καμία τιμή εισόδου. Όπως καταλαβαίνουμε, υπολογιστικά το παραπάνω άθροισμα είναι αρκετά βεβαρημένο.



Σχήμα 7 - Διάγραμμα λειτουργίας νευρώνα

Η τιμή της $f(x)$ χρησιμοποιείται για να ταξινομήσει τα δεδομένα εισόδου (x) είτε ως θετικό είτε ως αρνητικό στιγμιότυπο, στην περίπτωση ενός δυαδικού προβλήματος

ταξινόμησης. Η μεταβλητή b χρησιμοποιείται για την μετατόπιση της συνάρτησης ενεργοποίησης ή για να δώσει στον νευρώνα εξόδου ένα βασικό επίπεδο δραστηριότητας. Αν το βάρος είναι αρνητικό τότε ο βεβαρημένος συνδυασμός των εισόδων πρέπει να παραγάγει μία θετική τιμή μεγαλύτερη του έτσι ώστε να αναγκάσει τον νευρώνα που ταξινομεί να έχει τιμή άνω του κατωφλίου 0. Χωρικά, το b μεταβάλλει την θέση (αλλά όχι τον προσανατολισμό) του συνόρου απόφασης.



Σχήμα 8- Διάγραμμα αναπαράστασης της $f(x)$ συναρτήσεως της b μεταβλητής (κατώφλι).

Εφόσον οι εισοδοί τροφοδοτούνται στο δίκτυο άμεσα μέσω των βεβαρημένων συνδέσεων, ο νευρώνας μπορεί να θεωρηθεί ως ένα απλό είδος νευρωνικού δικτύου εμπρός τροφοδότησης.

Τρόπος λειτουργίας νευρώνα

Μια από τις πιο βασικές ιδιότητες των Νευρωνικών Δικτύων είναι η ικανότητά τους για εκπαίδευση (training). Η εκπαίδευση αυτή επιτυγχάνεται μέσω της ανταλλαγής τιμών και βαρών, που αποσκοπεί στη βαθμιαία σύλληψη της πληροφορίας η οποία στη συνέχεια θα είναι διαθέσιμη προς ανάκτηση. Υπάρχουν, βέβαια, πολλοί αλγόριθμοι που η εφαρμογή τους έχει στόχο την προσαρμογή των τιμών των βαρών ενός Τεχνητού Νευρωνικού Δικτύου. Όλες οι μέθοδοι μάθησης μπορούν να καταταχθούν σε δύο βασικές κατηγορίες : τη μάθηση με επίβλεψη (supervised learning) και τη μάθηση χωρίς επίβλεψη (unsupervised learning).

Επιβλεπόμενη μάθηση

Η μάθηση αυτή είναι μια διαδικασία η οποία συνδυάζει έναν εξωτερικό εκπαιδευτή και τη συνολική ή γενικευμένη πληροφορία. Κάποιες από τις μεθόδους οι οποίες

συγκαταλέγονται σε αυτή την κατηγορία είναι η μάθηση με διόρθωση σφάλματος, η στοχαστική μάθηση. Παραδείγματα τα οποία αντιπροσωπεύουν την μάθηση με επίβλεψη συμπεριλαμβάνουν αποφάσεις για το πότε θα πρέπει να σταματήσει η διαδικασία εκπαίδευσης, αποφάσεις αναφορικά με τη συχνότητα παρουσίασης στο δίκτυο, πρότυπα εκπαίδευσης και παρουσίαση προόδου του δικτύου.

Η μάθηση με επίβλεψη χωρίζεται σε δύο ακόμα κατηγορίες: στη δομική (structural) και στην προσωρινή (temporal) εκμάθηση. Οι αλγόριθμοι οι οποίοι βρίσκονται στην πρώτη κατηγορία, χρησιμοποιούνται για την εύρεση της βέλτιστης σχέσης μεταξύ εισόδων και εξόδων για κάθε ξεχωριστό ζευγάρι προτύπων. Παραδείγματα της δομικής εκμάθησης αποτελούν η αναγνώριση και η κατηγοριοποίηση προτύπων, ενώ παραδείγματα της προσωρινής εκμάθησης η πρόβλεψη και ο έλεγχος.

Μη επιβλεπόμενη μάθηση

Οι αλγόριθμοι της εν λόγω μάθησης αναφέρονται ως αυτό-οργανώμενοι (self-organized) και είναι διαδικασίες οι οποίες δεν απαιτούν να είναι παρών ένας «εξωτερικός» δάσκαλος ή επιβλέπων. Βασίζονται, μάλιστα, μόνο σε τοπική πληροφορία καθ' όλη τη διάρκεια της εκπαίδευσης του Τεχνητού Νευρωνικού Δικτύου. Οι συγκεκριμένοι αλγόριθμοι οργανώνουν τα δεδομένα και ανακαλύπτουν τις σημαντικές συλλογικές ιδιότητες. Για παράδειγμα, αλγόριθμοι εκπαίδευσης χωρίς επίβλεψη είναι ο αλγόριθμος Hebbian, ο διαφορικός αλγόριθμος Hebbian και ο Min-Max αλγόριθμος.

Οι περισσότερες διαδικασίες εκπαίδευσης είναι off line. Off line ονομάζονται οι διαδικασίες μάθησης όταν χρησιμοποιείται όλο το δείγμα προτύπων για την τροποποίηση των τιμών των βαρών, πριν την τελική χρήση του δικτύου ως εφαρμογή. Οι αλγόριθμοι εκπαίδευσης off line έχουν την απαίτηση να βρίσκονται στην εκπαίδευση του δικτύου παρόντα όλα τα πρότυπα. Το γεγονός αυτό αποκλείει την πιθανότητα εισαγωγής νέων πληροφοριών μέσω νέων προτύπων. Βέβαια, υπάρχουν και Τεχνητά Νευρωνικά Δίκτυα τα οποία δεν αποκλείουν την εισαγωγή νέας πληροφορίας, μετά την τελική τους μοντελοποίηση. Αν παρουσιαστεί ανάγκη εισαγωγής νέου προτύπου στο δίκτυο, μπορεί να γίνει απευθείας χωρίς τον κίνδυνο να χαθεί κανένα μέρος της αρχικής πληροφορίας. Το πλεονέκτημα των δικτύων που χρησιμοποιούν off line διαδικασίες εκπαίδευσης επικεντρώνεται κυρίως στη δυνατότητα να δίνουν καλύτερες λύσεις σε δύσκολα προβλήματα.

Αρχικά, θα πρέπει να εκπαιδεύσουμε τον νευρώνα με κάποια δεδομένα εκπαίδευσης. Αυτό το στάδιο χρειάζεται για να αποκτήσει ο νευρώνας την απαραίτητη γνώση, για το πώς να αντιδρά σε περίπτωση που βλέπει κάποιο συγκεκριμένο πρότυπο (pattern) και ονομάζεται «Στάδιο Εκπαίδευσης». Όταν ο νευρώνας αποκτήσει κάποια γνώση σημαίνει ότι τα βάρη του και η μεταβλητή b αλλάζουν. Αφού ολοκληρωθεί το στάδιο της εκπαίδευσης τα βάρη (w) όπως και η σταθερά b έχουν πλέον οριστικοποιηθεί.

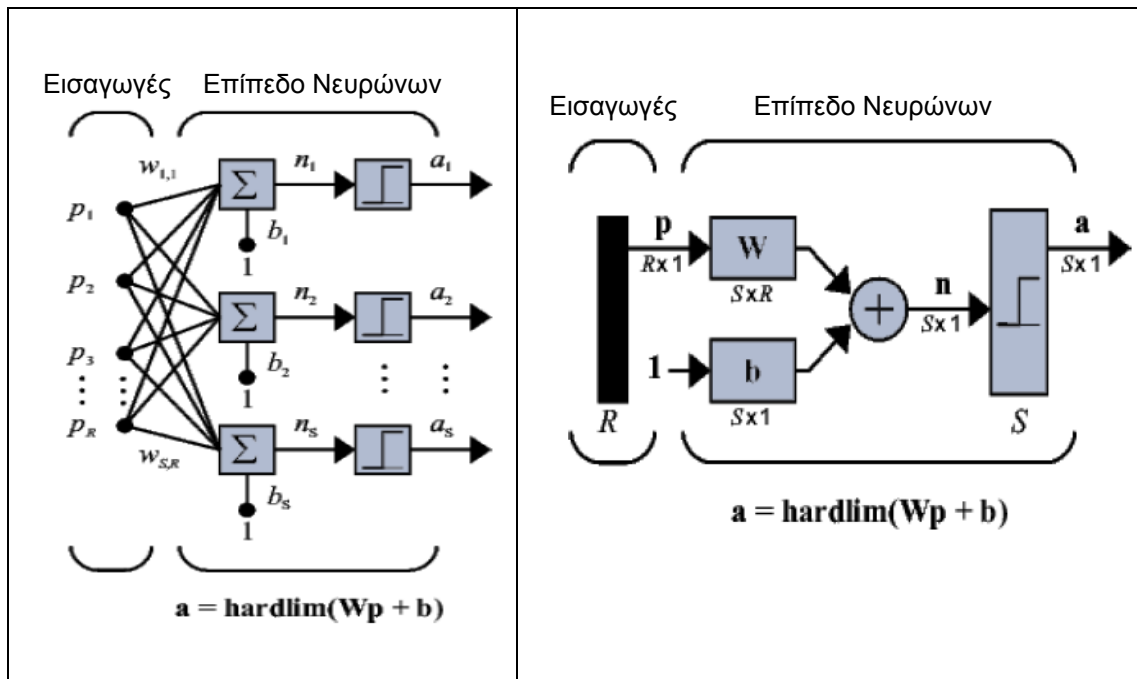
Εκπαίδευση νευρώνα.

Το κύριο χαρακτηριστικό των νευρωνικών δικτύων είναι η εγγενής ικανότητα μάθησης. Ως μάθηση μπορεί να οριστεί η σταδιακή βελτίωση της ικανότητας του δικτύου να επιλύει κάποιο πρόβλημα (π.χ. η σταδιακή προσέγγιση μίας συνάρτησης). Η μάθηση επιτυγχάνεται μέσω της εκπαίδευσης, μίας επαναληπτικής διαδικασίας σταδιακής προσαρμογής των παραμέτρων του δικτύου (συνήθως των βαρών και της πόλωσης του) σε τιμές κατάλληλες ώστε να επιλύεται με επαρκή επιτυχία το προς εξέταση πρόβλημα. Αφού ένα δίκτυο εκπαιδευτεί, οι παράμετροί του συνήθως «παγώνουν» στις κατάλληλες τιμές και από εκεί κι έπειτα είναι σε λειτουργική κατάσταση. Το ζητούμενο είναι το λειτουργικό δίκτυο να χαρακτηρίζεται από μία ικανότητα γενίκευσης: αυτό σημαίνει πως δίνει ορθές εξόδους για εισόδους καινοφανείς και διαφορετικές από αυτές με τις οποίες εκπαιδεύτηκε.

Πλέον, αφού έχει ολοκληρωθεί το στάδιο της εκπαίδευσης, μπορούμε να τροφοδοτήσουμε τον νευρώνα με οποιαδήποτε είσοδο θέλουμε. Αξιοσημείωτο είναι το γεγονός, ότι ακόμα και αν η είσοδος που θα του δώσουμε είναι ένα πρότυπο που δεν έχει καμία απολύτως σχέση με τα δεδομένα εκπαίδευσης, ο νευρώνας θα καταφέρει να απαντήσει. Η δυνατότητα να δίνει απαντήσεις ανεξάρτητα αν έχει ξαναδεί το συγκεκριμένο πρότυπο αποτελεί και την δύναμη του τεχνητού νευρώνα. Για να εκπαιδεύσουμε έναν νευρώνα θα πρέπει να προετοιμάσουμε ένα πακέτο εκπαίδευσης (training set). Το πακέτο εκπαίδευσης αποτελείται από διανύσματα εισόδου και τις επιθυμητές εξόδους τους. Ουσιαστικά τροφοδοτούν τον νευρώνα σαν ζευγάρια πληροφοριών. Κάθε εισαγωγή έχει και το δικό της στόχο (target).

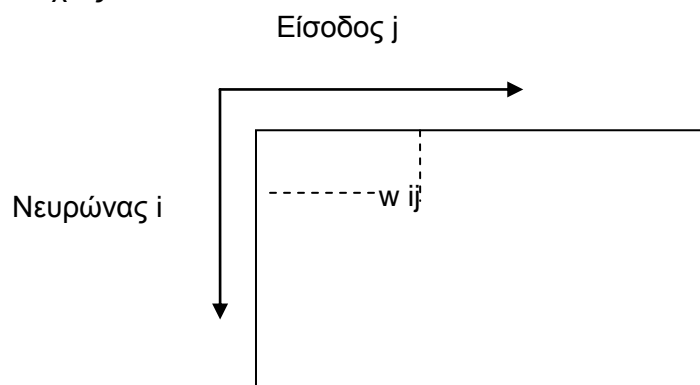
Νευρώνες ενός επιπέδου

Με τα παρακάτω σχήματα, θα δούμε την λειτουργία νευρώνων. Για λόγους κατανόησης οι νευρώνες έχουν υπεραπλουστευτεί αποτελούμενοι από ένα και μοναδικό επίπεδο.



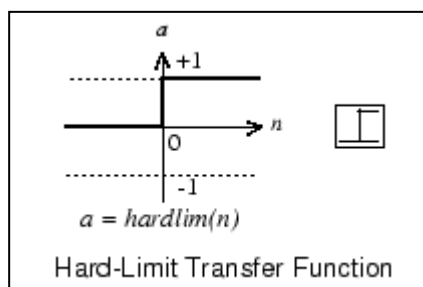
Σχήμα 9- Διάγραμμα Νευρώνων ενός επιπέδου

Στο παραπάνω διάγραμμα νευρώνα ενός επιπέδου, τα δεδομένα εισόδου αποτελούν ένα διάνυσμα (πίνακας $R \times 1$) από R στοιχεία ενώ το S υποδηλώνει τον αριθμό των νευρώνων στο συγκεκριμένο επίπεδο. Όπως βλέπουμε παραπάνω, R εισοδοι, συνδέονται με τα αντίστοιχα βάρη τους w_{ij} τα οποία έχουν πλέον καταχωρηθεί σε έναν πίνακα i και j διαστάσεων. Οι δείκτες i και j μας ορίζουν τον νευρώνα και την είσοδο αντιστοίχως.



Σχήμα 10- Πίνακας βαρών

Με την ίδια φιλοσοφία πλέον δεν έχουμε ένα και μοναδικό κατώφλι αλλά έναν διάνυσμα κατωφλίων που εκφράζεται από τον πίνακα διαστάσεων S^*1 .



Σχήμα 11 – Συνάρτηση Ενεργοποίησης

Η συνάρτηση Hard-Limit Transfer Function είναι μια συνάρτηση μεταφοράς νευρωνικών δικτύων η οποία εξάγει ανάλογα με το κατώφλι δύο τιμές. Εάν η τιμή είναι μεγαλύτερη από 0 εξάγει 1 και αν όχι εξάγει -1.

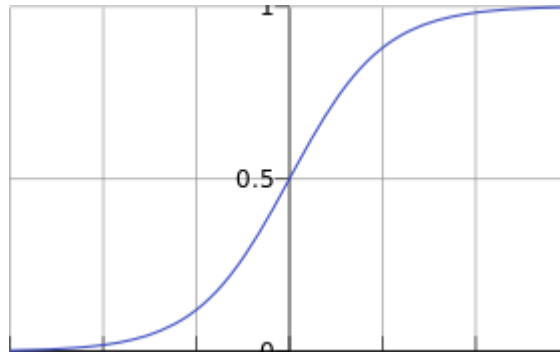
Η συνάρτηση ενεργοποίησης μπορεί να είναι βηματική (step transfer function), γραμμική (linear transfer function), ή μη γραμμική (non-linear transfer function).

- **Βηματική συνάρτηση ενεργοποίησης**

Η βηματική συνάρτηση ενεργοποίησης μπορεί να είναι (ή οποιαδήποτε άλλη βηματική συνάρτηση):

$$\phi(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

Η βηματική συνάρτηση δεν θεωρείται χρήσιμη ως συνάρτηση ενεργοποίησης στα ΤΝΔ (Τεχνητά Νευρωνικά Δίκτυα), καθώς σύμφωνα με τον απειροστικό λογισμό έχει το βασικό μειονέκτημα να είναι η παράγωγός της ίση με μηδέν. Έτσι προέκυψε η ανάγκη συναρτήσεων ενεργοποίησης που η γραφική τους παράσταση να μοιάζει με τη βηματική, αλλά ταυτόχρονα να είναι συνεχείς και παραγωγίσιμες σε όλο το πεδίο ορισμού τους. Τέτοια συνάρτηση είναι και η σιγμοειδής.



Σχήμα 12 - Λογιστική σιγμοειδής συνάρτηση

- **Γραμμική συνάρτηση ενεργοποίησης**

Η γραμμική συνάρτηση ενεργοποίησης μπορεί να είναι (ή οποιαδήποτε άλλη βηματική συνάρτηση):

$$\phi(x) = x$$

- **Μη γραμμική συνάρτηση ενεργοποίησης**

Η μη γραμμική συνάρτηση ενεργοποίησης που χρησιμοποιείται συνήθως στα νευρωνικά δίκτυα καλείται σιγμοειδής συνάρτηση. Οι τυπικές σιγμοειδείς είναι δύο:

Λογιστική σιγμοειδής	Υπερβολική εφαπτομένη
$\phi(x) = \frac{1}{1 + e^x}$	$\phi(x) = \tanh x$

Πίνακας 10 - Πίνακας Τυπικών Σιγμοειδών Συναρτήσεων

Πολλαπλών επιπέδων, εμπροσθοτροφοδοτούμενο, back propagation, δίκτυο.

Εάν είναι x_i η v -οστή είσοδος του νευρώνα, w_i το v -οστό συναπτικό βάρος του νευρώνα και ϕ η συνάρτηση ενεργοποίησης του νευρωνικού δικτύου, τότε η έξοδος y_i του νευρώνα δίνεται από την εξίσωση:

$$y_k = \phi \left(\sum_{i=0}^N x_{ki} w_{ki} \right)$$

Στο ν-οστό νευρώνα υπάρχει ένα συναπτικό βάρος με ιδιαίτερη σημασία, το οποίο καλείται πόλωση ή κατώφλι (bias, threshold). Η τιμή της εισόδου του είναι πάντα η μονάδα. Εάν το συνολικό άθροισμα από τις υπόλοιπες εισόδους του νευρώνα είναι μεγαλύτερο από την τιμή αυτή, τότε ο νευρώνας ενεργοποιείται. Εάν είναι μικρότερο, τότε ο νευρώνας παραμένει ανενεργός. Η ιδέα προέκυψε από τα βιολογικά νευρικά κύτταρα.

Όπως είναι φανερό, οι αριθμοί οι οποίοι συναποτελούν το διάνυσμα εισόδου (κάθε στοιχείο του διανύσματος τροφοδοτείται κατά τη λειτουργία του δικτύου σε έναν νευρώνα εισόδου), αλλά και οι αριθμοί οι οποίοι συναποτελούν το διάνυσμα εξόδου (κάθε στοιχείο του οποίου εμφανίζεται, μετά το πέρας του ολικού υπολογισμού, σε έναν νευρώνα εξόδου), περιγράφουν χαρακτηριστικά του προς επίλυση προβλήματος. Συνήθως αυτό που μας ενδιαφέρει είναι το δίκτυο να απεικονίζει με ορθό τρόπο διανύσματα εισόδου σε κατάλληλα διανύσματα εξόδου, το πρόβλημα δηλαδή είναι η υλοποίηση μίας συνάρτησης πολλαπλών μεταβλητών, κατά κανόνα περίπλοκης, και με άγνωστο ακριβή τύπο. Τέτοιες απεικονίσεις έχουν εφαρμογή σε ποικιλία τομέων της επιστήμης και της τεχνολογίας, αφού λειτουργούν ως αριθμητικά μοντέλα για πολλά διαφορετικά ζητήματα. Το ίδιο δίκτυο μπορεί να υλοποιήσει άπειρες διαφορετικές απεικονίσεις, μία για κάθε διαφορετική επιλογή συνόλου συναπτικών βαρών.

Βασικές αρχές των Γραμμικών Μηχανών Διανυσματικής Στήριξης (Support Vector Machines – SVM)

Η SVM είναι μια χρήσιμη τεχνική για την ταξινόμηση των δεδομένων. Ακόμα κι αν τα Νευρωνικά Δίκτυα θεωρούνται ότι είναι ευκολότερα στη χρήση από αυτή, μερικές φορές λαμβάνονται μη ικανοποιητικά αποτελέσματα. Μια εργασία ταξινόμησης περιλαμβάνει συνήθως δεδομένα εκπαίδευσης και εξέτασης που αποτελούνται από κάποιες περιπτώσεις (στιγμιότυπα) δεδομένων. Κάθε στιγμιότυπο στο σύνολο της κατάρτισης περιέχει μία τιμή-στόχο και διάφορα χαρακτηριστικά. Ο στόχος της SVM είναι να παράγει ένα μοντέλο το οποίο προβλέπει την τιμή-στόχο των δεδομένων στο σύνολο των δοκιμών. Η κατάταξη στην SVM είναι ένα παράδειγμα μάθησης με πλήρη επίβλεψη. Γνωστές ετικέτες βοηθάνε στην αναφορά εάν το σύστημα εκτελείται σε σωστό δρόμο ή όχι. Αυτή η πληροφορία παραπέμπει σε μια επιθυμητή απάντηση,

την επικύρωση της ακρίβειας του συστήματος, ή να χρησιμοποιείται για να βοηθήσει το σύστημα να μάθει να ενεργεί σωστά. Ένα βήμα για την ταξινόμηση της SVM περιλαμβάνει την αναγνώριση και είναι άρρηκτα συνδεδεμένο με τις γνωστές κατηγορίες. Αυτό ονομάζεται επιλογή χαρακτηριστικών ή εξαγωγή χαρακτηριστικών. Η δυνατότητα επιλογής και ταξινόμησης της SVM έχει από κοινού χρήση, ακόμη και όταν η πρόβλεψη των άγνωστων δειγμάτων δεν είναι απαραίτητη. Μπορούν να χρησιμοποιηθούν για να προσδιορίσουν τα βασικά σύνολα που εμπλέκονται στις διεργασίες για διάκριση μεταξύ των τάξεων.

Η SVM έχει βρεθεί να είναι επιτυχής όταν χρησιμοποιείται για προβλήματα ταξινόμησης. Η εφαρμογή της προσέγγισης του Support Vector σε ένα συγκεκριμένο πρακτικό πρόβλημα αφορά την επίλυση μια σειράς από ερωτήματα με βάση τον ορισμό του προβλήματος και τον σχεδιασμό που εμπλέκονται με αυτό. Μία από τις μεγαλύτερες προκλήσεις είναι η επιλογή κατάλληλου πυρήνα για τη συγκεκριμένη εφαρμογή. Υπάρχουν επιλογές, όπως η Gaussian ή το πολυώνυμο του πυρήνα που είναι προεπιλεγμένες αλλά, αν αυτά αποδειχθούν αναποτελεσματικά ή αν οι είσοδοι είναι διακριτές δομές, θα χρειαστούν πιο περίτεχνοι πυρήνες. Με σιωπηρό καθορισμό ενός χώρου χαρακτηριστικών, ο πυρήνας παρέχει τη χαρακτηριστική γλώσσα που χρησιμοποιείται από τη μηχανή για την προβολή των δεδομένων. Μετά την επιλογή του κριτηρίου του πυρήνα και τη βελτιστοποίηση έχουν τεθεί οι βασικές συνιστώσες του συστήματος.

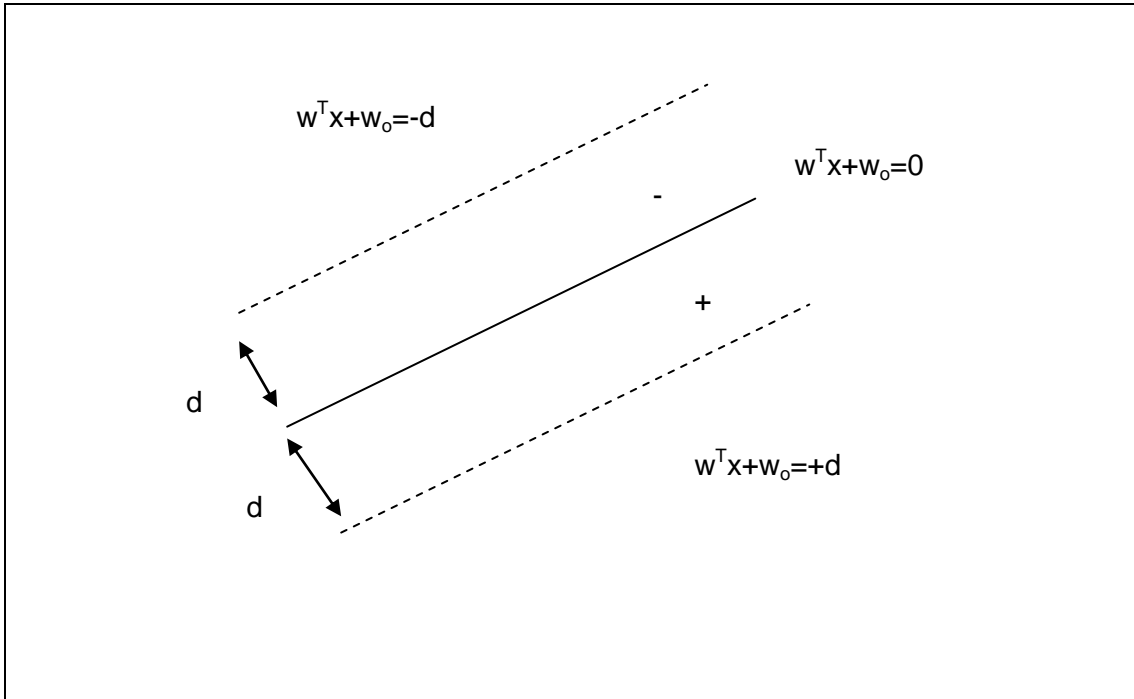
Στην καρδιά του σχεδιασμού κάθε SVM ταξινομητή βρίσκεται η έννοια του περιθωρίου (margin). Θεωρούμε τον γραμμικό ταξινομητή

$$w^T x + w_0 = 0$$

Το περιθώριο είναι η περιοχή μεταξύ δύο παράλληλων υπερεπιπέδων

$$w^T x + w_0 = +1 \text{ και } w^T x + w_0 = -1$$

Η Ευκλείδεια απόσταση μεταξύ οποιουδήποτε σημείου πάνω σε ένα από τα δύο προηγούμενα υπερεπίπεδα και του υπερεπιπέδου του παραπάνω γραμμικού ταξινομητή, είναι ίση με $1/\|w\|$, όπου $\| \cdot \|$ είναι η Ευκλείδεια νόρμα.

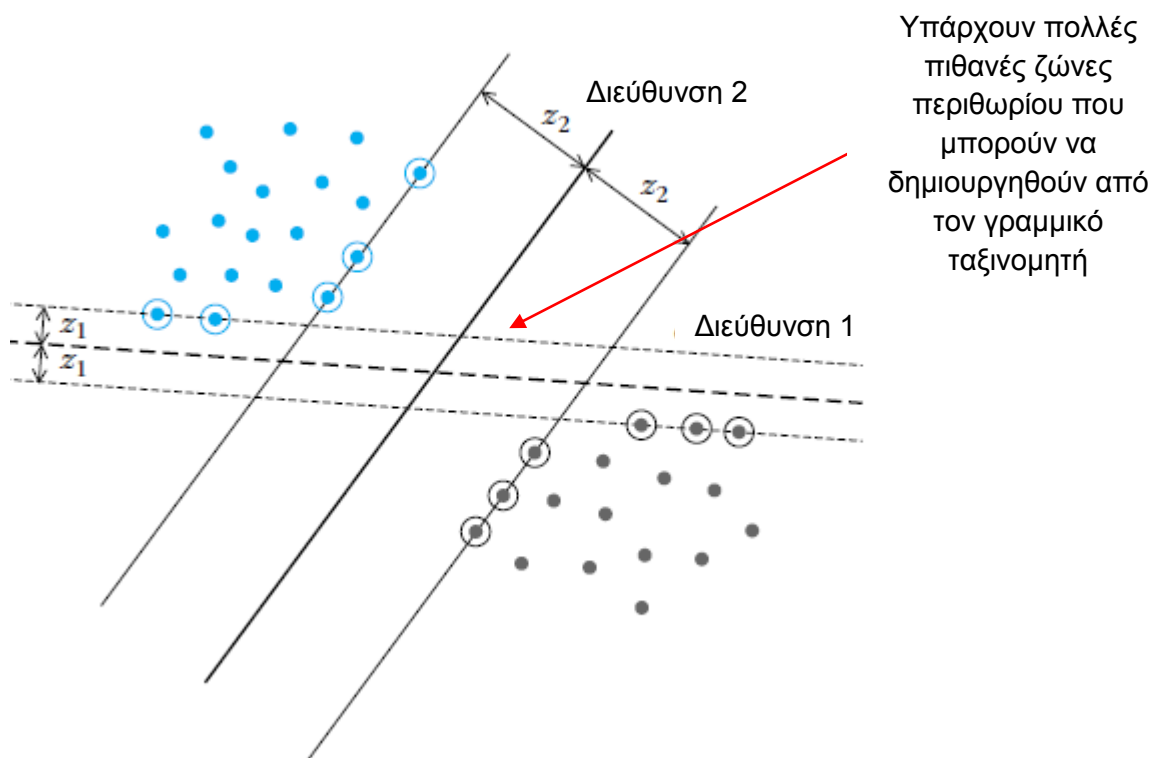


Σχήμα 13 - Γραμμή και το περιθώριο, μεγέθους $2d$

Παρατήρηση: Το γεγονός ότι το περιθώριο ορίζεται με τους «μαγικούς» αριθμούς $+1$ και -1 δεν θα πρέπει να μας προβληματίζει. Έστω ότι τα δύο υπερεπίπεδα ορίζονταν από την σχέση:

$$w^T x + w_0 = +d$$

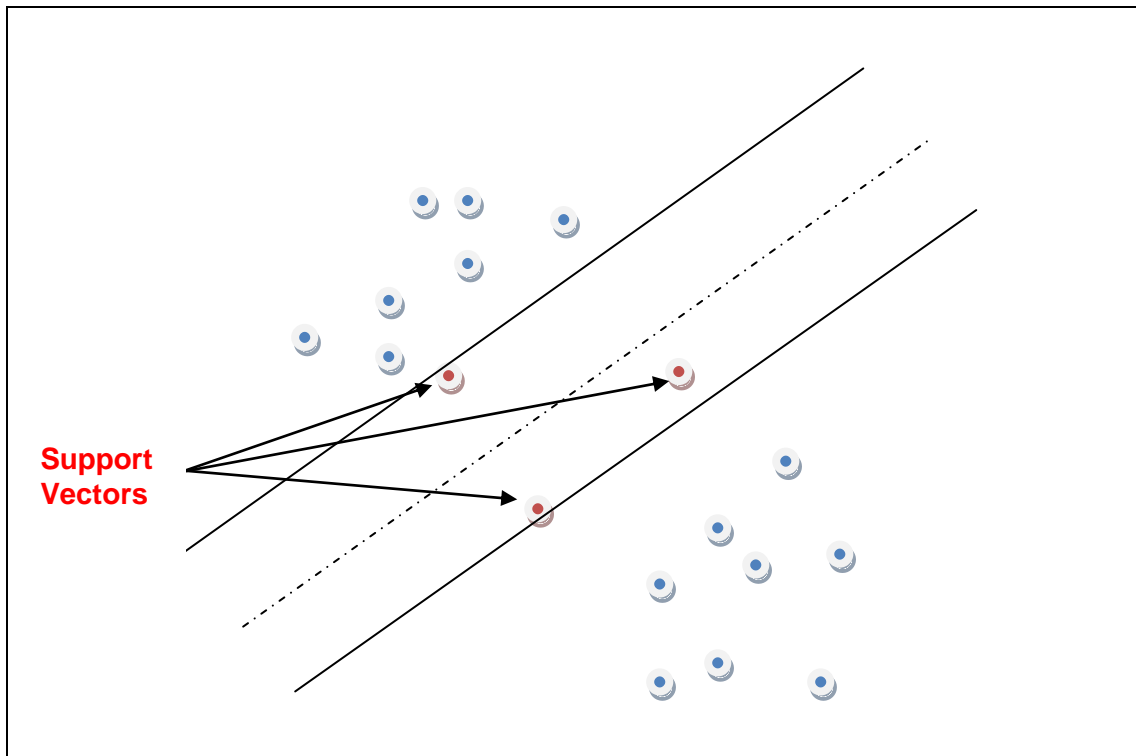
Η τιμή d δύναται να πάρει οποιαδήποτε τιμή, που σημαίνει ότι τα δύο υπερεπίπεδα μπορούν να πλησιάσουν μεταξύ τους ή να απομακρυνθούν ανάλογα του d . Στην περίπτωση που διαιρέσουμε και τα δύο μέλη με την τιμή d , τότε στο δεύτερο μέλος έχουμε και πάλι $+1$ και -1 . Η κανονικοποίηση με μια σταθερά d δεν επηρεάζει τα σημεία που βρίσκονται πάνω σε ένα υπερεπίπεδο (και που ουσιαστικά το ορίζουν). Πρακτικά δεν μεταβάλλεται ούτε η διεύθυνση ούτε η θέση στον χώρο των δύο υπερεπιπέδων.



Σχήμα 14 - Δύο γραμμικοί ταξινομητές

Θα πρέπει να επισημάνουμε ότι κατά την ταξινόμηση των σημείων μέσω της διαχωριστικής γραμμής, όπως φαίνεται και στο Σχήμα 14, έχουμε την εσφαλμένη εντύπωση ότι μόνο τα σημεία τα οποία βρίσκονται στην λανθασμένη πλευρά της επιφάνειας απόφασης που ορίζει ο ταξινομητής συνεισφέρουν στην συνάρτηση υπολογισμού σφάλματος. Η πραγματικότητα είναι πολύ διαφορετική, αφού θεωρούνται ως σφάλματα και τα σημεία που βρίσκονται εντός του περιθωρίου ακόμα και αν εντοπίζονται στην σωστή πλευρά του ταξινομητή. Μόνο τα σημεία εκείνα τα οποία βρίσκονται εκτός περιθωρίου και ταυτόχρονα στην σωστή πλευρά του ταξινομητή, δεν συνεισφέρουν στον υπολογισμό του κόστους.

Στην Σχήμα 14 εμφανίζονται δύο γραμμικοί ταξινομητές οι οποίοι έχουν σχεδιαστεί με τέτοιο τρόπο έτσι ώστε να περιλαμβάνουν στις ζώνες περιθωρίου πέντε και έξι σημεία αντίστοιχα. Προφανώς και η «Διεύθυνση 2» είναι πολύ καλύτερη αφού η ζώνη διαχωρισμού είναι μεγαλύτερη και άρα τα δεδομένα δύναται να μετακινηθούν με μεγαλύτερη άνεση στον χώρο, δίχως την ενεργοποίηση οποιασδήποτε λανθασμένης κατάταξης. Αυτός είναι και ο λόγος που η Διεύθυνση 2 παρόλο που έχει μικρότερη απόδοση (αφού έχει 6 σημεία σε αντίθεση με τα 5 της Διεύθυνσης 1), καταλήγει στο τέλος να έχει ανώτερη γενικευμένη απόδοση σε άγνωστα δεδομένα. Στόχος μας είναι η δημιουργία ενός ταξινομητή που να παρουσιάζει πολύ χαμηλό σφάλμα στο σύνολο των δεδομένων εκπαίδευσης αλλά εξίσου υψηλές αποδόσεις σε άγνωστα δεδομένα.



Σχήμα 15 - Διανύσματα Υποστήριξης Απλού Γραμμικού Ταξινομητή (LSVM Linear Support Vector Machine)

Όπως έχουμε προαναφέρει και σε προηγούμενες παραγράφους, ο κύριος στόχος των SVM είναι ο διαχωρισμός των δεδομένων από το υπερσύνολο και η επέκταση σε μη γραμμικά όρια χρησιμοποιώντας το τέχνασμα του πυρήνα. Ανάμεσα σε όλα τα πιθανά υπερσύνολα, η SVM επιλέγει εκείνο όπου η απόσταση του επιπέδου είναι όσο το δυνατόν μεγαλύτερη (για τους λόγους που προαναφέραμε). Το επιθυμητό επίπεδο που μεγιστοποιεί το περιθώριο διχοτομεί επίσης τις γραμμές μεταξύ των πλησιέστερων σημείων στο κυρτό των δύο συνόλων δεδομένων.

Η ελαχιστοποίηση του κόστους συνιστά συμβιβασμό μεταξύ ενός μεγάλου περιθωρίου και ενός μικρού αριθμού σφαλμάτων περιθωρίου. Προκύπτει ότι η λύση δίνεται από έναν ζυγισμένο μέσο των σημείων εκπαίδευσης, δηλαδή:

$$w = \sum_{i=1}^N (\lambda_i * y_i * x_i)$$

Οι συντελεστές λ_i είναι οι Lagrange πολλαπλασιαστές της βελτιστοποίησης και είναι ίσοι με μηδέν για όλα τα σημεία εκτός περιθωρίου που ταυτόχρονα βρίσκονται στην σωστή πλευρά του ταξινομητή. Επομένως, τα σημεία αυτά δεν συνεισφέρουν στον καθορισμό της διεύθυνσης του ταξινομητή. Τα υπόλοιπα σημεία (Διανύσματα

Υποστήριξης) με μη μηδενικά λί, είναι και τα μοναδικά που συνεισφέρουν στην κατασκευή του w .

Η μη γραμμική περίπτωση SVM

Για να εφαρμόσουμε την SVM τεχνική, στη λύση ενός μη γραμμικού προβλήματος ταξινόμησης, υιοθετούμε τη φιλοσοφία της απεικόνισης των διανυσμάτων χαρακτηριστικών σε ένα χώρο μεγαλύτερης διάστασης, όπου αναμένουμε ότι, με μεγάλη πιθανότητα, οι κλάσεις θα είναι γραμμικώς διαχωρίσιμες. Αυτό μας το εγγυάται το γνωστό θεώρημα Cover. Η απεικόνιση έχει ως εξής:

$$x \mapsto \varphi(x) \in H$$

Όπου η διάσταση του H είναι μεγαλύτερη του R και ανάλογα με την επιλογή της (μη γραμμικής) $\varphi(\cdot)$, μπορεί να είναι ακόμα και άπειρη. Επιπλέον, αν η συνάρτηση απεικόνισης επιλεγεί προσεχτικά, μέσα από μια οικογένεια συναρτήσεων που παρουσιάζουν ορισμένες επιθυμητές ιδιότητες, τότε το εσωτερικό γινόμενο μεταξύ των εικόνων ($\varphi(x_1)$, $\varphi(x_2)$, μετά την απεικόνιση) των δύο σημείων x_1 , x_2 μπορεί να γραφτεί ως

$$\langle \varphi(x_1), \varphi(x_2) \rangle = k(x_1, x_2)$$

Όπου $\langle \cdot, \cdot \rangle$ δηλώνει το εσωτερικό γινόμενο στον H και $k(\cdot, \cdot)$ είναι συνάρτηση γνωστή ως συνάρτηση πυρήνα (kernel function). Δηλαδή, τα εσωτερικά γινόμενα στον υψηλότερης διάστασης χώρο, μπορούν να υπολογιστούν αξιοποιώντας τη σχετική συνάρτηση πυρήνα που λειτουργεί στον αρχικό, χαμηλής διάστασης χώρο. Ο χώρος H , που σχετίζεται με την $k(\cdot, \cdot)$, είναι γνωστός ως Χώρος Hilbert με Αναπαράγωγο Πυρήνα.

Ένα αξιοσημείωτο χαρακτηριστικό SVM βελτιστοποίησης είναι ότι όλες οι πράξεις μπορούν να εκφραστούν με την βοήθεια εσωτερικών γινομένων. Επομένως, για να λύσουμε ένα γραμμικό πρόβλημα σε χώρο υψηλότερης διάστασης (μετά την απεικόνιση), αρκεί να αντικαταστήσουμε τα εσωτερικά γινόμενα με τις αντίστοιχες πράξεις πυρήνα. Τυπικά παραδείγματα συναρτήσεων πυρήνα είναι (α) η συνάρτηση ακτινωτής βάσης που ορίζεται ως:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{\sigma^2}\right)$$

Όπου σ είναι παράμετρος που ρυθμίζεται από τον χρήστη και καθορίζει το ρυθμό πτώσης της $k(x,y)$ προς το μηδέν καθώς το y απομακρύνεται από το x και (β) η πολυωνυμική συνάρτηση, που ορίζεται ως:

$$k(x, y) = (x^T y + \beta)^n$$

Δοθέντος ενός αγνώστου x , το απεικονίζουμε πρώτα στο $\phi(x)$ και στη συνέχεια ελέγχουμε αν το αποτέλεσμα της παρακάτω έκφρασης είναι μικρότερο ή μεγαλύτερο του μηδενός.

Η δύναμη και η αδυναμία της SVM

Οι βασικές δυνατότητες της SVM είναι ότι η εκπαίδευση είναι σχετικά εύκολη. Κλιμακώνεται σε σχετικά καλές υψηλές διαστάσεις των δεδομένων και η εξισορρόπηση μεταξύ της ταξινόμησης της πολυπλοκότητας και του λάθους μπορεί να ελεγχθεί ρητά. Περιλαμβάνει την ανάγκη για μια καλή λειτουργία του πυρήνα, βασίζεται στην στατιστική θεωρία μάθησης, μπορεί να χρησιμοποιηθεί για την πρόβλεψη μελλοντικών δεδομένων, εκπαιδεύεται από την επίλυση ενός περιορισμένου τετραγωνικού προβλήματος βελτιστοποίησης, υλοποιεί τη χαρτογράφηση των συντελεστών παραγωγής σε ένα υψηλό τρισδιάστατο χώρο χρησιμοποιώντας ένα σύνολο μη γραμμικών βασικών συναρτήσεων, μπορεί να χρησιμοποιηθεί για μια ποικιλία από αναπαραστάσεις, όπως τα νευρωνικά δίκτυα, splines, πολυωνυμικούς εκτιμητές, κ.λπ., αλλά υπάρχει μια μοναδική βέλτιστη λύση για κάθε επιλογή των SVM παραμέτρων. Αυτό είναι διαφορετικό σε άλλες μηχανές μάθησης, όπως τα τυποποιημένα Νευρωνικά Δίκτυα που χρησιμοποιούν την προς τα πίσω διάδοση.

Με λίγα λόγια, η ανάπτυξη της SVM είναι εντελώς διαφορετική από τους συνήθεις αλγόριθμους που χρησιμοποιούνται για τη μάθηση και παρέχει μια νέα άποψη μάθησης. Τα τέσσερα πιο σημαντικά χαρακτηριστικά της SVM είναι η δυαδικότητα, οι πυρήνες, η κυρτότητα και η σποραδικότητα. Οι SVMs λειτουργούν ως μια από τις καλύτερες προσεγγίσεις για τη μοντελοποίηση δεδομένων. Συνδυάζουν τον γενικευμένο έλεγχο ως μια τεχνική για τον έλεγχο των διαστάσεων. Η χαρτογράφηση του πυρήνα παρέχει μια κοινή βάση για τα περισσότερα από τα συνηθισμένα

απασχολούμενα αρχιτεκτονικά μοντέλα, που επιτρέπει τις συγκρίσεις που πρέπει να εκτελεστούν. Στα προβλήματα ταξινόμησης επιτυγχάνεται γενικευμένος έλεγχος με τη μεγιστοποίηση του περιθωρίου κέρδους, το οποίο αντιστοιχεί στην ελαχιστοποίηση του διανύσματος βάρους σε ένα κανονικό πλαίσιο. Η ελαχιστοποίηση του διανύσματος βάρους μπορεί να χρησιμοποιηθεί ως κριτήριο σε προβλήματα παλινδρόμησης, με μια τροποποιημένη λειτουργία απώλειας. Οι μελλοντικές κατευθύνσεις περιλαμβάνουν, μια τεχνική για την επιλογή της λειτουργίας του πυρήνα και επιπλέον έλεγχος ικανότητας. Τέλος, οι νέες κατευθύνσεις που αναφέρονται στη νέα SVM σχετίζονται με σκευάσματα μάθησης που προτάθηκαν πρόσφατα από τον Vapnik .

$$p(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} Fp(\omega) * e^{-j\omega t} d\omega$$

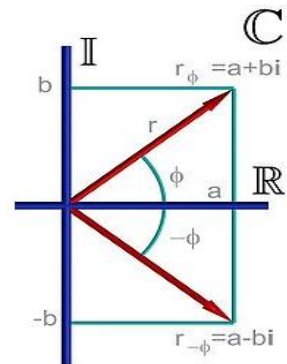
3.2.3. Μετασχηματισμός Fourier

Η μέθοδος που θα χρησιμοποιήσουμε στον 2ο Αλγόριθμο είναι ο εντοπισμός αντικειμένων, μεταφέροντας τα σήματα από το πεδίο του χρόνου στο πεδίο των συχνοτήτων. Είναι μια πολύ αξιόπιστη μέθοδος και επιτυγχάνεται με την χρήση του μετασχηματισμού Fourier. Ακολούθως θα αναλύσουμε συνοπτικά τόσο τον διακριτό όσο και τον δισδιάστατο μετασχηματισμό Fourier:

$$Fp(\omega) = \int_{-\infty}^{+\infty} p(t) * e^{-j\omega t} * dt$$

- $Fp(\omega)$: είναι ο μετασχηματισμός Fourier
- ω : η γωνιακή συχνότητα ($\omega = 2\pi f$) σε rad/sec
- $p(t)$: συνεχές σήμα
- $e(-j\omega t) : \cos(\omega t) - j * \sin(\omega t)$ η εξίσωση που μας δίνει τις συνιστώσες της συχνότητας του σήματος $p(t)$.

Το αποτέλεσμα του μετασχηματισμού Fourier είναι ένας σύνθετος αριθμός. Αυτό σημαίνει ότι δύναται να αναπαρασταθεί με την βοήθεια του πλάτους και της γωνίας διεύθυνσης, όπως φαίνεται παρακάτω:



	Πραγματικό	Φανταστικό
$\int_{-\infty}^{+\infty} p(t) * e^{-j\omega t} = \boxed{R[Fp(\omega)]} + j * \boxed{I[Fp(\omega)]}$		

Κατά συνέπεια το πλάτος του μετασχηματισμού (μέτρο) είναι:

$$\left| \int_{-\infty}^{+\infty} p(t) * e^{-j\omega t} * dt \right| = \sqrt{R[Fp(\omega)]^2 + I[Fp(\omega)]^2}$$

Και αντίστοιχα η γωνία διεύθυνσης (φάση):

$$\int_{-\infty}^{+\infty} p(t) * e^{-j\omega t} = \tan^{-1} \frac{I[Fp(\omega)]}{R[Fp(\omega)]}$$

Το πλάτος του μετασχηματισμού, υποδηλώνει την "ποσότητα" από κάθε συνιστώσα της συχνότητας που περιέχεται στο σήμα, ενώ η φάση δείχνει το πότε αυτή συμβαίνει στον χρόνο. Το μετασχηματισμένο, στο πεδίο της συχνότητας, σήμα μπορεί να επανέλθει στο πεδίο του χρόνου, με τον αντίστροφο μετασχηματισμό Fourier (inverse Fourier), ο οποίος δίνεται στην παρακάτω σχέση:

$$p(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} Fp(\omega) * e^{j\omega t} d\omega$$

Πάρα πολύ σημαντικό ρόλο στην όλη διαδικασία μεταφοράς του σήματος από το πεδίο του χρόνου προς το πεδίο της συχνότητας και αντίστροφα έχει το κριτήριο δειγματοληψίας του Nyquist-Shannon. Ένα βασικό ερώτημα που αφορά στη δειγματοληψία είναι το «κάτω από ποιες συνθήκες μπορούμε να ανακατασκευάσουμε (πλήρως) το δειγματοληπτημένο σήμα;». Μέρος της απάντησης αυτής δίνεται από το θεώρημα Nyquist-Shannon. Το θεώρημα δειγματοληψίας εγγυάται ότι σήματα περιορισμένου φάσματος (π.χ. σήματα τα οποία στο πεδίο συχνοτήτων έχουν τιμές μέχρι μια μέγιστη συχνότητα) μπορούν να ανακατασκευαστούν πλήρως από την δειγματοληπτημένη μορφή τους εάν η συχνότητα δειγματοληψίας είναι μεγαλύτερη ή ίση από το διπλάσιο της μέγιστης συχνότητάς τους. Στην αντίθετη περίπτωση, υπάρχει σημαντική απώλεια πληροφορίας κατά τον μετασχηματισμό Fourier.

Στην περίπτωσή μας οι εικόνες αποτελούν σήματα διακριτά και όχι συνεχή. Επομένως, έχουμε ανάγκη από μια παραλλαγή του μετασχηματισμού Fourier που να δύναται να χειρίζεται διακριτά δεδομένα και όχι συνεχή. Αυτή η μορφή του μετασχηματισμού Fourier ονομάζεται «Διακριτός Μετασχηματισμός Fourier (DFT-Discrete Fourier Transform) και δίνεται από την παρακάτω σχέση:

$$p(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} Fp(\omega) * e^{-j\omega t} d\omega \longrightarrow Fp_u = \frac{1}{\sqrt{N}} * \sum_{x=0}^{N-1} p_x * e^{-j(\frac{2\pi}{N})xu}$$

Όπως παρατηρούμε στο διακριτό ανάλογο του μετασχηματισμού Fourier, το συνεχές σήμα έχει αντικατασταθεί από ένα σύνολο δειγμάτων, οι συνεχείς συχνότητες από δείγμα και το ολοκλήρωμα σε ένα άθροισμα. Αντίστοιχα, για να επανέλθουμε από το πεδίο των συχνοτήτων στο πεδίο του χρόνου, γίνεται η χρήση του αντίστροφου μετασχηματισμού Fourier, ο οποίος φαίνεται στην παρακάτω σχέση:

$$p_x = \sum_{u=0}^{N-1} Fp_u * e^{j(\frac{2\pi}{n})ux}$$

Όπως προαναφέραμε και παραπάνω στην περίπτωση μας το σήμα που έχουμε προς επεξεργασία είναι δισδιάστατο. Κατά συνέπεια θέλουμε τον δισδιάστατο διακριτό μετασχηματισμό Fourier. Άρα έχουμε δύο διαστάσεις συχνοτήτων (u-οριζόντια και v-κάθετα) όπως έχουμε και δύο διαστάσεις στον πίνακα της εικόνας προς επεξεργασία. Ο διδιάστατος διακριτός μετασχηματισμός Fourier υπολογίζει τα δεδομένα των συχνοτήτων $Fp_{u,v}$, από μια εικόνα $N*N$ εικονοστοιχείων, με βάση την παρακάτω σχέση:

$$Fp_{u,v} = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} P_{x,y} * e^{-j(\frac{2\pi}{n})(ux+vy)}$$

Αντίστοιχα, ο αντίστροφος δισδιάστατος διακριτός μετασχηματισμός Fourier, που επαναφέρει το σήμα από το πεδίο συχνοτήτων στο πεδίο του χρόνου είναι ο κάτωθι:

$$P_{x,y} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} Fp_{u,v} * e^{j(\frac{2\pi}{N})(ux+vy)}$$

Ακολούθως θα αναφερθούμε στις ιδιότητες εκείνες που καθιστούν τον παραπάνω μετασχηματισμό σε ένα εξαιρετικό εργαλείο επεξεργασίας εικόνων.

3.2.4. Συνέλιξη (Convolution) – Συσχέτιση (Correlation)

Ο πολλαπλασιασμός μεταξύ δύο σημάτων στο πεδίο του χρόνου αντιστοιχεί στο πεδίο της συχνότητας στην πράξη της συνέλιξης (convolution). Επομένως, μπορούμε να πούμε ότι οι δύο πράξεις αυτές είναι άρρηκτα συνδεδεμένες στα δύο πεδία.

Η πράξη της συνέλιξης ορίζεται με βάση την παρακάτω σχέση:

$$p_1(t) * p_2(t) = \int_{-\infty}^{+\infty} p_1(\tau) * p_2(t-\tau) d\tau$$

Η παραπάνω σχέση αποτελεί την βάση της θεωρίας συστημάτων, όπου η έξοδος του συστήματος είναι η συνέλιξη μιας διέγερσης (p_1) και της απόκρισης του συστήματος (p_2). Ταυτόχρονα η πράξη της συσχέτισης (correlation) ορίζεται αντίστοιχα με την σχέση:

$$p_1(t) \odot p_2(t) = \int_{-\infty}^{+\infty} p_1(\tau) * p_2(t+\tau) d\tau$$

Η πράξη της συσχέτισης μας δίνει ουσιαστικά ένα μέτρο ομοιότητας μεταξύ των δύο σημάτων p_1 και p_2 . Στην περίπτωση που $p_1(\omega) = p_2(\omega)$, τότε στην πραγματικότητα έχουμε μια συσχέτιση ενός σήματος με τον εαυτό του, η οποία ονομάζεται αυτοσυσχέτιση. (autocorrelation)

Για τον εντοπισμό των αντικειμένων που ψάχνουμε (αεροπλάνων) θα χρησιμοποιήσουμε τις ιδιότητες του μετασχηματισμού Fourier που αναφέρθηκαν προηγουμένως. Θα χρησιμοποιήσουμε τον δεισμό της συνέλιξης και του πολλαπλασιασμού. Ταυτόχρονα θα δούμε στην συνέχεια ότι υπάρχει άμεση σχέση της πράξης της συνέλιξης και της συσχέτισης.

Ο πολλαπλασιασμός στο πεδίο του χρόνου, όπως έχουμε αναφέρει, ισοδυναμεί με συνέλιξη στο πεδίο της συχνότητας. Το γεγονός αυτό επιτρέπει να ξεπεράσουμε το κύριο πρόβλημα των αλγορίθμων αναζήτησης που δεν είναι άλλο από το μεγάλο υπολογιστικό φορτίο (απαιτείται τεράστια υπολογιστική ισχύς). Έτσι λοιπόν, αφού μεταφέρουμε τα δύο σήματα από το πεδίο του χρόνου στο πεδίο των συχνοτήτων, το πρόβλημα έχει αναχθεί σε πολύ πιο απλό υπολογιστικά. Δηλαδή, για να βρούμε ένα

σχήμα αρκεί να υπολογίσουμε την συσχέτιση σαν πολλαπλασιασμό στο πεδίο των συχνοτήτων. Όμως, όπως προαναφέραμε, το αντίστοιχο του πολλαπλασιασμού είναι η συνέλιξη (convolution) και όχι η συσχέτιση (correlation).

Αναδιατυπώνουμε την συνέλιξη ως εξής:

$$I \odot T = \sum_{x,y \in W} I_{x',y'} * T_{x'-i,y'-j}$$

Όπου $x' = x+i$ και $y' = y+i$. Η συνέλιξη ορίζεται ως:

$$I * T = \sum_{x,y \in W} I_{x',y'} * T_{i-x',j-y'}$$

Άρα, μπορούμε να γράψουμε ότι

$$I \odot T = I * T' = \sum_{x,y \in W} I_{x',y'} * T_{i-x',j-y'}$$

Όπου $T' = T-x, -y$

Επομένως, η συσχέτιση είναι ισοδύναμη με την συνέλιξη, όταν το σήμα πρότυπο αλλάζει σύμφωνα με την τελευταία σχέση. Η εξίσωση αυτή αντιστρέφει το σύστημα των συντεταγμένων, ενώ ταυτόχρονα αντιμεταθέτει τον οριζόντιο και τον κάθετο άξονα.

Επομένως, συνοπτικά τα βήματα που ακολουθούμε είναι τα κάτωθι:

1. Μετασχηματίζουμε την εικόνα του αντικειμένου – προτύπου (αντικείμενο που θα αναζητήσουμε template), και την εικόνα που θα εξετάσουμε, στο πεδίο της συχνότητας, με την βοήθεια του δισδιάστατου διακριτού μετασχηματισμού Fourier.
2. Πραγματοποιούμε την πράξη της συσχέτισης στο πεδίο της συχνότητας, δηλαδή τον πολλαπλασιασμό των μετασχηματισμένων σημάτων, pixel προς pixel.

3. Μετατρέπουμε το αποτέλεσμα που βγάλαμε στο πεδίο του χρόνου με τον αντίστροφο μετασχηματισμό. Το αποτέλεσμα της παραπάνω πράξης μας δίνει τα σημεία στα οποία βρέθηκε ταύτιση μεταξύ των δύο εικόνων.

Παρακάτω φαίνεται ένα παράδειγμα της παραπάνω διαδικασίας. Έστω ότι διαθέτουμε τον πίνακα

$$B = \begin{array}{|c|c|c|c|c|c|c|} \hline 38 & 5 & 35 & 4 & 7 & 34 & 56 \\ \hline 71 & 25 & 3 & 32 & 86 & 36 & 1 \\ \hline 54 & 68 & 1 & 7 & 8 & 9 & 21 \\ \hline 22 & 33 & 14 & 4 & 24 & 56 & 78 \\ \hline 54 & 5 & 6 & 74 & 12 & 47 & 35 \\ \hline 15 & 1 & 2 & 35 & 45 & 54 & 51 \\ \hline \end{array}$$

Και τον πίνακα Γ:

$$\Gamma = \begin{array}{|c|c|c|} \hline 1 & 5 & 2 \\ \hline 5 & 4 & 9 \\ \hline 6 & 2 & 3 \\ \hline \end{array}$$

Προφανώς και ο πίνακας του αποτελέσματος θα είναι ο ίδιος σε διαστάσεις με τον πίνακα Β. Ακολουθώντας θα εφαρμόσουμε τυχαία το ρίκελ (2,4) του πίνακα εξόδου. Όλα τα ρίκελς προκύπτουν με την ίδια φιλοσοφία. Τοποθετούμε λοιπόν τον πίνακα Γ έτσι ώστε το κεντρικό ρίκελ να βρίσκεται στην θέση (2,4) και ακολούθως πολλαπλασιάζουμε κάθε στοιχείο του πίνακα Α με το αντίστοιχο του πίνακα Γ. Το τελικό άθροισμα όλων των πολλαπλασιασμών αντικαθιστά

Επομένως, το στοιχείο (2,4) θα έχει τιμή :

$$35 \cdot 1 + 4 \cdot 5 + 7 \cdot 2 + 5 \cdot 3 + 32 \cdot 4 + 86 \cdot 9 + 1 \cdot 6 + 7 \cdot 2 + 8 \cdot 3 = 35 \cdot 20 + 14 \cdot 15 + 128 \cdot 7 + 4 \cdot 6 + 14 \cdot 24 = 1030$$

Το παραπάνω αποτέλεσμα μας δείχνει τον βαθμό ομοιότητας που έχει το πρότυπο και η εικόνα σε κάθε σημείο. Αυτό σημαίνει ότι όσο πιο όμοιοι είναι οι δύο πίνακες τόσο μεγαλύτερο το αποτέλεσμα και αντίστροφα. Επομένως, δύναται να εντοπιστούν διάφορα σημεία σε μια εικόνα που να μοιάζουν με το πρότυπο, άλλα περισσότερο

και άλλα λιγότερο. Με την δυνατότητα της κατωφλίωσης μπορούμε να επιλέγουμε μόνο όσα είναι πάνω από μια ορισμένη τιμή – κατώφλι. (threshold).

Η παραπάνω μεθοδολογία παρουσιάζει και κάποια σημαντικά προβλήματα. Ένα σοβαρό είναι η περίπτωση διαφοροποίησης της κλίμακας του αντικειμένου, ενώ η περίπτωση της περιστροφής του αντικειμένου στην εικόνα ελέγχου αποτελεί την δεύτερη περίπτωση σημαντικού προβλήματος. Και τα δύο παραπάνω προβλήματα θα αναλυθούν παρακάτω για το πώς καταφέρνουμε να τα αντιμετωπίσουμε.

3.2.5. Φίλτρα Gabor

Από τις μεγαλύτερες δυσκολίες στην επεξεργασία της εικόνας είναι η υπολογιστική προσομοίωση του ανθρώπινου συστήματος όρασης (HVS) σε νευρωνικό επίπεδο. Έχουν προταθεί διάφορες μέθοδοι για μια ικανοποιητική προσέγγιση αλλά τα φίλτρα Gabor παρουσιάζουν την βέλτιστη από αυτές.

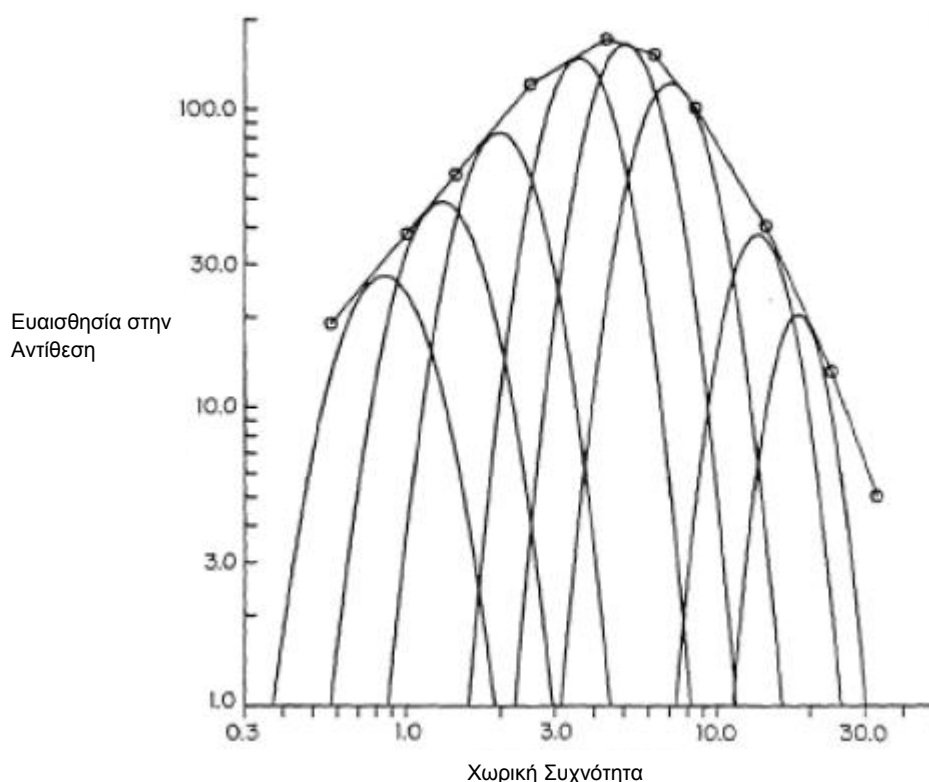
Τα φίλτρα Gabor ονομάζονται έτσι προς τιμή του Ούγγρου φυσικού Denes Gabor (Nobel 1971 – Theory of communications). Ο Denes Gabor θεωρώντας σήματα αποτελούμενα από κυματοπακέτα και χρησιμοποιώντας τις γνώσεις του στην κβαντική φυσική, απέδειξε ότι τα φίλτρα Gabor αποτελούν την βέλτιστη λύση αναπαράστασης σε χωρικό (spatial domain) και συχνοτικό επίπεδο (frequency domain). Από μόνη της μια συχνοτική (Fourier) ή μια χωρική (εντάσεις) μέθοδος δεν μπορεί να περιγράψει στην ολότητά της την οπτική πληροφορία, για αυτό και απαιτείται μια μέθοδος που να συνδυάζει και τα δυο παραπάνω.

Έτσι, λοιπόν, δημιουργείται μια τράπεζα φίλτρων Gabor (Filter Bank) αποτελούμενη από φίλτρα Gabor σε διάφορες διευθύνσεις και σε διάφορες συχνότητες (επομένως και κλίμακες). Τα φίλτρα Gabor βρίσκουν εφαρμογή κυρίως στις παρακάτω εφαρμογές:

1. Επεξεργασία εικόνας (ταξινόμηση – τμηματοποίηση)
2. Όραση Υπολογιστών - Computer Vision
3. Ιατρική (Κατάτμηση αξονικές – μαγνητικές εικόνες)

3.2.6. Ανθρώπινο Σύστημα Όρασης

Το ανθρώπινο σύστημα όρασης (HVS-ΑΣΟ) αποτελείται από πολλαπλά ζωνοδιαβατά χωρικά φίλτρα, συντονισμένα σε διάφορες συχνότητες (Multiple Spatial Frequency Channels). Στο σχήμα που φαίνεται παρακάτω βλέπουμε ότι το ΑΣΟ αποτελείται από έναν αριθμό ανεξάρτητων ανιχνευτών, όπου πριν από αυτούς υπάρχει ένα φίλτρο συντονισμένο σε διαφορετική συχνότητα. Το εύρος ζώνης κάθε απλού κυττάρου είναι περίπου μια οκτάβα (Η ηχητική απόσταση ανάμεσα στον πρώτο και τον τελευταίο ήχο/νότα μιας οκτάφθογγης κλίμακας, ονομάζεται οκτάβα).



Εικόνα 13 – Αναπαράσταση του Ανθρώπινου Συστήματος Όρασης σε συνάρτηση με την χωρική συχνότητα και την ευαισθησία στην αντίθεση

Πειράματα που έχουν πραγματοποιηθεί σε νευρωνικά κύτταρα έχουν δείξει ότι αντιδρούν διαφορετικά στον χώρο και στην συχνότητα. Πιο συγκεκριμένα, διαπιστώθηκε ότι έχουν πολύ μεγάλη ευαισθησία σε συγκεκριμένες διευθύνσεις και συχνότητες που σημαίνει ότι οι άνθρωποι έχουν ισχυρή χωρική και κατευθυντική συχνότητα ευαισθησίας. Αυτό ισοδυναμεί με την απαίτηση να αναπαρασταθεί μαθηματικά το ανθρώπινο σύστημα όρασης, χρησιμοποιώντας συναρτήσεις που να υποστηρίζουν τόσο την συχνοτική όσο και την χωρική συμπεριφορά. Συνδυάζοντας τα παραπάνω, η υπάρχουσα θεωρία των φίλτρων Gabor περιλαμβάνει μια οικογένεια συναρτήσεων τις οποίες τις θεωρεί ως κυματοπακέτα (wavelets)

3.2.7. Συναρτήσεις Gabor

Ο Gabor δημιούργησε μια οικογένεια συναρτήσεων, αποσυνθέτοντας τα σήματα σε «χρονο-συχνοτικά» πακέτα. Μια μονοδιάστατη συνάρτηση Gabor είναι ένα μιγαδικό συνημιτονικό σήμα (sine & cosine), διαμορφωμένο από μια Gaussian περιβάλλουσα.

$$g(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(i2\pi \frac{x}{\lambda}\right)$$

σ : Η τυπική απόκλιση της Gaussian (ορίζει το εύρος).

λ : Το μήκος κύματος του μιγαδικού συνημιτονοειδούς.

$2\pi/\lambda$: Καθορίζει την χωρική συχνότητα.

Το ισοδύναμο της παραπάνω σχέσης σε δύο διαστάσεις είναι:

$$g(x,y) = K \exp(-\pi(a^2(x-x_0)^2 + b^2(y-y_0)^2))$$

$$\exp(j(2\pi(u_0x + v_0y) + P))$$

K : Παράγοντας κλιμάκωσης της γκαουσιανής περιβάλλουσας.

(a,b) : Κλίμακα των δύο αξόνων της γκαουσιανής περιβάλλουσας.

θ : Γωνία περιστροφής της γκαουσιανής περιβάλλουσας.

(x_0, y_0) : Κέντρο της γκαουσιανής περιβάλλουσας.

(u_0, v_0) : Χωρικές συχνότητες του ημιτονικού φέροντος σε καρτεσιανές συντεταγμένες.

P : Φάση του ημιτονοειδούς φέροντος.

Το πραγματικό και το φανταστικό μέρος της δισδιάστατης συνάρτησης Gabor γράφεται όπως παρακάτω:

Πραγματικό

$$\text{Re}\{g(x,y)\} = K \exp(-\pi[(x-x_0)_r^2 a^2 + (y-y_0)_r^2 b^2]) \cos(2\pi(u_0(x-x_0) + v_0(y-y_0)) + P)$$

Φανταστικό

$$\text{Im}\{g(x,y)\} = K \exp(-\pi[(x-x_0)_r^2 a^2 + (y-y_0)_r^2 b^2]) \sin(2\pi(u_0(x-x_0) + v_0(y-y_0)) + P)$$

Μετασχηματίζοντας τις παραπάνω συναρτήσεις κατά Fourier, έχουμε την σχέση:

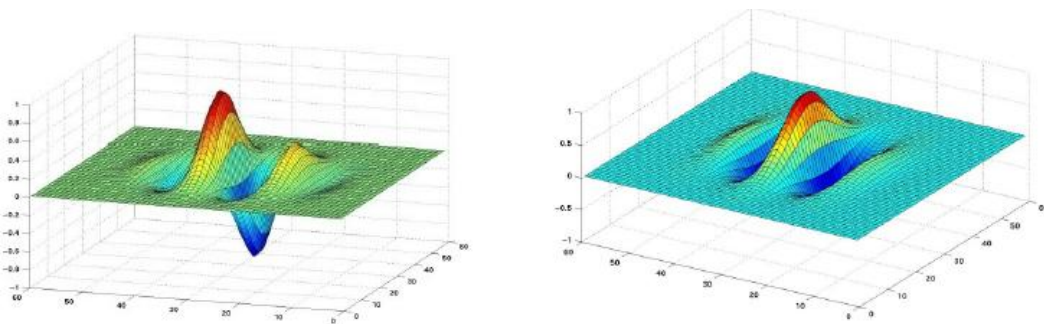
$$G(u,v) = \frac{K}{ab} \exp(-2\pi(x_o(u-u_o) + y_o(v-v_o)) + P) \exp(-\pi[\frac{(u-u_o)_r^2}{a^2} + \frac{(v-v_o)_r^2}{b^2}])$$

με αντίστοιχο μέτρο και φάση:

$$\text{Magnitude}(\hat{g}(u,v)) = \frac{K}{ab} \exp(-\pi[\frac{(u-u_o)_r^2}{a^2} + \frac{(v-v_o)_r^2}{b^2}])$$

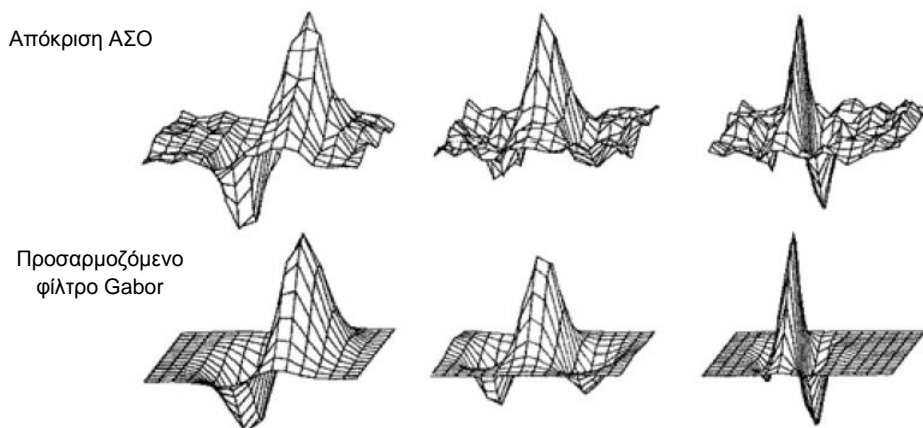
$$\text{Phase}(g(u,v)) = -2\pi[x_o(u-u_o) + y_o(v-v_o)] + P$$

Τα δύο τυπικά δισδιάστατα φίλτρα Gabor είναι τα κάτωθι:



Εικόνα 14 – Αναπαράσταση δισδιάστατων φίλτρων Gabor (Άρτιο – Αριστερά & Περιπτώ – Δεξιά)

Στις εικόνες που ακολουθούνε βλέπουμε την απόκριση του ανθρώπινου ματιού και το αντίστοιχο προσαρμοζόμενο φίλτρο Gabor τέτοιο ώστε να έχει την βέλτιστη δυνατή προσέγγιση με το ΑΣΟ.



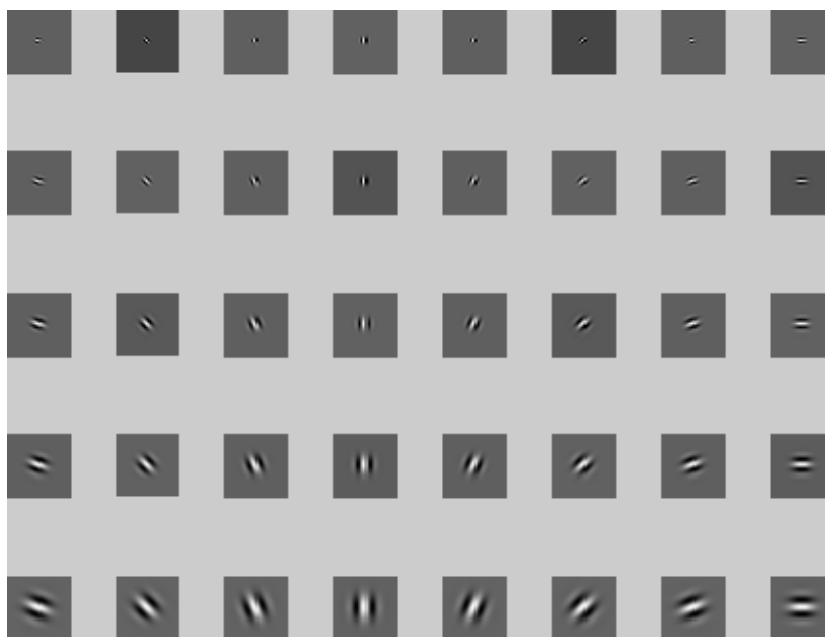
Εικόνα 15 – Αναπαράσταση αποκρίσεων του ΑΣΟ στο αντίστοιχο προσαρμοζόμενο φίλτρο Gabor

3.2.8. Δημιουργία Φίλτρων Gabor

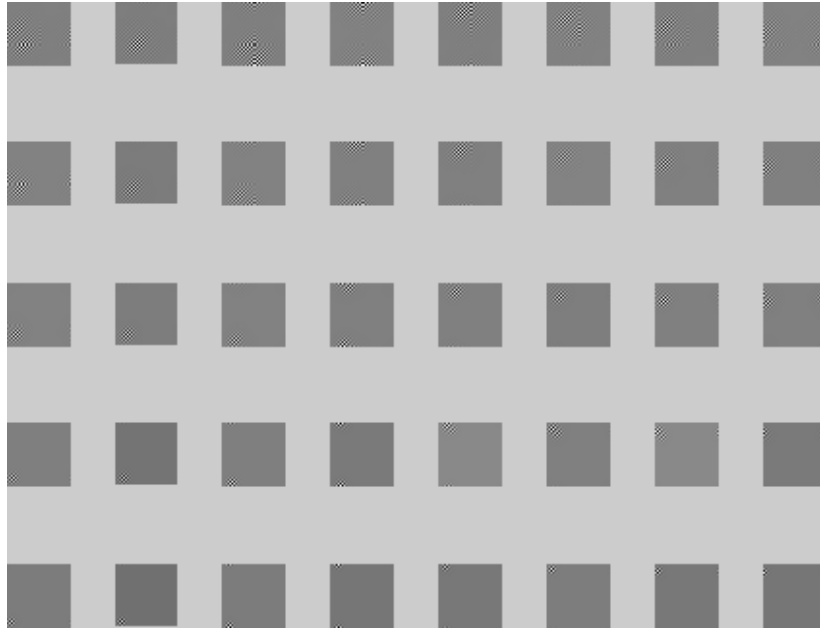
Κατά την δημιουργία των φίλτρων Gabor αυτό που πρέπει να προσέξουμε στα χαρακτηριστικά τους είναι :

1. Η λειτουργικότητα
2. Ο αριθμός
3. Ο προσανατολισμός
4. Η μεταξύ τους απόσταση

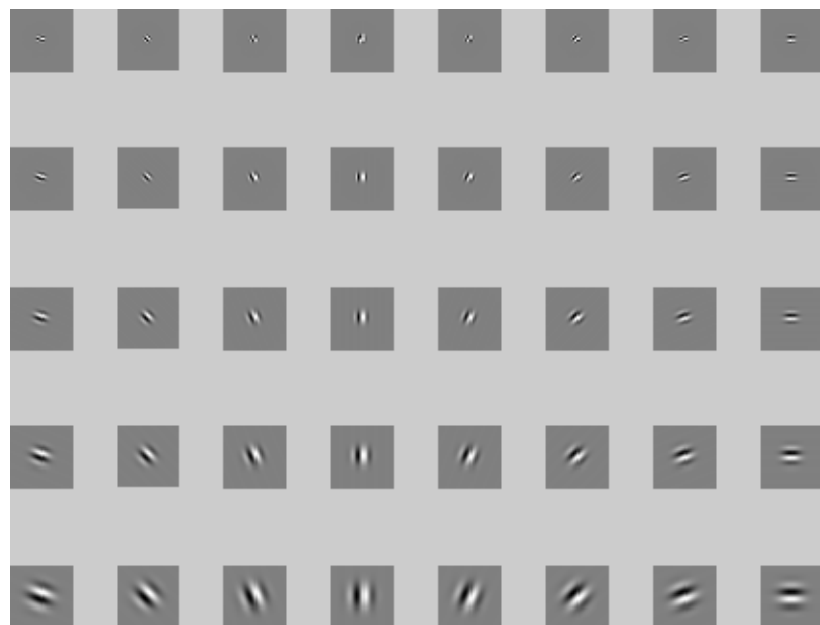
Το βασικό βήμα στο οποίο θα πρέπει να προβούμε κατά την σχεδίαση είναι οι κλίμακες και οι προσανατολισμοί που θα επιλέξουμε για την Filter Bank. Οι συχνότητες πρέπει να επιλεγούν με ιδιαίτερη προσοχή, σκεπτόμενοι πάντα τις ιδιαιτερότητες των εικόνων μας. Αντίστοιχα, το πλήθος των προσανατολισμών θα πρέπει να είναι τέτοιο που να καλύπτει οριακά την εξαγωγή των χαρακτηριστικών. Ένας τυπικός αριθμός διευθύνσεων (orientations) είναι 6, αλλά εμείς χρησιμοποιήσαμε 8 και συγκεκριμένα 0, 22,5, 45, 67,5, 90, 112,5, 135, 157,5 ενώ το σύνηθες πλήθος συχνοτήτων είναι περίπου πέντε. Μια τέτοια τράπεζα φίλτρων Gabor φαίνεται στα σχήματα που ακολουθούν, όπου πρώτα παρουσιάζονται τα πραγματικά μέρη των συναρτήσεων $G(u,v)$ (2D φίλτρων) και στην συνέχεια οι αντίστοιχες αναπαράστασεις των FFT (Fast Fourier Transform) για όλες τις συχνότητες και όλους τους προσανατολισμούς.



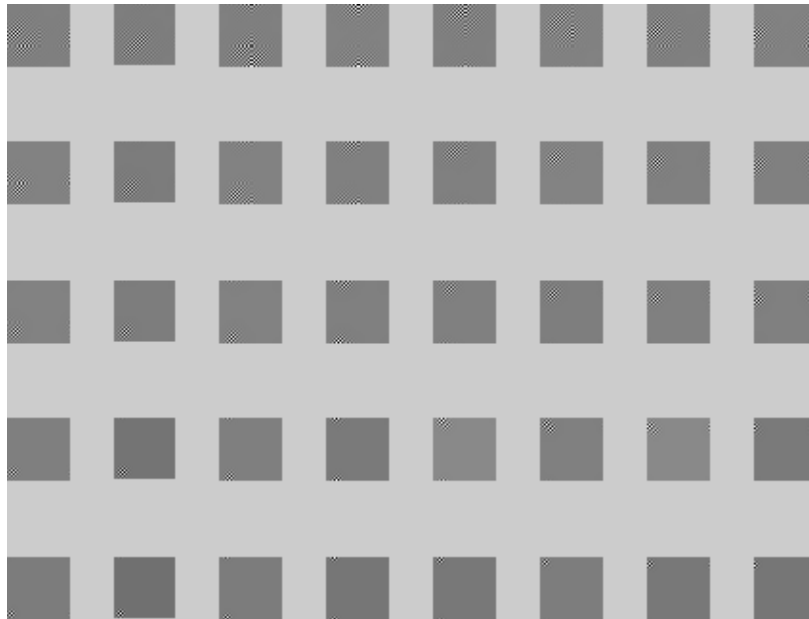
Εικόνα 16 – Αναπαράσταση φίλτρων Gabor (Πραγματικά μέρη)



Εικόνα 17– Αναπαράσταση φίλτρων Gabor (πραγματικά μέρη) μετασχηματισμένα στο πεδίο των συχνοτήτων



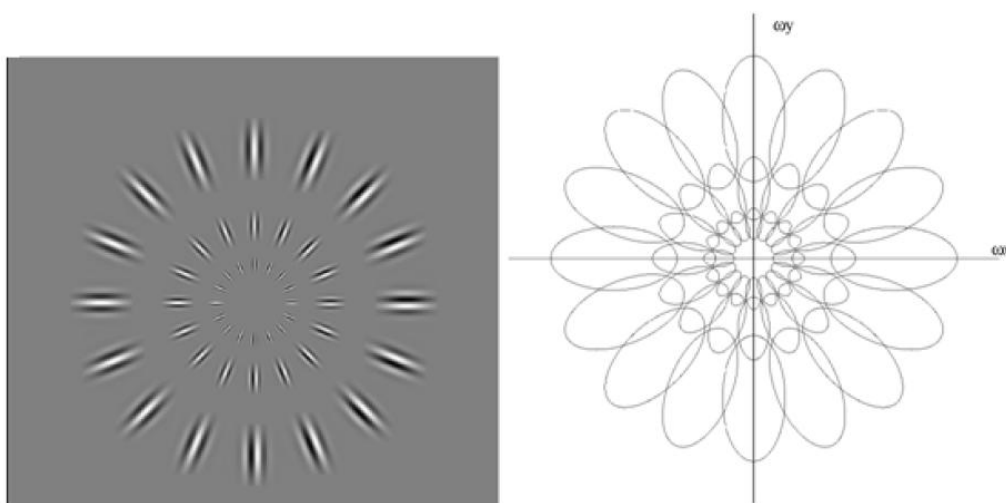
Εικόνα 18 – Αναπαράσταση φίλτρων Gabor (Φανταστικά μέρη)



Εικόνα 19 – Αναπαράσταση φίλτρων Gabor (φανταστικά μέρη) μετασχηματισμένα στο πεδίο των συχνοτήτων

Ο σχεδιασμός της τράπεζας φίλτρων γίνεται έτσι ώστε να καλύπτεται εάν είναι δυνατόν, ολόκληρος ο χώρος των συχνοτήτων. Αυτό μπορεί κανείς να το δει εύκολα, στην παραπάνω εικόνα απεικόνισης του πραγματικού μέρους. Εάν για παράδειγμα U_l και U_h είναι η χαμηλότερη και η υψηλότερη αντίστοιχα κεντρική συχνότητα που ενδιαφέρει, K ο αριθμός των διευθύνσεων και S ο αριθμός των διαφορετικών κλιμάκων, τότε είθισται η σχεδίαση των φίλτρων Gabor να γίνεται έτσι ώστε οι αποκρίσεις ημίσεως εύρους των φίλτρων να εφάπτονται.

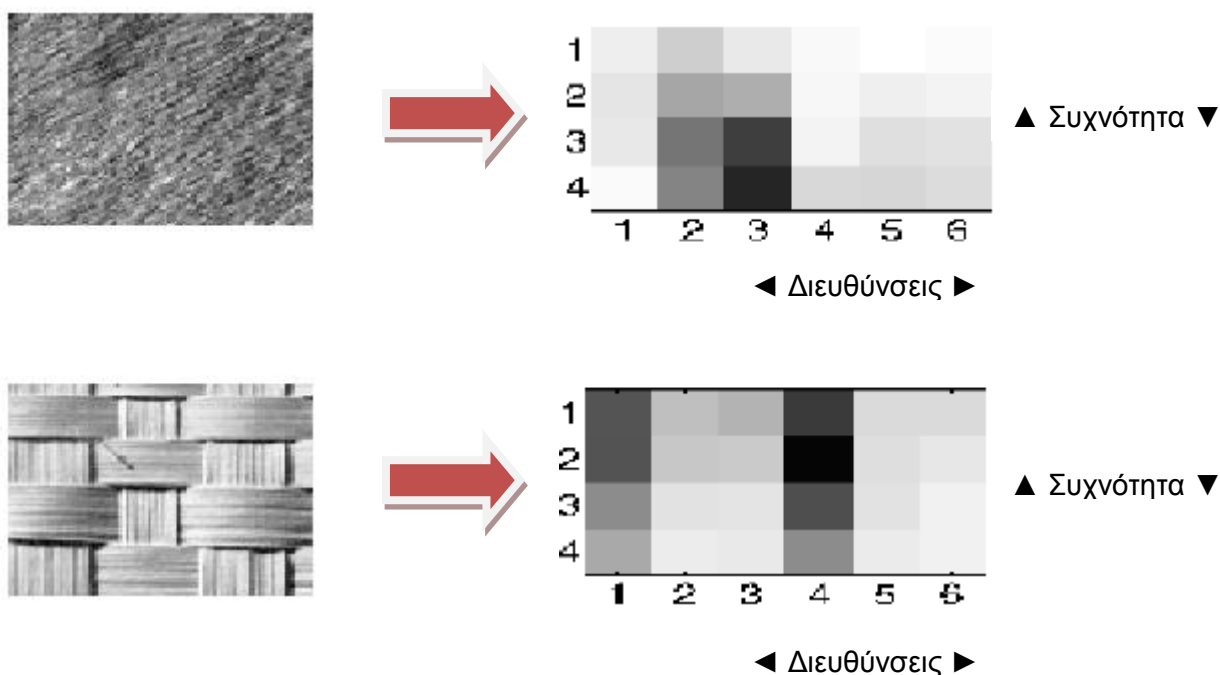
Παρόλα ταύτα, αυτό δεν είναι απαγορευτικό, αφού, όπως φαίνεται και παρακάτω, οι αποκρίσεις δύναται να επικαλύπτονται ανάλογα πάντα και με την εφαρμογή τους. Το μεγαλύτερο μειονέκτημα των φίλτρων Gabor είναι ότι, ενώ αποτελούν μια ολοκληρωμένη βάση, δεν είναι ορθογώνια.



Εικόνα 20 – Αναπαράσταση φίλτρων Gabor (συχνοτική κάλυψη AP & Αποκρίσεις ΔΕΞ)

Ένας άλλος εξίσου σημαντικός παράγοντας στην επεξεργασία με φίλτρα Gabor είναι η επιλογή των χαρακτηριστικών που θα εξαχθούν. Κατά την συνέλιξη μιας εικόνας με ένα φίλτρο Gabor, το αποτέλεσμα είναι μια άλλη εικόνα ίδιων διαστάσεων της οποίας τα εικονοστοιχεία περιέχουν την πληροφορία για την υφή της αρχικής εικόνας. Προφανώς και δεν θα μπορούσε να εξαχθεί κάποιο χαρακτηριστικό διάνυσμα με όλες τις τιμές των εικονοστοιχείων αυτών, αφού το υπολογιστικό κόστος θα ήταν τεράστιο. Για αυτό το λόγο, επιλέγονται χαρακτηριστικά τα οποία μπορούν να αντιπροσωπεύουν την εικόνα σε ικανοποιητικό βαθμό. Ένα τέτοιο χαρακτηριστικό διάνυσμα για μια αρχική εικόνα $m \times n$ έχει την μορφή: $[\mu_{00}, \sigma_{00}, \dots, \mu_{mn}, \sigma_{mn}]$. Όπου μ και σ είναι η μέση τιμή και η τυπική απόκλιση αντίστοιχα του εικονοστοιχείου (i, j) . Στην πράξη η εφαρμογή των φίλτρων σε ολόκληρη την εικόνα δεν μας προσφέρει κάτι για την εξαγωγή της υφής. Για αυτό τον λόγο, η εφαρμογή των φίλτρων πραγματοποιείται σε μικρά τετράγωνα (patch) γύρω από κάθε εικονοστοιχείο, τα οποία έχουν συνήθως διαστάσεις 32×32 ή 24×24 .

Η σύγκριση δύο υφών δύναται να γίνει με διάφορους τρόπους. Ένας από αυτούς είναι με τους χάρτες ενέργειας, που προκύπτουν από την ενέργεια για κάθε συχνότητα(κλίμακα) και διεύθυνση. Η ενέργεια από κάθε τετράγωνο (patch) προέρχεται από τον μέσο όρο των τιμών των εντάσεων που προκύπτουν από την συνέλιξη του φίλτρου Gabor με το τετράγωνο (patch) της αρχικής εικόνας. Παρακάτω βλέπουμε την διαφορά αυτή μεταξύ δύο εικόνων που ακόμα και με γυμνό μάτι είναι ορατό ότι έχουν διαφορετικές διευθύνσεις-συχνότητες με μεγαλύτερη ενέργεια.



Εικόνα 21 – Συγκριτικά αποτελέσματα φίλτρων Gabor για διάφορες περιπτώσεις συχνότητων και διευθύνσεων

3.2.9. Αναλυτική Περιγραφή Αναπτυγμένου Αλγορίθμου και Συναρτήσεων

Στο κεφάλαιο αυτό θα παρουσιάσουμε αναλυτικά τον κώδικα που χρησιμοποιήσαμε. Θα πρέπει να τονίσουμε ότι ο κώδικας που χρησιμοποιήσαμε ήταν φτιαγμένος για τον εντοπισμό προσώπων και με τις απαραίτητες μετατροπές προσαρμόστηκε στα δικά μας αντικείμενα στόχους που δεν ήταν άλλα από σταθμευμένα αεροσκάφη.

Το Matlab που χρησιμοποιήθηκε είναι το «Matlab 2013a» και οι αντίστοιχες βιβλιοθήκες που αξιοποιήθηκαν είναι οι:

1. Computer Vision System Toolbox
2. Image Processing Toolbox
3. Neural Network Toolbox
4. Optimization Toolbox

Ο κώδικας εντοπισμού αεροπλάνων αποτελείται από 9 script αρχεία τα οποία θα παρουσιάσουμε αναλυτικά γραμμή – γραμμή. Θα αναφέρουμε τις εντολές της Matlab που χρησιμοποιούνται και θα περιγράψουμε την λειτουργία τους. Στις γραμμές που ακολουθούνε, όσα τμήματα του κειμένου εσωκλείονται σε μπλε πλαίσιο, σημαίνει ότι είναι αυτούσια κομμάτια κώδικα από το αντίστοιχο m-file της Matlab.

Τα script τα οποία αποτελούν το πρόγραμμα είναι τα κάτωθι:

1. Main.m
2. Drawrec.m
3. Gabor.m
4. Create_gabor.m
5. Im2vec.m
6. Imscan.m
7. Loadimages.m
8. Mminmax.m
9. Trainet.m

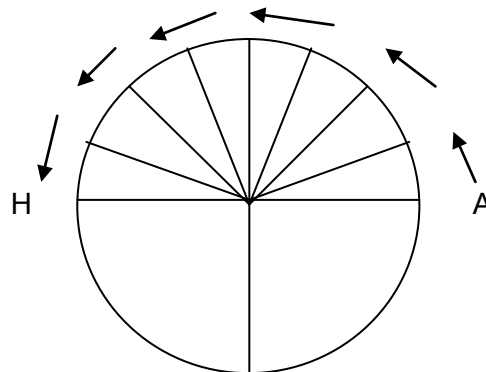
Συνάρτηση *Gabor.m*

Μια καλή επιλογή να ξεκινήσουμε είναι με το αρχείο του υπολογισμού του πίνακα Gabor. Συγκεκριμένα, το αρχείο *Gabor.m* έχει σχεδιαστεί για τον υπολογισμό του πίνακα Gabor αν δώσουμε τα απαραίτητα ορίσματα τα οποία είναι:

1. *Plaisio_gabor*, δηλαδή η διάσταση του πίνακα. Όπως προαναφέραμε και παραπάνω, ο πίνακας Gabor συνήθως επιλέγεται 32*32 ή 24*24. Εμείς έχουμε επιλέξει την διάσταση 32*32.
2. *Klimaka_sixnotita* : Αφορά στην συχνότητα που θα υπολογίζεται κάθε φορά το φίλτρο Gabor. Η κλίμακα Gabor παίρνει στην δικιά μας περίπτωση 5 τιμές και συγκεκριμένα 0,1,2,3,4.
3. *Prosanatolismos*: Αφορά την διεύθυνση υπολογισμού του φίλτρου Gabor. Ο σχεδιασμός της τράπεζας φίλτρων γίνεται έτσι ώστε να καλύπτεται, ολόκληρος ο χώρος των συχνοτήτων. Έτσι, επιλέξαμε 8 διαφορετικές διευθύνσεις οι οποίες είναι σε μοίρες:



- A. 0°
- B. 22,5°
- C. 45°
- D. 67,5°
- E. 90°
- F. 112,5°
- G. 135°
- H. 157,5°



4. *Kmax*: Γωνία
5. *f*: Τετραγωνική ρίζα
6. *sig*: Γωνία

```
K = Kmax/f^Klimaka_sixnotita * exp(i*Prosanatolismos*pi/8);
Kreal = real(K);
Kimag = imag(K);
NK = Kreal^2+Kimag^2;
Psi = zeros(m,n);
for x = 1:m
    for y = 1:n
        Z = [x-m/2;y-n/2];
        Psi(x,y) = (sig^(-2))*exp((-0.5)*NK*(Z(1)^2+Z(2)^2)/(sig^2))*...
            (exp(i*[Kreal Kimag]*Z)-exp(-(sig^2)/2));
```

Συνάρτηση *Create_Gabor.m*

Θα πρέπει να τονίσουμε ότι τα υπολογισμένα φίλτρα Gabor εισάγονται σε υπερπίνακες cell της Matlab. Μετά την εκτέλεση του αρχείου *create_gabor.m* αποθηκεύονται στο αρχείο *Gabor.mat* (*G{}*) όλα τα αποτελέσματα των φίλτρων Gabor και για τις $5 \times 8 = 40$ διαφορετικές περιπτώσεις όπως φαίνεται παρακάτω:

	1	2	3	4	5	6	7	8
1	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...
2	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...
3	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...
4	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...
5	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...	<32x32 co...

Εικόνα 22 - Υπερ-πίνακας (cell matrix) Gabor

Το κάθε κελί εμπεριέχει το φίλτρο Gabor για τον αντίστοιχο συνδυασμό διεύθυνσης & κλίμακας που υπολογίστηκε. Προφανώς ο κάθε πίνακας που κρύβεται πίσω από κάθε κελί του *G*, έχει διαστάσεις 32×32 και τα στοιχεία του είναι όλα μιγαδικοί αριθμοί. Παρακάτω παρουσιάζεται ενδεικτικά ο πίνακας που υπάρχει πίσω από το κελί (1,1):

32

	1	2	3	4
1	1.8012e-99 + 7.0514e-100i	-3.5349e-95 + 3.8611e-93i	-2.6495e-87 + 1.0095e-87i	-5.1231e-82 - 5.6055e-82i
2	-3.1933e-93 - 2.2108e-93i	1.7729e-87 - 7.4342e-87i	5.4987e-81 - 7.0982e-82i	7.0237e-76 + 1.3179e-75i
3	1.8410e-87 + 2.1136e-87i	-2.6200e-81 + 4.9525e-81i	-4.0771e-75 - 4.5395e-76i	-2.8555e-70 - 1.0571e-69i
4	-3.5633e-82 - 6.6933e-82i	9.7657e-76 - 1.1271e-75i	1.0186e-69 + 3.7562e-70i	3.0933e-66 + 2.9324e-64i
5	1.7466e-77 + 7.2398e-77i	-1.2211e-70 + 8.5014e-71i	-9.0432e-65 - 6.0025e-65i	6.1495e-60 - 2.8245e-59i
6	-3.1878e-74 - 2.7092e-72i	4.9635e-66 - 1.9582e-66i	2.6188e-60 + 2.8890e-60i	-4.7484e-55 + 9.4187e-55i
7	-8.6727e-69 + 3.5203e-68i	-7.1638e-62 + 9.5875e-63i	-2.5590e-56 - 4.5874e-56i	8.9446e-51 - 1.0746e-50i
8	7.9871e-65 - 1.5830e-64i	3.4823e-58 + 3.7017e-59i	6.5139e-53 + 2.4840e-52i	-5.6199e-47 + 4.0782e-47i
9	-2.1266e-61 + 2.4331e-61i	-6.0488e-55 - 2.1524e-55i	-1.4637e-50 - 4.6502e-49i	1.1441e-43 - 4.7782e-44i
10	1.7272e-58 - 1.2409e-58i	3.5095e-52 + 2.3367e-52i	-6.8192e-47 + 3.0228e-46i	-8.2641e-41 + 1.2710e-41i
11	-4.8995e-56 + 1.9401e-56i	-7.1360e-50 - 7.6068e-50i	3.2658e-44 - 6.8038e-44i	2.0095e-38 + 1.7336e-39i
12	4.6318e-54 - 6.6003e-55i	4.4865e-48 + 8.1582e-48i	-4.4034e-42 + 5.2423e-42i	-1.7464e-36 - 5.8319e-37i
13	-1.5641e-52 - 1.5362e-53i	-8.4738e-47 - 2.9819e-46i	1.7959e-40 - 1.3451e-40i	5.0753e-35 + 3.2358e-35i
14	1.7799e-51 + 6.2992e-52i	8.9026e-47 + 3.7674e-45i	-2.5527e-39 + 1.0691e-39i	-5.1779e-34 + 5.3070e-34i
15	-7.2022e-51 - 4.6468e-51i	3.3722e-45 - 1.6527e-44i	1.2075e-38 - 1.9660e-39i	1.6443e-33 + 2.8548e-33i
16	9.5226e-51 + 1.0232e-50i	-1.2251e-44 + 2.5109e-44i	-2.0398e-38 - 1.6012e-39i	-1.5962e-33 - 5.2242e-33i
17	-4.2622e-51 - 7.4120e-51i	1.0587e-44 - 1.3065e-44i	1.1615e-38 + 3.8531e-39i	1.4318e-34 + 3.3021e-33i
18	5.0527e-52 + 1.8288e-51i	-3.0390e-45 + 2.2662e-45i	-2.3533e-39 - 1.4538e-39i	1.3303e-34 - 7.2466e-34i
19	-6.9497e-54 - 1.5593e-52i	2.8203e-46 - 1.2217e-46i	1.5620e-40 + 1.6131e-40i	-2.5557e-35 + 5.5102e-35i
20	-9.7829e-55 + 4.6165e-54i	-9.2805e-48 + 1.5510e-48i	-3.5255e-42 - 5.8615e-42i	1.1180e-36 - 1.4369e-36i
21	2.1964e-56 - 4.7340e-56i	1.0277e-49 + 7.5042e-51i	2.1553e-44 + 7.2410e-44i	-1.6135e-38 + 1.2532e-38i
22	-1.3607e-58 + 1.6634e-58i	-4.0681e-52 - 1.3000e-52i	-1.9869e-47 - 3.0889e-46i	7.5019e-41 + 3.4279e-41i
23	2.5342e-61 - 1.9505e-61i	5.3887e-55 + 3.3367e-55i	-8.7575e-50 + 4.5747e-49i	-1.2355e-43 + 2.3141e-44i
24	-1.6425e-64 + 7.1308e-65i	-2.5084e-58 - 2.5049e-58i	1.0334e-52 - 2.3478e-52i	6.8444e-47 + 3.6338e-48i
25	3.5392e-68 - 6.2417e-69i	3.6496e-62 + 6.1484e-62i	-3.2486e-56 + 4.1342e-56i	-1.3554e-50 - 4.0409e-51i
26	-2.7229e-72 - 1.7821e-73i	-1.6401e-66 - 5.1275e-66i	3.0399e-60 - 2.4372e-60i	8.9911e-55 + 5.3238e-55i
27	7.0624e-77 + 2.2400e-77i	8.3368e-72 + 1.4762e-70i	-9.8751e-65 + 4.5190e-65i	-2.0992e-59 - 2.0156e-59i
28	-6.5203e-82 - 3.9121e-82i	2.5092e-76 - 1.4754e-75i	1.0648e-69 - 2.0966e-70i	1.5407e-64 + 2.4824e-64i
29	1.9755e-87 + 1.9872e-87i	-2.2900e-81 + 5.1109e-81i	-4.0981e-75 - 1.8780e-76i	-3.5420e-70 - 1.0367e-69i
30	-2.0407e-93 - 3.2949e-93i	4.6013e-87 - 6.0768e-87i	5.3183e-81 + 1.5714e-81i	1.1390e-76 + 1.4932e-75i
31	5.7870e-100 + 1.8549e-99i	-3.0353e-93 + 2.4213e-93i	-2.4582e-87 - 1.4101e-87i	1.1186e-82 - 7.4657e-82i
32	-2.7917e-107 - 3.6042e-106i	6.4340e-100 - 3.0429e-100i	3.7372e-94 + 3.6133e-94i	-5.4963e-89 + 1.2942e-88i

32

Εικόνα 23 - Διαστάσεις Υπερ-πίνακα (cell matrix) Gabor

Αρχικά ορίζεται η διάσταση του υπερπίνακα (5,8) και ακολούθως οι διαστάσεις των υποπίνακων που θα δεχτεί για κάθε κελί του. Καταχωρούνται όλες οι θέσεις με μηδέν για να δεσμευθεί ο απαραίτητος χώρος μνήμης.

```
G = cell(5,8);
for s = 1:5
    for j = 1:8
        G{s,j}=zeros(32,32);
    end
end
```

Έπειτα για όλους τους συνδυασμούς κλίμακας και διευθύνσεων που είχαμε προ επιλέξει υπολογίζεται το φίλτρο Gabor, καλώντας την συνάρτηση Gabor.m που αναλύσαμε παραπάνω. Όπως βλέπουμε και παρακάτω τα μοναδικά ορίσματα που αλλάζουν είναι η διεύθυνση και η συχνότητα. Ακολούθως εμφανίζονται τα φίλτρα Gabor παρουσιάζοντας μόνο το πραγματικό μέρος.

Στις σελίδες 70-71 μπορούμε να δούμε τόσο το πραγματικό όσο και το φανταστικό μέρος.

```
for s = 1:5
    for j = 1:8
        G{s,9-j} = gabor([32 32], (s-1), j-1, pi, sqrt(2), pi);
    end
end
figure;
for s = 1:5
    for j = 1:8
        subplot(5,8, (s-1)*8+j);
        imshow(real(G{s,j}), []);
    end
end
```

Τέλος, υπολογίζεται για όλους τους υποπίνακες ένας νέος πίνακας που περιέχει το φίλτρο Gabor στο πεδίο των συχνοτήτων μέσω του δισδιάστατου διακριτού μετασχηματισμού Fourier. Οπτικά τα αποτελέσματα της μετατροπής αυτής φαίνονται και πάλι στην σελίδα 71-72. Το τελικό βήμα είναι να αποθηκευτεί ο πίνακας G ο οποίος είναι αρκετά διαφοροποιημένος ως προς τις τιμές σε σχέση με τον προηγούμενο πίνακα G.

```

for s = 1:5
    for j = 1:8
        G{s,j}=fft2(G{s,j});
    end
end
save gabor G

```

	1	2	3	4	5
1	0.0010 - 0.0000i	-0.0024 - 0.0005i	0.0038 + 0.0016i	-0.0047 - 0.0032i	0.0051 + 0.0051i
2	9.5184e-06 + 1.8933e-06i	0.0011 + 0.0004i	-0.0021 - 0.0014i	0.0027 + 0.0027i	-0.0029 - 0.0043i
3	6.2420e-04 + 2.5855e-04i	-0.0016 - 0.0011i	0.0024 + 0.0024i	-0.0027 - 0.0040i	0.0024 + 0.0057i
4	-0.0026 - 0.0017i	0.0035 + 0.0035i	-0.0038 - 0.0056i	0.0033 + 0.0080i	-0.0020 - 0.0101i
5	0.0054 + 0.0054i	-0.0058 - 0.0086i	0.0051 + 0.0122i	-0.0031 - 0.0156i	0.0000 + 0.0183i
6	-0.0082 - 0.0122i	0.0073 + 0.0176i	-0.0046 - 0.0230i	-0.0000 + 0.0276i	0.0061 - 0.0305i
7	0.0096 + 0.0232i	-0.0062 - 0.0312i	0.0000 + 0.0386i	0.0087 - 0.0440i	-0.0192 + 0.0462i
8	-0.0077 - 0.0388i	0.0000 + 0.0496i	0.0116 - 0.0586i	-0.0264 + 0.0638i	0.0426 - 0.0638i
9	0.0000 + 0.0588i	0.0143 - 0.0719i	-0.0337 + 0.0813i	0.0564 - 0.0843i	-0.0795 + 0.0795i
10	0.0162 - 0.0815i	-0.0396 + 0.0956i	0.0688 - 0.1030i	-0.1009 + 0.1009i	0.1314 - 0.0878i
11	-0.0431 + 0.1040i	0.0778 - 0.1164i	-0.1184 + 0.1184i	0.1602 - 0.1070i	-0.1971 + 0.0816i
12	0.0812 - 0.1216i	-0.1285 + 0.1285i	0.1807 - 0.1207i	-0.2310 + 0.0957i	0.2714 - 0.0540i
13	-0.1291 + 0.1291i	0.1886 - 0.1260i	-0.2505 + 0.1038i	0.3059 - 0.0608i	-0.3451 + 0.0000i
14	0.1822 - 0.1217i	-0.2515 + 0.1042i	0.3191 - 0.0635i	-0.3742 + 0.0000i	0.4061 + 0.0808i
15	-0.2338 + 0.0968i	0.3082 - 0.0613i	-0.3756 + 0.0000i	0.4237 + 0.0843i	-0.4416 - 0.1829i
16	0.2756 - 0.0548i	-0.3491 - 0.0000i	0.4092 + 0.0814i	-0.4432 - 0.1836i	0.4413 + 0.2949i
17	-0.3003 - 0.0000i	0.3658 + 0.0728i	-0.4118 - 0.1706i	0.4262 + 0.2848i	-0.4010 - 0.4010i
18	0.3028 + 0.0602i	-0.3542 - 0.1467i	0.3810 + 0.2546i	-0.3726 - 0.3726i	0.3239 + 0.4848i
19	-0.2821 - 0.1168i	0.3153 + 0.2107i	-0.3205 - 0.3205i	0.2896 + 0.4334i	-0.2207 - 0.5328i
20	0.2416 + 0.1614i	-0.2552 - 0.2552i	0.2396 + 0.3586i	-0.1898 - 0.4583i	0.1071 + 0.5328i
21	-0.1880 - 0.1880i	0.1835 + 0.2747i	-0.1511 - 0.3648i	0.0886 + 0.4454i	0.0000 - 0.5025i
22	0.1301 + 0.1947i	-0.1113 - 0.2688i	0.0678 + 0.3411i	0.0000 - 0.4000i	-0.0864 + 0.4342i
23	-0.0759 - 0.1831i	0.0481 + 0.2416i	0.0000 - 0.2946i	-0.0661 + 0.3324i	0.1435 - 0.3465i
24	0.0315 + 0.1582i	-0.0000 - 0.2006i	-0.0468 + 0.2353i	0.1057 - 0.2551i	-0.1698 + 0.2542i
25	-0.0000 - 0.1260i	-0.0306 + 0.1538i	0.0719 - 0.1735i	-0.1201 + 0.1798i	0.1694 - 0.1694i
26	-0.0184 + 0.0926i	0.0450 - 0.1087i	-0.0784 + 0.1173i	0.1150 - 0.1150i	-0.1499 + 0.1002i
27	0.0259 - 0.0625i	-0.0470 + 0.0703i	0.0719 - 0.0719i	-0.0976 + 0.0652i	0.1204 - 0.0499i
28	-0.0256 + 0.0383i	0.0410 - 0.0410i	-0.0582 + 0.0389i	0.0749 - 0.0310i	-0.0885 + 0.0176i
29	0.0209 - 0.0209i	-0.0313 + 0.0209i	0.0423 - 0.0175i	-0.0523 + 0.0104i	0.0597 + 0.0000i
30	-0.0146 + 0.0097i	0.0211 - 0.0087i	-0.0276 + 0.0055i	0.0333 + 0.0000i	-0.0369 - 0.0073i
31	0.0086 - 0.0035i	-0.0124 + 0.0025i	0.0161 + 0.0000i	-0.0191 - 0.0038i	0.0207 + 0.0086i
32	-0.0039 + 0.0008i	0.0061 + 0.0000i	-0.0082 - 0.0016i	0.0098 + 0.0041i	-0.0105 - 0.0070i

Εικόνα 24 - Τελικός πίνακας Gabor μετασχηματισμένος στο πεδίο των συχνοτήτων

Συνάρτηση *Im2Vec.m*

Η συνάρτηση αυτή είναι η πιο σημαντική όλου του προγράμματος. Χρησιμοποιείται περισσότερο από οποιαδήποτε άλλη συνάρτηση και είναι ουσιαστικά σαν ένας σύνδεσμος μεταξύ του 33*32 παραθύρου patch και του νευρωνικού δικτύου. Η δουλειά του είναι να περάσει κάθε υπο-παράθυρο στο νευρωνικό δίκτυο και αντίστροφα. Εάν υπάρχει κάποια εξαγωγή χαρακτηριστικών σημείων, αυτό συμβαίνει σε αυτή την συνάρτηση.

Αφού έχουμε μιλήσει προηγουμένως για το πώς δημιουργούμε ένα νευρωνικό δίκτυο και το πώς το εκπαιδεύουμε, πλέον πρέπει να ασχοληθούμε με το πώς πρέπει να προετοιμάσουμε τα δεδομένα μας για να είναι έτοιμα για τροφοδοσία στο ΤΝΔ.

Την απάντηση στα παραπάνω προβλήματα την δίνει η συνάρτηση *im2vec*. Το όρισμα της συνάρτησης είναι ένα παράθυρο 33*32 pixels (ουσιαστικά είναι η κάθε εικόνα εκπαίδευσης). Η έξοδος της συνάρτησης είναι ένα διάνυσμα ($m \times 1$) στοιχείων το οποίο είναι έτοιμο προς τροφοδοσία του ΤΝΔ. Τα βήματα τα οποία υλοποιούνται στο εσωτερικό της συνάρτησης είναι πολύ σημαντικά γιατί ουσιαστικά αυτά είναι που εξάγουν τα διάφορα χαρακτηριστικά της εικόνας ενώ παράλληλα έχουν πολύ μεγάλη επιρροή στην γενικότερη απόδοση του προγράμματος αφού εμπεριέχει τον μεγαλύτερο αριθμό πράξεων.

Η συνάρτηση εκτελείται σε δύο φάσεις. Η πρώτη φάση είναι όταν θέλεις να εκπαιδεύσεις το ΤΝΔ σου. Πριν την εκπαίδευση πρέπει να παράγουμε κάποια δεδομένα τα οποία να δύνανται να τροφοδοτήσουν το δίκτυό μας. Η δεύτερη φάση είναι όταν η εκπαίδευση έχει τελειώσει και κάποιος θέλει να ελέγξει το σύστημα πάνω σε κάποιες πραγματικές εικόνες, για παράδειγμα, έστω ότι έχουμε μια ευρύτερη εικόνα ενός Α/Δ του οποίου θέλουμε να εντοπίσουμε αν έχει αεροπλάνα ή όχι.

Πιο απλή προσέγγιση είναι να κοπεί κάθε πιθανό παράθυρο 33*32 στην εικόνα δοκιμής. Εάν γνωρίζουμε κάθε πιθανό παράθυρο (οριζόμενο και από το κεντρικό pixel) τότε έχουμε κατά προσέγγιση έναν μεγάλο αριθμό “n” παραθύρων τα οποία είναι κατά κάτι λιγότερα από τα αρχικά pixel της εικόνας δοκιμής. Τα παράθυρα είναι λίγο λιγότερα από τα συνολικά pixel, γιατί κάποια παράθυρα 33*32 τα οποία βρίσκονται στα όρια της εικόνας δοκιμής, δεν καλύπτουν εξ ολοκλήρου την εικόνα, με αποτέλεσμα να μην υπάρχουν τα απαραίτητα δεδομένα σε αυτές τις περιπτώσεις.

Παρακάτω φαίνονται τα παράθυρα για τα οποία μιλάμε:



Εικόνα 25 – Αναπαράσταση περιοχών αφαίρεσης από περαιτέρω επεξεργασία

Το πρώτο πράγμα που κάνει η συνάρτηση είναι να φορτώσει την τράπεζα φίλτρων Gabor (πυρήνες Gabor) που δημιουργήθηκε από την συνάρτηση `create_gabor`. Υπενθυμίζεται ότι τα φίλτρα Gabor έχουν ήδη μετασχηματιστεί στον χώρο των συχνοτήτων μέσω του δισδιάστατου διακριτού μετασχηματισμού Fourier.

```
function IMVECTOR = im2vec (W33x32)
load gabor;
[gr_im_tr,st_im_tr] = size(W33x32);
gr_im_gabor = 32;
st_im_gabor = 32;
```

Η αλλαγή της αντίθεσης (contrast) ή της φωτεινότητας (brightness) αναπαριστά στην πραγματικότητα τις αλλαγές φωτισμού (illumination). Τι γίνεται στην περίπτωση που αλλάζει η πηγή φωτισμού για το αντικείμενό μας; Στην περίπτωση αυτή οι επιπτώσεις φωτισμού δεν είναι γραμμικές και σε καμία περίπτωση δεν είναι ίδιες για ολόκληρη την εικόνα. Αυτό φαίνεται πολύ εύκολα στις παρακάτω εικόνες:



Εικόνα 26 – Αναπαράσταση χαρακτηριστικών περιπτώσεων διαφορετικής αντίθεσης και φωτισμού σε εικόνα δοκιμής

Υπάρχουν δύο τρόποι να διορθώσουμε το παραπάνω πρόβλημα.

- 1ος Τρόπος-α (Background Illumination Model)

Ένας τρόπος είναι αφαιρώντας τον Μ.Ο από όλα τα pixels ή φιλτράροντας την εικόνα με Γκαουσιανό (Gaussian) φίλτρο. Ακολούθως, αν αφαιρεθεί το παραπάνω αποτέλεσμα από την αρχική εικόνα, τότε αυτό που μένει είναι το αντικείμενο χωρίς την επιρροή του υποβάθρου.

Αφαίρεση Μ.Ο:



Εικόνα 27– Παράδειγμα εξασθένησης επιρροής background με την αφαίρεση επί της αρχικής εικόνας του Μ.Ο

```

W33x32 = imread('im7.jpg');
figure(1); imshow(W33x32);
B = sum(sum(W33x32));
[d1,d2] = size(W33x32);
c = B/(d1*d2);
AAA = zeros(d1,d2);
for i=1:d1
    for j=1:d2
        AAA(i,j)=W33x32(i,j)-c;
    
```

- 1ος Τρόπος-β (Gaussian filter)



Εικόνα 28 - Παράδειγμα εξασθένησης επιρροής background με τη χρήση φίλτρου Gauss

```
W33x32 = imread('im7.jpg');  
  
h = fspecial('gaussian', 7, 1.5);  
Gaussianimage = imfilter(W33x32,h,'same')  
figure(3); imshow(Gaussianimage);  
  
[d1,d2] = size(W33x32);  
  
AAA = zeros(d1,d2);  
for i=1:d1  
    for j=1:d2  
        AAA(i,j)=W33x32(i,j)-Gaussianimage(i,j);  
    end  
end
```

- 2ος Τρόπος (Adaptive Histogram Equalization)

Ο δεύτερος τρόπος να επιλύσεις το παραπάνω πρόβλημα είναι χρησιμοποιώντας την ισοστάθμιση ιστογράμματος αλλά όχι για ολόκληρη την εικόνα. Ο υπολογισμός

προσαρμόζεται πραγματοποιούμενος σε μικρές περιοχές, ορισμένων pixels. Η εντολή στην Matlab είναι η παρακάτω:

```
W33x32 = adapthisteq(W33x32,'Numtiles',[8 3]);
```

Όπως προαναφέραμε, η ισοστάθμιση ιστογράμματος της εικόνας πραγματοποιείται έτσι ώστε να είναι μελλοντικά, ισχυρή σε διάφορες περιπτώσεις, έντονης αντίθεσης ή έντονου φωτισμού. Η ισοστάθμιση γίνεται σε tiles [8,3] pixels. Η Matlab έχει σαν default διάσταση παραθύρου την [8*8].

Ακολούθως ορίζεται ένας υπερ-πίνακας cell "Feature165*256" ο οποίος περιλαμβάνει στο εσωτερικό του 40 πίνακες τοποθετημένους στην ίδια φιλοσοφία με τον πίνακα G} των φίλτρων Gabor. Δηλαδή έχει 5 γραμμές (5 διαφορετικές συχνότητες) και 8 στήλες (8 διαφορετικές διευθύνσεις /22,5°). Ο κάθε πίνακας θα κρατάει την απάντηση από κάθε αντίστοιχο πυρήνα Gabor.

Το αποτέλεσμα είναι μιγαδικός αριθμός, για αυτό χρησιμοποιείται η εντολή "abs" για να μας δώσει το μέτρο ($\text{magnitude} = \sqrt{x^2+y^2}$) το οποίο είναι πραγματικός αριθμός. Η εικόνα εισαγωγής μετατρέπεται και αυτή στο πεδίο των συχνοτήτων μέσω του γρήγορου μετασχηματισμού Fourier.

```
A = fft2 (double(W33x32),gr_im_gabor,st_im_gabor)
```

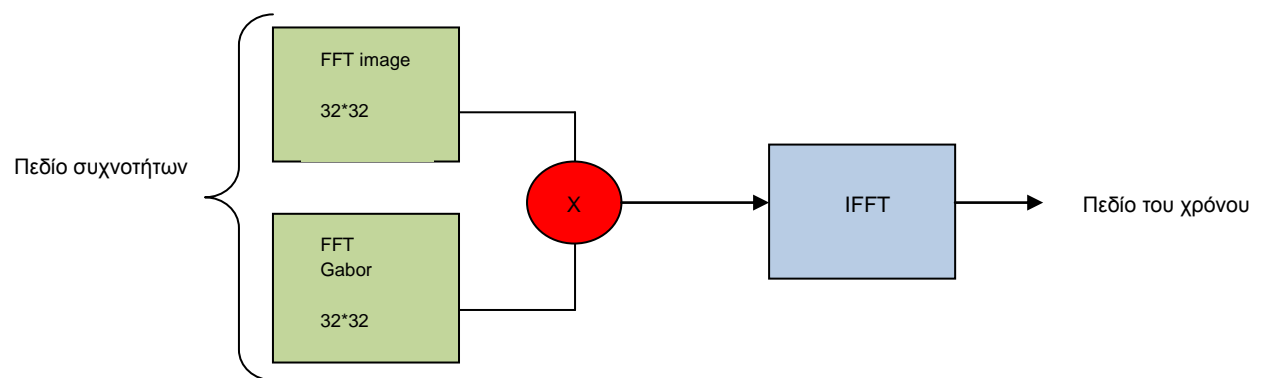
Στον μετασχηματισμό αυτό δίνονται ορίσματα οι διαστάσεις του πίνακα Gabor, δηλαδή 32*32 και αυτό επειδή για να πραγματοποιηθεί η πράξη της συνέλιξης με το αντίστοιχο φίλτρο Gabor, οι διαστάσεις πρέπει να είναι προφανώς ίδιες.

Το επόμενο βήμα είναι να πραγματοποιηθεί ο πολλαπλασιασμός της εικόνας (παραθύρου πλέον 32*32 στο πεδίο συχνοτήτων) με τον κάθε πίνακα Gabor (για όλες τις 40 διαφορετικές περιπτώσεις) pixel προς pixel. Αυτό υλοποιείται με την πολλαπλασιασμό πινάκων χρησιμοποιώντας τον συμβολισμό «.*».

```
B = G{s,j}.*A
```

Το παραπάνω αποτέλεσμα αφορά στο πεδίο συχνοτήτων και για να είναι αξιοποιήσιμο θα πρέπει να γίνει η επαναφορά στον χώρο του χρόνου χρησιμοποιώντας τον αντίστροφο μετασχηματισμό Fourier. Σε αυτόν τον μετασχηματισμό τα ορίσματα που δίνουμε για τον πίνακα που θα δημιουργήσουμε είναι οι αρχικές διαστάσεις της εικόνας εισόδου στην συνάρτηση, δηλαδή 33*32, και αυτό για να επανέλθει στις αρχικές της διαστάσεις.

Παρατήρηση 1: Για να χρησιμοποιήσουμε την συνάρτηση FFT θα πρέπει και τα δύο μέρη να είναι διαστάσεων δυνάμεως του 2. Τα φίλτρα Gabor ήταν ήδη, ενώ την εικόνα την κάναμε στο προηγούμενο βήμα. Τονίζεται ότι η μέγιστη κλίμακα του φίλτρου Gabor δεν πρέπει ξεπερνάει την διάσταση 32*32 γιατί το κύμα θα πρέπει να εσωκλείεται μέσα στον πυρήνα.



Σχήμα 16 - Αναπαράσταση πράξης συνέλιξης μέσω του FFT και του IFFT

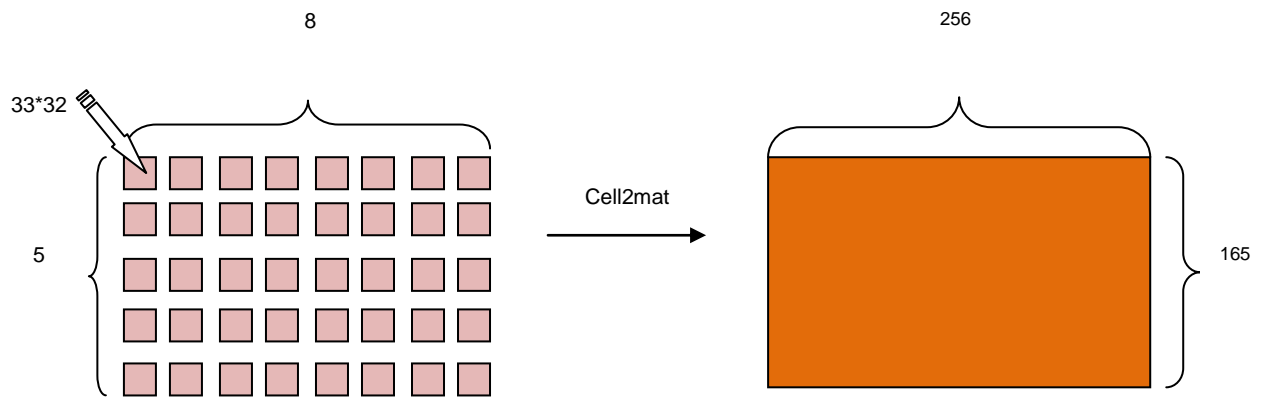
Παρατήρηση: Πάντα θα πρέπει να μειώνονται οι διαστάσεις της συνέλιξης στο μισό των διαστάσεων του πυρήνα Gabor. (Αιτία = Σχεδίαση των φίλτρων Gabor έτσι ώστε οι αποκρίσεις ημίσεως εύρους των φίλτρων να εφάπτονται)

```

Features165x256 = cell(5,8);
for s = 1:5
    for j = 1:8
        Features165x256{s,j} = abs(iff2(B,gr_im_tr,st_im_tr));
    end
end
end
  
```

Η εντολή cell2mat μετατρέπει τον υπερπίνακα «Features165x256» σε έναν ενιαίο πίνακα. Ουσιαστικά ενώνει τους 5*8 = 40 πίνακες σε έναν ενιαίο. Ο πίνακας αυτός θα έχει διαστάσεις :

1. Στήλες : $8 \times 32 = 256$
2. Γραμμές : $5 \times 33 = 165$



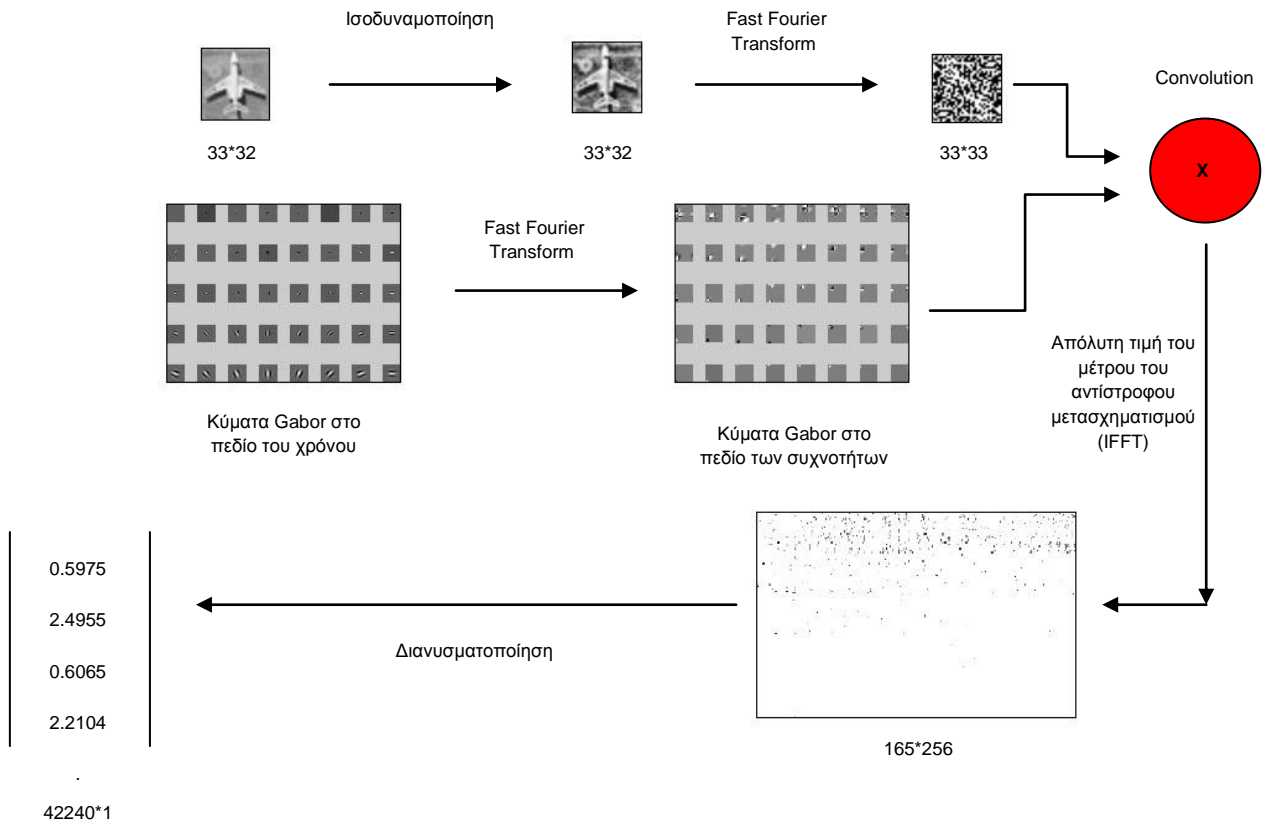
Σχήμα 17 - Αναπαράσταση μετατροπής πίνακα cell σε ενιαίο πίνακα

```
Features165x256 = cell2mat(Features165x256);
```

Τέλος, πραγματοποιείται η μετατροπή του παραπάνω πίνακα σε έναν πίνακα διάνυσμα διαστάσεων $m \times 1$, όπου $m = 165 \times 256 = 42240$. Το νευρωνικό δίκτυο για να εκπαιδευτεί για ένα διάνυσμα χαρακτηριστικών, όπως αυτό, θα πρέπει να διαθέτει έναν πάρα πολύ μεγάλο αριθμό νευρώνων εισαγωγής, κάτι το οποίο δεν είναι λογικό εκτιμώντας τον χρόνο που χρειαζόμαστε για εκπαίδευση και για προσομοίωση. Σε αυτό το στάδιο κάτι το οποίο θα μπορούσε να μας βοηθήσει για να βελτιώσουμε τον χρόνο απόκρισης του προγράμματος θα ήταν να μειώναμε τις διαστάσεις του τελικού πίνακα (συνήθως με κάποιον αλγόριθμο μείωσης διαστάσεων όπως ο PCA-Principal Component Analysis).

```
GRAMMES_STHLH = (gr_im_tr*5)*(st_im_tr*8);
IMVECTOR = reshape (Features165x256,[GRAMMES_STHLH 1]); %
```

Παρακάτω παρουσιάζεται συνολικά η διαδικασία που ακολουθείται από την συνάρτηση `im2vec.m`:



Σχήμα 18 - Αναπαράσταση των σταδίων που λαμβάνουν χώρα στην συνάρτηση `im2vec`

Συνάρτηση `Loadimages.m`

Η συνάρτηση `loadimages` χρησιμοποιείται για την φόρτωση των εικόνων εκπαίδευσης για το νευρωνικό δίκτυο που θα κατασκευάσουμε στην συνάρτηση `trainnet`. Αρχικά, εισάγουμε τα ονόματα των φακέλων σε μεταβλητές `string` για να μας βοηθήσει μετέπειτα στην αυτόματη ανάγνωση όσων εικόνων υπάρχουν στο εσωτερικό των αντίστοιχων φακέλων. Τα δεδομένα εκπαίδευσης ανήκουν σε δύο κατηγορίες, τα δεδομένα που αποτελούν τον επιθυμητό στόχο και τα δεδομένα που δεν περιλαμβάνουν τον στόχο. Και οι δύο κατηγορίες είναι απαραίτητες για την εκπαίδευση του νευρωνικού δικτύου. Θα πρέπει να σημειώσουμε ότι, αν θέλουμε να διαβάσει διαφορετικό `format` εικόνων, απλά αλλάζουμε την επιλογή `“png”`.

Ακολούθως, ορίζει σαν επιθυμητές μεταβλητές εξόδου του νευρωνικού δικτύου τις τιμές 0 και 1 για τα μη-αεροσκάφη και τα αεροσκάφη αντίστοιχα.

```
plane_folder = 'Αεροσκάφη/';
non_plane_folder = 'Μή Αεροσκάφη/';
```



```
file_ext = '.png';  
out_max = 1;  
out_min = 0;
```

Η συνάρτηση `loadimages` έχει ως αποτέλεσμα ένα αρχείο που λέγεται `IMDB.mat` και το οποίο περιλαμβάνει όλες τις εικόνες μαζί με τις τιμές εξόδου που τους αντιστοιχεί (0 ή 1) και το αντίστοιχο `path` εντοπισμού. Για οικονομία χρόνου και υπολογιστικού φόρτου, το αρχείο αυτό στην περίπτωση που υπάρχει φορτώνεται αυτόματα και η συνάρτηση `loadimages` τερματίζει χωρίς να προχωρήσει στα επόμενα βήματα. Το αρχείο `imdb` είναι και αυτό ένας υπερπίνακας (`cell`), όπως και ο πίνακας `G` που αναλύσαμε προηγουμένως.

```
if exist('imgdb.mat','file')  
    load imgdb;  
else  
    IMGDB = cell (3,[]);  
End
```

Το επόμενο βήμα είναι η καταχώρηση στην μεταβλητή “`nplane`” του αριθμού όλων των φωτογραφιών που υπάρχουν στον φάκελο «Αεροπλάνων» έτσι ώστε να γνωρίζουμε για πόσο θα τρέχει ο «`for loop`». Έπειτα, ξεκινάει η ανάγνωση των εικόνων μία προς μία. Σε περίπτωση που η εικόνα δεν έχει διαστάσεις `33*32` την παρακάμπτει και συνεχίζει στην επόμενη.

Τονίζεται ότι τα `Dataset` των εικόνων είχαν προσαρμοστεί στις ανάγκες του κώδικα και είχαν μετατραπεί όλες οι εικόνες σε `32*33pixels – grayscale`.

```
fprintf ('Φόρτωση εικόνων A/Φ');  
folder_content = dir ([face_folder, '*', file_ext]);  
nplane = size (folder_content,1);  
for k=1:nplane  
    string = [plane_folder, folder_content(k,1).name];  
    image = imread(string);  
    [m n] = size(image);  
  
    if (m~=33 || n~=32)  
  
        continue;  
    end  
    f=0;
```

Έπειτα, κάνει έλεγχο στο όνομα της εικόνας που μόλις διάβασε με το αρχείο IMDB που τυχόν μπορεί να είχε φορτώσει προηγουμένως. Σε περίπτωση που είναι ίδιο απλά την προσπερνάει και πάει στο επόμενο βήμα. Ο έλεγχος γίνεται για όλες τις εικόνες του IMDB με την εικόνα που μόλις διάβασε.

```
for i=1:length(IMGDB)

    if strcmp(IMGDB{1,i},string)
        f=1;
    end
end
if f==1
    continue;
End
```

Ακολουθως, για κάθε εικόνα που διάβασε (με A/Φ) παράγει 10 διαφορετικά αντίγραφα. Οι εικόνες που παράγονται είναι ουσιαστικά διάφοροι συνδυασμοί κατοπτρικών εικόνων άλλοτε από αριστερά προς τα δεξιά και άλλοτε από πάνω προς τα κάτω. Αυτό γίνεται για να εμπλουτίσει τα δεδομένα εκπαίδευσης με 10πλάσιο αριθμό εικόνων για διάφορες παραπλήσιες περιπτώσεις. Τέλος, καταχωρεί στον υπερπίνακα imgdb στην πρώτη γραμμή το όνομα αρχείου (Αεροσκάφη/i.png), στην δεύτερη γραμμή την τιμή εξόδου που πρέπει να έχει το νευρωνικό δίκτυο και στην τρίτη γραμμή καταχωρείται η εικόνα σε διάνυσμα στήλης.

Τονίζεται ότι η εικόνα μετατρέπεται σε διάνυσμα διαστάσεων [m*1], μέσω της συνάρτησης im2vec (image to vector). Στην εικόνα 29 φαίνεται πώς έχει διαμορφωθεί ο πίνακας Imgdb (Image Database).

```
fprintf ('.');
IM {1} = im2vec (image);
IM {2} = im2vec (fliplr(image));

IM {3} = im2vec (circshift(image,1));
IM {4} = im2vec (circshift(image,-1));
IM {5} = im2vec (circshift(image,[0 1]));
IM {6} = im2vec (circshift(image,[0 -1]));
IM {7} = im2vec (circshift(fliplr(image),1));
IM {8} = im2vec (circshift(fliplr(image),-1));
IM {9} = im2vec (circshift(fliplr(image),[0 1]));
IM {10} = im2vec (circshift(fliplr(image),[0 -1]));
for i=1:10
```

```

    IMGDB {1,end+1}= string;
    IMGDB {2,end} = out_max;
    IMGDB {3,end} = {IM{i}};
end
end

```

IMGDB <3x4436 cell>					
	1	2	3	4	5
1	'Αεροσκάφη/1.png'	'Αεροσκάφη/1.png'	'Αεροσκάφη/1.png'	'Αεροσκάφη/1.png'	'Αεροσκάφη/1.png'
2	1	1	1	1	1
3	<42240x1 double>	<42240x1 double>	<42240x1 double>	<42240x1 double>	<42240x1 double>

Εικόνα 29 - Στιγμιότυπο από τον πίνακα imgdb

Το τελευταίο βήμα ουσιαστικά είναι μια επανάληψη όλων των προηγούμενων, για την κατηγορία των δεδομένων εκπαίδευσης «Μη Αεροπλάνων». Ακολουθούνται τα ίδια βήματα με την διαφορά ότι αναπαράγονται από κάθε εικόνα τρεις νέες εικόνες και όχι δέκα. Ουσιαστικά τα αντίτυπα που παράγονται εκφράζουν το βάρος της κάθε κατηγορίας στην διαδικασία της εκπαίδευσης του νευρωνικού δικτύου. (Αεροσκάφη 10 – Μη Αεροσκάφη 4)

```

fprintf ('\nΦόρτωση εικόνων μή Α/Φ ');
folder_content = dir ([non_face_folder, '*', file_ext]);
nnplane = size (folder_content,1);
for k=1:nnface
    string = [non_plane_folder, folder_content(k,1).name];
    image = imread(string);
    [m n] = size(image);
    if (m~=33 || n~=32)
        continue;
    end
    f=0;
    for i=1:length(IMGDB)
        if strcmp(IMGDB{1,i},string)
            f=1;
        end
    end
    if f==1
        continue;
    end
    fprintf ('.');
    IM {1} = im2vec (image);
    IM {2} = im2vec (fliplr(image));
    IM {3} = im2vec (flipud(image));
    IM {4} = im2vec (flipud(fliplr(image)));
    for i=1:4

```

```
        IMGDB {1,end+1}= string;
        IMGDB {2,end} = out_min;
        IMGDB {3,end} = {IM{i}};
    end
end
fprintf('\n');
save imgdb IMGDB;
```

Συνάρτηση *Mminmax.m*

Δεν πρέπει να ξεχνάμε το γεγονός ότι κάθε γραμμικοποιημένο διάνυσμα θα πρέπει να έχει κανονικοποιηθεί σε ένα προκαθορισμένο εύρος. Το πιο συνηθισμένο είναι το [-1,1], ενώ είναι καλό να παρουσιάζει συμμετρία γύρω από το μηδέν, γιατί θα μας βοηθήσει για το στάδιο της εκπαίδευσης και ειδικά όταν χρησιμοποιούμε μια ομοιόμορφη αρχικοποίηση.

```
function output = mminmax(input)

max_ = max(max(input));

min_ = min(min(input));

output = ((input-min_)/(max_-min_) - 0.5) * 2;
```

Συνάρτηση *Trainet.m*

Το στάδιο της εκπαίδευσης σε ένα νευρωνικό δίκτυο, όπως έχουμε προαναφέρει στο Χ Κεφάλαιο, είναι πολύ απλά μια διαδικασία κατά την οποία το νευρωνικό δίκτυο διαρκώς αλλάζει τα συναπτικά βάρη και το κατώφλι (bias) με τέτοιο τρόπο ώστε όταν κάνει έλεγχο στα δεδομένα εκπαίδευσης να πετυχαίνει την επιθυμητή έξοδο.

Αρχικά, ορίζουμε τον επιτρεπτό αριθμό επαναλήψεων (στην συγκεκριμένη περίπτωση είναι 100000). Ακολούθως, εισάγουμε τα απαραίτητα ορίσματα στην συνάρτηση δημιουργίας του Τεχνητού Νευρωνικού Δικτύου (svmtrain). Το βασικό όρισμα που παίρνει είναι ο υπερπίνακας IMGDB, όμως πίσω από αυτό κρύβονται τα πραγματικά ορίσματα της έτοιμης συνάρτησης "svmtrain". Αρχικά, θέλει τις τιμές από όλες τις γραμμικοποιημένες εικόνες εκπαίδευσης [P = cell2mat(IMGDB(3,:))] ή αλλιώς, δεδομένα εκπαίδευσης και ακολούθως τις αντίστοιχες τιμές εκπαίδευσης αντικείμενου αναζήτησης (Αεροπλάνο == 1 & Μη Αεροπλάνο == 0) [P = cell2mat(IMGDB(2,:))] ή αλλιώς, δεδομένα ομαδοποίησης.

```
function NET = trainnet(IMGDB)

options = optimset('maxiter',100000);%Maximum number of iterations allowed

fprintf('Δημιουργία & εκπαίδευση της μηχανής ->\n');
```

```

fprintf('Η διαδικασία αυτή μπορεί να διαρκέσει αρκετά λεπτά, \n');
fprintf('Παρακαλώ περιμένετε....\n');

T = cell2mat(IMGDB(2,:));

P = cell2mat(IMGDB(3,:));

net
=
svmtrain(P,T,'Kernel_Function','linear','Polyorder',2,'quadprog_opts',options);

```

Οι λοιπές παράμετροι της svmtrain ορίζουν τον τρόπο εκπαίδευσης του ΤΝΔ. Πιο αναλυτικά:

Παρατήρηση: Μετά τα δεδομένα εκπαίδευσης και τα δεδομένα ομαδοποίησης δύναται να ορισθούν διάφορες παράμετροι απλά ορίζοντας το όνομά τους μέσα σε εισαγωγικά και ακολούθως χωρίς εισαγωγικά την αντίστοιχη τιμή της παραμέτρου (π.χ: 'Polyorder',2). Αυτό μπορεί να γίνει για όλες τις δυνητικά διαθέσιμες παραμέτρους αρκεί να διατηρηθεί η δομή:

(Όνομα_Παραμέτρου_1,Τιμή_Παραμέτρου_1, Όνομα_Παραμέτρου_2,Τιμή_Παραμέτρου_2, ...)

- *Autoscale*

Τύπος δεδομένων Αληθείας (Boolean), που καθορίζει εάν η συνάρτηση svmtrain κεντράρει τα σημεία εισόδου στον μέσο όρο και αλλάζει την κλίμακά τους έτσι ώστε να έχουν τυπική απόκλιση ίση με την μονάδα πριν το στάδιο της εκπαίδευσης. Η default τιμή είναι true.

- *Boxconstraint*

Πραγματοποιεί μια κανονικοποίηση των δεδομένων εισαγωγής P και T με βάση τους παρακάτω τύπους:

1. $N \cdot (2 \cdot N_1)$, όπου N_1 είναι ο αριθμός των στοιχείων του πίνακα P.
2. $N \cdot (2 \cdot N_2)$, όπου N_2 είναι ο αριθμός των στοιχείων του πίνακα T.

Όπου $N = N_1 + N_2$. Αυτή η κανονικοποίηση πραγματοποιείται στην περίπτωση που τα δεδομένα εισόδου έχουν πάρα πολύ διαφορετικές τιμές. Η default τιμή είναι 1.

- *Kernelcachelimit*

Παράμετρος που καθορίζει το μέγεθος του πίνακα του πυρήνα για την μέθοδο εκπαίδευσης SMO (Sequential Minimal Optimization). Η default τιμή είναι 5000.

- *Kernel_function*

Η συνάρτηση πυρήνα χρησιμοποιείται για την χαρτογράφηση των δεδομένων εκπαίδευσης στον χώρο του πυρήνα. Η default μέθοδος αναπαράστασης είναι με τελείες. Η συνάρτηση μπορεί να είναι μια από τις παρακάτω μεθοδολογίες:

1. *Linear*

Γραμμικός πυρήνας που σημαίνει προϊόν με τελείες.

2. *Quadratic*

Τετραγωνικός πυρήνας

3. *Polynomial*

Πολυωνυμικός πυρήνας. Η default διάσταση είναι 3 (3ου βαθμού)

4. *Rbf*

Ακτινική Γκαουσιανή Βάση Συνάρτησης Πυρήνα, με αρχική τιμή κλίμακας, $[\sigma = 1]$.

5. *Mlp*

Πολλαπλών επιπέδων νευρώνας με default κλίμακα $[1 \ -1]$.

- *KKtviolationlevel*

Η παράμετρος αυτή καθορίζει το κλάσμα των μεταβλητών που επιτρέπεται να καταστρατηγήσουν τις KKT-συνθήκες κατά την SMO μέθοδο εκπαίδευσης. Για παράδειγμα, εάν θέσουμε την παραπάνω μεταβλητή στο 0,05 τότε το 5% των μεταβλητών επιτρέπεται να καταστρατηγήσουν τις KKT-συνθήκες.

- *Method*

Μέθοδοι που χρησιμοποιούνται για τον εντοπισμό του επιπέδου διαχωρισμού των κλάσεων. Οι μέθοδοι που διατίθενται είναι οι κάτωθι:

1. *QP*

Τετραγωνικός προγραμματισμός (απαιτείται το Optimization Toolbox license). Ο ταξινομητής (classifier) αποτελείται από διπλή νόρμα ελαστικών ορίων support Vector Machine.

2. *SMO*

Επαναλαμβανόμενη Βελτιστοποίηση. Αυτή είναι η default προεπιλεγμένη μεθοδολογία.

3. *LS*

Μέθοδος Ελαχίστων Τετραγώνων.

- *Mlp_params*

Παράμετροι που είναι απαραίτητοι για το Πολυεπίπεδο Νευρώνα. Ουσιαστικά αφορά τις τιμές των δύο ορίων στις οποίες πάνω και κάτω αντίστοιχα θα υλοποιείται η ταξινόμηση στις αντίστοιχες κλάσεις. Η default προεπιλογή είναι [1 -1].

- *Options*

Επιλογές της δομής εκπαίδευσης. Όταν η μέθοδος εκπαίδευσης είναι η 'SMO' καθορίζονται οι δομές χρησιμοποιώντας την εντολή 'statset'. Οι επιλογές είναι οι εξής:

1. *Display*

Περιγράφει την εξέλιξη του αριθμού των επαναλήψεων που πραγματοποιούνται.

- Off : Είναι η default προεπιλογή και δεν αναφέρει τίποτα για την εξέλιξη των επαναλήψεων.
- Iter: Αναφέρει σε κάθε 500 επαναλήψεις
- Final: Αναφέρει μόνο το σύνολο των επαναλήψεων στο τέλος του αλγορίθμου.

2. *MaxIter*

Ακέραιος ο οποίος καθορίζει τον μέγιστο επιτρεπόμενο αριθμό επαναλήψεων. Η default προεπιλογή είναι 15000.

- *Polyorder*

Ορίζει τον πολυωνυμικό πυρήνα με default διάσταση 3.

- *Rbf_sigma*

- *Showplot*

Τύπος δεδομένων Αληθείας (Boolean), ο οποίος είναι default ψευδής και καθορίζει το αν θα σχεδιαστεί η γραμμή διαχωρισμού. Υλοποιεί την σχεδίαση μόνο όταν τα δεδομένα είναι σε δύο στήλες.

- *Tolkk*

Τιμή που καθορίζει την ανοχή στην οποία οι συνθήκες-KKT ελέγχονται.

- *SupportVectors*

Πίνακας από δεδομένα σημείων όπου κάθε γραμμή αφορά σε ένα υποστηριζόμενο διάνυσμα στον κανονικοποιημένο χώρο. Αυτός ο πίνακας είναι ένα υποσύνολο του πίνακα των δεδομένων εκπαίδευσης, αφού έχει προηγηθεί πρώτα η κανονικοποίηση των δεδομένων.

- *Alpha*

Διάνυσμα βαρών για τα υποστηριζόμενα διανύσματα. Το πρόσημο για τα υποστηριζόμενα στην πρώτη ομάδα δεδομένων είναι θετικό και αρνητικό για την δεύτερη.

- *Bias*

Κατώφλι για τον καθορισμό των επιπέδων που διαχωρίζουν τις δύο κλάσεις.

- *SupportVectorIndices*

Διάνυσμα δεικτών το οποίο καθορίζει τις γραμμές των δεδομένων εκπαίδευσης οι οποίες επιλέχθηκαν ως υποστηριζόμενα διανύσματα αφού τα δεδομένα κανονικοποιήθηκαν. Παρακάτω βλέπουμε το σημείο του κώδικα στο οποίο εμφανίζονται τα τελικά επιλεγμένα διανύσματα υποστήριξης (support vector).

```
fprintf('Αριθμός διανυσμάτων υποστήριξης (SVM=support vector machines):\n', size(net.SupportVectors,1));
```

- *ScaleData*

- *FigureHandles*

Τέλος, έχουμε την εντολή `svmclassify`. Η δομή της συνάρτησης είναι η εξής:

Συνάρτηση `svmclassify(SVMStruct,Sample).m`

Η συνάρτηση αυτή έχει αποστολή να ομαδοποιήσει κάθε γραμμή από τα δεδομένα του πίνακα `sample` (ο πίνακας `P`), χρησιμοποιώντας τις πληροφορίες σε μια δομή κατηγοριοποίησης μηχανής υποστηριζόμενων διανυσμάτων. (Support Vector

Machine Classifier Structure). Όπως τα δεδομένα εκπαίδευσης χρησιμοποιήθηκαν για να δημιουργήσουν την δομή SVMstruct, έτσι και ο πίνακας Sample είναι ένας πίνακας όπου κάθε γραμμή ανταποκρίνεται σε μια ανταπόκριση ή μια αναπαράσταση και κάθε στήλη ανταποκρίνεται σε ένα χαρακτηριστικό ή μια μεταβλητή. Συνεπώς, ο πίνακας του δείγματος 'sample' (P) πρέπει να έχει την ίδια διάσταση στηλών με τα δεδομένα εκπαίδευσης. Αυτό οφείλεται στο γεγονός ότι ο αριθμός των στηλών ορίζει τον αριθμό των χαρακτηριστικών.

```
classes = svmclassify(net,P');  
fprintf('done. %d \n',sum(abs(classes-T')));  
save net net  
NET = net;
```

Η εντολή svmclassify συνάρτηση χρησιμοποιεί τα αποτελέσματα της svmtrain για να ταξινομήσει τα διανύσματα 'x' σύμφωνα με την παρακάτω εξίσωση:

$$C = \sum_i a_i * k(s_i, x) + b$$

Όπου s_i είναι τα υποστηριζόμενα διανύσματα, a_i είναι τα βάρη, b είναι το κατώφλι και k είναι συνάρτηση πυρήνα (kernel function). Στην περίπτωση ενός γραμμικού πυρήνα για να γίνει πιο κατανοητό, k είναι το προϊόν σημείων (dot product). Εάν το $c > 0$, τότε το x κατηγοριοποιείται σαν μέλος του πρώτου group, ενώ σε αντίθετη περίπτωση σαν μέλος του δεύτερου.

Συνάρτηση *Drawrec.m*

Η συνάρτηση *drawrec* έχει δημιουργηθεί για την επισήμανση με το κατά προτίμηση πλαίσιο, όλων των περιοχών ενδιαφέροντος. Το πλαίσιο τοποθετείται στο κέντρο μάζας της κάθε περιοχής που έχει εντοπιστεί από προηγούμενη συνάρτηση (*imscan*). Το παρακάτω πρότυπο σχεδιάζει ένα τετράγωνο κόκκινου χρώματος και πλευράς 16 pixels, ενώ ταυτόχρονα το σημείο ορισμού χρωματίζεται και αυτό κόκκινο.







```
function out = drawrec (in,w)
[m n]=size(in); [LUTm LUTn]=find(in); out = zeros (m,n);
for i =1:size(LUTm,1)
    try
        out (LUTm(i),LUTn(i))=1;
    end
    try
        out (LUTm(i)-8:LUTm(i)+8,LUTn(i)-8)=1; % stili LUTn(i)-9
    end
    try
        out (LUTm(i)-8:LUTm(i)+8,LUTn(i)+8)=1; % stili LUTn(i)+8
    end
    try
        out (LUTm(i)-8,LUTn(i)-8:LUTn(i)+8)=1;%grammi LUTm(i)-14
    end
    try
        out (LUTm(i)+8,LUTn(i)-8:LUTn(i)+8)=1;%grammi LUTm(i)+14
    end
end
end
```

Συνάρτηση *Imscan.m*

Η συνάρτηση “*imscan*” είναι μια συνάρτηση που δέχεται τέσσερα ορίσματα μεταξύ των οποίων και το ΤΝΔ που δημιουργήθηκε στην συνάρτηση “*trainnet*”. Τα υπόλοιπα ορίσματα είναι η εικόνα δοκιμής (η οποία πρέπει να είναι σε format *.jpg) και οι διαστάσεις των εικόνων εκπαίδευσης.

Για να γίνει πιο γρήγορα η αναζήτηση στην εικόνα δοκιμής, η συνάρτηση τροφοδοτείται και με δύο πρότυπα εικόνων από το αναζητούμενο αντικείμενο. Οι εικόνες αυτές έχουν αντίστοιχα ένα μαύρο και ένα γκρι υπόβαθρο για να εκμηδενίσουν τυχόν επιρροή στους υπολογισμούς των υποβάθρων τους τα οποία

μας είναι παντελώς αδιάφορα. Παρακάτω φαίνονται δύο παραδείγματα υποδειγμάτων που χρησιμοποιήθηκαν:

		
<p>Εικόνα προτύπου με σκούρο υπόβαθρο</p>	<p>Εικόνα προτύπου με ανοιχτό υπόβαθρο</p>	<p>Αρχική εικόνα</p>
		
<p>Εικόνα προτύπου με σκούρο υπόβαθρο</p>	<p>Εικόνα προτύπου με ανοιχτό υπόβαθρο</p>	<p>Αρχική εικόνα</p>

Εικόνα 30 – Παραδείγματα εικόνων προτύπων

Η διαδικασία που υλοποιείται είναι μια συσχέτιση (cross correlation) μεταξύ των δύο εικόνων υπολογίζοντας ουσιαστικά ένα μέτρο ομοιότητας. Υλοποιείται με την πράξη της συνέλιξης χωρίς να λαμβάνει χώρα στροφή του πυρήνα. Η μοναδική πρόθεση με την χρησιμοποίηση των εικόνων προτύπων είναι ο κατά προσέγγιση εντοπισμός των πιο πιθανών θέσεων στην εικόνα δοκιμής για εντοπισμό των αεροσκαφών.

Αφού διαβαστούν τόσο η εικόνα δοκιμής (im) όσο και οι εικόνες προτύπων, ακολούθως πραγματοποιείται μια κανονικοποίηση και των τριών σε τιμές μεταξύ [-1 1].

```

function im_out = imscan (net,im,grammes,styles)
close all
% PARAMETERS
SCAN_FOLDER = 'imscan/';
UT_FOLDER = 'imscan/under-thresh/';
TEMPLATE1 = 'Πρότυπο1.png';
TEMPLATE2 = 'Πρότυπο2.png';
warning off;
delete ([UT_FOLDER, '*.*']);
delete ([SCAN_FOLDER, '*.*']);

[m n]=size(im);
% First Section
C1 = mmminmax(double(im));
C2 = mmminmax(double(imread (TEMPLATE1)));% kai edo iso thelei allages
C3 = mmminmax(double(imread (TEMPLATE2)));% kai edo iso thelei allages

```

Η πράξη της συνέλιξης γίνεται, όπως φαίνεται και παρακάτω, με την εντολή “conv2” και πραγματοποιείται και για τις δύο εικόνες προτύπων. Τα αποτελέσματα της συνέλιξης τα χρησιμοποιούμε στην συνάρτηση “Imregionalmax” για να εντοπίσουμε τα μέγιστα και στις δύο εικόνες. Το αποτέλεσμα της κάθε εικόνας στην συνάρτηση Imregional είναι μια δυαδική εικόνα με τις ίδιες διαστάσεις με την αρχική, όπου η πλειονότητα των pixels είναι μηδέν, αλλά υπάρχουν και κάποια που είναι μονάδα (ON-pixels), ακριβώς στην θέση του κάθε περιφερειακού μεγίστου (Η μονάδα βρίσκεται στο κέντρο του κάθε παραθύρου 33*32 που πρέπει να ελεγχθεί).

```







Corr_1 = double(conv2 (C1,C2,'same'));



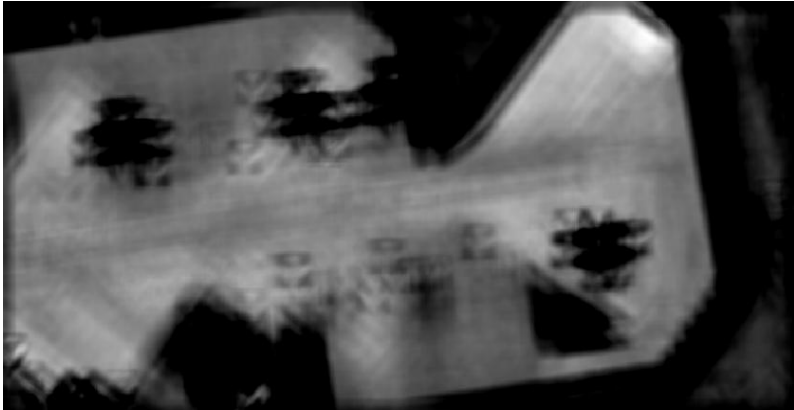

Corr_2 = double(conv2 (C1,C3,'same'));

Cell.state = int8(imregionalmax(Corr_1) | imregionalmax(Corr_2));

```

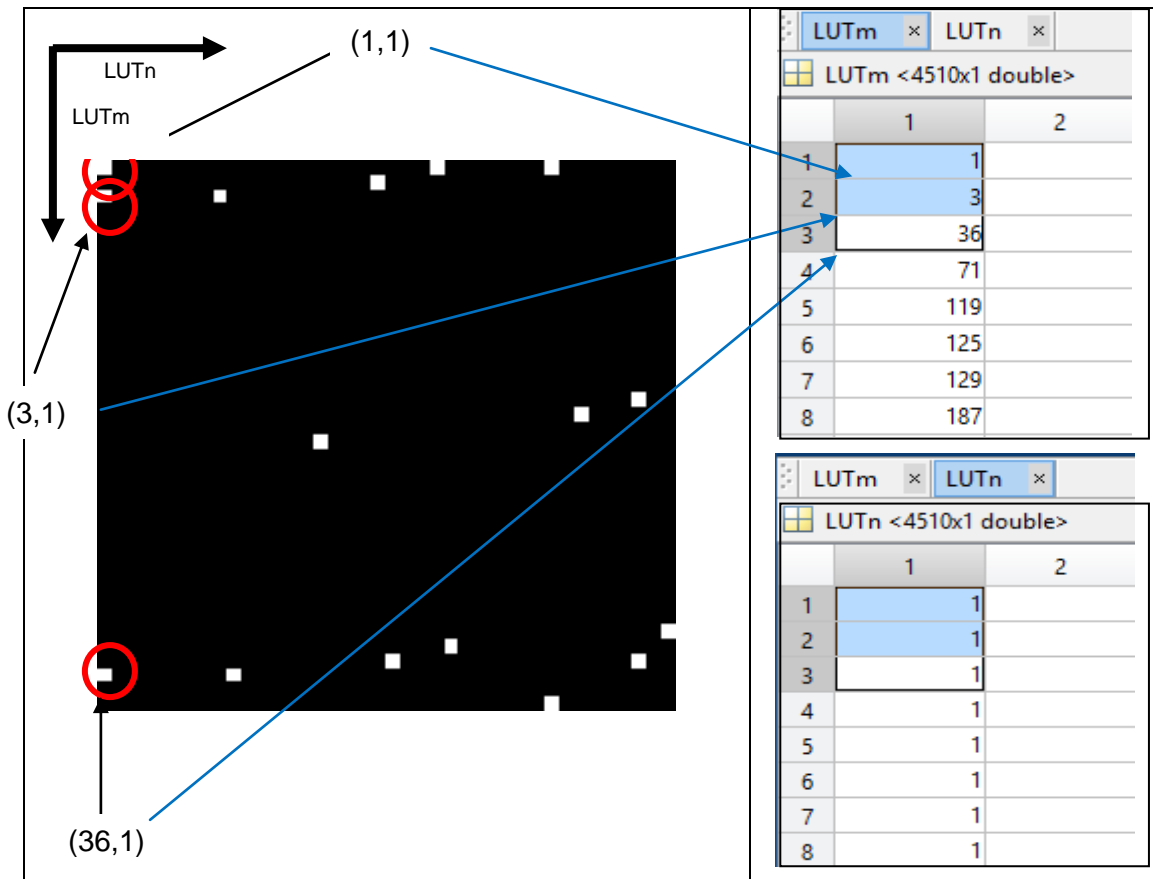
Ενδεικτικά αποτελέσματα των μεταβλητών όλης της αλληλουχίας αποτελεσμάτων φαίνεται παρακάτω για την καλύτερη κατανόηση:

<p>(im)</p> <p>Αρχική εικόνα δοκιμής</p>	
<p>(C1)</p> <p>Αρχική εικόνα δοκιμής κανονικοποιημένη σε εύρος [-1 1]</p>	
<p>(Πρότυπο 1)</p>	
<p>(C2)</p> <p>Πρότυπο1 κανονικοποιημένο σε εύρος [-1 1]</p>	
<p>(Πρότυπο 2)</p>	
<p>(C3)</p> <p>Πρότυπο2 κανονικοποιημένο σε εύρος [-1 1]</p>	

<p>(Corr_1)</p> <p>Αποτέλεσμα συνέλιξης με το πρότυπο 1</p>	
<p>Imregionalmax (Corr_1) (Τιμές 1 & 0)</p>	
<p>(Corr_2)</p> <p>Αποτέλεσμα συνέλιξης με το πρότυπο 2</p>	
<p>Imregionalmax (Corr_2) (Τιμές 1 & 0)</p>	

Πίνακας 11 – Βήματα επεξεργασίας εικόνας δοκιμής και πρότυπων εικόνων για την εξαγωγή σημείων ελέγχου

Έπειτα, μετατρέπονται κάποια σημεία των παραπάνω εξαγόμενων εικόνων σε «-1» έτσι ώστε να μην πραγματοποιηθεί αναζήτηση σε αυτά. Αυτά τα σημεία έχει επιλεγεί να είναι όσα είναι από τα όρια της εικόνας και 16 pixels εσωτερικά. Δεν πρέπει να ξεχνάμε ότι οι εικόνες εκπαίδευσης είναι 33*32 pixels. Ουσιαστικά τα σημεία στα οποία δεν χωράει το παράθυρο της εικόνας αποκλείονται από μελλοντικές αναζητήσεις. Ο πίνακας cell.net είναι άλλη μια εικόνα δυαδική, η οποία κρατάει τα αποτελέσματα του νευρωνικού δικτύου για κάθε pixel και αρχικοποιείται έχοντας όλα τα pixels τιμή «-1». Οι μεταβλητές LUTm και LUTn είναι δύο διανύσματα τα οποία καταγράφουν την θέση για κάθε ON-pixel σε καρτεσιανές συντεταγμένες. Ταυτόχρονα, εμφανίζεται η αρχική εικόνα δοκιμής και σε επίθεση αυτής τα σημεία ενδιαφέροντος ON-pixels σε χρώμα κίτρινο.



Σχήμα 19 - Αναπαράσταση του τρόπου καταχώρησης των καρτεσιανών συντεταγμένων


```

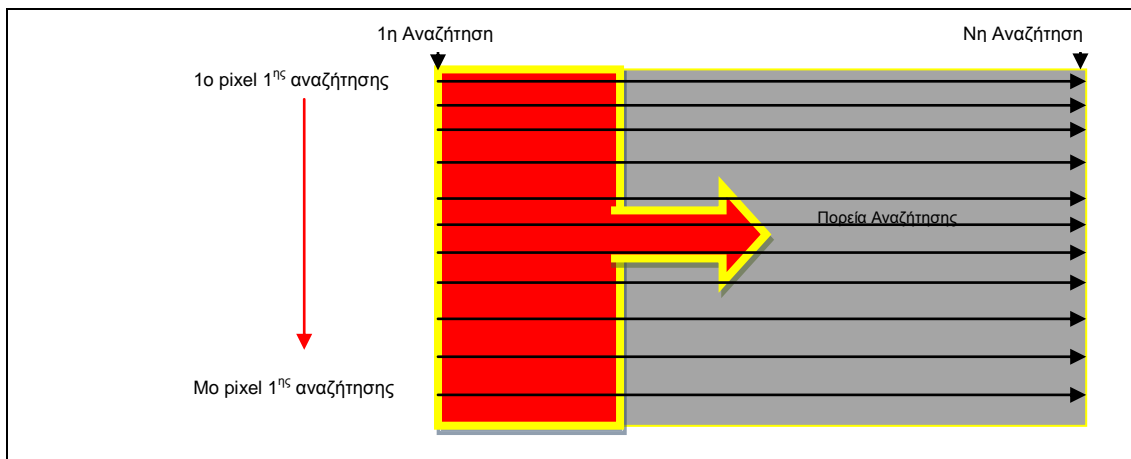
Cell.state(1:16,:)=-1;
Cell.state(end-16:end,:)=-1;
Cell.state(:,1:16)=-1;
Cell.state(:,end-16:end)=-1;
Cell.net = ones(m,n)*-1;
[LUTm LUTn]= find(Cell.state == 1);
imshow(im);
hold on
plot(LUTn,LUTm,'.y');pause(0.0001);

```

Το πρώτο πράγμα που πρέπει να κάνουμε είναι να δημιουργήσουμε έναν ατέρμονα κύκλο υπολογισμών. Το πρόγραμμα δεν θα πρέπει να μπορεί να σταματήσει, εάν δεν έχει ελέγξει όλα τα προσημειωμένα pixels (ON-pixels). Ο τρόπος δημιουργίας του ατέρμονα κύκλου (endless loop) υλοποιείται με την εντολή “while (1==1)”. Παρακάτω θα αναλυθεί ο τρόπος παύσης του κύκλου με συνθήκη που θα χρησιμοποιεί την εντολή “break”. Σε κάθε επανάληψη του κύκλου ελέγχεται μόνο ένα pixel. Κατά συνέπεια, το πρώτο πράγμα που πρέπει να κάνουμε είναι να εντοπίσουμε ένα pixel με την τιμή «1». Αυτό πραγματοποιείται με την παρακάτω εντολή:

```
[y x] = find(Cell.state==1,1);
```

Η εντολή find στην Matlab πάντα ξεκινάει την αναζήτηση των γραμμών από την κορυφή προς τα κάτω και από τα αριστερά προς τα δεξιά. Για αυτό και κατά την διαδικασία ελέγχου της εικόνας δοκιμής φαίνεται να επεξεργάζεται την εικόνα από αριστερά προς τα δεξιά. Η θέση του κάθε pixel που εντοπίζεται με τιμή καταχωρείται στις μεταβλητές y και x. Η παραπάνω διαδικασία παρουσιάζεται στο παρακάτω σχεδιάγραμμα:



Σχήμα 20- Διαδικασία αναζήτησης σε πίνακα M*N με την εντολή find

Οι επόμενες γραμμές αποτελούν την συνθήκη για τον τερματισμό του προγράμματος. Στην περίπτωση που δεν υπάρχει μεταβλητή y , τότε ο αλγόριθμος υποθέτει ότι δεν υπάρχει κάποιο εναπομένον pixel (με τιμή μονάδα) και τερματίζει τον κύκλο.

```
if isempty(y)
    break;
end
```

Στην συνέχεια, ο αλγόριθμος κόβει το παράθυρο (διαστάσεων τόσο όσο οι εικόνες εκπαίδευσης όσο και των προτύπων) στην εικόνα εκπαίδευσης με κέντρο του παραθύρου το pixel με τιμή ίση με «1».

```
imcut = im(y-16:y+16,x-16:x+15);
```

Έπειτα, για να είμαστε σίγουροι ότι δεν θα επανελεγχθεί το συγκεκριμένο Pixel, το μετατρέπουμε σε «-1», έτσι ώστε στην επόμενη επανάληψη να μην επιλεγθεί πάλι το ίδιο σημείο.

```
Cell.state(y,x) = -1;
```

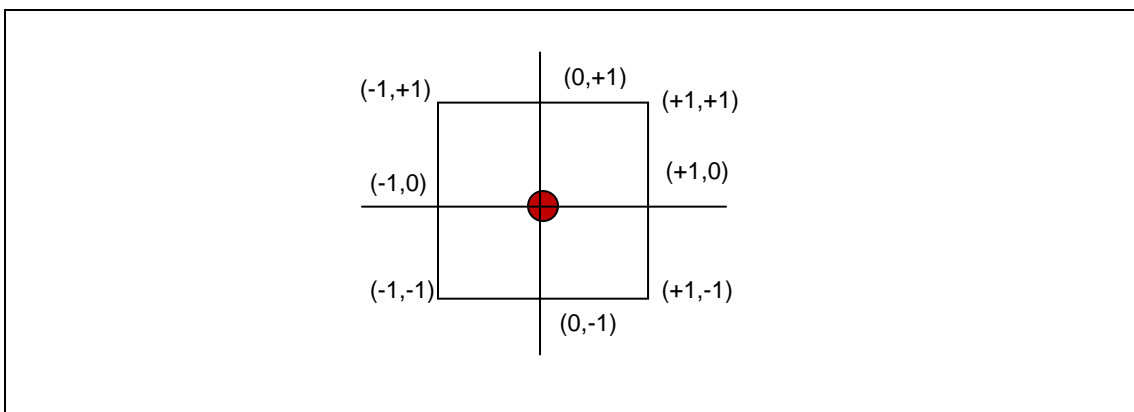
Μετά ακολουθεί η πιο σημαντική γραμμή της συνάρτησης. Το παράθυρο που έχει κοπεί διαστάσεων 33*32 θα ελεγχθεί από το Τεχνητό Νευρωνικό Δίκτυο. Όμως, πριν πραγματοποιηθεί αυτό θα πρέπει να μετατραπεί σε «εικόνα-διάνυσμα» μέσω της συνάρτησης `im2vec`, που έχουμε αναλύσει παραπάνω. Η “Cell.net” κρατάει το αποτέλεσμα από το ΤΝΔ. Σε περίπτωση που το αποτέλεσμα του ΤΝΔ είναι μονάδα (δηλαδή αεροσκάφος) τότε χρωματίζεται με πράσινο χρώμα, ενώ σε κάθε άλλη περίπτωση με το χρώμα κόκκινο.

```
Cell.net(y,x) = svmclassify(net,im2vec(imcut)');
if (Cell.net(y,x) == 1)
    plot(x,y,'.g');pause(0.0001);
```

Με βάση το παραπάνω αποτέλεσμα θα πρέπει να πάρουμε μια απόφαση για τα γειτονικά pixels. Για αυτό και δημιουργούνται δύο πίνακες με όλα τα πιθανά ζευγάρια των γειτονικών pixels. Οι πίνακες έχουν διαστάσεις 1*8 και καλύπτουν και τα 8 περιμετρικά pixels, όπως φαίνεται και στο παρακάτω σχήμα:

$$X_o = [-1, -1, -1, 0, 0, +1, +1, +1]$$

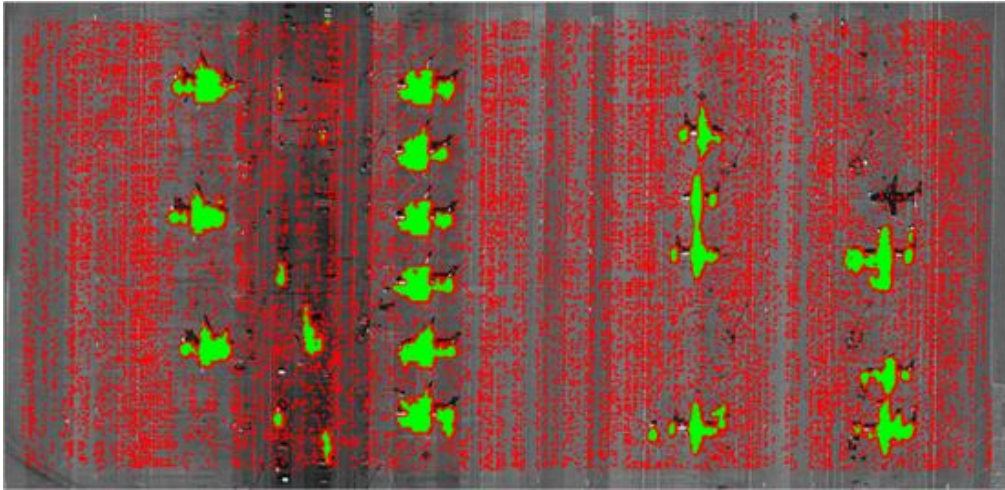
$$Y_o = [-1, 0, +1, -1, +1, -1, 0, +1]$$



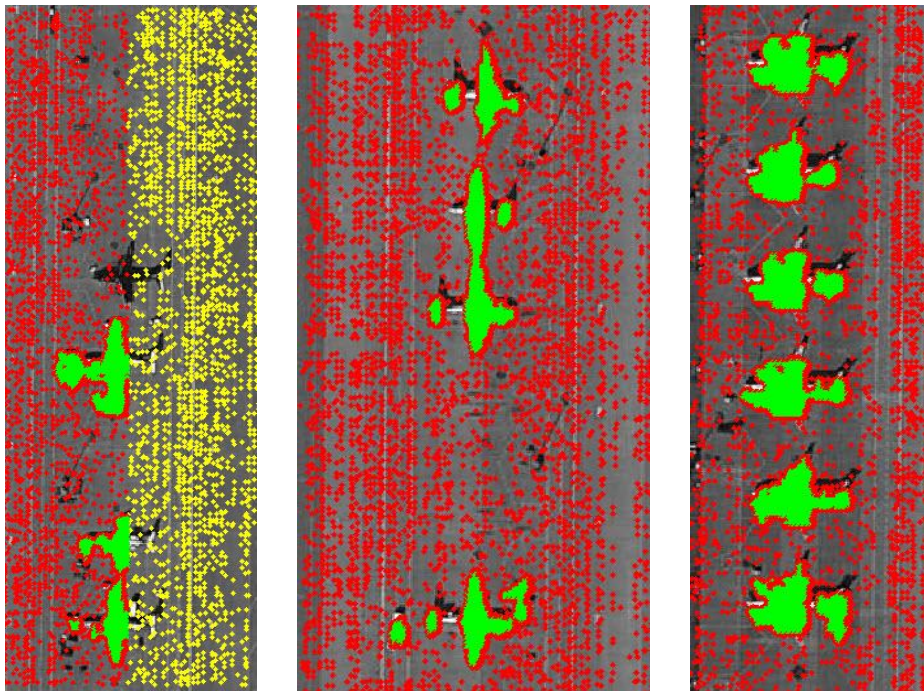
Σχήμα 21 - Αναπαράσταση όλων των γειτονικών pixels στα οποία λαμβάνει χώρα έλεγχος από το ΤΝΔ

```
xo = [-1,-1,-1, 0, 0,+1,+1,+1];
yo = [-1, 0,+1,-1,+1,-1, 0,+1];
```

Σε περίπτωση που το αποτέλεσμα δεν είναι «-1», δηλαδή όχι αεροσκάφος, μετατρέπεται άμεσα σε «1», δηλαδή σε αεροσκάφος. Στις παρακάτω εικόνες φαίνεται ένα ολοκληρωμένο αποτέλεσμα χρωματισμού της εικόνας δοκιμής μέσω της συνάρτησης imscan.



Εικόνα 31 - Τελική εικόνα αναζήτησης



Εικόνα 32 - Τελικές εικόνες αναζήτησης σε μεγέθυνση

```

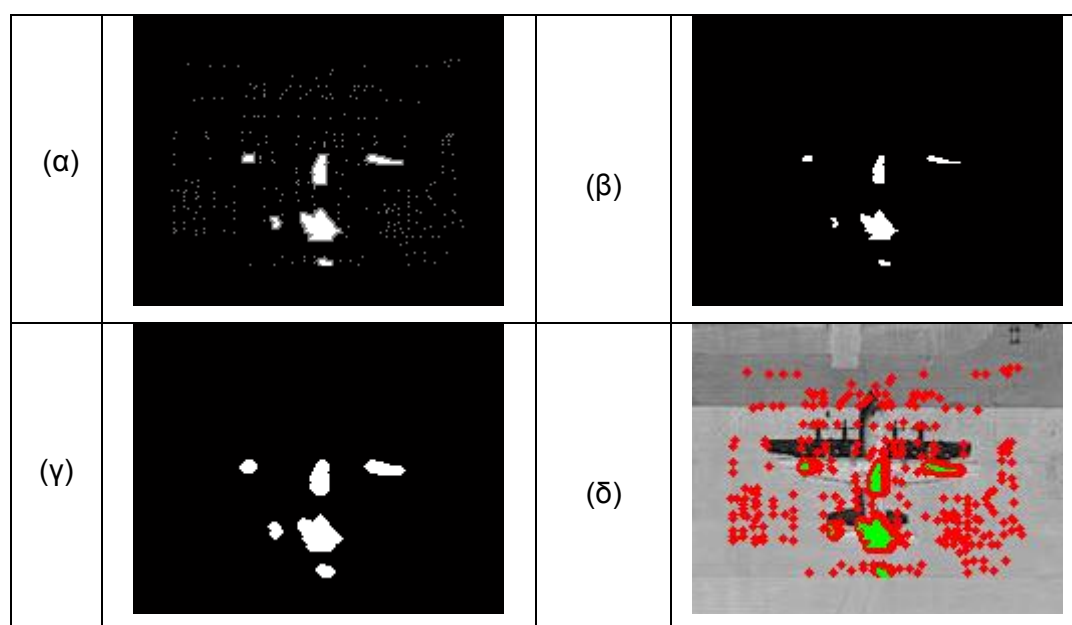
for k=1:8
    nx = x + xo(k);
    ny = y + yo(k);
    if (Cell.state(ny,nx) ~= -1)
        Cell.state(ny,nx) = 1;
    end
end

```

Τα αποτελέσματα, όπως είχαμε προαναφέρει παραπάνω, καταχωρούνται στον πίνακα Cell.net. Εάν εμφανίσουμε τα αποτελέσματα ως έχουν, θα έχουμε ένα οπτικό

αποτέλεσμα όπως η εικόνα (α). Πρέπει να κάνουμε μια επεξεργασία της εικόνας για αυτό και την κατωφλιώνουμε έτσι ώστε μόνο όσα pixels έχουν τιμές (προερχόμενες από το ΤΝΔ) μεγαλύτερες του μηδενός να εμφανιστούν με τιμή 1 και αντίστροφα. Με αυτό τον τρόπο διώχνουμε τα γκρι όρια της εικόνας (α) και παίρνουμε την εικόνα (β). Έπειτα, πραγματοποιούμε μια επαύξηση των δομών που έχουμε από την εικόνα (β) μέσω της εντολής “imdilate” σε συνδυασμό με την εντολή «strel», έτσι ώστε να μειώσουμε την πιθανότητα να εμφανίζονται πολλαπλοί εντοπισμοί του ίδιου αεροσκάφους. Παράδειγμα της παραπάνω αλλαγής φαίνεται στην εικόνα (γ).

```
xy_ = Cell.net > 0;
xy_ = imregionalmax(xy_);
xy_ = imdilate (xy_,strel('disk',2,4));
```



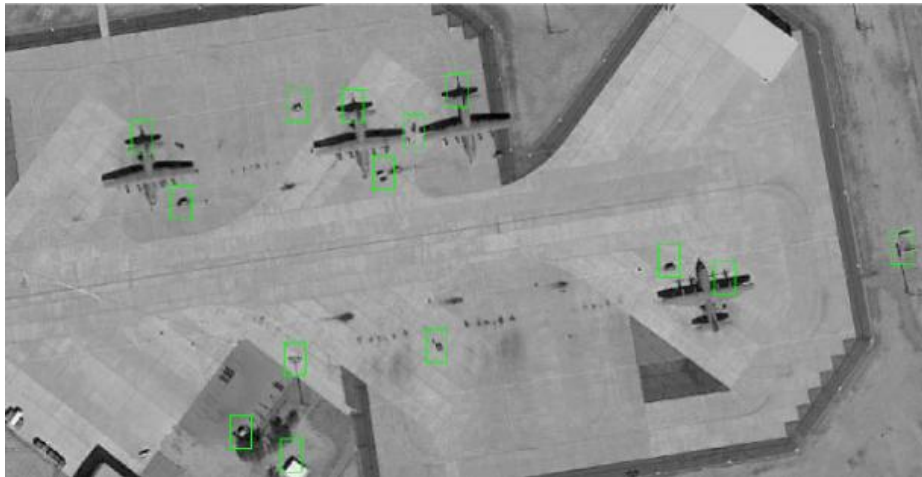
Πίνακας 12 – Στιγμιότυπα τελικών αποτελεσμάτων εντοπισμού

Ακολούθως χρησιμοποιούμε τις ίδιες εντολές όπως είχαμε πράξει και στο 1ο Κεφάλαιο για να ταυτοποιηθούν οι τελικές «τρύπες» της δυαδικής εικόνας με βάση συγκεκριμένα χαρακτηριστικά. Υπάρχει η δυνατότητα να καταγραφούν όλα τα διαθέσιμα από την Matlab χαρακτηριστικά. Στην συγκεκριμένη περίπτωση χρειαζόμαστε μόνο το “centroid” το οποίο μας δίνει το κέντρο μάζας του κάθε εμβადού. Αυτό το χρειαζόμαστε για να σχεδιάσουμε στην αρχική εικόνα ένα layer με τετράγωνα που να ορίζουν τα τελικά αντικείμενα που ανιχνεύτηκαν.

Πραγματοποιείται μια στρογγυλοποίηση των συντεταγμένων των κέντρων μάζας με την εντολή fix (για τον εντοπισμό του αντίστοιχου pixel) και μετά τροφοδοτείται η

συνάρτηση “drawrec.m” που έχουμε αναλύσει παραπάνω για να σχεδιάσει τα πλαίσια στις πιθανές θέσεις αεροσκαφών. Τα πλαίσια έχουν χρώμα κόκκινο με διαστάσεις 16*16 pixels.

```
[LabelMatrix,nLabel] = bwlabeln(xy_,4);
CentroidMatrix = regionprops(LabelMatrix,'centroid');
xy_ = zeros(m,n);
for i = 1:nLabel
    xy_(fix(CentroidMatrix(i).Centroid(2)),...
        fix(CentroidMatrix(i).Centroid(1))) = 1;
end
xy_ = drawrec(xy_,[grammes stiles]);
im_out (:,:,1) = im;im_out (:,:,2) = im;im_out (:,:,3) = im;
for i = 1:m
    for j=1:n
        if xy_(i,j)==1
            im_out (i,j,1)=255;im_out (i,j,2)=0;im_out (i,j,3)=0;
        end
    end
end
end
```



Εικόνα 33 - Τελική εικόνα εντοπισμού μετά των πλαισίων επισήμανσης

Συνάρτηση *Main.m*

Το τελευταίο script-κώδικα που θα αναλύσουμε είναι η *main* η οποία είναι και η κύρια συνάρτηση του προγράμματος. Μέσω αυτής υλοποιείται η σωστή αλληλουχία στην εκτέλεση όλων των παραπάνω συναρτήσεων που αναλύσαμε.

Αρχικά, φορτώνονται για οικονομία χρόνου όλοι οι απαραίτητοι πίνακες εκτός και αν δεν υπάρχουν που σημαίνει ότι κατασκευάζονται.

1. *Gabor.mat*
2. *Imgdb.mat*
3. *Net.mat*

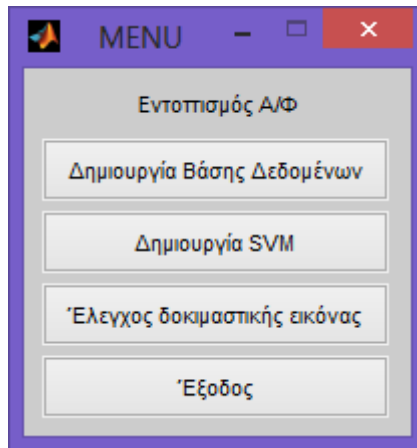
Ακολουθως, εμφανίζονται όλα τα βασικά βήματα (4 επιλογές) σε παραθυρική διάταξη για είναι πιο φιλική στον χρήστη. Η κάθε επιλογή ενεργοποιεί και την αντίστοιχη συνάρτηση. Σε περίπτωση που υπάρχουν οι απαραίτητοι πίνακες απλά ο χρήστης εξετάζει απευθείας δοκιμαστική εικόνα.

```
if ~exist('gabor.mat','file')
    fprintf ('Δημιουργία φίλτρων Gabor ...');
    create_gabor;
end

if exist('imgdb.mat','file')
    load imgdb;
else
    IMGDB = loadimages;
end

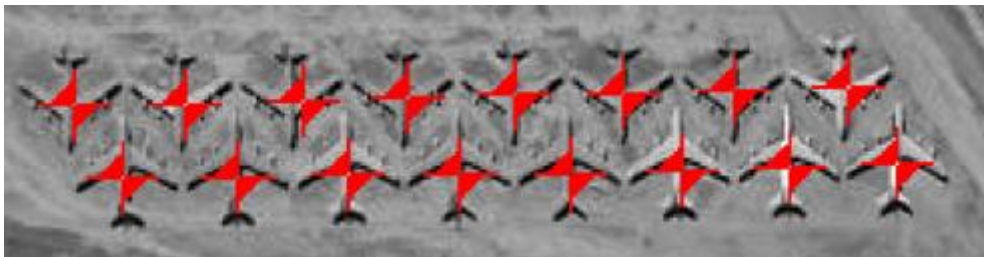
if exist('net.mat','file')
    load net;
end

while (1==1)
    choice=menu('Εντοπισμός A/Φ',...
               'Δημιουργία Βάσης Δεδομένων',...
               'Δημιουργία SVM',...
               'Έλεγχος δοκιμαστικής εικόνας',...
               'Έξοδος');
    if (choice == 1)
        IMGDB = loadimages;
    end
    if (choice == 2)
        net = trainnet(IMGDB);
    end
end
```



Εικόνα 34 - Στιγμιότυπο του main script

Στην 3η περίπτωση έχουμε την εξέταση της δοκιμαστικής εικόνας μέσω αναζήτησης από τον Microsoft Explorer. Ακολούθως γίνεται μετατροπή της σε grayscale και εισάγεται σαν όρισμα στην συνάρτηση «imscan» που μόλις αναλύσαμε. Η τελική εικόνα παρουσιάζεται σε μεγέθυνση.



Εικόνα 35 - Στιγμιότυπο τελικού εντοπισμού

Παρακάτω έχει σχεδιαστεί ένας πίνακας που παρουσιάζει τα ορίσματα και τις σχέσεις μεταξύ των διαφόρων συναρτήσεων του προγράμματος:

Φορά ανάγνωσης	Main	Drawrec	Gabor	Create gabor	Im2vec	Loadimages	Mminmax	Trainnet	Imscan
Main	-	-	-	Χρήση	-	Χρήση	-	Χρήση	Χρήση
Drawrec	-	-	-	-	-	-	-	-	-
Gabor	-	-	-	-	-	-	-	-	-
Create_gabor	-	-	Χρήση	-	-	-	-	-	-
Im2vec	-	-	-	Χρήση	-	-	-	-	-
Loadimages	-	-	-	-	Χρήση	-	-	-	-
Mminmax	-	-	-	-	-	-	-	-	-
Trainnet	-	-	-	-	-	Όρισμα	-	Εξαγωγή	-
Imscan	-	Χρήση	-	-	Χρήση	-	Χρήση	Όρισμα	-

Πίνακας 13 - Πίνακας αναπαράστασης σχέσεων των συναρτήσεων του προγράμματος

3.3. Script Αρχεία

Τα script αρχεία που χρησιμοποιήθηκαν για κάθε μεθοδολογία εμπεριέχονται στο επισυναπτόμενο CD, που συνοδεύει το τεύχος της Διπλωματικής:

1. Χρήση τεχνικών αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας

- Afairesi_Katofliosi.m
- Detection_bw.m
- Dilation.m
- Diploma4.m
- inverse_problem.m
- progressBar.m
- proxeiroV3.m
- Radiocorr.m
- REDtoBW.m
- Test_logical.m
- Vislabels.m

2. Χρήση τεχνικών αναγνώρισης και διανυσμάτων υποστήριξης

- drawrec
- create_gabor
- gabor
- im2vec
- imscan
- loadimages
- trainnet
- mminmax
- main

4. Πειραματικά αποτελέσματα, Ποσοτική και Ποιοτική Αξιολόγηση

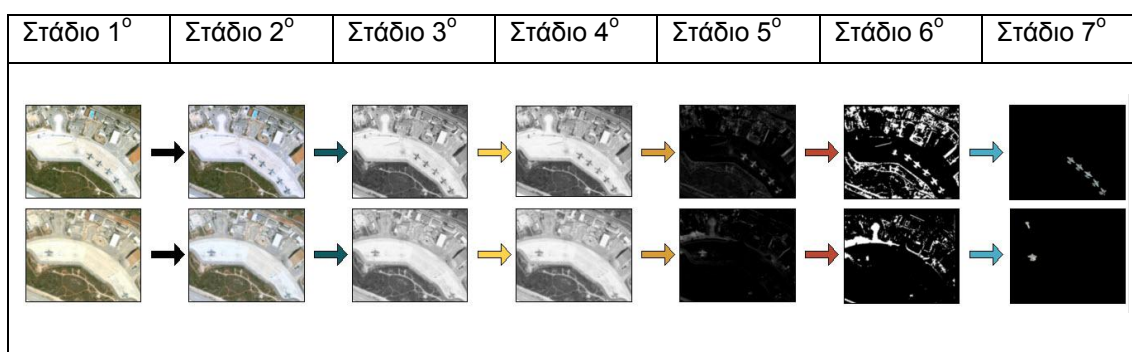
Στο κεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα και τα συμπεράσματα και των δύο προαναφερθεισών μεθοδολογιών, με βάση καθιερωμένους στην διεθνή βιβλιογραφία ποσοτικούς δείκτες.

4.1. Αξιολόγηση με βάση τεχνικές αντιστοίχισης, κατωφλίωσης και μαθηματικής μορφολογίας.

Ο εντοπισμός των αεροσκαφών πραγματοποιήθηκε με βάση μια σειρά βημάτων προ-επεξεργασίας ενός ζεύγους ψηφιακών εικόνων που αποκόπηκαν από το Google Earth (Αναλυτικά η μεθοδολογία σελίδα: 24) Συνοπτικά η διαδικασία περιλαμβάνει τα ακόλουθα βήματα:

A/A	Στάδιο Επεξεργασίας
1.	Αντιστοίχιση εικόνων (Registration)
2.	Αποκατάσταση και Ραδιομετρική Διόρθωση
3.	Μετατροπή σε grayscale
4.	Ομαλοποίηση εικόνων (Gaussian Filter)
5.	Αφαίρεση
6.	Κατωφλίωση
7.	Δημιουργία κατωφλίων για την εξαγωγή των επιθυμητών στόχων

Πίνακας 14- Βήματα Επεξεργασίας ζεύγους εικόνων για την 1^η Μεθοδολογία



Πίνακας 15- Αποτελέσματα των βημάτων επεξεργασίας ζεύγους εικόνων για την 1^η Μεθοδολογία

Στις επόμενες σελίδες παρουσιάζονται αναλυτικά τα αποτελέσματα από τα αεροδρόμια της Σούδας (Greece) και του Holladay (USA) και για τα επτά ανωτέρω στάδια. Τονίζεται ότι το 7^ο στάδιο εμπεριέχει δύο ξεχωριστά βήματα κατωφλίωσης για την επίτευξη του τελικού εντοπισμού της εν λόγω μεθοδολογίας.



Αρχική εικόνα αποκοπής **Σούδα**
4-9-2005 (Εικόνα A – RGB 8bit -)



Αρχική εικόνα αποκοπής **Σούδα**
5-27-2012 (Εικόνα B – RGB 8bit -)



Registered, Clipped, Resized **Σούδα**
(Εικόνα A)



Registered, Clipped, Resized **Σούδα**
(Εικόνα B)



Ραδιομετρικά Διορθωμένη **Σούδα**
(Εικόνα A)



Ραδιομετρικά Διορθωμένη **Σούδα**
(Εικόνα B)



Μετατροπή σε grayscale **Σούδα**
(Εικόνα A)



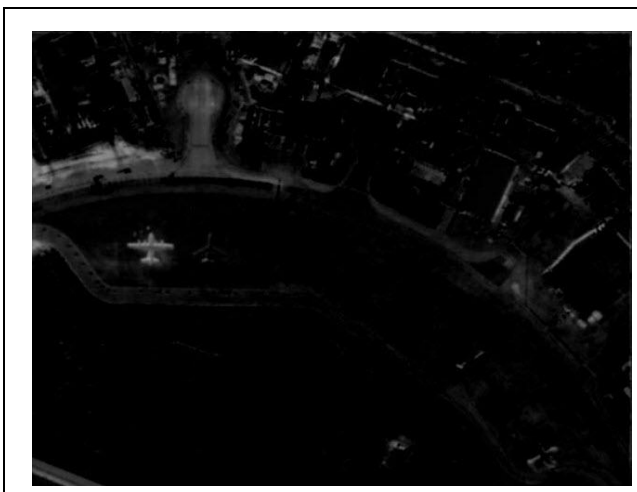
Μετατροπή σε grayscale **Σούδα**
(Εικόνα B)



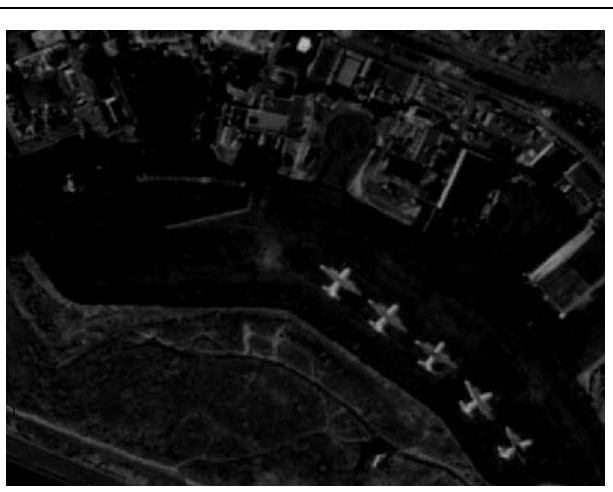
Gaussian Filter $5 \times 5 \sigma=1.0$ **Σούδα**
(Εικόνα A)



Gaussian Filter $5 \times 5 \sigma=1.0$ **Σούδα**
(Εικόνα B)



Αφαίρεση εικόνων **Σούδα** (Εικόνα B - Εικόνα A)



Αφαίρεση εικόνων **Σούδα** (Εικόνα A - Εικόνα B)



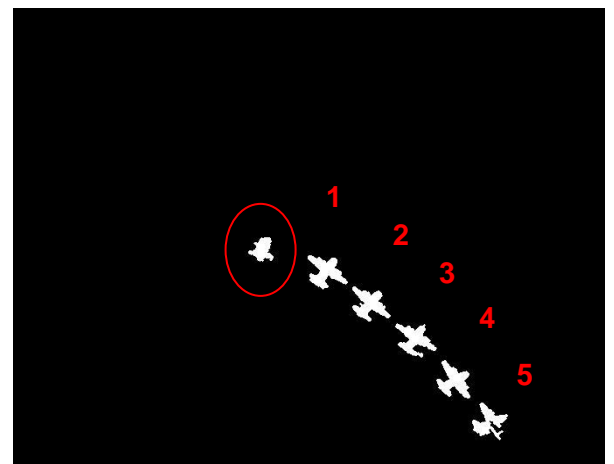
Κατωφλίωση Σούδα (Εικόνα Β - Εικόνα Α)



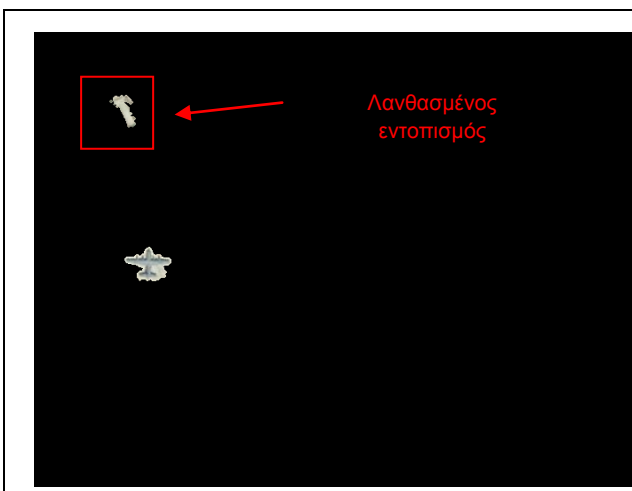
Κατωφλίωση Σούδα (Εικόνα Α - Εικόνα Β)



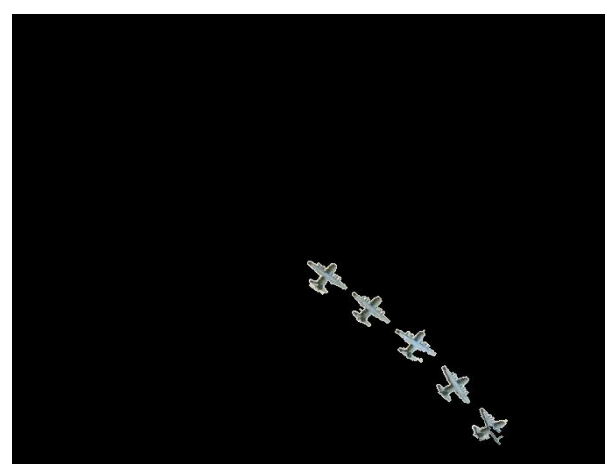
Δημιουργία κατωφλίων για την εξαγωγή των επιθυμητών στόχων, με βάση γεωμετρικά χαρακτηριστικά των οντοτήτων
Σούδα Εικόνα Β & Εικόνα Α



Δημιουργία κατωφλίων για την εξαγωγή των επιθυμητών στόχων, με βάση γεωμετρικά χαρακτηριστικά των οντοτήτων
Σούδα Εικόνα Α & Εικόνα Β



Κατωφλίωση με βάση φασματικά χαρακτηριστικά των οντοτήτων
Σούδα Εικόνα Β & Εικόνα Α



Κατωφλίωση με βάση φασματικά χαρακτηριστικά των οντοτήτων
Σούδα Εικόνα Α & Εικόνα Β



Αρχική εικόνα αποκοπής **Holladay**
4-28-2007 (Εικόνα Α – RGB 8bit -)



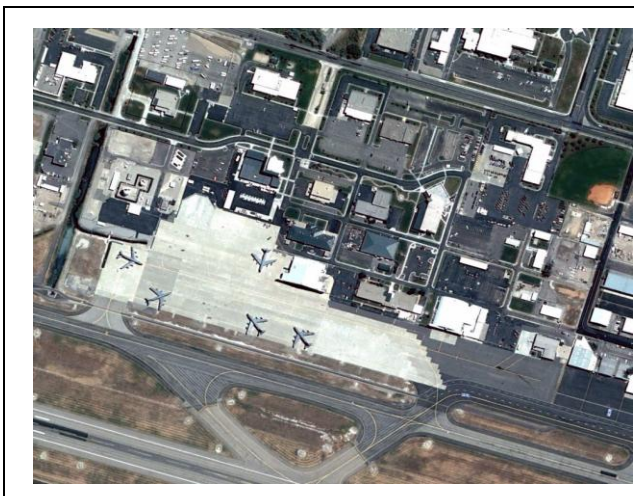
Αρχική εικόνα αποκοπής **Holladay**
6-24-2005 (Εικόνα Β – RGB 8bit -)



Registered, Clipped, Resized **Holladay**
(Εικόνα Α)



Registered, Clipped, Resized **Holladay**
(Εικόνα Β)



Ραδιομετρικά Διορθωμένη **Holladay**
(Εικόνα Α)



Ραδιομετρικά Διορθωμένη **Holladay**
(Εικόνα Β)



Μετατροπή σε grayscale **Holladay**
(Εικόνα Α)



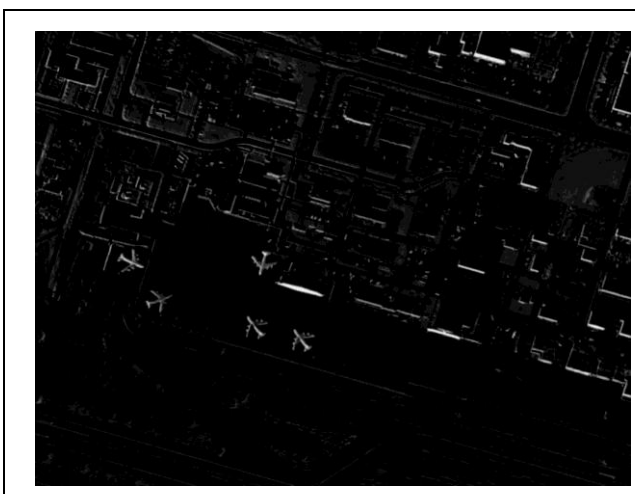
Μετατροπή σε grayscale **Holladay**
(Εικόνα Β)



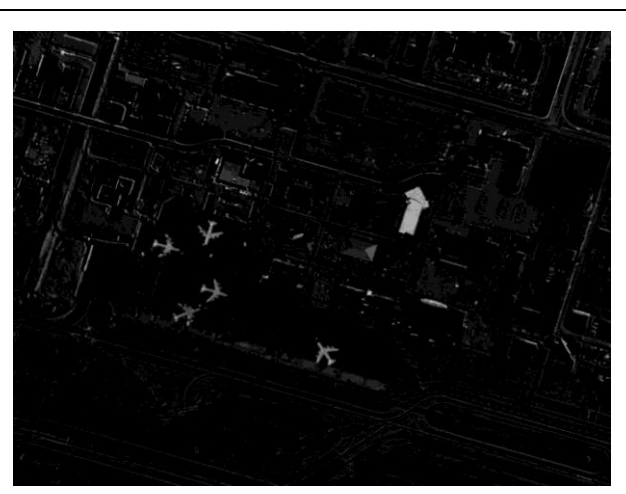
Gaussian Filter 5×5 $\sigma=1.0$ **Holladay**
(Εικόνα Α)



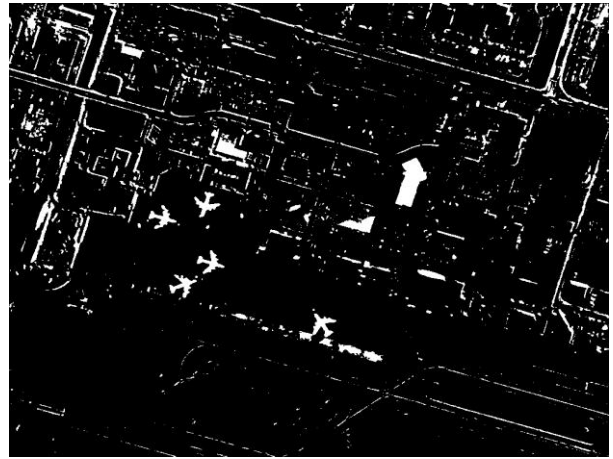
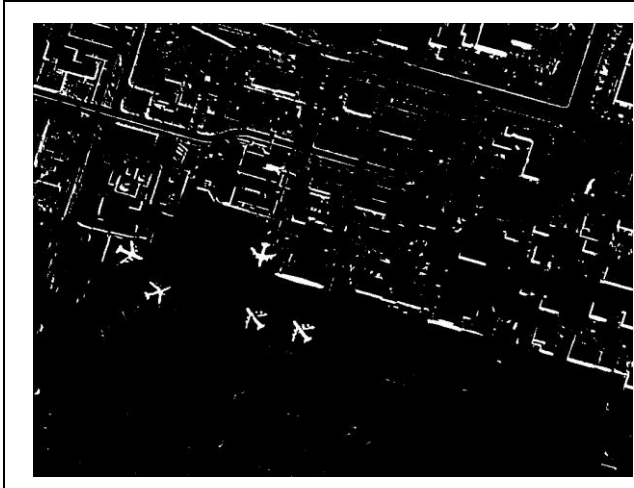
Gaussian Filter 5×5 $\sigma=1.0$ **Holladay**
(Εικόνα Β)



Αφαίρεση εικόνων **Holladay** (Εικόνα Β - Εικόνα Α)

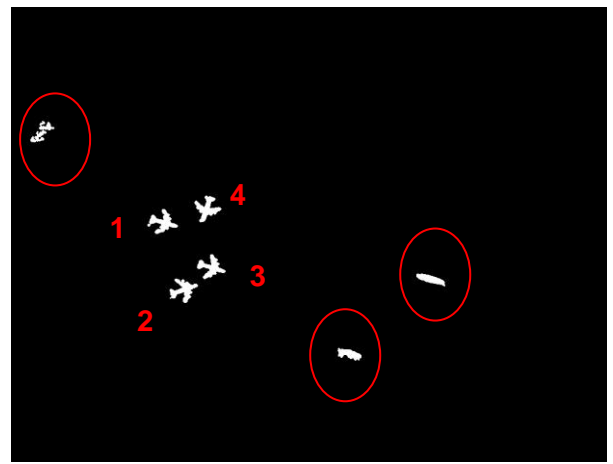


Αφαίρεση εικόνων **Holladay** (Εικόνα Α - Εικόνα Β)



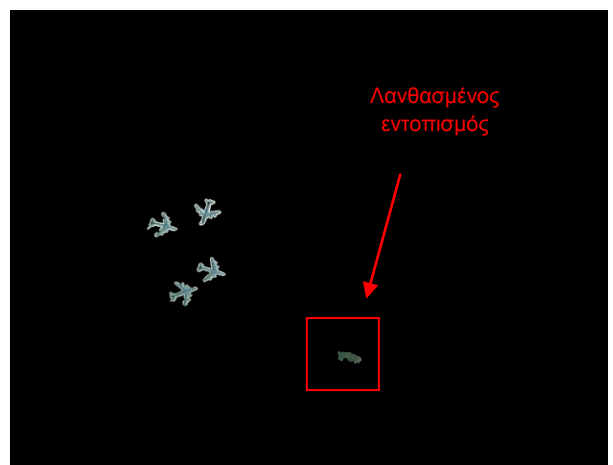
Κατωφλίωση **Holladay** (Εικόνα Β - Εικόνα Α)

Κατωφλίωση **Holladay** (Εικόνα Α - Εικόνα Β)



Δημιουργία κατωφλίων για την εξαγωγή των επιθυμητών στόχων, με βάση γεωμετρικά χαρακτηριστικά των οντοτήτων
Holladay Εικόνα Β & Εικόνα Α

Δημιουργία κατωφλίων για την εξαγωγή των επιθυμητών στόχων, με βάση γεωμετρικά χαρακτηριστικά των οντοτήτων
Holladay Εικόνα Α & Εικόνα Β



Κατωφλίωση με βάση φασματικά χαρακτηριστικά των οντοτήτων
Holladay Εικόνα Β & Εικόνα Α

Κατωφλίωση με βάση φασματικά χαρακτηριστικά των οντοτήτων **Holladay** Εικόνα Α & Εικόνα Β

Οι ποσοτικοί δείκτες με τους οποίους έγινε η αξιολόγηση των αποτελεσμάτων είναι οι κάτωθι:

A/A	Δείκτες Ποιότητας Εντοπισμού	Τρόπος Υπολογισμού
1	Δείκτης Πληρότητας (Completeness)	$= \frac{\text{Αριθμός εντοπισμένων αεροσκαφών}}{\text{Συνολικός Αριθμός Αεροσκαφών Εικόνας}}$
2	Δείκτης Ορθότητας (Correctness)	$= \frac{\text{Αριθμός εντοπισμένων αεροσκαφών}}{\text{Συνολικός αριθμός εντοπισμένων αντικειμένων εικόνας}}$

Πίνακας 16- Δείκτες Ποιότητας Εντοπισμού

Τα αποτελέσματα και για τις δύο περιπτώσεις αεροδρομίων (Σούδα & Holladay) φαίνονται στον πίνακα 4.3. Παρατηρούμε ότι το αεροδρόμιο Holladay παρουσιάζει καλύτερα αποτελέσματα από το αεροδρόμιο της Σούδας. Αυτό οφείλεται κυρίως στην καλύτερη αρχική ανάλυση των εικόνων αλλά και στην ποιότητα του registration. Τέλος, θα πρέπει να τονίσουμε ότι το αεροδρόμιο της Σούδας μπορεί να εμφανίζει μικρή ορθότητα (25%) στην εικόνα A, αλλά αν παρατηρήσουμε την ποιότητα της εικόνας θα δούμε ότι είναι αρκετά κακή, ενώ δύο από τα τρία αεροσκάφη που δεν εντοπίστηκαν είχαν χρώμα άσπρο (κάτι που καθιστούσε τον εντοπισμό ακόμα πιο δύσκολο).

Δείκτης Ποιότητας		Πληρότητα	Ορθότητα
Σούδα	Εικόνα A	25% (σελίδα 116)	50% (σελίδα 116)
	Εικόνα B	100% (σελίδα 116)	100% (σελίδα 116)
Holladay	Εικόνα A	100% (σελίδα 119)	100% (σελίδα 119)
	Εικόνα B	67% (σελίδα 119)	80% (σελίδα 119)

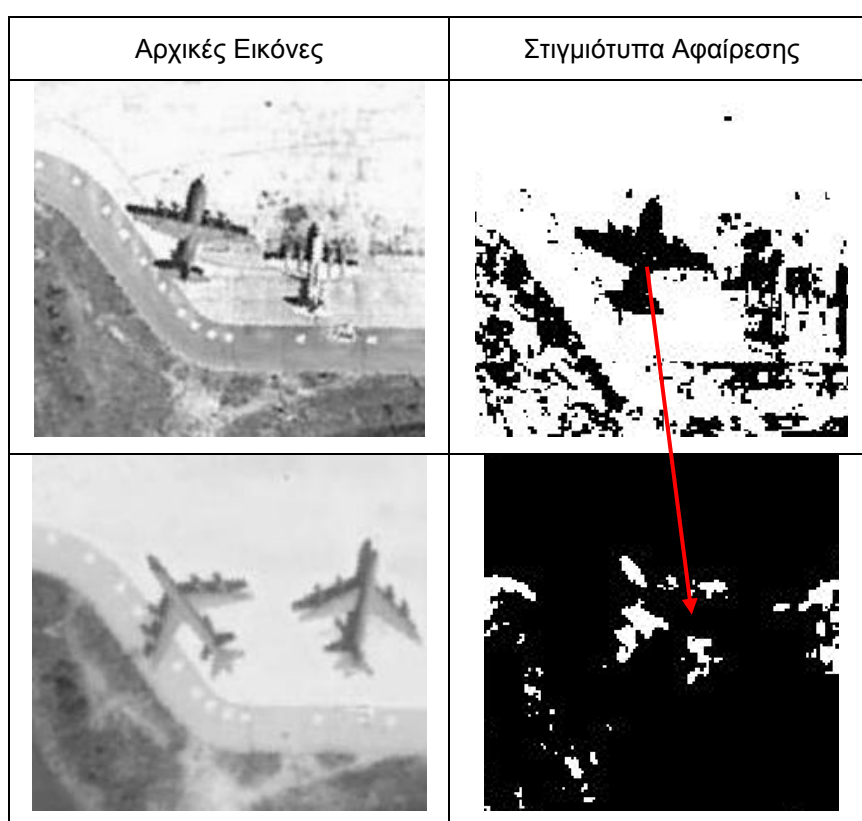
Πίνακας 17- Αποτελέσματα Δεικτών Ποιότητας Εντοπισμού για την 1^η Μεθοδολογία

Παρατηρήσεις

Ενδεικτικά παραδείγματα από τις τελικές binary εικόνες στις οποίες πραγματοποιήθηκε η αναζήτηση, φαίνονται στο Στάδιο 6^ο και των δύο ανωτέρω αεροδρομίων. Παρατηρούμε ότι είναι πολύ δύσκολο να εντοπιστούν οι επιθυμητές οντότητες, για αυτό και χρησιμοποιήθηκαν αρκετά γεωμετρικά χαρακτηριστικά στον

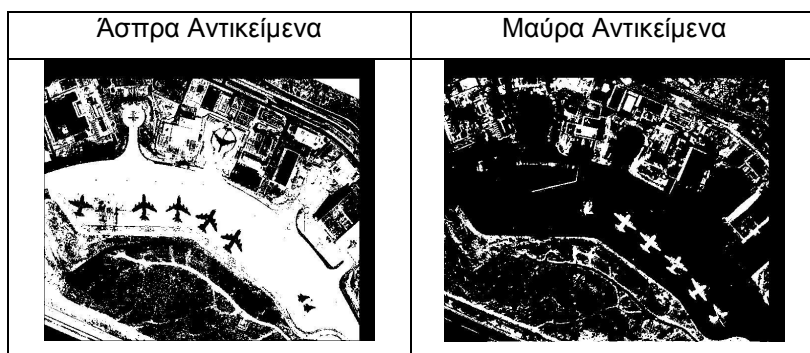
αντίστοιχο τύπο του κάθε αεροσκάφους. Είναι πλέον εμφανές ότι το στάδιο του registration είναι κομβικής σημασίας για την καθαρή εξαγωγή των σχημάτων των αεροσκαφών. Εξού και το γεγονός, ότι τα σχήματα είναι, κατά κάποια pixels, μεγαλύτερα σε σχέση με τα αντίστοιχα των “ground-truth” εικόνων.

Περιπτώσεις στις οποίες τα σταθμευμένα αεροσκάφη βρίσκοντουσαν σε παραπλήσια θέση αλλά με διαφορετικό προσανατολισμό, δημιουργεί αδυναμία εντοπισμού στο συγκεκριμένο ζευγάρι εικόνων. Ένα χαρακτηριστικό παράδειγμα φαίνεται στο Σχήμα 4.2., για το πώς η τελική εικόνα είναι αλλοιωμένη και ο μοναδικός τρόπος εντοπισμού των συγκεκριμένων αεροσκαφών είναι από άλλο συνδυασμό ψηφιακών εικόνων.



Πίνακας 18- Παράδειγμα προβληματικού εντοπισμού λόγω στάθμευσης των αεροσκαφών στην ίδια θέση με διαφορετικό προσανατολισμό

Μια άλλη ιδιαιτερότητα στον εντοπισμό των οντοτήτων είναι οι περιπτώσεις στις οποίες εμφανίζονται οι τρύπες αντεστραμμένες, δηλαδή με μαύρο τα αντικείμενα και με άσπρο ο υπόλοιπος περιβάλλοντας χώρος (Πίνακας 19). Στην συγκεκριμένη περίπτωση δημιουργήθηκε ένα 2^ο script το οποίο τροφοδοτούταν μόνο με αντεστραμμένες εικόνες.



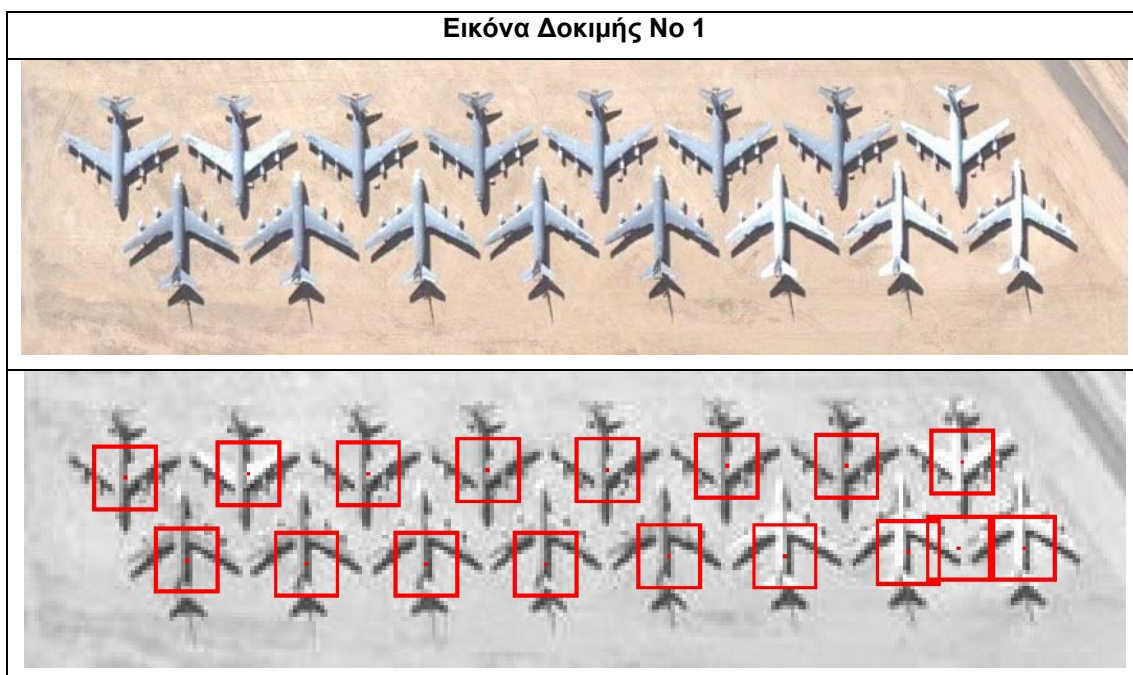
Πίνακας 19 - Αντίστοιχα παραδείγματα λευκών και μαύρων αεροσκαφών

Όπως έχουμε προαναφέρει, ο εντοπισμός έγινε αρχικά με καθορισμό φίλτρων για τα διάφορα γεωμετρικά χαρακτηριστικά που χρησιμοποιήθηκαν. Περιπτώσεις στις οποίες το αεροσκάφος εμφανιζόταν μισό ή ένα κομμάτι αυτού, καθιστούσε αδύνατο τον εντοπισμό. Η προσαρμογή των φίλτρων και για αυτές τις περιπτώσεις δεν προτιμήθηκε γιατί εισήγαγε σημαντικό αριθμό «σκουπιδιών» που τύχαινε να πληρούν τα αντίστοιχα κριτήρια. (χαρακτηριστικό παράδειγμα η περίπτωση του A/Δ της Σούδας.). Τα αποτελέσματα του εντοπισμού, όπως φαίνονται και παραπάνω, με την χρήση της αρχικής φασματικής πληροφορίας βελτιώθηκαν αρκετά.

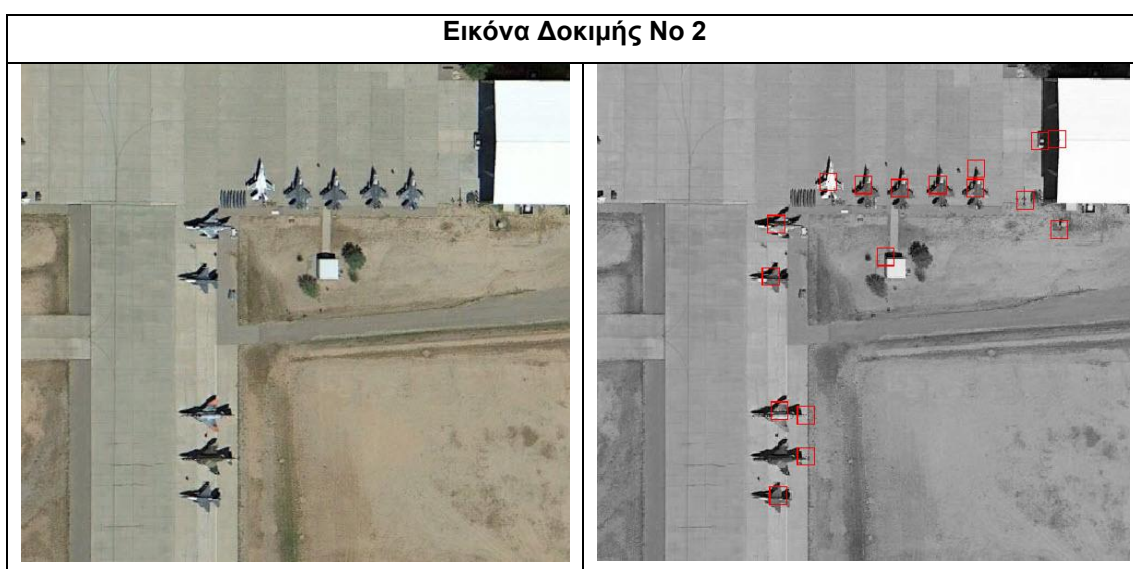
Τα αποτελέσματα στο αεροδρόμιο Holladay είναι αρκετά καλύτερα από το αεροδρόμιο της Σούδας και οφείλεται στην καλύτερη ανάλυση των αρχικών εικόνων αλλά και στο γεγονός ότι τα εύρη προσαρμόστηκαν εξαρχής στα χαρακτηριστικά των αεροσκαφών του A/Δ Holladay και ακολούθως εφαρμόστηκαν και στο A/Δ της Σούδας.

4.2. Αξιολόγηση με βάση τεχνικές αναγνώρισης και διανύσματα υποστήριξης.

Ο εντοπισμός των αεροσκαφών πραγματοποιήθηκε με βάση την μεθοδολογία που αναλύθηκε στο Κεφάλαιο 3°. Παρακάτω παρατίθενται μια σειρά αποτελεσμάτων και δοκιμών που πραγματοποιήθηκαν στα πλαίσια της αξιολόγησης του προγράμματος ανίχνευσης.



Εικόνα 36- Αποτελέσματα Εντοπισμού στην εικόνα Νο1 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 94,1%



Εικόνα 37- Αποτελέσματα Εντοπισμού στην εικόνα Νο2 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 58,82%

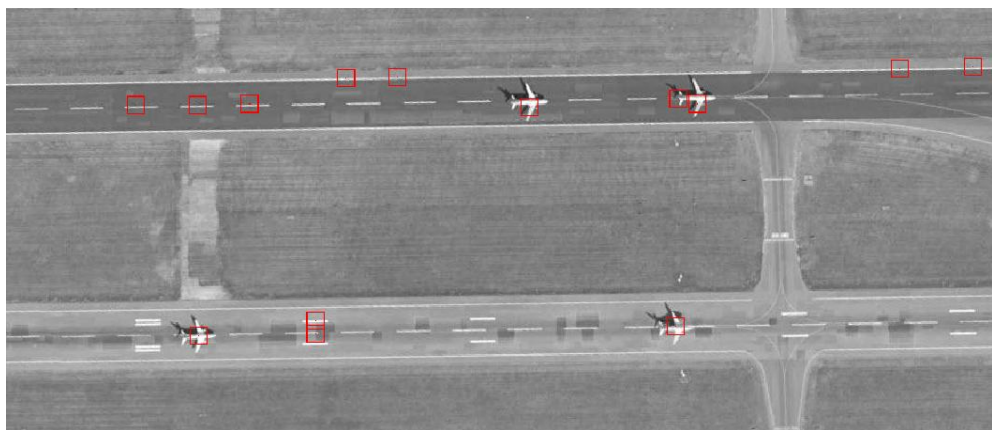
Εικόνα Δοκιμής Νο 3



Ανίχνευση με χρήση του πρότυπου A

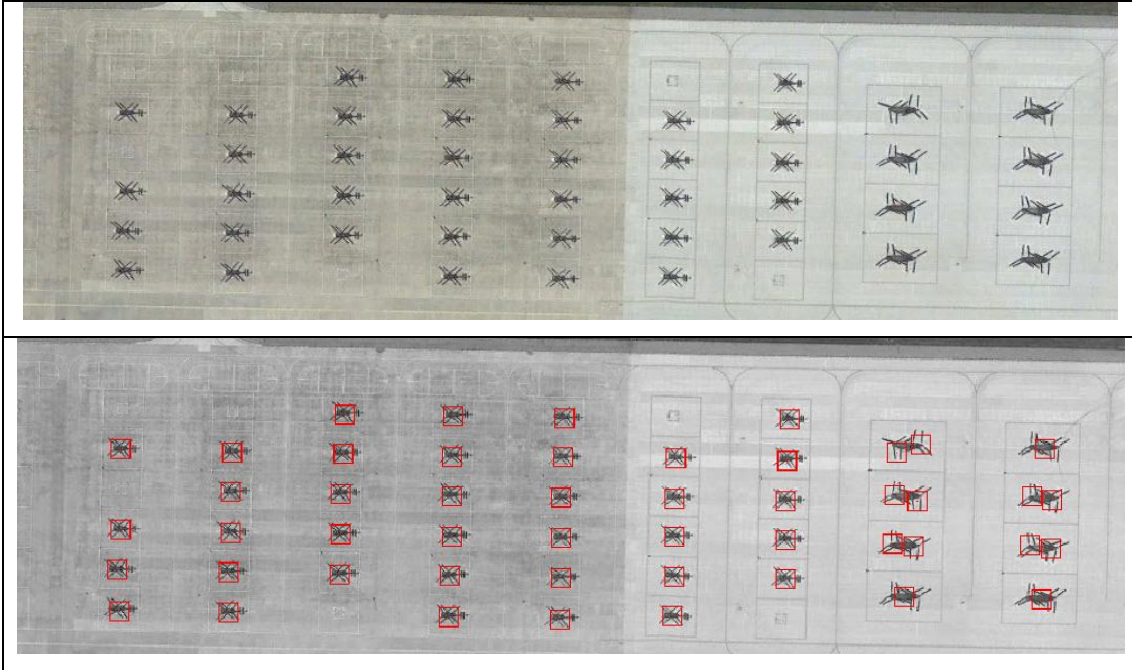


Ανίχνευση με χρήση του πρότυπου B



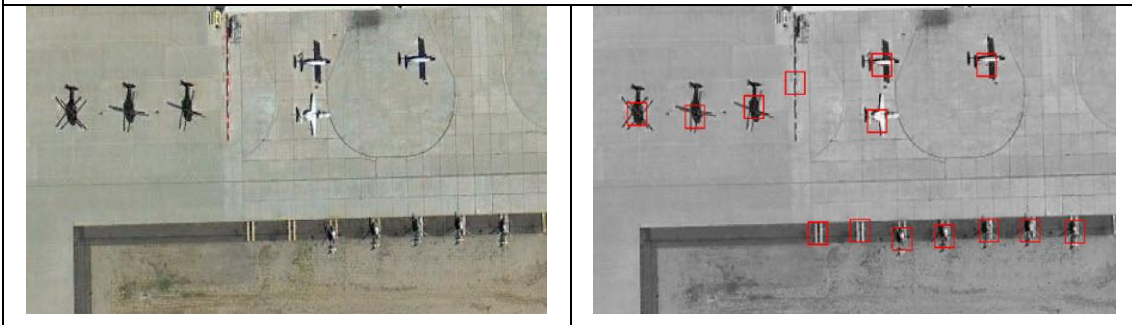
Εικόνα 38 - Αποτελέσματα Εντοπισμού στην εικόνα Νο3 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 30,77%

Εικόνα Δοκιμής Νο 4



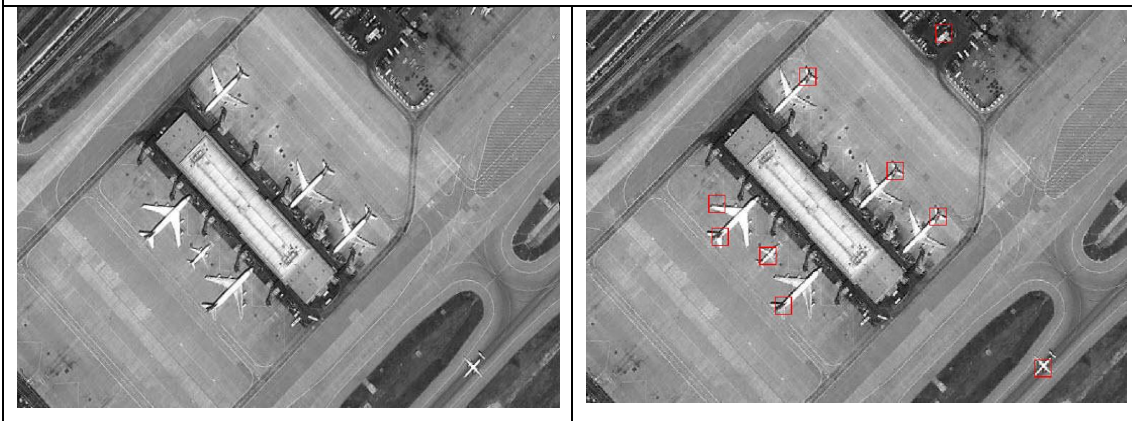
Εικόνα 39- Αποτελέσματα Εντοπισμού στην εικόνα Νο4 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 91,66%

Εικόνα Δοκιμής Νο 5



Εικόνα 40- Αποτελέσματα Εντοπισμού στην εικόνα Νο5 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 78,57%

Εικόνα Δοκιμής Νο 6

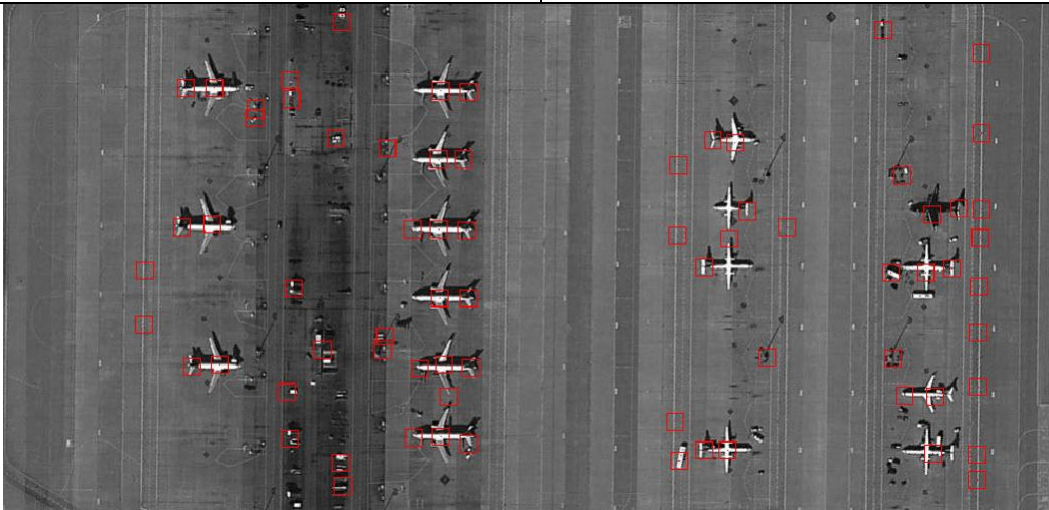


Εικόνα 41- Αποτελέσματα Εντοπισμού στην εικόνα Νο6 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 77,78%

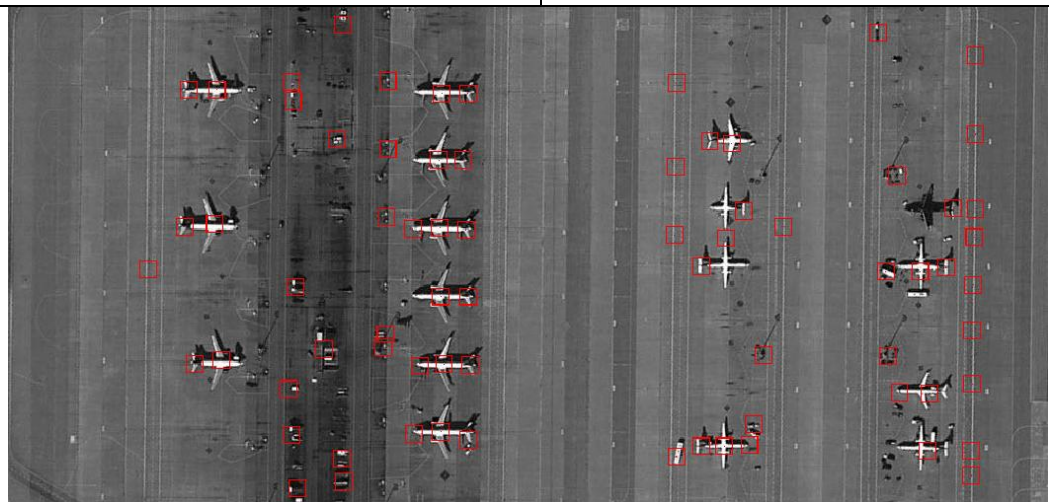
Εικόνα Δοκιμής Νο 7



Ανίχνευση με χρήση του πρότυπου A



Ανίχνευση με χρήση του πρότυπου B



Εικόνα 42- Αποτελέσματα Εντοπισμού στην εικόνα Νο7 (Tucson-USA, Aircraft Boneyard) Πληρότητα εντοπισμού: 100%, Ορθότητα εντοπισμού: 23,94% (A) - 30,91% (B)

Τα αποτελέσματα για όλες τις παραπάνω περιπτώσεις του αεροδρομίου Tucson (USA) παρουσιάζονται στον πίνακα 4.4. Το εν λόγω αεροδρόμιο αποτελεί «νεκροταφείο» πολεμικών αεροσκαφών των ΗΠΑ και διαθέτει όλους τους τύπους μαχητικών και μεταγωγικών αεροσκαφών που χρησιμοποιούνται σήμερα από την USAF (United States Air Force). Αρχικά, με μια γρήγορη οπτική αξιολόγηση των εντοπισμών, μπορούμε να παρατηρούμε ότι τα αποτελέσματα του αλγορίθμου είναι αρκετά ικανοποιητικά και προφανώς πολύ καλύτερα από τα αντίστοιχα αποτελέσματα της 1^{ης} μεθοδολογίας.

Στην εικόνα δοκιμής 3 και 7, παρουσιάζονται αποτελέσματα για δύο διαφορετικά πρότυπα εικόνων. Τονίζεται ότι οι δείκτες αξιολόγησης έχουν υπολογιστεί αρχικά μη λαμβάνοντας υπόψη το φαινόμενο του πολλαπλού εντοπισμού, ενώ στην συνέχεια (όσες γραμμές έχουν αστερίσκο) υπολογίστηκαν με την σύμβαση ότι οι περιπτώσεις πολλαπλού εντοπισμού θα θεωρούνται ως λανθασμένοι εντοπισμοί.

Αποτελέσματα Εντοπισμού							
Εικόνες Δοκιμής	A/Φ	Helicopter	Πολλαπλοί εντοπισμοί A/Φ-Ελικ.	Λανθασμένοι Πολλαπλοί εντοπισμοί	Λανθασμένοι εντοπισμοί	Πληρότητα	Ορθότητα
1 ^η	16	-	1	0	0	100%	100%
1 ^{η*}			0	0	1	100%	94,11%
2 ^η	10	-	2	1	4	100%	71,43%
2 ^{η*}			0	0	7	100%	58,82%
3 ^η –Πρότυπο A	4	-	3	0	6	100%	40,00%
3 ^η –Πρότυπο A*			0	0	9	100%	30,77%
3 ^η –Πρότυπο B	4	-	1	1	7	100%	36,36%
3 ^η –Πρότυπο B*			0	0	9	100%	30,77%
4 ^η	-	44	4	0	0	100%	100%
4 ^{η*}			0	0	4	100%	91,66%
5 ^η	3	8	0	0	3	100%	78,57%
5 ^{η*}			0	0	3	100%	78,57%
6 ^η	7	-	1	0	1	100%	87,50%
6 ^{η*}			0	0	2	100%	77,78%
7 ^η –Πρότυπο A	17	-	21	2	31	100%	35,42%
7 ^η –Πρότυπο A*			0	0	54	100%	23,94%
7 ^η –Πρότυπο B	17	-	18	1	36	100%	47,22%
7 ^η –Πρότυπο B*			0	0	55	100%	30,91%
					M.O=	100%	66,28%
						100%	57,48%

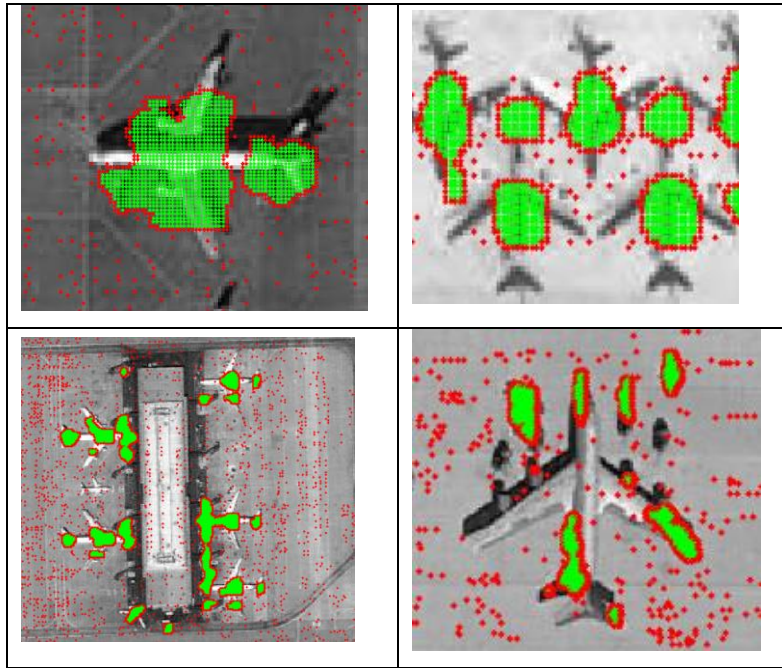
Πίνακας 20- Αποτελέσματα Δεικτών Ποιότητας Εντοπισμού για την 2^η Μεθοδολογία (Tucson-USA, Aircraft Boneyard)

Όπως βλέπουμε από τα αποτελέσματα στον πίνακα 4.4, σε όλες τις εικόνες δοκιμής επιτυγχάνεται ποσοστό πληρότητας 100%. Αυτό σημαίνει ότι όσα αεροσκάφη υπάρχουν στις δοκιμαστικές εικόνες, εντοπίστηκαν χωρίς την παραμικρή απώλεια. Η ποιότητα όμως του αλγορίθμου αξιολογείται ουσιαστικά με το κατά πόσο ορθοί είναι όλοι οι εντοπισμοί του. Για παράδειγμα, μπορεί να βρήκε όλα τα αεροσκάφη αλλά ο αριθμός των λανθασμένων εντοπισμών να είναι πολύ μεγαλύτερος από τους σωστούς εντοπισμούς. Έτσι, λοιπόν, βλέπουμε αντίστοιχα ότι το ποσοστό ορθότητας κυμαίνεται από 23% μέχρι και 100%. Παρατηρούμε ότι το πρόβλημα των πολλαπλών εντοπισμών μειώνει αρκετά τα ποσοστά ορθότητας των εντοπισμών και αποτελεί πεδίο για μελλοντικές βελτιώσεις του αλγορίθμου. Είναι ενδιαφέρον το γεγονός ότι ο εντοπισμός των ελικοπτέρων παρουσιάζει ιδιαίτερα καλά αποτελέσματα. Τέλος, θα πρέπει να αναφέρουμε ότι τα αποτελέσματα παρότι σε ορισμένες περιπτώσεις είναι αρκετά ικανοποιητικά οι M.O ανά κατηγορία δεικτών δεν ξεπερνάνε, στην ευνοϊκή περίπτωση, το 66,28% ενώ, στην ρεαλιστικότερη υπόθεση υπολογισμού (δηλαδή συμπερίληψης και των πολλαπλών εντοπισμών ως λανθασμένων) πέφτουν στο επίπεδο του 57.48%. Τα ποσοστά προφανώς και είναι αρκετά μακριά από το φράγμα του 75%, το οποίο αποτελεί την ελάχιστη δυνατή απαίτηση από έναν αλγόριθμο εντοπισμού αντικειμένων που θεωρείται αξιόπιστος.

Παρατηρήσεις

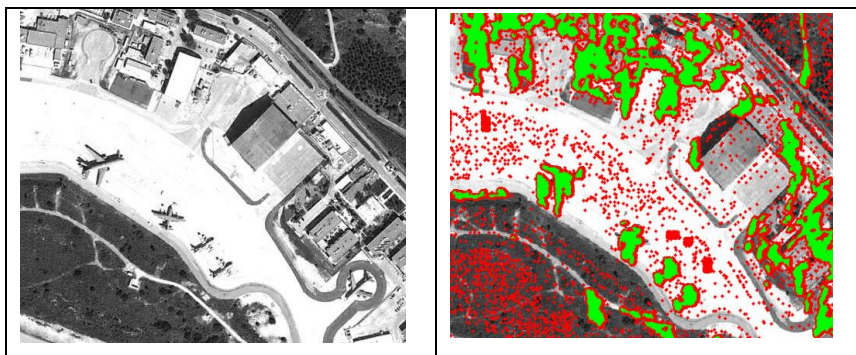
Ο αλγόριθμος εξάγει αρκετά ικανοποιητικά αποτελέσματα σε εικόνες οι οποίες δεν εμφανίζουν μεγάλο βαθμό πολυπλοκότητας. Στις παραπάνω εικόνες φαίνονται κάποιες πολύ χαρακτηριστικές περιπτώσεις εικόνων στις οποίες ο αλγόριθμος εντόπισε όλα τα αεροσκάφη, με πολύ μικρό αριθμό «σκουπιδιών».

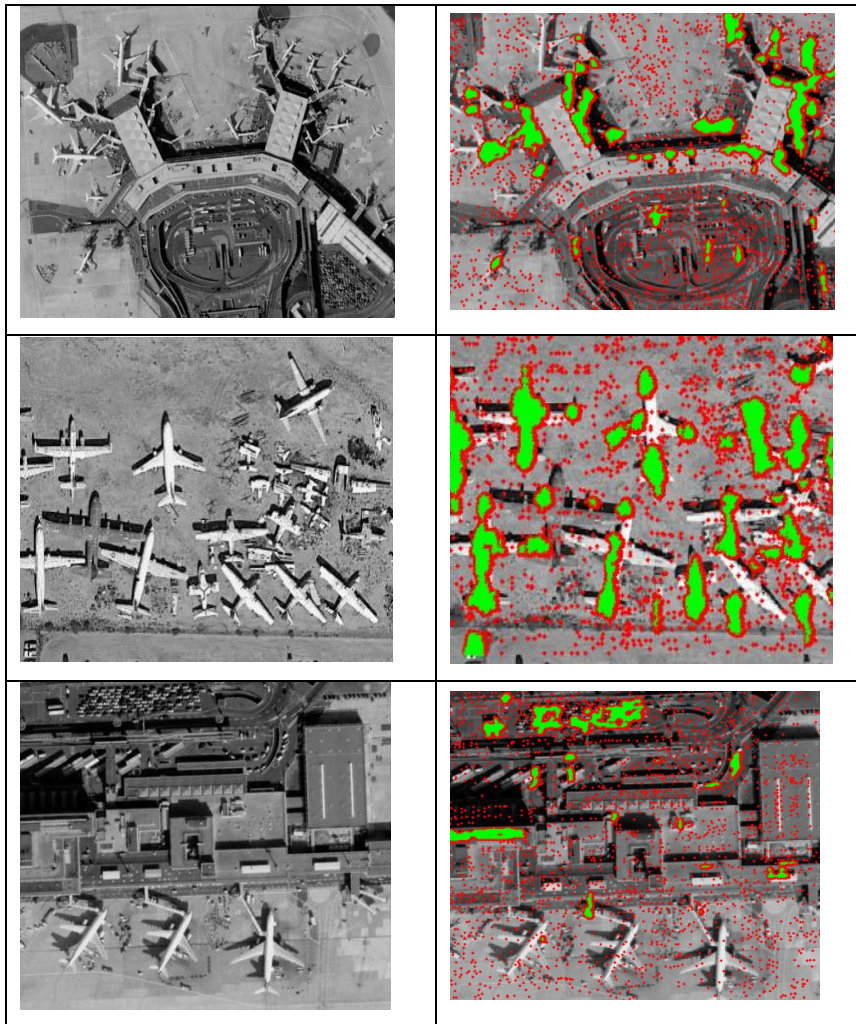
Ένα χαρακτηριστικό πρόβλημα στην ανίχνευση των αεροσκαφών είναι το γεγονός ότι παρατηρούταν, έντονα το φαινόμενο του πολλαπλού εντοπισμού. Αυτό οφείλεται σε μεγάλο βαθμό στη ουρά των αεροσκαφών η οποία μπορεί να θεωρηθεί στην ίδια κλίμακα ως ένα ξεχωριστό αεροσκάφος σε μικρότερο μέγεθος. Το φαινόμενο των πολλαπλών εντοπισμών παρατηρήθηκε και στην περίπτωση ελικοπτέρων τα οποία διέθεταν διπλό στροφέιο (Chinook: στον συγκεκριμένο έλεγχο που έγινε κατασκευάστηκαν δεδομένα εκπαίδευσης για ελικόπτερα μονού στροφείου). Ενώ, ο έλεγχος του αλγορίθμου σε εικόνα που διέθετε τόσο ελικόπτερα όσο και αεροπλάνα (με δεδομένα εκπαίδευσης για αεροσκάφη) ανίχνευε και τα δύο αντικείμενα παρά το γεγονός ότι στα δεδομένα εκπαίδευσης δεν υπήρχαν εικόνες ελικοπτέρων. Παρακάτω φαίνονται μερικά κλασικά παραδείγματα πολλαπλού εντοπισμού:



Εικόνα 43- Παραδείγματα πολλαπλών εντοπισμών

Όπως προαναφέραμε και παραπάνω, στις περιπτώσεις ιδιαίτερα αυξημένης πολυπλοκότητας των εικόνων δοκιμής, δηλαδή μέσα στην ίδια εικόνα εμπεριέχονταν αρκετά αντικείμενα άσχετα με τα αεροσκάφη, τα αποτελέσματα ήταν σχεδόν απογοητευτικά. Τέτοιες περιπτώσεις θα μπορούσαν να χαρακτηριστούν τα πολιτικά αεροδρόμια που δοκιμάστηκαν και υπήρχε πρόβλημα τόσο με τους θαλάμους-διαδρόμους εισόδου στο εσωτερικό των αεροπλάνων, όσο και με διάφορες εγκαταστάσεις που βρίσκονταν στον περιβάλλοντα χώρο.





Εικόνα 44 - Παραδείγματα εντοπισμών σε εικόνες με μεγάλη πολυπλοκότητα

5. Συμπεράσματα Διπλωματικής Εργασίας και Προτάσεις για περαιτέρω έρευνα

Στο κεφάλαιο αυτό παρουσιάζονται τα συμπεράσματα που προέκυψαν από όλες τις μεθοδολογίες που χρησιμοποιήθηκαν κατά την διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας. Πέραν αυτών, αξίζει να αναφερθούν ορισμένες προτάσεις που προέκυψαν για την βελτίωση των αποτελεσμάτων και την διαχείριση των προβλημάτων που αντιμετωπίστηκαν και από τις δύο μεθοδολογίες που εφαρμόστηκαν.

5.1. Γενικά Συμπεράσματα

- Στην πρώτη μεθοδολογία (§3.1. σελίδα 25) η χρήση αρχικών εικόνων με αρκετά καλύτερη χωρική ανάλυση, σε συνδυασμό με πολύ καλό registration πολύ πιθανόν να βελτιστοποιούσε τα τελικά ποσοστά της μεθοδολογίας. Πάνω στην ίδια κατεύθυνση, η αντικατάσταση των βημάτων αντιστοίχισης(ArcGis 10) των εικόνων, με αυτόματες μεθοδολογιών όπως των αλγορίθμων εύρεσης χαρακτηριστικών σημείων μεταξύ δύο εικόνων (Sift/Surf/Open Surf κ.ά.), ίσως να έχει σημαντική συνεισφορά στην βελτίωση των τελικών αποτελεσμάτων.

Όσον αφορά την δεύτερη προσέγγιση (§3.2. σελίδα 46) θα μπορούσαμε να παρατηρήσουμε τα κάτωθι συμπεράσματα:

- Παρά το γεγονός ότι ο Η/Υ στον οποίο πραγματοποιήθηκαν όλες οι δοκιμές είχε επεξεργαστή Core i7 και μνήμη RAM 8Gb, παρουσιάστηκαν αδυναμίες στην εκπαίδευση του SVM για μεγάλο αριθμό εικόνων.
- Ο χρόνος ελέγχου δοκιμαστικής εικόνας επί παραδείγματι διαστάσεων 486*1003 pixels, ξεπερνούσε σε διάρκεια τις 5 ώρες, με αποτέλεσμα ο πειραματισμός διαφόρων παραμέτρων για τον εντοπισμό του τρόπου επιρροής τους στο τελικό αποτέλεσμα εντοπισμού να καθίσταται αρκετά δύσκολος.
- Ο εντοπισμός των πιθανών σημείων για έλεγχο στην δοκιμαστική εικόνα υλοποιήθηκε μέσω της συνέλιξης της δοκιμαστικής εικόνας με τις εικόνες προτύπων. Τα σημεία εξαγωγής ήταν εκείνα τα οποία είχαν και την μεγαλύτερη συσχέτιση μεταξύ των κανονικοποιημένων εικόνων δοκιμής και των προτύπων. Η διαδικασία

αυτή θα μπορούσε να δοκιμαστεί με τους αλγορίθμους εντοπισμού ακμών ή αλγορίθμους εντοπισμού χαρακτηριστικών σημείων όπως ο Harris, Surf, Sift κ.ά.

- Ύπαρξη πολλών παραμέτρων σε όλα τα βήματα υλοποίησης του αλγορίθμου, που καταδεικνύει προφανώς τις ικανότητες του αλγορίθμου, αλλά και την εύκολη μεταβολή των αποτελεσμάτων με αλλαγή σημαντικών παραμέτρων, όπως π.χ. των προτύπων (templates).
- Χαρακτηριστικό παράδειγμα των δυνατοτήτων του αλγορίθμου είναι το γεγονός ότι δημιουργήθηκε για τον εντοπισμό προσώπων και με τις απαραίτητες μετατροπές λειτούργησε αρκετά ικανοποιητικά στον εντοπισμό σταθμευμένων αεροσκαφών, επιτυγχάνοντας ποσοστό εντοπισμού που κυμαινόταν στο 66,28%.

5.2. Ειδικότερα Συμπεράσματα

Στην πρώτη προσέγγιση (§3.1. σελίδα 25) αναπτύχθηκε η μεθοδολογία με σημαντικό βαθμό τυποποίησης. Αν και ο αλγόριθμος διαμορφώθηκε αρχικά με γνώμονα τον αυτοματισμό στον εντοπισμό των αεροσκαφών, στην εξέλιξή του διαπιστώθηκε ότι η παρέμβαση του χειριστή ήταν απαραίτητη σε κομβικά σημεία της όλης μεθοδολογίας. Επομένως, θα μπορούσαμε να πούμε ότι περισσότερο παρουσιάζει χαρακτηριστικά ημιαυτόματης μεθοδολογίας εντοπισμού παρά αυτόματης.

Θα πρέπει να τονισθεί ότι ο αυτοματισμός της 2^{ης} προσέγγισης την καθιστά πολύ περισσότερο φιλική για μελλοντικές εφαρμογές. Η 1^η προσέγγιση παρά το γεγονός ότι υπό προϋποθέσεις εξάγει αρκετά ικανοποιητικά αποτελέσματα θα μπορούσε να χαρακτηριστεί αρκετά απλοϊκή ενώ ίσως ενδιαφέρον θα είχε η προσπάθεια συνδυασμού των αποτελεσμάτων της με την 2^η προσέγγιση των μηχανών υποστήριξης διανυσμάτων.

5.3. Προτάσεις

- Προτείνεται η διερεύνηση της εγκυρότητας των αποτελεσμάτων εντοπισμού και για τις δύο ανωτέρω προσεγγίσεις με βάση Datasets που έχουν χρησιμοποιηθεί από άλλες εργασίες. Ταυτόχρονα, θεωρείται απαραίτητη η σύγκριση των αποτελεσμάτων με βάση μέτρα ακρίβειας, για να εξαχθούν έγκυρα και ασφαλή συμπεράσματα.

- Οι περιορισμοί της υπολογιστικής ισχύος του Η/Υ μπορούν να παρακαμφθούν με την χρήση του “Central CloudFront” του Ε.Μ.Π. Πραγματοποιήθηκε προσπάθεια για την εκμετάλλευση της εφαρμογής, αλλά διαπιστώθηκε ότι η έκδοση Matlab που είναι εγκατεστημένη ήταν του 2008 με αποτέλεσμα να μην παρέχονται και οι αντίστοιχες εργαλειοθήκες που χρησιμοποιήθηκαν κατά την διάρκεια εκπόνησης της διπλωματικής.
- Προτείνεται ο προσανατολισμός των αλγορίθμων σε περισσότερες κατηγορίες στρατιωτικών στόχων.
- Ενδιαφέρον έχει ο συνδυασμός των δύο προσεγγίσεων με ένα «έμπειρο σύστημα» το οποίο θα στηρίζεται σε μια βάση γνώσης για την εξαγωγή ασφαλών συμπερασμάτων ταξινόμησης. Ταυτόχρονα θεωρείται απαραίτητο να δημιουργηθεί μια ολοκληρωμένη βάση δεδομένων στρατιωτικών στόχων η οποία εκτός από τα γεωμετρικά τους χαρακτηριστικά να διαθέτει και πρότυπα φασματικών υπογραφών.
- Τέλος, θα πρέπει να τονισθεί ότι το αντικείμενο του αυτόματου εντοπισμού διάφορων αντικειμένων σε ψηφιακές εικόνες δεν είναι ένα καθαρά τοπογραφικό αντικείμενο. Ανήκει στον ευρύτερο χώρο της «Υπολογιστικής όρασης» ή αλλιώς «Όρασης Υπολογιστών» (Computer Vision) και προφανώς διαρκώς αναπτύσσεται και εξελίσσεται κυρίως από μηχανικούς που ανήκουν στην κοινότητα των Ηλεκτρολόγων Μηχανολόγων Ηλεκτρονικών Υπολογιστών.
- Η απουσία βασικών μαθημάτων επεξεργασίας ψηφιακών εικόνων καθιστά την ενασχόληση με επιστημονικούς προβληματισμούς πάνω σε θέματα της όρασης υπολογιστών ακόμα πιο δύσκολη, σε συνδυασμό πάντα και με τις αντικειμενικές δυσκολίες στην γλώσσα προγραμματισμού στην οποία υλοποιούνται οι δοκιμές.

6. Βιβλιογραφία

- [1] Omid Sakhi, “Face Detection Code for Matlab-A practical Guide to Face Detecting Using Gabor Feature Extraction and Neural Networks”, 2010.
- [2] Sotiris Douscas, “Develop and manufacture of stretcher with camera and algorithm of image processing for the control of movement of a robotic fish”, 2011.
- [3] Tselios Konstantinos, “Planning and development of system of biometrics of recognition based on the finger imprints”, 2008
- [4] Nicolai Petkov, “2D Gabor functions and filters for image processing and computer vision”, 2011.
- [5] Judson P.Jones and Larry A.Palmer, “An evaluation of the two dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex”, 1987
- [6] Karadimas Hlias, “Picture recuperation based on texture with the use of eye tracker”, 2010
- [7] Konstantinos Batargias, “Driver of development of postgraduate papers Οδηγός Εκπόνησης Διπλωματικής Εργασίας”, 2007
- [8] Karras George, “Photogrammetry notes III”, 2013.
- [9] Karantzalos Konstantinos, “Automatic feature extraction from aerial and satellite imagery with computer vision techniques”, 2007.
- [10] Poulos Dimitrios, “Create point cloud from 2D images”, 2013
- [11] Boureksi Fotiou, “Photogrammetric analysis and evaluation on results of depictions TerraSar – X”, 2012
- [12] Xaroula Andreou, “Possibilities and Restrictions of hyper spectral remote sensing in detection qualitative characteristics of paving”, 2008
- [13] Kosmidis Eustathios, “Investigation on Georeferencing Satellite Pictures”, 2012.
- [14] Argialas Dimitrios., Karantzalos Konstantinos., Maurantza O., Noutsou B., Tzotzos A., Koutroubas G., “Picture analysis, pattern recognition and computer vision in remote sensing – Methods and applications”, 2005
- [15] Sofia Samoli, “Automatic Recognition of vehicles from multispectral and elevation data (lidar) with object oriented analysis”, 2008.
- [16] Yanfeng Gu, Chen Wang, Shizhe Wang, Ye Zhang, “Pattern Recognition Letters”, 2010
- [17] G. Simone, A. Farina, F.C. Morabito, S.B. Serpico, L. Bruzzone, “Image fusion techniques sensing applications”, 2001
- [18] Vassilis Tsagaris, Member, IEEE, Vassilis Anastassopoulos, Member, IEEE, and George A. Lampropoulos, Member, IEEE, “Fusion of Hyperspectral Data Using Segmented PCT for Color Representation and Classification”, 2005.
- [19] Cem Unsalan, “Detecting Changes in Multispectral Satellite Images using Time Dependent Angle Vegetation Indices”, 2007
- [20] Themelis Athanasios, “Classification of Urban Space with the use of hyper spectral Remote Sensing, 2010.
- [21] Taejung Kima, Jae-In Kim, “Automated Assesment of NIIRS and GRD of High Resolution Satellite Images through Edge Profile Analysis of Natural Targets”, 2013
- [22] A. Erener, “Classsification method, spectral diversity, band combination and accuracy assessment evaluation for urban feature detection”, 2011.

- [23] P. Nalil and Dr. Natarajan, "Automatic Road Extraction from High Resolution Imagery Based On Statistical Region Merging and Skeletonization", 2010
- [24] Georgios Karagiannis and Demetre Argialas, "Semi-automatic Object-Based Building Change Detection in Suburban Areas from Quickbird Imagery Using the ERDAS Imagine Objective Software",
- [25] M. Mokhtarzade, M. J. Valadan Zoej, H. Ebadi, "Automatic road extraction from resolution satellite images using neural networks, texture analysis, fuzzy clustering and genetic algorithms", 2006.
- [26] Affine transformation, Wikipedia.
- [27] Edouard Oyallon, Jullien Rabin, "An analysis and implementation of the SURF method and its comparison to SIFT", 2013.
- [28] Sargur Srihari, "The Hessian Matrix", 2013
- [29] Christos, "Least Square image registration", 2007.
- [30] Eigenvalues and eigenvectors, Wikipedia.
- [31] Gaussian function, Wikipedia
- [32] Michael Mathioudaki, "Comparison of summaries (wavelet synopses) for various metric fault", 2006.
- [33] Kalisperakis Hlias, "The epipolar geometry in the projective and euclidian space", 2010.
- [34] Laplacian Matrix, Wikipedia.
- [35] Mahalanobis distance, Wikipedia
- [36] Matrix Norm, Wikipedia
- [37] Xaritos Dakolias, "Import in not linear programming", 2011
- [38] Jacob Toft Pedersen, "SURF: Feature detection & description", 2011
- [39] Rigas Kouskouridas, Basileios Belagiannis, Dimitrios Xrisostomou and Adonios Gasteratos, "Comparative Study of Algorithms on Export Characteristics", 2009
- [40] Helbert Bay, Tinne Tuytelaars and Luc Van Gool, " SURF:: Speeded Up Robust Features", 2004.
- [41] A.C Murillo, J. J Guerrero and C. Sagues, "SURF features for efficient robot localization with omnidirectional images", 2006.
- [42] Eirini Braxou, "Image Classification Methods, 2010.
- [43] Matrix Jacobi, Wikipedia.
- [44] Cheolha Pedro Lee, "Mine Detection Techniques Using Multiple Sensors", 2013
- [45] S.M. Khanna, F. Paquet, R. Apps and J.S. Seregelyi, "A new remote –sensing method for mine detection using HPM irradiation and IR detection, 1999.
- [46] B.H.P. Maathuis, "Remote Sensing Based Detection of Minefields", 2003.
- [47] Hao Sun, Xian Sun, Hongqi Wang, Yu Li, and Xiangjuan Li, "Automatic Target Detection in High-Resolution Remote Sensing Images Using Spatial Sparse Coding Bag-of-Words Model", 2012
- [48] A. Khashman, "Automatic Detection of Military Targets utilizing Neural Networks and Scale Analysis", 2000.
- [49] K.C Tiwatri, M.K Arora, D. Singh, "An assessment of independent component analysis for detection of military targets from hyperspectral images", 2011.
- [50] The UK Military Space Primer", 2010

- [51] L.J Denes, M. Gottlieb, B. Kaminsky, "Spectro-Polarimetric Imaging for Object Recognition", 1997.
- [52] Zhenwei Shi, Shuo Yang, Zhen Qin, "A regularized non-Gaussianity based multiple-target detector for hyperspectral images", 2012
- [53] F.A Kruse and A.B. Lefkoff, "Analysis of spectral data of Manmade Materials, Military Targeys and Background Using an Expert System Based Approach", 2013
- [54] Kailash Chandra Tiwari, Manoj K. Arora, Dharmendra Singh, Deepti Yadav, "Military target detection using spectrally modeled algorithm and independent component analysis", 2011.