



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης  
Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«ΑΝΙΧΝΕΥΣΗ ΚΑΙ ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΣΩΠΟΥ ΑΠΟ ΓΩΝΙΑ 90 ΜΟΙΡΩΝ»

**Όνοματεπώνυμο:** Παγούνης Ανδρέας (ΑΜ: 1098)

**Επιβλέπων καθηγητής:** Τριανταφυλλίδης Γεώργιος

**Επιτροπή Αξιολόγησης:**

**Ημερομηνία παρουσίασης:**

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστίσω θερμά τον Καθηγητή μου Κο Τριανταφυλλίδη για την επίβλεψη της πτυχιακής εργασίας. Χωρίς την διορατικότητά του και τής άμεσες απαντήσεις του σε κάθε ερώτηση ή πρόβλημα η εργασία αυτή δεν θα ήταν δυνατόν να ολοκληρωθεί.

## **ABSTRACT**

Over the last ten years, face recognition has become a popular area of research in computer vision and one of the most successful applications of image analysis and. Because of the nature of the problem, not only computer science researchers are interested in it, but neuroscientists and psychologists also. Side-view face recognition is a challenging problem with many applications. Especially in real-life scenarios where the environment is uncontrolled, is an important task for face recognition. In this paper we discuss the use of side view face detection and recognition techniques and how they can be used in applications. Our aim is to recognize people from an angle of 90 degrees (as they pass through a door). Here, we compare available databases appropriate for this task, using current available datasets, methods and techniques for profile face detection and recognition.

## ΣΥΝΟΨΗ

Κατά τη διάρκεια των τελευταίων δέκα ετών, η αναγνώριση προσώπου έχει γίνει ένας δημοφιλής τομέας έρευνας στην όραση υπολογιστών και μια από τις πιο επιτυχημένες εφαρμογές στην ανάλυση εικόνας. Λόγω της φύσης του προβλήματος, ενδιαφέρον δείχνουν όχι μόνο οι ερευνητές πληροφορικής, αλλά και επιστήμονες νευρολογίας και ψυχολογίας επίσης. Η αναγνώριση προσώπου από γωνία 90 μοιρών είναι ένα πρόβλημα με πολλές εφαρμογές. Ειδικά σε πραγματικές συνθήκες όπου το περιβάλλον είναι ανεξέλεγκτο, η αντιμετώπιση θέτει του είναι ένας σημαντικός στόχος για την αναγνώριση προσώπου. Σε αυτή την εργασία συζητάμε τη χρήση των τεχνικών ανίχνευσης και αναγνώρισης προσώπου από γωνία 90 μοιρών, και πώς αυτές μπορούν να χρησιμοποιηθούν στις εφαρμογές. Ο στόχος μας είναι να αναγνωρίσουμε τους ανθρώπους από μια γωνία 90 μοιρών (π.χ. καθώς περνούν μέσω μιας πόρτας), χρησιμοποιώντας τις διαθέσιμες βάσεις δεδομένων, και τις κατάλληλες μεθόδους και τεχνικές για αυτόν τον στόχο.

## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ABSTRACT.....	3
ΣΥΝΟΨΗ .....	4
ΠΙΝΑΚΑΣ ΣΧΗΜΑΤΩΝ.....	5
ΛΙΣΤΑ ΕΙΚΟΝΩΝ.....	8
ΛΙΣΤΑ ΠΙΝΑΚΩΝ.....	10
1. ΕΙΣΑΓΩΓΗ.....	11
2. ΔΙΑΘΕΣΗΜΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ.....	13
2.1. Γενικά.....	13
2.2. Βάσεις Δεδομένων.....	14
2.2.1. <i>CMU multi-PIE</i> .....	15
2.2.2. <i>UMIST</i> .....	15
2.2.3. <i>GUARANTEE</i> .....	17
3. ΑΝΙΧΝΕΥΣΗ ΠΡΟΣΩΠΩΝ.....	18
3.1. Γενικά.....	18
3.2 Περιγραφή του προβλήματος της Ανίχνευσης Προσώπου.....	18
3.3 Μέθοδοι .....	18
3.4 Μικρό Ιστορικό των κυριότερων Μεθόδων Ανίχνευσης.....	20
3.5 Μέθοδοι Βασισμένες στην Εμφάνιση.....	21
3.5.1 <i>Δείγματα προσώπων και μη-προσώπων</i> .....	21
3.5.2 <i>Χρησιμοποιούμενοι Ταξινομητές</i> .....	23
3.5.3 <i>Αναπαράσταση Προσώπων</i> .....	23
3.5.4 <i>Προεπεξεργασία</i> .....	24
3.5.5 <i>Αναζήτηση στον Χώρο και την Κλίμακα</i> .....	25
3.5.6 <i>Μετεπεξεργασία</i> .....	25
3.6 Ανίχνευση Προσώπου κατά Viola & Jones.....	26
3.6.1 <i>Υπολογισμός Χαρακτηριστικών</i> .....	26

3.6.2	Επιλογή Χαρακτηριστικών με την AdaBoost.....	28
3.6.3	Κατασκευή του Αδύναμου Ταξινομητή.....	30
3.6.4	Ταξινόμηση με έναν Καταρράκτη Ταξινομητών.....	30
3.7	Ανίχνευση Προσώπου κατά Lienhart.....	32
3.8	Σύνοψη του Συστήματος Ανίχνευσης Προσώπου.....	34
3.9	Μέθοδος Haar + AdaBoost : Συμπεράσματα.....	34
4.	ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ ΑΝΙΧΝΕΥΣΗΣ ΠΡΟΣΩΠΟΥ.....	36
4.1	Η Βιβλιοθήκη OpenCV.....	36
4.2	Ανίχνευση Προσώπων με την OpenCV .....	36
4.2.1	Η Εκπαίδευση του Ταξινομητή.....	37
4.2.2	Η Συνάρτηση Ανίχνευσης.....	37
4.2.3	Οι Μέθοδοι Ανίχνευσης.....	40
4.3	Εκπαίδευση Viola & Jones .....	41
4.4	Πειραματικό μέρος.....	43
4.4.1	Έρευνα Παραμέτρων .....	44
4.4.2	Το φαινόμενο των διαφορετικών αποτελεσμάτων μεταξύ αριστερών και δεξιών προφίλ.....	49
4.4.3	Επανεκπαιδευμένο Σύστημα .....	51
5.	ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΣΩΠΟΥ - ΤΕΧΝΙΚΕΣ – ΠΡΟΒΛΗΜΑΤΑ.....	54
5.1.	Αναγνώριση προσώπων.....	54
5.2	ΕΦΑΡΜΟΓΕΣ ΑΝΑΓΝΩΡΙΣΗΣ ΠΡΟΣΩΠΩΝ.....	56
5.3	Τεχνικές αναγνώρισης προσώπων.....	61
5.3.1	Η Τεχνική των Eigenfaces.....	61
5.3.2	Αναγνώριση προσώπων με LDA.....	64
5.3.3	Ο Αλγόριθμος EGM.....	65
5.3.3.1	Εισαγωγή.....	65
5.3.3.2	Ο κλασσικός αλγόριθμος EGM.....	65
6.	ΥΛΟΠΟΙΗΣΗ ΑΝΑΓΝΩΡΙΣΗΣ ΠΡΟΣΩΠΟΥ ΜΕ Eigenfaces.....	69

6.1 Πειραματικό Μέρος.....	69
6.2 Προετοιμασία για το Eigenface.....	69
6.3 Επεξήγηση κώδικα αναγνώρισης.....	71
6.4 Αποτελέσματα του κώδικα με την βάση δεδομένων UMIST.....	81
6.5 Τρόποι Βελτίωσης του Eigenface.....	83
7. ΣΥΜΠΕΡΑΣΜΑΤΑ.....	84
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	86

## Λίστα εικόνων

<b>Εικόνα 2.1:</b>	CMU multi-PIE Database.....	15
<b>Εικόνα 2.2:</b>	UMIST Database.....	16
<b>Εικόνα 2.3:</b>	GUARANTEE Database.....	17
<b>Εικόνα 3.1:</b>	Παραδείγματα ανίχνευσης προσώπων.....	21
<b>Εικόνα 3.2:</b>	Πρόσωπα που χρησιμοποιούνται στο σύνολο των θετικών δειγμάτων.....	22
<b>Εικόνα 3.3:</b>	Μη πρόσωπα που χρησιμοποιούνται στο σύνολο των αρνητικών δειγμάτων.....	22
<b>Εικόνα 3.4:</b>	Τεχνητός δημιουργούμενα εικονικά δείγματα προσώπων .....	23
<b>Εικόνα 3.5:</b>	Διάφορες αναπαραστάσεις στηριγμένες στον τεμαχισμό σε μπλοκ.....	24
<b>Εικόνα 3.6:</b>	Διαδικασίες προ-επεξεργασίας εικόνας προσώπου .....	24
<b>Εικόνα 3.7:</b>	Συγχώνευση πολλαπλών ανιχνεύσεων και απαλοιφή των λανθασμένων.....	25
<b>Εικόνα 3.8:</b>	Μορφή των χαρακτηριστικών Viola & Jones .....	27
<b>Εικόνα 3.9:</b>	Αναπαράσταση της "Εικόνας Ολοκλήρωμα".....	27
<b>Εικόνα 3.10:</b>	Τα 2 κυριότερα χαρακτηριστικά εφαρμοσμένα σε ένα τυπικό πρόσωπο.....	30
<b>Εικόνα 3.11:</b>	Υπολογισμός της τιμής χαρακτηριστικών πάνω σε πρόσωπα και μη-πρόσωπα.....	30
<b>Εικόνα 3.12:</b>	Σχηματική παράσταση μιας ανίχνευσης με καταρράκτη ταξινομητών.....	31
<b>Εικόνα 3.13:</b>	Εκτεταμένο σύνολο στραμμένων Haar χαρακτηριστικών.....	32
<b>Εικόνα 3.14:</b>	Ένα δένδρο ταξινόμησης CART και ένας καταρράκτης.....	33
<b>Εικόνα 4.1:</b>	OpenCV's internal detection rectangles. To see these, use <code>min_neighbors = 0</code> .....	39
<b>Εικόνα 4.2:</b>	Παράδειγμα .txt αρχείου θετικών παραδειγμάτων. ....	41
<b>Εικόνα 4.3:</b>	Παράδειγμα εκπαίδευσης ως το στάδιο "STAGE 18". ....	43
<b>Εικόνα 4.4:</b>	ScaleFactor = 1.05 , windowSize = 260.....	45
<b>Εικόνα 4.5:</b>	ScaleFactor = 1.05 , minNeighbors = 1.....	45
<b>Εικόνα 4.6:</b>	minNeighbors = 1 , windowSize = 260.....	46
<b>Εικόνα 4.7:</b>	ScaleFactor = 1.05 , windowSize = 80.....	47
<b>Εικόνα 4.8:</b>	ScaleFactor = 1.05 , minNeighbors = 1.....	47
<b>Εικόνα 4.9:</b>	minNeighbors = 1 , windowSize = 80.....	48
<b>Εικόνα 4.10:</b>	ScaleFactor = 1.001 , windowSize = 200.....	48
<b>Εικόνα 4.11:</b>	ScaleFactor = 1.05 , minNeighbors = 1.....	49
<b>Εικόνα 4.12:</b>	minNeighbors = 1 , windowSize = 2.....	49
<b>Εικόνα 4.13:</b>	Some examples about the differences between the right and the left profile images.....	50
<b>Εικόνα 5.1:</b>	Αρχιτεκτονική συστημάτων αναγνώρισης προσώπων και εκφράσεων.....	56
<b>Εικόνα 5.2:</b>	(α) και (β) Συσκευές καταγραφής βάθους.....	57
<b>Εικόνα 5.3:</b>	Παραδείγματα ενδοατομικών παραλλαγών (intrapersonal variations), δηλαδή παραλλαγών που εμφανίζονται σε διαφορετικές εικόνες του ίδιου προσώπου. Η εμφάνιση ενός προσώπου μπορεί να μεταβληθεί σημαντικά λόγω παραγόντων όπως η πόζα, ο φωτισμός, οι εκφράσεις, το μακιγιάζ, το χτένισμα, η χρήση γυαλιών, καπέλων, κτλ, αλλά και λόγω της φυσιολογικής γήρανσης του ανθρώπου.....	60
<b>Εικόνα 5.4 :</b>	Παραδείγματα έγχρωμων εικόνων του προσώπου και των αντίστοιχων χαρτών βάθους. Στους χάρτες βάθους οι πιο σκούρες αποχρώσεις του γκρι αντιστοιχούν σε σημεία του προσώπου που απέχουν μικρότερη απόσταση από το επίπεδο της κάμερας, ενώ οι πιο ανοιχτές αποχρώσεις αντιστοιχούν σε σημεία που απέχουν μεγαλύτερη απόσταση από την κάμερα.....	61
<b>Εικόνα 5.5:</b>	Εικόνες ιδιοδιανυσμάτων (eigenfaces).....	63
<b>Εικόνα 5.6:</b>	Ανακατασκευή του προσώπου από τις εικόνες ιδιοδιανυσμάτων.....	64



<b>Εικόνα 5.7:</b>	Τα δήματα του EGM.....	68
<b>Εικόνα 6.1:</b>	Τα δύο αρχεία εισαγωγής για το πρόγραμμα Eigenface: train.txt και test.txt.....	71
<b>Εικόνα 6.2:</b>	Η συνάρτηση main() του προγράμματος Eigenface.....	72
<b>Εικόνα 6.3:</b>	Η βοηθητική συνάρτηση printUsage().....	73
<b>Εικόνα 6.4:</b>	Η συνάρτηση learn(). Φάση εκμάθησης του Eigenface.....	74
<b>Εικόνα 6.5:</b>	Η συνάρτηση loadFaceImgArray (). Φορτώνει τις εικόνες και τους αριθμούς ταυτότητας προσώπων και για τις φάσεις εκμάθησης και αναγνώρισης.....	75
<b>Εικόνα 6.6:</b>	Η συναρτηση doPCA (). Βρίσκει PCA subspace.....	76
<b>Εικόνα 6.7:</b>	Η συνάρτηση storeTrainingData(). Αποθηκεύει τα στοιχεία της εκπαίδευσης ως αρχείο XML.....	77
<b>Εικόνα 6.8:</b>	Η συνάρτηση recognition(). Η φάση αναγνώρισης του προγράμματος Eigenface.....	79
<b>Εικόνα 6.9:</b>	Οι συναρτήσεις του OpenCV' για τη φόρτωση των δεδομένων εκπαίδευσης από το αρχείο XML.....	80
<b>Εικόνα 6.10:</b>	Η συναρτηση findNearestNeighbor (). Υπολογίζει την απόσταση από την προβαλλόμενη εικόνα δοκιμής σε κάθε προβαλλόμενο παράδειγμα εκπαίδευσης για να βρεί την πιο κοντινή εικόνα εκπαίδευσης.....	81
<b>Εικόνα 6.11:</b>	Στιγμιότυπο δοκιμής του κώδικα με την βάση UMIST.....	83

## **Λίστα πινάκων**

<b>Πίνακας 2.1.</b> Βασεις δεδομενων με εικονες απο γωνια 90 μοιρων.....	13
<b>Πίνακας 2.2.</b> Side-view Face Databases.....	14

# Κεφάλαιο 1

## 1.Εισαγωγή

Η αναγνώριση προσώπου (face recognition) είναι μια ευρέως χρησιμοποιημένη βιομετρική τεχνική με πολλά πλεονεκτήματα. Τα τελευταία χρόνια, πολλές εφαρμογές συμπεριλαμβανομένων των συστημάτων παρακολούθησης, των έξυπνων σπιτιών, ή οποιασδήποτε εφαρμογής που εξετάζει την ταυτοποίηση ανθρώπων μέσω της χρήσης βίντεο, αντιμετωπίζουν την αναγνώριση ως αρχική βιομετρική. Η αναγνώριση προσώπου όμως έχει σημαντικά πλεονεκτήματα σε σχέση με τις άλλες τεχνολογίες της βιομετρίας, όπως η αναγνώριση δακτυλικού αποτυπώματος, η αναγνώριση φωνής, η αναγνώριση της ίριδας του ματιού και η αναγνώριση υπογραφής: είναι διαδικασία πιά φυσική, λιγότερο αδιάκριτη και ενοχλητική και ευκολότερη στη χρήση της [1].

Για τον άνθρωπο η αναγνώριση προσώπων είναι μία καθημερινή, φυσική και εύκολη σχετικά διαδικασία. Αντίθετα, η αναγνώριση προσώπων με τη βοήθεια ενός ευφυούς συστήματος έχει αποδειχθεί μία δύσκολη προσπάθεια, ειδικά σε μη ελεγχόμενα περιβάλλοντα, όπου η θέση παρατήρησης, ο φωτισμός, οι μορφασμοί και τα αντικείμενα ή αξεσουάρ που αποκρύπτουν το πρόσωπο, ποικίλουν χαρακτηριστικά. Παρολαυτά, η απόδοση των συστημάτων αναγνώρισης προσώπου έχει βελτιωθεί σημαντικά.

Η αναγνώριση προσώπου γενικά περιλαμβάνει δύο στάδια:

1. Η ανίχνευση προσώπου: όπου ψάχνουμε μια φωτογραφία για να βρεθεί οποιοδήποτε πρόσωπο, και κατόπιν επεξεργασίας καθαρίζουμε το πρόσωπο επάνω στην εικόνα για ευκολότερη αναγνώριση.
2. Αναγνώριση προσώπου: όπου αυτός που ανίχνευσε και επεξεργάστηκε το πρόσωπο, το συγκρίνει με μια βάση δεδομένων των γνωστών προσώπων, για να αποφασίσει ποιο από αυτά είναι.

### Ανίχνευση Προσώπου

Ενώ η ανίχνευση προσώπου είναι ένας σημαντικός και ενεργός τομέας στην έρευνα της ορασης υπολογιστών, οι περισσότερες από τις εφαρμογές τώρα απαιτούν όχι μόνο ακρίβεια αλλά και αποδοτικότητα. Συχνά, για να συνδυάσει αυτούς τους δύο παράγοντες ικανοποιητικά, ένα σύστημα ανίχνευσης λαμβάνει υπόψη μόνο μια συγκεκριμένη κλάση από αντικείμενα στόχους, ακόμα κι αν ο περιορισμός αυτός μπορεί να περιορίσει την πρακτική του χρήση. Οι [2], Viola και Jones προτείνουν ένα αποτελεσματικό σχέδιο χρησιμοποιώντας το AdaBoost για να ανιχνεύσουν τα πρόσωπα μέσω ενός ωθημένων αλληλοκαλυψεων. Το μοντέλο τους έχει προτρέψει το ιδιαίτερο ενδιαφέρον για περαιτέρω έρευνα και χρήση της ώθησης των αλγορίθμων και των δομών αλληλοκαλυψεων για τη γρήγορη ανίχνευση αντικειμένων. Παραδείγματος χάριν, ο Li et Al. [6] ανέπτυξε ένα σύστημα για την ανιχνεύση προσώπων από γωνία 90 μοιρών με τη χρησιμοποίηση μιας χονδροειδής--λεπτής, απλής--σύνθετης αρχιτεκτονικής. Διαιρούν "από γωνία" πρόσωπα σε εννέα κατηγορίες, και αναφέρουν ότι ο προκύπτων ανιχνευτής απαιτεί περίπου τρεις φορές το χρόνο υπολογισμού σε σχέση με των Viola και Jones [2,3] για να ανιχνεύσει τα εννέα είδη "από γωνία" προσώπων. [7]

## Αναγνώριση Προσώπου

Στις περισσότερες από τις μεθόδους αναγνώρισης η απόδοση του συστήματος εξαρτάται από την ανίχνευση. Επομένως είναι ένα σημαντικό ζήτημα να ανιχνεύουμε πρόσωπα με μεγάλη ακρίβεια. Υπάρχουν πολλοί τρόποι να ανιχνευθεί ένα πρόσωπο κάποιοι είναι ευκολοί και κάποιοι δύσκολοι. Εμείς θα χρησιμοποιήσουμε την τεχνική της ανίχνευσης προσώπων σε αβίαστες σκηνές που είναι βασισμένη στη μεθοδο "Robust Real-Time Face Detection" των Viola & Jones [3]. Χρησιμοποιώντας ένα καταρράκτη " αδύνατων - classifiers" με τα απλά Haar features, μπορούμε - μετά από εκπαίδευση - να παράγουν εντυπωσιακά αποτελέσματα

Στόχος της παρούσας εργασίας είναι η ανάπτυξη μίας εφαρμογής, που θα αναγνωρίζει πρόσωπα απο γωνία 90 μοιρών. Προκειμένου να διατηρήσουμε τη μυστικότητα των ανθρώπων, θα περιορίσουμε το φωτισμό και τη γωνία λήψης των εικόνων, χρησιμοποιώντας μόνο φωτογραφίες που αποκρύπτουν μεγάλο μέρος των χαρακτηριστικών ενός προσώπου.

Στο κεφάλαιο 2 θα αναφέρουμε τις διαθέσιμες βάσεις δεδομένων, με εικόνες απο γωνία 90 μοιρών, που υπάρχουν στο διαδίκτυο, θα τις συγκρίνουμε μεταξύ τους και θα αναλύσουμε αυτές τις οποίες θα χρησιμοποιήσουμε..

Στα κεφάλαια 3 & 4 παρουσιάζονται τα προβλήματα της ανίχνευσης προσώπου, αναφέρουμε διάφορες μεθόδους ανίχνευσης, αναλύουμε τη μέθοδο των Viola & Jones και υλοποιούμε ένα σύστημα ανίχνευσης προσώπου, "εκπαιδεύοντας" το με τη μέθοδο των Haar Like Features.

Στα κεφάλαια 5 & 6 αναφέρουμε τις τεχνικές και τα προβλήματα που αντιμετωπίζουμε στην αναγνώριση προσώπων, αναλύουμε τους αλγόριθμους αναγνώρισης προσώπου που υπάρχουν και υλοποιούμε ένα σύστημα αναγνώρισης προσώπου με τη μέθοδο των Eigenfaces.

Και στο κεφάλαιο 7 παραθέτουμε τα συμπεράσματά μας.

## Κεφαλαίο 2

### Διαθέσιμες Βάσεις Δεδομένων

#### 2.1 Γενικά

Υπάρχουν διάφορες βάσεις δεδομένων προσώπου που περιέχουν εικόνες προφίλ. Μερικές απ' αυτές μπορείτε να δείτε στον πίνακα 1. Οι περισσότερες από τις διαθέσιμες βάσεις δεδομένων συλλέγονται σε ελεγχόμενες τοποθετήσεις όπως το ομοιόμορφο υπόβαθρο, οι τεχνητές αλλαγές φωτισμού ή περιορισμένες παραλλαγές πόζας.

Η CMU-Multi PIE είναι η μεγαλύτερη διαθέσιμη βάση δεδομένων, με 337 θέματα σε 15 διαφορετικές πόζες. Είναι μια εκτεταμένη έκδοση της βάσης δεδομένων cmu-PIE, η οποία περιέχει μόνο 68 θέματα και 13 διαφορετικές πόζες. Μια άλλη βάση δεδομένων που χρησιμοποιείται συνήθως στις εφαρμογές αναγνώρισης προσώπου από γωνία 90 μοιρών η είναι βάση δεδομένων FERET, συμπεριλαμβάνων τα 200 θέματα και 9 διαφορετικές πόζες.

Πίνακας 2.1. Βασίες δεδομένων με εικόνες από γωνία 90 μοιρών.

Name	Number of Subjects	No	Pose Yaw	Pitch	Illum	Occl	Expr	3D/Color/Gray/IR	Img / Vid
CMU-PIE	68	13	± 90	✓	43	glasses	4	Color	Img
CMU-MultiPIE	337	15	± 90	✓	18	glasses	6	Color	Img
FERET	200	9	± 90	-	✓	-	✓	Color	Img
SC-Face	130	9	± 90	-	-	-	-	Color	Img
CAS-PEAL	1040	21	± 90	✓	15	6	5	Color	Img
FacePix	30	19	± 90	-	-	-	-	Color	Img
UMIST/Sheffiel	20	19 - 48	90	-	-	-	-	Gray	Img
Stirling	35	3	90	-	-	-	3	Gray	Img
MUGSHOT	1573	2	-	-	✓	glasses	-	Gray	Img
Bern	30	5	± 90	✓	-	-	-	B/W	Img
XM2VTSDB	295	✓	± 90	✓	-	-	-	Color	Vid
M2VTS	37	✓	± 90	✓	-	glasses	✓	Color	Vid
MMI	19	1	90	-	-	-	✓	Color	Vid
UHDB1	141	5	± 90	-	-	-	-	3D/Color	Img
	141	17	✓	✓	✓	✓	2	3D/Color	Img
Bosphorus	105	13	± 90	± 20	-	4	34	3D/Color	Img

Υπάρχουν επίσης μερικές βάσεις δεδομένων που συλλέγονται σε περισσότερες ανεξέλεγκτες τοποθετήσεις. Η βάση δεδομένων UHDB1 περιέχει δέκα έξι διαφορετικές πόζες 141 θεμάτων, όπου το θέμα κάθεται σε ένα αυτοκίνητο, και μια φωτογραφική μηχανή που τοποθετείται στη σωστή γωνία συλλαμβάνει τη σκηνή.

Το καταγραμμένο στοιχείο συλλαμβάνει 7 διαφορετικές πόζες σε μια ουδέτερη έκφραση (neutral expression) και μια με ευτυχή έκφραση (happy expression). Επιπροσθετως, 5 τρισδιάστατες συλλήψεις σε διαφορετικές πόζες των ίδιων θεμάτων συμπεριλαμβάνονται επίσης στη βάση δεδομένων.

Η βάση δεδομένων MMI είναι μια βασισμένη στο WEB βάση δεδομένων εκφράσεων προσώπου συμπεριλαμβανομένων 1500 δειγμάτων 19 ανθρώπων. Περιέχει και στατικές

εικόνες και ακολουθίες εικόνας προσώπων, μετωπικές και από γωνία 90 μοιρών που επιδεικνύει τις διάφορες εκφράσεις.

Τρισδιάστατες βάσεις δεδομένων προσώπου, υπέρυθρες βάσεις δεδομένων, ή βάσεις δεδομένων που περιέχουν πολυμορφικές πληροφορίες μπορούν επίσης να χρησιμοποιηθούν για την αναγνώριση προσώπου από γωνία 90 μοιρών. Η XM2VTS βάση δεδομένων περιέχει τέσσερις καταγραφές 295 θεμάτων που λαμβάνονται για μία περίοδο τεσσάρων μηνών. Κάθε καταγραφή περιέχει έναν επικεφαλής πυροβολισμό ομιλίας (speaking head shot) και έναν περιστρεφόμενο επικεφαλής πυροβολισμό (rotating head shot). Η βάση δεδομένων περιλαμβάνει υψηλής ποιότητας εικόνες χρώματος, υγιή αρχεία, τηλεοπτικές ακολουθίες και ένα τρισδιάστατο πρότυπο. Είναι μια επέκταση της βάσης δεδομένων M2VTS, η οποία περιέχει τις ακολουθίες φωνής και κινήσεων 37 ανθρώπων, όπου οι άνθρωποι ήταν σε 3 διαφορετικές πόζες (το κεφάλι από - 90 έως +90).

Η βάση δεδομένων Bosphorus είναι μια πρόσφατη τρισδιάστατη βάση δεδομένων προσώπου που περιλαμβάνει ένα πλούσιο σύνολο εκφράσεων, διάφορος θέτει και διαφορετικοί τύποι αποφράξεων. Αν και υπάρχουν αρκετές χρήσιμες βάσεις δεδομένων προσώπου από γωνία 90 μοιρών, συλλέγονται συνήθως σε ελεγχόμενο περιβάλλον, αυτές συνήθως έχουν ελαχιστες διαφορετικές πόζες και γωνίες από τις οποίες τραβήχτηκε η φωτογραφία. Ακόμα κι αν υπάρχουν μερικές βάσεις δεδομένων, οι οποίες περιέχουν τα βίντεο των ανθρώπων σε λιγότερο ελεγχόμενες τοποθετήσεις, αυτές είτε περιέχουν λίγες παραλλαγές, είτε ένα μη ρεαλιστικό σενάριο. Επομένως, μια βάση δεδομένων που συλλέγεται σε ένα πραγματικό σενάριο, και ο περιεχει μεγάλο αριθμο από διαφορετικές πόζες παραλλαγές φωτισμου θα ήταν απαραίτητη για τις περαιτέρω εφαρμογές αναγνώρισης προσώπου.

## 2.2 Βασεις Δεδομενων

Για αυτήν την συγκεκριμένη μελέτη χρησιμοποιήθηκαν τρεις διαφορετικές βάσεις δεδομένων. Κάθε βάση δεδομένων περιείχε έναν διαφορετικό αριθμό χαρακτηριστικών όπως ο αριθμός διαφορετικών ανθρώπων, οι αλλαγές φωτισμού και διαφορετικές πόζες. Ο ακόλουθος πίνακας περιέχει τα προαναφερθέντα χαρακτηριστικά.

Πίνακας 2.2. Side-view Face Databases

Database	Number of images	Number of Subjects	Different Illumination	Different Poses	Left/Right Profile	3D/color/grey
CMU multi-PIE	3593	336	Yes	Yes	1792 L/1801 R	color
UMIST	217	20	No	Yes	0 L/217 R	color
GUARANTEE	764	2	Yes	Yes	79 L/95 R	color

### 2.2.1 CMU multi-PIE

Σε αυτήν την βάση δεδομένων τα υποκείμενα κάθονται σε μια καρέκλα. Αυτή η βάση δεδομένων περιέχει 3593 εικόνες όπου 1792 από αυτές παρουσιάζουν το αριστερό προφίλ ενός ανθρώπινου προσώπου και 1801 παρουσιάζουν το δεξί. Αυτές οι εικόνες λήφθηκαν από 336 διαφορετικούς ανθρώπους. Κάθε πρόσωπο φωτογραφήθηκε σε 20 διαφορετικούς τύπους φωτισμών και 15 διαφορετικές ποζες. Μερικά παραδείγματα φαίνονται στην **εικ. 2.1**.



**Εικόνα 2.1.** CMU multi-PIE Database

### 2.2.2 UMIST

Αυτή η βάση δεδομένων περιέχει εικόνες όπου μόνο τα πρόσωπα των υποκειμένων επιδεικνύονται σε ένα κενό υπόβαθρο. Συνολικά υπάρχουν 217 εικόνες που λήφθηκαν από 20 διαφορετικούς ανθρώπους. Ο φωτισμός για κάθε εικόνα είναι ο ίδιος. Κάθε πρόσωπο φωτογραφήθηκε περιστρεφόμενος το κεφάλι του από μια γωνία 90 μοιρών μέχρι να ερθει σε πλήρες μετωπικό πρόσωπο. Σε μερικές από τις εικόνες, τα μέρη του κεφαλιού όπως, η μύτη, τα μαλιά, κ.λπ. ίσως να λείπουν και για αυτό το λόγος η βάση δεδομένων δεν ήταν πολύ χρήσιμη. Στην εικόνα 2.2 μπορείτε να δείτε μερικά παραδείγματα.



**Εικόνα 2.2.** UMIST Database



### 2.2.3 GUARANTEE

Αυτή η βάση δεδομένων περιέχει εικόνες που φωτογραφήσαμε προσωπικά με τη χρησιμοποίηση μιας φωτογραφικής μηχανής που τοποθετήθηκε στην πλευρά μιας πόρτας σε μια γωνία 90 μοιρών που έπαιρνε μια σειρά εικόνων ενώ το θέμα διέσχισε την πόρτα. Μέχρι τώρα αυτή η βάση δεδομένων περιέχει 764 εικόνες από 2 διαφορετικούς ανθρώπους. Προκειμένου να υπάρξει ποικίλια σε ποζες, κάθε πρόσωπο διέσχισε την πόρτα με 18 διαφορετικούς τρόπους που άλλαξαν επίσης και τον τύπο φωτισμού. Στην εικ. 2.3 μπορείτε να δείτε μερικά παραδείγματα.



Εικόνα 2.3. GUARANTEE Database

## Κεφάλαιο 3

### ΑΝΙΧΝΕΥΣΗ ΠΡΟΣΩΠΩΝ

#### 3.1 Γενικά

Η *ανίχνευση προσώπου* (face detection) αποτελεί το πρώτο βήμα στην διαδικασία *αναγνώρισης προσώπου* (face recognition), το οποίο παίζει κυρίαρχο ρόλο στην απόδοση και στην χρησιμότητα ολόκληρου του συστήματος αναγνώρισης προσώπου. Σε μία φωτογραφία ή ένα βίντεο, ο κατάλληλος ανιχνευτής προσώπου πρέπει να είναι ικανός να ανακαλύπτει και να εντοπίζει όλα τα πρόσωπα που είναι παρόντα, ανεξάρτητα από τις συνθήκες φωτισμού, τη θέση, την κλίμακα, τον προσανατολισμό και τους μορφασμούς των προσώπων.

Την καλύτερη απόδοση ως προς την αποτελεσματικότητα και την ταχύτητα [11], μέχρι σήμερα εμφανίζεται να την έχει η μέθοδος των Viola & Jones [2,3] με τις βελτιώσεις του Lienhart [4,5] που στηρίζεται στην εκμάθηση με την τεχνική Adaboost και την χρήση χαρακτηριστικών γνωρισμάτων της μορφής Haar. Το ενδιαφέρον αυτής της μεθόδου προσδιορίζεται στο ότι είναι υλοποιημένη από τον Lienhart κ.ά. ως τμήμα της βιβλιοθήκης ανοικτού κώδικα OpenCV της Intel [12].

#### 3.2 Περιγραφή του προβλήματος της Ανίχνευσης Προσώπου

Η ανίχνευση προσώπου θεωρείται ως ένα πρόβλημα ταξινόμησης (classification) δύο κλάσεων, όπου μία περιοχή μιάς εικόνας ταξινομείται ως "πρόσωπο" ή "μη-πρόσωπο" [13]. Με την ανίχνευση προσώπου ανακαλύπτουμε εντοπίζουμε ανθρώπινα πρόσωπα σε μια εικόνα ανεξαρτήτως

- της θέσης τους
- του μεγέθους τους (κλίμακας)
- του προσανατολισμού τους (στροφή στο επίπεδο της εικόνας)
- της στάσης/πόζας τους (στροφή εκτός επιπέδου εικόνας)
- της έκφρασής τους (ύπαρξη μορφασμών)
- της ύπαρξης άλλων δομικών στοιχείων (μούσια, μουστάκια, γυαλιά)
- του περιεχομένου της εικόνας (ύπαρξη εμποδίων ή άλλων προσώπων)
- και του φωτισμού και των συνθηκών αποτύπωσης της εικόνας (ευαισθησία κάμερας, ανάλυση).

Όλα τα παραπάνω αποτελούν τις κύριες αιτίες της μεγάλης δυσκολίας που εμφανίζει το πρόβλημα της ανίχνευσης ενός προσώπου [13].

Οι παραλλαγές αυτές στην εμφάνιση του προσώπου, στην στάση, στην έκφραση και τον φωτισμό κάνουν πολύπλοκο το πολύπτυχο (manifold) του χώρου των προσώπων και δυσδιάκριτα τα όρια προσώπων / μη-προσώπων. Για να είναι εφικτή η διαχείριση αυτής της πολύπλοκης κατάστασης απαιτείται ένας μη γραμμικός ταξινομητής, καθώς επίσης και ένα μεγάλο σύνολο δειγμάτων εκπαίδευσης, τα οποία θα εμπεριέχουν αυτήν την ποικιλότητα στην εμφάνιση των προσώπων. Ένα επίσης σημαντικό θέμα, είναι η ταχύτητα για την απόδοση του συστήματος σε πραγματικό χρόνο. Έχει γίνει σημαντική ερευνητική προσπάθεια για την δημιουργία σύνθετων και γρήγορων ταξινομητών και από τη δεκαετία του '90 χρόνια έχει γίνει σημαντική πρόοδος στο θέμα αυτό[11].

### 3.3 Μέθοδοι

Κατά τον Yang [YANG02], οι υπάρχουσες τεχνικές για την ανίχνευση (face detection) ή τον εντοπισμό προσώπων (face localization) σε μονόχρωμες ή έγχρωμες εικόνες κατηγοριοποιούνται ως εξής:

- Μέθοδοι βασισμένες στη γνώση (knowledge-based methods):

Οι μέθοδοι Ανίχνευσης/Εντοπισμού Προσώπου, κωδικοποιούν την ανθρώπινη γνώση σε κανόνες αναφορικά με το τί συνιστά ένα τυπικό πρόσωπο. Ο ερευνητής κωδικοποιεί συνήθως τις σχέσεις (θέσεις, αποστάσεις) ανάμεσα στα χαρακτηριστικά του προσώπου. Η μετατροπή των σχέσεων των χαρακτηριστικών σε συγκεκριμένους κανόνες, αποτελούν τις δυσκολίες αυτής της μεθόδου. Αυτές οι μέθοδοι έχουν σχεδιαστεί για τον εντοπισμό του προσώπου.

- Προσεγγίσεις αναλλοίωτων χαρακτηριστικών (feature invariant approaches):

Οι μέθοδοι αυτοί, έχουν ως στόχο να βρουν τα δομικά χαρακτηριστικά ενός προσώπου ακόμα και όταν η στάση, η θέση παρατήρησης ή οι συνθήκες φωτισμού ποικίλουν. Με βάση αυτά τα δομικά χαρακτηριστικά (φρύδια, μάτια, στόμα κλπ.) χτίζονται στατιστικά μοντέλα μορφών (statistical shape models) [14], που στη συνέχεια μπορούν να χρησιμοποιηθούν για τον εντοπισμό του προσώπου. Η απόκρυψη των χαρακτηριστικών και η ύπαρξη πρόσθετων αντικειμένων που αλλοιώνουν τις μορφές, αποτελούν τις δυσκολίες αυτής της μεθόδου.

- Μέθοδοι συνταιριάματος υποδειγμάτων (template matching methods):

Για να περιγραφεί το πρόσωπο σαν σύνολο ή τα χαρακτηριστικά του προσώπου ξεχωριστά, σ'αυτή τη μέθοδο χρησιμοποιούνται διάφορα προκαθορισμένα τυπικά υποδείγματα (templates). Για την ανίχνευση υπολογίζεται η συσχέτιση μεταξύ της εικόνας εισόδου και των αποθηκευμένων υποδειγμάτων. Οι μέθοδοι αυτές χρησιμοποιούνται για ανίχνευση αλλά και για εντοπισμό των προσώπων. Οι δυσκολίες αυτής της μεθόδου οφείλονται στην αντιμετώπιση των παραλλαγών στην κλίμακα, την μορφή και τον προσανατολισμό, δεδομένου ότι τα υποδείγματα είναι προκαθορισμένα.

- Μέθοδοι βασισμένες στην εμφάνιση (appearance-based methods):

Σε αντίθεση με την μέθοδο των υποδειγμάτων όπου τα υποδείγματα είναι προκαθορισμένα, εδώ τα μοντέλα ή υποδείγματα προσώπου, μαθαίνονται από ένα σύνολο εικόνων εκπαίδευσης που είναι αντιπροσωπευτικά της ποικιλότητας της εμφάνισης των προσώπων. Αυτά τα μοντέλα από εκμάθηση χρησιμοποιούνται στη συνέχεια για την ανίχνευση των προσώπων. Κυρίως οι μέθοδοι αυτές χρησιμοποιούνται για την ανίχνευση, αλλά και για εντοπισμό των προσώπων.

Οι μέθοδοι που βασίζονται στην εμφάνιση (appearance-based), είναι οι πιο επιτυχημένες τεχνικές ανίχνευσης προσώπου και βασίζονται αποκλειστικά στην εμφάνιση των προσώπων, χωρίς να χρησιμοποιούνται άλλα ευρήματα.

Η διαδικασία ανίχνευσης έχει ως εξής :

Μια εικόνα εισόδου σαράνεται σε όλες τις δυνατές θέσεις και κλίμακες από ένα υπο-παράθυρο ανίχνευσης. Η ανίχνευση προσώπου αποφασίζεται από την ταξινόμηση του δείγματος στο υπο-παράθυρο σαν πρόσωπο ή μη-πρόσωπο. Ο ταξινομητής προσώπων / μη-πρόσωπων εκπαιδεύεται από ένα εκπαιδευτικό σύνολο δειγμάτων από πρόσωπα και μη-πρόσωπα χρησιμοποιώντας στατιστικές μεθόδους εκπαίδευσης [11].

### 3.4 Μικρό Ιστορικό των κυριότερων Μεθόδων Ανίχνευσης

Οι κυριότερες μέθοδοι υποδειγμάτων και εμφάνισης για την ανίχνευση προσώπων, που επηρέασαν ουσιαστικά την εξέλιξη της έρευνας στο συγκεκριμένο τομέα αναφέρονται παρακάτω:

• 1992: Craw et al.

Χρήση Υποδειγμάτων Μορφών (Shape Templates) [15]

• 1995: Lanitis et al.

Χρήση Μοντέλων Ενεργών Μορφών (Active Shape Models) [16]

• 1997: Osuna et al.

Χρήση μη γραμμικών Μηχανών Ανυσματικής Στήριξης (SVM)

• 1998: Papageorgiou and Poggio

Χρήση κυματιδίων Haar (Haar wavelets) με SVM ταξινομητή [17]

• 1998: Sung and Poggio

Χρήση Γκαουσιανών κατανομών και πολυστρωματικών αντιλήπτρων (multilayer perceptron) [18]

• 1998: Rowley et al.

Χρήση συνόλου νευρωνικών δικτύων (neural networks) [19]

• 2000: Schneiderman and Kanade

Χρήση στατιστικής για την τοπική εμφάνιση και εκμάθηση με AdaBoost [20]

• 2001: Viola and Jones.

Χρήση των κυματιδίων τύπου Haar σε ταξινομητές ενός-μόνο χαρακτηριστικού, που εκπαιδεύονται με τον AdaBoost αλγόριθμο και οργανώνονται σε μια διαδοχή αποφάσεων [3]

• 2002: Lienhart et al.

Χρήση ενός εκτεταμένου συνόλου περιστραμμένων χαρακτηριστικών Haar για την αντιμετώπιση των στροφών στο επίπεδο εικόνας [5]

• 2002: Li et al.

Χρήση ενός επεκτεταμένου συνόλου περι στραμμένων χαρακτηριστικών Haar και ενός πυραμιδοειδούς ανιχνευτή για την αντιμετώπιση των στροφών εκτός του επιπέδου της εικόνας [10]



• Viola & Jones



• Li et al.



• Schneiderman et al.



• Rowley et al.



**Εικόνα 3.1: Παραδείγματα ανίχνευσης προσώπων.**

Την καλύτερη απόδοση μέχρι σήμερα και ως προς την ακρίβεια και ως προς την ταχύτητα [13,11], την έχει επιτύχει η προσέγγιση της AdaBoost εκμάθησης χαρακτηριστικών τύπου Haar των Viola & Jones.

### 3.5 Μέθοδοι Βασισμένες στην Εμφάνιση

Οι μέθοδοι που είναι βασισμένες στην εμφάνιση (appearance-based methods), εμφανίζουν ορισμένα κοινά χαρακτηριστικά κατά την διαδικασία ανίχνευσης των προσώπων:

- Σύνολο εκπαίδευσης θετικών και συνήθως αρνητικών δειγμάτων προσώπων
- Εκπαίδευση και χρήση ενός ταξινομητή
- Συγκεκριμένο τρόπο αναπαράστασης των προσώπων
- Προεπεξεργασία
- Συγκεκριμένη στρατηγική ανίχνευσης/αναζήτησης στο χώρο και την κλίμακα
- Μετεπεξεργασία

Στις παρακάτω παραγράφους, θα εξετάσουμε αναλυτικότερα τα παραπάνω κοινά χαρακτηριστικά.

#### 3.5.1 Δείγματα προσώπων και μη-προσώπων

Για την εκπαίδευση του ταξινομητή χρησιμοποιούνται τόσο θετικά όσο και αρνητικά δείγματα προσώπων:

- *Θετικά δείγματα*

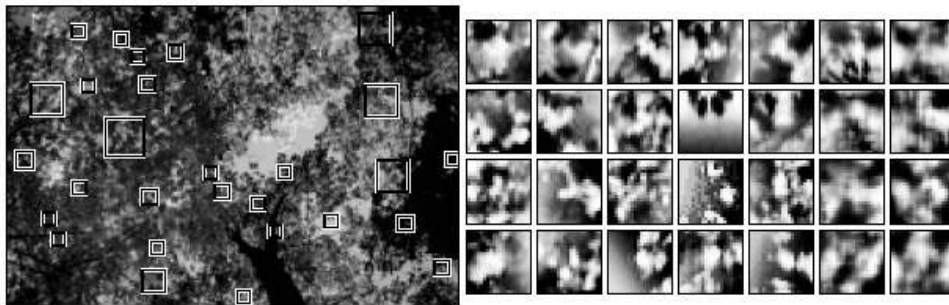
Για να περιλαμβάνονται περισσότερες εκδοχές του προσώπου που ανιχνεύεται, η λήψη θα πρέπει να είναι όσο το δυνατό μεγαλύτερης ποικιλίας δειγμάτων. Κόψιμο και κανονικοποίηση κάθε εικόνας προσώπου σε ένα συγκεκριμένο μέγεθος, π.χ. 20×20 pixels, και αύξηση του πλήθους των θετικών δειγμάτων με τη δημιουργία εικονικών δειγμάτων (Εικ. 3.2) [18].



**Εικόνα 3.2:** Πρόσωπα που χρησιμοποιούνται στο σύνολο των θετικών δειγμάτων

- *Αρνητικά δείγματα*

Τα αρνητικά δείγματα λαμβάνονται από οποιαδήποτε εικόνα που δεν περιέχει πρόσωπα (Εικ. 3.3). Οι εικόνες δείγματα των μη-προσώπων λαμβάνονται στο ίδιο μέγεθος με τις εικόνες των προσώπων. Εφαρμόζεται η τεχνική Bootstrapping [18].

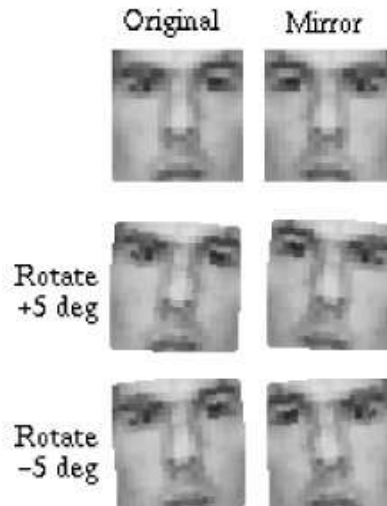


**Εικόνα 3.3:** Μη-πρόσωπα που χρησιμοποιούνται στο σύνολο των αρνητικών δειγμάτων

Κατά τη διάρκεια της εκπαίδευσης, το μερικώς εκπαιδευμένο σύστημα εφαρμόζεται σε εικόνες οι οποίες δεν περιέχουν πρόσωπα (όπως στην Εικ. 3.3 αριστερά). Κάθε περιοχή στην εικόνα η οποία ανιχνεύεται εσφαλμένα ως πρόσωπο (τα οποία παρατίθενται στην Εικ. 3.3 δεξιά) είναι δείγματα μη-προσώπων, τα οποία προστίθενται στο σύνολο των αρνητικών δειγμάτων εκπαίδευσης [19].

- *Εικονικά Θετικά Δείγματα*

Για τον πολλαπλασιασμό του συνόλου των δειγμάτων εκπαίδευσης, δημιουργούνται με τεχνητό τρόπο νέες εικόνες-δείγματα από τις υπάρχουσες.



**Εικόνα 3.4:** Τεχνητώς δημιουργούμενα εικονικά δείγματα προσώπων

Έτσι την αύξηση του αριθμού των δειγμάτων εκπαίδευσης και μικρότερη ευαισθησία στα σφάλματα λήψης των δειγμάτων, μπορούμε να την πετύχουμε με κατοπτρισμό, μικρές περιστροφές, μετατοπίσεις και κλιμακώσεις των δειγμάτων των προσώπων (Εικ. 3.4), [18,19].

### 3.5.2 Χρησιμοποιούμενοι Ταξινομητές

Οι κυριότεροι από τους ταξινομητές/αλγόριθμοι, που έχουν χρησιμοποιηθεί κατά καιρούς σε εργασίες ανίχνευσης προσώπων είναι:

- Αλγόριθμοι Προσαρμοστικής Ενίσχυσης (Adaptive Boosting – AdaBoost)
- Νευρωνικά Δίκτυα - Πολυστρωματικό Αντίληπτρο (Neural Networks - Multilayer Perceptron)
- Ανάλυση Θεμελιωδών Συνιστωσών (Principal Component Analysis - PCA),
- Μηχανές Ανυσματικής Στήριξης (Support Vector Machines - SVM)
- Μέθοδοι βασισμένες σε Κατανομές (Distribution-based methods)
- Απλοϊκός Ταξινομητής Bayes (Naïve Bayes classifier)
- Μοντέλα Κρυφών Μεταβλητών Markov (Hidden Markov Models)
- Επαγωγική Μάθηση (Inductive Learning)

### 3.5.3 Αναπαράσταση Προσώπων

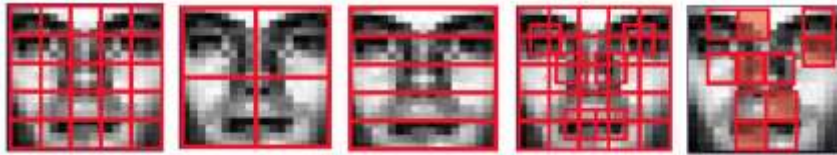
Σε διαφορετικές εργασίες, έχουν χρησιμοποιηθεί συγκεκριμένοι τρόποι αναπαράστασης των προσώπων ανάλογα με τα χαρακτηριστικά (features) που έχουν επιλεγεί:

- Ολιστική (Holistic): Χρησιμοποιείται ολόκληρη η εικόνα του κάθε προσώπου και αναπαρίσταται από ένα διάνυσμα τιμών έντασης.
- Βασισμένη σε μπλοκ (Block-based): Αποσύνθεση κάθε εικόνας προσώπου σε ένα σύνολο επικαλυμμένων ή μη επικαλυμμένων περιοχών (μπλοκ) όπως στην Εικ. 3.5.

ο Απεικόνιση σε πολλαπλές κλίμακες

ο Επιπλέον επεξεργασία με Διανυσματική Κβάντιση (Vector Quantization), Ανάλυση Θεμελειωδών Συνιστωσών (Principal Component Analysis), Κυματίδια (Wavelets) κλπ.

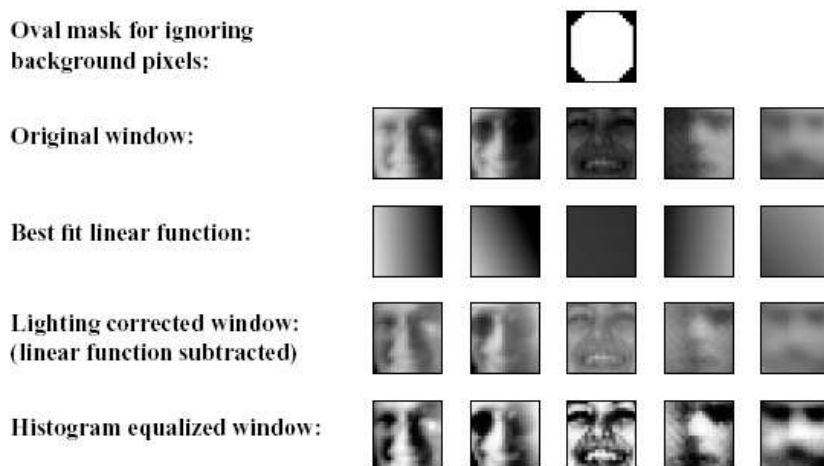
ο Χρήση χαρακτηριστικών γνωρισμάτων (features).



Εικόνα 3.5: Διάφορες αναπαραστάσεις στηριγμένες στον τεμαχισμό σε μπλοκ.

### 3.5.4 Προεπεξεργασία

Η διόρθωση των αποκλίσεων των παραμέτρων αποτύπωσης των εικόνων μέσω των καμερών, όπως επίσης και των παραλλαγών στις συνθήκες φωτισμού, βοηθείται από την προεπεξεργασία με κανονικοποίηση της έντασης των εικονοστοιχείων (pixels).



Εικόνα 3.6: Διαδικασίες προ-επεξεργασίας εικόνας προσώπου

Η διαδικασία προ-επεξεργασίας στο υπο-παράθυρο της εικόνας και κατά την διαμόρφωση των δειγμάτων των προσώπων ακολουθεί τα εξής βήματα [18,19]:

- Τροποποίηση μεγέθους εικόνας: Τροποποίηση όλων των προτύπων προσώπων σε ένα συγκεκριμένο μέγεθος (π.χ. σε 20×20 pixels).
- Μασκάρισμα: περιορισμός των μη επιθυμητών δομών του υποβάθρου (θόρυβος) κοντά στα όρια του πλαισίου ενός προτύπου προσώπου
- Επιδιόρθωση διαβαθμισμένου φωτισμού: Οι ακραίες διαβαθμίσεις φωτισμού (σκιές που δημιουργούνται από ακραίες γωνίες φωτισμού), επιδιορθώνονται με την ανεύρεση μιας γραμμικής συνάρτησης η οποία να ταιριάζει στην διαβάθμιση των τιμών έντασης του παραθύρου του προσώπου και της οποίας οι τιμές αφαιρεθούν.



- Εξισορρόπηση Ιστογράμματος: αντισταθμίζει τις επιδράσεις φωτισμού εξαιτίας διαφορετικών συνθηκών φωτισμού και διαφορετικών καμπύλων απόκρισης της κάθε κάμερας λήψης.

### 3.5.5 Αναζήτηση στον Χώρο και την Κλίμακα

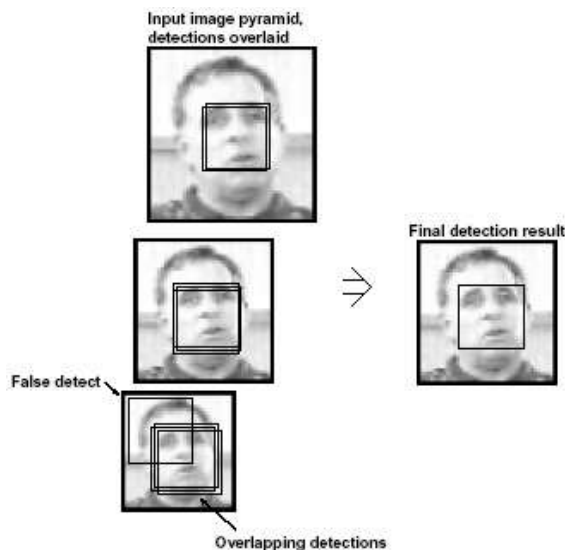
Το παράθυρο ανίχνευσης σαρώνει την υπό ανίχνευση εικόνα σε διαφορετικές κλίμακες και θέσεις:

- Σάρωση μιας εικόνας εισόδου με βήμα  $\Delta$ -pixels (π.χ. 2-pixels) οριζόντια και Κάθετα
- Υποδειματοληψία της εικόνας εισόδου κατά έναν παράγοντα και συνέχιση της αναζήτησης
- Συνέχιση της υποδειματοληψίας της εικόνας εισόδου και αναζήτηση μέχρι η εικόνα που σαρώνεται να γίνει πολύ μικρή, ώστε να είναι αδύνατο να περιέχει πρόσωπο.

Η ταχύτητα, όπως επίσης και η ακρίβεια του ανιχνευτή, επηρεάζεται από την επιλογή του βήματος σάρωσης και του παράγοντα υποδειματοληψίας. Μπορούμε να έχουμε αντίστοιχη μεγέθυνση του παραθύρου σάρωσης, αντί για σμίκρυνση της εικόνας εισόδου [2,3].

### 3.5.6 Μετεπεξεργασία

Πολλαπλές ανιχνεύσεις μπορεί να συμβούν σε κοντινές θέσεις ή σε πολλαπλές κλίμακες κατά την σάρωση μιας εικόνας, από τη στιγμή που ο ανιχνευτής δεν επηρεάζεται από μικρές μεταβολές στην μετατόπιση και την κλίμακα. Επίσης μπορεί να συμβούν και εσφαλμένες ανιχνεύσεις, αλλά συνήθως με λιγότερη συχνότητα από ότι πολλαπλές ανιχνεύσεις προσώπων. Είναι ωστόσο χρήσιμο να γίνει μετεπεξεργασία των θετικών παραθύρων ανίχνευσης, προκειμένου να συνδυαστούν οι επικαλυπτόμενες ανιχνεύσεις σε μια μόνο ανίχνευση και να περιοριστούν οι εσφαλμένες ανιχνεύσεις.



Εικόνα 3.7: Συγχώνευση πολλαπλών ανιχνεύσεων και απαλοιφή των λανθασμένων

Ο αριθμός των πολλαπλών επικαλυπτόμενων ανιχνεύσεων στη γειτονιά μιας θέσης μπορεί να χρησιμοποιηθεί σαν μια αποτελεσματική ένδειξη για την ύπαρξη ενός προσώπου σε αυτό το σημείο, ενώ σε αντίθετη περίπτωση να απορριφθεί ως εσφαλμένη ανίχνευση.

Οι παρακάτω τεχνικές, χρησιμοποιούνται για τον χειρισμό των πολλαπλών ανιχνεύσεων [2,3,11,19]:

- Ανέυρεση επικαλυπτόμενων ανιχνεύσεων (η τομή τους να ξεπερνά ένα κατώφλι, π.χ. 80%)
- Συγγώνευση πολλαπλών ανιχνεύσεων (ο αριθμός των πολλαπλών ανιχνεύσεων να ξεπερνά ένα κατώφλι, π.χ. 4)
- Καθορισμός του τελικού παραθύρου (επιλογή του παραθύρου ανίχνευσης με τη μέγιστη εμπιστοσύνη ταξινόμησης  $H(x)=\max$ , ή εύρεση του μέσου παράθυρου)

### 3.6 Ανίχνευση Προσώπου κατά Viola & Jones

Οι ιδέες-κλειδιά της μεθόδου αυτής είναι οι εξής [2,3]:

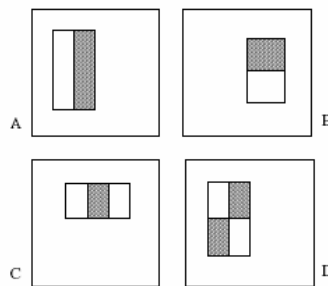
- Χρήση απλών χαρακτηριστικών τύπου Haar για απόκτηση γνώσης από δεδομένα μάθησης, που αποτελούνται από εικόνες προσώπων και μη-προσώπων.
- Χρήση μιας νέας εικόνας αναπαράστασης των προσώπων που ονομάζεται *Εικόνα Ολοκλήρωμα* (Integral Image), και που επιτρέπει τα χαρακτηριστικά που χρησιμοποιούνται από τον ανιχνευτή να υπολογίζονται πολύ γρήγορα.
- Χρήση ενός αλγόριθμου μάθησης, που στηρίζεται στον *AdaBoost*, ο οποίος επιλέγει ένα μικρό αριθμό από κρίσιμα οπτικά χαρακτηριστικά και αποδίδει άκρως αποτελεσματικούς ταξινομητές.
- Συνδυασμός των ταξινομητών σε διάταξη καταρράκτη (cascade), που επιτρέπει περιοχές υποβάθρου της εικόνας να απορρίπτονται γρήγορα, αναλώνοντας περισσότερο υπολογιστικό χρόνο σε περιοχές που μοιάζουν περισσότερο σε πρόσωπα.

#### 3.6.1 Υπολογισμός Χαρακτηριστικών

- Η ταξινόμηση των εικόνων στηρίζεται στην τιμή απλών βαθμωτών χαρακτηριστικών (features), κατά τη διαδικασία ανίχνευσης
- Η χρήση των χαρακτηριστικών αντί της εικόνας έντασης (intensity image) έχει ως στόχο την μείωση των διαφοροποιήσεων μέσα στην κλάση (intra-class variability) και την αύξηση των διαφοροποιήσεων μεταξύ των κλάσεων (inter-class variability), ώστε η ταξινόμηση να καταστεί ευκολότερη.
- Τα χαρακτηριστικά αυτά περιέχουν γνώση από συγκεκριμένες περιοχές της εικόνας, που είναι δύσκολο να κωδικοποιηθεί χρησιμοποιώντας πεπερασμένα δεδομένα μάθησης.

Τα χαρακτηριστικά είναι όμοια με τις συναρτήσεις βάσης τύπου Haar (Haar basis) που έχουν χρησιμοποιηθεί και από τον Παπαγεωργίου κ.ά. για τον ίδιο σκοπό [17].

- Εφαρμόζονται σε ασπρόμαυρες εικόνες και η τιμή τους εξαρτάται από την τιμή της υπολογιζόμενης σταθμισμένης ανάλογα με το εμβαδόν διαφοράς των άθροισμάτων των εντάσεων των εικονοστοιχείων πάνω σε ορθογώνιες περιοχές, θεωρώντας τις γκρίζες περιοχές θετικές και τις λευκές αρνητικές, όπως φαίνεται στην Εικ. 3.8.
- Τα χαρακτηριστικά καθορίζονται από την θέση, τις διαστάσεις και την τιμή τους.
- Το πλήθος των χαρακτηριστικών που δημιουργούνται για δείγματα προσώπων 24×24 pixels είναι ~ 45.000, που είναι ένα υπερπλήρες σύνολο σε σχέση με τις 576 τιμές έντασης του δείγματος. Για αυτόν τον λόγο απαιτείται μια διαδικασία επιλογής των κυριότερων χαρακτηριστικών από αυτά. Σύμφωνα με τους Viola & Jones ακόμα και ανιχνευτές με 2 χαρακτηριστικά είναι αρκετά αποτελεσματικοί.

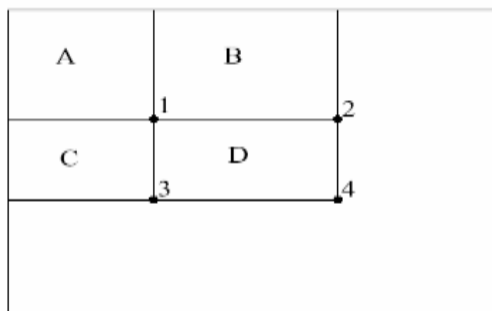


**Εικόνα 3.8: Μορφή των χαρακτηριστικών Viola & Jones**

Τα ορθογώνια χαρακτηριστικά των Viola & Jones μπορούν με τη χρήση μιας βοηθητικής εικόνας, να υπολογιστούν πολύ αποτελεσματικά ως "εικόνα ολοκλήρωμα" (integral image). Η εικόνα ολοκλήρωμα,  $I$ , έχει τιμή στη θέση  $(x, y)$  που καθορίζεται ως άθροισμα των εντάσεων των pixels του ορθογωνίου που ορίζεται από την πάνω αριστερή κορυφή  $(0, 0)$  και την κάτω δεξιά κορυφή  $(x, y)$ :

$$I(x, y) = \sum i(x', y'), \quad x' \leq x, \quad y' \leq y .$$

όπου  $i$  είναι η αρχική εικόνα εισόδου.



**Εικόνα 3.9: Αναπαράσταση της "Εικόνας Ολοκλήρωμα"**

Κάθε ορθογώνιο άθροισμα μπορεί να υπολογιστεί σε σταθερό χρόνο με τέσσερις αναφορές στις τιμές ενός πίνακα, χρησιμοποιώντας την εικόνα ολοκλήρωμα Έτσι το άθροισμα εντός του D (Εικ. 2.9) μπορεί να υπολογιστεί σαν  $I_4 + I_1 - (I_2 + I_3)$  [2].

### 3.6.2 Επιλογή Χαρακτηριστικών με την AdaBoost

Στο γενικότερο πρόβλημα της ενίσχυσης (boosting), συνδυάζεται ένα μεγάλο σύνολο λειτουργιών ταξινόμησης, αποδίδοντας μεγαλύτερο βάρος σε κάθε καλή λειτουργία ταξινόμησης και μικρότερο βάρος σε κάθε χειρότερη λειτουργία. *Αδύναμοι ταξινομητές* (weak classifiers), ονομάζονται οι ταξινομητές που συνδυάζονται, ώστε να αποτελέσουν τον τελικό *ισχυρό ταξινομητή* (strong classifier). Ο αλγόριθμος *προσαρμοστικής ενίσχυσης* (adaptive boosting) AdaBoost, είναι ο πιο δημοφιλής αλγόριθμος ενίσχυσης είναι ο οποίος ονομάζεται και *διακριτός* (discrete) AdaBoost, μιάς και αποδίδει διακριτές τιμές εξόδου [33].

Το πλήθος των ορθογώνιων χαρακτηριστικών που δημιουργούνται, όπως ήδη έχει αναφερθεί, είναι υπερβολικά μεγάλο, ώστε να απαιτείται η επιλογή των πλέον αποτελεσματικών χαρακτηριστικών που θα χρησιμοποιηθούν στον τελικό ταξινομητή. Οι Viola & Jones, θεωρώντας μια αντιστοίχιση μεταξύ αδύναμων ταξινομητών και χαρακτηριστικών, χρησιμοποίησαν τον αλγόριθμο AdaBoost [21] σαν μια αποτελεσματική διαδικασία για την ανεύρεση ενός μικρού αριθμού "καλών" χαρακτηριστικών που επιπλέον είναι σημαντικά διαφοροποιημένα. Ο περιορισμός του αδύναμου ταξινομητή σε λειτουργίες ταξινόμησης που εξαρτώνται από ένα μόνο χαρακτηριστικό, είναι μια απλή και πρακτική μέθοδος για την ολοκλήρωση αυτής της αντιστοίχισης [2,3].

Έτσι, η μέθοδος AdaBoost στοχεύει στην επίλυση των παρακάτω 3 θεμελιωδών προβλημάτων [11]:

- (1) εκμάθηση των πιο αποτελεσματικών χαρακτηριστικών από ένα μεγάλο σύνολο χαρακτηριστικών,
- (2) κατασκευή αδύναμων ταξινομητών, καθένας από τους οποίους στηρίζεται σε ένα μόνο από τα δημιουργηθέντα χαρακτηριστικά, και
- (3) συνδυασμός των αδύναμων ταξινομητών για την κατασκευή ενός ισχυρού ταξινομητή.

**Algorithm: AdaBoost Algorithm for Learning a Strong Classifier** [VIOL01b]Input

Given example images  $(x_1, y_1), \dots, (x_n, y_n)$  where  $y_i = 0, 1$  for negative and positive examples respectively.

Initialization

Initialize weights  $w_{l,i} = \frac{1}{2m}, \frac{1}{2l}$  for  $y_i = 0, 1$  respectively, where  $m$  and  $l$  are the number of negatives and positives respectively.

Inclusion

For  $t = 1, \dots, T$ :

1. Normalize the weights,

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}},$$

so that  $w_t$  is a probability distribution.

2. For each feature  $j$ , train a classifier  $h_j$ , which is restricted to using a single feature.

$$\text{The error is evaluated with respect to } w_t, \varepsilon_j = \sum_i w_i |h_j(x_i) - y_i|.$$

3. Choose the classifier  $h_t$  with the lowest error  $\varepsilon_t$ .

4. Update the weights:

$$w_{t+1,i} = w_{t,i} \beta_t^{1-\varepsilon_i},$$

where  $\varepsilon_i = 0$  if example  $x_i$  is classified correctly, otherwise  $\varepsilon_i = 1$ , i.e.:

$$w_{t+1,i} = w_{t,i} \beta_t \quad // \text{ if } x_i \text{ classified correctly}$$

$$w_{t+1,i} = w_{t,i} \quad // \text{ if } x_i \text{ classified incorrectly}$$

$$\text{where } \beta_t = \frac{\varepsilon_t}{1 - \varepsilon_t}.$$

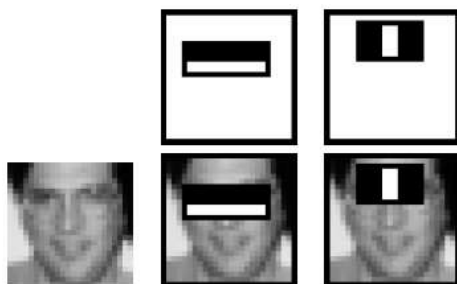
Output

The final strong classifier is:

$$H(x) = \begin{cases} 1, & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, & \text{otherwise} \end{cases}, \text{ where } \alpha_t = \ln \frac{1}{\beta_t}$$

Σύμφωνα με τους Viola & Jones [2,3] κατά την εφαρμογή του AdaBoost:

- Σε κάθε πέρασμα  $t$  εκπαιδεύεται ένας νέος αδύναμος ταξινομητής  $h_t$  που προστίθεται στο σύνολο με μεγαλύτερο συντελεστή  $\alpha_t$  όσο μικρότερο σφάλμα ταξινόμησης  $\varepsilon$  δίνει.
- Το βάρος κάθε δείγματος ενημερώνεται έτσι ώστε σε κάθε επόμενο πέρασμα τα δείγματα που ταξινομούνται σωστά να έχουν μικρότερη βαρύτητα κατά  $\beta_t$ .
- Ο τελικός ισχυρός ταξινομητής αποτελείται από  $T$  αδύναμους ταξινομητές που αντιστοιχούν στα ισχυρότερα χαρακτηριστικά, με βάρη αντιστρόφως ανάλογα προς το σφάλμα ταξινόμησης.
- Όπως αποδεικνύεται από τους Freund & Schapire [21] ο ισχυρός ταξινομητής μπορεί να πετύχει αυθαίρετα υψηλό ρυθμό ορθών ταξινομήσεων με αυθαίρετα χαμηλό ρυθμό εσφαλμένων ταξινομήσεων, αρκεί το πλήθος των αδύναμων ταξινομητών να 'ναι αρκετά μεγάλο.



Εικόνα 3.10: Τα 2 κυριότερα χαρακτηριστικά εφαρμοσμένα σε ένα τυπικό πρόσωπο [3]

### 3.6.3 Κατασκευή του Αδύναμου Ταξινομητή

Η "ρίζα" ("stump") ενός δένδρου απόφασης (decision tree), είναι ο απλούστερος τύπος ενός αδύναμου ταξινομητή. Μία ρίζα απόφασης, μπορεί να κατασκευαστεί όταν το χαρακτηριστικό παίρνει πραγματικές τιμές, και συγκρίνοντας απλά την τιμή του επιλεγμένου χαρακτηριστικού με μια συγκεκριμένη τιμή κατωφλίου. Έτσι ο αδύναμος αλγόριθμος μάθησης σχεδιάζεται ώστε να μπορεί να επιλέγει εκείνο το μοναδικό χαρακτηριστικό που διαχωρίζει καλύτερα τα θετικά από τα αρνητικά δείγματα. Για κάθε χαρακτηριστικό, ο αδύναμος ταξινομητής καθορίζει το ιδανικό κατώφλι λειτουργίας της ταξινόμησης, έτσι ώστε να ελαχιστοποιείται ο αριθμός δειγμάτων που ταξινομείται εσφαλμένα [2,3,11].



Εικόνα 3.11: Υπολογισμός της τιμής χαρακτηριστικών πάνω σε πρόσωπα και μη-πρόσωπα

Έτσι ο αδύναμος ταξινομητής  $h_j(x)$  αποτελείται από ένα χαρακτηριστικό  $j$  και ένα κατώφλι  $\theta_j$ :

- Για κάθε χαρακτηριστικό  $j$ , υπολογίζεται η  $f_j(x)$ , μια βαθμωτή τιμή του χαρακτηριστικού (εδώ οι διαφορές αθροισμάτων), όπου  $x$  είναι ένα θετικό ή αρνητικό δείγμα
- Κάθε χαρακτηριστικό χρησιμοποιείται σαν ένας αδύναμος ταξινομητής
- Καθορισμός τιμής κατωφλίου  $\theta_j$  για κάθε χαρακτηριστικό έτσι ώστε τα περισσότερα δείγματα να ταξινομούνται σωστά:

$$h_j(x) = \begin{cases} 1, & \text{if } f_j(x) < \theta_j \text{ (or } f_j(x) > \theta_j), \quad x: \text{positive} \\ 0, & \text{otherwise, } \quad x: \text{negative} \end{cases}$$

- Επιλογή χαρακτηριστικού και κατωφλίου με το χαμηλότερο σταθμισμένο σφάλμα ταξινόμησης
- Διαδοχική εκτίμηση όλων των χαρακτηριστικών

### 3.6.4 Ταξινόμηση με έναν Καταρράκτη Ταξινομητών

Οι Viola & Jones εισήγαγαν την έννοια του καταρράκτη ταξινομητών (cascade of classifiers), για να αποφύγουν την σύγκλιση του AdaBoost που απαιτεί μεγάλο αριθμό

δειγμάτων και καταλήγει σε μεγάλο πλήθος αδύναμων ταξινομητών που απαιτούν μεγάλο υπολογιστικό κόστος, [2,3].

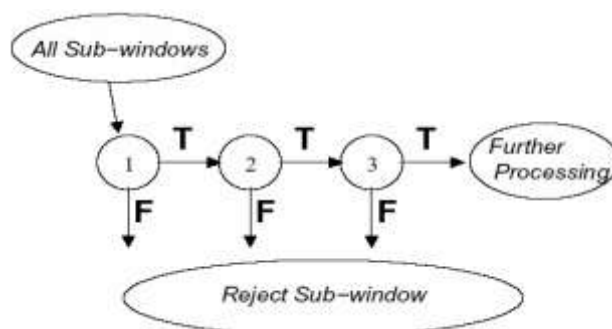
Με τη μέθοδο αυτή κατασκευάζεται μία διαδοχή ταξινομητών στη μορφή καταρράκτη (cascade) που επιτυγχάνει αυξημένη απόδοση στην ανίχνευση και μειώνει ριζικά τον χρόνο υπολογισμού. Η ιδέα είναι ότι μπορούν να κατασκευαστούν μικροί και ωστόσο αποτελεσματικοί, συνδυασμένοι ταξινομητές που απορρίπτουν πολλά από τα αρνητικά, ενώ ανιχνεύουν σχεδόν όλα τα θετικά περιστατικά.

Η μέθοδος του καταρράκτη ταξινομητών στηρίζεται στο γεγονός ότι σε μία οποιαδήποτε εικόνα η πλειονότητα των παραθύρων ανίχνευσης δεν περιλαμβάνει πρόσωπα. Έτσι πιο απλοποιημένοι και λιγότερο χρονοβόροι ταξινομητές χρησιμοποιούνται για να απορρίψουν την πλειονότητα των παραθύρων ανίχνευσης ως αρνητικά, προτού χρησιμοποιηθούν οι πιο σύνθετοι και περισσότερο χρονοβόροι ταξινομητές που θα επεξεργαστούν τις πιο πολύπλοκες περιπτώσεις και θα επιτύχουν χαμηλά επίπεδα εσφαλμένων θετικών ανιχνύσεων.

#### Παράδειγμα: Καταρράκτης Ταξινομητής 32 επιπέδων [3]

- 2-χαρακτηριστικών ταξινομητής στο πρώτο επίπεδο απορρίπτει το 60% των μη-προσώπων ενώ ανιχνεύει 100% τα πρόσωπα
- 5-χαρακτηριστικών ταξινομητής στο δεύτερο επίπεδο απορρίπτει το 80% των μη-προσώπων ενώ ανιχνεύει 100% τα πρόσωπα
- 20-χαρακτηριστικών ταξινομητής στα επίπεδα 3,4 και 5
- 50-χαρακτηριστικών ταξινομητής στα επίπεδα 6 και 7
- 100-χαρακτηριστικών ταξινομητής στα επίπεδα 8 έως και 12
- 200-χαρακτηριστικών ταξινομητής στα επίπεδα 13 έως και 32

Η συνολική διαδικασία ανίχνευσης είναι παρόμοια με ένα δένδρο απόφασης (decision tree). Ένα θετικό αποτέλεσμα από τον ταξινομητή πρώτου επιπέδου οδηγείται στον ταξινομητή δεύτερου επιπέδου, του οποίου το θετικό αποτέλεσμα οδηγείται στον ταξινομητή τρίτου επιπέδου κ.ο.κ. όπως στην Εικ. 3.12. Τα αρνητικά αποτελέσματα σε κάθε επίπεδο απορρίπτονται χωρίς να επανελέγχονται. Έτσι οι ταξινομητές των αρχικών επιπέδων ασχολούνται με τα εύκολα περιστατικά, ενώ οι επόμενοι αντιμετωπίζουν πιο δύσκολες περιπτώσεις



Εικόνα 3.12: Σχηματική παράσταση μιας ανίχνευσης με καταρράκτη ταξινομητών

Η εκπαίδευση του καταρράκτη ταξινομητή γίνεται χρησιμοποιώντας τον AdaBoost, και καθορίζει:

- τον αριθμό των επιπέδων του καταρράκτη ταξινομητή
- τον αριθμό των χαρακτηριστικών σε κάθε επίπεδο
- το κατώφλι σε κάθε επίπεδο

ώστε να ελαχιστοποιείται ο αριθμός των χρησιμοποιούμενων χαρακτηριστικών, δεδομένων των  $f$ ,  $d$  και  $F_{target}$ .

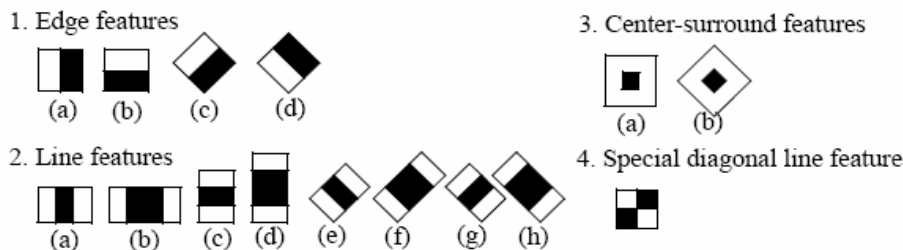
<p><b>Algorithm: Training Algorithm for Building a Cascade Detector</b> [3]</p> <p>Given:  P : a training set of Positive examples, N : a training set of Negative examples.  f : maximum acceptable false positive rate per layer .  d: minimum acceptable detection rate per layer.  <math>F_{target}</math>: target overall false positive rate.</p> <p>Initialize:  <math>F_0 = 1.0</math>; <math>D_0 = 1.0</math>  <math>i=0</math>  Iterate:  While <math>F_i &gt; F_{target}</math> do:  • <math>i \leftarrow i+1</math>  • <math>n_i=0</math>; <math>F_i=F_{i-1}</math>  • While <math>F_i &gt; f \times F_{i-1}</math> do:  ○ <math>n_i \leftarrow n_i+1</math>  ○ Use P and N to train a classifier with <math>n_i</math> features using AdaBoost  ○ Evaluate current cascaded classifier on validation test to determine <math>F_i</math> and <math>D_i</math>  ○ Decrease threshold for the <math>i</math>th classifier until the current cascaded classifier has a detection rate of at least <math>d \times D_{i-1}</math> (this also affects <math>F_i</math>)</p> <p>• <math>N \leftarrow 0</math>  • If <math>F_i &gt; F_{target}</math> then evaluate the current cascaded detector on the set of non-face images and put any false detection into the set N</p>
--

### 3.7 Ανίχνευση Προσώπου κατά Lienhart

Ο Lienhart κ.ά [4,5] εισήγαγαν δύο κύριες βελτιώσεις στην μέθοδο ανίχνευσης των Viola & Jones [2,3], πρώτον ένα εκτεταμένο σύνολο από ορθογώνια χαρακτηριστικά τύπου Haar, και δεύτερον την χρήση της gentle AdaBoost για την κατασκευή του ισχυρού ταξινομητή σε συνδυασμό με τη χρήση CART δένδρων στη θέση των αδύναμων ταξινομητών.

- Το εκτεταμένο σύνολο από Haar χαρακτηριστικά:

Το βασικό σύνολο των Haar χαρακτηριστικών εμπλούτισε ο Lienhart, με ένα πιο αποτελεσματικό σύνολο από περιστραμμένα κατά 45ο χαρακτηριστικά, προσθέτοντας επιπλέον γνώση στο σύστημα εκπαίδευσης και βελτιώνοντας την απόδοση του συστήματος [4,5].



Εικόνα 3.13: Εκτεταμένο σύνολο στραμμένων Haar χαρακτηριστικών



Τα χαρακτηριστικά αυτά φαίνονται στην Εικ. 3.13 και αποτελούνται από τέσσερα χαρακτηριστικά ακμών (Εικ. 3.13.1), οκτώ χαρακτηριστικά γραμμών (Εικ. 3.13.2) και δύο χαρακτηριστικά κέντρων (Εικ. 3.13.3), ενώ το χαρακτηριστικό διαγώνιας γραμμής (Εικ. 3.13.4) δεν χρησιμοποιείται σε αντίθεση με τους Viola & Jones.

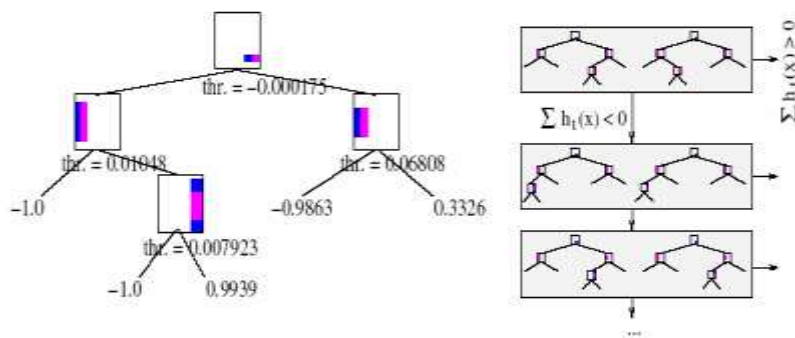
Τα χαρακτηριστικά καθορίζονται από τα ορθογώνια  $r = (x,y,w,h,a)$ ,  $a = \{0\circ, 45\circ\}$ , και παίρνουν τιμές την σταθμισμένη διαφορά των αθροισμάτων των εντάσεων στις λευκές και μαύρες περιοχές:

$$\text{feat}_i = \sum w_i \cdot \text{RecSum}(r_i)$$

Το πλήθος των ορθογώνιων χαρακτηριστικών του Lienhart για δείγματα  $24 \times 24$  pixels ανέρχεται σε  $\sim 118.000$ .

• CART δένδρα ως αδύναμοι ταξινομητές:

Ο Lienhart χρησιμοποίησε ως αδύναμους ταξινομητές, μικρά CART δένδρα με έως 4 χαρακτηριστικά αντί για απλές ρίζες απόφασης (stumps). Τα *Δένδρα Ταξινόμησης και Παλινδρόμησης* (Classification and Regression Trees – CART) είναι μικροί ταξινομητές δένδρων απόφασης, καλοί στην εκμάθηση εξαρτήσεων μεταξύ των χαρακτηριστικών. Η συνάρτηση  $h_i(x)$  καθορίζεται σύμφωνα με την διαδρομή μέσα στο δένδρο απόφασης.



Εικ. 3.14: Ένα δένδρο ταξινόμησης CART και ένας καταρράκτης

Η χρήση μικρών δένδρων απόφασης εμφανίζει καλύτερη απόδοση από τις απλές ρίζες απόφασης.

• Χρήση παραλλαγών της AdaBoost:

Ο Lienhart μελέτησε τρεις παραλλαγές της AdaBoost: η Discrete AdaBoost, που χρησιμοποιήθηκαν κι από τους Viola & Jones, και επι πλέον, η Real AdaBoost και η Gentle AdaBoost. Από αυτές η Gentle AdaBoost εμφάνισε τα καλύτερα αποτελέσματα και σε μικρότερους χρόνους.

Ο ταξινομητής που ταξινομεί ορθότερα τα δείγματα εκπαίδευσης με τη χρήση του αποτελεσματικότερου CART δένδρου, προστίθεται σε κάθε γύρω της ενίσχυσης (boosting). Ο κάθε ισχυρός ταξινομητής  $F$  είναι ένα σταθμισμένο άθροισμα αδύναμων ταξινομητών  $f_i$  από CART δένδρα ενός ή περισσότερων κόμβων:

$$F = \text{sign}(c_1 f_1 + c_2 f_2 + \dots + c_n f_n).$$

Τέλος, οι ισχυροί ταξινομητές συνδυάζονται σε διάταξη καταρράκτη (cascade). Ο αριθμός των αδύναμων ταξινομητών που αποτελούν το ισχυρό ταξινομητή αυξάνεται, καθώς αυξάνεται το επίπεδο του καταρράκτη ανιχνευτή, ώστε να επιτυγχάνεται ένας συγκεκριμένος ρυθμός εσφαλμένων ανιχνεύσεων για δεδομένο αριθμό επιτυχών ανιχνεύσεων [4,5].

### 3.8 Σύνοψη του Συστήματος Ανίχνευσης Προσώπου

- Υπολογίζονται απλά βαθμωτά χαρακτηριστικά. Υπάρχει ένας μεγάλος αριθμός από υποψήφια χαρακτηριστικά τύπου Haar.
- Επιλέγεται ένα μικρό υποσύνολο από αυτά και οι αντίστοιχοι αδύναμοι ταξινομητές από δένδρα απόφασης εκπαιδεύονται χρησιμοποιώντας AdaBoost.
- Ο ισχυρός ταξινομητής κατασκευάζεται σαν ένας γραμμικός συνδυασμός των αδύναμων και είναι το αποτέλεσμα του αλγόριθμου AdaBoost .
- Ο τελικός ανιχνευτής δημιουργείται από έναν ή από μία διαδοχή, σε διάταξη καταρράκτη, ισχυρών ταξινομητών που χρησιμοποιούν διαρκώς περισσότερα χαρακτηριστικά.
- Η προεπεξεργασία προετοιμάζει την εικόνα που θα ανιχνευτεί.
- Η μετεπεξεργασία συγχωνεύει ή απορρίπτει τις ανιχνεύσεις.

### 3.9 Μέθοδος Haar+AdaBoost : Συμπεράσματα

Αναφορικά με την προσέγγιση Haar+AdaBoost [2,3,4,5], μπορούμε να εξάγουμε τα παρακάτω συμπεράσματα [11] σχετικά με τα σύνολα των χαρακτηριστικών, τους αλγόριθμους boosting, τους αδύναμους ταξινομητές, τα μεγέθη των υποπαραθύρων, και τα μεγέθη του συνόλου εκπαίδευσης:

- Οι AdaBoost μέθοδοι εκπαίδευσης στην ανίχνευση προσώπων είναι οι πιο αποτελεσματικές μέχρι σήμερα. Αναφορικά με τα επίπεδα ανίχνευσης και σφαλμάτων, συγκρίνονται με την μέθοδο των νευρωνικών δικτύων του Rowley κ.ά. [19], αλλά σε μερικές περιπτώσεις είναι αρκετές φορές ταχύτερες.
- Ένα υπερπλήρες σύνολο από χαρακτηριστικά τύπου Haar είναι αποτελεσματικό για ανίχνευση προσώπου. Η χρήση της μεθόδου εικόνας ολοκληρώματος κάνει τον υπολογισμό αυτών των χαρακτηριστικών εφικτό και ανεξάρτητο από την κλίμακα.
- Τα εκτεταμένα χαρακτηριστικά τύπου Haar βοηθούν στην ανίχνευση των περιστραμμένων προσώπων.
- Η εκπαίδευση AdaBoost μπορεί να επιλέξει το καλύτερο υποσύνολο από ένα ευρύ σύνολο χαρακτηριστικών και να κατασκευάσει έναν ισχυρό μη γραμμικό ταξινομητή.
- Η διάταξη καταρράκτη βελτιώνει σημαντικά την ταχύτητα ανίχνευσης και μειώνει αποτελεσματικά τα σφάλματα με μικρό κόστος στους χρόνους ανίχνευσης.
- Πιο σύνθετοι αδύναμοι ταξινομητές όπως μικρά δένδρα CART μπορούν να μοντελοποιήσουν δεύτερης ή/και τρίτης τάξης εξαρτήσεις των χαρακτηριστικών, και μπορούν να είναι επωφελείς σε μη γραμμική επεξεργασία της ανίχνευσης προσώπου.
- Το ιδανικό μέγεθος του υπο-παραθύρου για την επεξεργασία ανίχνευσης προσώπων φαίνεται να είναι 20×20~24×24 pixels.
- Πιθανές βελτιώσεις μπορεί να είναι εφικτές με τον σχεδιασμό επιπλέον χαρακτηριστικών συμπληρωματικά στα ήδη υπάρχοντα, που υιοθετούν πιο προχωρημένες τεχνικές

εκπαίδευσης, και που θα μπορούσαν να καταλήξουν σε πιο σύνθετους ταξινομητές, αποφεύγοντας το πρόβλημα της υπερπροσαρμογής (overfitting).

- Γρήγορο και σχεδόν εύρωστο σύστημα που τρέχει σε πραγματικό χρόνο

- Αρνητικά στοιχεία :

- ο Απαιτείται η αναζήτηση στο χώρο και την κλίμακα

- ο Απαιτεί αρκετά θετικά και αρνητικά δείγματα

- ο Ανάλωση πολύ χρόνου στη φάση της εκπαίδευσης (μπορεί να απαιτεί μέρες εκπαίδευσης)

- ο Περιορισμένη προσέγγιση των περιπτώσεων διαφορετικού προσανατολισμού των προσώπων

- ο Απαιτεί αρκετή εργασία υλοποίησης.

## Κεφάλαιο 4

### ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ ΑΝΙΧΝΕΥΣΗΣ ΠΡΟΣΩΠΟΥ ΑΠΟ ΦΩΝΙΑ 90 ΜΟΙΡΩΝ

#### 4.1 Η Βιβλιοθήκη OpenCV

Η βιβλιοθήκη "Open Source Computer Vision Library" (OpenCV) είναι μία ελεύθερα διαθέσιμη, ανοικτού κώδικα συλλογή από ρουτίνες σε C++, που αφορούν την τεχνητή όραση. Η OpenCV παρέχει μια σταθερή υποδομή για την τεχνητή όραση που επιτρέπει στους ειδικούς να δουλέψουν σε ένα υψηλότερο επίπεδο από το στοιχειώδες. Με την τύπου BSD (Berkeley Software Distribution) άδεια χρήσης της, η OpenCV παρέχεται από την Intel για ελεύθερη εμπορική και ερευνητική χρήση. Η υποστήριξη της OpenCV στην τεχνητή όραση είναι εκτεταμένη. Υποστηρίζει ρουτίνες για την ανάγνωση, απεικόνιση, επεξεργασία και αποθήκευση απλών φωτογραφικών εικόνων αλλά και κινηματογραφικών ταινιών. Παρέχονται ένα πλήθος από ρουτίνες για την επεξεργασία εικόνων και υποστηρίζονται ρουτίνες υπολογιστικής γεωμετρίας, αναγνώρισης προτύπων και αλγόριθμοι ανίχνευσης αντικειμένων.

#### 4.2 Ανίχνευση Προσώπων με την OpenCV

Η OpenCV [22] 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.

Ο ταξινομητής αντικειμένων που προσφέρει είναι αυτός που έχει προταθεί από τους Viola & Jones [2,3] και έχει βελτιωθεί από τον Lienhart κ.ά. [4,5]. Ο ταξινομητής για τα πρόσωπα έχει εκπαιδευτεί με εκατοντάδες θετικά δείγματα προσώπων μεγέθους 20x20 και αυθαίρετα αρνητικά δείγματα ίδιου μεγέθους. Η OpenCV περιέχει την εφαρμογή Haartraining για την εκπαίδευση ενός ταξινομητή με ένα δεδομένο σύνολο θετικών και αρνητικών δειγμάτων, που αφορούν ένα συγκεκριμένο αντικείμενο. Η εφαρμογή αυτή δημιουργεί ένα xml αρχείο για Ανίχνευση Προσώπων και Σημείων Ενδιαφέροντος σε Πρόσωπα, 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 αποκρίνεται με "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. Έτσι μια χαμηλού επιπέδου ρουτίνα, για παράδειγμα, περιέχει τη δυνατότητα στον χρήστη να ελέγξει μία συγκεκριμένη περιοχή της εικόνας για το αν περιέχει ή όχι πρόσωπο. Βοηθητικές ρουτίνες υπολογίζουν τις εικόνες ολοκληρώματα, και ρυθμίζουν την κλίμακα αναζήτησης του ταξινομητή για πρόσωπα διαφορετικών μεγεθών κλπ. Αντίθετα, μία υψηλού επιπέδου συνάρτηση, όπως η cvHaarDetectObjects κάνει όλα τα προηγούμενα, χρησιμοποιώντας τον ταξινομητή που είναι αποθηκευμένος σε xml αρχείο, παρέχοντας αυτόματα τα αποτελέσματα ανίχνευσης στο χρήστη, πράγμα που είναι τις περισσότερες φορές αρκετό. Ανίχνευση Προσώπων και Σημείων Ενδιαφέροντος σε Πρόσωπα

#### 4.2.1 Η Εκπαίδευση του Ταξινομητή

Η κατασκευή του ταξινομητή και η εκπαίδευσή του γίνεται με την εφαρμογή Haartraining. Η κλήση της εφαρμογής γίνεται ως εξής:

```
Haartraining  
-data eyes_classifier  
-vec eyes.vec -w 20 -h 15  
-bg backgrounds.idx  
-nstages 15  
-nsplits 1  
-minhitrate 0.995  
-maxfalsealarm 0.5
```

Στο παραπάνω παράδειγμα ο ταξινομητής θα αποθηκευτεί στο αρχείο eyes\_classifier.xml. Τα θετικά δείγματα εκπαίδευσης θα ληφθούν από το αρχείο eyes.vec και θα έχουν διαστάσεις 20x15. Τα αρνητικά δείγματα θα ληφθούν από το αρχείο backgrounds.idx. Ο καταρράκτης ταξινομητής θα αποτελείται από nstages επίπεδα. Κάθε επίπεδο εκπαιδεύεται ώστε ο ταξινομητής που περιέχει να εμφανίζει ρυθμό θετικών ανιχνεύσεων τουλάχιστον minhitrate και ρυθμό εσφαλμένων ανιχνεύσεων το πολύ maxfalsealarm. Κάθε αδύναμος ταξινομητής θα έχει ακριβώς nsplits κόμβους (το 1 αντιστοιχεί σε ρίζα δένδρου απόφασης).

#### 4.2.2 Η Συνάρτηση Ανίχνευσης

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.

Αφού προχωρήσει η διαδικασία και συλλεγούν τα υποψήφια ορθογώνια (περιοχές που επαληθεύονται από τον ταξινομητή), ομαδοποιούνται σε ομάδες επικαλυπτόμενων

ορθογωνίων και επιστρέφονται σαν μία σειρά από μέσα ορθογώνια πλαίσια, ένα για κάθε ομάδα. Ανίχνευση Προσώπων και Σημείων Ενδιαφέροντος σε Πρόσωπα

Οι προκαθορισμένες παράμετροι (scale\_factor=1.1, min\_neighbors=3, flags=0) είναι ρυθμισμένες για μεγαλύτερης ακρίβειας αν και χρονοβόρα ανίχνευση.

Για ταχύτερη λειτουργία σε εικόνες βίντεο οι ρυθμίσεις είναι: scale\_factor=1.2, min\_neighbors=2, flags=CV\_HAAR\_DO\_CANNY\_PRUNING, min\_size=<minimum possible face size>.

Η συνάρτηση cvHaarDetectObjects δέχεται τις εξής παραμέτρους:

```
CvSeq* cvHaarDetectObjects( const CvArr* image,  
                            CvHaarClassifierCascade* cascade,  
                            CvMemStorage* storage,  
                            double scale_factor=1.1,  
                            int min_neighbors=3,  
                            int flags=0,  
                            CvSize min_size=cvSize(0,0) );
```

### **image**

Η εικόνα στην οποία ανιχνεύονται τα αντικείμενα.

### **cascade**

Η αναφορά στον χρησιμοποιούμενο ταξινομητή.

### **storage**

Μνήμη για την αποθήκευση της σειράς των ορθογωνίων που ανιχνεύτηκαν τα αντικείμενα.

### **Ποσοστό αύξησης κλίμακας (scaleFactor)**

Η τέταρτη παράμετρος καθορίζει πόσο γρήγορα το OpenCV πρέπει να αυξήσει την κλίμακα για τις ανιχνεύσεις προσώπου με κάθε πέρασμα που κάνει σε μια εικόνα. Όσο υψηλότερος είναι ο δείκτης κάνει τον ανιχνευτή να τρέχει γρηγορότερα (με το τρέξιμο λιγότερων περασμάτων), αλλά εάν είναι πάρα πολύ υψηλός, μπορεί να πηδάει πάρα πολύ γρήγορα μεταξύ των κλιμάκων και να χάνει τα πρόσωπα. Η προεπιλογή του OpenCV είναι 1.1, με άλλα λόγια, η κλίμακα αυξάνεται κατά 1,1 δηλ. (10%) ανα πέρασμα [9].

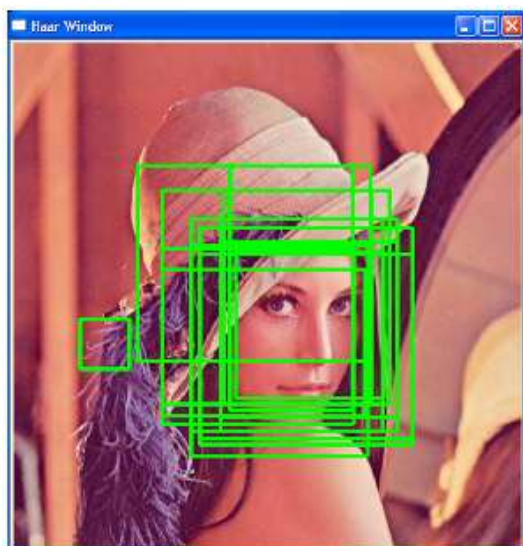
### **Ελάχιστο κατώτατο όριο γειτόνων (minNeighbors)**

Ένα από τα πράγματα που συμβαίνει "πίσω από τη σκηνή" όταν καλείτε ο ανιχνευτής προσώπου είναι ότι κάθε θετική περιοχή προσώπου παράγει πραγματικά πολλά χτυπήματα από τον ανιχνευτή Haar. Όπως θα δείτε στο σχημα 4 για την εικόνα παραδείγματος, lena.jpg. Η ίδια η περιοχή προσώπου παράγει τη μεγαλύτερη συστάδα των ορθογωνίων. Αυτοί επικαλύπτονται κατά ένα μεγάλο μέρος. Επιπλέον, υπάεχει μια μικρή ανίχνευση (στα αριστερά του παρατηρητή), και δύο μεγαλύτερες ανιχνεύσεις ελαφρώς πανω και αριστερα της κύριας συστάδας προσώπου.

Συνήθως, οι απομονωμένες ανιχνεύσεις είναι ψεύτικες ανιχνεύσεις, έτσι έχει νόημα να απορριφθούν αυτές. Έχει επίσης νόημα να συγχωνεύθουν κάπως οι πολλαπλές επικαλυπτομενες ανιχνεύσεις για κάθε περιοχή προσώπου σε μια ενιαία ανίχνευση. Το OpenCV τα κάνει και τα δύο πριν επιστρέψει τον κατάλογο ανιχνευμένων προσώπων του. Το βήμα συγχώνευσης πρώτα ομαδοποιεί τα ορθογώνια που επικάλυπτονται σε μεγάλο βαθμο και βρίσκει το μέσο ορθογώνιο για την ομάδα. Επειτα αντικαθιστά όλα τα ορθογώνια στην ομάδα με το μέσο ορθογώνιο.

Μεταξύ των απομονωμένων ορθογωνίων και των μεγάλων σχηματισμών ομάδας, υπάρχουν μικρότεροι σχηματισμοί ομάδας που μπορούν να είναι πρόσωπα, ή μπορούν να είναι ψεύτικες ανιχνεύσεις. Το κατώτατο όριο ελάχιστο-γειτόνων θέτει το όριο για την απόρριψη ή την κράτηση των ομάδων ορθογωνίων βασιζόμενο στο πόσες ακατέργαστες ανιχνεύσεις είναι στην ομάδα. Η προεπιλογή του openCV για αυτήν την παράμετρο είναι τρία, το οποίο σημαίνει να συγχωνεύσει τις ομάδες τριών ή περισσότερο και να απορρίψει τις ομάδες με λιγότερα ορθογώνια. Εάν διαπιστώνετε ότι ο ανιχνευτής προσώπου σας χάνει πολλά πρόσωπα, να δοκιμασετε να μειωσετε αυτό το κατώτατο όριο σε δύο ή ένα.

Εάν το θέσετε σε 0, το OpenCV θα επιστρέψει τον πλήρη κατάλογο ακατέργαστων ανιχνεύσεων από τον ταξινομητή Haar. Ενώ συντονίζετε τον ανιχνευτή προσώπου σας, είναι χρήσιμο να γίνεται αυτό για να δείτε ακριβώς τι συμβαίνει στο εσωτερικό του OpenCV. Η εξέταση των ακατέργαστων ανιχνεύσεων θα βελτιώσει τη διαίσθησή σας για τα αποτελέσματα της αλλαγής άλλων παραμέτρων, οι οποίες θα σας βοηθήσουν να τους συντονίσετε [9].



Εικόνα 4.1. OpenCV's internal detection rectangles. To see these, use `min_neighbors = 0`

### Έξυπνη σημαία περικοπής (flag)

Η έκτη παράμετρος είναι μια μεταβλητή σημαίων. Υπάρχουν μόνο δύο επιλογές: 0 ή `CV_HAAR_DO_CANNY_PRUNING`. Εάν επιλεγεί η έξυπνη επιλογή περικοπής, ο ανιχνευτής πηδά τις περιοχές εικόνας που είναι απίθανο να περιέχουν ένα πρόσωπο, αυτο μειώνει σημαντικά υπολογιστικό χρόνο και ενδεχομένως αποβάλλει μερικές ψεύτικες ανιχνεύσεις. Οι περιοχές που επιλεγονται να προσπεραστούν προσδιορίζονται με το τρέξιμο ενός ανιχνευτή ακρών (ο έξυπνος ανιχνευτής ακρών) στην εικόνα πριν τρέξει ο ανιχνευτής προσώπου.

Πάλι, η επιλογή εαν πρέπει να τεθεί αυτή η σημαία είναι μια επιλογή ανταλλαγής μεταξύ της ταχύτητας και της ανίχνευσης περισσότερων προσώπων. Η ρύθμιση αυτής της

σημαίας επιταχύνει την επεξεργασία, αλλά μπορεί να σας αναγκάσει για να χάσει μερικά πρόσωπα. Γενικά, μπορείτε να κάνετε καλά με το θέτετε, αλλά εάν you' επαν η κατοχή της δυσκολίας που ανιχνεύει τα πρόσωπα, που καθαρίζουν αυτήν την σημαία μπορεί να επιτρέψει σε σας για να ανιχνεύσει σοβαρότερα. Ο καθορισμός του κατώτατου ορίου ελάχιστο-γειτόνων σε 0 έτσι που μπορείτε να δείτε τις ακατέργαστες ανιχνεύσεις θα σας βοηθήσει να μετρήσετε καλύτερα την επίδραση της χρησιμοποίησης της έξυπνης περικοπής [9].

### **Ελάχιστη κλίμακα ανίχνευσης (windowSize)**

Η έβδομη παράμετρος είναι το μέγεθος του μικρότερου προσώπου για αναζήτηση. Εξ ορισμού, τίθεται στο μέγεθος των δειγμάτων που ο ταξινομητής είναι εκπαιδευμένος. Μπορείτε να το ανακαλύψετε με το άνοιγμα του XML αρχείου που χρησιμοποιείτε.

Ανάλογα με την ανάλυση που χρησιμοποιείται, αυτό το μέγεθος προεπιλογής μπορεί να είναι μια πολύ μικρή μερίδα της γενικής εικόνας σας. Μια εικόνα προσώπου τόσο μικρή μπορεί να μην είναι σημαντική ή χρήσιμη, και ανιχνεύοντας την αυξανετε τους κύκλους της CPU που θα μπορούσατε να χρησιμοποιήσετε για άλλους λόγους. Για αυτούς τους λόγους, και επίσης για να ελαχιστοποιήσετε τον αριθμό ανιχνεύσεων προσώπου που ο κώδικάς σας πρέπει να επεξεργαστεί, είναι καλύτερο να θέσετε την ελάχιστη κλίμακα ανίχνευσης μόνο τόσο μικρή όσο την χρειάζεστε αληθινά.

Για να θέσετε την ελάχιστη κλίμακα υψηλότερη από την προκαθορισμένη αξία, θέστε αυτήν την παράμετρο στο μέγεθος που θέλετε. Μια καλή εμπειροτεχνική μέθοδος είναι να θεσετε κάποιο μέρος του πλάτους ή του ύψους της εικόνας εισαγωγής ως ελάχιστη κλίμακα - παραδείγματος χάριν, 1/4 του πλάτους εικόνας. Όταν θετετε εσεις μια ελάχιστη κλίμακα εκτός από την προεπιλογή, σιγουρευτείτε ότι ο λόγος διάστασης(η αναλογία του πλάτους στο ύψος) είναι ο ίδιος με την προεπιλογή [9].

### **4.2.3 Οι Μέθοδοι Ανίχνευσης**

Η OpenCV παρέχει 4 μεθόδους ανίχνευσης προσώπων σε κατά μέτωπο (frontal) στάση, με τις οποίες έχει ήδη εκπαιδευτεί ο ταξινομητής και είναι αποθηκευμένες σε αντίστοιχα αρχεία xml, τα οποία μπορούν να φορτωθούν με τη χρήση της συνάρτησης cvLoad:

#### **• Stump-based 24x24 Discrete Adaboost with cascade Ανίχνευση Προσώπων και Σημείων Ενδιαφέροντος σε Πρόσωπα**

(haarcascade\_frontalface\_default.xml)

#### **• Stump-based 20x20 Gentle Adaboost with cascade**

(haarcascade\_frontalface\_alt.xml)

#### **• CART-based (2-nodes) 20x20 Gentle Adaboost with cascade**

(haarcascade\_frontalface\_alt2.xml)

#### **• Stump-based 20x20 Gentle Adaboost with tree of stage classifiers**

(haarcascade\_frontalface\_alt\_tree.xml)



Αυτές οι τέσσερις διαθέσιμες μέθοδοι ταξινόμησης θα αναφέρονται σε συντομία στη συνέχεια ως default, alt, alt2 και tree.

### 4.3 Εκπαίδευση Viola & Jones

Όπως ανέφερα παραπάνω, το OpenCV έρχεται ήδη με πολλούς εκπαιδευμένους ταξινομητές για την ανίχνευση. Ο μόνος αρκετά αξιόπιστος ταξινομητής εν τούτοις, είναι ο μετωπικός ανιχνευτής προσώπου. Εάν θέλετε το κάτι άλλο εκτός από τη μετωπική ανίχνευση προσώπου, η καλύτερη επιλογή είναι να χτισετε απο μόνι σας εναν καταρράκτη ταξινομητών που χρησιμοποιούν τη Haar-εκπαίδευση. Οτιδήποτε πρόκειται να χρειαστείτε χει εγκαταστάθει πιθανώς στον υπολογιστή σας με την εγκατάσταση openCV.

Τώρα ας δουμε έναν εύκολο τρόπο να χτιστεί ο καταρράκτης ωθημένων ταξινομητών χρησιμοποιώντας τα Haar- χαρακτηριστικά γνωρίσματα. Καταρχήν θα χρειαστείτε τις εφαρμογες «createsamples.exe», «haartraining.exe» και «performance.exe». Θα βρείτε αυτά τα αρχεία στο <OpenCV-directory>\bin . Δημιουργήστε έναν φάκελλο που θα δουλεψετε μέσα και να αντιγράψετε αυτά τα αρχεία μέσα στο φάκελλο. Τώρα πρέπει να οργανώσετε όλα τα στοιχεία που πρόκειται να χρειαστείτε για την κατάρτιση.

#### Βήμα 1 – Προετοιμασία

Δημιουργήστε τρεις φακέλλους μέσα στο φάκελλό σας, έναν για τα αρνητικά παραδείγματα, ένα για τα θετικά παραδείγματα και ένα ονομασμένο «data».

**Αρνητικά παραδείγματα:** είναι οι εικόνες που δεν περιλαμβάνουν πρόσωπα. Πρέπει να έχετε πολλές εικόνες (περίπου 5.000 έως 10.000) για αρνητικά παραδείγματα (υποβάθρου). Για να εργαστείτε με αυτές τις εικόνες, χρειάζεστε μια λιστα των αρχείων. Θα χρειαστείτε ένα αρχείο .txt με έναν κατάλογο εικόνων σας. Εάν δεν ξέρετε πώς να κάνετε αυτό, ανοίγετε το “command prompt window” πηγαίνετε στο φάκελλο που έχετε τα αρνητικά παραδείγματα και πληκτρολογήστε «dir /b /s > infofile.txt». Κατόπιν ανοίξτε το αρχείο και αφαιρέστε το “path” μπροστά από τα ονόματα αρχείου.

**Θετικά παραδείγματα:** είναι οι εικόνες που περιλαμβάνουν πρόσωπα και πρόκειται να τις χρησιμοποιήσετε για την εκπαίδευση σας. Είναι καλό να χρησιμοποιηθούν εικόνες που λαμβάνονται σε πραγματικές συνθηκες επειδή έχουν διαφορετικές αντανakλάσεις, φωτισμούς και υπόβαθρα. Πρέπει να χαρακτηρίσετε τα πρόσωπα σε αυτές τις εικόνες και να κάνετε ένα αρχείο .txt με έναν κατάλογο των συντεταγμένων χαρακτηρισμού, όπως την Εικ 4.2.

```
31 /1n004.pgm 1 5 12 179 179
32 /1p003.pgm 1 37 39 142 142
33 /1p004.pgm 1 33 32 149 149
34 /1r010.pgm 1 12 15 165 165
35 /1r011.pgm 1 14 15 160 160
36 /1s009.pgm 1 49 25 149 149
37 /1s010.pgm 1 47 24 150 150
38 /1t003.pgm 1 48 77 117 117
39 /01_002_2b_008.jpg 1 193 223 350 350
40 /01_002_2b_009.jpg 1 212 223 368 368
41 /01_002_2b_010.jpg 1 263 229 359 359
42 /01_002_2b_011.jpg 1 290 226 374 374
43 /01_002_2b_012.jpg 1 322 226 383 383
44 /01_002_2b_013.jpg 1 362 227 383 383
45 /01_002_2b_014.jpg 1 404 224 386 386
```

**Εικόνα 4.2:** Παράδειγμα .txt αρχείου θετικών παραδειγμάτων (από αρ. προς τα δεξιά): ον. αρχείου, αριθμός ανιχνεύσεων, συντεταγμένες ανίχνευσης ( τα 2 πρώτα νούμερα μας δίνουν την πάνω αρ. γωνία και τα 2 τελευταία το μήκος & το πλάτος της).

Σημειώσεις: Δεν μπορείτε να χρησιμοποιήσετε τις ίδιες εικόνες για την εκπαίδευση και τη δοκιμή του συστήματος, γιατί μην χρησιμοποιείτε όλα τα θετικά παραδείγματά σας για την εκπαίδευση. Προετοιμάστε την εκπαίδευση σας κάθε φορά ανάλογα με ποιες εικόνες πρόκειται να την δοκιμάσετε αργότερα.

## **Βήμα 2 - Δημιουργία Δειγμάτων**

Τώρα πρόκειται να χρησιμοποιήσετε το «createsamples.exe» για να δημιουργήσετε τα θετικά δείγματά σας από τις θετικές εικόνες σας. Το «createsamples.exe» πρόκειται να επιλέξει μόνο τη χαρακτηρισμένη θέση από τα θετικά παραδείγματά σας και να μειώσει το μέγεθος τους, για να καταστήσει την εκπαίδευση ευκολότερη. Το 20x20 είναι ένα καλό μέγεθος και χρησιμοποιείται αυτό. Μπορείτε να επιλέξετε το μέγεθος που θέλετε αλλά να θυμάστε όσο μεγαλύτερο το μέγεθος που θα επιλέξετε, τόσο πιο αργή πρόκειται να είναι η εκπαίδευση. Μόλις έχετε όλα τα σύνολά σας τακτοποιήμενα οι εικόνες αντικειμένου πρέπει «να συσκευαστούν» σε ένα vec-αρχείο στον φάκελλο «data». Η εντολή θα είναι κάπως έτσι:

```
createsamples.exe -info positives/train.txt -vec data/positives.vec -num 300 -w 20 -h 20
```

Πρέπει να ελεγχθεί εάν το vec-αρχείο περιέχει πραγματικά τις επιθυμητές εικόνες. Θα το δείτε αυτό με την ακόλουθη εντολή.

```
createsamples.exe -vec data/positives.vec -W 20 -h 20
```

## **Βήμα 3 – Εκπαίδευση**

Το επόμενο βήμα μετά από τη δημιουργία δειγμάτων είναι η εκπαίδευση του ταξινομητή. Εκτελείται από την εφαρμογή «haartraining.exe». Οι περισσότερες από τις παραμέτρους είναι καλές στις περισσότερες περιπτώσεις, οπότε μόνο μερικές από τις παραμέτρους θα αλλάξουν.

Ο αριθμός σταδίων πρέπει να είναι τουλάχιστον 20. Εάν αυτά είναι πάρα πολλά στάδια μπορείτε να ακυρώσετε την εκπαίδευση οποιαδήποτε στιγμή. Εάν αυτά είναι λιγα μπορείτε να ξαναξεκινήσετε την εκπαίδευση και τα στάδια θα προστεθούν στον υπάρχοντα καταρράκτη (η αφετηρία είναι το τελευταίο ολοκληρωμένο στάδιο). Εάν το αντικείμενο είναι συμμετρικό η παράμετρος «- nonsym» δεν απαιτείται. Αυτό κερδίζει χρόνο υπολογισμού χαρακτηριστικών γνωρισμάτων και χρήση μνήμης σε κάθε στάδιο.

Η εκπαίδευση θα αρχίσει από την ακόλουθη εντολή παρόμοια με αυτό:

```
haartraining.exe -data data/cascade -vec data/positives.vec -bg negatives/train.txt  
-npos 300 -nneg 1200 -nstages 20 -mem 2000 -mode ALL -w 20 -h 20
```

Ενώ η εκπαίδευση τρέχει, μπορείτε ήδη να πάρετε ένα «συναίσθημα» εάν θα είναι κατάλληλος ο ταξινομητής ή κάτι πρέπει να βελτιωθεί στην εκπαίδευση ή/και στους παραμέτρους σας. Η γραμμή που αρχίζει με «POS: » παρουσιάζει το ποσοστό χτυπήματος στο σύνολο δειγμάτων. Η επόμενη γραμμή αρχίζει με «NEG» δείχνει το ποσοστό ψεύτικων

συναγερμών. Το ποσοστό των θετικών πρέπει να είναι ίσο ή κοντά σε 1.0 (δεδομένου ότι είναι «στο στάδιο 0»). Το ποσοστό ψεύτικων συναγερμών πρέπει να φθάσει τουλάχιστον σε  $5 \cdot 10^{-6}$  (πέντε μηδενικά) έως ότου είναι ένας χρησιμοποιήσιμος ταξινομητής (εικ.4.3).

```

Parent node: 18
*** 1 cluster ***
POS: 304 322 0.944099
NEG: 1132 1.58019e-006
BACKGROUND PROCESSING TIME: 5792.31
Precalculation time: 1.00
+-----+-----+-----+-----+-----+-----+
| N | %SMP | F | ST. THR | HR | FA | EXP. ERR |
+-----+-----+-----+-----+-----+-----+
| 1 | 100% | - | -0.878586 | 1.000000 | 1.000000 | 0.591226 |
+-----+-----+-----+-----+-----+-----+
| 2 | 100% | - | -0.316812 | 1.000000 | 0.787102 | 0.238162 |
+-----+-----+-----+-----+-----+-----+
| 3 | 83% | - | -0.604094 | 1.000000 | 0.787986 | 0.269499 |
+-----+-----+-----+-----+-----+-----+
| 4 | 83% | - | -0.911880 | 0.996711 | 0.734099 | 0.228412 |
+-----+-----+-----+-----+-----+-----+
| 5 | 79% | - | -0.938654 | 0.996711 | 0.737633 | 0.188719 |
+-----+-----+-----+-----+-----+-----+
| 6 | 79% | - | -1.062000 | 0.996711 | 0.661661 | 0.175487 |
+-----+-----+-----+-----+-----+-----+
| 7 | 78% | - | -0.886720 | 0.996711 | 0.575972 | 0.163649 |
+-----+-----+-----+-----+-----+-----+
| 8 | 74% | - | -0.746645 | 0.996711 | 0.468198 | 0.151811 |
+-----+-----+-----+-----+-----+-----+
Stage training time: 454.37
Number of used features: 16

```

Εικόνα 4.3: Παράδειγμα εκπαίδευσης ως το στάδιο “STAGE 18”. NEG: 1.58019e-006 (five zeros) ώστε να έχουμε χρήσιμο ταξινομητή.

#### Βήμα 4 - Δοκιμη

Ένας ταξινομητής μπορεί να εξεταστεί με το εργαλείο απόδοσης ή μπορείτε απλά να αντικαταστήσετε το προ-εκπαιδευμένο .xml αρχείο από τον κώδικά σας με το νέο που μόλις δημιουργήσατε.

Εάν θέλετε να εξετάσετε τον ταξινομητή σας με το εργαλείο απόδοσης πρέπει να έχετε ένα διαφορετικό σύνολο θετικών και αρνητικών παραδειγμάτων και να τα προετοιμάσετε πάλι όπως «στο βήμα 1 - προετοιμασία». Το αρχείο πληροφοριών για αυτήν την χρησιμότητα απόδοσης δεν πρέπει να περιέχει το “path” της εικόνας. Μόνο το όνομα αρχείου.

Ένα παράδειγμα στον τρόπο με τον οποίο η συντάσσεται η εντολή μπορείτε να δείτε παρακατω.

*performance.exe -data data/cascade -info positives/testing/testing.txt -w 20 -h 20 -rs 30*

Θα περάσει από όλες τις εικόνες και θα προσπαθεί να ανιχνεύσει το αντικείμενο. Όταν ένα αντικείμενο βρίσκεται, θα σώζει την τρέχουσα εικόνα.

#### 4.4 Πειραματικό Μέρος

Σε αυτή την ενότητα θα παρουσιάσουμε μερικά αποτελέσματα και θα προσπαθήσουμε να τα αναλύσουμε και να τα εξιγήσουμε. Στο 4.4.1 θα αναφερθούμε λίγο περισσότερο στις παραμέτρους, την ευελιξία τους και πώς επηρεάζουν τα αποτελέσματα. Στο 4.4.2 πρόκειται να σας παρουσιάσω μερικά αποτελέσματα από το προ-εκπαιδευμένο σύστημα και θα αναφέρουμε μερικά πλεονεκτήματα και μειονεκτήματα, και τελικά στο 3.3 θα δούμε μερικά αποτελέσματα από το επανεκπαιδευμένο σύστημα και θα τα συγκρίνουμε με

προ-εκπαιδευμένα. Αλλά πρώτα ας εξηγήσουμε μερικούς όρους που θα συναντήσουμε αργότερα και θα μας βοηθήσουν να διαβάσουμε ευκολότερα και να κατανοήσουμε τα αποτελέσματα.

Αφού δοκιμάσαμε τον προ-εκπαιδευμένο κώδικα με αρκετές εικόνες, χωρίσαμε τα αποτελέσματα σε 5 κατηγορίες.

**True Positive in Face Image (T.P.F.I)**: είναι η σωστή ανίχνευση ενός προσώπου. Έχουμε μια περίπτωση (T.P.F.I) όταν σε μια εικόνα που περιέχει πρόσωπο το σύστημα κάνει σωστή ανίχνευση.

**False Positive in Face Image (F.P.F.I)**: είναι μια λανθασμένη ανίχνευση ενός προσώπου. Έχουμε μια περίπτωση (F.P.F.I) όταν σε μια εικόνα που περιέχει πρόσωπο το σύστημα κάτι άλλο και όχι το πρόσωπο.

**True Negative in Non-Face Image (T.N.N.F)**: είναι μια σωστή μη-ανίχνευση. Έχουμε μια περίπτωση (T.N.N.F) όταν σε μια εικόνα που δεν περιέχει πρόσωπο το σύστημα σωστά δεν ανιχνεύει τίποτα.

**False Negative in Face Image (F.N.F.I)**: έχουμε μια περίπτωση (F.N.F.I) όταν σε μια εικόνα που περιέχει πρόσωπο το σύστημα λανθασμένα δεν ανιχνεύει τίποτα.

**False Positive in Non-Face Image (F.P.N.F)**: έχουμε μια περίπτωση (F.P.N.F) όταν σε μια εικόνα που δεν περιέχει πρόσωπο το σύστημα λανθασμένα ανιχνεύει κάτι.

#### 4.4.1 Έρευνα Παραμέτρων

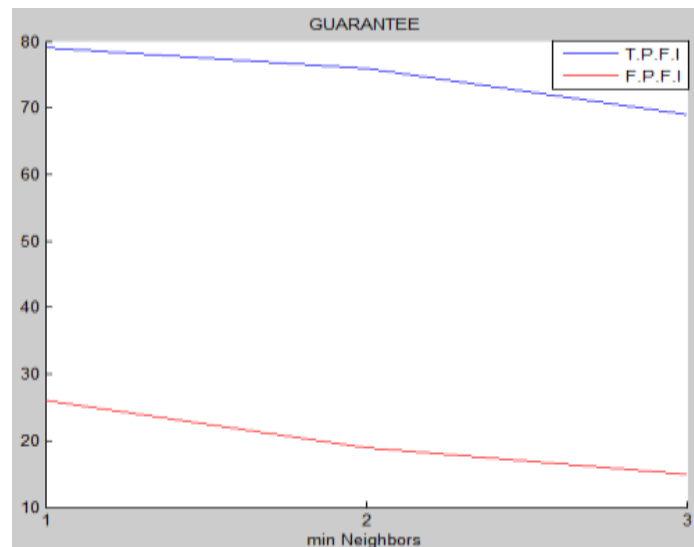
Προσπάθησαμε να εξηγήσουμε το πώς κάθε παράμετρος επιρρεάζει τα αποτελέσματα. Προκειμένου να το πετύχουμε κρατήσαμε 2 παραμέτρους σταθερές και μεταβάλαμε κάθε φορά μόνο τη μία. Το κάναμε αυτό για κάθε βάση δεδομένων χωριστά παίρνοντας 3 διαφορετικά σχεδιαγράμματα, ένα για κάθε παράμετρο:

1. Στον πρώτο σχεδιάγραμμα κρατάμε σταθερά το minSize και το scaleFactor και μεταβάλλουμε το minNeighbors.
2. Δεύτερο σχεδιάγραμμα οι τιμές minNeighbors και minSize παραμένουν οι ίδιες και αλλάζει μόνο το windowSize.
3. Και στο τρίτο σχεδιάγραμμα οι τιμές minNeighbors και windowSize παραμένουν σταθερές και αλλάζει η παράμετρος scale factor.

- **GUARANTEE**

Σε αυτήν την βάση δεδομένων έχουμε 174 εικόνες που περιέχουν πρόσωπα. Η γραμμή T.P.F.I (η μπλε γραμμή) παρουσιάζει σωστές ανιχνεύσεις και η γραμμή F.P.F.I (η κόκκινη γραμμή) παρουσιάζει τις λανθασμένες ανιχνεύσεις.

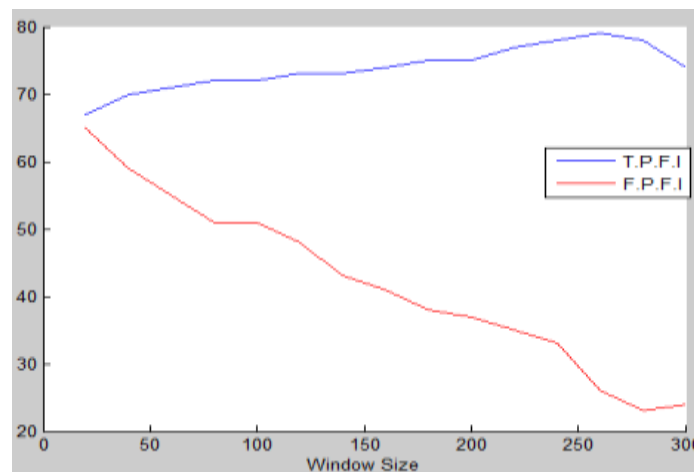
1.



Εικ 4.4: ScaleFactor =1.05 , windowSize = 260

Σε αυτό το σχεδιάγραμμα μπορούμε να δούμε τι συμβαίνει όταν αυξάνουμε την τιμή minNeighbors. Όπως αναφέραμε στην παράγραφο 4.2.2 όταν αυξανουμε το minNeighbors το σύστημα χρειάζεται μια μεγαλύτερη ομάδα ανιχνύσεων για να αποφασίσει εάν υπάρχει ένα πρόσωπο ή όχι.

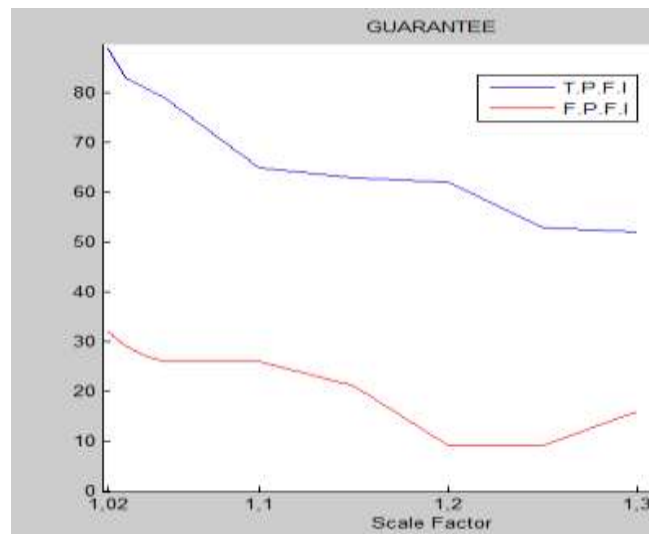
2.



Εικόνα 4.5: ScaleFactor = 1.05 , minNeighbors = 1

Στην εικ. 4.5 βλέπουμε πόσο σημαντικό είναι να βρεθεί η σωστή τιμή του windowSize. επιλέγοντας μια πολύ μικρή τιμή όχι μόνο οι σωστές ανιχνεύσεις δεν είναι καλές, αλλά παίρνουμε και πολλές ψεύτικες ανιχνεύσεις επειδή όταν αρχίζει το σύστημα μια εικόνα με τόσο μικρό παράθυρο βρίσκει πάντα πολλές λανθασμένες ανιχνεύσεις ορθογωνίων στο υπόβαθρο.

3.



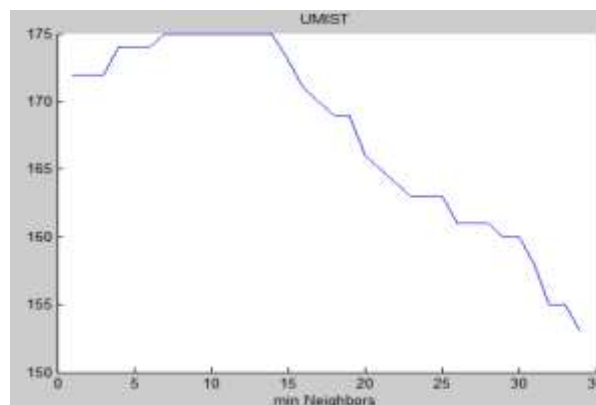
Εικ 4.6: minNeighbors = 1 , windowSize = 260

Σε αυτήν την εικόνα μπορούμε να δούμε πώς το scaleFactor επηρεάζει το σύστημα. Όπως ανέφερα στην παράγραφο 4.2.2 με την επιλογή μιας μικρής κλίμακας παράγοντα όχι μόνο καθιστάτε το σύστημα πιο αργό αλλά αυξάνετε επίσης τον αριθμό των λανθασμένες ανιχνεύσεις επειδή το σύστημα ανιχνεύει τον κώδικα πάρα πολλοί χρόνοι. Και δεδομένου ότι αυξάνετε το scaleFactor το σύστημα γίνεται πολύ γρηγορότερο αλλά χάνει στην ακρίβεια επειδή αυξάνεται τόσο γρήγορα ότι χάνουμε τα πρόσωπα.

- **UMIST**

Σε αυτήν την βάση δεδομένων έχουμε 217 εικόνες που περιέχουν πρόσωπα. Η γραμμή T.P.F.I (η μπλε γραμμή) παρουσιάζει τις σωστές ανιχνεύσεις ενώ η γραμμή F.P.F.I (η κόκκινη γραμμή) παρουσιάζει τις λανθασμένες.

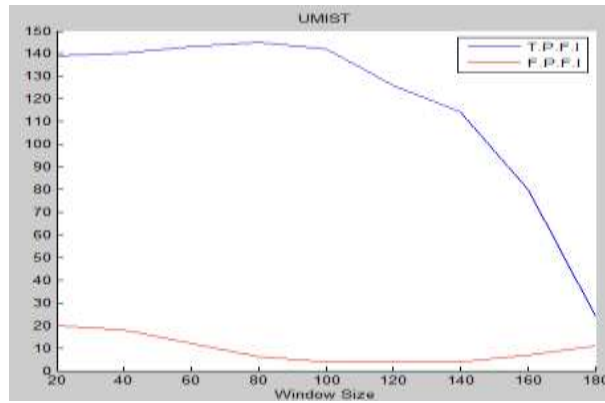
1.



Εικ 4.7: : ScaleFactor =1.05 , windowSize = 80

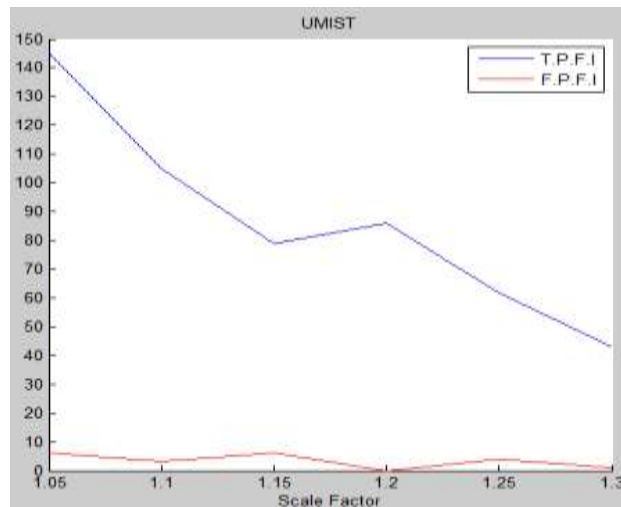
Στην εικ. 4.7 μπορούμε να δούμε πώς το `scaleFactor` και το `minNeighbors` υσχετίζονται. Επιλέγοντας τόσο μικρό `scaleFactor`, το σύστημα τόσο πολλές φορές την εικόνα και έκανε πολύ μεγάλες ομάδες ορθογωνίων. Αυτός είναι ο λόγος που η τιμή του `minNeighbors` πρέπει να είναι τόσο μεγάλη (πάνω από 15) για να δουμε αλλαγές στα αποτελέσματα.

2.



Εικ 4.8: `ScaleFactor = 1.05` , `minNeighbors = 1`

3

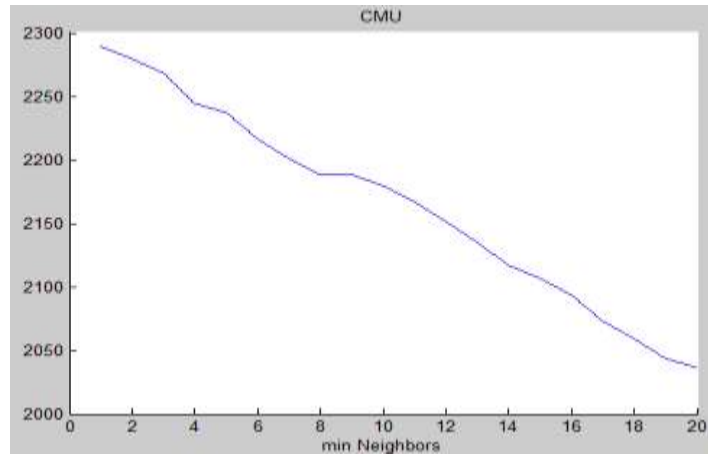


Εικ 4.9: `minNeighbors = 1` , `windowSize = 80`

- **CMU**

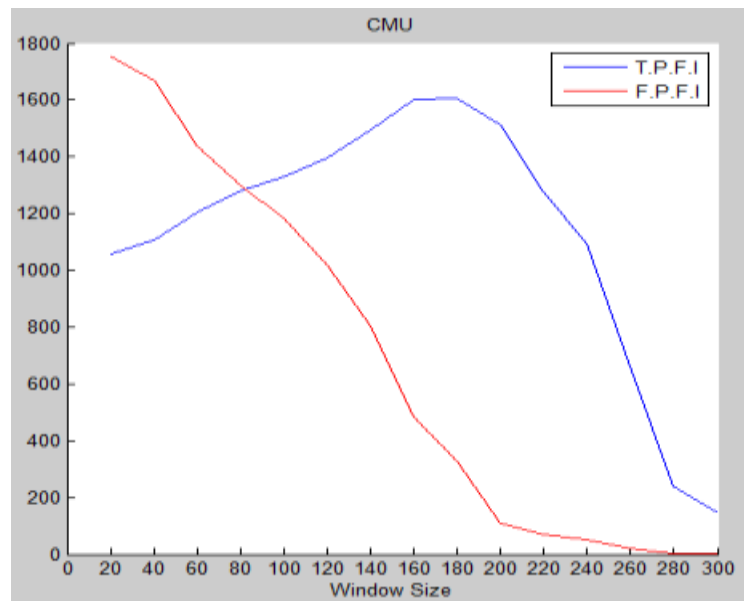
Σε αυτήν την βάση δεδομένων έχουμε 3593 εικόνες που περιέχουν πρόσωπα. Η γραμμή T.P.F.I (η μπλε γραμμή) παρουσιάζει σωστές ανιχνεύσεις και η γραμμή F.P.F.I (η κόκκινη γραμμή) παρουσιάζει τις λανθασμένες ανιχνεύσεις

1.



Εικ 4.10: : ScaleFactor =1.001 , windowSize = 200

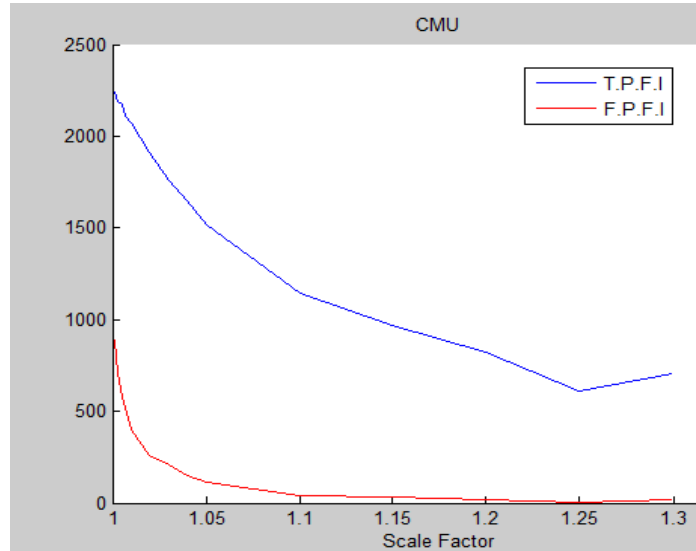
2.



Εικ 4.11: ScaleFactor = 1.05 , minNeighbors = 1



3.

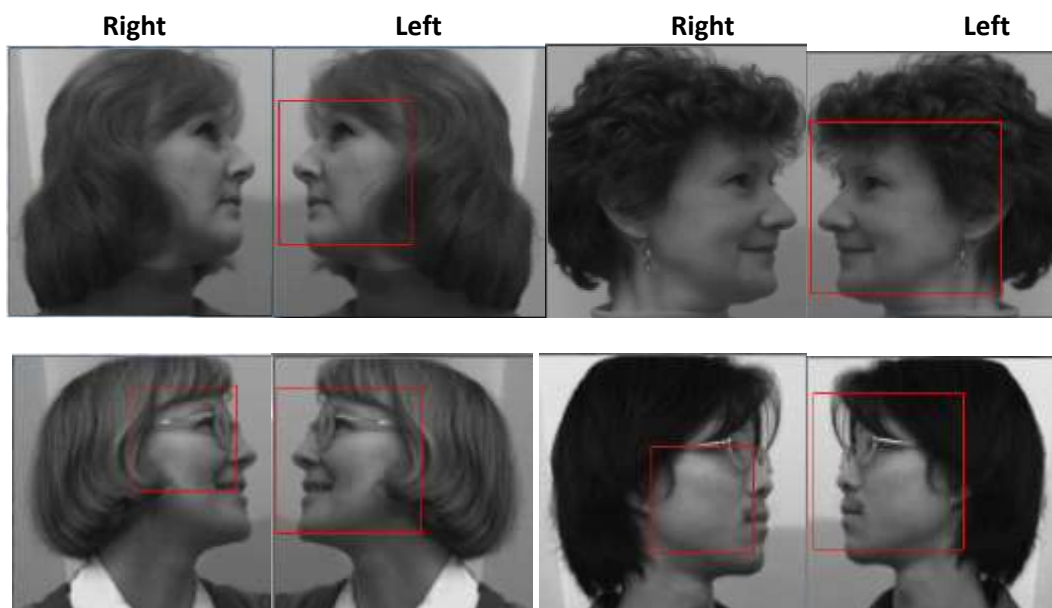


Εικ 4.12: : minNeighbors = 1 , windowSize = 2

#### 4.4.2 Το φαινόμενο των διαφορετικών αποτελεσμάτων μεταξύ αριστερών και δεξιών προφίλ

Ενώ εξετάζαμε το σύστημα με τις εικόνες παρατηρήσαμε ένα ενδιαφέρον πράγμα. Παρατηρήσαμε ότι οι περισσότερες από τις σωστές ανιχνεύσεις ήταν στις εικόνες που παρουσίαζαν αριστερό προφίλ ενός προσώπου και οι περισσότερες από τις λανθασμένες ανιχνεύσεις ήταν στις εικόνες που έδειχναν δεξιά προφίλ προσώπων. Έτσι αποφασίσαμε να πάρουμε όλες τις εικόνες με τα δεξιά προφίλ, να τις αντιστρεψουμε ώστε να φαίνονται σαν αριστερές και να αφήσουμε τις ήδη αριστερές εικόνες όπως ήταν και να επανεξετάσουμε το σύστημα και στις 3 βάσεις δεδομένων.

Στην **εικ.4.13** μπορούμε να δούμε μερικά παραδείγματα.



Εικόνα 4.13. Some examples about the differences between the right and the left profile images.

Αυτό που κάναμε ήταν αφού αλλάξαμε όλες τις διεξιές εικόνες σε αριστερές να επιλέξουμε τις παραμέτρους που είχαν τα καλύτερα αποτελέσματα σε κάθε βάση δεδομένων μέχρι τώρα και εξετάσουμε το σύστημα πάλι. Όπως μπορείτε να δείτε παρακάτω, η διαφορά στο αποτέλεσμα ήταν σημαντική και συγκεκριμένα στην βάση δεδομένων UMIST όπου οι αρχικές εικόνες περιείχαν μόνο δεξιά προφίλ σχεδιαγράμματος.

- **Guarantee profile**

Total images : 764

Include face: 174

Don't include face: 590

Parameters : 1.20\_1\_0\_240\_240

Original (regular images)

True Positive F.I	62	8.12 %		
False Positive F.I	14	1.83%		
True Negative N.F	578	75.65 %	Total True:	83.77 %
False Negative F.I	98	12.83 %		
False Positive N.F	12	1.57 %	Total False:	16.23 %

Left profile (mirror images)

True Positive F.I	75	9.81 %		
False Positive F.I	12	1.57%		
True Negative N.F	578	75.65 %	Total True:	85.46 %
False Negative F.I	87	11.38 %		
False Positive N.F	12	1.57 %	Total False:	14.54 %

- **UMIST profile**

Total images : 217

Include face: 217

Don't include face: 0

Parameters : 1.002\_5\_0\_80\_80

Original (regular images)

True Positive F.I	174	80.18 %		
False Positive F.I	12	5.53%		
True Negative N.F	0	00.00 %	Total True:	80.18 %
False Negative F.I	31	14.29%		
False Positive N.F	0	00.00 %	Total False:	19.82 %

Left profile (mirror images)

True Positive F.I	208	95.86 %		
False Positive F.I	8	3.68%		
True Negative N.F	0	00.00%	Total True:	95.86 %
False Negative F.I	1	0.46 %		
False Positive N.F	0	00.00%	Total False:	4.14 %

- **CMU profile**

Total images : 3593  
 Include face: 3593  
 Don't include face: 0

Parameters : 1.001\_1\_0\_200\_200

Original (regular images)

True Positive F.I	2237	62.26 %		
False Positive F.I	886	24.66%		
True Negative N.F	0	00.00 %	Total True:	62.26 %
False Negative F.I	470	13.08%		
False Positive N.F	0	00.00 %	Total False:	37.74 %

Left profile (mirror images)

True Positive F.I	2931	81.57 %		
False Positive F.I	114	3.18%		
True Negative N.F	0	00.00 %	Total True:	81.57 %
False Negative F.I	548	15.25 %		
False Positive N.F	0	00.00 %	Total False:	18.43 %

#### **4.4.3 Επανεκπαιδευμένο Σύστημα**

Επανεκπαίδευσαμε το σύστημα αρκετές φορές με διαφορετικά θετικά παραδείγματα κάθε φορά ώστε να καταλάβουμε πώς λειτουργεί η εφαρμογή εκπαίδευσης. Παρακατω συγκρίνουμε τα αποτελέσματα των επανεκπαιδευμένων συστημάτων με τα προ-εκπαιδευμένα που χρησιμοποιήσαμε στην πρώτη φορά. Την πρώτη φορά χρησιμοποιήσαμε 201 θετικά παραδείγματα και 800 αρνητικ. Αυτά τα 201 θετικά παραδείγματα λήφθηκαν από 119 διαφορετικούς ανθρώπους. Συγκεκριμένα 100 άνθρωποι ήταν από τη βάση δεδομένων CMU, 1 άτομο ήταν από την GUARANTEE, 64 εικόνες όλες τους με διαφορετικούς φωτισμούς και πόζες. Και 18 άνθρωποι ήταν από την βάση UMIST, 2 εικόνες από καθε έναν με διαφορετική πόζα κάθε φορά, 38 εικόνες. Εξέτασαμε το σύστημα στις υπόλοιπες εικόνες των βάσεων CMU και GUARANTEE και τα αποτελέσματα είναι τα εξής:

#### **CMU**

Total images: 2345

Parameters: 1.005\_1\_200

Re-trained System

True Positive F.I	2052	87.51 %		
False Positive F.I	84	3.58 %		
True Negative N.F	0	00.00 %	Total True:	87.51 %
False Negative F.I	209	8.91 %		
False Positive N.F	0	00.00 %	Total False:	11.49 %

Pre-trained System

True Positive F.I	1381	58.89 %		
False Positive F.I	384	16.38 %		
True Negative N.F	0	00.00 %	Total True:	58.89 %
False Negative F.I	580	24.73 %		
False Positive N.F	0	00.00 %	Total False:	41.11 %

**GUARANTEE**

Total images : 514 100 %  
Include face: 110 21.4 %  
Don't include face: 404 78.6 %

Parameters: 1.05\_1\_260

Re-trained System

True Positive F.I	39	7.58 %		
False Positive F.I	32	6.22 %		
True Negative N.F	312	60.70 %	Total True:	68.29 %
False Negative F.I	39	7.58 %		
False Positive N.F	92	17.89 %	Total False:	31.71 %

Pre-trained System

True Positive F.I	63	12.26 %		
False Positive F.I	19	3.69 %		
True Negative N.F	123	23.93 %	Total True:	36.19 %
False Negative F.I	28	5.45 %		
False Positive N.F	281	54.67 %	Total False:	63.81 %

Τη δεύτερη φορά χρησιμοποιήσαμε 322 θετικά παραδείγματα και 1200 αρνητικά παραδείγματα. Αυτά τα 322 θετικά παραδείγματα λήφθηκαν από 69 διαφορετικούς ανθρώπους. Συγκεκριμένα 50 άνθρωποι ήταν από τη βάση δεδομένων CMU, πήραμε 4 εικόνες από κάθε έναν με 2 διαφορετικούς φωτισμούς και 2 διαφορετικές πόζες. 220 εικόνες συλλέχθηκαν από τη βάση CMU με 20 διαφορετικούς φωτισμούς και 10 διαφορετικές πόζες. 1 άτομο ήταν από την GUARANTEE, 64 εικόνες όλες τους με τους διαφορετικούς φωτισμούς και πόζες. Και τέλος 18 άνθρωποι ήταν από τη βάση UMIST, 2 εικόνες από κάθε έναν με μια διαφορετική πόζα κάθε φορά, στο σύνολο 38 εικόνες.

Εξέτασαμε το σύστημα στο υπόλοιπο των εικόνων CMU και GUARANTEE. Όπως μπορείτε να δείτε από τα αποτελέσματα κατωτέρω εάν τα συγκρίνουμε με το προηγούμενο επανεκπαιδευμένο σύστημα, στη βάση δεδομένων CMU η τιμή T.P.F.I είναι λίγο χαμηλότερη επειδή αυτή τη φορά χρησιμοποιήσαμε τις εικόνες μόνο από 50 διαφορετικούς ανθρώπους της βάσης CMU. Αφ' ενός στην GUARANTEE έχουμε πολύ καλύτερα αποτελέσματα επειδή αυτή τη φορά χρησιμοποιήσαμε πολλές εικόνες με διαφορετικές πόζες από 90 βαθμούς μέχρι σχεδόν το πλήρες μετωπικό πρόσωπο.

## CMU

Total images: 2345

Parameters: 1.005\_1\_200

### Re-trained System

True Positive F.I	1992	84.94 %		
False Positive F.I	218	9.30 %		
True Negative N.F	0	00.00 %	Total True:	84.94 %
False Negative F.I	135	5.76 %		
False Positive N.F	0	00.00 %	Total False:	15.06 %

## GUARANTEE

Total images : 514 100 %  
Include face: 110 21.4 %  
Don't include face: 404 78.6 %

Parameters: 1.1\_1\_260

### Re-trained System

True Positive F.I	58	11.29 %		
False Positive F.I	10	1.95 %		
True Negative N.F	281	54.66 %	Total True:	65.95 %
False Negative F.I	42	8.17 %		
False Positive N.F	123	23.93 %	Total False:	34.05 %

## Κεφάλαιο 5

### ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΣΩΠΟΥ - ΤΕΧΝΙΚΕΣ – ΠΡΟΒΛΗΜΑΤΑ

#### 5.1. Αναγνώριση προσώπων

Το ανθρώπινο πρόσωπο δεν έχει μόνο ανατομικό και λειτουργικό ρόλο, αλλά έχει επιπλέον ένα γενικότερο ατομικό και διαπροσωπικό ρόλο. Αποτελεί για κάθε άνθρωπο ένα εντελώς ιδιαίτερο γνώρισμα, του οποίου η σπουδαιότητα έχει κοινωνικές, ψυχολογικές, καλλιτεχνικές και φιλοσοφικές προεκτάσεις.

Η αναγνώριση προσώπου είναι μια εναλλακτική πρόταση ταυτοποίησης ανθρώπων χωρίς τη χρήση εγγράφων, όπως είναι τα δακτυλικά αποτυπώματα, η ίριδα του ματιού κ.ά. Ο όρος αναγνώριση προσώπων αναφέρεται σε δυο διαφορετικά σενάρια εφαρμογής:

- την ταυτοποίηση (identification) και
- την πιστοποίηση (verification ή authentication).[23,24]

Και στις δυο περιπτώσεις, πριν από την αναγνώριση λαμβάνει χώρα το στάδιο της καταγραφής (enrollment) των εικόνων των οποίων η ταυτότητα ή/και η έκφραση είναι γνωστή. Κατά την καταγραφή υπολογίζονται τα χαρακτηριστικά (features) των προσώπων και αποθηκεύονται σε μια βάση δεδομένων που αποτελεί τη συλλογή (gallery) του συστήματος. Μετά την καταγραφή το σύστημα μπορεί να δεχτεί νέα άγνωστα πρόσωπα δοκιμής (test faces) και να αναγνωρίσει την ταυτότητα ή την έκφραση τους.

Κατά την ταυτοποίηση το σύστημα καλείται να βρει την ταυτότητα του ατόμου της συλλογής στο οποίο ανήκει η εικόνα δοκιμής. Το σύστημα απαντά στο ερώτημα "ποιος είναι;" ο εικονιζόμενος. Εδώ υπάρχουν δύο ακόμα υποπεριπτώσεις, η ταυτοποίηση ανοιχτού και κλειστού συνόλου.

Με τον όρο κλειστό σύνολο εννοούμε ότι η εικόνα δοκιμής ανήκει σε κάποιο από τα άτομα της συλλογής, ενώ με τον όρο ανοιχτό εννοούμε ότι η εικόνα δοκιμής μπορεί να ανήκει και σε άτομο που δεν έχει προηγουμένως καταγραφεί στη συλλογή. Στο κλειστό σύνολο, το σύστημα συγκρίνει την εικόνα δοκιμής μία προς μία με όλες τις εικόνες συλλογής για τον καθορισμό ενός μέτρου ομοιότητας (similarity score), σύμφωνα με το οποίο αποφασίζει σε ποιο άτομο ανήκει η εικόνα δοκιμής. Στο ανοιχτό σύνολο το καλύτερο μέτρο ομοιότητας συγκρίνεται και με ένα κατώφλι (threshold) για να ελεγχθεί μήπως η εικόνα δοκιμής δεν ανήκει σε κανένα από τα άτομα της συλλογής. Σε κάθε δοκιμή λοιπόν κατασκευάζεται ένας μονοδιάστατος πίνακας από τα μέτρα ομοιότητας της εικόνας δοκιμής με κάθε εικόνα συλλογής, στη συνέχεια τα άτομα της συλλογής κατατάσσονται με φθίνουσα σειρά των μέτρων ομοιότητας και τέλος επιστρέφεται η ταυτότητα του ατόμου με το καλύτερο μέτρο. [23,24]

Κατά την πιστοποίηση, το σύστημα αναγνώρισης προσώπων καλείται να εξακριβώσει αν η ταυτότητα που επικαλείται κάποιος χρήστης είναι αληθής ή όχι, καλείται δηλαδή να απαντήσει στο ερώτημα "είναι πράγματι αυτός που ισχυρίζεται ότι είναι;" κι έτσι να διακρίνει αν πρόκειται για εξουσιοδοτημένη χρήση (legitimate user) ή για πλαστο-προσωπία (impostor). Για να γίνει αυτό, η εικόνα δοκιμής συγκρίνεται με τις εικόνες συλλογής του ατόμου του οποίου η ταυτότητα επικαλείται και στη συνέχεια το μέτρο ομοιότητας που προκύπτει συγκρίνεται με ένα κατώφλι για την αποδοχή ή όχι του αιτήματος. Ενώ λοιπόν η ταυτοποίηση είναι μια διαδικασία σύγκρισης ένα-προς πολλά (1:N), η πιστοποίηση είναι μια διαδικασία σύγκρισης ένα προς ένα (1:1) κι έχει δυαδικό αποτέλεσμα: Ναι, Όχι.

Η απόδοση ενός συστήματος πιστοποίησης εξαρτάται από την επιλογή του κατωφλίου, γιατί αυτό καθορίζει τη συναλλαγή (trade-off) ανάμεσα στην ασφάλεια και την άνεση του χρήστη. Όσο υψηλότερο είναι το κατώφλι τόσο πιο αυστηρό γίνεται το σύστημα και η ασφάλεια αυξάνεται. Ταυτόχρονα όμως αυξάνεται και η πιθανότητα να απορριφθούν εξουσιοδοτημένοι χρήστες με αποτέλεσμα το σύστημα να γίνεται λιγότερο φιλικό προς το χρήστη. Το αντίστροφο συμβαίνει όταν μειώνεται το κατώφλι. Αυτή η συμπεριφορά του

συστήματος κωδικοποιείται στη χαρακτηριστική λειτουργίας δέκτη (Receiver Operating Characteristic-ROC) που δείχνει την τιμή του ρυθμού εσφαλμένης αποδοχής (False Accept Rate-FAR) και του ρυθμού εσφαλμένης απόρριψης (False Reject Rate-FRR) για κάθε τιμή του κατωφλίου.[23,24]

Μόλις τα δέκα τελευταία χρόνια, άρχισαν ουσιαστικά να ερευνώνται οι δυνατότητες της γεωμετρίας του προσώπου για την αναγνώριση προσώπων και εκφράσεων. Το υψηλό κόστος των αισθητήρων σε συνδυασμό με τη χαμηλή ποιότητα των δεδομένων που παρήγαγαν και το μεγάλο χρόνο απόκρισης οδήγησαν τη πλειοψηφία των ερευνητών στη χρήση έγχρωμων εικόνων ή εικόνων φωτεινότητας και την ανάπτυξη φωτομετρικών (photometry) μεθόδων αναγνώρισης.

Ένας τεράστιος αριθμός τεχνικών έχει προταθεί, στον τομέα της αναγνώρισης προσώπων, πολλές από τις οποίες έχουν ήδη χρησιμοποιηθεί για την ανάπτυξη εμπορικών συστημάτων.

Η έρευνα γύρω από την 2Δ ή 3Δ αναγνώριση προσώπων και εκφράσεων δέχτηκε σημαντική ώθηση, με την πρόοδο των αισθητήρων βάθους τα τελευταία χρόνια όπως και με τη διαθεσιμότητα δεδομένων στην επιστημονική κοινότητα για την διεξαγωγή πειραμάτων..



**Εικόνα 5.1: Αρχιτεκτονική συστημάτων αναγνώρισης προσώπων και εκφράσεων**

Όλα τα συστήματα αναγνώρισης προτύπων ακολουθούν μια τυπική αρχιτεκτονική που περιλαμβάνει τέσσερα βασικά στάδια επεξεργασίας:

- τη λήψη μετρήσεων για κάποιες από τις ιδιότητες του αντικειμένου,
- την προ-επεξεργασία των μετρήσεων για τη μείωση του θορύβου ή την κανονικοποίηση των μετρήσεων,
- την εξαγωγή χαρακτηριστικών με τα οποία γίνεται η διάκριση των προτύπων και
- την ταξινόμηση, που περιλαμβάνει τη σύγκριση των χαρακτηριστικών του αντικειμένου είτε με κάποια κατώφλια είτε με αντίστοιχα προ-αποθηκευμένα χαρακτηριστικά γνωστών αντικειμένων, ώστε να αποφασιστεί σε ποια κλάση κατατάσσεται το αντικείμενο. [23,24]

Τα συστήματα 3Δ αναγνώρισης προσώπων και εκφράσεων ακολουθούν την ίδια τυπική

αρχιτεκτονική με την επιπλέον ιδιαιτερότητα ότι τα δυο πρώτα στάδια, της καταγραφής του βάθους και της προ-επεξεργασίας, είναι ακριβώς τα ίδια, ενώ τα στάδια της εξαγωγής χαρακτηριστικών και της ταξινόμησης χρησιμοποιούν παρόμοιες τεχνικές.

Η καταγραφή της επιφάνειας του προσώπου γίνεται με μεθόδους που μπορούν να χωριστούν σε δυο μεγάλες κατηγορίες, τις ενεργητικές και τις παθητικές. Οι ενεργητικές μέθοδοι στηρίζονται στο φωτισμό της επιφάνειας με ένα γνωστό πρότυπο φωτός (light pattern). Το πρότυπο φωτός προβάλλεται μέσω ενός προβολέα πάνω στο αντικείμενο και μια κάμερα συλλαμβάνει το είδωλο του. Στη συνέχεια, για κάθε σημείο της επιφάνειας υπολογίζεται η αντίστοιχη θέση του πάνω στην εικόνα και μέσω αυτής της αντιστοιχίας και απλών τριγωνομετρικών υπολογισμών εκτιμάται η απόσταση του σημείου από το επίπεδο της κάμερας. Το πρότυπο φωτός μπορεί να είναι ένα απλό σημείο ή μια γραμμή. Επειδή όμως η χρήση τόσο απλών προτύπων καταλήγει σε χρονοβόρες διαδικασίες αντιστοίχισης,

τελευταία χρησιμοποιούνται πιο πολύπλοκα πρότυπα που επιτρέπουν παράλληλη επεξεργασία, όπως πλέγματα ή χρωματικά κωδικοποιημένες λωρίδες. Επίσης για να μην ενοχλείται ο χρήστης από την προβολή φωτός, γίνονται προσπάθειες για την αντικατάσταση των ορατών προτύπων από υπέρυθρες ακτίνες **Εικ 5.2**. [23,24,28]

Οι παθητικοί αισθητήρες χρησιμοποιούν στερεοσκοπικές μεθόδους για τον υπολογισμό του βάθους, στηρίζονται δηλαδή στην επεξεργασία δυο ή περισσότερων εικόνων του αντικειμένου από διαφορετικές όψεις. Το κύριο πρόβλημα σ' αυτές τις μεθόδους είναι το πρόβλημα της αντιστοίχισης, η εύρεση δηλαδή των εικονοστοιχείων (pixels) σε όλες τις εικόνες που αντιστοιχούν στο ίδιο 3D σημείο της επιφάνειας. Για την επίλυση του, συνήθως χρησιμοποιείται η μέθοδος της οπτικής ροής (optical flow) σε συνδυασμό με επιπολικούς περιορισμούς που περιορίζουν το εύρος της αναζήτησης. Η μέθοδος αυτή δεν μπορεί να εφαρμοστεί για αντικείμενα που δε διαθέτουν υφή (texture) και αυτό αποτελεί μειονέκτημα για την συγκεκριμένη μέθοδο. Αυτό βέβαια μπορεί να αντιμετωπιστεί εφαρμόζοντας ενεργητικές τεχνικές προβολής κατάλληλου πρότυπου υφής. Το ανθρώπινο πρόσωπο περιέχει αρκετή πληροφορία για την επίλυση του προβλήματος της αντιστοίχισης χωρίς να χρειάζεται η προβολή προτύπου υφής. Στην εικ. 5.2 φαίνεται ένας στερεοσκοπικός αισθητήρας που χρησιμοποιεί δυο όψεις του προσώπου για την εξαγωγή του βάθους. [23,24]



**Εικόνα 5.2: (α) και (β) Συσκευές καταγραφής βάθους**

Η επιλογή του αισθητήρα εξαρτάται κυρίως από την εφαρμογή για την οποία προορίζεται. Γενικά, είναι θέμα συμβιβασμού ανάμεσα στην ταχύτητα, το κόστος και την ποιότητα των παραγόμενων δεδομένων. Οι παθητικοί αισθητήρες είναι ταχύτεροι και φθηνότεροι, αλλά παράγουν δεδομένα με περισσότερο θόρυβο. Από την άλλη πλευρά, οι ενεργητικοί αισθητήρες είναι πιο ακριβείς, αλλά είναι ακριβότεροι και απαιτούν την ακινητοποίηση του χρήστη για το διάστημα της καταγραφής.

Οι δυο τεχνικές στηρίζονται σε φωτομετρικές μεθόδους (2Δ) για την εκτίμηση του βάθους. Αυτό σημαίνει ότι ενώ το βάθος εν γένει δεν εξαρτάται από τις συνθήκες φωτισμού, ωστόσο η καταγραφή (sensing) του μπορεί να επηρεάζεται από αυτές. [23,24,28]

## 5.2 ΕΦΑΡΜΟΓΕΣ ΑΝΑΓΝΩΡΙΣΗΣ ΠΡΟΣΩΠΩΝ

Οι εφαρμογές της αναγνώρισης προσώπων είναι ποικίλες. Οι σημαντικότερες από αυτές θα μπορούσαν να συνοψιστούν στις παρακάτω :

- Έλεγχος πρόσβασης. Η πιο κοινή από τις εφαρμογές της αναγνώρισης προσώπου είναι η εξακρίβωση της ταυτότητας φυσικών προσώπων για ελεγχόμενη πρόσβαση σε κτίρια



και περιοχές ασφαλείας είναι. Στο σημείο εισόδου μία κάμερα καταγράφει μία εικόνα του προσώπου, η οποία στη συνέχεια συγκρίνεται με προαποθηκευμένες εικόνες μίας βάσης δεδομένων. Η είσοδος επιτρέπεται μόνο αν υπάρξει ταύτιση με τις εικόνες ενός εξουσιοδοτημένου χρήστη. Τέτοιου τύπου συστήματα αναγνώρισης προσώπου εγκαθίστανται π.χ. σε αεροδρόμια, ώστε να διευκολύνουν τα πληρώματα των πτήσεων και το προσωπικό του αεροδρομίου να περνούν από διάφορα σημεία ελέγχου χωρίς να επιδεικνύουν ταυτότητα ή διαβατήριο.

- Ασφάλεια ηλεκτρονικών συναλλαγών. Χρησιμοποιείται για τη διεξαγωγή ασφαλών τραπεζικών και εμπορικών συναλλαγών μέσω Διαδικτύου ή κινητών τηλεφώνων, για την πρόσβαση σε δίκτυα υπολογιστών, καθώς και σε εξατομικευμένες εφαρμογές όπως η ηλεκτρονική υγεία (e-health) και η ηλεκτρονική εκπαίδευση (e-learning). Σε όλες αυτές τις εφαρμογές η αναγνώριση προσώπου υποκαθιστά κλασικούς τρόπους επαλήθευσης της ταυτότητας του χρήστη όπως οι προσωπικοί αριθμοί ή κωδικοί πρόσβασης (PIN, password), οι οποίοι μπορούν εύκολα να υποκλαπούν ή να ξεχαστούν. Η πληθώρα των εφαρμογών της αναγνώρισης προσώπου, πολλές από τις οποίες έχουν σημαντικό κοινωνικό και ατομικό αντίκτυπο, καθώς και οι συνεχείς τεχνολογικές εξελίξεις ιδιαίτερα στην περιοχή της περιβάλλουσας νοημοσύνης, καθιστούν την ανάπτυξη αξιόπιστων συστημάτων αναγνώρισης προσώπου που θα παρουσιάζουν σταθερά υψηλή επίδοση σε οποιοσδήποτε συνθήκες λειτουργίας, μία επιτακτική ανάγκη. [25,26,27, 29,30]

Παρατηρώντας την εκπληκτική ευκολία με την οποία οι άνθρωποι αναγνωρίζουν τα πρόσωπα γύρω τους, η υλοποίηση ενός συστήματος αυτόματης αναγνώρισης προσώπων που στηρίζεται στην καταγραφή και την ανάλυση εικόνων του προσώπου εμφανίζεται ως ένα σχετικά εύκολο εγχείρημα. Ωστόσο, όπως και στην περίπτωση των περισσότερων λειτουργιών της αντίληψης, η ευκολία αυτή είναι παραπλανητική. Επιστημονικές έρευνες έχουν δείξει ότι πάνω από το 50% του φλοιού του εγκεφάλου παρουσιάζει αυξημένη δραστηριότητα κατά τη διάρκεια της επεξεργασίας οπτικών ερεθισμάτων, ενώ φαίνεται ότι ο ανθρώπινος εγκέφαλος αφιερώνει περισσότερους πόρους στην όραση παρά σε φαινομενικά πιο απαιτητικές νοητικές διεργασίες όπως η επίλυση μαθηματικών προβλημάτων. Το γεγονός αυτό είναι ενδεικτικό του βαθμού δυσκολίας που εμπεριέχει η ανάπτυξη ενός αυτόματου συστήματος αναγνώρισης προσώπου.

Η αυτόματη αναγνώριση προσώπου είναι γενικά μία σύνθετη διαδικασία που περιλαμβάνει την ανίχνευση του προσώπου σε μία εικόνα, την εξαγωγή κατάλληλων διακριτικών χαρακτηριστικών και τελικά την αναγνώριση του προσώπου με βάση αυτά τα χαρακτηριστικά. Είναι προφανές ότι η επιτυχία της διαδικασίας αυτής βασίζεται κυρίως στην επιλογή κατάλληλων χαρακτηριστικών και στην εύρεση ενός αποτελεσματικού μέτρου για τη σύγκρισή τους. Στην ιδανική περίπτωση οι τιμές των χαρακτηριστικών θα πρέπει να είναι όμοιες για εικόνες του ίδιου προσώπου και αρκετά διαφορετικές για εικόνες που ανήκουν σε διαφορετικά πρόσωπα. Ουσιαστικά δηλαδή, αναζητείται μία αναπαράσταση του προσώπου που θα ελαχιστοποιεί τις ενδοατομικές παραλλαγές (intrapersonal variations), δηλαδή τις παραλλαγές που εμφανίζονται σε διαφορετικές εικόνες του ίδιου προσώπου, ενώ παράλληλα θα μεγιστοποιεί τις διατομικές παραλλαγές (extrapersonal variations), δηλαδή τις διαφορές που εμφανίζονται μεταξύ εικόνων διαφορετικών προσώπων.

Η ύπαρξη ενδοατομικών παραλλαγών αποτελεί το μεγαλύτερο πρόβλημα της αναγνώρισης προσώπου, καθώς έχει αποδειχθεί ότι οι αλλαγές στην εμφάνιση ενός προσώπου που προκαλούνται από τις εκφράσεις, την πόζα ή το φωτισμό, είναι μεγαλύτερες από τις αντιληπτές διαφορές μεταξύ διαφορετικών προσώπων. Οι κύριοι παράγοντες από τους οποίους επηρεάζεται η εμφάνιση ενός προσώπου σε μία εικόνα θα μπορούσαν να συνοψιστούν στους παρακάτω:

- Οπτική γωνία. Η κάμερα που καταγράφει μεταβάλλεται, καθώς αλλάζει η θέση της κάμερας ή ο άνθρωπος στρέφει το κεφάλι του. Η μεταβολή αυτή οφείλεται αφενός μεν στην προβολική παραμόρφωση, η οποία οδηγεί σε επέκταση κάποιων τμημάτων του προσώπου και συμπύκνωση κάποιων άλλων και αφετέρου στην απόκρυψη περιοχών

του προσώπου από το οπτικό πεδίο της κάμερας. Αν ο αλγόριθμος αναγνώρισης προσώπου έχει εκπαιδευτεί χρησιμοποιώντας εικόνες του προσώπου με συγκεκριμένη πόζα (pose), δηλαδή υπό συγκεκριμένη οπτική γωνία, τότε είναι γενικά δύσκολο να αναγνωρίσουμε το ίδιο πρόσωπο από μία διαφορετική οπτική γωνία. [25,26,27]

- Φωτισμός. Όπως οι μεταβολές της πόζας, έτσι και οι μεταβολές του φωτισμού είναι αναπόφευκτες. Ο περιβαλλοντικός φωτισμός (ambient light) αλλάζει σημαντικά κατά τη διάρκεια της ημέρας, από ημέρα σε ημέρα, και μεταξύ διαφορετικών χώρων και τόπων (π.χ. σε ένα εσωτερικό περιβάλλον και έναν εξωτερικό χώρο). Επιπλέον, λόγω της 3D δομής του προσώπου, πηγές φωτισμού κοντά στο πρόσωπο δημιουργούν έντονες σκιές και προκαλούν μεταβολές των φυσικών χρωμάτων της επιφάνειας του προσώπου, με αποτέλεσμα κάποια χαρακτηριστικά ή περιοχές του προσώπου να τονίζονται περισσότερο ή ακόμη και να αλλοιώνονται, ενώ οι περιοχές που βρίσκονται στη σκιά να μην είναι ορατές.
- Εκφράσεις. Το πρόσωπο είναι μία 3D επιφάνεια που περιέχει και παραμορφώσιμες περιοχές. Οι εκφράσεις του προσώπου που οφείλονται στα συναισθήματα και η ομιλία παραμορφώνουν ένα μεγάλο τμήμα του προσώπου και αλλοιώνουν δραστικά την εμφάνιση του.
- Απόκρυψη (occlusion). Πολλές φορές τμήματα του προσώπου αποκρύπτονται από άλλα αντικείμενα της σκηνής, από γυαλιά ηλίου, από κασκόλ, σκούφους, καπέλα και άλλα παρόμοια αξεσουάρ.
- Χρόνος. Καθώς ο χρόνος περνάει, τα πρόσωπα αλλάζουν. Εκτός από τις αλλαγές που οφείλονται στη φυσιολογική γήρανση του ανθρώπου, η εμφάνιση του προσώπου μεταβάλλεται επίσης από παράγοντες όπως το χτένισμα, τα γένια, η χρήση προϊόντων καλλωπισμού και περιποίησης προσώπου που μεταβάλλουν π.χ. την υφή του δέρματος, το φυσικό μαύρισμα που προκαλείται από την έκθεση στον ήλιο, αλλά και οι αισθητικές επεμβάσεις. [25,26,27]

Η εικόνα 5.3 δείχνει ένα χαρακτηριστικό παράδειγμα ενδοατομικών παραλλαγών. Προφανώς μία αξιόπιστη μέθοδος αναγνώρισης προσώπου θα πρέπει να μην επηρεάζεται από μεταβολές της εμφάνισης των προσώπων, που οφείλονται σε παράγοντες όπως αυτοί που αναφέρθηκαν προηγουμένως.

Η μεγάλη πλειοψηφία των μεθόδων που έχουν προταθεί στη βιβλιογραφία, αντιμετωπίζουν το πρόβλημα της αναγνώρισης προσώπων ως ένα πρόβλημα αναγνώρισης τρισδιάστατων αντικειμένων από δισδιάστατες (2D) εικόνες έντασης της φωτεινότητας (intensity images). Παρόλο που η αναγνώριση προσώπου από 2D εικόνες έχει φτάσει σε μία εποχή ωριμότητας παρουσιάζοντας υψηλές επιδόσεις σε ελεγχόμενες συνθήκες, οι 2D μέθοδοι αναγνώρισης προσώπου εξακολουθούν να επηρεάζονται σημαντικά από παράγοντες όπως ο φωτισμός, οι στροφές του κεφαλιού και οι εκφράσεις του προσώπου, όπως έδειξαν πρόσφατες δημόσιες αξιολογήσεις εμπορικών συστημάτων και αλγορίθμων αναγνώρισης προσώπου. Συγκεκριμένα αποδείχθηκε ότι ενώ σε εσωτερικό χώρο με ελεγχόμενες συνθήκες φωτισμού επιτυγχάνεται ρυθμός σωστής αναγνώρισης ίσος με 90%, στην περίπτωση εικόνων που έχουν καταγραφεί σε εξωτερικό χώρο, όπου ο φωτισμός είναι ανεξέλεγκτος, το ποσοστό επιτυχούς αναγνώρισης είναι μόλις 50%. Οι στροφές του κεφαλιού αποδεικνύεται ότι δημιουργούν ακόμη μεγαλύτερο πρόβλημα. Ενώ ο ρυθμός αναγνώρισης πρόσθιων όψεων (frontal views) των προσώπων είναι 90%, ο ρυθμός αυτός μειώνεται δραματικά στο 26% στην περίπτωση εικόνων που απεικονίζουν διαφορετικές πόζες. Οι επιδόσεις των συστημάτων αναγνώρισης προσώπου επηρεάζονται επίσης από τις εκφράσεις, από την ηλικία και το φύλο των χρηστών, καθώς και από το χρονικό διάστημα που μεσολαβεί μεταξύ της λήψης δύο εικόνων του ίδιου προσώπου, οι οποίες συγκρίνονται μεταξύ τους. [25,26,27]

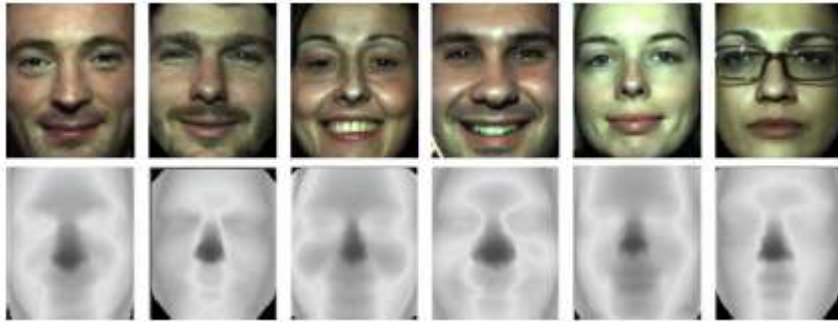


**Εικόνα 5.3: Παραδείγματα ενδοατομικών παραλλαγών (intrapersonal variations), δηλαδή παραλλαγών που εμφανίζονται σε διαφορετικές εικόνες του ίδιου προσώπου. Η εμφάνιση ενός προσώπου μπορεί να μεταβληθεί σημαντικά λόγω παραγόντων όπως η πόζα, ο φωτισμός, οι εκφράσεις, το μακιγιάζ, το χτένισμα, η χρήση γυαλιών, καπέλων, κτλ, αλλά και λόγω της φυσιολογικής γήρανσης του ανθρώπου.**

Οι ερευνητικές προσπάθειες στρέφονται προς την ανάπτυξη τεχνικών που θα παρουσιάζουν σταθερότητα σε μεταβολές των προηγούμενων παραγόντων καθώς η χρήση συστημάτων αναγνώρισης προσώπου επεκτείνεται σε περιβάλλοντα χώρο με μη ελεγχόμενες συνθήκες (π.χ. εφαρμογές επίβλεψης δημόσιων χώρων). Τρεις είναι οι κύριες κατευθύνσεις που ακολουθούνται:

- α) ανάπτυξη καινοτόμων τεχνικών κατάταξης και εκτίμησης της ομοιότητας των προσώπων (classification algorithms),
- β) αντιστάθμιση των παραλλαγών της εμφάνισης του προσώπου (compensation of variations), και
- γ) βελτίωση των υπαρχόντων συστημάτων αναγνώρισης προσώπου με την ενσωμάτωση επιπλέον πληροφοριών που δεν παρουσιάζουν ευαισθησία σε αυτές τις παραλλαγές.

Τα τελευταία χρόνια, στα πλαίσια της τρίτης κατεύθυνσης, προτάθηκαν νέες τεχνικές που στηρίζονται στην ανάλυση εικόνων που αναπαριστούν την τρισδιάστατη δομή του προσώπου, οι οποίες αναφέρονται γενικά με τον όρο τρισδιάστατες (3Δ) εικόνες του προσώπου. Οι 3Δ εικόνες του προσώπου είναι στην ουσία δισδιάστατες εικόνες όπου οι τιμές των εικονοστοιχείων εκφράζουν την απόσταση του αντίστοιχου σημείου του προσώπου από το οπτικό επίπεδο της κάμερας. Οι εικόνες αυτές ονομάζονται επίσης εικόνες ή χάρτες βάθους (depth/range images). Στην εικόνα 5.4 παρουσιάζονται μερικά χαρακτηριστικά παραδείγματα 3Δ εικόνων. Επίσης αρκετές τεχνικές στηρίζονται στη χρήση 3Δ μοντέλων του προσώπου, τα οποία αποτελούνται από κατάλληλα διατεταγμένα πολύγωνα. [25,26,27]



**Εικόνα 5.4 :** Παραδείγματα έγχρωμων εικόνων του προσώπου και των αντίστοιχων χαρτών βάθους. Στους χάρτες βάθους οι πιο σκούρες αποχρώσεις του γκρι αντιστοιχούν σε σημεία του προσώπου που απέχουν μικρότερη απόσταση από το επίπεδο της κάμερας, ενώ οι πιο ανοιχτές αποχρώσεις αντιστοιχούν σε σημεία που απέχουν μεγαλύτερη απόσταση από την κάμερα.

Η χρήση 3D εικόνων για την αναγνώριση του προσώπου παρουσιάζει μία σειρά από πλεονεκτήματα. Καταρχήν η 3D γεωμετρία του προσώπου είναι αρκετά χαρακτηριστική για κάθε άνθρωπο και μπορεί να χρησιμοποιηθεί ως στοιχείο διάκρισης. Επιπλέον, δεν επηρεάζεται από μεταβολές του φωτισμού ή από τη χρήση καλλυντικών, σε αντίθεση με τις αντίστοιχες δισδιάστατες απεικονίσεις. Τέλος, η χρήση πληροφορίας για την 3D δομή του προσώπου μπορεί να βοηθήσει σημαντικά τις διαδικασίες εκτίμησης και αντιστάθμισης της πόζας του προσώπου (pose estimation and compensation), επιλύοντας ουσιαστικά το πρόβλημα των στροφών του κεφαλιού.

Τα τελευταία χρόνια έχουν επίσης προταθεί τεχνικές που συνδυάζουν τη χρήση 2D και 3D εικόνων του προσώπου, ακολουθώντας μία γενικότερη τάση που έχει εκδηλωθεί στην περιοχή της βιομετρίας, η οποία προωθεί το συνδυασμό διαφορετικών βιομετρικών χαρακτηριστικών (π.χ. πρόσωπο και φωνή, πρόσωπο και δακτυλικά αποτυπώματα, κτλ) με στόχο την ανάπτυξη βιομετρικών συστημάτων υψηλής ακρίβειας. Αυτές οι τεχνικές ονομάζονται πολυτροπικές τεχνικές (multimodal techniques) και αποδεικνύεται πως είναι πιο αξιόπιστες από τη χρήση ενός μόνο τύπου πληροφορίας, π.χ. 2D ή 3D εικόνων του προσώπου στην περίπτωση μας.

Η πρώτη εργασία στο πεδίο της 3D αναγνώρισης προσώπων παρουσιάστηκε πριν από το 1990, ο μεγαλύτερος όγκος της έρευνας πραγματοποιήθηκε μετά το 2000. [25,26,27]

Γενικά, οι 3D αλγόριθμοι αναγνώρισης προσώπων μπορούν να χωριστούν σε δυο μεγάλες κατηγορίες, σε αυτούς που θεωρούν το πρόσωπο ως μια σταθερή επιφάνεια (rigid surface) και σε αυτούς που λαμβάνουν υπ' όψιν τη παραμόρφωση που προκαλούν οι εκφράσεις. Οι αλγόριθμοι της πρώτης κατηγορίας μπορούν παραπέρα να ομαδοποιηθούν σε αυτούς που βασίζονται στην καμπυλότητα της επιφάνειας, σε αυτούς που στηρίζονται σε υποχώρους (subspace) για την αναπαράσταση των προσώπων, σε αυτούς που βασίζονται στην ευθυγράμμιση των επιφανειών μέσω του αλγορίθμου Iterative Closest Point-ICP και σε αυτούς που χρησιμοποιούν τοπικούς περιγραφείς της επιφάνειας. Από την άλλη πλευρά, οι αλγόριθμοι που αντιμετωπίζουν την παραμόρφωση που προκαλούν οι εκφράσεις ακολουθούν δυο προσεγγίσεις: είτε εντοπίζουν τα παραμορφώσιμα μέρη της επιφάνειας και τα αγνοούν κατά τη σύγκριση των επιφανειών είτε χρησιμοποιούν αναπαραστάσεις και περιγραφείς που δεν εξαρτώνται από τις εκφράσεις.

### 5.3 Τεχνικές αναγνώρισης προσώπων

Για το πρόβλημα της αναγνώρισης προσώπων, έχουν προταθεί πολλές τεχνικές τα τελευταία χρόνια. Εκ των οποίων οι τρεις δημοφιλέστερες είναι :

- Αναγνώριση προσώπου με Ανάλυση Πρωτευουσών Συνιστωσών γνωστή και ως η τεχνική των eigenfaces.
- Αναγνώριση προσώπου με Γραμμική Διακριτή Ανάλυση (LDA).
- Αναγνώριση προσώπου με μεθόδους ταιριάσματος Ελαστικών Γράφων (EGM). Στις παρακάτω ενότητες αναλύονται οι προαναφερθέντες τεχνικές.

#### 5.3.1 Η Τεχνική των Eigenfaces

Στο Τεχνολογικό Ινστιτούτο της Μασαχουσέτης (MIT), αναπτύχθηκε η τεχνική των Eigenfaces όπου παρακινήθηκε από μια τεχνική που αναπτύχθηκε από τους Sirovich και Kirby το 1987 για αποτελεσματική αναπαράσταση των εικόνων προσώπων χρησιμοποιώντας την ανάλυση πρωτευουσών συνιστωσών (PCA). Παραλλαγές της τεχνικής των eigenfaces χρησιμοποιούνται συχνά ως βάση άλλων μεθόδων αναγνώρισης προσώπου. Αυτή η τεχνική, όπως υποστηρίζεται, δεν κατέχει οποιαδήποτε ομοιότητα στον τρόπο που οι άνθρωποι αναγνωρίζουν και μετρούν την ομοιότητα μεταξύ των προσώπων. Σημείωση: οι εικόνες προς εκπαίδευση και οι προς εξέταση εικόνες θα πρέπει να έχουν όλες το ίδιο μέγεθος, αλλά και να κανονικοποιηθούν ώστε τα μάτια και το στόμα να ευθυγραμμιστούν.

Έστω λοιπόν μία συλλογή από  $N$  πρόσωπα με  $m \times n$  διαστάσεις το καθένα. Η κάθε εικόνα  $A_i$ , όπου  $i = 1, 2, 3, \dots, N$  μπορεί να αναπαρασταθεί ως ένα διάνυσμα  $\Gamma_i$  διάστασης  $1 \times (mn)$ .

Απο αυτά τα διανύσματα αφαιρούμε το μέσο διάνυσμα:

$$\begin{aligned}\Phi_i &= \Gamma_i - \Psi \\ \Psi &= \frac{1}{N} \sum_{i=1}^M \Gamma_i\end{aligned}\tag{5.1}$$

Επειτα σχηματίζουμε τον πίνακα αυτοσυσχέτισης :

$$C = \frac{1}{N} \sum_{i=1}^M \Phi_n \Phi_n^T\tag{5.2}$$

Επόμενο βήμα είναι ο υπολογισμός των ιδιοδιανυσμάτων του  $C$ . Εδώ όμως υπάρχει ένα πρόβλημα, ο πίνακας  $C$  είναι ένας πολύ μεγάλος πίνακας  $((mn) \times (mn))$  διαστάσεων.

Έστω  $A = [\Phi_1, \Phi_2, \dots, \Phi_N]$  τότε ο  $C$  γράφεται ως  $AAT$

Υπολογίζω τα ιδιοδιανύσματα  $v_i$  του  $A^T A$  (πίνακας  $N \times N$ ) :

$$A^T A v_i = \mu_i v_i \quad (5.3)$$

Μπορώ όμως από τα ιδιοδιανύσματα  $v_i$  να υπολογίσω τα ιδιοδιανύσματα  $u_i$  του πίνακα  $C$  :

$$\begin{aligned} A^T A v_i = \mu_i v_i &\Rightarrow A A^T A v_i = A \mu_i v_i \\ A A^T A v_i = \mu_i A v_i &\Rightarrow C A v_i = \mu_i A v_i \end{aligned}$$

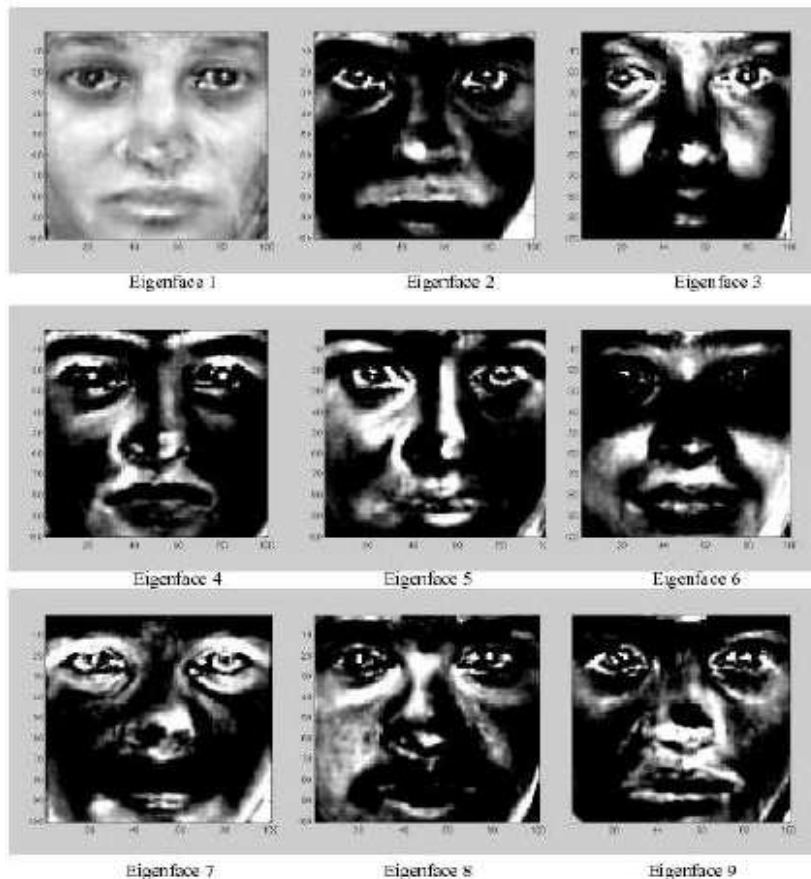
Άρα:

$$C u_i = \mu_i u_i \quad (5.4)$$

όπου,  $u_i = A v_i$ .

Αυτό υποδηλώνει ότι ο  $C$  και ο  $A^T A$  έχουν τις ίδιες ιδιοτιμές και τα ιδιοδιανύσματα τους σχετίζονται κατά αυτή την σχέση :  $u_i = A v_i$

Σημείωση ο πίνακας  $C$  έχει μέχρι  $mn$  ιδιοτιμές και ιδιοδιανύσματα, και ο  $A^T A$  έχει μέχρι  $N$  ιδιοτιμές και ιδιοδιανύσματα. Οι  $N$  ιδιοτιμές του πίνακα  $A^T A$  αντιστοιχούν στις  $N$  σημαντικότερες (μεγαλύτερες) ιδιοτιμές του  $C$ . Σε αυτό το σημείο θα κρατήσουμε μόνο  $K$  ιδιοδιανύσματα, τα οποία ανταποκρίνονται στις  $K$  μεγαλύτερες ιδιοτιμές.



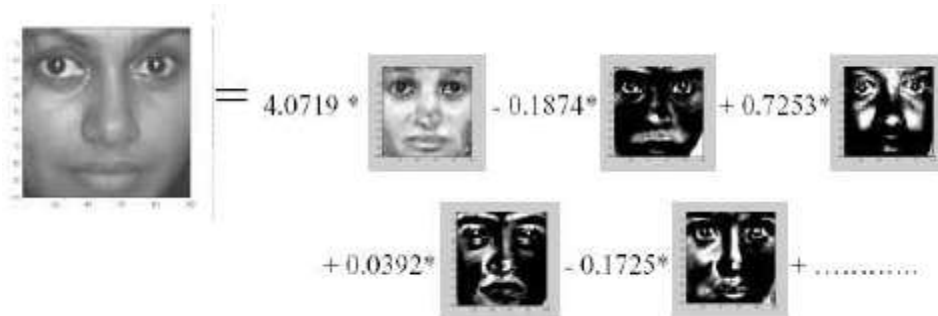
Εικόνα 5.6: Εικόνες ιδιοδιανυσμάτων (eigenfaces)



Κάθε πρόσωπο  $\Phi_i$  μπορεί να αναπαρασταθεί σαν ένας γραμμικός συνδιασμός των  $K$  σημαντικότερων ιδιοδιανυσμάτων:

$$\hat{\Phi}_i = \sum_{j=1}^K u_j^T \Phi_i u_j \quad (5.5)$$

Έτσι κάθε εικόνα εκπαίδευσης  $\Phi_i$  αναπαριστάται σε αυτή την βάση με ένα διάνυσμα :



Εικόνα 5.7: Ανακατασκευή του προσώπου από τις εικόνες ιδιοδιανυσμάτων

$$\Omega_i = [w_1^i, w_2^i, \dots, w_K^i]^T, \quad i = 1, 2, \dots, N \quad (5.6)$$

όπου,  $w_i = u_i^T \Phi_i$

Έπειτα δοθέντος ενός προσώπου εξέτασης  $\Gamma^t$  το οποίο είναι κεντραρισμένο όσον αφορά τα μάτια και το στόμα και ίδιων διαστάσεων με τις εικόνες εκπαίδευσης, ακολουθούμε τα εξής βήματα :

Αφαιρούμε την μέση εικόνα  $\Psi$  :  $\Phi^t = \Gamma^t - \Psi$

Έπειτα την προβάλλουμε στον χώρο των ιδιοδιανυσμάτων :

$$\hat{\Phi}^t = \sum_{j=1}^K u_j^T \Phi^t u_j$$

Αναπαριστάμε το  $\Phi^t$  ως  $\Omega^t = [w_1^t, w_2^t, \dots, w_K^t]$  και βρίσκουμε την ελάχιστη απόσταση από τις εικόνες εκπαίδευσης:

$$e_r = \min_i (\|\Omega^t - \Omega^i\|) \quad (5.7)$$

Αν το  $e_i$  είναι μικρότερο από ένα κατώφλι  $Tr$  και αντιστοιχεί στην απόσταση από το πρόσωπο εκπαίδευσης  $i$  τότε το πρόσωπο προς εξέταση αναγνωρίστηκε ως το πρόσωπο  $i$ .

### 5.3.2 Αναγνώριση προσώπων με LDA

Μία προσέγγιση που βασίζεται στη στατιστική ταξινόμηση των δειγμάτων, είναι η τεχνική LDA (Linear Discriminant Analysis) όπου ανήκει σε μία άγνωστη κλάση, δηλαδή σε κλάσεις που δημιουργήθηκαν από τα δεδομένα εκπαίδευσης.

Η εν λόγω τεχνική έχει σαν στόχο την μεγιστοποίηση της διασποράς μεταξύ διαφορετικών κλάσεων και την ελαχιστοποίηση της διασποράς εντός της κάθε κλάσης. Όταν έχουμε να κάνουμε με δεδομένα που χαρακτηρίζονται με διανύσματα πολλών διαστάσεων, για να μην υπάρξουν προβλήματα, θα πρέπει τα δεδομένα εκπαίδευσης να είναι σχετικά πολλά.

Εστωι κλάσεις των εικόνων εκπαίδευσης  $c_1, c_2, \dots, c_L$  με  $N_1, N_2, \dots, N_L$  στοιχεία η κάθε μία αντίστοιχα. Εστω επίσης οι μέσες εικόνες κάθε κλάσης  $M_1, M_2, \dots, M_L$  και η ολική μέση εικόνα  $M$ . Υπολογίζουμε τους εντός κλάσης και ανάμεσα στις κλάσεις πίνακες διασποράς:

$$F_w = \sum_{i=1}^L p(c_i) A_i = \sum_{i=1}^L E\{[x - M_i][x - M_i]^T\} \quad (5.8)$$

$$F_b = \sum_{i=1}^L p(c_i) [M_i - M][M_i - M]^T \quad (5.9)$$

όπου  $p(c_i)$  η εκ των προτέρων πιθανότητα της κλάσης  $c_i$  και  $A_i$  ο πίνακας συμμεταβλητότητας της κλάσης  $c_i$ . Η μέθοδος LDA δημιουργεί έναν πίνακα  $\Psi$ , τα διανύσματα του οποίου αποτελούν μια βάση του χώρου  $R^n$  και ο οποίος μεγιστοποιεί τον λόγο  $\frac{\Psi^T F_b \Psi}{\Psi^T F_w \Psi}$ .

Για τη μεγιστοποίηση του λόγου αυτού ο πίνακας  $\Psi$  πρέπει να αποτελείται από τα ιδιοδυναύσματα του πίνακα  $F_w^{-1} F_b$ :

$$F_w^{-1} F_b \Psi = \Psi D \quad (5.10)$$

όπου οι πίνακες  $\Psi, D \in R^{n \times n}$  είναι οι πίνακες ιδιοδυναυσμάτων και ιδιοτιμών του πίνακα  $F_w^{-1} F_b$ . Ενώ οι πίνακες  $F_w$  και  $F_b$  είναι συμμετρικοί, ο πίνακας δεν είναι  $F_w^{-1} F_b$  πάντα συμμετρικός. Σε κάθε περίπτωση όμως οι πίνακες  $\Psi$  και  $D$  μπορούν να υπολογιστούν ως αποτέλεσμα ταυτόχρονης διαγωνιοποίησης των  $F_w$  και  $F_b$ . Η μέθοδος LDA αποφεύγει ένα από τα μειονεκτήματα της PCA (eigenfaces) και συγκεκριμένα αυτό της ικανότητας διαχωρισμού των κλάσεων. Αυτό επιτυγχάνεται με την εύρεση των αξόνων που μεγιστοποιείται ταυτόχρονα η ολική διασπορά του πληθυσμού αλλά και των επιμέρους κλάσεων όπως αυτές εκφράζονται από το κέντρο τους. Για την ικανοποίηση αυτής της απαίτησης καταλήγουμε σε μη ορθογώνιους άξονες, πράγμα που δεν συμβαίνει με την PCA.



Στην περίπτωση της αναγνώρισης προσώπων η μέθοδος LDA παρουσιάζει δύο σημαντικά προβλήματα :

- Ο μεγάλος αριθμός κλάσεων με λίγες εικόνες σε κάθε κλάση, δημιουργεί πρόβλημα όσον αφορά την εκτίμηση των πινάκων  $A_i$  και κατά συνέπεια του πίνακα  $F_b$ .
- Λόγου του μεγάλου μεγέθους του πίνακα  $F-1F$  η διαδικασία διαγωνιοποίησης του γίνεται ιδιαίτερα δύσκολη.

### 5.3.3 Ο Αλγόριθμος EGM

#### 5.3.3.1 Εισαγωγή

Μία ευρέως γνωστή τεχνική για την αναγνώριση και επαλήθευση προσώπου είναι ο αλγόριθμος ταιριάσματος ελαστικού γράφου (Elastic Graph Matching) ή (EGM). Στον EGM, ο γράφος αναφοράς ενός αντικειμένου δημιουργείται με την τοποθέτηση ενός ορθογώνιου αραιού ελαστικού γράφου στην εικόνα αντικειμένου και τον υπολογισμό μιας απόκρισης κυματιδίων Gabor σε κάθε κόμβο του γράφου. Η διαδικασία ταιριάσματος των γράφων υλοποιείται από μια στοχαστική βελτιστοποίηση μιας συνάρτησης κόστους που λαμβάνει υπόψη και τις ομοιότητες των jets και τις παραμορφώσεις του πλέγματος. Από την εφευρέσή του, ο EGM έχει μία πολύ σημαντική προσφορά στην έρευνα για την αναγνώριση και επαλήθευση προσώπου.

#### 5.3.3.2 Ο κλασικός αλγόριθμος EGM

Στο πρώτο βήμα του αλγορίθμου EGM ένας αραιός γράφος κατάλληλος για την αντιπροσώπευση του προσώπου επιλέγεται [32]. Η περιοχή του προσώπου εικόνας αναλύεται και ένα σύνολο τοπικών περιγραφών εξάγεται σε κάθε κόμβο του γράφου. Η ανάλυση εκτελείται συνήθως με την οικοδόμηση μιας πυραμίδας πληροφοριών χρησιμοποιώντας τεχνικές scale-space. Στον απλό EGM, ένα δισδιάστατο φίλτρο Gabor χρησιμοποιείται για την ανάλυση εικόνας . Η παραγωγή των μορφολογικών διαδικασιών διαστολή-διάβρωσης ή η μορφολογική αποσύνθεση σημάτων σε διάφορες κλίμακες είναι μη γραμμικές παραλλαγές των φίλτρων Gabor για πολυδιάστατη ανάλυση, και τα δύο έχουν χρησιμοποιηθεί επιτυχώς για την ανάλυση της εικόνας προσώπου. Σε κάθε κόμβο  $l$  του γράφου που έχει  $x_l$  συντεταγμένες στην εικόνα, ένα jet (διάνυσμα χαρακτηριστικών)  $j(x)$  διαμορφώνεται :

$$J(x^l) = [f_1(x^l), \dots, f_m(x^l)]^T, \quad (5.11)$$

Όπου το  $f_i(x_l)$  είναι το αποτέλεσμα ενός τοπικού τελεστή που εφαρμόζεται στην εικόνα  $f$  στην  $i$  διάσταση και  $M$  είναι η διάσταση του jet. Το επόμενο βήμα του EGM είναι να μεταφέρει και να παραμορφώσει τον γράφο αναφοράς στην εικόνα δοκιμής ούτως ώστε να βρει την απόκριση του γράφου αναφοράς στην εικόνα δοκιμής . Αυτό επιτυγχάνεται ελαχιστοποιώντας μία συνάρτηση κόστους που ανταποκρίνεται τις στις ομοιότητες των jets των κόμβων και στις σχέσεις γειτονιάς κόμβων. Έστω  $t$  και  $r$  μία εικόνα δοκιμής και αναφοράς αντίστοιχα (ή γράφος). Η  $L_2$  νόρμα ανάμεσα στα διανύσματα χαρακτηριστικών στον  $i$  κόμβο του γράφου αναφοράς και δοκιμής χρησιμοποιείται σαν ένα μέγεθος

ομοιότητας μεταξύ των jets .

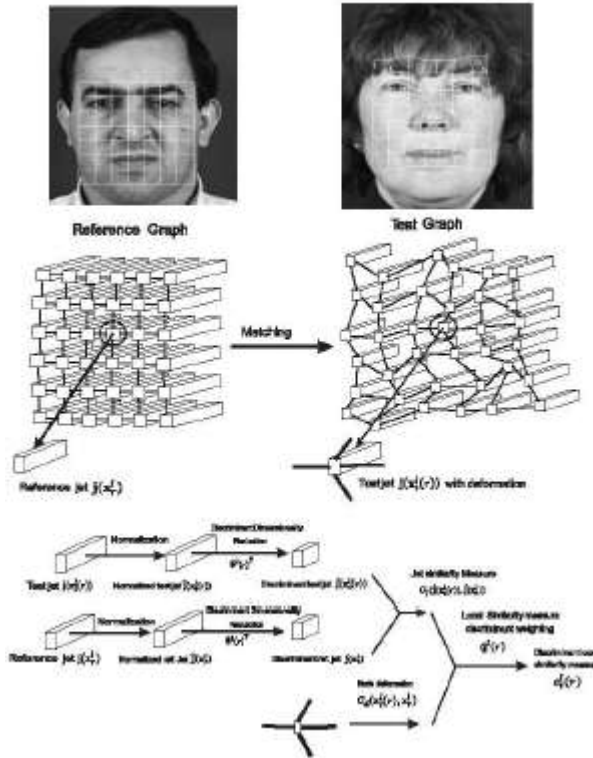
$$C_f(j(x_t^l), j(x_r^l)) = \|j(x_r^l) - j(x_t^l)\| \quad (5.12)$$

Έστω  $V$  το σύνολο των ακμών ενός γράφου ορισμένης εικόνας προσώπου. Οι γράφοι που εξετάζονται σε αυτήν την περίπτωση είναι ορθογώνιοι γράφοι όποιοι είναι τοπολογικά ισοδύναμοι με ένα ορθογώνιο υποσύνολο του  $Z$  (το  $Z$  είναι το σύνολο ακέραιων αριθμών). Κατά συνέπεια, όλοι οι κόμβοι, εκτός από τους ακριανούς κόμβους, έχουν ακριβώς τέσσερις συνδεδεμένους κόμβους. Η εικόνα 5.8 δείχνει ένα τυπικό ορθογώνιο γράφο αναφοράς που χρησιμοποιείται σε αυτήν την περίπτωση. Έστω  $H(l)$  η γειτονιά του κόμβου  $l$ . Για να ποσοτικοποιήσουμε τις σχέσεις γειτνίασης χρησιμοποιώντας μία μετρική, η τοπική παραμόρφωση είναι :

$$C_d(x_t^l, x_r^l) = \sum_{\xi \in H(l)} \|(x_t^l - x_r^l) - (x_t^\xi - x_r^\xi)\| \quad (5.13)$$

Σκοπός είναι να βρεθεί ένα σύνολο από ακμές  $\{x_t^l(r); l \in V\}$  στην εικόνα δοκιμής τέτοιο ώστε να ελαχιστοποιεί την συνάρτηση κόστους :

$$C(x_t^l) = \sum_{l \in V} \{C_f(j(x_t^l), j(x_r^l)) + C_d(x_t^l, x_r^l)\} \quad (5.14)$$



Εικόνα 5.8 : Τα δήματα του EGM

Το jet του κόμβου  $l$  που έχει παραχθεί μετά από την διαδικασία ταιριάσματος με διαδικασία του γράφου του προσώπου αναφοράς  $\rho$  στην εικόνα του προσώπου δοκιμής  $\tau$  προσδιορίζεται ως  $j(x_l(\tau))$ . Αυτή η αναφορά γίνεται γιατί διαφορετικοί γράφοι αναφοράς  $\rho$  μας δίνουν άλλα τεστ θετς  $j(x_l(\tau))$ . Κατά συνέπεια, το θετ του  $l$  κόμβου του γράφου  $\tau$  είναι μια συνάρτηση του γράφου αναφοράς  $\rho$ . Ο συμβολισμός  $j(x_l)$  χρησιμοποιείται μόνο όταν ο  $l$  κόμβος είναι σε μια προεπιλεγμένη θέση της εικόνας προσώπου. Η βελτιστοποίηση της σχέσης 5.14 έχει υλοποιηθεί ως προσομοιωμένη ανόπτηση με ποινές που επιβάλλονται από τις παραμορφώσεις των γράφων [31]. Συγκεκριμένα, η σχέση 5.14 μπορεί να πάρει την μορφή :

$$D_t(r) = \sum_{l \in V} \{C_f(j(x_t^l), j(x_r^l))\} \quad (5.15)$$

υπό την προϋπόθεση :

$$x_t^l = x_r^l + s + \delta_l, \|\delta_l\| \leq \delta_{max} \quad (5.16)$$

όπου το  $s$  είναι η ολική μετατόπιση του γράφου και το  $\pm 1$  υποδηλώνει μία τοπική διαταραχή των κόμβων του γράφου. Η επιλογή των  $l$  στην σχέση 5.14 και του  $\pm max$  στην 5.14 ελέγχουν την πλαστικότητα του γράφου. Προφανώς και οι δύο σχέσεις ορίζουν ένα μέτρο ομοιότητας μεταξύ δύο προσώπων. Μετά την διαδικασία ταιριάσματος η απόσταση

$Dt(r)$  χρησιμοποιείται σαν ποσοτικό μέτρο για την ομοιότητα δύο προσώπων .

Εξετάζοντας προσεκτικά τη διαδικασία EGM από την σκοπιά της αναγνώρισης προτύπων, οι ακόλουθες ερωτήσεις προκύπτουν : Όλες οι διαστάσεις του jet κατέχουν χρήσιμες πληροφορίες ; Η παραμόρφωση κόμβων κατέχουν αξιοποιήσιμη πληροφορία ; Είναι όλοι οι κόμβοι του γράφου εξίσου σημαντική για την επαλήθευση της ταυτότητας μιας εικόνας προσώπου ;

Προκειμένου να απαντήσουμε σε όλες αυτές τις ερωτήσεις, ένα γενικό πλαίσιο που ενισχύει την απόδοση του EGM με έναν εποπτευμένο τρόπο προτείνεται [32].

Λεπτομερέστερα, οι διακρίνουσες τεχνικές χρησιμοποιούνται για την επιλογή των πιο διακρινόντων χαρακτηριστικών τις κάθε κλάσης προσώπου. Το μέτρο ομοιότητας των jet συνδυάζεται κατά διακρίνοντα τρόπο με την παραμόρφωση των κόμβων προκειμένου να διαμορφωθεί ένα τοπικό διακρίνων μέτρο ομοιότητας μεταξύ των κόμβων. Η χρήση της παραμόρφωσης κατά έναν διακρίνων τρόπο μπορεί να εξηγηθεί διαισθητικά ως εξής. Ο γράφος του προσώπου έχει τους κόμβους που μπορούν να αντιστοιχούν στα ορόσημα (τα ορόσημα αντιστοιχούν στα σημεία του προσώπου) των οποίων η παραμόρφωση μπορεί να θεωρηθεί άκαμπτη είτε ελαστική για ένα ιδιαίτερο πρόσωπο. Για παράδειγμα, οι κόμβοι που αντιστοιχούν στα σημάδια προσώπου ενός ατόμου που είναι σε κάποια άκαμπτη περιοχή όπως το μέτωπο ή τη μύτη δεν μπορούν να κινηθούν εύκολα, ενώ μερικοί κόμβοι που αντιστοιχούν στα σημεία μέσα τα χείλια μπορούν να κινηθούν πιο ελεύθερα.

Εάν είχαμε εκ των προτέρων διαθέσιμη την πληροφορία για την ελαστικότητα ή ακαμψία κάθε περιοχής του προσώπου, θα μπορούσαμε να την είχαμε ενσωματώσει στην διαδικασία ταιριάσματος του πλέγματος. Εντούτοις, αυτή η πληροφορία είναι συγκεκριμένη για κάθε πρόσωπο και έτσι πρέπει να ανακτηθεί χρησιμοποιώντας μια διαδικασία εκπαίδευσης και να ληφθεί υπ' όψη όταν σχηματίζουμε το μέτρο τοπικής ομοιότητας μεταξύ των κόμβων. Το τελευταίο βήμα το DEGM είναι να μάθει ποιοί κόμβοι περιέχουν σημαντική διακριτή πληροφορία έτσι ώστε να χρησιμοποιήσει αντίστοιχα δάρη όταν υπολογίζει την ομοιότητα μεταξύ των προσώπων. Αυτό το δήμα προκύπτει από το γεγονός ότι συγκεκριμένα χαρακτηριστικά του προσώπου είναι πιο διακριτά από κάποια άλλα.

## Κεφάλαιο 6

### Υλοποίηση αναγνώρισης προσώπου με Eigenfaces

#### 6.1 Πειραματικό Μέρος

Το Eigenface αποτελείται από δύο φάσεις: εκμάθηση και αναγνώριση. Στη φάση εκμάθησης, δίνουμε στο eigenface μια ή περισσότερες εικόνες προσώπου για κάθε πρόσωπο που θέλουμε να αναγνωρίσουμε. Αυτές οι εικόνες ονομάζονται εικόνες εκπαίδευσης. Στη φάση αναγνώρισης, όπου δίνουμε στο eigenface μια εικόνα προσώπου, αποκρίνεται με το να μας πει ποια από τις εικόνες εκπαίδευσης είναι πιο “κοντά” στην εικόνα που του δώσαμε.

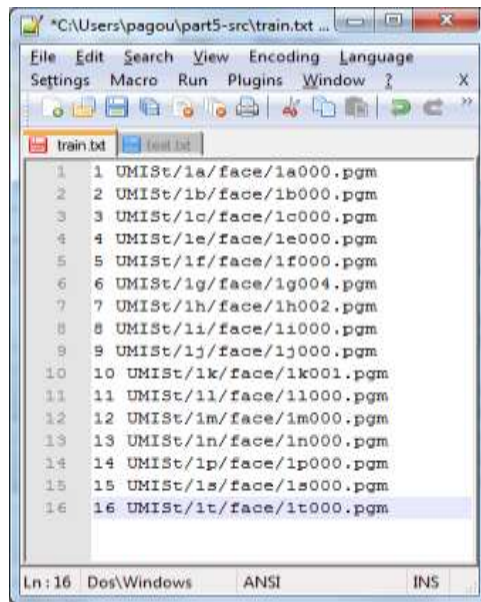
Το Eigenface χρησιμοποιεί τις εικόνες εκπαίδευσης για να “μάθει” το πρότυπο ενός προσώπου. Αυτό το πρότυπο προσώπου δημιουργείται με την εφαρμογή μιας μεθόδου αποκαλούμενης ανάλυση κύριων τμημάτων Principal Components Analysis (PCA) για να μειωθεί η “ διαστατικότητα”( dimensionality) από αυτές τις εικόνες. Το Eigenface καθορίζει τη διαστατικότητα εικόνας ως αριθμό εικονοκυττάρων σε μια εικόνα.

Η χαμηλότερη διαστατικότητα που βρίσκει το eigenface κατά τη διάρκεια της φάσης εκπαίδευσης καλείται υποχώρος (subspace). Στη φάση αναγνώρισης, μειώνει τη διαστατικότητα της εικόνας εισαγωγής προβάλλοντάς την επάνω στον υποχώρο subspace που βρήκε κατά τη διάρκεια της εκπαίδευσης. “ Προβολή επάνω σε ένα υποχώρο” σημαίνει να βρει το πιο στενό σημείο πάνω σε αυτόν. Αφότου έχει προβληθεί η άγνωστη εικόνα προσώπου, το eigenface υπολογίζει την απόσταση μεταξύ της νέας εικόνας και κάθε εικόνας εκπαίδευσης. Η παραγωγή της είναι ένας δείκτης στην πιο στενή εικόνα κατάρτισης. Μπορείτε έπειτα να ανατρέξετε ποιο πρόσωπο που επρόκειτο να δει ποιων eigenface έχει προσδιορίσει.

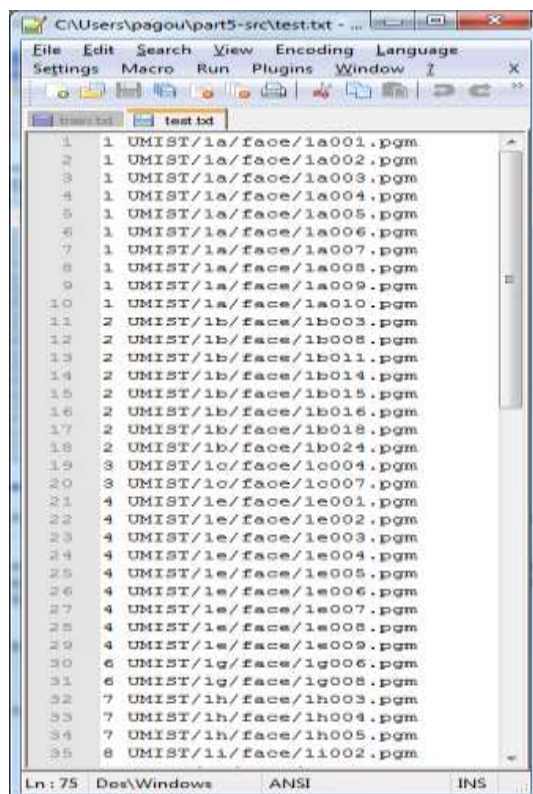
#### 6.2 Προετοιμασία για το Eigenface

Μπορούμε αν θέλουμε να συνδυάσουμε το eigenface με τη μέθοδο αντίχτυσης προσώπου που παρουσιάζεται στο κεφάλαιο 3. Για να απλοποιήσουμε λίγο τον κώδικα παραδείγματος μπορούμε να χρησιμοποιήσουμε το σύνολο εικόνων κατάρτισης και δοκιμής που χρησιμοποιήσαμε και για την αντίχτυση. Αυτές οι εικόνες πρέπει όλες να έχουν ακριβώς το ίδιο μέγεθος.

Θα χρειαστούμε επίσης δύο αρχεία εισαγωγής: train.txt και test.txt. Η εικ. 6.1 παρουσιάζει ένα παράδειγμα αυτών των αρχείων εισαγωγής. Και τα δύο αρχεία χρησιμοποιούν το ίδιο σχήμα: αριθμός προσώπων, whitespace, πορεία στο αρχείο εικόνας(path).



```
1 1 UMIST/1a/face/1a000.pgm
2 2 UMIST/1b/face/1b000.pgm
3 3 UMIST/1c/face/1c000.pgm
4 4 UMIST/1e/face/1e000.pgm
5 5 UMIST/1f/face/1f000.pgm
6 6 UMIST/1g/face/1g004.pgm
7 7 UMIST/1h/face/1h002.pgm
8 8 UMIST/1i/face/1i000.pgm
9 9 UMIST/1j/face/1j000.pgm
10 10 UMIST/1k/face/1k001.pgm
11 11 UMIST/1l/face/1l000.pgm
12 12 UMIST/1m/face/1m000.pgm
13 13 UMIST/1n/face/1n000.pgm
14 14 UMIST/1p/face/1p000.pgm
15 15 UMIST/1s/face/1s000.pgm
16 16 UMIST/1t/face/1t000.pgm
```



```
1 1 UMIST/1a/face/1a001.pgm
2 1 UMIST/1a/face/1a002.pgm
3 1 UMIST/1a/face/1a003.pgm
4 1 UMIST/1a/face/1a004.pgm
5 1 UMIST/1a/face/1a005.pgm
6 1 UMIST/1a/face/1a006.pgm
7 1 UMIST/1a/face/1a007.pgm
8 1 UMIST/1a/face/1a008.pgm
9 1 UMIST/1a/face/1a009.pgm
10 1 UMIST/1a/face/1a010.pgm
11 2 UMIST/1b/face/1b003.pgm
12 2 UMIST/1b/face/1b008.pgm
13 2 UMIST/1b/face/1b011.pgm
14 2 UMIST/1b/face/1b014.pgm
15 2 UMIST/1b/face/1b015.pgm
16 2 UMIST/1b/face/1b016.pgm
17 2 UMIST/1b/face/1b018.pgm
18 2 UMIST/1b/face/1b024.pgm
19 3 UMIST/1c/face/1c004.pgm
20 3 UMIST/1c/face/1c007.pgm
21 4 UMIST/1e/face/1e001.pgm
22 4 UMIST/1e/face/1e002.pgm
23 4 UMIST/1e/face/1e003.pgm
24 4 UMIST/1e/face/1e004.pgm
25 4 UMIST/1e/face/1e005.pgm
26 4 UMIST/1e/face/1e006.pgm
27 4 UMIST/1e/face/1e007.pgm
28 4 UMIST/1e/face/1e008.pgm
29 4 UMIST/1e/face/1e009.pgm
30 6 UMIST/1g/face/1g006.pgm
31 6 UMIST/1g/face/1g008.pgm
32 7 UMIST/1h/face/1h003.pgm
33 7 UMIST/1h/face/1h004.pgm
34 7 UMIST/1h/face/1h005.pgm
35 8 UMIST/1i/face/1i002.pgm
```

Εικόνα 6.1. Τα δύο αρχεία εισαγωγής για το πρόγραμμα Eigenface: train.txt και test.txt.

Μπορεί να είχατε παρατηρήσει ότι οι πρώτες τρεις εικόνες δοκιμής είναι οι ίδιες με τις εικόνες εκπαίδευσης. Αυτό είναι χρήσιμο επειδή το eigenface πρέπει πάντα να δώσει τη σωστή απάντηση όταν του ζητάμε να αναγνωρίσει μια από τις εικόνες εκπαίδευσης. Εάν δε το κάνει τότε ξέρουμε ότι θα έχουμε διορθώσεις να κάνουμε.

Για να τρέξετε τη φάση εκπαίδευσης αυτού του προγράμματος, εισάγουμε :

- eigenface train

στο command prompt και αφού έχουμε μπει στο φάκελο που έχουμε αποθηκεύσει τον κώδικα.

Για να τρέξετε τη φάση αναγνώρισης, εισάγουμε

- eigenface test

### 6.3 Επεξήγηση κώδικα αναγνώρισης

Η εικ. 6.2 παρουσιάζει τη λίστα πηγών για το Eigenface, ένα πρόγραμμα για να μαθευτούν και να αναγνωριστούν τα πρόσωπα με μεθόδους του eigenface σε OpenCV. Στη γραμμή 4, περιλαμβάνει το αρχείο “cvaux.h”. Μέχρι τώρα, για την ανίχνευση χρειαζόμασταν μόνο τα “cv.h” και “highgui.h”. Εδώ θα κάνουμε χρήση δύο λειτουργιών για την αναγνώριση προσώπου, την cvCalcEigenObjects () και την cvEigenDecomposite (), οι οποίες καθορίζονται από το “cvaux.h”.

```
main()
1 #include <stdio.h>
2 #include <string.h>
3 #include "cv.h"
4 #include "cvaux.h"
5 #include "highgui.h"
6
7 /// Global variables
8 int nTrainFaces = 0; // number of training images
9 int nEigens = 0; // number of eigenvalues
10 IplImage ** faceImgArr = 0; // array of face images
11 CvMat * personNumTruthMat = 0; // array of person numbers
12 IplImage * pAvgTrainImg = 0; // the average image
13 IplImage ** eigenVectArr = 0; // eigenvectors
14 CvMat * eigenValMat = 0; // eigenvalues
15 CvMat * projectedTrainFaceMat = 0; // projected training faces
16
17 /// Function prototypes
18 void learn();
19 void recognize();
20 void doPCA();
21 void storeTrainingData();
22 int loadTrainingData(CvMat ** pTrainPersonNumMat);
23 int findNearestNeighbor(float * projectedTestFace);
24 int loadFaceImgArray(char * filename);
25 void printUsage();
26
27 void main( int argc, char** argv )
28 {
29 // validate that an input was specified
30 if( argc != 2 )
31 {
32 printUsage();
33 return;
34 }
35
36 if( !strcmp(argv[1], "train") ) learn();
37 else if( !strcmp(argv[1], "test") ) recognize();
38 else
39 {
40 printf("Unknown command: %s\n", argv[1]);
41 printUsage();
42 }
43 }
```

Εικόνα 6.2. Η συνάρτηση main() του προγράμματος Eigenface

Οι αρχικές μεταβλητές καθορίζονται στις γραμμές 8-15. Ένας από τους τύπους δεδομένων (datatype) που χρησιμοποιούμε για πρώτη φορά εδώ, είναι το CvMat. Αυτό είναι ένας τύπος πίνακα του OpenCV.

Εάν πρέπει μόνο να φυλάξουμε κάποια στοιχεία προσωρινά στα πλαίσια ενός προγράμματός, ένας συνηθισμένος πίνακας της C είναι η ευκολότερη λύση. Αλλά το CvMat datatype μπορεί να είναι αποτελεσματικότερο όταν θέλουμε να εκμεταλλευτείτε τις λειτουργίες του OpenCV για την εργασία με τα στοιχεία πινάκων. Αυτούς δηλαδή που θα χρησιμοποιήσουμε στον συγκεκριμένο κώδικα. Με αυτούς, μπορούμε να αποθηκεύσουμε τα στοιχεία ενός πίνακα με μια ενιαία γραμμή κώδικα και η ανάγνωση τους στο πρόγραμμα αργότερα είναι εξίσου εύκολη.

Όπως βλέπουμε η συνάρτηση main() δεν κάνει κάτι περίπλοκο. Απλά διαβάζει κάποιες μεταβλητές και στην συνέχεια καλεί τις συναρτήσεις learn() και recognize().

Η εικόνα 6.3 δείχνει την βοηθητική συνάρτηση printUsage()

```
printUsage ()  
1 void printUsage()  
2 {  
3     printf("Usage: eigenface <command>\n",  
4           " Valid commands are\n",  
5           "   train\n",  
6           "   test\n");  
7 }
```

**Εικόνα 6.3 : Η βοηθητική συνάρτηση printUsage()**

### Η συνάρτηση learn()

Το σχήμα 4 παρουσιάζει την συνάρτηση learn() που υλοποιεί την εκμάθηση σε 4 βήματα:

1. Φορτώνει τα στοιχεία εκπαίδευσης, (γραμμή 6).
2. Κάνει PCA πάνω τους για να βρει έναν υποχώρο(subspace), (γραμμή 16).
3. Προβολή των εικόνων εκπαίδευσης επάνω στον υποχώρο, (γραμμές 20-29).
4. Αποθήκευση όλων των πληροφοριών εκπαίδευσης, (γραμμή 32).
  - α. Eigenvalues
  - β. Eigenvectors
  - γ. The average training face image
  - δ. Projected faces
  - ε. Person ID numbers

Στις παρακάτω 4 υποενότητες θα αναλύσουμε αυτά τα βήματα.



```

learn()
1 void learn()
2 {
3     int i;
4
5     // load training data
6     nTrainFaces = loadFaceImgArray("train.txt");
7     if( nTrainFaces < 2 )
8     {
9         fprintf(stderr,
10            "Need 2 or more training faces\n"
11            "Input file contains only %d\n", nTrainFaces);
12        return;
13    }
14
15    // do PCA on the training faces
16    doPCA();
17
18    // project the training images onto the PCA subspace
19    projectedTrainFaceMat = cvCreateMat(nTrainFaces, nEigens, CV_32FC1);
20    for(i=0; i<nTrainFaces; i++)
21    {
22        cvEigenDecomposite(
23            faceImgArr[i],
24            nEigens,
25            eigenVectArr,
26            0, 0,
27            pAvgTrainImg,
28            projectedTrainFaceMat->data.fl + i*nEigens);
29    }
30
31    // store the recognition data as an xml file
32    storeTrainingData();
33 }

```

Εικόνα 6.4 : Η συνάρτηση learn(). Φάση εκμάθησης του Eigenface

### **1<sup>ο</sup> Βήμα : Φόρτωση εικόνων προσώπου για την εκπαίδευση ή τη δοκιμή**

Η συνάρτηση loadFaceImgArray () φορτώνει τις εικόνες προσώπων και και την ταυτότητα των ατόμων για τις φάσεις εκμάθησης και αναγνώρισης.

Οι εικόνες προσώπου, αποθηκεύονται στη μεταβλητή faceImgArr.

Η συνάρτηση loadFaceImgArray () επιστρέφει τον αριθμό των εικόνων προσώπου που φορτώνονται.

Οι αριθμοί ταυτότητας προσώπων αποθηκεύονται σε μια μεταβλητή τύπου CvMat, (personNumTruthMat). Αλήθεια(Truth) εδώ αναφέρεται στον όρο της τεχνητής νοημοσύνης (AI), " ground truth." Σημαίνει ότι οι τιμές σε αυτήν την μεταβλητή είναι οι αληθινές (correct) τιμές για κάθε εικόνα προσώπου. Κατά τη διάρκεια της φάσης εκμάθησης, αυτός είναι ο μόνος τύπος αριθμών ταυτότητας προσώπων που έχουμε. Αλλά κατά τη διάρκεια της φάσης αναγνώρισης, το πρόγραμμα θα έχει και τις τιμές " ground truth." (που διευκρινίζονται στο αρχείο test.txt) και αυτές που θα παράγει το Eigenface. Η κατοχή και των δύο μας επιτρέπει να αξιολογήσουμε πόσο καλά δουλεύει το Eigenface.

Η συνάρτηση cvCreateMat (), που καλείται στη γραμμή 16, δημιουργεί τη μεταβλητή personNumTruthMat. Αυτή η λειτουργία παίρνει τρεις παραμέτρους: τον αριθμό σειρών, τον αριθμό στηλών, και το datatype για τον πίνακα. Σε ένα τριανταδυάμπιτο λειτουργικό σύστημα, το datatype για τον πίνακα των τιμών INT είναι CV\_32SC1. Το S αντιπροσωπεύει εδώ υπογεγραμμένος "signed" και το C1 δείχνει ότι ο πίνακας έχει ένα κανάλι. (Ενας πίνακας μπορεί να έχει μέχρι τέσσερα κανάλια.)

```

loadFaceImgArray()
1 int loadFaceImgArray(char * filename)
2 {
3     FILE * imgListFile = 0;
4     char imgFilename[512];
5     int iFace, nFaces=0;
6
7     // open the input file
8     imgListFile = fopen(filename, "r");
9
10    // count the number of faces
11    while( fgets(imgFilename, 512, imgListFile) ) ++nFaces;
12    rewind(imgListFile);
13
14    // allocate the face-image array and person number matrix
15    faceImgArr = (IplImage **)cvAlloc( nFaces*sizeof(IplImage *) );
16    personNumTruthMat = cvCreateMat( 1, nFaces, CV_32SC1 );
17
18    // store the face images in an array
19    for(iFace=0; iFace<nFaces; iFace++)
20    {
21        // read person number and name of image file
22        fscanf(imgListFile,
23            "%d %s", personNumTruthMat->data.i+iFace, imgFilename);
24
25        // load the face image
26        faceImgArr[iFace] = cvLoadImage(imgFilename, CV_LOAD_IMAGE_GRAYSCALE);
27    }
28
29    fclose(imgListFile);
30
31    return nFaces;
32 }

```

Εικόνα 6.5. Η συνάρτηση loadFaceImgArray (). Φορτώνει τις εικόνες και τους αριθμούς ταυτότητας προσώπων και για τις φάσεις εκμάθησης και αναγνώρισης.

Το CvMat datatype είναι μια δομή, με τα ακατέργαστα στοιχεία που αποθηκεύονται στα στοιχεία δεδομένων μιας δομής. Το στοιχείο ορίζεται ως μια ένωση (ο καθορισμός δίνεται στην τεκμηρίωση CXCORE), με τα ακέραια στοιχεία (INT) που προσεγγίζονται ως data.i. Οι γραμμές 22-23 παρουσιάζουν το μόνο δρόμο για να προσεγγιστούν οι τιμές CvMat – σαν offsets από την αρχή του απομονωτή στοιχείων(data buffer).

Οι σειρές των πινάκων ευθυγραμμίζονται για να αρχίσουν σε 4 διαστήματα ψηφιολέξεων. Το πλάτος σειρών -μνήμης, στις ψηφιολέξεις, αποθηκεύεται στο CvMat.step. Δεδομένου ότι χρησιμοποιούμε μια ψηφιολέξη 4 datatype σε αυτόν τον πίνακα (και επίσης, δεδομένου ότι έχει μόνο μια σειρά), μπορούμε να αγνοήσουμε την CvMat.step. Αλλά, εάν δημιουργήσουμε ένα πίνακα, παραδείγματος χάριν, τα στοιχεία χαρακτήρων (char), μπορεί να πρέπει να λάβουν το μέγεθος βημάτων υπόψη όταν έχετε πρόσβαση στα στοιχεία στοιχείων.

## 2<sup>ο</sup> Βήμα : Εύρεση του υποχώρου(subspace) PCA

Στην εικ. 6.6 είναι το κομμάτι κώδικα που βρίσκει το PCA subspace. Καλεί την ενσωματωμένη συνάρτηση του OpenCV που κάνει PCA, cvCalcEigenObjects (), στις γραμμές 27-36. Το υπόλοιπο της doPCA () δημιουργεί τις μεταβλητές που θα κρατήσουν τα PCA αποτελέσματα που επιστρέφει η cvCalcEigenObjects ().

Στη γραμμή 8, ο αριθμός eigenvalues τίθεται σε ένα λιγότερο από τον αριθμό εικόνων εκπαίδευσης. Αυτός ορίζεται ως ο μέγιστος αριθμός eigenvalues που μπορούμε να βρούμε.

```

doPCA()
1 void doPCA()
2 {
3     int i;
4     CvTermCriteria calcLimit;
5     CvSize faceImgSize;
6
7     // set the number of eigenvalues to use
8     nEigens = nTrainFaces-1;
9
10    // allocate the eigenvector images
11    faceImgSize.width = faceImgArr[0]->width;
12    faceImgSize.height = faceImgArr[0]->height;
13    eigenVectArr = (IplImage**)cvAlloc(sizeof(IplImage*) * nEigens);
14    for(i=0; i<nEigens; i++)
15        eigenVectArr[i] = cvCreateImage(faceImgSize, IPL_DEPTH_32F, 1);
16
17    // allocate the eigenvalue array
18    eigenValMat = cvCreateMat( 1, nEigens, CV_32FC1 );
19
20    // allocate the averaged image
21    pAvgTrainImg = cvCreateImage(faceImgSize, IPL_DEPTH_32F, 1);
22
23    // set the PCA termination criterion
24    calcLimit = cvTermCriteria( CV_TERMCRIT_ITER, nEigens, 1);
25
26    // compute average image, eigenvalues, and eigenvectors
27    cvCalcEigenObjects(
28        nTrainFaces,
29        (void*)faceImgArr,
30        (void*)eigenVectArr,
31        CV_EIGOBJ_NO_CALLBACK,
32        0,
33        0,
34        &calcLimit,
35        pAvgTrainImg,
36        eigenValMat->data.fl);
37 }

```

Εικόνα 6.6. Η συνάρτηση doPCA (). Βρίσκει PCA subspace.

Οι γραμμές 11-15 δημιουργούν τον εικονικό πίνακα eigenVectArr. Όταν η cvCalcEigenObjects () επιστρέφει, την κάθε εικόνα, αυτός ο πίνακας θα κρατήσει ένα eigenvector, ή αλλιώς ένα "eigenface". Αυτές είναι δεκαδικές(float) εικόνες, οπότε χρησιμοποιούμε βάθος στοιχείων = IPL\_DEPTH\_32F

Στη γραμμή 18, δημιουργείται ένας άλλος πίνακας, ο πίνακας eigenValMat. Αυτός ο πίνακας θα κρατήσει τα eigenvalues. Τα Eigenvalues είναι δεκαδικοί(float) αριθμοί, και για αυτό χρειαζόμαστε μόνο ένα κανάλι, έτσι ο τύπος του πίνακα είναι CV\_32FC1. Αυτός μας δίνει ένα πίνακα ενός-καναλιού, με τις τιμές 32-bit, float στοιχείων.

Για να κάνει PCA, το σύνολο δεδομένων πρέπει πρώτα να είναι κεντραρισμένο(centered). Για τις εικόνες προσώπου, αυτό σημαίνει ότι πρέπει να βρει τη μέση εικόνα - μια εικόνα στην οποία κάθε εικονοκύτταρο(pixel) περιέχει τη μέση αξία για εκείνο το pixel σε όλες τις εικόνες προσώπου εκπαίδευσης. Το σύνολο δεδομένων κεντροθετείται με την αφαίρεση της μέσης τιμής του κάθε εικονοκυττάρου από κάθε εικόνα εκπαίδευσης.

Αυτό συμβαίνει μέσα στη cvCalcEigenObjects (). Αλλά πρέπει να κρατήσουμε τη μέση εικόνα, επειδή θα τη χρειαστούμε για να προβάσουμε αργότερα τα στοιχεία. Έτσι δεσμεύουμε μνήμη για τη μέση εικόνα. Ο κώδικας που το κάνει αυτό είναι στη γραμμή 21. Όπως και τα eigenvectors, έτσι και αυτό είναι μια float εικόνα.

Το τελευταίο βήμα, πριν καλέσουμε την cvCalcEigenObjects () είναι να προετοιμάσουμε μια δομή δεδομένων αποκαλούμενη CvTermCriteria. Οι τομείς σε αυτήν την δομή διευκρινίζουν τα κριτήρια λήξης για τους επαναληπτικούς αλγορίθμους όπως το PCA. Απλά του λέμε να υπολογίσει κάθε eigenvalue, και μετά να σταματήσει, αφού αυτό είναι το μόνο που χρειαζόμαστε. Ο κώδικας για αυτό είναι στη γραμμή 24.

Τώρα που όλες οι μεταβλητές είναι έτοιμες, καλούμε την cvCalcEigenObjects () για να υπολογίσουμε τον PCA subspace για τα πρόσωπα εκπαίδευσης. Η τελευταία παράμετρος, eigenValMat-> data.fl, είναι ο δείκτης στις τιμές στοιχείων στο eigenValMat. Εδώ, χρησιμοποιούμε το data.fl και, όχι το data.i, δεδομένου ότι αυτή η μεταβλητή του πίνακα φυλάσσει τα float δεδομένα.

### **3° Βήμα : Προβολή των εικόνων εκπαίδευσης**

Τώρα που έχουμε βρει το subspace χρησιμοποιώντας το PCA, μπορείτε να μετατρέψουμε τις εικόνες εκπαίδευσης σε σημεία πάνω σε αυτό το subspace. Αυτό το βήμα καλείται "projecting". Η συνάρτηση του OpenCV για αυτό το βήμα καλείται cvEigenDecomposite() (σχήμα 4, γραμμή 22).

Τα ονόματα των συναρτήσεων του OpenCV, δυστυχώς, δεν μας διευκολύνουν, πιο πολύ μας παραπλανούν και μας μπερδεύουν. Όχι μόνο η συνάρτηση προβολής έχει περίεργη ονομασία, αλλά επίσης υπάρχει και συνάρτηση που ονομάζεται "EigenProjection" αλλά δεν προβάλλει τα στοιχεία εικόνας προγράμματος επάνω στο subspace. Στην πραγματικότητα, κάνει το αντίθετο. Αποκαθιστά (ασυμπιεστα) προβαλλόμενα στοιχεία, και τα επαναφέρει στην αρχική εικόνα.

Τέλος θα χρειαστούμε μια θέση για να βάλουμε τις προβαλλόμενες εικόνες εκπαίδευσης. Η γραμμή 19, στο σχήμα 4, δημιουργεί ένα πίνακα ακριβώς για αυτό το σκοπό. Ο βρόχος For, στις γραμμές 20-29, καλεί cvEigenDecomposite() μιά φορά για κάθε εικόνα κατάρτισης.

### **4° Βήμα : Αποθήκευση της πρότυπης εικόνας εκπαίδευσης**

Εδώ θα καταλάβουμε για ποιο λόγο χρησιμοποιήσαμε την μεταβλητή τύπου CvMat για να σωθούν τα στοιχεία κατάρτισης! Η εικ. 7 παρουσιάζει τον κώδικα για τη διάσωση όλων των στοιχείων για την αντιπροσώπηση εκπαίδευσης των προσώπων μας ως αρχείο XML χρησιμοποιώντας τις λειτουργίες του OpenCV.

```
storeTrainingData()
1 void storeTrainingData()
2 {
3     CvFileStorage * fileStorage;
4     int i;
5
6     // create a file-storage interface
7     fileStorage = cvOpenFileStorage( "facedata.xml", 0, CV_STORAGE_WRITE );
8
9     // store all the data
10    cvWriteInt( fileStorage, "nEigens", nEigens );
11    cvWriteInt( fileStorage, "nTrainFaces", nTrainFaces );
12    cvWrite( fileStorage, "trainPersonNumMat", personNumTruthMat, cvAttrList(0,0));
13    cvWrite( fileStorage, "eigenValMat", eigenValMat, cvAttrList(0,0));
14    cvWrite( fileStorage, "projectedTrainFaceMat", projectedTrainFaceMat, cvAttrList(0,0));
15    cvWrite( fileStorage, "avgTrainImg", pAvgTrainImg, cvAttrList(0,0));
16    for(i=0; i<nEigens; i++)
17    {
18        char varname[200];
19        sprintf( varname, "eigenVect_%d", i );
20        cvWrite( fileStorage, varname, eigenVectArr[i], cvAttrList(0,0));
21    }
22
23    // release the file-storage interface
24    cvReleaseFileStorage( &fileStorage );
25 }
```

Εικόνα 6.7. Η συνάρτηση storeTrainingData(). Αποθηκεύει τα στοιχεία της εκπαίδευσης ως αρχείο XML.

Στη γραμμή 7, η κλήση της συνάρτησης `cvOpenFileStorage ()` ανοίγει ένα τύπου XML π.χ. `profile-face_data.xml`. Η τελευταία παράμετρος αυτής της συνάρτησης ελέγχει τον τρόπο πρόσβασης. Εδώ, είναι `CV_STORAGE_WRITE`, το οποίο σημαίνει να δημιουργήσει (ή να επικαλύψει) το προαναφερθέν αρχείο και να το ανοίξει για γράψιμο.

Για να γράψουμε τα βασικά της C-γλώσσας π.χ. ακέραιους αριθμούς, float τιμές, και συμβολοσειρές(strings) – σε μορφή XML, μπορούμε να χρησιμοποιήσουμε τις συναρτήσεις `cvWrite ()`. Παραδείγματος χάριν, η κλήση `cvWriteInt ()`, στη γραμμή 10, γράφει τον αριθμό `eigenvalues` ως 2.

Καλό με τις συναρτήσεις του `OpenCV` είναι ότι είναι εύκολο να σώσεις σύνθετα `datatypes`, όπως μια εικόνα ή ένας πίνακας. Οι γραμμές 12 - 15 προσθέτουν τρεις πίνακες και μια εικόνα στο ίδιο αρχείο XML. Οι ενσωματωμένες λειτουργίες αποθηκεύουν, όχι μόνο τις σειρές και τις στήλες με τα δεδομένα, αλλά και όλες τις πληροφορίες επικεφαλίδων επίσης. Εδώ είναι το αποτέλεσμα σε XML που παράγει η γραμμή 13 της εικόνας 6.7:

```
<eigenValMat type_id="opencv-matrix">  
<rows>1</rows>  
<cols>2</cols>  
<dt>f</dt>  
<data>  
14279064. 9614034.</data></eigenValMat>
```

Η δεύτερη παράμετρος των συναρτήσεων `cvWrite ()` είναι μια σειρά. Η σειρά μπορεί να μην είναι κατι που μας ενδιαφέρει, αλλά για να εξασφαλίσουμε μοναδικότητα, καλό είναι να έχει το ίδιο όνομα με το όνομα της μεταβλητής.

Όταν τελειώσουμε το γράψιμο δεδομένων, κλείνουμε το αρχείο και απελευθερώνουμε την μνήμη που δεσμεύσαμε όπως στη γραμμή 24.

## Η φάση της αναγνώρισης

Το σχήμα 8 παρουσιάζει τη συνάρτηση `recognition()`, η οποία εφαρμόζει τη φάση αναγνώρισης του προγράμματος `Eigenface` και έχει ακριβώς τρία βήματα. Τα δύο από αυτά - φόρτωση των εικόνων προσώπου και η προβολή τους επάνω στο `subspace` - είναι ήδη γνωστά.

```

recognize()
1 void recognize()
2 {
3     int i, nTestFaces = 0;           // the number of test images
4     CvMat * trainPersonNumMat = 0;  // the person numbers during training
5     float * projectedTestFace = 0;
6
7     // load test images and ground truth for person number
8     nTestFaces = loadFaceImgArray("test.txt");
9     printf("%d test faces loaded\n", nTestFaces);
10
11    // load the saved training data
12    if( !loadTrainingData( &trainPersonNumMat ) ) return;
13
14    // project the test images onto the PCA subspace
15    projectedTestFace = (float *)cvAlloc( nEigens*sizeof(float) );
16    for(i=0; i<nTestFaces; i++)
17    {
18        int iNearest, nearest, truth;
19
20        // project the test image onto the PCA subspace
21        cvEigenDecomposite(
22            faceImgArr[i],
23            nEigens,
24            eigenVectArr,
25            0, 0,
26            pAvgTrainImg,
27            projectedTestFace);
28
29        iNearest = findNearestNeighbor( projectedTestFace );
30        truth     = personNumTruthMat->data.i[i];
31        nearest  = trainPersonNumMat->data.i[iNearest];
32
33        printf("nearest = %d, Truth = %d\n", nearest, truth);
34    }
35 }

```

Εικόνα 6.8. Η συνάρτηση recognition(). Η φάση αναγνώρισης του προγράμματος Eigenface.

Όπως περιγράφεται παραπάνω, οι εικόνες προσώπου για τη δοκιμή αναγνώρισης πρέπει να απαριθμηθούν σε ένα αρχείο test.txt, χρησιμοποιώντας την ίδια μορφή όπως στο train.txt. Στη γραμμή 8, η κλήση της loadFaceImgArray () τις φορτώνει στο faceImgArr και τα κρατάει το “ground truth” για τον αριθμό ταυτότητας προσώπων στο personNumTruthMat. Αυτό το βήμα είναι παρόμοιο με τη γραμμή 6 της συνάρτησης learn(), στην εικ. 6.4. Εδώ, ο αριθμός εικόνων προσώπου αποθηκεύεται στην τοπική μεταβλητή, nTestFaces.

Πρέπει επίσης να φορτώσουμε τη global μεταβλητή nTrainFaces καθώς επίσης και τα περισσότερα από τα άλλα δεδομένα της εκπαίδευσης - nEigens, EigenVectArr, pAvgTrainImg, και ούτω καθεξής. Αυτό το κάνει για μας η συνάρτηση loadTrainingData(), στο σχήμα 9. Και πάλι οι συναρτήσεις του OpenCV καθιστούν αυτό το βήμα εύκολο. Για να ανοίξουμε ένα αποθηκευμένο αρχείο για ανάγνωση, χρησιμοποιήστε τη σημαία CV\_STORAGE\_READ. Κατόπιν, απλά καλούμε την συνάρτηση read() για κάθε μεταβλητή. Το OpenCV εντοπίζει και φορτώνει κάθε στοιχείο που θέλουμε στο αρχείο XML με βάση το όνομα του. Όταν η μεταβλητή είναι ένας πίνακας τύπου CvMat, το OpenCV δημιουργεί ένα νέο πίνακα για μας αυτόματα, και τον γεμίζει με τις τιμές του CvMat.



```

loadTrainingData()
1 int loadTrainingData(CvMat ** pTrainPersonNumMat)
2 {
3     CvFileStorage * fileStorage;
4     int i;
5
6     // create a file-storage interface
7     fileStorage = cvOpenFileStorage( "facedata.xml", 0, CV_STORAGE_READ );
8     if( !fileStorage )
9     {
10        fprintf(stderr, "Can't open facedata.xml\n");
11        return 0;
12    }
13
14    nEigens = cvReadIntByName(fileStorage, 0, "nEigens", 0);
15    nTrainFaces = cvReadIntByName(fileStorage, 0, "nTrainFaces", 0);
16    *pTrainPersonNumMat = (CvMat *)cvReadByName(fileStorage, 0, "trainPersonNumMat", 0);
17    eigenValMat = (CvMat *)cvReadByName(fileStorage, 0, "eigenValMat", 0);
18    projectedTrainFaceMat =
19        (CvMat *)cvReadByName(fileStorage, 0, "projectedTrainFaceMat", 0);
20    pAvgTrainImg = (IplImage *)cvReadByName(fileStorage, 0, "avgTrainImg", 0);
21    eigenVectArr = (IplImage **)cvAlloc(nTrainFaces*sizeof(IplImage *));
22    for(i=0; i<nEigens; i++)
23    {
24        char varname[200];
25        sprintf( varname, "eigenVect_%d", i );
26        eigenVectArr[i] = (IplImage *)cvReadByName(fileStorage, 0, varname, 0);
27    }
28    // release the file-storage interface
29    cvReleaseFileStorage( &fileStorage );
30
31    return 1;
32 }

```

**Εικόνα 6.9.** Οι συναρτήσεις του OpenCV' για τη φόρτωση των δεδομένων εκπαίδευσης από το αρχείο XML

Η τελευταία παράμετρος της συνάρτησης read() είναι μια προκαθορισμένη τιμή. Εάν μια συγκεκριμένη μεταβλητή λείπει από το αρχείο XML, τότε την θέτει στην προεπιλεγμένη τιμή. Για τους τύπους δεικτών, όπως οι πίνακες, είναι καλή ιδέα να θέσουμε την προεπιλογή σε 0. Μπορούμε μετά να ελέγχουμε αυτούς τους δείκτες για να σιγουρευτούμε ότι έχουν μια διαφορετική από το μηδέν τιμή προτού να τους χρησιμοποιήσουμε.

Αφού φορτωθούν όλα τα στοιχεία, το τελικό βήμα στη φάση αναγνώρισης είναι να προβληθεί κάθε εικόνα δοκιμής επάνω στο subspace του PCA και να βρεθεί η πιο κοντινή προβαλλόμενη εικόνα εκπαίδευσης. Ο βρόχος For, στις γραμμές 16-34 της συνάρτησης recognition() (εικ. 6.8), εφαρμόζει αυτό το τελικό βήμα. Η κλήση της cvEigenDecomposite(), που προβάλλει τις εικόνες δοκιμής, είναι παρόμοια της συνάρτησης learn().

Όπως και πριν, της περνάμε τις τιμές eigenvalues (nEigens), και τη σειρά eigenvectors (eigenVectArr). Αυτή τη φορά, όμως, περνάμε μια εικόνα δοκιμής, αντί μιας εικόνας εκπαίδευσης, ως πρώτη παράμετρο. Η παραγωγή της cvEigenDecomposite() αποθηκεύεται σε μια τοπική μεταβλητή - projectedTestFace. Επειδή δεν χρειάζεται να αποθηκευτεί η προβαλλόμενη εικόνα δοκιμής, χρησιμοποιούμε απλό πίνακα της C για την projectedTestFace, και όχι καποιον πίνακα του OpenCV.

### Εύρεση του κοντινότερου γείτονα

Το eigenface αναγνωρίζει μια εικόνα προσώπου ψάχνοντας για την εικόνα εκπαίδευσης που είναι πιο κοντά στο subspace του PCA. Η εύρεση της πιο κοντινής εικόνας εκπαίδευσης σε ένα subspace είναι μια πολύ κοινή τεχνική της τεχνητής νοημοσύνης(AI). Ονομάζεται κοντινότερο ταίριασμα γειτόνων(Nearest Neighbor matching).

Η εικόνα 6.10 παρουσιάζει τον κώδικα της συνάρτησης findNearestNeighbor(). Υπολογίζει την απόσταση από την προβαλλόμενη εικόνα δοκιμής σε κάθε προβαλλόμενο παράδειγμα εκπαίδευσης. Η απόσταση εδώ είναι μια "Τετραγωνική Ευκλείδεια απόσταση". Για να υπολογίσουμε μια Ευκλείδεια απόσταση μεταξύ δύο σημείων, προσθέτουμε την τετραγωνική απόσταση σε κάθε διάσταση, και υπολογίζουμε την τετραγωνική της ρίζα. Εδώ, παίρνουμε το ποσό, αλλά πηδάμε το βήμα της τετραγωνικής ρίζας. Το τελικό αποτέλεσμα είναι το ίδιο, επειδή ο γείτονας με τη μικρότερη απόσταση έχει επίσης τη μικρότερη τετραγωνική απόσταση, έτσι μπορούμε να κερδίσουμε κάποιο χρόνο υπολογισμού με τη σύγκριση των τετραγωνικών τιμών.

```

findNearestNeighbor()
1 int findNearestNeighbor(float * projectedTestFace)
2 {
3     double leastDistSq = DBL_MAX;
4     int i, iTrain, iNearest = 0;
5
6     for(iTrain=0; iTrain<nTrainFaces; iTrain++)
7     {
8         double distSq=0;
9
10        for(i=0; i<nEigens; i++)
11        {
12            float d_i =
13                projectedTestFace[i] -
14                projectedTrainFaceMat->data.fl[iTrain*nEigens + i];
15            distSq += d_i*d_i;
16        }
17
18        if(distSq < leastDistSq)
19        {
20            leastDistSq = distSq;
21            iNearest = iTrain;
22        }
23
24    return iNearest;
25 }

```

**Εικόνα 6.10.** Η συνάρτηση findNearestNeighbor (). Υπολογίζει την απόσταση από την προβαλλόμενη εικόνα δοκιμής σε κάθε προβαλλόμενο παράδειγμα εκπαίδευσης για να βρεί την πιο κοντινή εικόνα εκπαίδευσης

Ο βρόχος For στις γραμμές 6-22 υπολογίζει την τετραγωνική απόσταση σε κάθε προβαλλόμενη εικόνα εκπαίδευσης, και κρατά(στις γραμμές 18-21) την πιο κοντινή εικόνα εκπαίδευσης.

Η τιμή που επιστρέφει είναι ο δείκτης στην πιο κοντινή εικόνα εκπαίδευσης. Στη συνάρτηση recognize() στο σχήμα 8, αυτή η τιμή χρησιμοποιείται, στη γραμμή 31, για να βρεθεί ο αριθμός ταυτότητας προσώπων που συνδέεται με την κοντινότερη εικόνα κατάρτισης.



#### 6.4 Αποτελέσματα του κώδικα με την βάση δεδομένων UMIST

Όπως αναφέραμε και στο κεφ. 2, η UMIST είναι μια βάση δεδομένων εικόνων από γωνία 90 μοιρών που αποτελείται από εικόνες που έχουν τραβηχτεί από 18 διαφορετικά άτομα και διαφέρουν σε πόζες και φωτισμούς. Δηλαδή το κάθε άτομο καθόταν σε γωνία 90 μοιρών από μια κάμερα και του ζητούνταν να στρέφει το κεφάλι του αργά και σταθερά μέχρι να έρθει κατά μέτωπο με την κάμερα, κατά τη διάρκεια αυτής της κίνησης η κάμερα τραβούσε ένα τυχαίο αριθμό εικόνων.

Εμείς για την δοκιμή του κώδικα πήραμε 1 εικόνα από 16 από τα 18 διαφορετικά άτομα για να χρησιμοποιήσουμε σαν εικόνες εκπαίδευσης, και 80 τυχαίες εικόνες και από τα 18 άτομα εξαιρουμένων των 16 εικόνων που χρησιμοποιήθηκαν για την εκπαίδευση.

Όπως θα δείτε και από την στιγμιότυπο στην επόμενη σελίδα. Τα αποτελέσματα το λιγότερο ικανοποιητικά. Από 80 εικόνες δοκιμής 74 σωστές αναγνωρίσεις και 6 λάθος. Δηλαδή ποσοστό ακρίβειας του κώδικα λίγο μεγαλύτερο από 92%.

```
ca. Command Prompt
80 test faces loaded
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 1, Truth = 1
nearest = 2, Truth = 2
nearest = 2, Truth = 2
nearest = 2, Truth = 2
nearest = 2, Truth = 2
nearest = 14, Truth = 2
nearest = 2, Truth = 2
nearest = 15, Truth = 2
nearest = 15, Truth = 2
nearest = 3, Truth = 3
nearest = 3, Truth = 3
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 4, Truth = 4
nearest = 6, Truth = 6
nearest = 6, Truth = 6
nearest = 7, Truth = 7
nearest = 7, Truth = 7
nearest = 7, Truth = 7
nearest = 8, Truth = 8
nearest = 8, Truth = 8
nearest = 8, Truth = 8
nearest = 9, Truth = 9
nearest = 9, Truth = 9
nearest = 9, Truth = 9
nearest = 9, Truth = 9
nearest = 9, Truth = 9
nearest = 12, Truth = 9
nearest = 12, Truth = 9
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 10, Truth = 10
nearest = 11, Truth = 11
nearest = 11, Truth = 11
nearest = 11, Truth = 11
nearest = 11, Truth = 11
nearest = 11, Truth = 11
nearest = 12, Truth = 12
nearest = 12, Truth = 12
nearest = 12, Truth = 12
nearest = 12, Truth = 12
nearest = 13, Truth = 13
nearest = 13, Truth = 13
nearest = 13, Truth = 13
nearest = 14, Truth = 14
nearest = 14, Truth = 14
nearest = 14, Truth = 14
nearest = 14, Truth = 14
nearest = 15, Truth = 15
nearest = 15, Truth = 15
nearest = 15, Truth = 15
nearest = 15, Truth = 15
nearest = 15, Truth = 15
nearest = 16, Truth = 16
nearest = 16, Truth = 16
nearest = 16, Truth = 16
C:\Users\pagou\part5-src >_
```

Εικόνα 6.11 : Στιγμιότυπο δοκιμής του κώδικα με την βάση UMIST

## 6.5 Τρόποι Βελτίωσης του Eigenface

Για να βελτιώσουμε την απόδοση αναγνώρισης, υπάρχουν πολλά πράγματα που μπορούμε να κάνουμε, μερικά απ' αυτά αρκετά εύκολα να εφαρμόσουν. Παραδείγματος χάριν, θα μπορούσαμε να προσθέσουμε επεξεργασία χρώματος, ανίχνευση ακρών (edge detection) κ.α.

Μπορούμε συνήθως να βελτιώσουμε την ακρίβεια αναγνώρισης προσώπου με τη χρησιμοποίηση περισσότερων εικόνων εισαγωγής, τουλάχιστο 50 ανά άτομο, με τη λήψη περισσότερων φωτογραφιών κάθε προσώπου, ιδιαίτερα από διαφορετικές γωνίες και φωτισμό. Υπάρχουν διάφορες απλές τεχνικές που θα μπορούσαμε να χρησιμοποιήσουμε για να πάρουμε περισσότερες εικόνες εκπαίδευσης, παράγοντας νέες εικόνες από τις ήδη υπάρχουσες :

- Θα μπορούσαμε να δημιουργήσουμε αντίγραφα καθρεφτών των εικόνων μας, έτσι ώστε να έχουμε δύο φορές τις εικόνες εκπαίδευσης.
- Θα μπορούσαμε να μεταφράσουμε ή να επαναταξινομήσουμε ή να περιστρέψουμε τις εικόνες μας για να παράγουμε ελαφρώς πολλές εναλλακτικές εικόνες για την εκπαίδευση, έτσι ώστε να είναι λιγότερο ευαίσθητος ο κώδικας στους ακριβείς όρους.
- Θα μπορούσαμε να προσθέσουμε θόρυβο εικόνας για να έχουμε περισσότερες εικόνες εκπαίδευσης που βελτιώνουν την ανοχή στο θόρυβο.

Είναι σημαντικό να έχουμε πολλές παραλλαγές των όρων για κάθε πρόσωπο, έτσι ώστε ο ταξινομητής θα είναι σε θέση να αναγνωρίσει το πρόσωπο σε διαφορετικούς όρους και θέσεις φωτισμού.

Μια άλλη βελτίωση που μπορούμε να κάνουμε είναι να αλλάξουμε τον τρόπο μέτρησης της απόστασης. Ο αρχικός κώδικας του Eigenface χρησιμοποίησε την τετραγωνική Ευκλείδεια απόσταση μεταξύ δυο σημείων, που χρησιμοποιείται από τη συνάρτηση `findNearestNeighbors()`. Αλλά υπάρχει και μια διαφορετική μονάδα μέτρησης της απόστασης. Η αποκαλούμενη απόσταση "Mahalanobis" και συνήθως δίνει καλύτερα αποτελέσματα.

Ένα από τα πράγματα που συμβαίνει όταν προβάλλουμε μια εικόνα προσώπου επάνω στο subspace του PCA είναι ότι κάθε διάσταση τεντώνεται σε ένα μικρό βαθμό. Το ποσό τεντώματος δεν είναι το ίδιο σε κάθε κατεύθυνση. Οι συντεταγμένες που αντιστοιχούν στα μεγαλύτερα eigenvalues τεντώνονται (παραμορφώνονται) πολύ περισσότερο από τις συντεταγμένες που συνδέονται με μικρότερα eigenvalues. Επειδή η Ευκλείδεια απόσταση αγνοεί αυτό το τέντωμα, η χρησιμοποίησή της για να μετρήσει την απόσταση είναι περίπου η ίδια με το να χρησιμοποιήσουμε μόνο ένα eigenvector και να αγνοήσουμε τους υπόλοιπους.

Είναι εύκολο να αλλάξουμε την απόσταση από Ευκλείδεια σε απόσταση "Mahalanobis". Απλά αλλάζουμε ο γνώρισμα της συνάρτησης `findNearestNeighbors()` στη γραμμή 15 από :

```
distSq += d_i*d_i;
```

σε :

```
distSq += d_i*d_i/eigenValMat->data.fl[i];
```

Η αλλαγή αυτή στην απόσταση εξαλείφει το λάθος κακού που προαναφέραμε, και επομένως αυξάνει και την ακρίβεια αναγνώρισης.

Είναι σημαντικό να γίνει πολύς πειραματισμός εάν θέλουμε καλύτερα αποτελέσματα, και εάν δεν μπορούμε να πάρουμε τα επιθυμητά αποτελέσματα ίσως να πρέπει να δοκιμάσουμε πιο περίπλοκο αλγόριθμο – που δεν έχει αναλυθεί σε αυτή την εργασία - αναγνώρισης προσώπου από PCA (Eigenfaces), όπως η τρισδιάστατη αναγνώριση προσώπου ή τα ενεργά πρότυπα εμφάνισης (Active Appearance Models).

## Κεφάλαιο 7

### Συμπεράσματα

Σε αυτό το έγγραφο, παρουσιάστηκε μια αναθεώρηση των προσεγγίσεων ανίχνευσης προσώπου από γωνία 90 μοιρών. Αρχικά μιλήσαμε για τις διαθέσιμες βάσεις δεδομένων που ήταν χρησιμότερες για την εργασία μας. Ξεκινήσαμε χτίζοντας ένα σύστημα ανίχνευσης προσώπων από γωνία 90 μοιρών αξιόπιστο και αποδοτικό ώστε να μπορεί να χρησιμοποιηθεί για την αναγνώριση προσώπου από γωνία 90 μοιρών. Προκειμένου να γίνει αυτό, είναι σημαντικό να ανιχνευθούν τα πρόσωπα με ακρίβεια περίπου 95%.

Έχουμε δει ότι, οι προηγούμενες δουλειές όπως ο προ-εκπαιδευμένος κώδικας του openCV για ανίχνευση προσώπων από γωνία 90 μοιρών, μπορεί να είναι πολύ ευέλικτος εάν αλλάζουμε τις παραμέτρους και αποδίδει καλύτερα σε ελεγχόμενα περιβάλλοντα και σε εικόνες που περιέχουν ένα πρόσωπο σε ένα κενό υπόβαθρο, αλλά δεν αποδίδει το ίδιο στις ανεξέλεγκτες τοποθετήσεις με διαφορετικούς φωτισμούς και “πόζες” και στα δύσκολα υπόβαθρα όπου τα πειραματικά αποτελέσματα έδειξαν ότι στη βάση δεδομένων GUARANTEE το σύστημα ανίχνευσε σωστά 75 από τις 174 εικόνες. Έτσι συνειδητοποιήσαμε ότι εάν θέλουμε να ανιχνεύσουμε τα πρόσωπα κάτω από τις συγκεκριμένες περιστάσεις η καλύτερη λύση είναι να εκπαιδευσουμε ένα δικό μας σύστημα με παρόμοιες εικόνες ως θετικά παραδείγματα.

Κατόπιν είδαμε πώς λειτουργεί η εφαρμογή εκπαίδευσης, και πώς να οργανώσουμε τις εικόνες προσώπου και τις εικόνες υποβάθρου, για να ολοκληρώσουμε την εκπαίδευση του κώδικα με τα χαρακτηριστικά γνωρίσματα haar. Μέσα από μια σειρά διαφορετικών εκπαιδύσεων παρατηρήσαμε ότι η εκπαίδευση Haar έχει πολλές παραμέτρους που μπορούν να αλλάξουν σημαντικά τα αποτελέσματα.

Ένα από τα σημαντικότερα πράγματα στην εκπαίδευση είναι να αποφασιστεί τι είδος εικόνων πρόκειται να χρησιμοποιήσετε. Εάν θέλετε να έχετε τα καλά αποτελέσματα, τα θετικά παραδείγματα πρέπει να είναι παρόμοια με τις εικόνες που πρόκειται να εξετάσετε τον καταρράκτη.

Στην πρώτη εκπαίδευση οι εικόνες ήταν παρόμοιες με τη βάση δεδομένων CMU και αυτός είναι ο λόγος που είχαμε τα καλά αποτελέσματα σε αυτήν την βάση δεδομένων. Ενώ δεν ήταν παρόμοια με τη βάση δεδομένων GUARANTEE και αυτό εξηγεί τα όχι καλά αποτελέσματα. Στη δεύτερη εκπαίδευση υπήρξε μια αύξηση στην απόδοση εξαιτίας του γεγονότος ότι εκπαιδύσαμε το σύστημα χρησιμοποιώντας θετικά παραδείγματα με περισσότερες “πόζες”.

Μια άλλη σημαντική παράμετρος της εκπαίδευσης είναι ο αριθμός των σταδίων. Συνειδητοποιήσαμε ότι συσχετίζεται με τις λανθασμένα - θετικές ανιχνεύσεις. Όσα περισσότερα στάδια χρησιμοποιείτε για να εκπαιδύσετε έναν καταρράκτη, τόσο λιγότερες λανθασμένα - θετικές ανιχνεύσεις πρόκειται να έχετε. Κάνει επίσης το σύστημα λίγο πιο αργό, έτσι εάν έχετε μια βάση δεδομένων όπου όλες οι εικόνες περιλαμβάνουν πρόσωπα, δεν πρέπει να εκπαιδύσετε τον καταρράκτη σας μέσω πολλών σταδίων. Αλλά εάν η βάση δεδομένων σας περιλαμβάνει επίσης έναν μεγάλο αριθμό εικόνων χωρίς πρόσωπα είναι καλύτερο να εκπαιδευθεί ο καταρράκτης σας μέσω πολλών σταδίων.

Λόγω του ότι ο μόνο προ-εκπαιδευμένος καταρράκτης δεν είναι επαρκής σε εικόνες τραβηγμένες σε ανεξέλεγκτο περιβάλλον, η χρήση της haar-εκπαίδευσης του openCV, μπορεί να θεωρηθεί ως μια από τις καλύτερες λύσεις για να χτίσει ένας καταρράκτης ανίχνευσης προσώπου από γωνία 90 μοιρών που θα λειτουργήσει με αποδοτικό και αξιόπιστο τρόπο.

Όσον αφορά την αναγνώριση προσώπου από γωνία 90 μοιρών αναλύσαμε τις υπάρχουσες τεχνικές και μεθόδους και υλοποιήσαμε ένα σύστημα αναγνώρισης προσώπων από γωνία 90 μοιρών, με την τεχνική των Eigenfaces (PCA). Καταλήξαμε στο συμπέρασμα ότι ένα σύστημα αναγνώρισης προσώπων είναι άμεσα εξαρτώμενο από το καλά θα λειτουργεί το σύστημα της ανίχνευσης.

Για να έχουμε τα καλύτερα δυνατά αποτελέσματα, προτείνουμε το σύστημα ανίχνευσης να “εκπαιδευτεί” με εικόνες ίδιου τύπου (δηλαδή με πρόμοιο φωτισμό και εάν

είναι δυνατόν με το ίδιο υπόβαθρο “background”), με αυτές που θα χρησιμοποιούνται για την αναγνώριση.

Με την εκπόνηση της εργασίας αυτής μου έγινε φανερό ότι το πεδίο της Αναγνώρισης Προσώπων είναι ένα ευρύ πεδίο άκρως ενδιαφέρουσας έρευνας που απαιτεί τη χρήση νέων αλλά και δοκιμασμένων μεθόδων και τεχνικών από τα Γραφικά, την Επεξεργασία Εικόνας, την Τεχνητή Όραση και την Αναγνώριση Προτύπων.

## ΑΝΑΦΟΡΕΣ \ ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] R. Hietmeyer. Biometric identification promises fast and secure processing of airline passengers. *The International Civil Aviation Organization Journal*, 55(9):10-11, 2000.
- [2] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001.
- [3] P. Viola and M. Jones. Robust Real-time Object Detection. In *IEEE ICCV Workshop on Statistical and Computational Theories of Vision*, 2001.
- [4] R. Lienhart and J. Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. *IEEE ICIP* 2002.
- [5] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. MRL Technical Report, Intel Labs, 2002
- [6] Li, S., Zhu, L., Zhang, Z., Blake, A., Zhang, H., Shum, H.: Statistical learning of multiview face detection. In: *Proc. Seventh European Conf. Computer Vision*. Volume 4., Copenhagen, Denmark (2002) 67–81
- [7] Yen-Yu Lin, Tyng-Luh Liu, Chiou-Shann Fuh : Face Detection with Occlusions. Inst. of Information Science, Academia Sinica, Taiwan Department of CSIE, National Taiwan University
- [8] How-to build a cascade of boosted classifiers based on Haar-like features. [http://lab.cntl.kyutech.ac.jp/~kobalab/nishida/opencv/OpenCV\\_ObjectDetection\\_HowTo.pdf](http://lab.cntl.kyutech.ac.jp/~kobalab/nishida/opencv/OpenCV_ObjectDetection_HowTo.pdf)
- [9] [http://www.cognotics.com/opencv/servo\\_2007\\_series/part\\_2/page\\_2.html](http://www.cognotics.com/opencv/servo_2007_series/part_2/page_2.html)
- [10] S. Z. Li, L. Zhu, Z. Q. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical Learning of Multi-view Face Detection. In *Proceedings Of the European Conference on Computer Vision*, 2002.
- [11] S. Z. Li and A. K. Jain (ed.). *Handbook of Face Recognition*, Springer 2005
- [12] OpenCV Library. [www.intel.com/research/mrl/research/opencv](http://www.intel.com/research/mrl/research/opencv).
- [13] M.-H. Yang, D. Kriegman, and N. Ahuja. Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.
- [14] I. Dryden and K. Mardia. *Statistical Shape Analysis*, Wiley 1998.
- [15] I. Craw, D. Tock and A. Benett. Finding Face Features. In *Proc. Second European Conf. Computer Vision*, 1992.
- [16] A. Lanitis, C. J. Taylor and T. Cootes. An Automatic Face Identification System Using Flexible Appearance Models. In *Image and Vision Computing*, 1995.
- [17] C. P. Papageorgiou, M. Oren, and T. Poggio. A General Framework for Object Detection. In *Proceedings of IEEE International Conference on Computer Vision*, 1998.

- [18] K-K. Sung and T. Poggio. Example-Based Learning for View-Based Human Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.
- [19] H. A. Rowley, S. Baluja, and T. Kanade. Neural Network-Based Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.
- [20] H. Schneiderman and T. Kanade. A Statistical Approach to 3D Object Detection Applied to Faces and Cars. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2000*.
- [21] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting, AT&T Labs, 1995.
- [22] G. Bradski, A. Kaehler and V. Pisarevsky. Learning-Based Computer Vision with Intel's Open Source Computer Vision Library. In *Intel Technology Journal*, Vol. 09-Iss.02, May 2005.
- [23] D. Blackburn, J. Bone, and P. Phillips, "Face Recognition Vendor Test 2000, Evaluation Report," Tech. Rep., February 2001.
- [24] P. Phillips, P. Grother, R. Micheals, D. Blackburn, E. Tabassi, and J. Bone, "Face Recognition Vendor Test 2002, Evaluation Report," Tech. Rep., March 2003.
- [25] N. Tsapatsoulis : Προηγμένες Τεχνικές Αναγνώρισης Προσώπων και Ανάλυσης Εκφράσεων, Ph.D. Thesis, School of Electrical and Computer Engineering, National Technical University of Athens, 2000.
- [26] C. Havran, L. Hupet, J. Czyz, J. Lee, L. Vandendorpe, M. Verleysen : Independent Component Analysis for face authentication, Universite catholique de Louvain, Electricity Dept., Belgium, 2002.
- [27] Xiaoguang Lu : Image Analysis for Face Recognition, Dept. of Computer Science & Engineering Michigan State University, 2003, pp. 7-10.
- [28] A. J. Bell and T. J. Sejnowski : n information-maximization approach to blind separation and blind deconvolution, *Neural Comput.*, vol. 7, no. 6, 1995, pp. 1129-1159.
- [29] P. Ekman : Strong evidence for universals in facial expressions: a reply to Russell's mistaken critique, *Psychological Bulletin* 115(2), 1994, pp. 268-287.
- [30] L. Yin, X. Wei, Y. Sun, J. Wang, M. Rosato: A 3D Facial Expression Database For Facial Behavior Research, *Proceedings of 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 211-216.
- [31] C. Kotropoulos, A. Tefas, I. Pitas, Frontal face authentication using discriminating grids with morphological feature vectors, *IEEE Trans. Multimedia* 2 (1) (2000) 14–26.
- [32] Stefanos Zafeiriou, Anastasios Tefas, Ioannis Pitas. The discriminant elastic graph matching algorithm applied to frontal face verification. 2007.
- [33] S. Theodoridis, K. Koutroumbas. *Pattern Recognition*, 3d Ed. Academic Press 2006.