



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ
ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

**«ΜΑΘΗΜΑΤΙΚΗ ΠΡΟΤΥΠΟΠΟΙΗΣΗ σε ΣΥΓΧΡΟΝΕΣ ΤΕΧΝΟΛΟΓΙΕΣ
και την ΟΙΚΟΝΟΜΙΑ»**

**Ασαφείς Μηχανές Διανυσμάτων Υποστήριξης (SVM)
για ανίχνευση αντικειμένων σε πραγματικό χρόνο**

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΣΕΡΕΤΗΣ ΓΕΩΡΓΙΟΣ

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 09312036

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΣΤΕΦΑΝΟΣ ΚΟΛΛΙΑΣ

ΑΘΗΝΑ, ΦΕΒΡΟΥΑΡΙΟΣ 2015

Abstract

In this thesis a Takagi-Sugeno (TS) fuzzy system is proposed, which is trained by a support vector machine (SVM) with principal component analysis (PCA) (TFS-SVMPC) and provides real time object detection. Its performance is compared with the mostly used methods in this field. The first part of the TFS-SVMPC classifier is based on a fuzzy clustering algorithm. The free parameters in this part relate to the principal component analysis. Then a linear SVM is used to adjust the parameters and give the system a better general performance. In this thesis the above system is used as a classifier in a real time object detection system. This system includes two stages. In the first stage, a color histogram of the global color hue of the object is used as a real time detection feature. In the second stage, a dependent on the appearance of local (point) colors geometry is used as a color feature. A comparison between the proposed method and existing classifiers and tentative methods is presented, confirming the good performance of the proposed, based on the TFS-SVMPC model, detection method.

Περίληψη

Στην εργασία αυτή προτείνεται ένα Takagi-Sugeno (TS) ασαφές σύστημα, το οποίο εκπαιδεύεται μέσω μιας μηχανής διανυσμάτων υποστήριξης (support vector machine - SVM) με ανάλυση κύριων συνιστωσών (Principal component analysis - PCA) (TFS-SVMPC) και παρέχει ανίχνευση αντικειμένων σε πραγματικό χρόνο. Η απόδοσή του συγκρίνεται με τις κυριότερες μεθόδους στο πεδίο αυτό. Το πρώτο μέρος του TFS-SVMPC ταξινομητή, βασίζεται σε έναν αλγόριθμο ασαφούς ομαδοποίησης. Οι ελεύθερες παράμετροι σε αυτό το μέρος αφορούν στην ανάλυση κύριων συνιστωσών. Στη συνέχεια χρησιμοποιείται ένα γραμμικό SVM για να προσαρμόσει τις παραμέτρους και να δώσει στο σύστημα καλύτερη γενικευμένη επίδοση. Στην παρούσα εργασία το ανωτέρω σύστημα χρησιμοποιείται ως ταξινομητής σε ένα σύστημα ανίχνευσης αντικειμένων σε πραγματικό χρόνο. Το σύστημα αυτό αποτελείται από δύο στάδια. Το πρώτο στάδιο χρησιμοποιεί ένα χρωματικό ιστόγραμμα της ολικής χρωματικής απόχρωσης του αντικειμένου ως χαρακτηριστικό ανίχνευσης σε πραγματικό χρόνο. Το δεύτερο στάδιο χρησιμοποιεί γεωμετρία εξαρτώμενη από την εμφάνιση τοπικών (σημειακών) χρωμάτων, ως χρωματικό χαρακτηριστικό. Παρατίθεται σύγκριση με υπάρχοντες ταξινομητές και ανιχνευτικές μεθόδους, επιβεβαιώνοντας την καλή επίδοση της προτεινόμενης, βασισμένης στο μοντέλο TFS-SVMPC, μεθόδου ανίχνευσης.

Περιεχόμενα

1	Εισαγωγή.....	1
2	Θεωρητική Βάση.....	3
2.1	Μηχανές Διανυσμάτων Υποστήριξης (SVMs).....	3
2.1.1	Γραμμικά SVMs.....	3
2.1.2	Μη γραμμικά Kernel based SVMs.....	8
2.2	Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis – PCA).....	12
2.3	Ασαφείς ταξινομητές (fuzzy classifiers).....	18
2.3.1	Fuzzy rule-based classifiers.....	18
2.3.2	A Fuzzy prototype-based classifier.....	19
2.4	Takagi – Sugeno fuzzy systems (TS model).....	20
2.5	Μέθοδοι Ανίχνευσης.....	22
2.5.1	OpenCV (Open Source Computer Vision Library).....	22
2.5.2	SIFT (Μετασχηματισμός Κλιμακωτά Αναλλοίωτων Χαρακτηριστικών).....	23
2.5.3	Histogram based (Κατάτμηση Ιστογράμματος).....	29
3	Προτεινόμενη Μέθοδος.....	31
3.1	Η μέθοδος TFS-SVMPC.....	31
3.1.1	TFS-SVMPC – Μαθηματικές Συναρτήσεις.....	31
3.1.2	TFS-SVMPC – Δομική Μάθηση.....	33
3.2	Η μέθοδος TFS-SVMPC – Εκμάθηση Παραμέτρων.....	34
3.2.1	Η βασική ιδέα ενός γραμμικού SVM.....	34
3.2.2	Μείωση της διάστασης των χαρακτηριστικών διανυσμάτων με τη χρήση της PCA.....	35
3.3	Parameter Learning με ένα γραμμικό SVM.....	37
4	Ανίχνευση με βάση Περιγραφείς Ολικούς-Τοπικούς και την Προτεινόμενη Μέθοδο.....	38
4.1	Διαίρεση του Χρωματικού Χώρου και Ιστόγραμμα.....	39
4.2	Εξαγωγή Χαρακτηριστικών Ιστογράμματος.....	40
4.3	Παραγωγή Υποψήφιου Αντικειμένου.....	43
4.4	Ανίχνευση με βάση Περιγραφείς Ολικούς-Τοπικούς.....	44
5	Πειράματα.....	46
5.1	Offline Πειράματα και Αποτελέσματα.....	46
5.2	Συγκρίσεις Με Άλλους Ταξινομητές.....	52

5.3	Συγκρίσεις Με Άλλες Μεθόδους Ανίχνευσης.....	53
5.4	Πείραμα Ανίχνευσης Αντικειμένων Σε Πραγματικό Χρόνο	54
6	Συμπεράσματα.....	58
7	Βιβλιογραφία	59
8	Παράρτημα – Πίνακες.....	60

1 Εισαγωγή

Οι τεχνικές ανίχνευσης βασισμένες σε υπολογιστική όραση, έχουν δείξει πως έχουν πολλές εφαρμογές, όπως η αυτόματη ανίχνευση ελαττωμάτων και τα συστήματα ανίχνευσης αντικειμένων. Η ανίχνευση αντικειμένων περιλαμβάνει τον προσδιορισμό του εάν ένα γνωστό αντικείμενο βρίσκεται σε μία σκηνή (οπτικό χώρο) και, εάν βρίσκεται, τον καθορισμό της ακριβούς θέσης του αντικειμένου. Οι εφαρμογές ανίχνευσης αντικειμένων μπορούν να ταξινομηθούν σε ανίχνευση μίας ομάδας αντικειμένων ενδιαφέροντος (όπως πρόσωπα και αυτοκίνητα) και σε ανίχνευση ενός συγκεκριμένου αντικειμένου (όπως ενός συγκεκριμένου μεταλλικού δοχείου σε μία κουζίνα). Η παρούσα εργασία επικεντρώνεται στην τελευταία κατηγορία και το προτεινόμενο σύστημα είναι ιδιαίτερα χρήσιμο για αναγνώριση ενός γνωστού αντικειμένου με ποικιλία χρωμάτων σε μη-ομοιογενή περιβάλλοντα. Η προτεινόμενη προσέγγιση ανίχνευσης αντικειμένων μπορεί να εφαρμοσθεί σε ευφυή ρομπότ για ανίχνευση στόχου ή σε ρομπότ πλοήγησης.

Ένα σημαντικό στάδιο της ανίχνευσης αντικειμένων είναι η ταξινόμηση. Ταξινομητές που χρησιμοποιούν τις τεχνικές της αντιστοίχισης περιγράμματος, των νευρωνικών δικτύων (NNs), των ποσοτικά αυξανόμενων δενδρικών ταξινομητών και των support vector machine (SVM), έχουν προταθεί και εφαρμοσθεί σε προβλήματα ανίχνευσης διαφόρων αντικειμένων. Πρόσφατα, έχουν επίσης προταθεί διάφοροι νευρωνικοί και SVM-εκπαιδευόμενοι fuzzy ταξινομητές. Για τους νευρωνικούς fuzzy ταξινομητές, οι παράμετροι υπολογίζονται ελαχιστοποιώντας το σφάλμα εκπαίδευσης (training error) (ή το αντίστοιχο εμπειρικό ρίσκο), το οποίο όμως δεν μπορεί να υπολογιστεί σε μικρές δοκιμές. Για να ξεπεραστεί αυτό το μειονέκτημα, έχουν προταθεί αρκετοί SVM-εκπαιδευόμενοι ταξινομητές. Αντί να ελαχιστοποιούν μία αντικειμενική συνάρτηση βασισμένη στην εκπαίδευση (training), οι SVM-εκπαιδευόμενοι fuzzy ταξινομητές επιχειρούν να ελαχιστοποιήσουν έναν περιορισμό για το γενικευμένο σφάλμα. Ένα αυτο-οργανούμενο Takagi-Sugeno (TS) fuzzy δίκτυο με support vector learning (SOTFN-SV) που χρησιμοποιεί πρώτης τάξης TSK κανόνες, αναλύεται στην παρούσα εργασία. Ωστόσο, αντίστοιχα με τους μηδενικής τάξης fuzzy κανόνες στους περισσότερους SVM-εκπαιδευόμενους ταξινομητές, υπάρχει μόνο μία ελεύθερη προκύπτουσα παράμετρος στα SOTFN-SV. Αυτή η εργασία προτείνει ένα TS fuzzy σύστημα εκπαιδευόμενο μέσω ενός SVM με διάστημα που προκύπτει από ανάλυση κύριων συνιστωσών (TFS-SVMPC) με πρώτης τάξης TS fuzzy κανόνες. Σε αντίθεση με τα SOTFN-SV, όλες οι ελεύθερες προκύπτουσες παράμετροι είναι αρμονικές στα TFS-SVMPC, ενώ προτείνεται μία νέα μέθοδος εκπαίδευσης των προκύπτουσών παραμέτρων. Όμως, το μέγεθος του σετ των παραμέτρων είναι μεγάλο για ένα πρώτης τάξης TS fuzzy σύστημα, ιδιαίτερα σε περιπτώσεις μεγάλων διαστάσεων. Το προτεινόμενο TFS-SVMPC αντιμετωπίζει το αποτέλεσμα του προβλήματος εκπαίδευσης διανύσματος μεγάλων διαστάσεων σχεδιάζοντάς το στο διάστημα των κύριων συνιστωσών. Βασίζεται σε γραμμικά SVM τα οποία βοηθούν στην εκμάθηση (learning) των προκύπτουσών παραμέτρων, σε ένα διάστημα κύριων συνιστωσών. Τα δύο μεγαλύτερα ενδιαφέροντα στην υλοποίηση fuzzy συστημάτων είναι η ακρίβεια/ορθότητα του μοντέλου και η δυνατότητα αιτιολογήσεων των προτάσεων του μοντέλου. Η παρούσα εργασία εστιάζει στο σχεδιασμό fuzzy

συστημάτων για ταξινομήσεις με μεγάλη ακρίβεια. Το TFS-SVMPC σύστημα εφαρμόζεται στη συνέχεια για ανίχνευση διαφορετικών αντικειμένων επιβεβαιώνοντας την καλά επίδοση της ανίχνευσης.

Εκτός από το σχεδιασμό του ταξινομητή, ένα άλλο σημαντικό έργο στην ανίχνευση αντικειμένων είναι η εξαγωγή χαρακτηριστικών. Οι γεωμετρικές πληροφορίες και το σχήμα των αντικειμένων είναι δύο ευρέως χρησιμοποιούμενα χαρακτηριστικά. Οι βασισμένες στο σχήμα τεχνικές συνήθως απαιτούν σύνθετους υπολογισμούς για να ανιχνεύσουν αντικείμενα, με απόκλιση στην άποψη (viewpoint) και στη θέση, ιδιαίτερα σε σύνθετες σκηνές (complex scenes). Άλλα χαρακτηριστικά περιλαμβάνουν μετατροπή της τιμής του επιπέδου του γκρι (gray-level values), σημεία κλειδιά στην εμφάνιση (appearance key points), πληροφορία χρώματος (color information), ή εξαγωγή σημείου κλειδιού (key point extraction), τα οποία χρησιμοποιούνται στη μέθοδο μετατροπής αμετάβλητης κλίμακας χαρακτηριστικών (SIFT). Στην παρούσα εργασία προτείνεται μία μέθοδος ανίχνευσης αντικειμένων δύο φάσεων, στην οποία προτείνονται νέα χρωματικά χαρακτηριστικά, χρησιμοποιώντας καθολική και τοπική χρωματική ανάλυση.

Η παραπάνω εισαγωγή δείχνει πώς η επίδοση ενός συστήματος ταξινόμησης εξαρτάται τόσο από τα μοντέλα ταξινόμησης, όσο και από τα εξαγόμενα χαρακτηριστικά. Η επίδοση ενός συστήματος ανίχνευσης αντικειμένων βελτιώνεται λαμβάνοντας υπ' όψη και τους 2 αυτούς παράγοντες. Συνολικά, η συνεισφορά της προτεινόμενης προσέγγισης της βασισμένη σε TFS-SVMPC ανίχνευσης αντικειμένων, είναι τριπλή: **1)** Προτείνεται ένας νέος TS fuzzy ταξινομητής, ο TFS-SVMPC, περιλαμβάνοντας μία νέα παραμετρική μέθοδο εκμάθησης που χρησιμοποιεί SVM κύριων συνιστωσών διαστήματος για να προσδώσει στον TFS-SVMPC καλή ικανότητα γενίκευσης. **2)** Προτείνεται ένα νέο, βασισμένο στο χρώμα, χαρακτηριστικό, για την ανίχνευση αντικειμένων. **3)** Η προτεινόμενη, βασισμένη σε TFS-SVMPC, προσέγγιση της ανίχνευσης αντικειμένων, εφαρμόζεται σε ένα σύστημα που χρησιμοποιεί μία pan-tilt κάμερα δωματίου.

Στο Κεφάλαιο 2, επεξηγείται η λειτουργία των Μηχανών Διανυσμάτων Υποστήριξης (SVM) και γίνεται μια παρουσίαση των γραμμικών και μη γραμμικών περιπτώσεών τους. Ακολουθούν η θεωρητική βάση και η επεξήγηση της Ανάλυσης Κύριων Συνιστωσών (Principal Component Analysis - PCA), καθώς και των Ασαφών Ταξινομητών (Fuzzy Classifiers) και του διαχωρισμού τους. Στη συνέχεια παρουσιάζεται και επεξηγείται το γενικό Takagi-Sugeno μοντέλο (TS model), πάνω στο οποίο βασίζεται η εξεταζόμενη σε αυτήν την εργασία μέθοδος. Τέλος, επεξηγούνται οι μέθοδοι ανίχνευσης με τις οποίες γίνεται σύγκριση της εξεταζόμενης μεθόδου. Συνολικά, το Κεφάλαιο 2 αφιερώνεται στην παρουσίαση και κατανόηση των θεωρητικών βάσεων που είναι απαραίτητες για την παρούσα εργασία.

Στο Κεφάλαιο 3, γίνεται μία παρουσίαση και ανάλυση της μεθόδου TFS-SVMPC, στη βάση τόσο των μαθηματικών συναρτήσεων και της δομικής μάθησης, όσο και της εκμάθησης παραμέτρων και της μείωσης της διάστασης των χαρακτηριστικών που επεξεργάζεται.

Ακολουθεί, στο Κεφάλαιο 4, η διαδικασία ανίχνευσης αντικειμένων σε όλα τα στάδια (από τον χαρακτηρισμό των υποψήφιων αντικειμένων ως την τελική απόρριψη ή αναγνώρισή τους ως αντικείμενα) και η παρουσίαση των πειραμάτων και σύγκριση με άλλους ταξινομητές στο Κεφάλαιο 5.

Τέλος, στο Κεφάλαιο 6, παρουσιάζονται συγκεντρωτικά τα συμπεράσματα της παρούσας εργασίας και προτάσεις για πιθανή μελλοντική έρευνα.

2 Θεωρητική Βάση

2.1 Μηχανές Διανυσμάτων Υποστήριξης (SVMs)

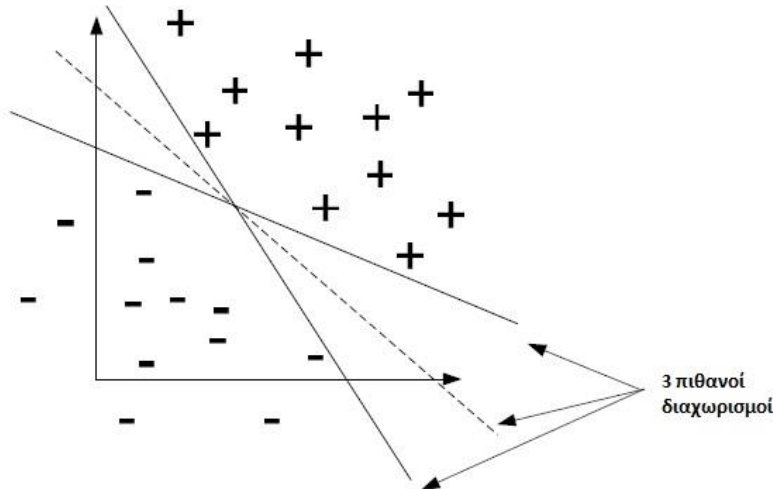
Οι μηχανές διανυσμάτων υποστήριξης (SVMs) ανήκουν στην ευρύτερη κατηγορία των γενικευμένων γραμμικών ταξινομητών, καθώς επιτυγχάνουν το διαχωρισμό των δεδομένων με τη χρήση υπερεπιπέδων. Η θεωρία των SVMs έχει τις αρχές της στα τέλη της δεκαετίας του '70, αλλά η πρώτη παρουσίασή της έγινε το 1992 από τους Boser, Guyon και Vapnik στο COLT-92 (Computational Learning Theory – 92). Τα τελευταία χρόνια, ωστόσο, παρατηρείται πολύ συχνή χρήση της σε πολλούς τομείς αναγνώρισης προτύπων, όπως στην αυτόματη αναγνώριση των χαρακτήρων του αλφαβήτου, στην αναγνώριση αντικειμένων, στον εντοπισμό προσώπων σε εικόνες, κτλ. Οι SVMs αποτελούν ένα σύνολο μεθόδων μάθησης με επίβλεψη που χρησιμοποιούνται για την ταξινόμηση και την παλινδρόμηση. Με άλλα λόγια, η μηχανή διανυσμάτων υποστήριξης (SVM) είναι ένα εργαλείο ταξινόμησης και πρόβλεψης συναρτήσεων, που χρησιμοποιεί τη θεωρία εκμάθησης μηχανών (computer learning) για να μεγιστοποιεί την ακρίβεια πρόβλεψης, ενώ αποφεύγει την υπερπροσαρμογή (over-fitting) στα στοιχεία. Μπορούν να οριστούν ως τα συστήματα που μετατρέπουν τον χώρο υπόθεσης των γραμμικών συναρτήσεων σε έναν χώρο χαρακτηριστικών μεγάλης διάστασης, που εκπαιδεύεται με έναν αλγόριθμο βελτιστοποίησης. Οι μηχανές SVM έγιναν διάσημες όταν έδωσαν ακρίβεια συγκρίσιμη με αυτή περίπλοκων νευρωνικών δικτύων στην προσπάθεια αναγνώρισης γραφής.

2.1.1 Γραμμικά SVMs

Αποτελούν την απλούστερη περίπτωση SVM και εκπαιδεύονται σε γραμμικώς διαχωρίσιμα δεδομένα. Ας θεωρήσουμε ένα πρόβλημα δυαδικής ταξινόμησης. Ο στόχος είναι η εύρεση ενός υπερεπιπέδου που να χωρίζει το σύνολο κατά τέτοιο τρόπο ώστε τα διανύσματα ίδιας κατηγορίας να ανήκουν στην ίδια πλευρά του υπερεπιπέδου.

2.1.1.1 Γενική θεωρία

Έστω στοιχεία εκπαίδευσης $D = \{(\vec{x}_i, y_i), i = 1, \dots, N\}$ με $y_i \in \{-1, +1\}$. Το στοιχείο \vec{x} έχει διάσταση n και ανήκει σε μία από τις 2 κατηγορίες -1 ή +1.

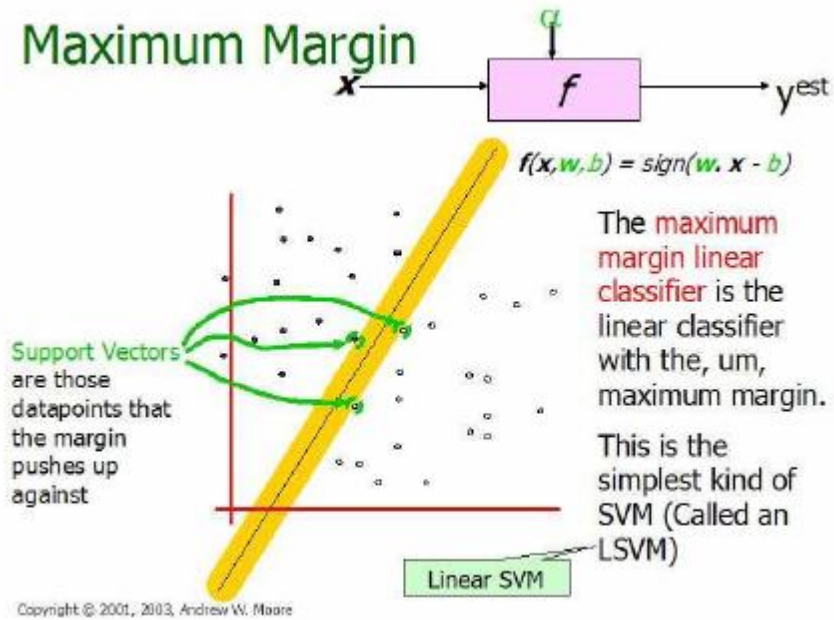


Εικόνα 1 - Παράδειγμα ταξινόμησης 2D

Στην εικόνα 1 φαίνεται πως υπάρχουν αρκετοί πιθανοί τρόποι διαχωρισμού των δεδομένων. Κατά συνέπεια, πρέπει να βρεθεί ο καλύτερος γραμμικός ταξινομητής (classifier) του τύπου

$$f(\vec{x}) = \vec{w}^T \vec{x} + b$$
$$(= w_1 x_1 + w_2 x_2 + w_3 x_3 + \dots + w_n x_n + b) \quad (1.1)$$

από ένα άπειρο αριθμό υπερεπιπέδων που επιτυγχάνουν ακρίβεια 100% στην κατηγοριοποίηση των στοιχείων εκπαίδευσης. Είναι απαραίτητο το βέλτιστο υπερεπίπεδο, να μην προσεγγίζει περισσότερο το ένα από τα δύο σύνολα δεδομένων και κατά συνέπεια να παρέχει κάποια ανοχή και σε περίπτωση εισαγωγής νέων δεδομένων. Προφανής λύση είναι ο ταξινομητής μέγιστου περιθωρίου (maximum margin) μεταξύ των δύο συνόλων δεδομένων (θετικών και αρνητικών). Βέλτιστος γραμμικός ταξινομητής είναι το υπερεπίπεδο στη μέση του περιθωρίου.



Εικόνα 2 - Παρουσίαση γραμμικού SVM με maximum margin classifier (Andrew W. Moore 2003)

Δοθείσης της συνάρτησης $f(x)$, η ταξινόμηση λαμβάνεται από τον τύπο

$$\hat{y} = \text{sign}(f(x)) = \begin{cases} +1, & f(x) > 0 \\ -1, & f(x) < 0 \end{cases} \quad (1.2)$$

Διαφορετικά w και b μπορούν να οδηγήσουν στην ίδια ταξινόμηση. Δηλαδή:

$$\hat{y} = \text{sign}(a(w^T x + b)) = \text{sign}(w^T x + b) \quad (1.3)$$

Επομένως, υπάρχουν πολλές λύσεις.

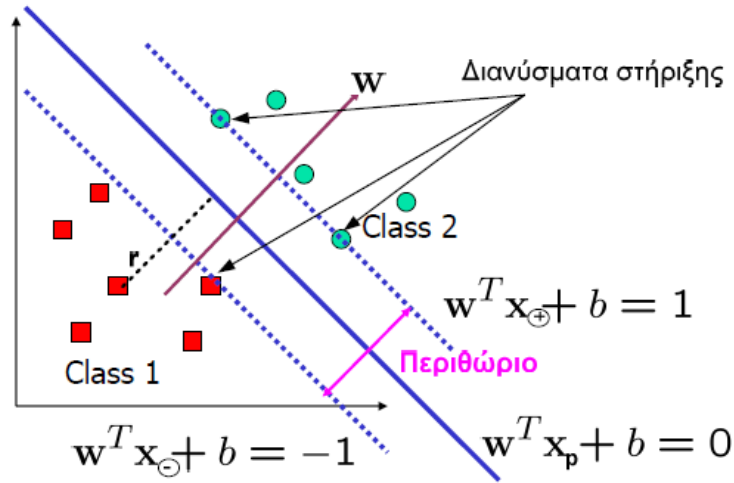
Από όλες τις πιθανές λύσεις αναζητείται η $f(x) = (w^T x + b)$ με το μέγιστο περιθώριο, έτσι ώστε για τα σημεία x_i πιο κοντά στο διαχωριστικό υπερεπίπεδο να ισχύει

$$|w^T x + b| = 1 \quad (1.4)$$

Τα σημεία αυτά ονομάζονται **διανύσματα υποστήριξης**.

Για τα υπόλοιπα σημεία θα ισχύει

$$|w^T x + b| > 1 \quad (1.5)$$



Εικόνα 3 - Απεικόνιση περιθωρίου (margin) και διανυσμάτων στήριξης (support vectors)

Αν υπάρχει ένα υπερεπίπεδο που να ικανοποιεί τις εξισώσεις (1.4) και (1.5), τότε το πρόβλημα είναι γραμμικά διαχωρίσιμο. Αν w και b είναι οι βέλτιστες τιμές για το διάνυσμα βαρών και το bias, τότε το βέλτιστο υπερεπίπεδο περιγράφεται από τη σχέση

$$w^T x_p + b = 0 \quad (1.6)$$

Τα x_i για τα οποία η εξίσωση (1.4) ισχύει σαν ισότητα ονομάζονται «Διανύσματα Υποστήριξης» (Support Vectors). Τα x_i αυτά, βρίσκονται πιο κοντά στο επίπεδο απόφασης και κατά συνέπεια είναι δυσκολότερο να ταξινομηθούν. Πιθανή μετακίνησή τους οδηγεί σε διαφορετικό βέλτιστο υπερεπίπεδο και συνεπάγεται αλλαγή της λύσης του προβλήματος.

Όσον αφορά στο μήκος του περιθωρίου (margin) ως συνάρτηση του w , έστω ότι r είναι η απόσταση του σημείου x από το υπερεπίπεδο, όπως φαίνεται στην εικόνα 3. Λαμβάνοντας υπόψη ότι το w είναι μία διανυσματική κάθετος στο υπερεπίπεδο, έχουμε

$$f(x) = f\left(x_p + \frac{w}{\|w\|} r\right) = w^T x_p + w^T \frac{w}{\|w\|} r + b = \|w\| r \quad (\text{όπου } w^T x_p + b = 0) \quad (1.7)$$

Επομένως: $r = \frac{f(x)}{\|w\|} \quad (1.8)$

Το μήκος του περιθωρίου ρ είναι

$$\rho = \frac{f(x_{\oplus})}{\|w\|} - \frac{f(x_{\ominus})}{\|w\|} = \frac{1}{\|w\|} - \frac{-1}{\|w\|} = \frac{2}{\|w\|} \quad (1.9)$$

Επομένως η μεγιστοποίηση του περιθωρίου ισοδυναμεί με την ελαχιστοποίηση του $\|w\|$.

2.1.1.2 Πρόβλημα εκμάθησης

Έχοντας ένα γραμμικά διαχωρίσιμο σύνολο δεδομένων, ο στόχος των συντελεστών μάθησης w και b της μηχανής διανυσμάτων υποστήριξης $f(x) = (w^T x + b)$ είναι η επίλυση του ακόλουθου προβλήματος βελτιστοποίησης με περιορισμούς

- Να βρεθούν τα w και b που ελαχιστοποιούν την $\frac{1}{2} \|w\|^2$ (1.10)

- Με τον περιορισμό $y_i(w^T x_i + b) \geq 1, \forall i$ (1.11)

Αυτό το πρόβλημα βελτιστοποίησης μπορεί να λυθεί με τη χρήση της lagrangian συνάρτησης που ορίζεται ως

$$L(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^N \alpha_i [y_i(w^T x + b) - 1], \quad \text{με } \alpha_i \geq 0, \forall i \quad (1.12)$$

όπου $(\alpha_1, \alpha_2, \dots, \alpha_N)$ είναι πολλαπλασιαστές Lagrange και $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$.

Η λύση του προβλήματος βελτιστοποίησης με περιορισμούς καθορίζεται από το σαγματικό σημείο της $L(w, b, \alpha)$ που πρέπει να ελαχιστοποιηθεί ως προς τα w και b και να μεγιστοποιηθεί ως προς το α .

- Εάν $y_i(w^T x + b) > 1$, η τιμή $\alpha_i = 0$ μεγιστοποιεί την $L(w, b, \alpha)$
- Εάν $y_i(w^T x + b) < 1$, η τιμή $\alpha_i = +\infty$ μεγιστοποιεί την $L(w, b, \alpha)$
Εφόσον τα w και b θέλουμε να ελαχιστοποιούν την $L(w, b, \alpha)$, πρέπει να αλλάξουν με τέτοιο τρόπο, ώστε $y_i(w^T x + b) = 1$
- Οι Kuhn Tucker συνθήκες είναι: $\alpha_i \{y_i(w^T x + b) - 1\} = 0, \forall i$

Τα σημεία x_i με $\alpha_i > 0$ είναι τα **διανύσματα υποστήριξης**.

Οι απαραίτητες συνθήκες για το σαγματικό σημείο της $L(w, b, \alpha)$ είναι

$$\begin{aligned} \frac{\partial L}{\partial w_j} &= 0, \quad \forall j \\ \frac{\partial L}{\partial \alpha_i} &= 0, \quad \forall i \end{aligned} \quad (1.13)$$

ή, διαφορετικά $\nabla_w L = 0$
 $\nabla_\alpha L = 0$

βρίσκοντας τα αποτελέσματα με τις απαραίτητες συνθήκες στο

$$w = \sum_{i=1}^N \alpha_i y_i x_i \quad (1.14)$$

$$\sum_{i=1}^N \alpha_i y_i = 0 \quad (1.15)$$

Με αντικατάσταση της εξίσωσης (1.14) στη Lagrangian και τη χρήση της εξίσωσης (1.15) ως νέο περιορισμό, το δυικό πρόβλημα βελτιστοποίησης μπορεί να εκφραστεί ως εξής:

Να βρεθεί το α για το οποίο μεγιστοποιείται η παράσταση

$$\sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (1.16)$$

με τη συνθήκη

$$\sum_{i=1}^N \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad \forall i \quad (1.17)$$

που αποτελεί ένα πρόβλημα κύρτωσης τετραγωνικού προγραμματισμού με υπαρκτό σφαιρικό ελάχιστο, που μπορεί να επιλυθεί από διάφορες ρουτίνες βελτιστοποίησης. Η βελτιστοποίηση μπορεί να επιλυθεί σε $O(N^3)$ χρόνο (κυβική πολυπλοκότητα ως προς το μέγεθος των στοιχείων κατάρτισης) και σε γραμμικό χρόνο ως προς τον αριθμό ιδιοτήτων.

Δεδομένων των τιμών (a_1, a_2, \dots, a_N) που λαμβάνονται από την επίλυση, ο τελικός SVM predictor μπορεί να εκφραστεί από την εξίσωση (1.14) ως

$$f(x) = w^T x_i + b = \sum_{i=1}^N \alpha_i y_i x_i^T x + b \quad (1.18)$$

όπου

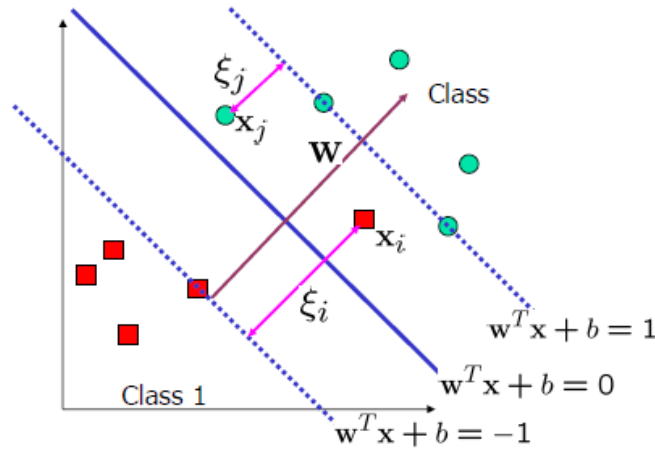
$$b = \frac{1}{|I_{support}|} \sum_{i \in I_{support}} (y_i - \sum_j \alpha_j y_j x_j^T x_i) \quad (1.19)$$

και $I_{support}$ είναι το σύνολο των διανυσμάτων υποστήριξης.

2.1.2 Μη γραμμικά Kernel based SVMs

Στις περισσότερες πραγματικές εφαρμογές το σύνολο των δεδομένων δεν είναι γραμμικά διαχωρίσιμο, όπως στην παραπάνω περίπτωση. Εισάγοντας χαλαρές (slack) μεταβλητές ξ_i , $i = 1, 2, \dots, N$, ο περιορισμός $y_i(w^T x + b) \geq 1$ αναδιατυπώνεται σε

$$y_i(w^T x + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (1.20)$$



Εικόνα 4 - Μη γραμμικά διαχωρίσιμα δεδομένα

Στην ιδανική περίπτωση που όλες οι χαλαρές μεταβλητές είναι ίσες με μηδέν, υπάρχει πλήρης ισοδυναμία με την γραμμικά διαχωρίσιμη περίπτωση της προηγούμενης παραγράφου. Στην περίπτωση των μη γραμμικά ταξινομημένων στοιχείων, το πρόβλημα μπορεί να γραφεί ως εξής:

$$\text{Να βρεθούν τα } w \text{ και } b \text{ που ελαχιστοποιούν την } \frac{1}{2} \|w\|^2 + c \sum_i \xi_i^k \quad (1.21)$$

$$\text{με τη συνθήκη } y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i \quad (1.22)$$

όπου $c > 0$ είναι μία κατάλληλα επιλεγμένη παράμετρος.

Ο πρόσθετος όρος $c \sum_i \xi_i^k$ αναγκάζει όλες τις χαλαρές μεταβλητές, να έλθουν όσο το δυνατόν κοντύτερα στο μηδέν. Στην περίπτωση που $k = 0$, ο δεύτερος όρος της εξίσωσης (eq.1-21) μετράει τον αριθμό των λαθών κατάρτισης. Στην περίπτωση που $k = 2$, έχουμε μεγαλύτερη ευαισθησία στα απομακρυσμένα στοιχεία (outliers).

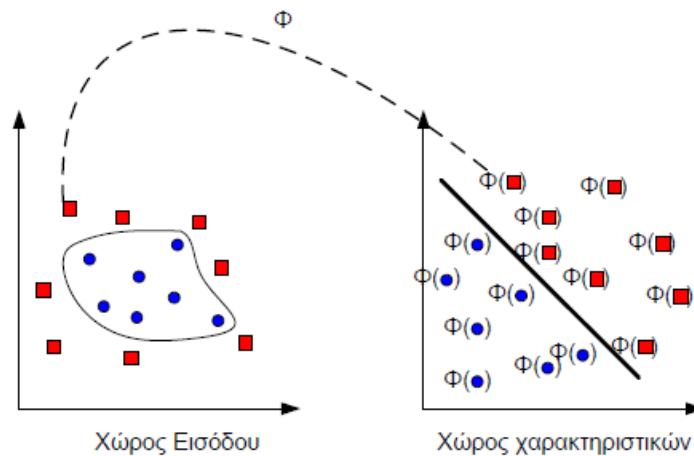
Το παραπάνω πρόβλημα βελτιστοποίησης μπορεί να μετατραπεί σε δυϊκό πρόβλημα, γράφοντάς το στην ακόλουθη μορφή:

$$\text{Να βρεθεί το } \alpha \text{ που ελαχιστοποιεί την } \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (1.23)$$

$$\text{με τις συνθήκες } \sum_{i=1}^N \alpha_i y_i = 0 \quad (1.24) \\ 0 \leq \alpha_i \leq c, \quad \forall i$$

Η εισαγωγή της παραμέτρου c έχει ως σκοπό τον περιορισμό της σειράς των αποδεκτών τιμών των πολλαπλασιαστών Lagrange α_i . Το c είναι αποτελεί μία παράμετρο κανονικοποίησης που ελέγχει την αλληλεπίδραση μεταξύ της μεγιστοποίησης του περιθωρίου και της ελαχιστοποίησης του όρου λάθους κατάρτισης. Το καταλληλότερο c εξαρτάται από το εκάστοτε σύνολο διαθέσιμων στοιχείων. Εάν είναι πολύ μικρός αριθμός, τότε χρειάζεται προσαρμογή στον διαχωρισμό των δεδομένων κατάρτισης. Στην αντίθετη περίπτωση, που είναι πολύ μεγάλο, τότε ο αλγόριθμος παρουσιάζει επικαλύψεις στα στοιχεία εκπαίδευσης. Το χρήσιμο εύρος του c βρίσκεται μεταξύ της τιμής για την οποία όλοι οι πολλαπλασιαστές Lagrange ισούνται με c και της τιμής για την οποία μόνο ένας πολλαπλασιαστής Lagrange έχει όριο το c . Μια απλή και πολύ διαδεδομένη μέθοδος προσδιορισμού του c είναι η μέθοδος της διασταύρωσης (cross-validation). Στην περίπτωση που $\alpha_i = c$, εάν $\xi_i > 1$ τότε τα σημεία είναι λάθος κατηγοριοποιημένα, εάν $0 < \xi_i \leq 1$ τότε τα σημεία είναι σωστά κατηγοριοποιημένα, αλλά σε απόσταση από το βέλτιστο υπερεπίπεδο μικρότερη από το μισό του περιθωρίου $\rho/2$, ή σε πολύ σπάνιες περιπτώσεις εάν $\xi_i = 0$ τότε τα στοιχεία είναι σωστά κατηγοριοποιημένα και πάνω στο περιθώριο. Σχεδόν σε κάθε περίπτωση, τα διανύσματα υποστήριξης για τα οποία $\alpha_i = c$ αναφέρονται ως σφάλματα.

Το σύνολο των δεδομένων εισόδου D πολλές φορές μπορεί να μην είναι γραμμικά διαχωρίσιμο στον χώρο εισόδου, αλλά να είναι γραμμικά διαχωρίσιμο σε έναν άλλο χώρο διαφορετικής (κατά κανόνα μεγαλύτερης) διάστασης. Βάση του θεωρήματος του Cover για το διαχωρισμό των δεδομένων (Cover, 1965) γίνεται ο μετασχηματισμός του χώρου εισόδου σε έναν νέο χώρο μεγαλύτερης διάστασης που το D' είναι γραμμικά διαχωρίσιμη και ονομάζεται χώρος χαρακτηριστικών. Στο νέο αυτό χώρο διαχωρίζονται οι «εικόνες» των προτύπων (δεδομένων εισόδου).



Εικόνα 5 - Μετασχηματισμός του χώρου εισόδου σε χώρο χαρακτηριστικών

Ο μετασχηματισμός γίνεται με τη βοήθεια μίας απεικόνισης $\Phi: \mathcal{R}^M \rightarrow F^N$ από τον αρχικό χώρο M -διαστάσεων σε έναν χώρο μεγαλύτερης διάστασης ιδιοτήτων F .

$$X = (x_1, \dots, x_p) \mapsto \phi(x) = (\phi_1, \dots, \phi_p) \quad (1.25)$$
$$F = \{\phi(x): x \in X\}$$

Με βάση τη συνάρτηση Φ (1.25) το δυϊκό πρόβλημα (1.23 – 1.24) μετασχηματίζεται σε

$$\text{Να βρεθεί } \alpha \text{ που να μεγιστοποιεί την } \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \Phi(x_i)^T \Phi(x_j) \quad (1.26)$$

$$\text{υπό συνθήκες } \begin{cases} \sum_{i=1}^N \alpha_i y_i = \mathbf{0} \\ \mathbf{0} \leq \alpha_i \leq c, \quad \forall i \end{cases} \quad (1.27)$$

Ο τελικός predictor μπορεί να εκφραστεί και ως

$$f(x) = w^T \Phi(x) + b = \sum_{i=1}^N \alpha_i y_i \Phi(x_i)^T \Phi(x) + b \quad (1.28)$$

Αρκεί λοιπόν να βρεθεί μια έκφραση για το εσωτερικό γινόμενο στο χώρο ιδιοτήτων (χαρακτηριστικών) που να χρησιμοποιεί μόνο τα σημεία του χώρου εισόδου. Δηλαδή:

$$\Phi(x_i) \Phi(y_j) = K(x_i, y_j) \quad (1.29)$$

Η συμμετρική συνάρτηση K ονομάζεται **πυρήνας (kernel)**.

Σύμφωνα με το θεώρημα Mercer, υπάρχει μια κατηγορία απεικόνισης Φ με την ακόλουθη ιδιότητα:

$$\Phi(x) \Phi(y) = K(x, y) \quad (1.30)$$

όπου K μία αντίστοιχη συνάρτηση kernel.

Το δυϊκό πρόβλημα (1.26 – 1.27) μπορεί πλέον να γραφεί ως

$$\text{Να βρεθεί } \alpha \text{ που να μεγιστοποιεί την } \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j K(x_i, y_j) \quad (1.31)$$

$$\text{με τις συνθήκες } \begin{cases} \sum_{i=1}^N \alpha_i y_i = \mathbf{0} \\ \mathbf{0} \leq \alpha_i \leq c, \quad \forall i \end{cases} \quad (1.32)$$

και αντίστοιχα ο SVM predictor είναι ο εξής:

$$f(x) = w^T \Phi(x_i) + b = \sum_{i=1}^N \alpha_i y_i K(x_i, y_j) + b \quad (1.33)$$

Μόνη απαίτηση για τον πυρήνα $K(x, x_i)$ είναι να ικανοποιεί το θεώρημα Mercer. Κατά συνέπεια υπάρχει αρκετή ελευθερία σχετικά με την επιλογή του. Στον Πίνακα 1 φαίνονται οι τρεις συνηθέστεροι πυρήνες που χρησιμοποιούνται σε SVM.

Τύπος του SVM δικτύου	Πυρήνας Εσωτερικού Γινομένου $K(x, x_i)$	Παρατηρήσεις
Γραμμική Πολυωνυμική μηχανή μάθησης	$x^T x_i$ $(x^T x_i + 1)^p$	Η δύναμη p καθορίζεται εκ των προτέρων
Δίκτυο Ακτινικών Συναρτήσεων Βάσης	$\exp\left(-\frac{\gamma}{2\sigma^2} \ x - x_i\ ^2\right)$	Το πλάτος σ^2 είναι κοινό για όλους τους πυρήνες και καθορίζεται εκ των προτέρων και $\gamma > 0$
Διεπίπεδο Perceptron	$\tanh(\beta_0 x^T x_i + \beta_1)$	Το θεώρημα του Mercer δεν ικανοποιείται για όλες τις τιμές των β_0 και β_1

Πίνακας 1 - Οι συνηθέστεροι πυρήνες εσωτερικού γινομένου σε SVM

2.2 Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis – PCA)

2.2.1.1 Γενική θεωρία

Σύμφωνα με τη μέθοδο αυτή τα δεδομένα περιγράφονται από έναν γραμμικό υπόχωρο, ο οποίος ορίζεται από τα ιδιοδιανύσματα της μήτρας συνδιασποράς των δεδομένων.

Η ανάλυση κύριων συνιστωσών PCA (για αναγνώριση εικόνας) εξετάζει τις εικόνες μέσα σε έναν υπόχωρο ο οποίος δημιουργείται από τα ιδιοδιανύσματα της μήτρας συνδιασποράς των δεδομένων/εικόνων εκπαίδευσης.

Το 1987 οι Sirovich και Kirby χρησιμοποίησαν ανάλυση κύριων συνιστωσών (PCA) για να επιτύχουν μία συμπίεσμένη αναπαράσταση ψηφιακών εικόνων προσώπου. Το 1991 οι Turk και

Pentland χρησιμοποίησαν προβολές PCA ως χαρακτηριστικά διανύσματα για να λύσουν το πρόβλημα της αναγνώρισης προσώπου, χρησιμοποιώντας ως μέτρο ομοιότητας την ευκλείδεια γεωμετρία. Το σύστημα αυτό, το οποίο ήταν η πρώτη προσέγγιση αναγνώρισης προσώπου βασισμένη σε υποχώρο, αργότερα ονομάστηκε μέθοδος με Ιδιοπρόσωπα (Eigenfaces) και αποτέλεσε την έμπνευση και αφετηρία πολλών αντίστοιχων συστημάτων που χρησιμοποιούν διαφορετικές μεθόδους προβολής και συναρτήσεις ομοιότητας.

Αν και ορισμένες λεπτομέρειες είναι πιθανό να διαφέρουν ανά περίπτωση, ο βασικός αλγόριθμος εντοπισμού εικόνων από την προβολή τους στον υπόχωρο είναι κοινός.

Αρχικά επιλέγεται ένας υπόχωρος στον οποίο θα προβληθούν οι εικόνες. Στη συνέχεια όλες οι εικόνες εκπαίδευσης προβάλλονται σε αυτόν. Κάθε εικόνα που λαμβάνεται ως πραγματική είσοδος προβάλλεται στον ίδιο υπόχωρο και συγκρίνεται με όλες τις εικόνες εκπαίδευσης με τη χρήση κάποιου μέτρου ομοιότητας ή απόστασης. Η εικόνα εκπαίδευσης που είναι όμοια, ή προσεγγίζει περισσότερο, την εικόνα ελέγχου χρησιμοποιείται για την αναγνώριση της εικόνας ελέγχου.

Κάθε εικόνα μπορεί να θεωρηθεί ως ένα διάνυσμα εικονοστοιχείων (pixels), όπου η τιμή της καταχώρισης στο διάνυσμα είναι η τιμή, στην κλίμακα του γκρι, του αντίστοιχου pixel. Μια εικόνα, για παράδειγμα, 4×4 pixels, μπορεί να αναδιαμορφωθεί και να αντιμετωπιστεί ως ένα διάνυσμα μήκους 16 στοιχείων. Η εικόνα υπάρχει σε έναν χώρο N – διαστάσεων, όπου N ο αριθμός των pixels, ήτοι το μήκος του διανύσματος. Αυτή η διανυσματική αναπαράσταση θεωρείται η αρχική διάσταση της εικόνας.

Ο Ιδιόχωρος (Eigenspace) υπολογίζεται προσδιορίζοντας τα ιδιοδιανύσματα της μήτρας συνδιασποράς ενός συνόλου εικόνων εκπαίδευσης. Τα ιδιοδιανύσματα που αντιστοιχούν σε μη μηδενικές ιδιότητες της μήτρας συνδιασποράς σχηματίζουν μία ορθοκανονική βάση που περιστρέφεται και/ή αντανακλά τις εικόνες στον N – διάστατο χώρο. Κάθε εικόνα αποθηκεύεται σε ένα διάνυσμα \mathbf{x} , μεγέθους N

$$\mathbf{x}^i = [x_1^i, \dots, x_N^i]^T \quad (2.1)$$

Οι εικόνες κανονικοποιούνται αφαιρώντας τη μέση τιμή κάθε στοιχείου από το σύνολο των στοιχείων εκπαίδευσης.

$$\bar{\mathbf{x}}^i = \mathbf{x}^i - m, \quad \text{όπου} \quad m = \frac{1}{P} \sum_{i=1}^P \mathbf{x}^i \quad (2.2)$$

Τα διανύσματα αυτά συνδυάζονται (δίπλα-δίπλα) για να δημιουργηθεί μία μήτρα δεδομένων μεγέθους $N \times P$, όπου P ο αριθμός των εικόνων.

$$\bar{\mathbf{X}} = [\bar{\mathbf{x}}^1 | \bar{\mathbf{x}}^2 | \dots | \bar{\mathbf{x}}^P] \quad (2.3)$$

Η μήτρα δεδομένων $\bar{\mathbf{X}}$ πολλαπλασιασμένη με την ανάστροφή της $\bar{\mathbf{X}}^T$ δίνει τελικά τη μήτρα συνδιασποράς.

$$\Omega = \bar{\mathbb{X}}\bar{\mathbb{X}}^T \quad (2.4)$$

Υποθέτοντας ότι $P < N$, η μήτρα συνδιασποράς έχει μέχρι και P ιδιοδιανύσματα συνδεδεμένα με μη μηδενικές ιδιοτιμές. Οι ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα υπολογίζονται για τη μήτρα συνδιασποράς.

$$\Omega V = \Lambda V \quad (2.5)$$

όπου V είναι η μήτρα των ιδιοδιανυσμάτων που συνδέονται με τις ιδιοτιμές που περιέχονται στο διαγώνιο πίνακα Λ .

Τα ιδιοδιανύσματα $v_i \in V$ αυτά ταξινομούνται από το μεγαλύτερο προς το μικρότερο, σύμφωνα με τις αντίστοιχες ιδιοτιμές τους $\lambda_i \in \Lambda$. Το ιδιοδιάνυσμα με τη μεγαλύτερη ιδιοτιμή είναι αυτό που βρίσκει τη μεγαλύτερη διακύμανση στις εικόνες, το διάνυσμα με τη δεύτερη μεγαλύτερη ιδιοτιμή βρίσκει τη δεύτερη μεγαλύτερη διακύμανση στις εικόνες, κ.ο.κ.

Η μήτρα των ιδιοδιανυσμάτων είναι ο ιδιόχωρος V , όπου κάθε στήλη της V είναι ένα ιδιοδιάνυσμα

$$V = [v_1 | v_2 | \dots | v_P] \quad (2.6)$$

Για να προβληθεί μία εικόνα στον ιδιόχωρο, υπολογίζεται το εσωτερικό γινόμενο του διανύσματος της εικόνας με κάθε ένα από τα διατεταγμένα ιδιοδιανύσματα.

$$\tilde{\mathbb{X}}^i = V^T \mathbb{X}^i \quad (2.7)$$

Κατά συνέπεια τα νέο διάνυσμα $\tilde{\mathbb{X}}^i$ θα περιέχει τόσες τιμές όσες και τα ιδιοδιανύσματα.

2.2.1.2 Αναγνώριση εικόνων ελέγχου

Για να γίνει η αναγνώριση, η εικόνα ελέγχου y^i κανονικοποιείται και εν συνεχεία προβάλλεται στον ιδιόχωρο που ορίζεται από τη μήτρα ιδιοδιανυσμάτων V

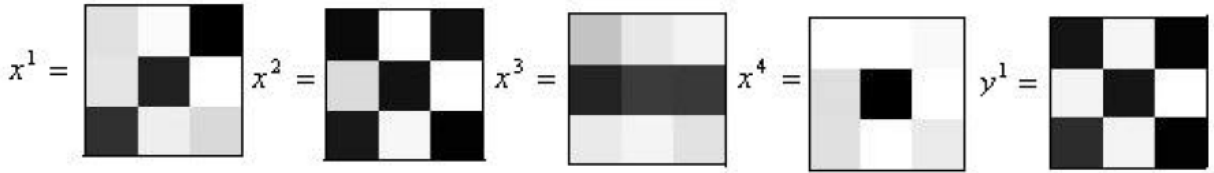
$$\bar{y}^i = y^i - m, \quad \text{όπου} \quad m = \frac{1}{P} \sum_{i=1}^P \mathbb{X}^i \quad (2.8)$$

$$\hat{y}^i = V^T \bar{y}^i \quad (2.9)$$

Η εικόνα δοκιμής συγκρίνεται με κάθε μία από της εικόνες εκπαίδευσης και όποια από τις τελευταίες προσεγγίζει περισσότερο την εικόνα δοκιμής, χρησιμοποιείται για την αναγνώριση. Για τη σύγκριση μπορεί να χρησιμοποιηθεί οποιοδήποτε μέτρο ομοιότητας, με πιο κοινό την L_2 νόρμα.

2.2.1.3 Παράδειγμα

Έστω οι εικόνες εκπαίδευσης x^1, \dots, x^4 και η εικόνα δοκιμής y^1 της εικόνας 6.



Εικόνα 6 - Οι εικόνες εκπαίδευσης (x) και η εικόνα δοκιμής (y).

Τα διανύσματα των εικόνων εκπαίδευσης και η μέση εικόνα είναι

$$\mathbf{x}^1 = \begin{bmatrix} 225 \\ 229 \\ 48 \\ 251 \\ 33 \\ 238 \\ 0 \\ 255 \\ 217 \end{bmatrix} \quad \mathbf{x}^2 = \begin{bmatrix} 10 \\ 219 \\ 24 \\ 255 \\ 18 \\ 247 \\ 17 \\ 255 \\ 2 \end{bmatrix} \quad \mathbf{x}^3 = \begin{bmatrix} 196 \\ 35 \\ 234 \\ 232 \\ 59 \\ 244 \\ 243 \\ 57 \\ 226 \end{bmatrix} \quad \mathbf{x}^4 = \begin{bmatrix} 255 \\ 223 \\ 224 \\ 255 \\ 0 \\ 255 \\ 249 \\ 255 \\ 235 \end{bmatrix} \quad \mathbf{m} = \begin{bmatrix} 171.5 \\ 176.5 \\ 135.5 \\ 248.25 \\ 27.5 \\ 246 \\ 127.25 \\ 205.5 \\ 170 \end{bmatrix}$$

Οι κανονικοποιημένες εικόνες μετά την αφαίρεση της μέσης τιμής είναι

$$\bar{\mathbf{x}}^1 = \begin{bmatrix} 53.5 \\ 52.5 \\ -84.5 \\ 2.75 \\ 5.5 \\ -8 \\ 127.25 \\ 49.5 \\ 47 \end{bmatrix} \quad \bar{\mathbf{x}}^2 = \begin{bmatrix} -161.5 \\ 42.5 \\ -108.5 \\ 6.75 \\ -9.5 \\ 1 \\ -110.25 \\ 49.5 \\ -168 \end{bmatrix} \quad \bar{\mathbf{x}}^3 = \begin{bmatrix} 24.5 \\ -141.5 \\ 101.5 \\ -16.25 \\ 31.5 \\ -2 \\ 115.75 \\ -148.5 \\ 56 \end{bmatrix} \quad \bar{\mathbf{x}}^4 = \begin{bmatrix} 83.5 \\ 46.5 \\ 91.5 \\ 6.75 \\ -27.5 \\ 9 \\ 121.75 \\ 49.5 \\ 65 \end{bmatrix}$$

Συνδυάζοντας όλες τις κανονικοποιημένες εικόνες προκύπτει η μήτρα δεδομένων

$$\bar{\mathbf{X}} = \begin{bmatrix} 53.5 & -161.5 & 24.5 & 83.5 \\ 52.5 & 42.5 & -141.5 & 46.5 \\ -84.5 & -108.5 & 101.5 & 91.5 \\ 2.75 & 6.75 & -16.25 & 6.75 \\ 5.5 & -9.5 & 31.5 & -27.5 \\ -8 & 1 & -2 & 9 \\ -127.25 & -110.25 & 115.75 & 121.75 \\ 49.5 & 49.5 & -148.5 & 49.5 \\ 47 & -168 & 56 & 65 \end{bmatrix}$$

και εν συνεχεία η μήτρα συνδιασποράς

$$\mathbf{\Omega} = \bar{\mathbf{X}}\bar{\mathbf{X}}^T = \begin{bmatrix} 36517 & -3639 & 23129 & -778 & 304 & 113 & 24000 & -4851 & 36446 \\ -3639 & 26747 & -19155 & 3045 & -5851 & 324 & -22083 & 28017 & -9574 \\ 23129 & -19155 & 37587 & -1997 & 1247 & 1188 & 45603 & -0097 & 25888 \\ -778 & 3045 & -1996 & 363 & -746.5 & 78 & -2153 & 3217 & -1476 \\ 304 & -5851 & 1247 & -747 & 1869 & -364 & 645 & -6237 & 1831 \\ 113 & 324 & 1188 & 78 & -364 & 150 & 1772 & 396 & -71 \\ 24000 & -22083 & 45603 & -2153 & 645.5 & 1772 & 56569 & -22919 & 26937 \\ -4851 & 28017 & -20097 & 3218 & -6237 & 396 & -22919 & 29403 & -11088 \\ 36446 & -9574 & 25888 & -1476 & 1831 & -71 & 26937 & -11088 & 37794 \end{bmatrix}$$

Τα μη μηδενικά ιδιοδιανύσματα της μήτρας συνδιασποράς και οι αντίστοιχες ιδιοτιμές

$$\mathbf{v}_1 = \begin{bmatrix} 0.356 \\ -0.279 \\ 0.480 \\ -0.031 \\ 0.035 \\ 0.009 \\ 0.560 \\ -0.296 \\ 0.402 \end{bmatrix} \quad \mathbf{v}_2 = \begin{bmatrix} -0.052 \\ -0.489 \\ 0.044 \\ -0.048 \\ 0.105 \\ -0.004 \\ 0.112 \\ 0.492 \\ -0.432 \end{bmatrix} \quad \mathbf{v}_3 = \begin{bmatrix} -0.264 \\ 0.347 \\ 0.309 \\ 0.064 \\ -0.222 \\ 0.078 \\ 0.585 \\ 0.401 \\ -0.391 \end{bmatrix}$$

$$\lambda_1 = 153520 \quad \lambda_2 = 50696 \quad \lambda_3 = 22781$$

Η μήτρα προβολής (από την οποία ορίζεται ο ιδιόχωρος)

$$\mathbf{V} = \begin{bmatrix} 0.356 & -0.052 & -0.264 \\ -0.279 & -0.489 & 0.347 \\ 0.480 & 0.044 & 0.309 \\ -0.031 & -0.048 & 0.064 \\ 0.035 & 0.105 & -0.222 \\ 0.009 & -0.004 & 0.078 \\ 0.560 & 0.112 & 0.585 \\ -0.296 & 0.492 & 0.401 \\ 0.402 & -0.432 & -0.391 \end{bmatrix}$$

Και οι τέσσερις κανονικοποιημένες εικόνες που προβάλλονται στον ιδιοχώρο

$$\bar{\mathbf{x}}^1 = \mathbf{V}^T \mathbf{x}^1 = \begin{bmatrix} -103.09 \\ -117.31 \\ -96.57 \end{bmatrix} \quad \bar{\mathbf{x}}^2 = \mathbf{V}^T \mathbf{x}^2 = \begin{bmatrix} -265.92 \\ 98.29 \\ 47.45 \end{bmatrix}$$

$$\bar{\mathbf{x}}^3 = \mathbf{V}^T \mathbf{x}^3 = \begin{bmatrix} 229.76 \\ 125.9 \\ -46.14 \end{bmatrix} \quad \bar{\mathbf{x}}^4 = \mathbf{V}^T \mathbf{x}^4 = \begin{bmatrix} 139.24 \\ -106.88 \\ 95.26 \end{bmatrix}$$

Η εικόνα ελέγχου ως διάνυσμα, πριν και μετά την κανονικοποίηση

$$\mathbf{y}^1 = \begin{bmatrix} 20 \\ 244 \\ 44 \\ 246 \\ 21 \\ 244 \\ 4 \\ 255 \\ 2 \end{bmatrix} \quad \bar{\mathbf{y}}^1 = \begin{bmatrix} -151.5 \\ 67.5 \\ -88.5 \\ -2.25 \\ -6.5 \\ -2 \\ -123.25 \\ 49.5 \\ -168 \end{bmatrix}$$

Η προβαλλόμενη εικόνα ελέγχου είναι

$$\tilde{\mathbf{y}}^1 = \mathbf{V}^T \bar{\mathbf{y}}^1 = \begin{bmatrix} -266.65 \\ 80.75 \\ 50.6 \end{bmatrix}$$

Χρησιμοποιώντας την L_2 νόρμα (L_2 norm), οι αποστάσεις των εικόνων x_1, x_2, x_3, x_4 από την εικόνα ελέγχου είναι 296, 18, 508 και 449, αντίστοιχα. Μικρότερη είναι η απόσταση της εικόνας x_2 και κατά συνέπεια, η εικόνα ελέγχου αναγνωρίστηκε ότι ανήκει στην ίδια τάξη εικόνων με τη x_2 .

2.3 Ασαφείς ταξινομητές (fuzzy classifiers)

Κάθε ταξινομητής που χρησιμοποιεί ασαφή σύνολα ή ασαφή λογική κατά τη διαδικασία εκπαίδευσης ή λειτουργίας του, χαρακτηρίζεται ως ασαφής (fuzzy) ταξινομητής. Αυτού του τύπου οι ταξινομητές συνήθως χρησιμοποιούνται για να παράγουν στην έξοδό τους μεταγενέστερες πιθανότητες και/ή να προσφέρουν soft labeling. Μπορούμε να αντιληφθούμε έναν ασαφή (fuzzy) ταξινομητή D που παράγει ασαφείς ετικέτες ως μία προσέγγιση συνάρτησης $D: F \rightarrow [0,1]^c$, όπου F ο χώρος χαρακτηριστικών και c ο αριθμός των κλάσεων.

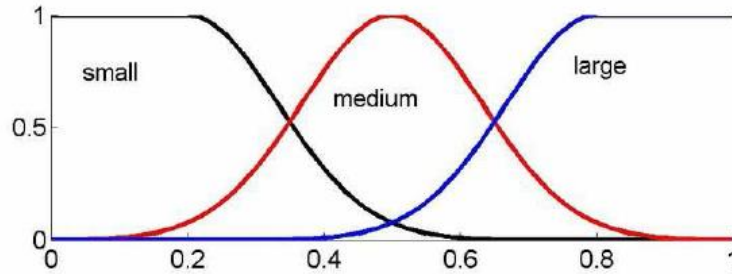
2.3.1 Fuzzy rule-based classifiers

Πρόκειται για ένα IF-THEN σύστημα (fuzzy IF-THEN system), καθ' όλα παρόμοια με αυτά που χρησιμοποιούνται στον ασαφή έλεγχο. Για τη διευκόλυνση της κατανόησης της λειτουργίας ενός fuzzy rule-based ταξινομητή θα χρησιμοποιηθεί ένα παράδειγμα με 2 διαστάσεις και 3 κλάσεις.

Ένας ασαφής ταξινομητής κατασκευάζεται προσδιορίζοντας αρχικά τους κανόνες ταξινόμησης:

$$\begin{aligned} &\text{IF } x_1 \text{ is } \textit{medium} \text{ AND } x_2 \text{ is } \textit{small} \text{ THEN class is } 1 \\ &\text{IF } x_1 \text{ is } \textit{medium} \text{ AND } x_2 \text{ is } \textit{large} \text{ THEN class is } 2 \\ &\text{IF } x_1 \text{ is } \textit{large} \text{ AND } x_2 \text{ is } \textit{small} \text{ THEN class is } 2 \\ &\text{IF } x_1 \text{ is } \textit{small} \text{ AND } x_2 \text{ is } \textit{large} \text{ THEN class is } 3 \end{aligned} \quad (3.1)$$

Τα χαρακτηριστικά x_1 και x_2 είναι αριθμητικά, αλλά οι κανόνες χρησιμοποιούν γλωσσικές τιμές για κάθε χαρακτηριστικό. Έτσι, αν ο αριθμός των χαρακτηριστικών είναι n και των εννοιών (π.χ. small, medium, large) είναι M , ο αριθμός των πιθανών IF-THEN κανόνων αυτού του τύπου θα είναι M^n . Κάθε γλωσσική τιμή αντιπροσωπεύεται από μία συνάρτηση συμμετοχής.



Εικόνα 7 - Συναρτήσεις συμμετοχής για τους γλωσσικούς όρους του x_1

Για κάθε δοθέν ζεύγος $x = (x_1, x_2)$ ο βαθμός ικανοποίησης του τμήματος υπόθεσης του κάθε κανόνα είναι αυτός που καθορίζει τη δύναμη πυροδότησης του κανόνα. Έτσι, η δύναμη πυροδότησης του κανόνα 1 υπολογίζεται ως εξής:

$$\tau_1(x) = \mu_{medium}^{(1)}(x_1) \text{ AND } \mu_{small}^{(2)}(x_2) \quad (3.2)$$

Η λειτουργία *AND* τυπικά υλοποιείται ως ελάχιστο (minimum). Ο κανόνας «ψηφίζει» για την κλάση του επί μέρους συμπεράσματος. Το βάρος της «ψήφου» αυτής είναι $\tau_1(x)$. Για να βρεθεί η έξοδος του ταξινομητή, οι ψήφοι από όλους τους κανόνες συνυπολογίζονται (με τη χρήση μεθόδων όπως η maximum aggregation method).

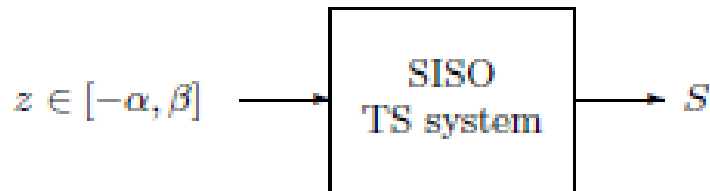
2.3.2 A Fuzzy prototype-based classifier

Πρόκειται για ασαφείς (fuzzy) ταξινομητές που κατασκευάζονται πάνω στην ιδέα της «ασαφοποίησης» συμβατικών ταξινομητών. Ένας ευρύτατα χρησιμοποιούμενος αντιπρόσωπος αυτής της κατηγορίας είναι ο ταξινομητής K – κοντινότερου γείτονα (K – nearest neighbor classifier / K – nn). Στον κλασικό K – nn , το αντικείμενο x χαρακτηρίζεται με βάση την πλειοψηφία των K κοντινότερων γειτόνων του, σε ένα σύνολο δεδομένων αναφοράς. Ο ασαφής K – nn χρησιμοποιεί τις αποστάσεις των γειτόνων, καθώς και τις ασαφείς ετικέτες τους. Για τον ταξινομητή αυτόν, το σύνολο αναφοράς δεν είναι απαραίτητο να επιλεγεί από τα υπάρχοντα δεδομένα. Ένα σύνολο από σχετικά πρότυπα (αντικείμενα) με σαφείς ή ασαφείς ετικέτες μπορεί να κατασκευαστεί και να χρησιμοποιηθεί. Η κλάση του x λαμβάνεται συνδυάζοντας τις ομοιότητες μεταξύ του x και των προτύπων.

2.4 Takagi – Sugeno fuzzy systems (TS model)

Τα μοντέλα Takagi – Sugeno (TS) για fuzzy συστήματα χρησιμοποιούν μη γραμμικές ή πολυωνυμικές συναρτήσεις συμμετοχής. Στην απλή τους μορφή είναι συστήματα απλής εισόδου – απλής εξόδου (single-input and single-output systems / SISO TS) που χρησιμοποιούν μη γραμμικές συναρτήσεις συμμετοχής. Το πρόβλημα συνίσταται στον προσδιορισμό των ασαφών κανόνων (fuzzy rules) που «μοντελοποιούν»/προσεγγίζουν επακριβώς μια μη γραμμική συνάρτηση.

Θεωρούμε την είσοδο $z \in [-\alpha, \beta]$ και την έξοδο S ενός TS συστήματος.



Εικόνα 8 - Single-input-single-output TS system

Υποθέτουμε ότι $\alpha + \beta \neq 0$. Με N και P συμβολίζουμε δύο ασαφή (fuzzy) σύνολα τα οποία θα προσδιορίζονται τόσο με τις γλωσσικές ετικέτες τους, όσο και με τις συναρτήσεις συμμετοχής $N(z)$ και $P(z)$ αντίστοιχα. Έτσι, $N, P: [-\alpha, \beta] \rightarrow [0,1]$. Το TS σύστημα καθορίζεται από δύο fuzzy κανόνες:

$$\left. \begin{array}{l} R_1: \text{If } z \text{ is } N, \text{ then } S = q_1, \\ R_2: \text{If } z \text{ is } P, \text{ then } S = q_2. \end{array} \right\} \quad (4.1)$$

Οι απαιτήσεις για τα fuzzy σύνολα είναι οι

1. Η $N(z)$ είναι συνεχής, μη αύξουσα συνάρτηση του z .
2. $N(-\alpha) = 1$ και $N(\beta) = 0$
3. $P(z) = 1 - N(z)$

Παρατηρούμε ότι η P είναι μία συνεχής, αύξουσα συνάρτηση του z που ικανοποιεί τις οριακές συνθήκες $P(\beta) = 1$ και $P(-\alpha) = 0$. Η συνέχεια, η μονοτονία και η ικανοποίηση των οριακών συνθηκών εξασφαλίζουν μία σαφή γλωσσική ερμηνεία των δύο συναρτήσεων συμμετοχής.

Θεωρούμε δοθείσα συνεχή και μονοτονική συνάρτηση $f(z): [-\alpha, \beta] \rightarrow \mathbb{R}$. Το πρόβλημα που προκύπτει είναι «Πώς να παραχθούν συναρτήσεις συμμετοχής για το fuzzy rule-based TS σύστημα, τέτοιες ώστε η έξοδος να είναι ακριβώς η ίδια, δηλαδή $S(z) = f(z) \quad \forall z \in [-\alpha, \beta]$ ».

Κατ' αρχάς πρέπει να ικανοποιούνται οι ακόλουθες συνθήκες

$$q_1 = f(-\alpha), \quad q_2 = f(\beta) \quad (4.2)$$

$$N(z) = \frac{f(z) - f(\beta)}{f(-\alpha) - f(\beta)} \quad (4.3)$$

δεδομένου ότι η έξοδος του TS συστήματος υπολογίζεται ως εξής

$$S(z) = \frac{q_1 N(z) + q_2 P(z)}{N(z) + P(z)}, \quad \text{για } z \in [-\alpha, \beta].$$

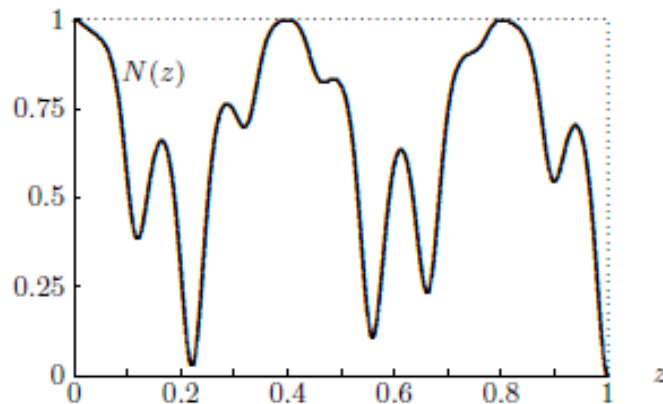
Έστω λοιπόν η συνεχής, ομαλή και εξαιρετικά μη γραμμικά συνάρτηση

$$f(z) = e^\pi - (e^\pi - \pi^e) \sin^2\left(\frac{5\pi z}{2}\right) \exp(-\sin^2(9\pi z)), \quad z \in [0,1] \quad (4.4)$$

Οι ασαφείς (fuzzy) κανόνες της μορφής (4.1) και τα ασαφή σύνολα μπορούν να βρεθούν, έτσι ώστε $S(z) = f(z)$ για $z \in [0,1]$. Από τους fuzzy κανόνες προκύπτουν οι σταθερές $q_1 = e^\pi$, $q_2 = \pi^e$ και οι συναρτήσεις συμμετοχής των fuzzy συνόλων N και P ικανοποιούν τις οριακές συνθήκες ($P(0) = 0$, $P(1) = 1$, $N(0) = 1$, $N(1) = 0$). Οι συναρτήσεις συμμετοχής λοιπόν είναι οι

$$N(z) = 1 - P(z), \quad P(z) = \sin^2\left(\frac{5\pi z}{2}\right) \exp(-\sin^2(9\pi z)), \quad z \in [0,1] \quad (4.5)$$

Αν και τα ασαφή (fuzzy) σύνολα δεν είναι εύκολο να ερμηνευτούν στην παρούσα περίπτωση (όπως φαίνεται στην εικόνα 9), η έξοδος S του TS συστήματος είναι ακριβώς η ίδια με τη συνάρτηση (4.4) για όλα τα σημεία και οι συναρτήσεις συμμετοχής ικανοποιούν τις οριακές συνθήκες.



Εικόνα 9 - Γραφική παράσταση της συνάρτησης συμμετοχής $N(z)$, όπως προέκυψε από την εξίσωση (4.5)

2.5 Μέθοδοι Ανίχνευσης

2.5.1 OpenCV (Open Source Computer Vision Library)

Η βιβλιοθήκη *Open Source Computer Vision Library* (OpenCV) είναι μία συλλογή ανοιχτού κώδικα από ρουτίνες σε C++, που αφορούν την τεχνητή όραση. Με BSD (Berkeley Software Distribution) άδεια χρήσης, παρέχεται για εμπορική ή ερευνητική χρήση ελεύθερα από την Intel. Χρησιμοποιεί απλά Haar-like χαρακτηριστικά και έναν «καταρράκτη» ενισχυμένων δένδρων ταξινομητών σαν στατιστικό της μοντέλο.

Η OpenCV παρέχει χαμηλού και υψηλού επιπέδου API για την Ανίχνευση Προσώπων ή άλλων Αντικειμένων, ανάλογα με την εκπαίδευση που έχει προηγηθεί στον ταξινομητή αντικειμένων που προσφέρει. Για την εκπαίδευση του ταξινομητή παρέχει την εφαρμογή *Haartraining* με ένα σύνολο θετικών και αρνητικών δειγμάτων, που αφορούν συγκεκριμένο αντικείμενο. Η εφαρμογή αυτή δημιουργεί ένα *xml* αρχείο που περιέχει τα δεδομένα που αφορούν τον εν λόγω ταξινομητή και μπορεί να χρησιμοποιηθεί στη συνέχεια για την ανίχνευση του συγκεκριμένου αντικειμένου.

Αφού εκπαιδευτεί ο ταξινομητής, μπορεί να εφαρμοστεί σε μία περιοχή ενδιαφέροντος μίας εικόνας εισόδου. Ο ταξινομητής δίνει "1" αν η περιοχή περιέχει το αναζητούμενο αντικείμενο και με "0" σε αντίθετη περίπτωση. Για να αναζητηθεί το αντικείμενο σε ολόκληρη την εικόνα, το παράθυρο αναζήτησης μπορεί να μετακινηθεί σε διαφορετικές θέσεις και να ελεγχθεί η απόκριση του ταξινομητή. Ο ταξινομητής έχει σχεδιαστεί έτσι ώστε το παράθυρο αναζήτησης να μπορεί να τροποποιηθεί ως προς τις διαστάσεις του εύκολα, για να ανευρεθούν αντικείμενα διαφορετικών διαστάσεων, αντί να απαιτείται αντίστοιχη μεταβολή των διαστάσεων της ίδιας της εικόνας. Έτσι, για να βρεθεί ένα αντικείμενο άγνωστων διαστάσεων σε μία εικόνα, η διαδικασία αναζήτησης μπορεί να γίνει όσες φορές χρειάζεται σε διαφορετικές κλίμακες.

Ο τελικός ταξινομητής αποτελείται από επίπεδα απλούστερων ταξινομητών σε διάταξη καταρράκτη (*cascade*). Κάθε επίπεδο του καταρράκτη αποτελείται από ένα ισχυρό ταξινομητή (*strong classifier*) που κατασκευάζεται από απλούστερους αδύναμους ταξινομητές (*weak classifiers*) με την τεχνική της ενίσχυσης (*boosting*). Οι τεχνικές ενίσχυσης που υποστηρίζονται είναι η *Discrete AdaBoost*, η *Real AdaBoost*, η *Gentle AdaBoost* και η *LogitBoost*. Οι αδύναμοι ταξινομητές που χρησιμοποιούνται είναι δένδρα απόφασης (*decision trees*) με 1 έως 4 κόμβους. Οι αδύναμοι ταξινομητές παίρνουν ως είσοδο ορθογώνια χαρακτηριστικά (*features*) τύπου Haar, που καθορίζονται από την θέση, τις διαστάσεις, τον προσανατολισμό και την μορφή τους και παίρνουν τιμές μία σταθμισμένη διαφορά των εντάσεων των εικονοστοιχείων στις περιοχές που ορίζουν. Τα αθροίσματα των εντάσεων υπολογίζονται με τη χρήση της εικόνας ολοκλήρωμα (*integral image*) που έχει προϋπολογιστεί.

Μία υψηλού επιπέδου συνάρτηση, όπως η *cvHaarDetectObjects* υλοποιεί όλα τα προηγούμενα, χρησιμοποιώντας τον ταξινομητή που είναι αποθηκευμένος σε *xml* αρχείο, παρέχοντας αυτόματα τα αποτελέσματα ανίχνευσης στο χρήστη.

Η συνάρτηση *cvHaarDetectObjects* βρίσκει ορθογώνιες περιοχές σε μια δεδομένη εικόνα που είναι πιθανό να περιέχουν αντικείμενα που ο ταξινομητής έχει εκπαιδευτεί να διακρίνει και επιστρέφει αυτές τις περιοχές σαν μία σειρά από ορθογώνια. Η συνάρτηση σαρώνει την εικόνα αρκετές φορές σε διαφορετικές κλίμακες, και κάθε φορά, για κάθε περιοχή, εφαρμόζει τον ταξινομητή για να ελέγξει την ύπαρξη του προς ανίχνευση αντικειμένου.

2.5.2 SIFT (Μετασχηματισμός Κλιμακωτά Αναλλοίωτων Χαρακτηριστικών)

Ο μετασχηματισμός κλιμακωτά αναλλοίωτων χαρακτηριστικών (SIFT), είναι ένας αλγόριθμος τεχνητής όρασης για τον εντοπισμό και την περιγραφή τοπικών χαρακτηριστικών σε εικόνες. Ο αλγόριθμος δόθηκε στην δημοσιότητα το 1999 από τον David Lowe και έχει κατοχυρωμένο δίπλωμα ευρεσιτεχνίας στις ΗΠΑ υπό την ιδιοκτησία του Πανεπιστημίου της Βρετανικής Κολούμπια, παρέχεται όμως για χρήση σε συστήματα ερευνητικού περιεχομένου. Οι εφαρμογές στις οποίες έχει χρησιμοποιηθεί είναι αναγνώριση αντικειμένων, παραγωγή χαρτών και πλοήγησης ρομπότ, “ράψιμο” εικόνων, παραγωγή τρισδιάστατων μοντέλων, αναγνώριση χειρονομιών, ανίχνευση αντικειμένων σε βίντεο και χρήση εφέ σε βίντεο.

Για κάθε αντικείμενο στην εικόνα, μπορούν να εξαχθούν σημεία ενδιαφέροντος σχετικά με το αντικείμενο, για να παρέχουν τη λεγόμενη “περιγραφή χαρακτηριστικών” (feature description). Αυτή η περιγραφή η οποία προέρχεται από μια εικόνα εκπαίδευσης, μπορεί στη συνέχεια να χρησιμοποιηθεί για τον εντοπισμό του αντικειμένου, σε μια δοκιμαστική εικόνα η οποία περιέχει πολλά άλλα αντικείμενα. Για να εκτελεστεί μια αξιόπιστη αναγνώριση, είναι σημαντικό τα χαρακτηριστικά που εξάγονται από την εικόνα εκπαίδευσης να ανιχνεύονται ακόμα και μετά από αλλαγές στην κλίμακα της εικόνας, πρόσθεσης θορύβου και αλλαγών φωτισμού. Αυτά τα σημεία βρίσκονται συνήθως σε περιοχές με μεγάλες αντιθέσεις μέσα στην εικόνα, τις λεγόμενες ακμές. Μια άλλη σημαντική ιδιότητα αυτών των χαρακτηριστικών είναι ότι οι σχετικές θέσεις μεταξύ τους στην αρχική σκηνή δεν πρέπει να αλλάζουν από τη μια εικόνα στην άλλη. Ωστόσο στην πράξη ο αλγόριθμος εντοπίζει και χρησιμοποιεί ένα πολύ μεγαλύτερο αριθμό χαρακτηριστικών από τις εικόνες, μειώνοντας έτσι τη συμβολή των σφαλμάτων που προκαλούνται από τις τοπικές διακυμάνσεις του μέσου σφάλματος από όλα τα λάθη ταιριάσματος στα χαρακτηριστικά.

Τα κύρια σημεία των αντικειμένων αρχικά εξάγονται από ένα σύνολο εικόνων αναφοράς και αποθηκεύονται σε μια βάση δεδομένων. Ένα αντικείμενο αναγνωρίζεται σε μια καινούργια εικόνα συγκρίνοντας ατομικά κάθε στοιχείο από την νέα εικόνα με αυτά από τη βάση δεδομένων, βρίσκοντας υποψήφια χαρακτηριστικά τα οποία ταιριάζουν με βάση την Ευκλείδεια απόσταση με τα διανύσματα

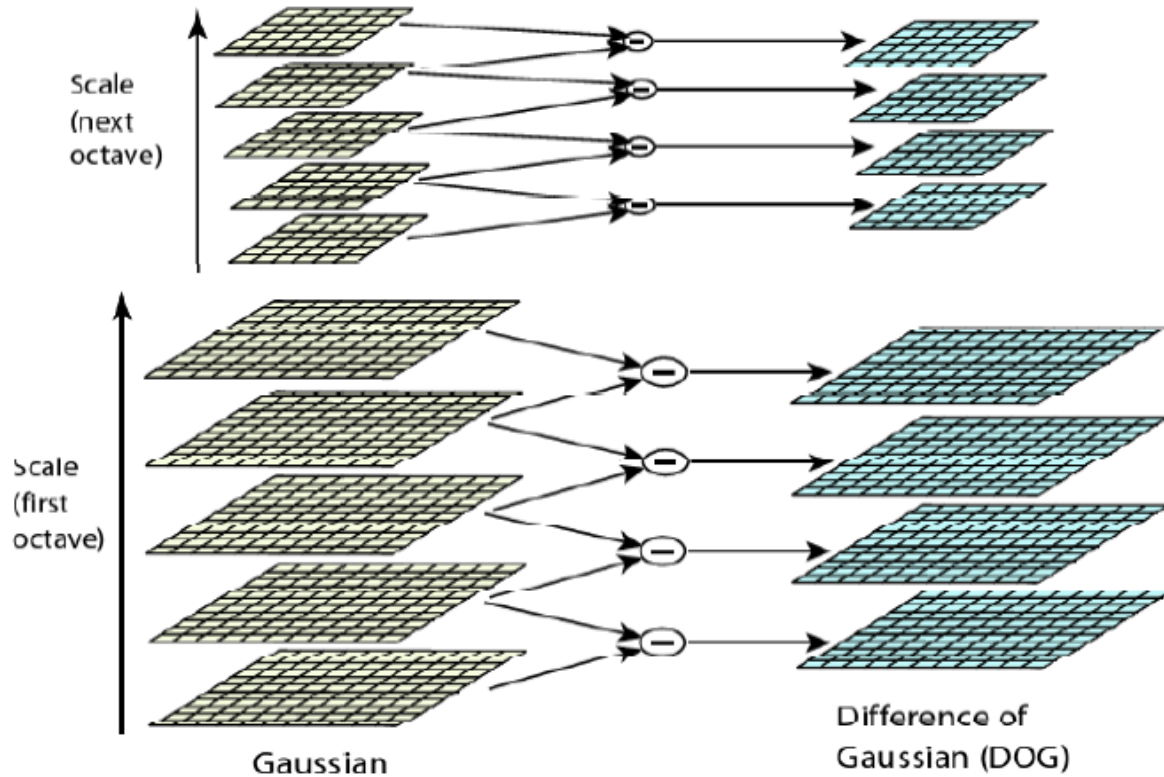
χαρακτηριστικών της εικόνας. Στη συνέχεια από το σύνολο των στοιχείων που ταιριάζουν, τα υποσύνολα των κύριων σημείων που συμφωνούν με το αντικείμενο και τη θέση του, την κλίμακα και τον προσανατολισμό στη νέα εικόνα, εντοπίζονται και φιλτράρονται. Ο προσδιορισμός των σταθερών συμπλεγμάτων γίνεται με μια αποτελεσματική εφαρμογή του πίνακα κατακερματισμού (hash table) του γενικευμένου μετασχηματισμού Hough. Κάθε σύμπλεγμα τριών ή περισσότερων χαρακτηριστικών τα οποία συμφωνούν σε ένα αντικείμενο υπόκειται σε περαιτέρω λεπτομερή έλεγχο και στη συνέχεια οι ακραίες τιμές απορρίπτονται.

Τέλος η πιθανότητα ένα συγκεκριμένο σύνολο χαρακτηριστικών να υποδηλώνει την παρουσία ενός αντικειμένου, υπολογίζεται δεδομένης της ακρίβειας του ταιριάσματος και του αριθμού των πιθανών εσφαλμένων ταιριασμάτων. Ταιριάσματα μεταξύ αντικειμένων που έχουν περάσει όλα τα παραπάνω στάδια, αναγνωρίζονται ως ορθά με μεγάλη αξιοπιστία.

Ανίχνευση αμετάβλητων σε κλίμακα χαρακτηριστικών

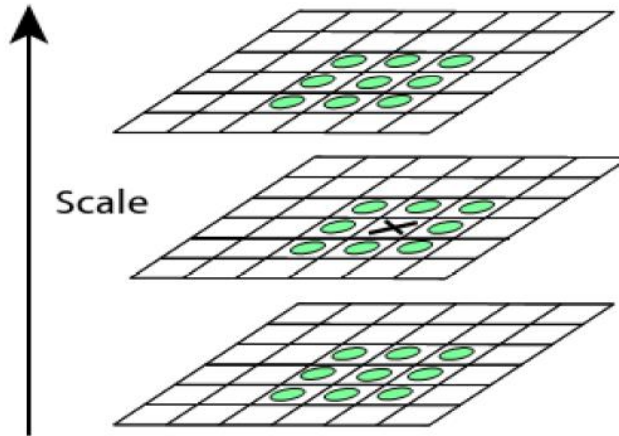
Η μέθοδος του Lowe για την εξαγωγή των χαρακτηριστικών μίας εικόνας μετατρέπει την εικόνα σε μια μεγάλη συλλογή από διανύσματα χαρακτηριστικών γνωρισμάτων κάθε ένα από τα οποία είναι αναλλοίωτο στην μετατόπιση, στην κλιμάκωση, στην περιστροφή, εν μέρει αμετάβλητο στις αλλαγές φωτισμού και ανθεκτικό στις τοπικές γεωμετρικές παραμορφώσεις. Τα χαρακτηριστικά αυτά έχουν παρόμοιες ιδιότητες με τους νευρώνες στο κάτω κροταφικό φλοιό που είναι υπεύθυνος για την αναγνώριση αντικειμένων στην πρωτεύουσα όραση. Υποψήφια σημεία χαμηλής αντίθεσης και σημεία απόκρισης από άκρη σε άκρη απορρίπτονται. Οι κυρίαρχες κατευθύνσεις ανατίθενται σε εντοπισμένα κύρια σημεία (keypoints). Αυτά τα βήματα διασφαλίζουν ότι τα κύρια σημεία είναι σταθερά για αντιστοίχιση και αναγνώριση. Οι SIFT περιγραφείς είναι ανθεκτικοί στις τοπικά συσχετισμένες στρεβλώσεις και προκύπτουν λαμβάνοντας υπόψη τα pixels γύρω από την ακτίνα μιας νευραλγικής θέσης, φιλτράροντας και παίρνοντας δείγματα των τοπικών επιπέδων προσανατολισμού της εικόνας.

Μία συνάρτηση Gaussian διαφοράς είναι η $G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2(L_{xx} + L_{yy})$ όπου το $(k - 1)$ θεωρείται σταθερά και μπορεί να παραληφθεί.



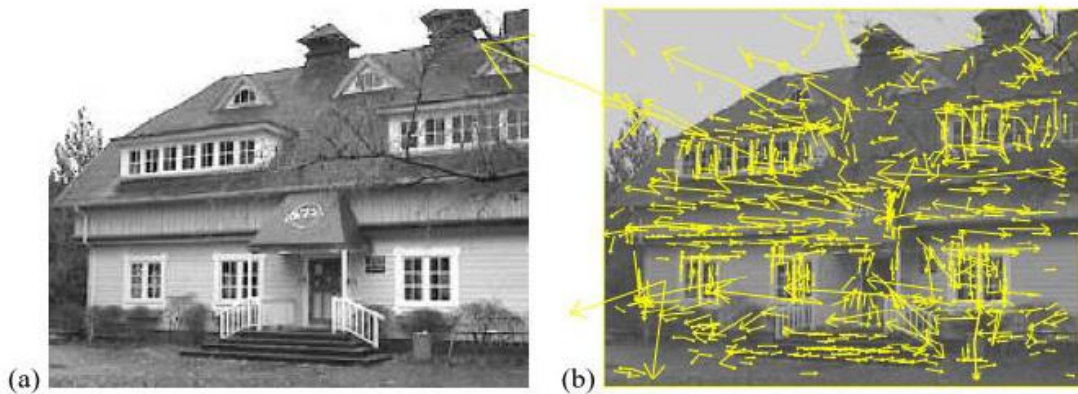
Εικόνα 10 - Αναπαράσταση συνέλιξης & αφαίρεσης κάθε οκτάβας για τη δημιουργία της Gaussian διαφοράς

Κάθε pixel στην Gaussian διαφορά συγκρίνεται με τους γείτονες του, στην δικιά του κλίμακα και στην κλίμακα των διπλανών του pixels. Ένα pixel επιλέγεται μόνο αν είναι πιο σκούρο ή πιο φωτεινό από όλους τους γείτονες του. Δεν είναι όλα τα ακρότατα χρήσιμα στην κλίμακα του χώρου. Ειδικά σημεία με χαμηλή αντίθεση είναι πολύ ευαίσθητα στο θόρυβο και δεν πρέπει να χρησιμοποιούνται. Η Gaussian διαφορά είναι πολύ ευαίσθητη στις ακμές ακόμη και αν η θέση του σημείου κατά μήκος της ακμής είναι ανεπαρκώς εντοπισμένη.

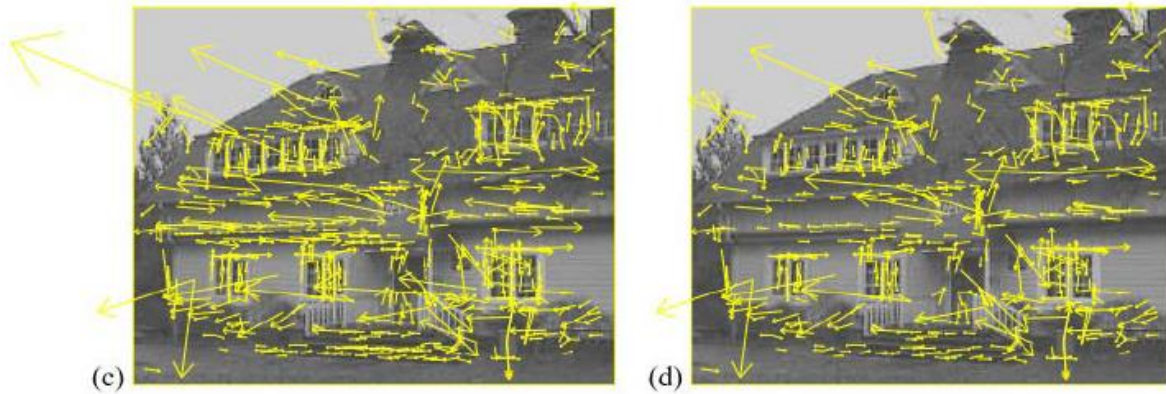


Εικόνα 11 - Ο υπολογισμός του μέγιστου και ελαχίστου της Gaussian διαφοράς σε ένα pixel

Οι παρακάτω εικόνες (εικόνες 12 και 13) απεικονίζουν τα στάδια επιλογής των σημείων κλειδιών σε μια εικόνα. Συγκεκριμένα στο παράθυρο (a) φαίνεται η 233x189 pixel πρωτότυπη εικόνα. Στο (b) φαίνονται οι αρχικές τοποθεσίες των 832 σημείων κλειδιά, στα μέγιστα και ελάχιστα της Gaussian διαφοράς. Τα σημεία κλειδιά εμφανίζονται ως διανύσματα υποδεικνύοντας την κλίμακα, τον προσανατολισμό και την τοποθεσία. Στο (c), μετά την εφαρμογή ενός κατωφλίου ελάχιστης αντίθεσης, παραμένουν 729 σημεία κλειδιά. Στο(d), φαίνονται τα τελικά 536 σημεία κλειδιά τα οποία παραμένουν, αφού έχουν υποστεί την εφαρμογή ενός επιπλέον κατωφλίου που υπολογίζει το λόγο των κύριων καμπυλοτήτων.



Εικόνα 12 - Τα αποτελέσματα της επιλογής σημείων κλειδιών σε μια εικόνα (α)



Εικόνα 13 - Τα αποτελέσματα της επιλογής σημείων κλειδιών σε μια εικόνα (β)

Στην επιστήμη των υπολογιστών ένα δένδρο k-d θεωρείται μια δομή δεδομένων η οποία διαχωρίζει χωρικά και χρησιμοποιείται για την οργάνωση των σημείων σε ένα k-διάστατο χώρο. Τα k-d δένδρα είναι μια ειδική περίπτωση των δυαδικών δένδρων διαχωρισμού του χώρου. Θεωρούνται μια χρήσιμη δομή δεδομένων για διάφορες εφαρμογές όπως έρευνες που αφορούν στον κοντινότερο γείτονα, όπως θα δούμε παρακάτω.

Ο Best bin first είναι ένας αλγόριθμος αναζήτησης ο οποίος έχει σχεδιαστεί για να βρίσκει αποτελεσματικά μια κατά προσέγγιση λύση στο πρόβλημα της αναζήτησης του κοντινότερου γείτονα για χώρους με πολύ μεγάλες διαστάσεις. Ο αλγόριθμος βασίζεται σε μια παραλλαγή του k-d δένδρου αναζήτησης ο οποίος επιτρέπει τη δημιουργία ευρετηρίου υψηλότερων χωρικών διαστάσεων.

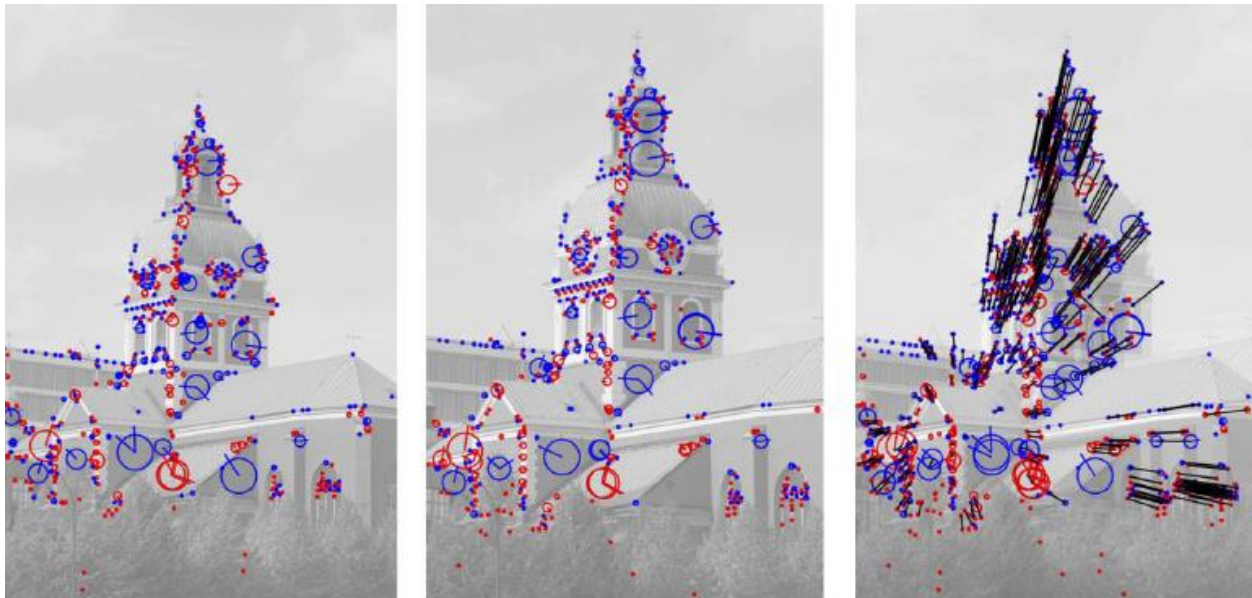
Αντιστοίχιση χαρακτηριστικών και προσθήκη σε ευρετήριο

Η προσθήκη στο ευρετήριο γίνεται με αποθήκευση χαρακτηριστικών SIFT και εντοπισμό όσων ταιριάζουν με αυτά της νέας εικόνας. Ο Lowe χρησιμοποίησε μια τροποποιημένη μορφή του k-d δένδρου που ονομάζεται Best-bin-first μέθοδος αναζήτησης η οποία μπορεί να εντοπίσει την πλησιέστερη γειτονιά (nearest neighbor) με υψηλή πιθανότητα, χρησιμοποιώντας χαμηλή υπολογιστική ισχύ.

Η παραπάνω μέθοδος αναζήτησης χρησιμοποιεί μια μορφοποιημένη ταξινόμηση του αλγορίθμου k-d δένδρου, έτσι ώστε τα δοχεία αποθήκευσης των χαρακτηριστικών να αναζητούνται και προς την πλησιέστερη θέση τους στο αρχικό σημείο ενδιαφέροντος. Αυτή η σειρά αναζήτησης απαιτεί τη χρήση ενός σωρού, βασισμένου στην ουρά προτεραιότητας για τον αποτελεσματικό καθορισμό της διάταξης για την αναζήτηση. Η καλύτερη αντιστοίχιση για κάθε χαρακτηριστικό βρίσκεται εντοπίζοντας την κοντινότερη γειτονιά του στη βάση δεδομένων των χαρακτηριστικών από τις εικόνες που εκπαιδεύονται. Ως κοντινότερες γειτονιές ορίζονται τα κύρια σημεία με ελάχιστη Ευκλείδεια απόσταση από κάθε δεδομένο διάνυσμα «περιγραφέα». Η πιθανότητα ότι μια αντιστοίχιση είναι σωστή μπορεί να καθοριστεί λαμβάνοντας υπόψη την αναλογία της απόστασης από το πλησιέστερο γείτονα, με την

απόσταση από το δεύτερο κοντινότερο γείτονα. Η υλοποίηση του Lowe απορρίπτει όλες τις πιθανές αντιστοιχίες των οποίων η αναλογία είναι μεγαλύτερη από 0.8. Με αυτό τον τρόπο εξαλείφεται το 90 % των λανθασμένων αντιστοιχιών ενώ παράλληλα απορρίπτεται μονάχα το 5% των σωστών. Για την περαιτέρω βελτίωση της αποτελεσματικότητας του αλγορίθμου, η αναζήτηση σταματά μετά τον έλεγχο των πρώτων 200 υποψήφιων πλησιέστερων γειτόνων. Έτσι για παράδειγμα, για μια βάση δεδομένων 100.000 σημείων με τη συγκεκριμένη υλοποίηση παρέχεται μια επιτάχυνση, πάνω ακριβώς από την αναζήτηση πλησιέστερου γείτονα, δυο τάξεων μεγέθους, αλλά με αποτελέσματα απώλειας σωστών αντιστοιχίσεων λιγότερο από 5% .

Στην παρακάτω εικόνα φαίνονται τα σημεία ενδιαφέροντος τα οποία ανιχνεύονται σε δυο διαφορετικές εικόνες της ίδιας σκηνής, με τα υπολογισμένα ταιριάσματα της εικόνας να σημειώνονται με μαύρες γραμμές μεταξύ των αντίστοιχων σημείων ενδιαφέροντος. Τα βέλη στο κέντρο των κύκλων, δείχνουν τον εκτιμώμενο προσανατολισμό, ο οποίος έχει αποκτηθεί από τις τοπικές κορυφές του ιστογράμματος προσανατολισμού γύρω από τα σημεία ενδιαφέροντος.



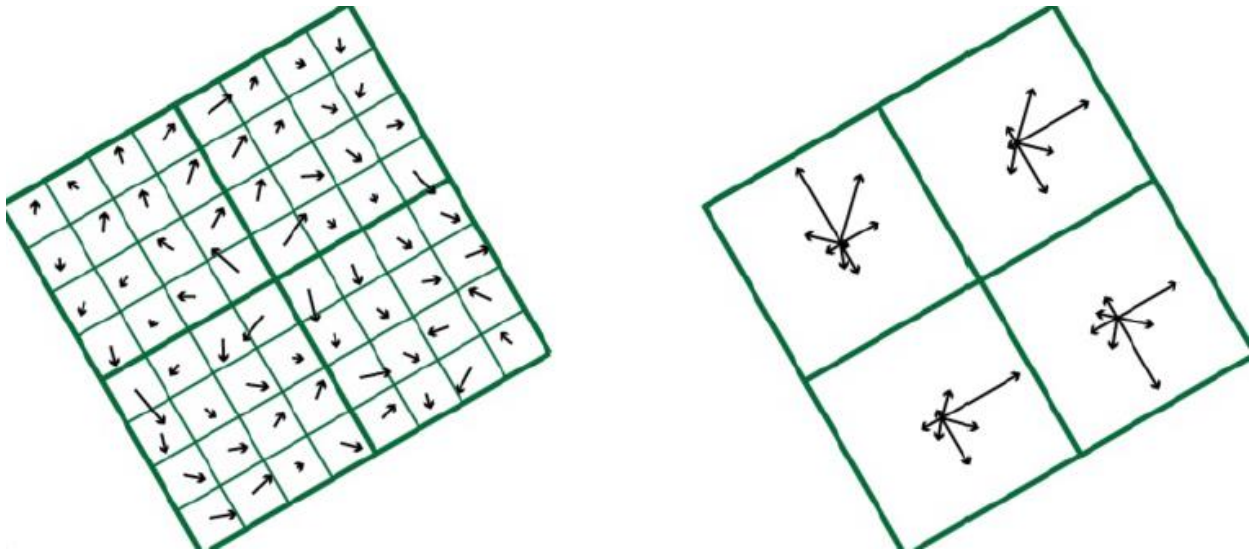
Εικόνα 14 - Παράδειγμα αντιστοίχισης SIFT χαρακτηριστικών

Εκχώρηση Προσανατολισμού

Για την επίτευξη του αμετάβλητου στην περιστροφή της εικόνας, σε κάθε σημείο κλειδί ανατίθενται μια ή περισσότερες κατευθύνσεις. Ένα ιστόγραμμα προσανατολισμού σχηματίζεται από τους προσανατολισμούς κλίσης των pixels γύρω από κάθε σημείο κλειδί. Κάθε δείγμα σταθμίζεται από μέγεθος της κλίσης και ένα Gaussian παράθυρο. Το μέγιστο του ιστογράμματος και οποιοδήποτε άλλο

τοπικό μέγιστο (με αναλογία έως 0.8), χρησιμοποιείται για την παραγωγή ενός σημείου κλειδί. Για να αυξηθεί η ακρίβεια των τοπικών ιστογραμμάτων χρησιμοποιείται τριγωνική παρεμβολή για τη διανομή των σταθμισμένων προσαναυξήσεων για τις μετρήσεις του δείγματος εικόνας σε διπλανούς κάδους ιστογράμματος. Με άλλα λόγια κάθε καταχώριση στον κάδο πολλαπλασιάζεται με ένα επιπλέον βάρος 1-d, όπου d είναι η απόσταση του δείγματος από την κεντρική θέση του κάδου, εκφρασμένη σε μονάδες απόστασης στο ιστόγραμμα.

Στην παρακάτω εικόνα απεικονίζεται το πώς οι SIFT περιγραφείς υπολογίζονται από την δειγματοληψία τιμών του προσανατολισμού της κλίσης και το μέγεθός της, πάνω σε τοπικά προσαρμοσμένο πλέγμα γύρω από κάθε σημείο ενδιαφέροντος. Η εικόνα αυτή δείχνει τον περιγραφέα που υπολογίστηκε σε μια περιοχή 2x2 ενώ συνήθως ο SIFT «περιγραφέας» υπολογίζεται για περιοχές μεγέθους 4x4.



Εικόνα 15 - Παράδειγμα υπολογισμού SIFT περιγραφέων (descriptors) σε περιοχή 2x2

2.5.3 Histogram based (Κατάτμηση Ιστογράμματος)

Οι μέθοδοι κατάτμησης ιστογράμματος (histogram based) είναι από τις πιο αξιόπιστες μεθόδους που χρησιμοποιούνται στην αναγνώριση αντικειμένων. Εντούτοις, είναι αρκετά δύσκολη σε ευρεία κλίμακα λόγω των πολύ μεγάλων αναγκών της σε απαιτούμενη μνήμη.

Στη μέθοδο αυτή, για κάθε εικόνα εξάγονται ιστογράμματα με όλα τα pixels των περιοχών ελέγχου (αρχικά ολόκληρης της εικόνας). Σαν σημεία ενδιαφέροντος χρησιμοποιούνται τα κύλα και οι κορυφές του ιστογράμματος. Το συνηθέστερο μέτρο σύγκρισης στις histogram based μεθόδους είναι το χρώμα ή η ένταση του κάθε pixel.

Στην περίπτωση που σαν μέτρο σύγκρισης χρησιμοποιείται η φωτεινότητα του κάθε pixel, είναι πολύ συχνά αναγκαία η χρήση βοηθητικών μεθόδων όπως Ισοστάθμιση Ιστογράμματος (Histogram Equalization) που αποτελούν μετασχηματισμούς στην κλίμακα του γκρι και έχουν ως στόχο την αύξηση της αντίθεσης. Με τον τρόπο αυτό δημιουργούνται νέες εικόνες με ισοκατανεμημένα επίπεδα φωτεινότητας σε όλο το φάσμα τους, στις οποίες η διαφορά στην ένταση των pixels γίνεται αρκετά εντονότερη και κατά συνέπεια πιο «ευανάγνωστη».

Όταν η κατάτμηση γίνεται μέσω χρώματος έχουμε μονοδιάστατο ιστόγραμμα της εικόνας, για μονόχρωμες και πολυδιάστατο για έγχρωμες. Η γενικότερη λογική είναι ότι η ανίχνευση κορυφών ή κοιλάδων στο ιστόγραμμα, οδηγούν στην εύρεση αντικειμένων και φόντου αντίστοιχα. Στις έγχρωμες εικόνες, η διαδικασία είναι πιο περίπλοκη, διότι τα τρία ιστογράμματα, του χρωματικού δηλαδή χώρου R.G.B. (Red Green Blue – Κόκκινο Πράσινο Μπλε), μπορούν να επεξεργαστούν χωριστά και να συνδυαστούν τα αποτελέσματά τους, με στόχο να επικρατήσει η ισχυρότερη υπόθεση κατάτμησης, δηλαδή αυτή που δίνει το αντικείμενο που θέλουμε καλύτερα, μέσω του συνδυασμού απόχρωσής του. Επίσης, οι αναπαραστάσεις H.S.V. (Hue Saturation Value - Απόχρωση Κορεσμός Μέγεθος) και H.S.I. (Hue Saturation Intensity - Απόχρωση Κορεσμός Ένταση) δίνουν το πλεονέκτημα της εύρεσης αντικειμένων με βάση τις αποχρώσεις, αφού συνήθως τα αντικείμενα έχουν ομοιογενή χρώματα, ενώ με αυτό τον τρόπο, οι εναλλαγές σε φωτεινότητα λόγω προβλημάτων φωτισμού, όπως σκιές, κυρτές και κοίλες επιφάνειες κ.α., δεν επηρεάζουν το αποτέλεσμα. Κοινό πρόβλημα σε τέτοιες τεχνικές είναι η παρουσία θορύβου, που θα εμφανίσει κορυφές στο ιστόγραμμα, έτσι η ομαλοποίηση (smoothing) του ιστογράμματος ως προεπεξεργασία του αλγορίθμου κατάτμησης ιστογράμματος, μερικές φορές είναι απαραίτητη.

Όταν η κατάτμηση γίνεται μέσω της φωτεινότητας του κάθε pixel, τα πράγματα είναι λιγότερο σύνθετα. Μιλώντας για μια ασπρόμαυρη (gray-scale) εικόνα, όπου φαίνεται πιο καθαρά η έννοια της φωτεινότητας, κάθε αντικείμενο με σχετικά μεγάλο μέγεθος δημιουργεί μια κατανομή pixels στο ιστόγραμμα της εικόνας γύρω από τη μέση τιμή φωτεινότητάς του. Όπως είναι γνωστό, για μονόχρωμες εικόνες, με τον όρο χαρακτηριστικά φωτεινότητας αναφερόμαστε στις τιμές που μπορεί να λάβει ένα pixel. Συγκεκριμένα, υπάρχουν 256 αποχρώσεις του γκρι με 0 να συμβολίζει το μαύρο και 255 το λευκό. Αν η εικόνα μας περιέχει ένα μόνο αντικείμενο σε σχετικά ομοιόμορφο φόντο (background) τότε έχουμε δυο κατανομές pixels στο ιστόγραμμα και αρκεί να βρούμε μια τιμή κατωφλίου T (Threshold) η οποία θα ξεχωρίζει εκατέρωθεν της την κατανομή pixels που αντιπροσωπεύει το αντικείμενο και την κατανομή pixels που αντιπροσωπεύει το φόντο. Βελτιώσεις της μεθόδου αυτής είναι η πολυκατωφλίωση όταν έχουμε πολλά αντικείμενα προς κατάτμηση, η ολική κατωφλίωση, δηλαδή σε όλη την εικόνα, όταν όμως γνωρίζουμε ότι το αντικείμενο έχει μεγαλύτερη ένταση απ' το φόντο, η τοπική κατωφλίωση όταν οι κατανομές στο ιστόγραμμα είναι δυσδιάκριτες και η μέθοδος Otsu που ελαχιστοποιεί την πιθανότητα εσφαλμένης ταξινόμησης των pixels μέσα στις κατανομές τους.

Μια άλλη βελτίωση της μεθόδου κατάτμησης ιστογράμματος είναι να εφαρμοστεί κατ' επανάληψη η κατασκευή ιστογράμματος στις συστάδες της εικόνας προκειμένου να διαιρεθούν σε μικρότερες συστάδες. Αυτό επαναλαμβάνεται σε ολόένα και μικρότερες συστάδες, έως ότου να μην δημιουργούνται άλλες συστάδες. Ένα μειονέκτημα της μεθόδου αυτής είναι ότι μπορεί να είναι δύσκολο να προσδιοριστούν τα σημαντικά κύλα και οι σημαντικές κορυφές στο ιστόγραμμα.

3 Προτεινόμενη Μέθοδος

3.1 Η μέθοδος TFS-SVMPC

3.1.1 TFS-SVMPC – Μαθηματικές Συναρτήσεις

Θεωρούμε ένα σύνολο S που αποτελεί το σύνολο των δεδομένων εκπαίδευσης και επιθυμητών εξόδων. Το σύνολο αυτό έχει αρχικά τη μορφή:

$$S = \{(\overline{x}_1, y_1), (\overline{x}_2, y_2), \dots, (\overline{x}_N, y_N)\}$$

όπου $\overline{x}_i \in \mathbb{R}^n$ και $y_i \in \{+1, -1\}$. Η έξοδος y_i παίρνει την τιμή +1 αν η είσοδος x_i είναι αντικείμενο (object) και τιμή -1 αν η είσοδος x_i δεν είναι αντικείμενο (non-object).

Κάθε κανόνας στον TFS-SVMPC έχει μορφή κανόνα if-then, όπως φαίνεται παρακάτω:

$$\text{Rule } i : \text{IF } x_i \text{ is } A_{i1} \text{ AND } \dots \text{ AND } x_n \text{ is } A_{in} \text{ THEN } y' = \sum_{j=0}^n a_{ij}x_j \quad (5.1)$$

όπου:

$$x_0 \triangleq 1, x_1, \dots, x_n$$

A_{ij} : είναι ένα fuzzy σύνολο

a_{ij} : είναι μία παράμετρος, δηλαδή ένας πραγματικός αριθμός (εδώ ο πολλαπλασιαστής Lagrange που χρησιμοποιούν οι SVM)

y' : είναι η έξοδος του συστήματος, που παίρνει μία πραγματική τιμή

Ο χαρακτηρισμός μίας εισόδου x_i ως αντικειμένου ή όχι, πραγματοποιείται εφαρμόζοντας μία συνάρτηση y' . Αρχικά τα δεδομένα εισόδου είναι ένα σύνολο δεδομένων εκπαίδευσης (test samples).

Βασική διαφορά ενός TFS-SVMPC με τα κοινά SVM-trained fuzzy systems είναι πως, ενώ τα τελευταία έχουν μόνο μία ελεύθερη παράμετρο ανά κανόνα, στον TFS-SVMPC υπάρχουν $n+1$ παράμετροι (a_{i0}, \dots, a_{in}) σε κάθε κανόνα.

Το fuzzy σύνολο A_{ij} χρησιμοποιείται με την ακόλουθη Gaussian συνάρτηση συμμετοχής:

$$G_{ij}(x_j) = \exp\left\{-\frac{(x_j - m_{ij})^2}{2\sigma_i^2}\right\} \quad (5.2)$$

όπου:

m_{ij} : είναι το κέντρο του fuzzy συνόλου

σ_i : είναι το εύρος του fuzzy συνόλου

Για κάθε σύνολο δεδομένων εισόδου $\vec{x} = [x_1, \dots, x_n]^T$, υπολογίζεται ένα firing strength $\mu_i(\vec{x})$ για κάθε κανόνα i . Το firing strength $\mu_i(\vec{x})$ υπολογίζεται από τη σχέση:

$$\mu_i(\vec{x}) = \prod_{j=1}^n G_{ij}(x_j) = \exp\left\{-\frac{\|\vec{x} - \vec{m}_i\|^2}{2\sigma_i^2}\right\} \quad (5.3)$$

όπου:

$$\vec{m}_i = [m_{i1}, \dots, m_{in}]^T$$

Για την αποασαφοποίηση (Defuzzification) ο TFS-SVMPC χρησιμοποιεί το απλό σταθμισμένο άθροισμα. Η συνάρτηση εξόδου του είναι:

$$y' = \sum_{i=1}^r \mu_i \left(\sum_{j=0}^n a_{ij} x_j \right) + b = \vec{a}^T \vec{\phi} + b \quad (5.4)$$

όπου:

$\vec{a} = [a_{10}, \dots, a_{1n}, \dots, a_{r0}, \dots, a_{rn}]^T \in \mathbb{R}^{r(n+1) \times 1}$: είναι το παραμετρικό διάνυσμα των παραμέτρων a_{ij}

$\vec{\phi} = [\mu_1 x_0, \dots, \mu_1 x_n, \dots, \mu_r x_0, \dots, \mu_r x_n]^T \in \mathbb{R}^{r(n+1) \times 1}$: είναι το χαρακτηριστικό διάνυσμα εισόδου

Στην τελευταία εξίσωση (5.4) φαίνεται πως ο TFS-SVMPC μετασχηματίζει κάθε διάνυσμα εισόδου \vec{x} σε ένα χαρακτηριστικό διάνυσμα $\vec{\phi}$. Το χαρακτηριστικό αυτό διάνυσμα $\vec{\phi}$ είναι αρχικά μεγάλων διαστάσεων ($\vec{\phi} \in \mathbb{R}^{r(n+1) \times 1}$). Εδώ r είναι ο αριθμός του κανόνα. Το παραμετρικό διάνυσμα \vec{a} εκπαιδεύεται μέσω ενός γραμμικού SVM (βλ. Κεφάλαιο III).

Παρατηρούμε την προσθήκη ενός όρου $+b$ στην εξίσωση εξόδου. Όπως εξηγείται αναλυτικότερα στο επόμενο κεφάλαιο, ένα γραμμικό SVM υπολογίζει ένα γραμμικό υπερεπίπεδο διαχωρισμού για τα δεδομένα εισόδου. Το γραμμικό αυτό υπερεπίπεδο είναι μία γραμμική συνάρτηση που εμπεριέχει κλίση. Για τη συναρτησιακή ισοδυναμία μεταξύ της εξόδου y' και του γραμμικού υπερεπιπέδου, προστέθηκε στην τελική έξοδο του TFS-SVMPC μία κλίση $+b$. Η κλίση αυτή υπολογίζει τον όρο της παραγόμενης κλίσης στην εκπαίδευση του SVM.

Για τον TFS-SVMPC η προσθήκη μιας κλίσης $+b$ μπορεί να θεωρηθεί ως προσθήκη ενός νέου κανόνα στον κανόνα r . Ο νέος αυτός κανόνας θα είναι ο κανόνας $r + 1$ και θα έχει την παρακάτω μορφή:

$$\text{Rule } r + 1 : \text{IF } x_i \text{ is } A_{r+1n} \text{ AND } \dots \text{ AND } x_n \text{ is } A_{r+1n} \text{ THEN } y' = \sum_{j=0}^n a_{r+1n} x_j + b \quad (5.5)$$

όπου:

$$G_{r+11}(x_1) = \dots = G_{r+1n}(x_n) \equiv 1 \quad (\text{για παράδειγμα : } \mu_i(\vec{x}) \equiv 1)$$

$$a_{r+11} = \dots = a_{r+1n} = 0$$

Για να λάβουμε, δηλαδή, τελικά μόνο την προσθήκη της κλίσης $+b$, κρατάμε τη συνάρτηση συμμετοχής $G_{ij}(x_j)$ σταθερή στο 1 και μηδενίζουμε τις παραμέτρους a_{ij} . Έτσι στην έξοδο του κανόνα $r + 1$ παίρνουμε μόνο την κλίση $+b$, αφού οι υπόλοιποι όροι μηδενίζονται από το μηδενισμό των a_{ij} . Τόσο η τιμή της κλίσης $+b$, όσο και το παραμετροποιημένο διάνυσμα \vec{a} , εκπαιδεύονται μέσω ενός γραμμικού SVM.

3.1.2 TFS-SVMPC – Δομική Μάθηση

Το αντικείμενο του structure learning είναι να καθορίσει τον αριθμό των κανόνων r . Στο structure learning της μεθόδου TFS-SVMPC, ένα cluster ορίζεται ως ένας κανόνας fuzzy. Το κριτήριο για την παραγωγή ενός fuzzy κανόνα (δηλαδή για την παραγωγή ενός νέου cluster) είναι το ακόλουθο:

Για ένα εισερχόμενο δεδομένο \vec{x}_k , βρίσκουμε:

$$I = \arg \max_{1 \leq i \leq r_k} \mu_i(\vec{x}_k) \quad (5.6)$$

όπου r_k είναι ο αριθμός των cluster που υφίστανται στο χρόνο k .

Υπολογίζουμε το firing strength του I (μ_I) και συγκρίνουμε με ένα προκαθορισμένο από δοκιμές μ_{th} . Εάν ισχύει η σχέση:

$$\mu_I \leq \mu_{th}$$

τότε δημιουργούμε ένα νέο cluster (για παράδειγμα : $r \leftarrow r + 1$) και κατά συνέπεια έναν νέο κανόνα.

Μετά την παραγωγή του νέου cluster πρέπει να ορισθούν τα χαρακτηριστικά του. Αυτά είναι το κέντρο (\vec{m}_{r_k}) και το εύρος (σ_{r_k}) αυτού. Ο ορισμός των παραμέτρων αυτών γίνεται με τη βοήθεια των παρακάτω σχέσεων:

$$\vec{m}_{r_k} = \vec{x}_k \quad (5.7)$$

$$\sigma_{r_k} = \beta \frac{\|\vec{x}_k - \vec{m}_I\|^2}{\sigma_I^2} \quad (5.8)$$

όπου β είναι ο βαθμός επικάλυψης μεταξύ δύο cluster. Είναι μία προκαθορισμένη από δοκιμές παράμετρος με τιμή fixed για όλα τα cluster. Για το β ισχύει πάντα ότι $\beta > 0$. Όπως φαίνεται από τη δεύτερη εξίσωση, το εύρος σ_{r_k} του cluster είναι μία γεωμετρική σχέση ανάμεσα στην είσοδο x_k και στο κοντινότερο σε αυτήν cluster I .

3.2 Η μέθοδος TFS-SVMPC - Εκμάθηση Παραμέτρων

3.2.1 Η βασική ιδέα ενός γραμμικού SVM

Θεωρούμε ένα σύνολο S που αποτελεί το σύνολο των εκπαιδευτικών δεδομένων εισόδου και εξόδου. Το σύνολο αυτό έχει αρχικά τη μορφή:

$$S = \{(\vec{x}_1, y_1), (\vec{x}_2, y_2), \dots, (\vec{x}_N, y_N)\}$$

όπου $\vec{x}_i \in \mathbb{R}^n$ και $y_i \in \{+1, -1\}$.

Ένα γραμμικό SVM επιχειρεί να βρει το υπερεπίπεδο διαχωρισμού (για τα δεδομένα εισόδου):

$$\vec{w}^T \vec{x} + b = 0 \quad (5.9)$$

ορισμένο για $\vec{w} \in \mathbb{R}^n$ και $b \in \mathbb{R}$.

Σύμφωνα με το υπερεπίπεδο αυτό, γίνεται ο διαχωρισμός των δεδομένων εισόδου από τη συνάρτηση απόφασης:

$$f(\vec{x}) = \text{sign}(\vec{w}^T \vec{x} + b) \quad (5.10)$$

ανάλογα δηλαδή με το πρόσημο που δίνει κάθε δεδομένο εισόδου στην εξίσωση του υπερεπιπέδου. Λύνοντας τη συνάρτηση απόφασης, με ένα άλλο γραμμικό SVM, λαμβάνουμε την τελική συνάρτηση απόφασης του υπερεπιπέδου, όπως φαίνεται στην επόμενη σχέση:

$$f(\vec{x}) = \text{sign}(\vec{w}^T \vec{x} + b) = \text{sign}\left(\sum_{i=1}^N y_i a_i \langle \vec{x}, \vec{x}_i \rangle + b\right) = \text{sign}\left(\sum_{i \in SV} y_i a_i \langle \vec{x}, \vec{x}_i \rangle + b\right) \quad (5.11)$$

όπου όλα τα εκπαιδευτικά δεδομένα για τα οποία ισχύει $a_i \neq 0$, είναι διανύσματα υποστήριξης (support vectors).

3.2.2 Μείωση της διάστασης των χαρακτηριστικών διανυσμάτων με τη χρήση της PCA

Ο TFS-SVMPC μετασχηματίζει κάθε είσοδο \vec{x} που λαμβάνει ένα μεγάλων διαστάσεων χαρακτηριστικό διάνυσμα $\vec{\phi}$, το οποίο στη συνέχεια χρησιμοποιείται σαν είσοδος του συστήματος.

Όπως είδαμε νωρίτερα, το χαρακτηριστικό διάνυσμα $\vec{\phi}$ είναι διάστασης $r(n+1)$:

$$\vec{\phi} = [\mu_1 x_0, \dots, \mu_1 x_n, \dots, \mu_r x_0, \dots, \mu_r x_n]^T \in \mathbb{R}^{r(n+1) \times 1} \quad (5.12)$$

Η διάσταση, λοιπόν, του χαρακτηριστικού διανύσματος $\vec{\phi}$, για μεγάλες τιμές των r και n , γίνεται πολύ μεγάλη, παίρνοντας τιμές που είναι απαγορευτικές για τη χρήση του σε υπολογισμούς,

αφού η έξοδος του συστήματος είναι ένας γραμμικός συνδυασμός όλων των συνιστωσών των χαρακτηριστικών διανυσμάτων. Χρησιμοποιώντας το χαρακτηριστικό διάνυσμα $\vec{\phi}$ ως είσοδο, προκύπτουν πολύ μεγάλοι χρόνοι υπολογισμού, που είναι ακατάλληλοι για πολλές εφαρμογές που απαιτούν υψηλή ταχύτητα.

Για να προσπεραστεί το συγκεκριμένο πρόβλημα, ο TFS-SVMPC κάνει χρήση της PCA (ανάλυση κύριας συνιστώσας) για να επιτύχει τη μείωση του μεγέθους του χαρακτηριστικού διανύσματος $\vec{\phi}$, με ταυτόχρονη διατήρηση (όσο αυτό είναι δυνατόν) της απόκλισης που υπάρχει στο αρχικό σύνολο δεδομένων.

Από τη στιγμή που το χαρακτηριστικό διάνυσμα $\vec{\phi}$ είναι πλέον η είσοδος του συστήματος, η αναπαράσταση των αρχικών δεδομένων (εκπαιδευτικών) μπορεί να πάρει τη μορφή:

$$S = \{(\vec{\phi}_1(\vec{x}_1), y_1), (\vec{\phi}_2(\vec{x}_2), y_2), \dots, (\vec{\phi}_N(\vec{x}_N), y_N)\} \quad (5.13)$$

Για να διατηρηθεί η απόκλιση των αρχικών δεδομένων εισόδου, υπολογίζεται ένας πίνακας συνδιακύμανσης των (εδώ εκπαιδευτικών) δεδομένων (K), με τη βοήθεια της παρακάτω σχέσης:

$$K = \frac{1}{N} \sum_{i=1}^N \vec{\Phi}_i \vec{\Phi}_i^T \in \mathbb{R}^{r(n+1) \times r(n+1)} \quad (5.14)$$

όπου:

$$\vec{\Phi}_i = \vec{\phi}_i - \vec{\underline{\phi}}_i \quad \text{και} \quad \vec{\underline{\phi}}_i = \frac{1}{N} \sum_{i=1}^N \vec{\phi}_i \quad (5.15)$$

Αποβάλλοντας το θόρυβο της εισόδου, ο TFS-SVMPC κάνει επιλογή των M μεγαλύτερων ιδιοτιμών που διατηρούνται ($\lambda_1, \dots, \lambda_M$). Από το σύνολο των ιδιοδιανυσμάτων ($\vec{v}_1, \dots, \vec{v}_M$) που αντιστοιχούν στις M αυτές ιδιοτιμές, παίρνουμε έναν πίνακα γραμμικού μετασχηματισμού V :

$$V = \begin{bmatrix} \vec{v}_1^T \\ \vdots \\ \vec{v}_M^T \end{bmatrix} \in \mathbb{R}^{M \times r(n+1)} \quad (5.16)$$

Με τη βοήθεια του πίνακα V υπολογίζεται το νέο χαρακτηριστικό διάνυσμα \vec{u} , από τη σχέση:

$$\vec{u} = V (\vec{\phi} - \vec{\underline{\phi}}) \in \mathbb{R}^{M \times 1} \quad (5.17)$$

Η διάσταση του νέου χαρακτηριστικού διανύσματος \vec{u} έχει μειωθεί στην τιμή M (από την αρχική $r(n + 1)$), μέσω της απόρριψης ενός αριθμού μικρών ιδιοτιμών. Η έξοδος του συστήματος παίρνει τώρα τη μορφή:

$$y' = \vec{w}^T \vec{u} + b' \quad (5.18)$$

ή αναλυτικότερα:

$$\left. \begin{aligned} \vec{u} &= V(\vec{\phi} - \underline{\phi}) \\ y' &= \vec{w}^T \vec{u} + b' \end{aligned} \right\} \Rightarrow y' = \vec{w}^T V(\vec{\phi} - \underline{\phi}) + b' = \vec{w}^T V \vec{\phi} - \vec{w}^T V \underline{\phi} + b' \quad (5.19)$$

Οπότε, το προκύπτον παραμετρικό διάνυσμα \vec{a} και η κλίση b , δίνονται από τις σχέσεις:

$$\vec{a}^T = \vec{w}^T V \quad (5.20)$$

$$b = -\vec{w}^T V \underline{\phi} + b' \quad (5.21)$$

Αφού λοιπόν καθοριστούν το διάνυσμα βάρους \vec{w} και η κλίση b' , καθορίζονται ανάλογα και το παραμετρικό διάνυσμα \vec{a} και η κλίση b .

3.3 Parameter Learning με ένα γραμμικό SVM

Το νέο χαρακτηριστικό διάνυσμα \vec{u} αποτελεί, από τη στιγμή που υπολογίζεται, την είσοδο του συστήματος. Το σύνολο των δεδομένων εκπαίδευσης, αναπαρίσταται πλέον με τη βοήθεια του νέου, μειωμένης διαστατικότητας, χαρακτηριστικού διανύσματος \vec{u} :

$$S = \{(\vec{u}_1(\vec{x}_1), y_1), (\vec{u}_2(\vec{x}_2), y_2), \dots, (\vec{u}_N(\vec{x}_N), y_N)\} \quad (5.22)$$

Το νέο χαρακτηριστικό διάνυσμα \vec{u} τροφοδοτείται σαν είσοδος σε ένα γραμμικό SVM και τα ζεύγη δεδομένων ταξινομούνται βάσει της συνάρτησης απόφασης που είδαμε νωρίτερα:

$$f(\vec{x}) = \text{sign}(\vec{w}^T \vec{x} + b) = \text{sign}\left(\sum_{i=1}^N y_i a_i \langle \vec{x}, \vec{x}_i \rangle + b\right) = \text{sign}\left(\sum_{i \in SV} y_i a_i \langle \vec{x}, \vec{x}_i \rangle + b\right) \quad (5.23)$$

Από τη συνάρτηση απόφασης φαίνεται πως η ευνοϊκότερη γραμμική συνάρτηση (εξόδου), στο νέο μειωμένο χαρακτηριστικό διάνυσμα, είναι η:

$$y' = \sum_{i=1}^N y_i a_i \langle \vec{u}(\vec{x}), \vec{u}_i(\vec{x}_i) \rangle + b' \quad (5.24)$$

που μπορεί να εκφραστεί και ως:

$$y'(\vec{x}) = \sum_{i=1}^N y_i a_i \left[\sum_{k=1}^M u_k(\vec{x}), u_k(\vec{x}_i) \right] + b' = \sum_{k=1}^M w_k u_k(\vec{x}) + b' \quad (5.25)$$

όπου:

$$w_k = \sum_{i=1}^N y_i a_i u_k(\vec{x}_i) = \sum_{i \in SV} y_i a_i u_k(\vec{x}_i), \quad k = 1, \dots, M$$

Η τελευταία σχέση λύνει το διάνυσμα βάρους \vec{w} στην εξίσωση της εξόδου:

$$y' = \vec{w}^T \vec{u} + b'$$

Με γνωστό το βάρος, λύνονται ανάλογα και το παραμετρικό διάνυσμα \vec{a} και η κλίση b , σύμφωνα με τις σχέσεις:

$$\vec{a}^T = \vec{w}^T V \quad (5.26)$$

$$b = -\vec{w}^T V \vec{\phi} + b' \quad (5.27)$$

που είδαμε νωρίτερα.

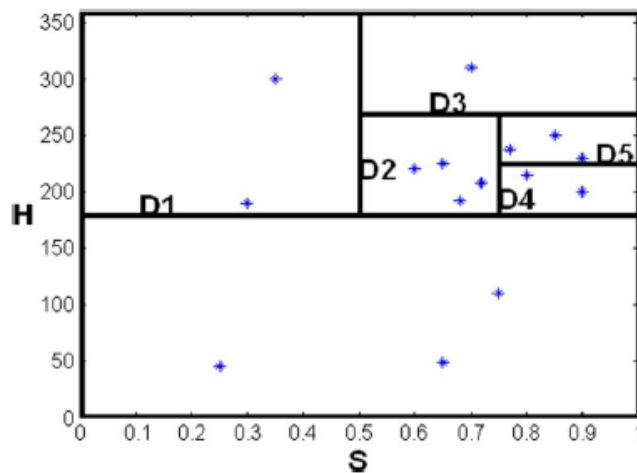
4 Ανίχνευση με βάση Περιγραφείς Ολικούς-Τοπικούς και την Προτεινόμενη Μέθοδο

Η προτεινόμενη, βασισμένη σε TFS-SVMPC, μέθοδος ανίχνευσης αποτελείται από 2 στάδια. Το πρώτο στάδιο ανιχνεύει ένα αντικείμενο χρησιμοποιώντας ένα χρωματικό χαρακτηριστικό, που προέρχεται από την καθολική (global) του εμφάνιση. Πολλά υποψήφια αντικείμενα παράγονται στο

στάδιο αυτό. Το επόμενο στάδιο χρησιμοποιεί τοπικά χρωματικά χαρακτηριστικά, σύμφωνα με τη γεωμετρική κατανομή για την τελική ανίχνευση.

4.1 Διαίρεση του Χρωματικού Χώρου και Ιστόγραμμα

Η απόχρωση (hue), η χρωματική καθαρότητα (saturation) και ο βαθμός φωτεινής έντασης του χρωματικού χώρου είναι ένας τυπικός μη-γραμμικός μετασχηματισμός του κόκκινου, πράσινου και μπλε χρωματικού χώρου. Στο παρελθόν αυτό χρησιμοποιείται ο HS χρωματικός χώρος για τον υπολογισμό του χρωματικού ιστογράμματος, ώστε να μειωθεί η επιρροή της φωτεινότητας. Ένα χρωματικό ιστόγραμμα επιτυγχάνεται υπολογίζοντας τον αριθμό των pixels που εμπίπτουν σε κάθε «αποθήκη / κελάρι» (bin) μέσα στον HS χώρο. Για την αναπαράσταση των πληροφοριών του ιστογράμματος με τη μεγαλύτερη δυνατή ακρίβεια, αντί μίας ομοιόμορφης διαμέρισης, χρησιμοποιείται μία ανομοιόμορφη διαμέριση του HS χώρου. Τα pixels ενός αντικειμένου είναι συνήθως ανομοιόμορφα κατανεμημένα στον HS χώρο. Αντίθετα με το clustering, ο κύριος στόχος της διαμέρισης (partition) είναι να γίνουν όλα τα pixels της εικόνας εξίσου κατανεμημένα σε κάθε «αποθήκη / κελάρι» (bin). Απομακρυσμένα χρώματα δεν είναι στο ίδιο «κελάρι» (bin).



Εικόνα 16 - Ανομοιόμορφη διαμέριση του HS χρωματικού χώρου, όπου D_i δηλώνει την i -ωστή διαμέριση.

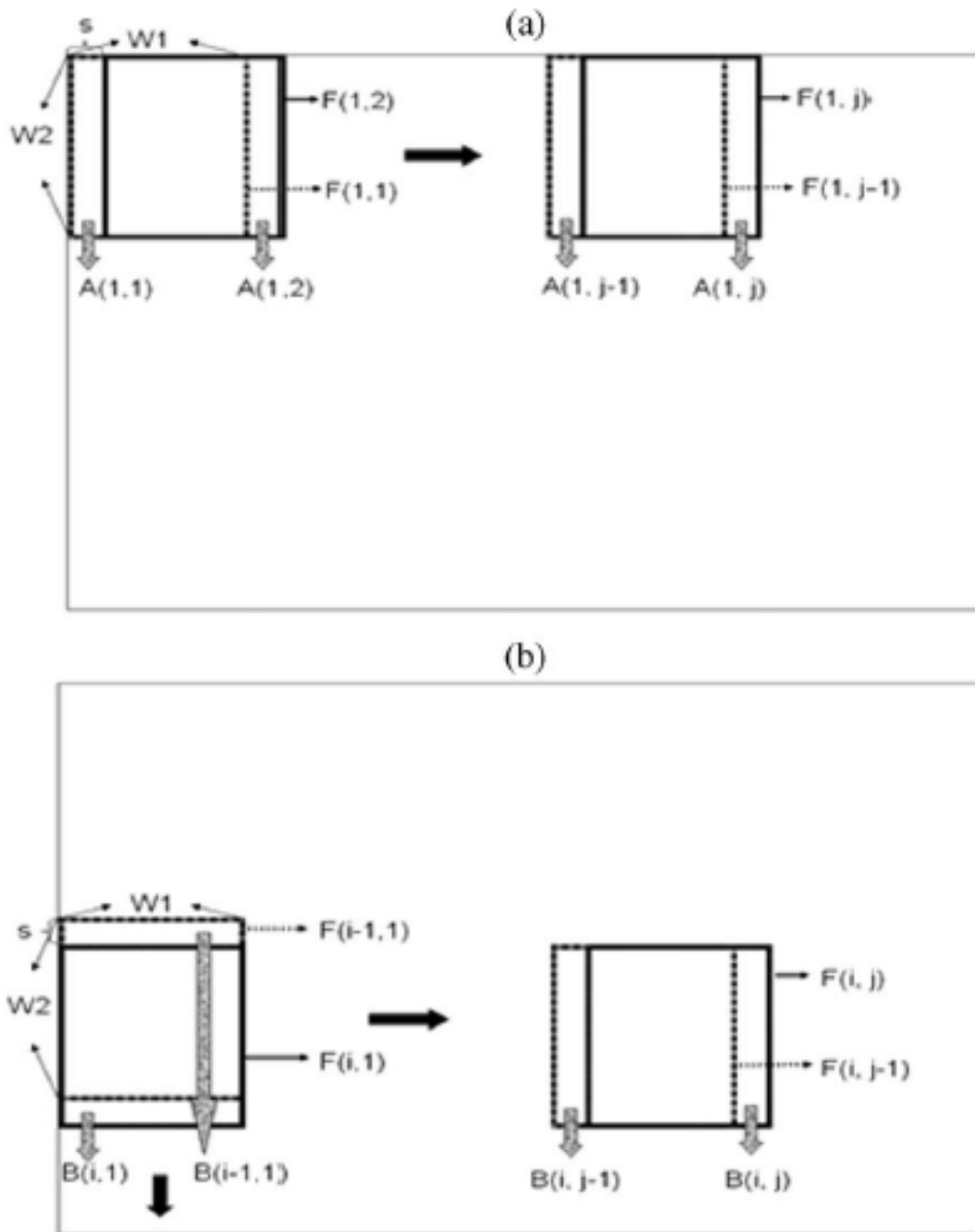
Για παράδειγμα, υποθέτουμε ότι υπάρχουν 9 pixels όπως φαίνεται στην Εικόνα 16. Η διαδικασία της ανομοιόμορφης διαμέρισης περιγράφεται ως εξής. Για τα διαδικασία της ανομοιόμορφης διαμέρισης του HS χώρου, η οριζόντια διαμέριση ακολουθείται από μία κάθετη διαμέριση, σε κάθε διαδικασία διαμέρισης. Για να εκτελεστεί μία νέα διαμέριση, υπολογίζεται ο αριθμός των pixels σε κάθε «κελάρι» (bin). Το «κελάρι» (bin) με το μεγαλύτερο αριθμό pixels διαμερίζεται. Σε κάθε διαδικασία διαμέρισης, πρώτα εκτελείται μία οριζόντια διαμέριση, που χωρίζει το «κελάρι» (bin) σε δύο «κελάρια» (bins) ίδιου μεγέθους. Τότε, το «κελάρι» (bin) με τα περισσότερα

pixels διαμερίζεται κάθετα σε 2 «κελάρια» (bins) ίδιου μεγέθους. Μία διαδικασία διαμέρισης τελειώνει μετά την ολοκλήρωση της κάθετης διαμέρισης. Με τον τρόπο αυτό, μετά από q διαδικασίες διαμέρισης, έχουμε $2q+1$ «κελάρια» (bins). Η Εικόνα 16 δείχνει τη διαδικασία της ανομοιόμορφης διαμέρισης. Μετά τη διαμέριση, τα pixels σε κάθε «κελάρι» (bin) γίνονται πιο ομοιόμορφα και ίσα κατανομημένα. Τα χρωματικά ιστογράμματα που χρησιμοποιούν ανομοιόμορφη διαμέριση του HS χώρου χρησιμοποιούνται σαν χαρακτηριστικά διανύσματα. Υπάρχουν αρκετές τεχνικές για την ανομοιόμορφη διαμέριση ενός χώρου, όπως το k-d δέντρο (k-d tree). Ωστόσο, για το πρόβλημα της ανίχνευσης αντικειμένων, στις περισσότερες μελέτες χρησιμοποιούνται ομοιόμορφες διαμερίσεις για την παραγωγή χρωματικών ιστογραμμάτων. Στη συνέχεια εφαρμόζουμε μία προσέγγιση ανομοιόμορφης διαμέρισης για να πάρουμε χρωματικά ιστογράμματα στον HS χώρο. Η προσέγγιση της απλής διαμέρισης και της γρήγορης εξαγωγής χαρακτηριστικών, που περιγράφονται στο επόμενο κεφάλαιο, καθιστούν εφικτή την ανίχνευση ενός αντικειμένου σε πραγματικό χρόνο.

4.2 Εξαγωγή Χαρακτηριστικών Ιστογράμματος

Για να ανιχνευτεί ένα αντικείμενο από μία δοκιμαστική εικόνα, καθορίζεται ένα τετραγωνικό παράθυρο διαστάσεων $\omega_1 \times \omega_2$ pixels, όπου η αναλογία μεταξύ ω_1 και ω_2 καθορίζεται από το μέγεθος του αντικειμένου. Για να ανιχνευτεί ένα αντικείμενο, οπουδήποτε μέσα σε μία εικόνα, πραγματοποιείται μία διεξοδική σάρωση της εικόνας με το τετραγωνικό παράθυρο. Κατά τη διάρκεια της σάρωσης, το τετραγωνικό παράθυρο αλλάζει θέση κατά s pixels ($s=5$ εδώ) προς μία κατεύθυνση τη φορά. Για την ανίχνευση διαφορετικού μεγέθους, η εισερχόμενη εικόνα αλλάζει συνεχώς μέγεθος κατά έναν σταθερό όρο γ (εδώ $\gamma=0.8$), $0 < \gamma < 1$. Εκτελείται διεξοδική σάρωση αυτών των resized εικόνων με το ίδιο fix-sized τετραγωνικό παράθυρο. Αυτή η διαδικασία της αλλαγής μεγέθους και σάρωσης της εισερχόμενης εικόνας, συνεχίζεται μέχρι η εικόνα να γίνει μικρότερη από το παράθυρο σάρωσης.

Κατά τη διαδικασία της σάρωσης της εικόνας, υπολογίζεται το ιστόγραμμα των pixels μέσα στο παράθυρο σάρωσης (στην περιοχή εστίασης). Τα ιστογράμματα μπορούν να επαναυπολογίζονται ανεξάρτητα και απευθείας από κάθε περιοχή εστίασης. Ωστόσο, μία τέτοια εξαγωγή ιστογράμματος είναι χρονοβόρα και άβολη για ανίχνευση αντικειμένων σε πραγματικό χρόνο. Εδώ χρησιμοποιούμε μία μέθοδο γρήγορης εξαγωγής ιστογράμματος κατά τη διάρκεια της διαδικασίας σάρωσης. Η Εικόνα 17(a) απεικονίζει την προτεινόμενη μέθοδο υπολογισμού.



Εικόνα 17 - (a) Πρώτη οριζόντια σάρωση για την εξαγωγή γρήγορου ιστογράμματος. (b) Κάθετες και οριζόντιες σαρώσεις για την εξαγωγή γρήγορου ιστογράμματος.

Η i περιοχή εστίασης και η j οριζόντια γραμμή σάρωσης δηλώνονται ως $F(i,j)$, και το ιστόγραμμα της περιοχής αυτής δηλώνεται ως $H_{F(i,j)}$. Στην αρχή της διαδικασίας σκαναρίσματος, το ιστόγραμμα της $F(1,1)$ υπολογίζεται της μεθόδου γενικού υπολογισμού ιστογραμμάτων. Το παράθυρο σάρωσης μετακινείται δεξιά κατά s pixels, από την $F(1,1)$ στην επόμενη περιοχή εστίασης $F(1,2)$. Οι δύο γειτονικές περιοχές εστίασης $F(1,1)$ και $F(1,2)$, έχουν μία περιοχή επικάλυψης της τάξης των $(\omega_1 - 2s) \times \omega_2$ pixels. Επομένως, είναι περιττό να υπολογίσουμε ξανά το ιστόγραμμα για αυτήν την περιοχή επικάλυψης. Μόνο ιστογράμματα μη-επικαλυπτόμενων περιοχών χρειάζεται να υπολογισθούν. Η αριστερή και δεξιά μη-επικαλυπτόμενες περιοχές δηλώνονται ως $A(1,1)$ και $A(1,2)$ [όπως φαίνεται στην Εικόνα 17(a)] αντιστοίχως, όπου και οι δύο περιοχές έχουν ακριβώς ίδιο μέγεθος της τάξης των $s \times \omega_2$. Τα ιστογράμματα των $A(1,1)$ και $A(1,2)$ δηλώνονται ως $H_{F(1,1)}$ και $H_{F(1,2)}$ αντίστοιχα. Το ιστόγραμμα της περιοχής $F(1,2)$ υπολογίζεται απλά από την παρακάτω σχέση:

$$H_{F(1,2)} = H_{F(1,1)} - H_{A(1,1)} + H_{A(1,2)} \quad (5.28)$$

Μια ίδια υπολογιστική διαδικασία εφαρμόζεται και στις υπόλοιπες περιοχές εστίασης, στην ίδια γραμμή οριζόντιας σάρωσης. Δηλαδή, βασισμένοι στα διαθέσιμα ιστογράμματα $H_{F(1,j-1)}$ και $H_{A(1,j-1)}$, το ιστόγραμμα της νέας περιοχής εστίασης $F(1,j)$ υπολογίζεται από τη σχέση:

$$H_{F(1,j)} = H_{F(1,j-1)} - H_{A(1,j-1)} + H_{A(1,j)} \quad (5.29)$$

Για τις υπόλοιπες γραμμές οριζόντιας σάρωσης (π.χ. $i \geq 2$), το ιστόγραμμα για κάθε αρχική περιοχή εστίασης υπολογίζεται όπως παρακάτω. Όπως φαίνεται στην Εικόνα 17(b), η περιοχή εστίασης $F(i,1)$ είναι s pixels κάτω από την περιοχή εστίασης $F(i-1,1)$, και αμφότερες οι περιοχές έχουν πολλά επικαλυπτόμενα pixels. Οι επάνω και κάτω μη-επικαλυπτόμενες περιοχές των $F(i,1)$ και $F(i-1,1)$ δηλώνονται ως $B(i-1,1)$ και $B(i,1)$ αντίστοιχα, ενώ τα ιστογράμματά τους δηλώνονται ως $H_{B(i-1,1)}$ και $H_{B(i,1)}$ αντίστοιχα. Βασισμένοι στα διαθέσιμα ιστογράμματα $H_{F(i-1,1)}$ και $H_{B(i-1,1)}$, το ιστόγραμμα της περιοχής εστίασης $F(i,1)$ υπολογίζεται από τη σχέση:

$$H_{F(i,1)} = H_{F(i-1,1)} - H_{B(i-1,1)} + H_{B(i,1)}, \quad i = 2, 3, \dots \quad (5.30)$$

Η παραπάνω εξίσωση δείχνει ότι αντί να επαναυπολογίσουμε το ιστόγραμμα για όλα τα pixels σε μία νέα περιοχή διαμέρισης $F(1,j)$, χρειάζεται να υπολογίσουμε μόνο το ιστόγραμμα μίας μικρής περιοχής $A(1,j)$. Αυτό μειώνει σημαντικά το χρόνο υπολογισμού του ιστογράμματος.

Για τις υπόλοιπες περιοχές εστίασης $F(i,j)$, $j=2,3,\dots$, στην οριζόντια γραμμή σάρωσης i , τα ιστογράμματά τους επίσης υπολογίζονται χρησιμοποιώντας την τελευταία σχέση. Δηλαδή, βασισμένοι στα διαθέσιμα ιστογράμματα $H_{F(i,j-1)}$ και $H_{A(i,j-1)}$, το ιστόγραμμα της νέας περιοχής εστίασης $F(i,j)$ υπολογίζεται από τη σχέση:

$$H_{F(i,j)} = H_{F(i,j-1)} - H_{A(i,j-1)} + H_{A(i,j)}, \quad j = 2, 3, \dots \quad (5.31)$$

Η διαδικασία γρήγορης εξαγωγής χρωματικού ιστογράμματος που παρουσιάστηκε παραπάνω, εφαρμόζεται σε εικόνες με διαφορετικές κλίμακες (διαφορετικό μέγεθος). Πειραματικά αποτελέσματα

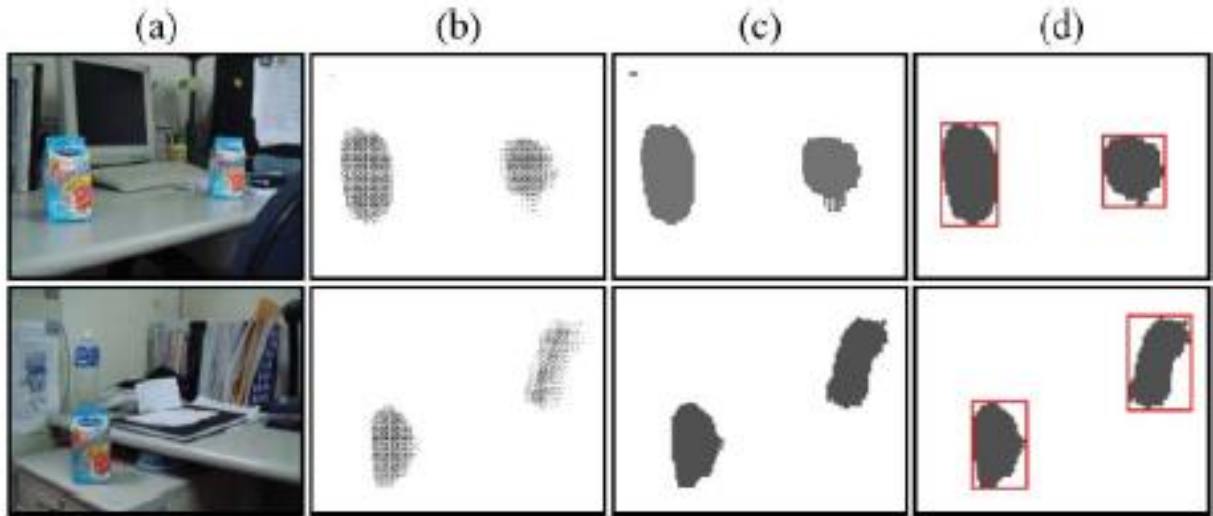
στο Κεφάλαιο V δείχνουν ότι η μέθοδος γρήγορου υπολογισμού μειώνει σημαντικά το χρόνο υπολογισμού του ιστογράμματος.

4.3 Παραγωγή Υποψήφιου Αντικειμένου

Για την πρωτότυπη εικόνα και όλες τις αλλαγμένης κλίμακας εικόνες, το ιστόγραμμα των pixels μέσα στο παράθυρο σάρωσης χρησιμοποιείται σαν ένα χαρακτηριστικό διάνυσμα. Αυτό το χαρακτηριστικό διάνυσμα τροφοδοτεί ένα προ-εκπαιδευμένο TFS-SVMPC. Όταν το TFS-SVMPC είναι εκπαιδευμένο σαν ταξινομητής (classifier) για ανίχνευση αντικειμένων, υπάρχει μόνο μία έξοδος γ αυτού. Η έξοδος παίρνει τις τιμές 1 και -1 , όταν οι είσοδοι είναι χαρακτηριστικά ιστογράμματα για «αντικείμενο (object)» και «μη – αντικείμενο (non-object)» αντίστοιχα. Για παράδειγμα, η Εικόνα 18(a) δείχνει τα παράθυρα αναζήτησης που ανιχνεύτηκαν σαν αντικείμενο (εδώ ένα κουτί) στη δοκιμαστική εικόνα. Η Εικόνα 18(b) δείχνει τις κεντρικές θέσεις αυτών των ανιχνεύσεων. Οι κεντρικές τοποθεσίες σε όλες τις scaled εικόνες, γίνονται rescaled στο αρχικό μέγεθος για περαιτέρω προσδιορισμό. Τοιουτοτρόπως, όλες οι ανιχνεύσεις αναπαρίστανται στην πρωτότυπη εικόνα. Οι Εικόνες 19(a),(b) δείχνουν τις δοκιμαστικές εικόνες και την κατανομή όλων των κέντρων των ανιχνευμένων αντικειμένων (από όλες τις scaled εικόνες) μέσα σε αυτές.



Εικόνα 18 - (a) Δοκιμαστική εικόνα και περιοχές ανίχνευσης υποψήφιων αντικειμένων. (b) Κέντρα των υποψήφιων περιοχών στην πραγματική κλίμακα (original scale)



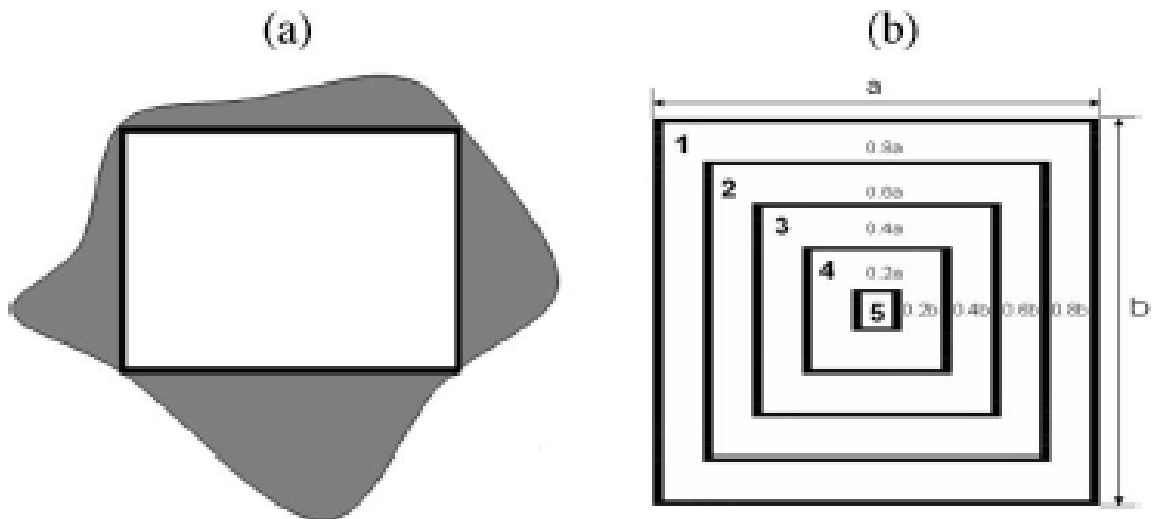
Εικόνα 19 - (a) Δοκιμαστικές εικόνες. (b) Κέντρα των περιοχών υποψήφιων αντικειμένων. (c) Μορφολογικά αρχικά αποτελέσματα. (d) MERs και ανιχνευμένα αντικείμενα μετά το 1ο στάδιο.

Τα περισσότερα πραγματικά αντικείμενα έχουν ανιχνευθεί σε πολλαπλές κοντινές θέσεις, αν και «λάθος συναγερμός» (false alarm – FA) συχνά λαμβάνει χώρα. Για να βρεθεί μια συνδεδεμένη περιοχή για κάθε υποψήφιο και για εξάλειψη των θορύβων, μία αρχική μορφολογική λειτουργία (operation) με δύο 5x5 εργασίες διαστολής ακολουθείται από μία 3x3 εργασία διάβρωσης στα ανιχνευμένα κεντρικά pixels. Η Εικόνα 19(c) δείχνει τις συνδεδεμένες περιοχές μετά την opening operation. Το περίγραμμα κάθε συνδεδεμένης περιοχής βρίσκεται με τη χρήση ενός αλυσιδωτού κώδικα αλγορίθμου. Με βάση το περίγραμμα, βρίσκεται ένα ελάχιστο περιβάλλον ορθογώνιο (MER) κάθε υποψήφιου αντικειμένου, όπως φαίνεται στη Εικόνα 19(d). Αν το μέγεθος του MER είναι μεγαλύτερο από μία προκαθορισμένη περιοχή (εδώ 40x40 pixels), τότε το υποψήφιο αντικείμενο διατηρείται. Διαφορετικά διαγράφεται. Η Εικόνα 19(c) δείχνει ένα υποψήφιο αντικείμενο με μικρή περιοχή που διαγράφεται. Τα υποψήφια αντικείμενα που διατηρήθηκαν προωθούνται στο επόμενο στάδιο για περαιτέρω επαλήθευση.

4.4 Ανίχνευση με βάση Περιγραφείς Ολικούς-Τοπικούς

Το 1^ο στάδιο χρησιμοποιεί την καθολική χρωματική μορφή σαν ένα χαρακτηριστικό ανίχνευσης αντικειμένων. Ωστόσο, περιοχές εστίασης των οποίων το χρωματικό ιστόγραμμα είναι παρόμοιο με το ιστόγραμμα του αντικειμένου που μας ενδιαφέρει πιθανόν να ανιχνευτούν εσφαλμένα ως αντικείμενο, πράγμα που δημιουργεί «λάθος συναγερμό» (FA). Για να διαγράψει τα FA, το 2^ο στάδιο χρησιμοποιεί τοπικά χρωματικά χαρακτηριστικά που λαμβάνουν επιπλέον υπ' όψιν τους τη γεωμετρική κατανομή

του χρώματος. Για να εξαχθεί το τοπικό χρωματικό χαρακτηριστικό (local color feature) κάθε υποψήφιου αντικειμένου από το 1^ο στάδιο, ευρίσκεται το μέγιστο ορθογώνιο (MR) στο εσωτερικό του περιγράμματος κάθε υποψήφιου αντικειμένου. Τότε, το MR διαιρείται σε 5 μη-επικαλυπτόμενες περιοχές. Οι Εικόνες 20(a),(b) δείχνουν το MR και την κατανομή των 5 περιοχών μέσα σε αυτό. Τα όρια των περιοχών 2,3,4 και 5 από scaling του αρχικού (original) MR σε παράγοντα αναλογίας 80%, 60%, 40% και 20% αντίστοιχα. Υπολογίζοντας τις μέσες HS τιμές για κάθε περιοχή, παράγεται ένα δέκα-διάστατο τοπικό χρωματικό χαρακτηριστικό (local color feature). Το τοπικό χρωματικό χαρακτηριστικό τροφοδοτείται σαν είσοδος σε ένα άλλο προ-εκπαιδευμένο TFS-SVMPC για τελική επαλήθευση του υποψήφιου αντικειμένου. Η Εικόνα 21 απεικονίζει τα MRs, τις 5 τοπικές περιοχές τους και το τελικό αποτέλεσμα της ανίχνευσης στην εικόνα της Εικόνα 19. Το χρώμα στην κάθε τοπική (local) περιοχή του μη-αντικειμένου (non-object) είναι διαφορετικό από αυτό του αντικειμένου (object). Με τον τρόπο αυτό, το 2^ο στάδιο διαγράφει επιτυχώς το μη-αντικείμενο (non-object).



Εικόνα 20 - (a) Το MR μέσα στο περίγραμμα. (b) Οι 5 περιοχές μέσα στο MR.



Εικόνα 21 - (a) Τα MER και MR των 2 υποψήφιών περιοχών από το 1ο στάδιο. (b) Οι 5 περιοχές σε κάθε MR. (c) Τελικά αποτελέσματα ανίχνευσης.

5 Πειράματα

5.1 Offline Πειράματα και Αποτελέσματα

Το προτεινόμενο TFS-SVMPC-based σύστημα ανίχνευσης αντικειμένων, εφαρμόστηκε για να ανιχνεύσει διαφορετικά αντικείμενα σε 3 πειράματα. Για ένα αντικείμενο που περιλαμβάνει ένα απλό χρώμα, το σχήμα του εξάγεται συνήθως σαν ένα χαρακτηριστικό ανίχνευσης. Εδώ τα αντικείμενα περιλαμβάνουν ποικίλλα χρώματα σε ανομοιογενείς κατανομές. Η εμφάνιση αυτή καθιστά δύσκολη την εξαγωγή του σχήματος του αντικειμένου ως χαρακτηριστικού ανίχνευσης. Επιπρόσθετα, όπως φαίνεται στις εικόνες 22 – 24, το πολύχρωμο φόντο των εικόνων επιβαρύνει το έργο της ανίχνευσης. Ο HS χώρος χωρίστηκε σε 49 «κελάρια» (bins) κατά την προσέγγιση της ανομοιόμορφης διαμέλισης. Ο αριθμός 49 επιλέχθηκε με βάση της επίδοσης της ανίχνευσης και του μεγέθους του ανιχνευτή εισόδου. Η παράμετρος του κόστους C του γραμμικού SVM για το TFS-SVMPC part learning ορίστηκε σε 2 για όλα τα πειράματα. Για την αποτίμηση της επίδοσης της προτεινόμενης μεθόδου ανίχνευσης αντικειμένων, χρησιμοποιήθηκαν ο βαθμός ανίχνευσης (detection rate - DR) και ο αριθμός των «λάθος συναγερμών» (false alarm - FA).

Ο βαθμός ανίχνευσης (DR) ορίζεται ως η αναλογία του αριθμού των αντικειμένων που ανιχνεύτηκαν επιτυχώς προς τον αριθμό των αντικειμένων που πραγματικά υπήρχαν. Ένας «λάθος συναγερμός» (FA) είναι μία ανίχνευση αντικειμένου όπου δεν υπάρχει πραγματικό αντικείμενο.



Εικόνα 22 - Αποτελέσματα ανίχνευσης από την προτεινόμενη TFS-SVMPC-based μέθοδο στο Πείραμα 1.



Εικόνα 23 - Αποτελέσματα ανίχνευσης από την προτεινόμενη TFS-SVMPC-based μέθοδο στο Πείραμα 2.



Εικόνα 24 - Αποτελέσματα ανίχνευσης από την προτεινόμενη TFS-SVMPC-based μέθοδο στο Πείραμα 3.

Πείραμα 1 (Ανίχνευση Κουτιού – Can Detection): αυτό το πείραμα ανίχνευσε ένα συγκεκριμένο κουτί σε διαφορετικές σκηνές. Η εικόνα 7 δείχνει διάφορες εικόνες του κουτιού σε διαφορετικά φόντα. Για τη συλλογή του εκπαιδευτικού δείγματος, επιλέχτηκαν 200 δείγματα κουτιού (can patterns) από διαφορετικές οπτικές. Κάθε δείγμα μετρήθηκε 45 x 90 pixels. Αυτό το μέγεθος ήταν το ελάχιστο μέγεθος ενδιαφέροντος του κουτιού για να ανιχνευτεί σε μία εικόνα και χρησιμοποιήθηκε ως μέγεθος του παραθύρου σκαναρίσματος. 100 δείγματα μη-αντικειμένων (non-can patterns) επιλέχθηκαν τυχαία από διαφορετικά φόντα. Κάθε non-can (non-object) pattern (δείγμα) μετρήθηκε επίσης 45 x 90 pixels. Ο HS χώρος ήταν ανομοιόμορφα κατανομημένος, σύμφωνα με τα 300 εκπαιδευτικά δείγματα.

Για μία εικόνα 640 x 480 pixels, ο πίνακας Table II δείχνει πώς το CPU time για την εξαγωγή του χαρακτηριστικού ιστογράμματος από ένα image pyramid, χρησιμοποιώντας τη μέθοδο γρήγορου υπολογισμού ιστογράμματος, ήταν 0.23 s. Ο πίνακας Table II δείχνει επίσης πώς η παραδοσιακή μέθοδος υπολογισμού του ιστογράμματος, που επαναυπολογίζει το ιστογράμμα για κάθε νέα περιοχή εστίασης, απαιτήσε 17.02 s.

Ο TFS-SVMPC ταξινομητής στο 1^ο στάδιο εκπαιδεύτηκε από 300 εκπαιδευτικά δείγματα . Το μέγεθος (διάσταση) εισόδου του TFS-SVMPC ήταν 49. Για την εκπαίδευση του TFS-SVMPC, η structure learning parameter μ_{th} ορίστηκε 0.008, το οποίο παράγει 8 κανόνες μετά την εκπαίδευση. Μεγαλύτερος αριθμός μ_{th} παράγει υψηλότερο αριθμό κανόνων. Το τελικό μ_{th} επιλέχθηκε με βάση το συμβιβασμό της επίδοσης και του αριθμού των κανόνων.

Το μέγεθος του προκύπτοντος παραμετρικού διανύσματος \vec{a} , ήτανε 50 x 8 = 400. Μέσα στο PCA, οι αλλαγές εύρους (πλάτους) των ιδιοτιμών δεν ήταν προφανείς. Ωστόσο, το ακόλουθο ποσοστό συσσώρευσης (accumulated rate - AR) χρησιμοποιήθηκε σαν κριτήριο για την επιλογή ιδιοτιμών:

$$AR = \frac{\sum_{i=1}^M \lambda_i}{\text{sum of all eigenvalues}} \quad (6.1)$$

Ο αριθμός των ιδιοτιμών επιλέχθηκε βρίσκοντας το M για το οποίο έχουμε AR > 95 %. Η τιμή του M βρέθηκε ίση με 22, γεγονός που υποδεικνύει ότι το μέγεθος (διάσταση) μειώθηκε από 400 σε 22. Μετά την εκπαίδευση, ο TFS-SVMPC πέτυχε ένα ποσοστό σφάλματος εκπαίδευσης κοντά στο μηδέν.

Ο TFS-SVMPC στο 2^ο στάδιο εκπαιδεύτηκε από το 10-διαστάσεων HS χρωματικό χαρακτηριστικό από το MR των 300 εκπαιδευτικών δειγμάτων (training patterns), όπου το MR κάθε εκπαιδευτικού δείγματος ήταν απλά το ορθογώνιο όριο κάθε δείγματος. Για την εκπαίδευση του TFS-SVMPC, η structure learning parameter μ_{th} ορίστηκε ίση με 0.016, που παράγει 9 κανόνες. Η επιλογή αυτής της μ_{th} βασίστηκε στις κατευθυντήριες γραμμές που περιγράφηκαν παραπάνω. Το μέγεθος του προκύπτοντος παραμετρικού διανύσματος \vec{a} , ήταν 11 x 9 = 99. Στην PCA, επειδή οι 6 μεγαλύτερες ιδιοτιμές ήταν πολύ μεγαλύτερες από τις άλλες, αυτές οι 6 ιδιοτιμές επιλέχθηκαν κατευθείαν. Δηλαδή, το μέγεθος (διάσταση) μειώθηκε από 99 σε 6. Μετά την εκπαίδευση, ο TFS-SVMPC πέτυχε ένα ποσοστό σφάλματος εκπαίδευσης 7.7%.

Στη συνέχεια τα δεδομένα ελέγχου περιελάμβαναν 500 εικόνες, κάθε μία από τις οποίες μετρήθηκε 640 x 480 pixels. Οι εικόνες αυτές περιελάμβαναν συνολικά 629 κουτιά (cans), και κάθε εικόνα μπορούσε να περιλαμβάνει κανένα ή και περισσότερα από ένα κουτιά (cans). Ο πίνακας Table I δείχνει το μέγεθος των TFS-SVMPC για τα δύο στάδια και τα αποτελέσματα της ανίχνευσης για τις 500 αυτές δοκιμαστικές εικόνες. Τα αποτελέσματα έδειξαν ότι ο συνολικός αριθμός των παραμέτρων στους fuzzy κανόνες των δύο εκπαιδευμένων TFS-SVMPCs στα στάδια 1 και 2 ήταν 810 και 199, αντίστοιχα. Ο αριθμός των παραμέτρων (κανόνων) ήταν μικρότερος από τους fuzzy ταξινομητές που χρησιμοποιήθηκαν για σύγκριση. Η Εικόνα 22 δείχνει τα αποτελέσματα της ανίχνευσης για μερικές από

τις δοκιμαστικές εικόνες. Ο μέσος χρόνος ανίχνευσης για μία δοκιμαστική εικόνα ήταν 1.4 s, όπως φαίνεται στον πίνακα Table II.

Πείραμα 2 (Ανίχνευση Κούπας – Cup Detection): Στο πείραμα ανιχνεύεται μία συγκεκριμένη κούπα σε διαφορετικές σκηνές. Η Εικόνα 23 δείχνει διάφορες εικόνες του κουτιού σε διαφορετικά φόντα. Κάθε δείγμα ήταν 45 x 45 pixels. Ο αριθμός των με αντικείμενα (object) και των χωρίς-αντικείμενα (non-object) δειγμάτων, ήταν ίδιος με αυτόν του πειράματος 1. Για τον TFS-SVMPC στο 1^ο στάδιο, η structure learning parameter μ_{th} ορίστηκε 0.018, το οποίο παράγει 7 κανόνες. Το μέγεθος (διάσταση) του προκύπτοντος παραμετρικού διανύσματος \vec{a} μειώθηκε από 350 σε 15 διαστάσεις (dimensions) όταν το AR στο 95% στο PCA. Το ποσοστό εκπαιδευτικού σφάλματος για αυτόν τον TFS-SVMPC ήταν 1%. Για τον TFS-SVMPC στο 2^ο στάδιο, η structure learning parameter μ_{th} ορίστηκε 0.25, το οποίο παράγει 9 κανόνες. Το PCA επέλεξε κατευθείαν τις 5 μεγαλύτερες ιδιοτιμές, που ήταν προφανώς πολύ μεγαλύτερες από τις άλλες. Με άλλα λόγια, το μέγεθος (διάσταση) μειώθηκε από 77 σε 5. Το ποσοστό εκπαιδευτικού σφάλματος για αυτόν τον TFS-SVMPC ήταν 20%.

Τα test data περιλάμβαναν 200 test images, καθεμία από τις οποίες ήταν 640 x 480 pixels. Στις εικόνες αυτές περιλαμβάνονταν 186 κουτιά (cans) συνολικά. Σημειώνουμε πως οι εικόνες περιλάμβαναν μία πράσινη κούπα του καφέ που ήταν ίδια με την κόκκινη κούπα του καφέ που μας ενδιαφέρει, εκτός από το χρώμα. Η ανίχνευση της πράσινης κούπας θεωρήθηκε «Λάθος Συναγερμός» (False Alarm - FA). Ο πίνακας Table III δείχνει το μέγεθος των δύο TFS-SVMPC στα δύο στάδια ανίχνευσης και τις επιδόσεις ανίχνευσης. Η Εικόνα 23 δείχνει τα ικανοποιητικά αποτελέσματα της ανίχνευσης. Ο πίνακας Table II δείχνει το χρόνο ανίχνευσης.

Πείραμα 3 (Ανίχνευση Τσάντας – Bag Detection): Στο πείραμα ανιχνεύεται μία συγκεκριμένη τσάντα σε διαφορετικές σκηνές. Η Εικόνα 23 δείχνει διάφορες εικόνες της τσάντας σε διαφορετικά φόντα. Κάθε δείγμα ήταν 45 x 90 pixels. Για τον TFS-SVMPC στο 1^ο στάδιο, η structure learning parameter μ_{th} ορίστηκε 0.01, το οποίο παράγει 11 κανόνες. Το μέγεθος του προκύπτοντος παραμετρικού διανύσματος \vec{a} μειώθηκε από 550 σε 17 διαστάσεις (dimensions) όταν το AR στο 95% στο PCA. Το ποσοστό εκπαιδευτικού σφάλματος για αυτόν τον TFS-SVMPC ήταν 1%. Για τον TFS-SVMPC στο 2^ο στάδιο, η structure learning parameter μ_{th} ορίστηκε 0.02, το οποίο παράγει 10 κανόνες. Το PCA επέλεξε κατευθείαν τις 5 μεγαλύτερες ιδιοτιμές, που ήταν προφανώς πολύ μεγαλύτερες από τις άλλες. Το ποσοστό σφάλματος εκπαίδευσης μετά την εκπαίδευση για αυτόν τον TFS-SVMPC ήταν 11%.

Τα test data περιλάμβαναν 256 test images, καθεμία από τις οποίες ήταν 640 x 480 pixels. Στις εικόνες αυτές περιλαμβάνονταν 256 τσάντες (bags) συνολικά. Ο πίνακας Table IV δείχνει το μέγεθος των δύο TFS-SVMPC στα δύο στάδια ανίχνευσης και τις επιδόσεις ανίχνευσης. Η Εικόνα 24 δείχνει τα επιτυχή αποτελέσματα της ανίχνευσης. Ο πίνακας Table II δείχνει το χρόνο ανίχνευσης.

5.2 Συγκρίσεις Με Άλλους Ταξινομητές

Η επίδοση του TFS-SVMPC συγκρίθηκε με αυτήν άλλων ταξινομητών, συμπεριλαμβανομένων ενός TS fuzzy neural network (FNN) (the Self-Constructing Neural Fuzzy Inference Network (SONFIN) – το αυτοκατασκευαζόμενο νευρωνικό δίκτυο ασαφούς συμπεράσματος), ενός Gaussian kernel-based SVM, SOTFN-SV και ενός TFS-SVMPC χωρίς PCA (σημειώνεται ως TFS-SVM). Οι δύο TFS-SVMPC ταξινομητές στην προτεινόμενη διαδικασία ανίχνευσης αντικαταστάθηκαν από τους ταξινομητές αυτούς που χρησιμοποιήθηκαν για σύγκριση. Σε έναν TFS-SVM το αρχικό παραμετρικό διάνυσμα \vec{a} είναι απευθείας συντονισμένο με ένα γραμμικό SVM χωρίς PCA. Οι πίνακες Table I, III και IV δείχνουν τις δομές, το συνολικό αριθμό των παραμέτρων των διαφορετικών ταξινομητών και τις επιδόσεις ανίχνευσής τους σε διαφορετικά πειράματα, αντίστοιχα. Συνολικά, τα αποτελέσματα έδειξαν ότι οι επιδόσεις των διαφορετικών ταξινομητών είναι παρόμοιες. Ωστόσο, ο αριθμός των παραμέτρων στον TFS-SVMPC είναι μικρότερος από αυτόν στους άλλους ταξινομητές. Η επίδοση ελέγχου του TFS-SVMPC είναι επίσης καλύτερη από αυτήν των άλλων ταξινομητών. Για τον Gaussian kernel-based SVM, ο συνολικός αριθμός των παραμέτρων είναι μεγάλος. Ο πίνακας Table II δείχνει πως οι χρόνοι ανίχνευσης με τη χρήση του Gaussian kernel-based SVM σε διαφορετικά πειράματα, είναι όλοι μεγαλύτεροι από 20 s, χρόνος πολύ μεγάλος και μη πρακτικός για εφαρμογές πραγματικού χρόνου.

Για τους TS-τύπου FNNs, έχουν προταθεί αρκετές learning approaches. Ωστόσο, αυτές οι προσεγγίσεις είναι βασισμένες στην ελαχιστοποίηση του σφάλματος εκπαίδευσης, που μπορεί να προκαλέσει χαμηλή επίδοση γενίκευσης. Ένας κλασικός TS FNN είναι ένα adaptive-network-based fuzzy inference system (ANFIS). Για την εκπαίδευση στο 1^ο στάδιο, το μέγεθος (διάσταση) της εισόδου είναι 49. Ακόμα και αν υπάρχουν μόνο 2 fuzzy σύνολα σε κάθε μεταβλητή εισόδου, ο αριθμός των κανόνων (rules) στο ANFIS είναι 2^{49} , νούμερο πολύ μεγάλο για πρακτικές εφαρμογές. Για να αντιμετωπίσουμε αυτό το curse-of-dimension πρόβλημα, μία δημοφιλής προσέγγιση είναι η χρήση ενός clustering αλγορίθμου για τον καθορισμό του κατάλληλου αριθμού κανόνων (rules). Ο SONFIN είναι ένας τυπικός TS fuzzy ταξινομητής που χρησιμοποιεί αυτήν την προσέγγιση, και ως εκ τούτου τον χρησιμοποιούμε για σύγκριση. Ο πίνακας Table III δείχνει πως, ενώ η επίδοση στην εκπαίδευση ενός SONFIN είναι καλή, όμως η επίδοσή του σε νέα δεδομένα είναι φτωχή εξαιτίας της μικρής του ικανότητας γενίκευσης.

Για έναν TFS-SVM, επειδή η διάσταση του προκύπτοντος παραμετρικού διανύσματος είναι υψηλή, η εκπαίδευση με αυτό το υψηλών διαστάσεων διάνυσμα αποτυγχάνει να βρει χαρακτηριστικά και προκαλεί χαμηλή επίδοση γενίκευσης, σε αντίθεση με έναν TFS-SVMPC. Επιπρόσθετα, για τη σύγκριση των δοκιμαστικών επιδόσεων, ο πίνακας Table V δείχνει τους εκπαιδευτικού και δοκιμαστικούς CPU χρόνους (CPU times) του TFS-SVMPC και του TFS-SVM στα 3 πειράματα. Τα αποτελέσματα δείχνουν πως οι χρόνοι εκπαίδευσης (training times) του TFS-SVMPC στο 1^ο και 2^ο στάδιο είναι μικρότεροι από αυτούς του TFS-SVM. Οι χρόνοι δοκιμών των 2 ταξινομητών ήταν ακριβώς οι ίδιοι, επειδή χρησιμοποιούν τον ίδιο αριθμό κανόνων και την ίδια rule forms, με μόνη διαφορά στο

προκύπτουν διάνυσμα \vec{a} . Αυτές οι συγκρίσεις, αποδεικνύουν την αποτελεσματικότητα και την αποδοτικότητα της χρήσης της μεθόδου PCA στον TFS-SVMPC. Τέλος, πρέπει να αναφερθεί πως στον TFS-SVMPC, οι διατηρούμενες διαστάσεις στο προκύπτουν μέρος δεν έχουν φυσική σημασία μετά την εφαρμογή της PCA. Για τη διαφύλαξη της φυσικής σημασίας, πρέπει να εφαρμοσθεί μία προσέγγιση επιλογής χαρακτηριστικού που επιλέγει ένα υποσύνολο των αρχικών χαρακτηριστικών. Αυτή η προσέγγιση συνήθως εφαρμόζεται για την εξάλειψη άσχετων χαρακτηριστικών σε μεγάλων διαστάσεων προβλήματα ταξινόμησης.

5.3 Συγκρίσεις Με Άλλες Μεθόδους Ανίχνευσης

Για σύγκριση, δοκιμάστηκαν 3 άλλες μέθοδοι ανίχνευσης αντικειμένων. Αυτές οι μέθοδοι περιλαμβάνουν την **histogram-based template matching (H-TM)**, την **open source computer vision library (OpenCV)** και την **SIFT**. Οι μέθοδοι ανίχνευσης περιγράφονται εν συντομία παρακάτω.

Στη μέθοδο **H-TM**, τα ιστογράμματα εξάγονται από 49 ομοιόμορφα κατανεμημένα «κελάρια» (bins) στον HS χώρο. Για κάθε παράδειγμα, το template του αντικειμένου περιλαμβάνει χαρακτηριστικά ιστογράμματος από όλα τα 200 object training patterns. Ο πίνακας Table VI δείχνει τα αποτελέσματα της ανίχνευσης. Συνολικά, τα αποτελέσματα δείχνουν ότι το προτεινόμενο σύστημα επιτυγχάνει καλύτερη επίδοση από την H-TM μέθοδο. Ένα άλλο μειονέκτημα της H-TM μεθόδου είναι πως η απαιτούμενη μνήμη για την αποθήκευση των templates είναι μεγάλη.

Η μέθοδος **OpenCV**, χρησιμοποιεί απλά Haar-like χαρακτηριστικά και έναν «καταρράκτη» ενισχυμένων δένδρων ταξινομητών σαν στατιστικό της μοντέλο. Ο πίνακας Table VI δείχνει τα αποτελέσματα της ανίχνευσης, τα οποία είναι πολύ «φτωχά» επειδή η μη-ομογενοποιημένη χρωματική κατανομή στην εμφάνιση των αντικειμένων κάνει δύσκολη την εξαγωγή ενός συνηθισμένου Haar-like χαρακτηριστικού για το ίδιο αντικείμενο με διαφορετικές όψεις.

Η μέθοδος **SIFT** και διάφορες παραλλαγές της, όπως η PCA-SIFT και το ιστογράμμο κλίσης τοποθεσίας-προσανατολισμού (gradient location-orientation histogram - GLOH), έχουν προταθεί για ανίχνευση συγκεκριμένων αντικειμένων. Τα χαρακτηριστικά SIFT είναι τοπικά και βασισμένα στην εμφάνιση του αντικειμένου σε ιδιαίτερου ενδιαφέροντος σημεία και είναι αμετάβλητα στο scale και το rotation της εικόνας. Η μέθοδος SIFT χρησιμοποιήθηκε για σύγκριση γιατί έχει χρησιμοποιηθεί ευρέως σε διαφορετικές εφαρμογές ανίχνευσης αντικειμένων. Δοθείσης μίας έγχρωμης εικόνας, η SIFT πρώτα τη μετατρέπει σε μονοχρωματικής έντασης εικόνα και εξάγει σημεία κλειδιά μέσα από μία σειρά εργασιών στη μονοχρωματική εικόνα. Οι περιγραφείς των keypoints δημιουργούνται αντιστοιχίζοντας τους κυρίαρχους προσανατολισμούς σε εντοπισμένα keypoints, με βάση τις κατευθύνσεις της τοπικής κλίσης της εικόνας. Ο πίνακας Table VI δείχνει τα αποτελέσματα ανίχνευσης της SIFT μεθόδου. Τα αποτελέσματα δείχνουν ότι η προτεινόμενη TFS-SVMPC πετυχαίνει καλύτερη επίδοση από τη SIFT μέθοδο. Για την ανάλυση της επίδοσης της μεθόδου SIFT, οι Εικόνες 25(a) έως 25(c) δείχνουν τα

keypoints, κέντρα των περιοχών των υποψήφιων αντικειμένων, και τα σωστά ανιχνευμένα αποτελέσματα, αντίστοιχα, στα πειράματα 1 έως 3. Παρομοίως, η Εικόνα 26 δείχνει τα αποτελέσματα των λάθος ανιχνεύσεων. Τα keypoints που βρέθηκαν στο πείραμα 1 (can) και στο πείραμα 2 (bag) πάρθηκαν από κάποιες όψεις που μπορεί να μην είναι ευδιάκριτες, όπως φαίνεται στην Εικόνα 25, γεγονός το οποίο οδηγεί σε αστοχία. Επιπρόσθετα, η ικανότητα γενίκευσης του template matching ταξινομητή είναι γενικά χαμηλότερη από αυτήν ενός fuzzy ταξινομητή. Σαν αποτέλεσμα, είναι πιο πιθανόν οι SIFT-based μέθοδοι να αποτυγχάνουν να ταιριάζουν ένα object keypoint που δεν είναι αποθηκευμένο στα templates.

Τα keypoints στην κούπα (cup) (πείραμα 2) που πάρθηκαν από διαφορετικές όψεις, είναι πολύ πιο διακριτά από αυτά στο κουτί (can) και στην τσάντα (bag), και επομένως, επιτυγχάνεται ένα υψηλότερο DR. Ο αριθμός των FA στο πείραμα ανίχνευσης της κούπας (cup) είναι υψηλός επειδή ο περιγραφητής των keypoints αποτυγχάνει να διακρίνει τη διαφορά μεταξύ πράσινης και κόκκινης κούπας (cup), όπως φαίνεται στην Εικόνα 26.

Τέλος, για να δείξουμε την επίδραση της μεθόδου ανομοιόμορφης διαμέρισης, πραγματοποιήσαμε μια επιπρόσθετη σύγκριση με τον TFS-SVMPC, χρησιμοποιώντας ένα ιστόγραμμα 49 ομοιόμορφα κατανομημένων «κελαριών» (bins) στο 1^ο στάδιο. Ο αριθμός των κανόνων (rules) στους TFS-SVMPCs είναι ο ίδιο με τα πειράματα 1 έως 3. Ο πίνακας Table VI παρουσιάζει τα αποτελέσματα της ανίχνευσης, που δείχνουν ότι ο TFS-SVMPC που χρησιμοποιεί την μέθοδο ομοιόμορφης διαμέρισης επιτυγχάνει χειρότερα αποτελέσματα από αυτόν που χρησιμοποιεί τη μέθοδο μη-ομοιόμορφης διαμέρισης.

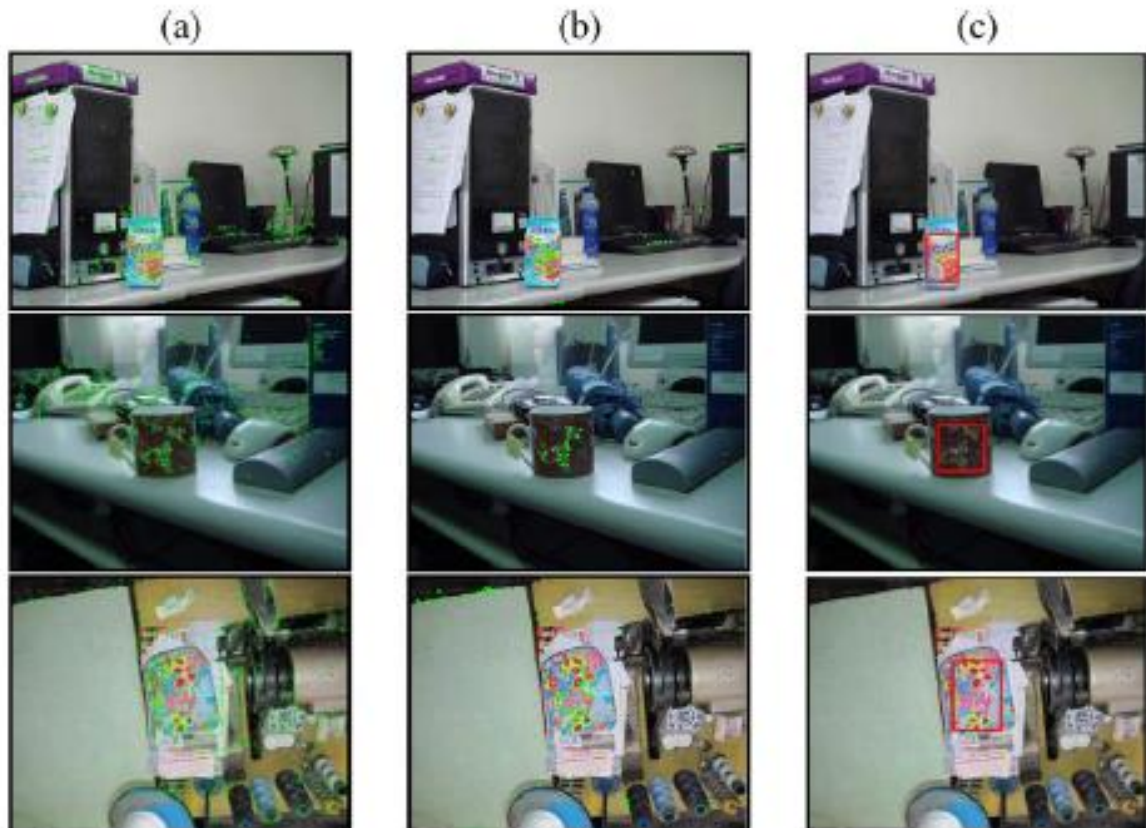
5.4 Πείραμα Ανίχνευσης Αντικειμένων Σε Πραγματικό Χρόνο

Η προτεινόμενη μέθοδος ανίχνευσης χρησιμοποιήθηκε σε ένα σύστημα ανίχνευσης αντικειμένων σε πραγματικό χρόνο, που χρησιμοποιεί μία κάμερα Sony EVI-D70.

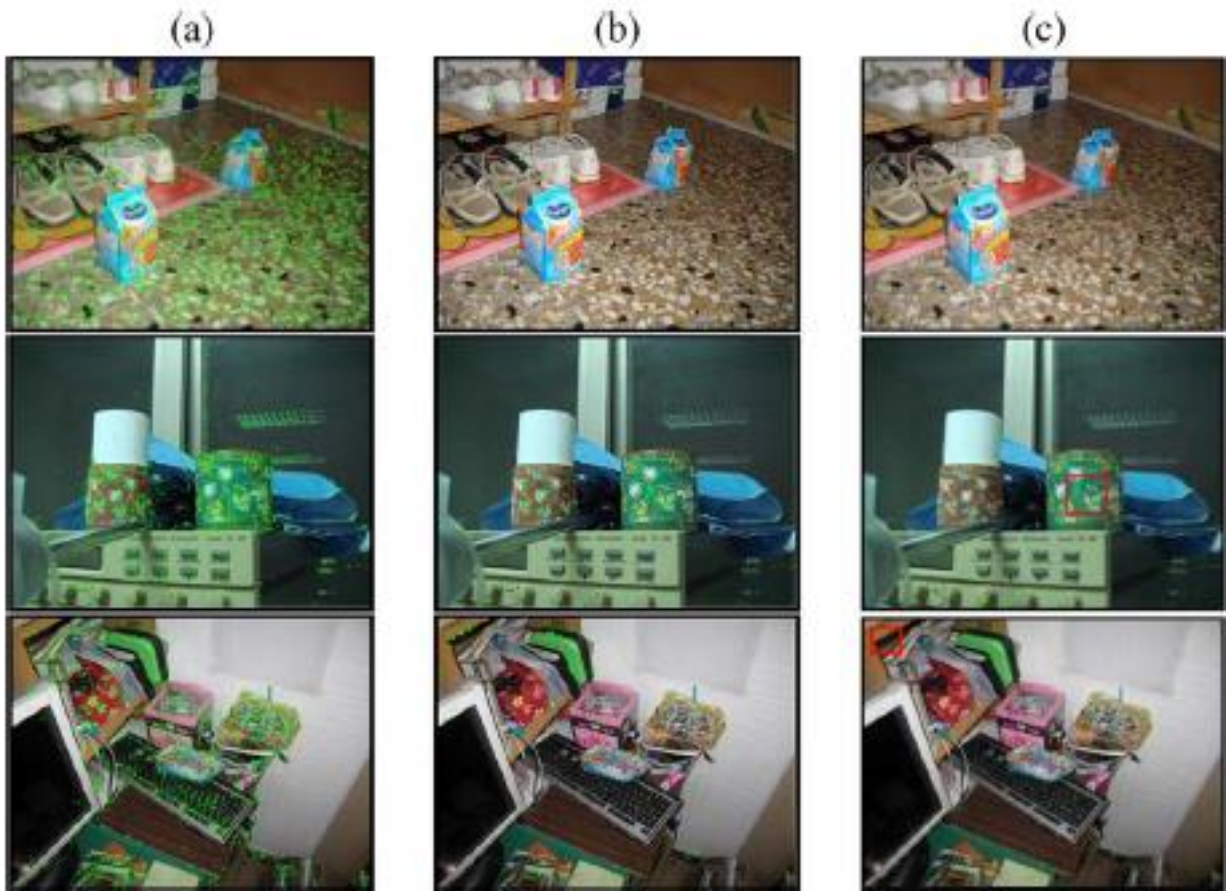
Πείραμα 4: Αυτό το πείραμα δείχνει τα αποτελέσματα της ανίχνευσης σε πραγματικό χρόνο της κούπας (cup) και της τσάντας (bag) σε διαφορετικά περιβάλλοντα. Η έκταση της αναζήτησης της κάμερας κυμάνθηκε από -25° έως 25° για την panning direction και από -20° έως 20° για την titling direction. Το μέγεθος του βήματος κίνησης της κάμερας ήταν 10° και 13° για τις panning και titling directions αντίστοιχα, και υπήρχαν συνολικά 15 directions. Η προτεινόμενη μέθοδος ανίχνευσης χρησιμοποίησε την captured εικόνα (640 x 480 pixels) σε κάθε μία από τις 15 directions διαδοχικά, μέχρι να ανιχνευθεί το συγκεκριμένο αντικείμενο. Χρειάστηκαν 1.4 s για να ανιχνευτεί ένα αντικείμενο στην εικόνα που καλύφθηκε από μία κατεύθυνση της κάμερας. Η Εικόνα 27 δείχνει την captured εικόνα από τις κατευθύνσεις 1, 2, 5 και 6, και την επιτυχώς ανιχνευμένη κούπα στη διαδικασία ανίχνευσης. Η Εικόνα 28 δείχνει την captured εικόνα από τις κατευθύνσεις 1, 6, 8 και 10, και την επιτυχώς

ανιχνευμένη τσάντα στη διαδικασία ανίχνευσης. Ο πίνακας Table VII δείχνει το συνολικό αριθμό των frames πριν ένα αντικείμενο (cup or bag) ανιχνευτεί και το συνολικό αριθμό των FA.

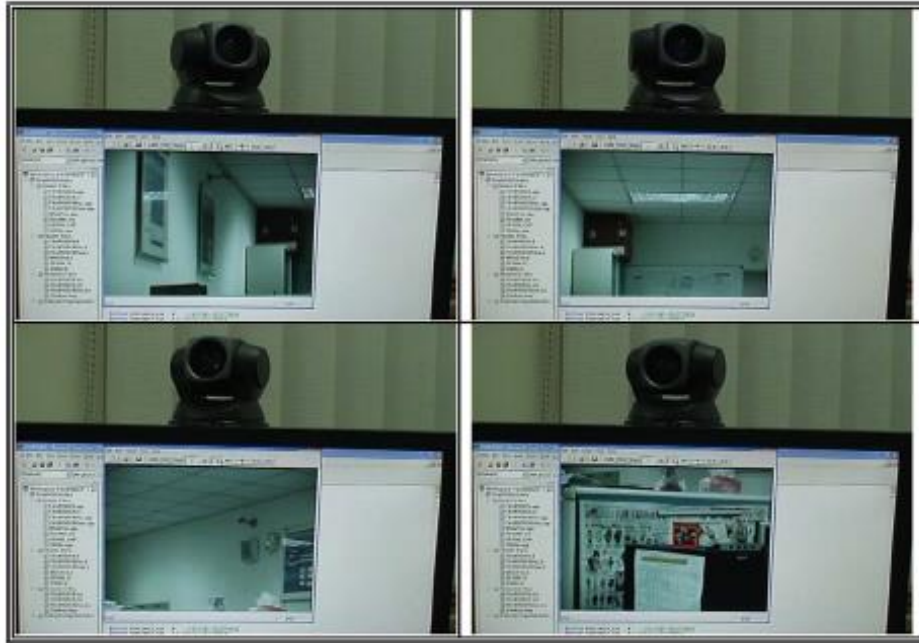
Πείραμα 5: Το πείραμα δείχνει τα αποτελέσματα της ανίχνευσης του κουτιού (can) σε ένα άλλο περιβάλλον με μεγαλύτερη έκταση από -45° έως 45° για την ranning direction. Επιπρόσθετα, το πείραμα δείχνει την κατάσταση όπου το αντικείμενο ήταν πολύ μικρό για να ανιχνευτεί σε μία αρχική διαδικασία ανίχνευσης (π.χ. μικρότερο από το μέγεθος του παραθύρου). Για την κατάσταση αυτή, η κάμερα εκτέλεσε zoom-in λειτουργία, και η όλη διαδικασία ανίχνευσης επαναλήφθηκε μέχρι να ανιχνευθεί το αντικείμενο. Η Εικόνα 29 δείχνει τα αποτελέσματα της ανίχνευσης με zoom-in λειτουργία. Στην εικόνα αυτή, οι 3 πρώτες φωτογραφίες ήτανε captured από τις κατευθύνσεις 1, 6 και 15 στην πρώτη διαδικασία ανίχνευσης. Η κούπα (cup) εμφανίστηκε στην εικόνα που είχε καλυφθεί από την κατεύθυνση 1, αλλά ήταν πολύ μικρή για να ανιχνευτεί. Μετά από μία zoom-in λειτουργία, το σύστημα επαναξεκίνησε μία νέα διαδικασία ανίχνευσης και ανίχνευσε επιτυχώς την κούπα (cup), όπως φαίνεται στην Εικόνα 29. Ο πίνακας Table VII δείχνει το συνολικό αριθμό των frames πριν το κουτί (can) ανιχνευτεί και το συνολικό αριθμό των FA.



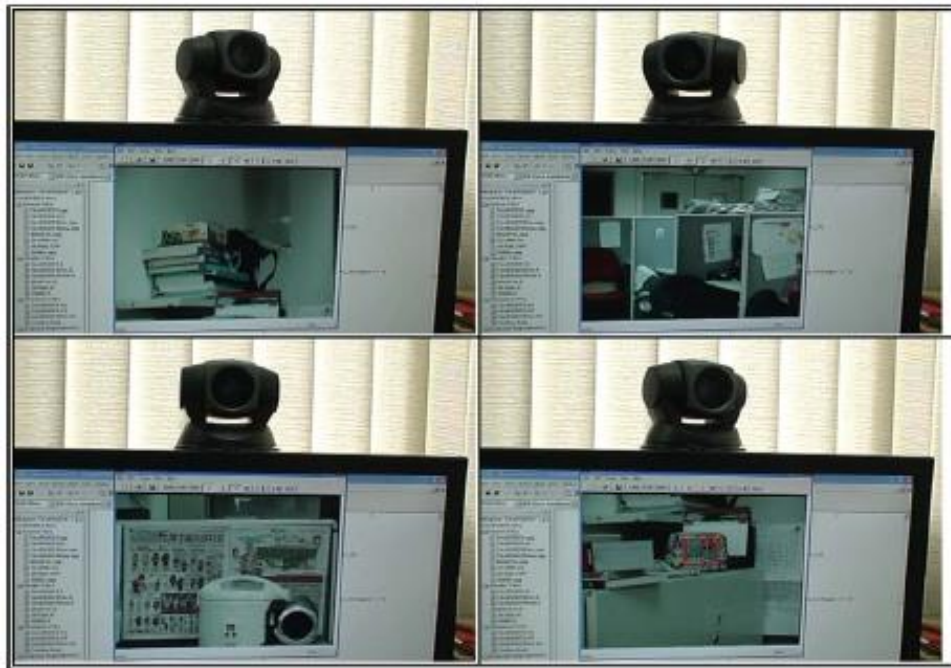
Εικόνα 25 - Αποτελέσματα αναζήτησης με τη χρήση της SIFT μεθόδου. (a) Keypoints. (b) Κέντρα των περιοχών υποψήφιων αντικειμένων. (c) Σωστά ανιχνευμένα αποτελέσματα.



Εικόνα 26 - Αποτελέσματα αναζήτησης με τη χρήση της SIFT μεθόδου. (a) Keypoints. (b) Κέντρα των περιοχών υποψήφιων αντικειμένων. (c) Λάθος ανιχνευμένα αποτελέσματα.



Εικόνα 27 - Διαδικασία αναζήτησης και τελικά αποτελέσματα ανίχνευσης του real-time συστήματος για cup detection.



Εικόνα 28 - Διαδικασία αναζήτησης και τελικά αποτελέσματα ανίχνευσης του real-time συστήματος για bag detection.



Εικόνα 29 - Διαδικασία αναζήτησης και τελικά αποτελέσματα ανίχνευσης του real-time συστήματος για can detection.

6 Συμπεράσματα

Στην εργασία αυτή εξετάσαμε μία νέα μέθοδο ανίχνευσης αντικειμένων με τη χρήση ενός καινοτόμου ταξινομητή (TFS-SVMPC) και χρωματικών χαρακτηριστικών. Τα πειραματικά αποτελέσματα δείχνουν ότι τα πλεονεκτήματα του TFS-SVMPC είναι διπλά. Πρώτον, το μέγεθος του μοντέλου του TFS-SVMPC είναι μικρό, γεγονός που το καθιστά πιο βολικό για λειτουργίες σε πραγματικό χρόνο από το ευρέως χρησιμοποιούμενο Gaussian kernel SVM. Δεύτερον, η χρήση του PCA ακολουθούμενη από ένα γραμμικό SVM διευκολύνει τη διαδικασία εκμάθησης και βοηθάει να βρεθεί ένα σύνολο από προκύπτουσες παραμέτρους με μεγάλη ικανότητα γενίκευσης. Για το χαρακτηριστικό ανίχνευσης αντικειμένου, προτείνεται ένα νέο καθολικό-τοπικό χρωματικό χαρακτηριστικό (global-local color feature). Η προτεινόμενη μέθοδος γρήγορου υπολογισμού του ιστογράμματος κάνει ολόκληρο το σύστημα ανίχνευσης κατάλληλο για λειτουργίες πραγματικού χρόνου. Η επίδοση του προτεινόμενου συστήματος έχει επίσηςδειχθεί ότι είναι καλύτερη από των ταξινομητών που χρησιμοποιήθηκαν για σύγκριση και με άλλες μεθόδους ανίχνευσης. Ο συνδυασμός του TFS-SVMPC με το νέο τύπο χαρακτηριστικού για ανίχνευση αντικειμένων άλλων τύπων και σχημάτων αποτελεί αντικείμενο μελλοντικής έρευνας.

7 Βιβλιογραφία

- [1] Chia-Feng Juang, Guo-Cyuan Chen, “A TS Fuzzy System Learned Through a Support Vector Machine in Principal Component Space for Real-Time Object Detection”, IEEE transactions on industrial electronics, vol. 59, no. 8, August 2012.
- [2] Tristan Fletcher, “Support Vector Machines Explained”, www.cs.ucl.ac.uk/sta_/T.Fletcher/, March 1, 2009.
- [3] Andrew Ng, “CS229 Lecture notes”.
- [4] Takahito Kawanishi, Hiroshi Murase, Shigeru Takagi and Martin Werner, “Dynamic active search for quick object detection with pan-tilt-zoom camera”.
- [5] Καλλιατάκης Γρηγόριος, “Αυτόματη Αναγνώριση Αντικειμένων Εικόνας” 2012.
- [6] Παναγιώτης Β. Περάκης, “Ανίχνευση Προσώπων και Σημείων Ενδιαφέροντος σε Πρόσωπα” Μάιος 2008.
- [7] Θεοχαρίδης Γεώργιος, “Αλγόριθμοι Ανάπτυξης Ταξινομητών Βασισμένων σε Ασαφείς Κανόνες για την Ταξινόμηση Προτύπων” Ιούλιος 2009.
- [8] Καββαδίας Αλέξανδρος, “Αναγνώριση προσώπων με επιλογή των κατάλληλων κύριων συνιστωσών” Μάιος 2014.
- [9] Κοσμίδης Δαμιανός, “Ανάλυση εταιρικών δεδομένων με αλγορίθμους Support Vector Machines (SVM)” Ιούνιος 2008.
- [10] Jacek Kluska, “Analytical Methods in Fuzzy Modeling and Control” Springer, Studies in Fuzziness and Soft Computing, Volume 241, 2009.
- [11] OpenCV Library. www.intel.com/research/mrl/research/opencv.
- [12] Nello Cristianini and John Shawe-Taylor, “An Introduction to Support Vector Machines and Other Kernel-based Learning Methods”, Cambridge University Press, 2000.
- [13] C. Cortes, V. Vapnik, “Support Vector Networks”, Machine Learning, 1995.
- [14] Burges C., “A tutorial on support vector machines for pattern recognition”, In “Data Mining and Knowledge Discovery”. Kluwer Academic Publishers, Boston, 1998, (Volume 2).
- [15] Sirovich L. and Kirby M. (1987), A low-dimensional procedure for the characterization of human faces, The Journal of the Optical Society of America, 4:519-524.
- [16] C. E. Thomaz and G. A. Giraldi. A new ranking method for Principal Components Analysis and its application to face image analysis, Image and Vision Computing, vol. 28, no. 6, pp. 902-913, June 2010.
- [17] T. Takagi και M. Sugeno, “Fuzzy Identification of Systems and its Application to Modeling and Control”, IEEE Transactions on Systems, Man and Cybernetics, Vol. 15, pp. 116 -132, 1985.
- [18] Νικόλαος Παπαμάρκος, “Ψηφιακή Επεξεργασία και Ανάλυση Εικόνας”, Εκδόσεις Β. Γκιούρδας, 2005.
- [19] Lowe, D. G., “Distinctive Image Features from Scale-Invariant Keypoints”, International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.
- [20] P. Viola and M. Jones. Robust Real-time Object Detection. In IEEE ICCV Workshop on Statistical and Computational Theories of Vision, 2001.

8 Παράρτημα – Πίνακες

TABLE I
DETECTION RESULTS OF DIFFERENT CLASSIFIERS IN EXPERIMENT 1

Classifiers		SONFIN[18]	Gaussian SVM	SOTFN-SV [22]	TFS-SVM	TFS-SVMPC
Training: Stage One	Structure	57 rules	49 SVs	44 rules	8 rules	8 rules
	Parameter number	11286	2452	2246	801	801
	Training error rate	0%	0.67%	2%	1.67%	0%
Training: Stage Two	Structure	52 rules	69 SVs	12 rules	9 rules	9 rules
	Parameter number	2184	761	146	199	199
	Training error rate	1.3%	7.7%	11.3%	7.7%	7.7%
Test Results	FA number	579	337	122	78	60
	DR	47.69%	88.87%	88.4%	86.01%	89.83%

TABLE II
AVERAGE HISTOGRAM EXTRACTION TIME AND TOTAL DETECTION TIME FOR A TEST IMAGE USING DIFFERENT METHODS OR CLASSIFIERS IN DIFFERENT EXPERIMENTS, WHERE THE FAST FEATURE COMPUTATION METHOD IS USED IN BOTH GAUSSIAN SVM AND TFS-SVMPC-BASED DETECTION SYSTEMS

Methods/classifiers	Feature Extraction Time		Total Detection Time	
	Original Computation	Proposed Computation	Gaussian SVM	TFS-SVMPC
Experiment 1	17.02 sec.	0.23 sec.	26.3 sec.	1.4 sec.
Experiment 2	17.02 sec.	0.23 sec.	27.7 sec.	1.8 sec.
Experiment 3	17.02 sec.	0.23 sec.	21.1 sec.	1.9 sec.

TABLE III
DETECTION RESULTS OF DIFFERENT CLASSIFIERS IN EXPERIMENT 2

Classifiers		SONFIN[18]	Gaussian SVM	SOTFN-SV [22]	TFS-SVM	TFS-SVMPC
Training: Stage one	Structure	33 rules	53 SVs	42 rules	7 rules	7 rules
	Parameter number	6534	2652	2142	701	701
	Training error rate	0%	0.3%	1%	0%	1%
Training: Stage two	Structure	19 rules	150 SVs	14 rules	7 rules	7 rules
	Parameter number	798	1652	170	155	155
	Training error rate	4%	17.3%	16%	20%	20%
Test Results	FA number	444	144	121	69	59
	DR	18.8%	91.4%	86.6%	54.30%	91.40%

TABLE IV
DETECTION RESULTS OF DIFFERENT CLASSIFIERS IN EXPERIMENT 3

Classifiers		SONFIN[18]	Gaussian SVM	SOTFN-SV [22]	TFS-SVM	TFS-SVMPC
Training: Stage one	Structure	69 rules	27 SVs	46 rules	11 rules	11 rules
	Parameter number	13662	1352	2348	1101	1101
	Training error rate	0%	0%	1.33%	0%	0.33%
Training: Stage two	Structure	27 rules	46 SVs	18 rules	10 rules	10 rules
	Parameter number	1134	508	218	221	221
	Training error rate	0.67%	0.67%	9%	11%	11%
Test Results	FA number	163	164	569	316	213
	DR	75.2%	70%	49.2%	41.2%	80%

TABLE V
COMPARISONS BETWEEN TFS-SVM AND TFS-SVMPC IN TRAINING AND TEST CPU TIME IN EXPERIMENTS 1 TO 3

Experiments	Experiment 1		Experiment 2		Experiment 3	
	TFS-SVM	TFS-SVMPC	TFS-SVM	TFS-SVMPC	TFS-SVM	TFS-SVMPC
Training Stage1	23.69 sec.	2.82 sec.	22.64 sec.	1.23 sec.	28.62 sec.	0.91 sec.
Training Stage2	3.82 sec.	2.18 sec.	6.36 sec.	1.00 sec.	2.84 sec.	1.09 sec.
Test time	0.46 sec.	0.46 sec.	0.47 sec.	0.47 sec.	0.47 sec.	0.47 sec.

TABLE VI
DETECTION RESULTS OF DIFFERENT METHODS IN EXPERIMENTS 1 TO 3

Methods		H-TM [9]	OpenCV [39]	SIFT [10]	TFS-SVMPC (uniform)	TFS-SVMPC
Experiment 1	FA number	122	4793	258	232	60
	DR	85.85%	77.42%	65.66%	88.55%	89.83%
Experiment 2	FA number	110	4343	128	173	59
	DR	81.72%	69.35%	90.86%	78.49%	91.40%
Experiment 3	FA number	273	3869	183	195	213
	DR	89%	70.3%	70.8%	38%	80%

TABLE VII
REAL-TIME DETECTION RESULTS IN EXPERIMENTS 4 AND 5

Object	#Frame number	#False alarm number	Object detected?
Cup (Exp. 4)	6	0	Yes
Bag (Exp. 4)	10	0	Yes
Can (Exp. 5)	16	0	Yes