



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ
ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ**

**ΣΥΣΤΗΜΑ ΑΝΑΓΝΩΡΙΣΗΣ ΠΡΟΣΩΠΟΥ ΓΙΑ ΕΛΕΓΧΟ
ΠΡΟΣΒΑΣΗΣ ΣΕ ΦΥΣΙΚΟ ΧΩΡΟ**

ΓΚΙΚΑ ΓΕΩΡΓΙΑ (ΑΜ: 2800)

ΕΠΙΒΛΕΠΩΝ: Ζαχαράκης Ιωάννης, Καθηγητής

ΠΑΤΡΑ 2021

Εγκρίθηκε από την εξεταστική επιτροπή

Πάτρα, 15/2/2021

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

- I. Ιωάννης Ζαχαράκης
- II. Βασίλειος Ταμπακάς
- III. Παναγιώτης Αλεφραγκής

Υπεύθυνη Δήλωση Φοιτητή

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τη συγκεκριμένη εργασία. Η έγκριση της διπλωματικής εργασίας από το Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος. Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία της φοιτήτριας, Γεωργίας Γκίκα που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης ο συγγραφέας/δημιουργός εκχωρεί στο Πανεπιστήμιο Πελοποννήσου, μη αποκλειστική άδεια χρήση του δικαιώματος αναπαραγωγής, προσαρμογής, δημόσιου δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσής τους διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος και για όλο το χρόνο διάρκειας των δικαιωμάτων πνευματικής ιδιοκτησίας. Η ανοικτή πρόσβαση στο πλήρες κείμενο για μελέτη και ανάγνωση δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, αποθήκευση, πώληση, εμπορική χρήση, μετάδοση, διανομή, έκδοση, εκτέλεση, «μεταφόρτωση» (downloading), «ανάρτηση» (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού. Ο συγγραφέας/δημιουργός διατηρεί το σύνολο ηθικών και περιουσιακών του δικαιωμάτων.

ΠΙΝΑΚΑΣ ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ	5
ΠΕΡΙΛΗΨΗ	6
I. ΚΕΦΑΛΑΙΟ 1	7
1. ΕΙΣΑΓΩΓΗ	7
a. Τι είναι Μηχανική Μάθηση και που χρησιμοποιείται	7
b. Παραδείγματα Μηχανικής Μάθησης	7
II. ΚΕΦΑΛΑΙΟ 2	10
1. ΕΞΕΛΙΞΗ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΤΑ ΠΡΟΒΛΗΜΑΤΑ ΑΥΤΗΣ	10
b. Εξέλιξη Τεχνολογίας	10
c. Βήματα Αναγνώρισης Προσώπου	10
d. Προβλήματα στον τομέα Αναγνώρισης Προσώπου	11
III. ΚΕΦΑΛΑΙΟ 3	13
1. ΕΡΕΥΝΑ ΑΛΓΟΡΙΘΜΩΝ	13
a. Είδη Μοντέλων	13
i. Αλγόριθμοι βασισμένοι στην εμφάνιση	14
ii. Principle Component Analysis	14
iii. Linear Discriminant Analysis	18
iii. Fast Independent Component Analysis	19
iv. Kernel Algorithms	22
v. Kernel Principal Component Analysis	23
vi. Kernel Linear Discriminant Analysis	26

2. ΑΛΓΟΡΙΘΜΟΙ ΒΑΣΙΖΟΜΕΝΟΙ ΣΕ ΜΟΝΤΕΛΑ ΔΥΟ ΚΑΙ ΤΡΙΩΝ ΔΙΑΣΤΑΣΕΩΝ	27
i. Hidden Markov Models	27
ii. Ταίριασμα Ελαστικών Δεσμών	30
iii. Τρισδιάστατο Μοντέλο Μορφοποίησης	31
3. ΕΠΙΛΕΓΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ	34
IV. ΚΕΦΑΛΑΙΟ 4	35
1. ΤΟ OpenCV ΚΑΙ Ο ΕΠΙΛΕΓΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ	35
a. OpenCV	35
2. Eigenfaces	37
3. ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ HAAR CASCADE	38
V. ΚΕΦΑΛΑΙΟ 5	41
1. ΥΛΙΚΑ ΚΑΙ ΛΟΓΙΣΜΙΚΟ	41
a. Raspberry Pi και PiNoIR	41
b. Υπόλοιπα Υλικά	43
3. ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ	46
d. Λειτουργικό Σύστημα	46
VI. ΚΕΦΑΛΑΙΟ 6	47
1. ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ	47
a. Εγκατάσταση κάμερας	47
b. Εγκατάσταση Βιβλιοθήκης OpenCV	49

2. ΣΥΝΔΕΣΕΙΣ	53
3. ΔΟΚΙΜΗ ΣΥΣΤΗΜΑΤΟΣ	54
VII. ΚΕΦΑΛΑΙΟ 7	61
1. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ ΠΕΡΑΙΤΕΡΩ ΑΝΑΠΤΥΞΗΣ	61
VIII. ΠΑΡΑΡΤΗΜΑ	62
1. ΚΩΔΙΚΑΣ ΡΥΘΜΩΝ	62
i. Face Recognition	62
ii. Training Recognizer	64
iii. Recognizer	67
2. ΕΙΚΟΝΕΣ	70
3. ΟΡΟΙ	73
IX. ΒΙΒΛΙΟΓΡΑΦΙΑ	74

ΠΙΝΑΚΑΣ ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ

<i>A.I.</i>	<i>Artificial Intelligence</i>
<i>CNN</i>	<i>Convolutional Neural Networks</i>
<i>HMM</i>	<i>Hidden Markov Model</i>
<i>OpenCV</i>	<i>Open source Computer Vision</i>
<i>ONNX</i>	<i>Open Neural Network Exchange</i>
<i>EBM</i>	<i>Elastic Bunch Matching</i>
<i>PCA</i>	<i>Principal Component Analysis</i>
<i>LDA</i>	<i>Linear Discriminant Analysis</i>
<i>IoT</i>	<i>Internet of Things</i>
<i>ICA</i>	<i>Independent Component Analysis</i>
<i>FastICA</i>	<i>Fast Independent Component Analysis</i>

ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική εργασία μελετώνται μοντέλα μηχανικής μάθησης για την επιλογή του καταλληλότερου, με στόχο την αναγνώριση εικόνας δύο διαστάσεων σε πραγματικό χρόνο. Στο πλαίσιο αυτό, η παρούσα εργασία εξετάζει κάποια μηχανικά μοντέλα μάθησης και επιλέγει το προτιμότερο για την υλοποίηση συστήματος αναγνώρισης προσώπου από λήψη κάμερας σε πλακέτα Raspberry Pi και μακέτα συστήματος ελέγχου ασφαλείας για πρόσβαση σε φυσικό χώρο.

Σκοπός λοιπόν αυτής της διπλωματικής εργασίας είναι η παροχή τρόπου και μέσων που χρειάζονται για την υλοποίηση ενός μικρού συστήματος ασφαλείας, χρησιμοποιώντας φθηνά υλικά και τεχνολογίες ανοιχτού κώδικα.

In the present dissertation, machine learning models are studied for the selection of the most appropriate one with the aim of recognising a two-dimensional image in real time. In this context, this dissertation examines some machine learning models and selects the best one for implementation of face recognition from a camera, on a Raspberry Pi board and a security control system for access to a physical space.

The purpose is to provide the means and tools needed to implement a small security system, using low cost materials and open source technologies.

I. ΚΕΦΑΛΑΙΟ 1

1. ΕΙΣΑΓΩΓΗ

a. Τι είναι Μηχανική Μάθηση και που χρησιμοποιείται

Η Τεχνητή Νοημοσύνη (AI) υπάρχει πλέον παντού γύρω μας για πολλές χρήσεις. Αν και δεν είναι νέο επιστημονικό πεδίο, τα τελευταία χρόνια διαπιστώνεται έντονο ενδιαφέρον με ανοδική τάση. Μία αιτία αυτής της εξέλιξης οφείλεται στην ανάγκη εξαγωγής συμπερασμάτων από τον μεγάλο όγκο δεδομένων που διατίθενται στο Διαδίκτυο με μεθόδους της Μηχανικής Μάθησης (Machine Learning). Η μηχανική μάθηση είναι μέθοδος που χρησιμοποιείται για την επινόηση πολύπλοκων μοντέλων και αλγορίθμων που οδηγούν στην πρόβλεψη. Με τη βοήθειά της, δίνουμε στους υπολογιστές την ικανότητα να μαθαίνουν. Πολλές φορές η μηχανική μάθηση και η εξόρυξη δεδομένων (data mining) συγχέονται όμως, η εξόρυξη δεδομένων αφορά την ανακάλυψη ιδιοτήτων μη γνωστών, ενώ η μηχανική μάθηση βασίζεται σε γνωστές ιδιότητες.

Για να καταλάβουμε καλύτερα τι είναι μηχανική μάθηση θα πρέπει να κατανοήσουμε πως δρα ο ανθρώπινος νους. Μία προσέγγιση αναφέρεται στον τρόπο που ένας άνθρωπος μαθαίνει δηλαδή, παρατηρώντας πληροφορίες από το περιβάλλον του και εξάγοντας μέσα από αυτές συγκεκριμένα συμπεράσματα.

Η μηχανική μάθηση λοιπόν, βασίζεται σε μια όμοια λογική. Αντί να προγραμματίζουμε κάθε πιθανή παράμετρο, έχουν δημιουργηθεί αλγόριθμοι που επεξεργάζονται ένα σύνολο πληροφοριών. Βάσει αυτών των πληροφοριών, ο αλγόριθμος ύστερα κατασκευάζει τη δική του λογική.

b. Παραδείγματα Μηχανικής Μάθησης

Η Μηχανική Μάθηση μπορεί να χρησιμοποιηθεί σε πολλούς τομείς. Κάποια παραδείγματα χρήσης Μηχανικής Μάθησης είναι τα παρακάτω:

- Αναγνώριση ομιλίας
- Ιατρική Διάγνωση

- Λογισμικά
- Βελτιστοποίηση
- Μετακίνηση
- Χρηματοπιστηριακή ανάλυση
- Marketing
- Οικονομία
- Βιοπληροφορική
- Χημειοπληροφορική και άλλα

Βλέπουμε ότι τα πεδία στα οποία αξιοποιείται η Μηχανική Μάθηση ποικίλουν και αφορούν και τη καθημερινότητά μας. Ενώ 10 χρόνια πριν η Τεχνητή Νοημοσύνη φαινόταν ως επιστημονική φαντασία, πλέον μπορεί να μην απέχουμε πολύ από ένα AI που να σκέφτεται σαν άνθρωπος.

Πώς λοιπόν μέσω της μηχανικής μάθησης γίνεται να δημιουργηθεί ένα σύγχρονο σύστημα ασφαλείας που να χρησιμοποιεί αναγνώριση προσώπου; Το πρόβλημα που αυτή η διπλωματική εργασία προσπαθεί να αντιμετωπίσει δεν είναι μόνο το πως λειτουργεί αυτό το σύστημα αλλά επίσης το πόσο εύκολα και γρήγορα μπορεί να υλοποιηθεί με υλικά που μπορούν να βρεθούν από το ευρύ κοινό σε ένα χαμηλό χρηματικό κόστος. Όπως φαίνεται και παρακάτω, θα υπάρξουν προβλήματα κατά την υλοποίηση του συστήματος αλλά η λύση τους είναι απλή οπότε, το αποτέλεσμα αυτής της εργασίας είναι ένα ολοκληρωμένο σύστημα αναγνώρισης προσώπου των 2 διαστάσεων.

Εκτός όμως από τα υλικά το μεγαλύτερο μέρος των κεφαλαίων αυτής της εργασίας αφορά την αναγνώριση προσώπου. Το ανθρώπινο πρόσωπο μπορεί να μην είναι μοναδικό (όπως τα δακτυλικά αποτυπώματα) μπορεί όμως να καταγραφεί. Ένα σύστημα αναγνώρισης προσώπου χαρτογραφεί τα χαρακτηριστικά του προσώπου από μια εικόνα ή βίντεο χρησιμοποιώντας βιομετρικά² στοιχεία. Ύστερα αναζητά και συγκρίνει τις πληροφορίες με μια βάση δεδομένων που περιέχει γνωστά πρόσωπα έτσι ώστε να βρεθεί μια αντιστοίχιση. Φυσικά αν και ένα σύστημα αναγνώρισης προσώπου μπορεί να

² Βιομετρία είναι επιστήμη που μέσω στατιστικής ανάλυσης των βιολογικών δεδομένων

βοηθήσει στην επαλήθευση ταυτότητας, μαζί με αυτό το πλεονέκτημα έρχονται και πολλά προβλήματα όπως ζητήματα απορρήτου.

Στα παρακάτω 3 κεφάλαια αναλύονται αλγόριθμοι που αφορούν τη μηχανική μάθηση και την αναγνώριση προσώπου, ενώ στα τελευταία 3 κεφάλαια γίνεται η υλοποίηση του συστήματος, η εκτέλεση του και τα αποτελέσματα αυτής.

II. ΚΕΦΑΛΑΙΟ 2

1. ΕΞΕΛΙΞΗ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΤΑ ΠΡΟΒΛΗΜΑΤΑ ΑΥΤΗΣ

b. Εξέλιξη Τεχνολογίας

Οι τεχνικές που χρησιμοποιούνται για την αναγνώριση προσώπου, έχουν αλλάξει δραματικά τα τελευταία χρόνια. Με βάση τα άρθρα των [1] και [2], έχουν γίνει αλλαγές από χειροκίνητες τεχνικές (hand engineered features) σε τεχνικές μάθησης και από σύνολα δεδομένων (datasets) με χιλιάδες εικόνες σε σύνολα δεδομένων με εκατομμύρια εικόνες.

Αν και έγινε αλλαγή από χειροκίνητες συσκευές σε βαθιά συγκεντρωτικά νευρωνικά δίκτυα (Convolutional Neural Networks/CNN), όλοι οι σύγχρονοι (state-of-the-art) μέθοδοι συνεχίζουν να χρησιμοποιούν συγκεκριμένες τεχνικές ευθυγράμμισης προσώπου (face-specific alignment) αντί για τεχνικές «προσαρμογής» (rough centering features) [3]. Αυτό κυμαίνεται από τεχνικές προσαρμογής σε χρήσεις 3D μάσκας προσώπου και 2D προβολές σύμφωνα και με το άρθρο του T. Hassner και άλλων[4].

c. Βήματα Αναγνώρισης Προσώπου

Μπορεί ο ανθρώπινος εγκέφαλος να αναγνωρίζει με αποτελεσματικότητα πρόσωπα, διότι έχει εξοικειωθεί με τα χαρακτηριστικά του προσώπου - μάτια, μύτη, στόμα - αλλά πώς γίνεται να αναγνωρίζει πρόσωπα ένα σύστημα; Με τον ίδιο ακριβώς τρόπο. Έτσι λειτουργεί και ένα σύστημα αναγνώρισης προσώπου αλλά σε μια μεγαλύτερη αλγοριθμική κλίμακα. Όταν ο άνθρωπος βλέπει πρόσωπα, η τεχνολογία βλέπει δεδομένα τα οποία και αποθηκεύει. Πώς λοιπόν γίνεται η αναγνώριση προσώπου; Αν και υπάρχουν πλέον διάφορες τεχνολογίες σε αυτόν το τομέα, παρακάτω βλέπουμε τα βασικά βήματα:

Βήμα 1: Καταγράφεται μια εικόνα ή ένα βίντεο του προσώπου. Η εικόνα μπορεί να είναι σχεδόν προφίλ ή να κοιτάει ευθέως την κάμερα.

Βήμα 2: Το λογισμικό αναγνώρισης διαβάζει τη γεωμετρία του προσώπου. Βασικοί συντελεστές είναι η απόσταση των ματιών μεταξύ τους και η απόσταση του μετώπου από το πιγούνι μας. Το λογισμικό εντοπίζει σημεία αναφοράς του προσώπου - 68 από αυτά - τα οποία είναι «κλειδί» για την διάκριση του προσώπου. Αποτέλεσμα: δημιουργήθηκε ένα, μοναδικό αποτύπωμα προσώπου (faceprint).

Βήμα 3: Το faceprint - το οποίο είναι έμμεσα μια μαθηματικά φόρμα - συγκρίνεται με μια βάση δεδομένων που περιέχει γνωστά πρόσωπα.

Βήμα 4: Γίνεται προσδιορισμός αν το faceprint ταιριάζει με κάποια εικόνα από τη βάση δεδομένων του συστήματος.

d. Προβλήματα στον τομέα Αναγνώρισης Προσώπου

Ένα σύστημα αναγνώρισης προσώπου χρησιμοποιεί βιομετρικά στοιχεία για τη χαρτογράφηση χαρακτηριστικών προσώπου από μια φωτογραφία ή από ένα βίντεο. Ύστερα, συγκρίνει τις πληροφορίες που έλαβε με τα γνωστά πρόσωπα που περιέχει μια βάση δεδομένων, έτσι ώστε να βρει ταιριαστό πρόσωπο. Αν και η αναγνώριση προσώπου μπορεί να βοηθήσει στην επαλήθευση της προσωπικής ταυτότητας, αυτό προφανώς σημαίνει ότι δημιουργεί και προβλήματα στην ιδιωτική ζωή. Κάποια από αυτά μπορεί να είναι τα παρακάτω.

- ▶ Προστασία απορρήτου: Αφορά τα δικαιώματα που έχουμε για τον έλεγχο των προσωπικών μας δεδομένων, καθώς και το αποτύπωμα προσώπου μας.
- ▶ Ασφάλεια των δεδομένων μας (*Security*): Τα στοιχεία του προσώπου μας μπορούν να συλλεχθούν και να αποθηκευτούν χωρίς την άδειά μας. Με άλλα λόγια ένας κακόβουλος θα μπορούσε να κλέψει τα δεδομένα μας.

- ▶ *Ασφάλεια (Safety)*: Η αναγνώριση προσώπου μπορεί να οδηγήσει σε διαδικτυακή παρενόχληση και καταδίωξη. Για παράδειγμα, κάποιος παίρνει την εικόνα μας σε ένα μετρό ή σε κάποιο άλλο δημόσιο μέρος και χρησιμοποιεί λογισμικό αναγνώρισης προσώπου για να μάθει ποιος είναι.
- ▶ *Ιδιωτικότητα*: Μπορεί το πρόσωπό μας να μας ανήκει όμως, η ψηφιακή εικόνα του προσώπου μας διαφέρει. Μπορεί εύκολα να συμφωνήσαμε σε όρους κάποιων μέσων κοινωνικής δικτύωσης, ότι μπορούν να έχουν τα δεδομένα μας ως δική τους ιδιοκτησία.

Αν τα πράγματα είναι έτσι όμως, τότε γιατί η αναγνώριση προσώπου είναι χρήσιμη [5]; Η αναγνώριση προσώπου είναι μια πολύ ισχυρή τεχνολογία που θα πρέπει να χρησιμοποιείται με προσοχή. Από τη μία πλευρά, φέρνει τεράστια πλεονεκτήματα στις εταιρείες και τους χρήστες, αφού τους βοηθά να ενισχύσουν την ασφάλειά τους και να εντοπίσουν τους παραβάτες αλλά από την άλλη πλευρά, μπορεί να γίνει κατάχρηση για ιδιωτικό όφελος και να οδηγήσει σε σοβαρές συνέπειες.

Θα πάρει αρκετό χρόνο όμως για την αναγνώριση του προσώπου να ανταποκρίνεται πλήρως στα ανθρώπινα δικαιώματα και στην ιδιωτικότητα. Μέχρι τότε, μπορούμε να ελπίζουμε πως οι μεγάλες βιομηχανίες, θα κάνουν αυτή την τεχνολογία πιο φιλική προς το χρήστη και ασφαλή ώστε όλοι να μπορούν να επωφεληθούν από αυτήν.

III. ΚΕΦΑΛΑΙΟ 3

1. ΕΡΕΥΝΑ ΑΛΓΟΡΙΘΜΩΝ

Το παρακάτω κεφάλαιο αναλύει αλγόριθμους αναγνώρισης προσώπου, που έχουν ήδη μελετηθεί, με στόχο την επιλογή κατάλληλου για την υλοποίηση του συστήματος ασφαλείας.

a. Είδη Μοντέλων

Ένα μεγάλο μέρος αυτής της διπλωματικής, βασίζεται στη μελέτη κάποιων γνωστών αλγορίθμων αναγνώρισης προσώπου [6] και να επιλεγθεί αυτός που θα αρμόζει καλύτερα. Η επιλογή του κατάλληλου αλγορίθμου θα γίνει με βάση κάποια κριτήρια που διατυπώνονται παρακάτω.

Οι αλγόριθμοι αναγνώρισης προσώπου μπορούν να κατηγοριοποιηθούν βάσει δύο κύριων προσεγγίσεων:

- i) Στους αλγόριθμους που βασίζονται στην εμφάνιση (appearance based)· χωρίζονται σε γραμμικούς και μη-γραμμικούς.
- ii) Στους αλγόριθμους που βασίζονται στη δημιουργία μοντέλων (model based) ικανών να περιγράψουν το ανθρώπινο πρόσωπο· χωρίζονται σε 2D και 3D διαστάσεων.

Για την εκπλήρωση των στόχων της εργασίας λοιπόν, μελετήθηκαν οι σημαντικότεροι αλγόριθμοι που βασίζονται στην εμφάνιση και σε μοντέλα. Συγκεκριμένα, παρακάτω παρουσιάζεται η αναλυτική περιγραφή των ακόλουθων αλγορίθμων: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), και αλγορίθμων πυρήνα (Kernel). Αναφέρονται επίσης και οι αλγόριθμοι Hidden Markov Models (HMM), Elastic Bunch Matching (EBM) και 3D Morphable Model που βασίζονται σε μοντέλα. Η απόδοση των αλγορίθμων είναι άμεσα συνδεδεμένη με την βάση προσώπων στην οποία εφαρμόζονται.

i. Αλγόριθμοι βασισμένοι στην εμφάνιση

Οι περισσότερες προσεγγίσεις για face recognition βασίζονται σε εικόνες χωρίς τη βοήθεια γεωμετρικών μοντέλων. Πολλές από αυτές τις τεχνικές εξαρτώνται από την αναπαράσταση των εικόνων και δημιουργούν διανυσματικούς χώρους προσώπων. Κάθε εικόνα $m * n$ μπορεί να θεωρηθεί ως ένα διάνυσμα στο \mathbb{R}^{mn} στον οποίο χώρο αυτό κάθε εικόνα-πρόσωπο αντιπροσωπεύεται από ένα σημείο.

Πολλές μέθοδοι εφαρμόζουν στατιστικές μεθόδους για τη ανάλυση της κατανομής των διανυσμάτων των εικόνων στον χώρο προσώπων καθώς παράγουν και μια αναπαράσταση (διάνυσμα χαρακτηριστικών) ικανή να περιγράψει το κάθε πρόσωπο. Για όλες τις εικόνες ελέγχου δημιουργείται ένα διάνυσμα χαρακτηριστικών• ύστερα συγκρίνεται με τα διανύσματα των εικόνων που δώσαμε αρχικά για την εκπαίδευση του μοντέλου. Η χρήση της διανυσματικής αναπαράστασης δίνει τη δυνατότητα χρήσης τεχνικών εκμάθησης για την εκπαίδευση των συστημάτων αναγνώρισης.

Σύμφωνα με μελέτη του 1991 οι M. Turk και A. Pentland [7] με την χρήση των ιδιοδιανυσμάτων, ανέπτυσαν ένα τρόπο προσδιορισμού της ύπαρξης ή όχι ενός προσώπου σε μία εικόνα. Αυτό, οδήγησε στη δημιουργία συστημάτων αυτόματης αναγνώρισης προσώπου σε πραγματικό χρόνο. Αν και η προσέγγιση αυτή είχε αρκετούς περιορισμούς, αύξησε το ενδιαφέρον της επιστημονικής κοινότητας για την δημιουργία αυτοματοποιημένων συστημάτων αναγνώρισης.

ii. Principle Component Analysis

Η ανάλυση σε Κύριες Συνιστώσες (Principle Component Analysis - PCA), χρησιμοποιείται ευρέως σε πληθώρα επιστημονικών εφαρμογών, από την νευροφυσιολογία ως και τα γραφικά υπολογιστών για τη μείωση των διαστάσεων του μεγάλου συνόλου δεδομένων. Αποτελεί εφαρμογή του μετασχηματισμού Karhunen-Loeve [8]. Είναι μέθοδος που χρησιμοποιείται ως

στατιστική προσέγγιση αναγνώρισης μοτίβων. Μια 2D εικόνα προσώπου με μέγεθος $N \times N$ μπορεί να θεωρηθεί ένα μονοδιάστατο διάνυσμα με διάσταση N^2 . Ένα σύνολο εικόνων έχει τη δυνατότητα να χαρτογραφηθεί ως σύνολο σημείων στον πολυδιάστατο χώρο. Οι εικόνες προσώπων έχουν παρόμοιες συνθέσεις, οπότε δεν κατανέμονται τυχαία στον χώρο αυτό καθώς και εμπίπτουν χωροταξικά σε ένα μικρό κομμάτι του.

Κεντρική ιδέα των Κύριων Συνιστωσών (Principal Components) είναι η εύρεση διανυσμάτων των οποίων οι γραμμικοί συνδυασμοί περιγράφουν ικανοποιητικά τις κατανομές των εικόνων προσώπων και ορίζουν ένα υποσύνολο που ονομάζεται «χώρος προσώπων» (face space). Κάθε από αυτά τα διανύσματα μεγέθους N^2 περιγράφει μια εικόνα $N \times N$.

Συνοπτικά, ο PCA μειώνει την υπολογιστική πολυπλοκότητα του μοντέλου οπότε κάνει τους αλγόριθμους μηχανικής μάθησης να τρέχουν γρηγορότερα. Βέβαια ένα μεγάλο ερώτημα είναι πόση ακρίβεια «θυσιάζει» για να έχει λιγότερο σύνθετες και μειωμένες διστάσεις δεδομένων· δεν υπάρχει ακριβής απάντηση σε αυτό το ερώτημα, ωστόσο προσπαθούν να διατηρήσουν το μεγαλύτερο μέρος της διακύμανσης κατά την επιλογή του τελικού συνόλου στοιχείων.

Παρακάτω, είναι η μαθηματική ανάλυση που προτάθηκε από τον Αθανάσιο Μπακάλη [9]:

- 1) Έστω ένα σύνολο εικόνων προσώπων $\Gamma_1, \dots, \Gamma_M$ σε μορφή διανυσμάτων. Ο μέσος όρος του συνόλου ορίζεται από την σχέση:

$$\Psi = \frac{1}{M} \sum_{\nu=1}^M \Gamma_{\nu} \quad (1)$$

- 2) Κάθε πρόσωπο διαφέρει από το μέσο όρο του συνόλου βάση του διανύσματος:

$$\Phi_i = \Gamma_i - \Psi \quad (2)$$

- 3) Το σύνολο των διανυσμάτων είναι αντικείμενο εφαρμογής της Ανάλυσης Κύριων Συνιστωσών (που αναζητά ένα υποσύνολο M ορθοκανονικών διανυσμάτων) και περιγράφει με τον καλύτερο τρόπο την κατανομή της πληροφορίας των προσώπων. Το k -οστό διάνυσμα U_k επιλέγεται έτσι ώστε να μεγιστοποιείται η ποσότητα:

$$\lambda_k = \frac{1}{M} \sum_{\nu=1}^M \left(U_k^T \Phi_{\nu} \right) \quad (3)$$

υπό τον περιορισμό ότι:

$$U_l^T U_k = \delta_{lk} = \begin{cases} 1 & \text{εάν } l = k \\ 0 & \text{εάν } l \neq k \end{cases} \quad (4)$$

- 4) Τα διανύσματα U_k και οι τιμές λ_k είναι τα ιδιοδιανύσματα και οι ιδιοτιμές του πίνακα συνδιακύμανσης:

$$C = \frac{1}{M} \sum_{\nu=1}^M \Phi_{\nu} \Phi_{\nu}^T = A A^T \quad (5)$$

Όπου $A = [\Phi_1, \Phi_2, \dots, \Phi_M]$

- 5) Τη παρακάτω διαδικασία χρησιμοποιούμε όταν ο πίνακας διακύμανσης C (ο οποίος είναι ένας $N^2 \times N^2$ πίνακας με πραγματικά στοιχεία) και ο υπολογισμός των N^2 ιδιοδιανυσμάτων είναι αυξημένης πολυπλοκότητας για τυπικά μεγέθη εικόνων.

Έστω ιδιοδιανύσματα ν_i του $A^T A$ τέτοια ώστε:

$$A^T A \nu_i = \mu_i \nu_i \quad (6)$$

Πολλαπλασιάζοντας και τα δύο μέρη με το A έχουμε:

$$A A^T A \nu_i = A \mu_i \nu_i \quad (7)$$

Έστω $A \nu_i = Q$,

$$A A^T Q = \mu_i Q \quad (8)$$

όπου $Q = A \nu_i$ είναι τα ιδιοδιανύσματα και μ_i οι ιδιοτιμές του C .

- 6) Βάσει των παραπάνω, αρκεί να δημιουργηθεί ένας πίνακας $M \times M$ $L = A^T A$, όπου $L_{mn} = \Phi_m^T \Phi_n$, και να βρεθούν τα M ιδιοδιανύσματα ν_i του L . Για τη δημιουργία των ιδιοπροσώπων U_I , τα παραπάνω διανύσματα καθορίζουν γραμμικούς συνδυασμούς των M εικόνων προσώπων:

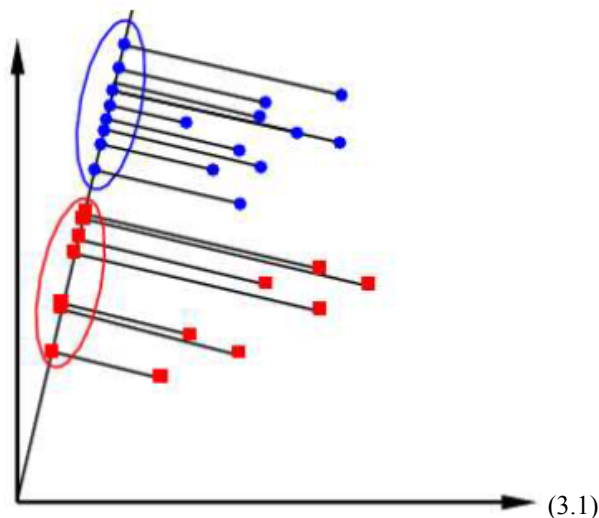
$$U_I = \sum_{k=1}^M \nu_{Ik} \Phi_k, I = 1, \dots, M \quad (9)$$

Με τη παραπάνω ανάλυση οι υπολογισμοί μειώνονται, από N^2 δηλαδή τον βαθμό του αριθμού των εικονοστοιχείων των εικόνων, στον βαθμό του αριθμού των εικόνων του συνόλου M .

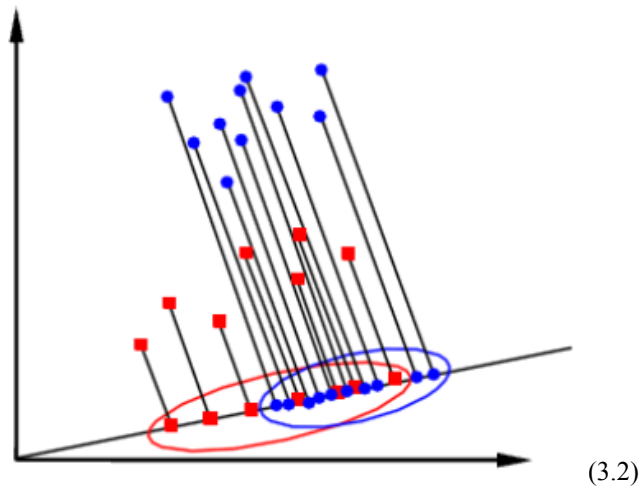
iii. Linear Discriminant Analysis

Η Γραμμική Διακρίνουσα Ανάλυση (Linear Discriminant Analysis - LDA) - ή αλλιώς η μέθοδος των Fisherfaces - είναι μέθοδος που στοχεύει στην υψηλή διαχωριστικότητα μεταξύ των πολλαπλών μοτίβων που πρέπει να ταξινομηθούν.

Η γραμμική διακρίνουσα του Fisher ομαδοποιεί τις εικόνες που ανήκουν στην ίδια κλάση ενώ ταυτόχρονα διαχωρίζει αυτές που ανήκουν σε διαφορετικές κλάσεις. Οι εικόνες προβάλλονται από το διδιάστατο χώρο σε ένα χώρο M διαστάσεων, όπου M ο αριθμός των διαφορετικών ατόμων στη βάση των προσώπων. Έστω ότι δύο σύνολα σημείων του 2D χώρου προβάλλονται σε μια ευθεία από την κλίση τους ευθείας τους εξαρτάται αν τα σημεία θα διαχωρίζονται πλήρως (εικόνα 3.1) ή θα υπάρχουν κάποια που θα συμπίπτουν (εικόνα 3.2).



Εικόνα 3.1: Σημεία που διαχωρίζονται πλήρως σε δύο κλάσεις κατά την προβολή τους σε ευθεία.



Εικόνα 3.2: Σημεία που συμπίπτουν χωρίς να είναι εφικτός ο ξεκάθαρος διαχωρισμός τους σε κλάσεις όταν προβάλλονται σε ευθεία.

Ο στόχος της γραμμικής διακρίνουσας του Fisher είναι να βρει την ευθεία που διαχωρίζει τα σημεία με το βέλτιστο δυνατό τρόπο. Η αναγνώριση μιας εικόνας όμως, γίνεται συγκρίνοντας την ίδια με κάθε μια από τις εικόνες του συνόλου εκπαίδευσης (training set). Ύστερα, η εικόνα προς αναγνώριση θεωρείται «όμοια» με την εικόνα του συνόλου εκπαίδευσης που παρουσιάζει τη μικρότερη απόσταση μεταξύ των προβολών τους.

Πριν όμως αναλύσουμε τον αλγόριθμο LDA, θα ήταν καλό να αναφερθούμε και στον Naive Bayes διότι αποτελεί τη βάση για το LDA. Και οι δύο είναι γραμμικοί αλγόριθμοι ταξινόμησης και υπάγονται στην κατηγορία των Γενετικών Μοντέλων. Ο LDA χρησιμοποιεί τη κατανομή Gauss και υποθέτει ότι υπάρχουν ίσες διακυμάνσεις ενώ ο NB επίσης χρησιμοποιεί τη κατανομή Gauss αλλά υποθέτει ότι οι μεταβλητές είναι ανεξάρτητες. Συνοπτικά, ο ταξινομητής LDA είναι ένας βέλτιστος NB (αγνοώντας οποιοδήποτε σφάλμα εκτίμησης) εάν οι κατανομές που αντιστοιχούν στις δύο κατηγορίες είναι του Gauss και έχουν ίση συνδιακύμανση.

iii. Fast Independent Component Analysis

Ο αλγόριθμος ICA εφαρμόζει μία διαδικασία εύρεσης ενός γραμμικού μετασχηματισμού που μειώνει την στατιστική εξάρτηση ανάμεσα στις εξαγόμενες συνιστώσες. Η ανάλυση σε ανεξάρτητες συνιστώσες έχει πολύ

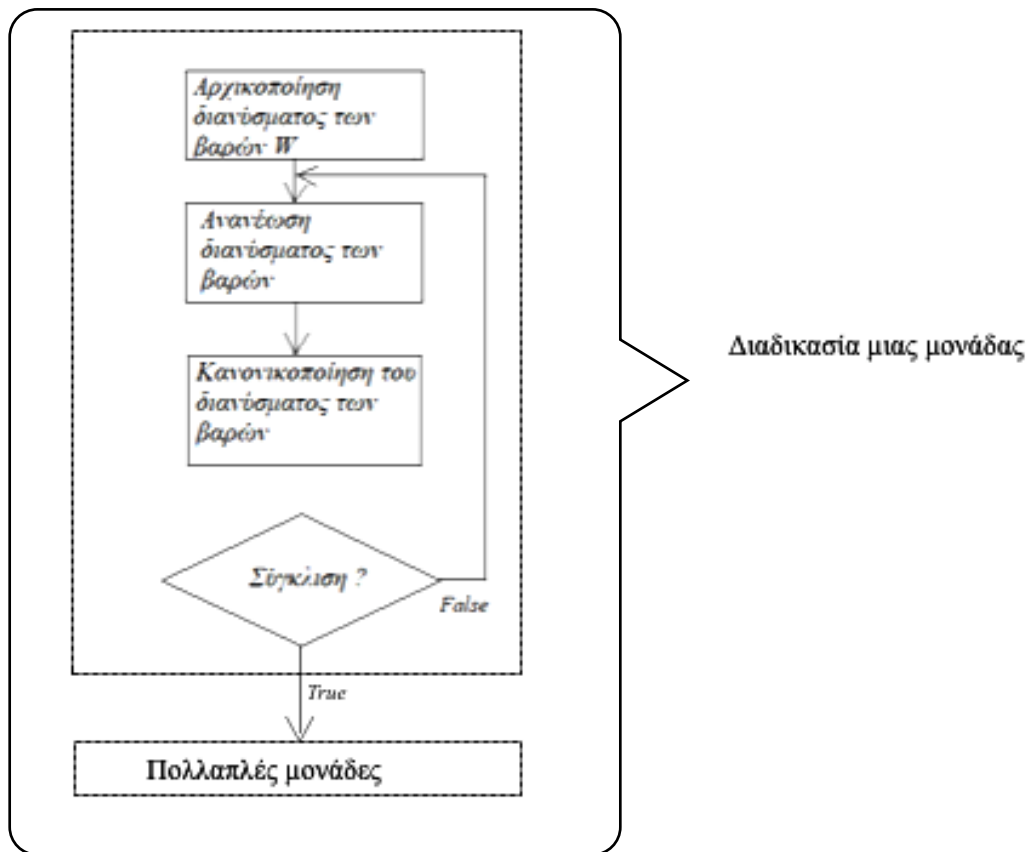
καλή απόδοση όταν εφαρμόζεται για εύρεση κατάλληλης αναπαράστασης των εικόνων, για συμπίεση, για εξαγωγή χαρακτηριστικών, για αναγνώριση μοτίβων και την αναπαράσταση εικόνων.

Ο FastICA είναι αλγόριθμος που ανήκει στην οικογένεια των αλγορίθμων ICA και έχει αρκετά πλεονεκτήματα. Κάποια από αυτά είναι:

- Δεν υπάρχουν παράμετροι που να αφορούν το learning rate του νευρώνα το οποίο κάνει τον FastICA απλό στη χρήση.
- Έχει όλα τα προτερήματα των αλγορίθμων των νευρωνικών δικτύων. Οι υπολογισμοί είναι παράλληλοι, εύκολοι στον υπολογισμό και χρησιμοποιούν μικρά μεγέθη μνήμης.
- Οι ανεξάρτητες συνιστώσες μπορούν να υπολογισθούν ξεχωριστά το οποίο είναι αρκετά χρήσιμο σε ανάλυση δεδομένων διότι ελαχιστοποιεί τους υπολογισμούς.

Αναλυτικά, ο Aaro Hyvärinen [10] είναι ο δημιουργός του FastICA. Η επιλογή λάθους ρυθμού εκμάθησης μπορεί να καταστρέψει την σύγκλιση (ο FastICA έχει σύγκλιση cubic) στην εκπαίδευση του νευρώνα για την δημιουργία του πίνακα των Ανεξάρτητων Συνιστωσών. Όμως, η χρήση ενός πολύ μικρού ρυθμού εκμάθησης επιβραδύνει τη λειτουργία του αλγορίθμου. Για το λόγο αυτό είναι σημαντική η δημιουργία μιας διαδικασίας που να επιταχύνει επιτυχώς και αξιόπιστα την εκπαίδευση του νευρώνα. Ένα παράδειγμα μιας τέτοιας διαδικασίας είναι η χρήση αλγορίθμου που βασίζεται σε σταθερό αριθμό επαναλήψεων.

Σε αυτούς τους αλγορίθμους οι υπολογισμοί γίνονται τμηματικά. Επιπλέον σαν νευρώνες χρησιμοποιούν παράλληλη επεξεργασία, είναι κατανεμημένοι, οι υπολογισμοί τους είναι απλοί και χρειάζονται μικρά μεγέθη μνήμης. Στους fixed-point αλγορίθμους, η είσοδος ακολουθεί μια διαδικασία λευκοποίησης (whitening), δηλαδή της γραμμικής μετατροπής της εισόδου έτσι ώστε ο πίνακας συσχέτισης να ισούται με I . Αρχικά μπορεί να οριστεί ένας αλγόριθμος σταθερού σημείου για μία είσοδο και μία έξοδο. Ο αλγόριθμος με τις επαναλήψεις προσπαθεί να υπολογίσει την διεύθυνση του διανύσματος των βαρών W , για τον οποίο πίνακα -των βαρών- ισχύει:



Στην περίπτωση που έχουμε N εισόδους και N εξόδους επαναλαμβάνουμε την διαδικασία N φορές για κάθε έναν από τους συντελεστές χρησιμοποιώντας n νευρώνες με διανύσματα βαρών $w_1 \dots w_N$. Για την αποφυγή του ενδεχομένου διαφορετικά διανύσματα της εισόδου να συγκλίνουν στο ίδιο μέγιστο πρέπει να αποσυσχετίζονται οι εξοδοί μετά από κάθε επανάληψη. Για την επίτευξη της αποσυσχέτισης μπορούν να εφαρμοστούν διάφορες διαδικασίες μια από τις οποίες είναι και η ακόλουθη διαδικασία με επαναλήψεις του Hyvärinen:

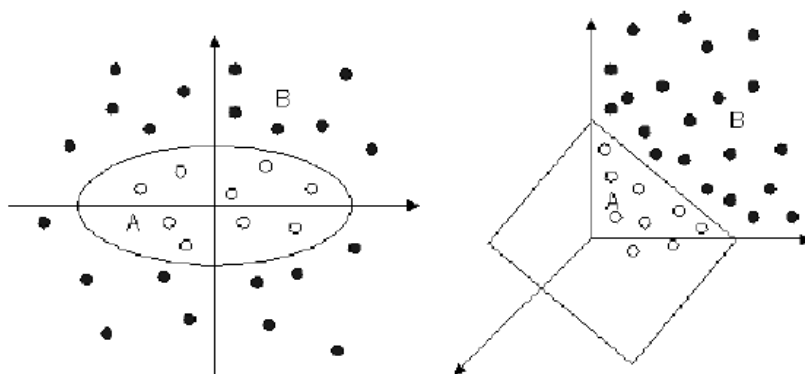
$$W = \frac{w}{\sqrt{\|w w^T\|}}$$

$$W = \frac{3}{2}W - \frac{1}{2}W W^T W$$

Επανάληψη βήματος 2 μέχρι να επιτευχθεί σύγκλιση.

iv. Kernel Algorithms

Οι προσεγγίσεις των αλγορίθμων PCA και LDA κωδικοποιούν πληροφορίες χρησιμοποιώντας εξαρτήσεις δεύτερης τάξεως. Οι εξαρτήσεις υψηλότερης τάξης σε μία εικόνα συμπεριλαμβάνουν μη γραμμικές συσχετίσεις στις τιμές της έντασης των εικονοστοιχείων. Παράδειγμα αποτελούν οι σχέσεις τριών ή περισσοτέρων εικονοστοιχείων που παρατηρούνται σε ακμές ή καμπύλες και εμπεριέχουν χρήσιμη πληροφορία για την αναγνώριση. Η βασική ιδέα των αλγορίθμων του πυρήνα μπορεί να γίνει καλύτερα κατανοητή αν χρησιμοποιήσουμε το ακόλουθο παράδειγμα:



Εικόνα 3.3: Παράδειγμα προβλήματος ταξινόμησης μοτίβων δύο κατηγοριών. (Από αριστερά προς δεξιά). Τα δείγματα βρίσκονται σε δυδιάστατο χώρο εισόδου όπου χρειάζεται ένα μη γραμμικό όριο αποφάσεων ελλειψοειδής μορφής ώστε να χωριστούν οι κατηγορίες A και B. Τα δείγματα αντιστοιχίζονται σε ένα τρισδιάστατο χώρο όπου ένα γραμμικό υπερεπίπεδο³ μπορεί να χωρίσει τις δύο κατηγορίες A και B.

Παραπάνω (εικόνα 3.3) βλέπουμε ότι στο χώρο εισόδου ένα μη γραμμικό ελλειψοειδές όριο απόφασης χρειάζεται για το διαχωρισμό των κλάσεων. Αντίθετα, οι δύο κλάσεις διαχωρίζονται γραμμικά στον υψηλότερης διάστασης χώρο. Προκειμένου να επιτευχθεί ο στόχος του γραμμικού διαχωρισμού, πρέπει η διάσταση του χώρου των χαρακτηριστικών να είναι ιδιαίτερα μεγάλη και πιθανώς άπειρη. Θετικό όμως είναι το γεγονός ότι δεν είναι απαραίτητη η εύρεση του εσωτερικού γινομένου των διανυσμάτων χαρακτηριστικών ($\varphi(N)$) καθώς ο χώρος των χαρακτηριστικών μπορεί να υπολογιστεί έμμεσα με τη

³ Υπερεπίπεδο είναι ένας υποχώρος του οποίου η διάσταση είναι μικρότερη από αυτή του περιβάλλοντος χώρου του.

βοήθεια μηχανών πυρήνα (Kernel Machine). Το τέχνασμα που χρησιμοποιείται σε αυτές τις μεθόδους είναι η αντικατάσταση των εσωτερικών γινομένων στον χώρο των χαρακτηριστικών με μία συνάρτηση πυρήνα στο χώρο εισόδου.

Για να γίνει το παραπάνω κατανοητό θα χρησιμοποιηθεί και πάλι το παράδειγμα της εικόνας 3.3, όπου ο χώρος των χαρακτηριστικών αποτελείται από τα μονώνυμα (το γινόμενο μεταβλητών και σταθερών που κάνουν μια αλγεβρική παράσταση) δεύτερης τάξης του χώρου εισόδου. Έστω $N_i \in \mathbb{R}^2$ και $N_j \in \mathbb{R}^2$ είναι δύο δείγματα του χώρου εισόδου. Το εσωτερικό γινόμενο των διανυσμάτων χαρακτηριστικών $\varphi(N_i) \in F$ και $\varphi(N_j) \in F$ (όπου F είναι ο χώρος εσωτερικών γινομένων - πιθανόν άπειρης διάστασης -) μπορεί να υπολογιστεί με την ακόλουθη συνάρτηση πυρήνα, $k(N_i, N_j)$ που ορίζεται στο $\mathbb{R}^2 \times \mathbb{R}^2$:

$$\varphi(N_i) \cdot \varphi(N_j) = [N_{i1}^2, \sqrt{2}N_{i1}N_{i2}, N_{i2}^2]^T [N_{j1}^2, \sqrt{2}N_{j1}N_{j2}, N_{j2}^2] = ([N_{i1}N_{i2}]^T [N_{j1}N_{j2}])^2 = (N_i \cdot N_j)^2 =: k(N_i, N_j)$$

Από το παράδειγμα αυτό φαίνεται ότι η κεντρική ιδέα για τη γενίκευση ενός αλγορίθμου γραμμικής εκμάθησης στην αντίστοιχη έκδοση του πυρήνα είναι η μετατροπή όλων των υπολογισμών του αλγορίθμου στο χώρο των χαρακτηριστικών έτσι ώστε να χρησιμοποιούνται μόνο εσωτερικά γινόμενα.

v. Kernel Principal Component Analysis

Δοθέντος ενός συνόλου από m κεντρικά (centered: θεωρούνται εκείνα τα δείγματα που έχουν μηδενικό μέσο και μοναδιαία διακύμανση) δείγματα και x κάθε διάνυσμα, $x_k = [x_{k1}, x_{k2}, \dots, x_{km}]^T \in \mathbb{R}^n$, ο αλγόριθμος PCA στοχεύει στην εύρεση εκείνων των διευθύνσεων προβολής που μεγιστοποιούν τη διακύμανση, C , των x_k κάτι που είναι ισοδύναμο με την εύρεση των ιδιοτιμών του πίνακα συνδιακύμανσης των δειγμάτων:

$$\lambda v = Cv \tag{1}$$

όπου $\lambda \geq 0$ οι ιδιοτιμές και $v \in \mathbb{R}^n$ τα ιδιοδιανύσματα.

Στον Kernel PCA, κάθε διάνυσμα x προβάλλεται από τον αρχικό χώρο, σε έναν χώρο χαρακτηριστικών υψηλής τάξης, \mathbb{R}^f το πρόβλημα εύρεσης ιδιοδυνασμάτων ορίζεται ως εξής:

$$\lambda v^\Phi = C^\Phi v^\Phi \quad (2)$$

όπου C^Φ ο πίνακας συνδιακύμανσης. Όλες οι λύσεις v^Φ με $\lambda \neq 0$ λαμβάνουν τιμές στο $\Phi(x_1), \dots, \Phi(x_m)$, και υπάρχουν συντελεστές a_i τέτοιοι ώστε:

$$v^\Phi = \sum_{i=1}^m a_i \Phi(x_i) \quad (3)$$

Θεωρώντας έναν $m \times m$ πίνακα K τέτοιο ώστε:

$$K_{ij} = k(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \quad (4)$$

το πρόβλημα του Kernel PCA μετασχηματίζεται ως εξής:

$$m\lambda K a = K^2 a \quad (5)$$

$$m\lambda a = K a \quad (6)$$

όπου a ένα διάνυσμα στήλη με τιμές a_1, \dots, a_m . Οι παραπάνω σχέσεις βασίζονται στην υπόθεση ότι όλα τα δείγματα $\Phi(x)$ που προβάλλονται είναι κεντρικά (centered) στο \mathbb{R}^f . Σημειώνεται σε αυτό το σημείο ότι ο αλγόριθμος PCA είναι μία ειδική περίπτωση του kernel PCA με πολυωνυμικό πυρήνα πρώτης τάξης.

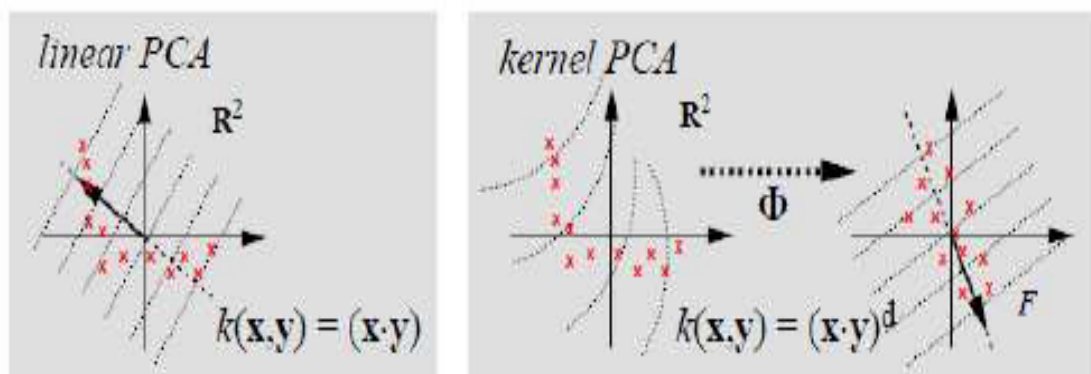
Με άλλα λόγια, ο Kernel PCA είναι μία γενίκευση της τυπικής Ανάλυσης Κύριων Συνιστωσών (PCA) εφόσον διαφορετικοί πυρήνες μπορούν να χρησιμοποιηθούν για να παράγουν διάφορες μη γραμμικές προβολές. Είναι εφικτό λοιπόν να ληφθούν οι προβολές των διανυσμάτων του \mathbb{R}^f σε ένα χώρο μικρότερης διάστασης που ορίζεται από τα ιδιοδιανύσματα v^Φ . Για παράδειγμα έστω ένα δείγμα x του οποίου η προβολή είναι $\Phi(x)$ στο \mathbb{R}^f . Τότε η προβολή του $\Phi(x)$ στα ιδιοδιανύσματα v^Φ είναι οι μη γραμμικές Κύριες Συνιστώσες που αντιστοιχούν στο Φ :

$$v^\Phi \cdot \Phi(x) = \sum_{i=1}^m a_i (\Phi(x_i) \cdot \Phi(x)) = \sum_{i=1}^m a_i k(x_i, x) \quad (7)$$

Για κάποιες επιλογές πυρήνα $k(x, y)$, μπορεί να αποδειχτεί με τη βοήθεια της συναρτησιακής ανάλυσης ότι υπάρχει συνάρτηση αντιστοίχισης Φ σε κάποιο χώρο εσωτερικών γινομένων F έτσι ώστε ο πυρήνας k να μπορεί να υπολογίσει τα εσωτερικά γινόμενα του χώρου F . Οι πυρήνες που επιτυχώς χρησιμοποιήθηκαν σε μηχανές υποστήριξης διανύσματος (Support Vector Machines) συμπεριλαμβάνουν τους πολυωνυμικούς $k(x,y) = (x \cdot y)^d$, τις συναρτήσεις ακτινωτής βάσης $k(x,y) = \exp(-\frac{\|x-y\|^2}{2\sigma^2})$ και τους σιγμοειδής $k(x,y) = \tanh(k(x \cdot y) + \Theta)$. Μπορεί να αποδειχτεί ότι οι πολυωνυμικοί πυρήνες βαθμού d είναι άρρηκτα συνδεδεμένοι με μία αντιστοίχιση Φ στο χώρο των χαρακτηριστικών που αποτελείται από όλα τα εσωτερικά γινόμενα d παραγόντων ενός μοτίβου εισόδου. Για παράδειγμα αν το πλήθος των παραγόντων είναι 2 και ο πολυωνυμικός βαθμός είναι επίσης δύο έχουμε:

$$(x \cdot y)^2 = (x_1^2, x_1x_2, x_2x_1, x_2^2)((y_1^2, y_1y_2, y_2y_1, y_2^2))^T \quad (8)$$

Αν τα μοτίβα εισόδου αντιστοιχούν σε εικόνες, μπορεί να χρησιμοποιηθεί ο χώρος όλων των εσωτερικών γινομένων d εικονοστοιχείων και συνεπώς να ληφθούν υπόψη υψηλότερης τάξης στατιστικές όταν εφαρμόζεται ο αλγόριθμος PCA. Έτσι αν αντικαταστήσουμε όλες τις εμφανίσεις του εσωτερικού γινομένου με συναρτήσεις πυρήνα λαμβάνουμε τον αλγόριθμο Kernel PCA η κύρια ιδέα του οποίου φαίνεται στην εικόνα που ακολουθεί:



Εικόνα 3.4 Βασική Ιδέα του αλγορίθμου Kernel PCA.

Χρησιμοποιώντας μία μη γραμμική συνάρτηση k αντί του τυπικού εσωτερικού γινομένου είναι δυνατή η εφαρμογή του PCA σε ένα χώρο μεγάλης διάστασης που δεν είναι γραμμικά συσχετιζόμενος με τον χώρο εισόδου. Οι διακεκομμένες γραμμές είναι οι ισοϋψείς καμπύλες των σταθερών τιμών των χαρακτηριστικών. Αρχικά, κατά την εφαρμογή του αλγορίθμου kernel PCA, υπολογίζεται ο $m \times m$ πίνακας K (4), έπειτα επιλύεται η (6) μετατρέποντας τον πίνακα σε διαγώνιο (πίνακας με μη μηδενικά στοιχεία μόνο στη διαγώνιο), στη συνέχεια κανονικοποιούνται οι συντελεστές a_i και τέλος εξάγονται οι Κύριες Συνιστώσες (Principal Components) υπολογίζοντας τις προβολές στα ιδιοδιανύσματα με βάση την (7).

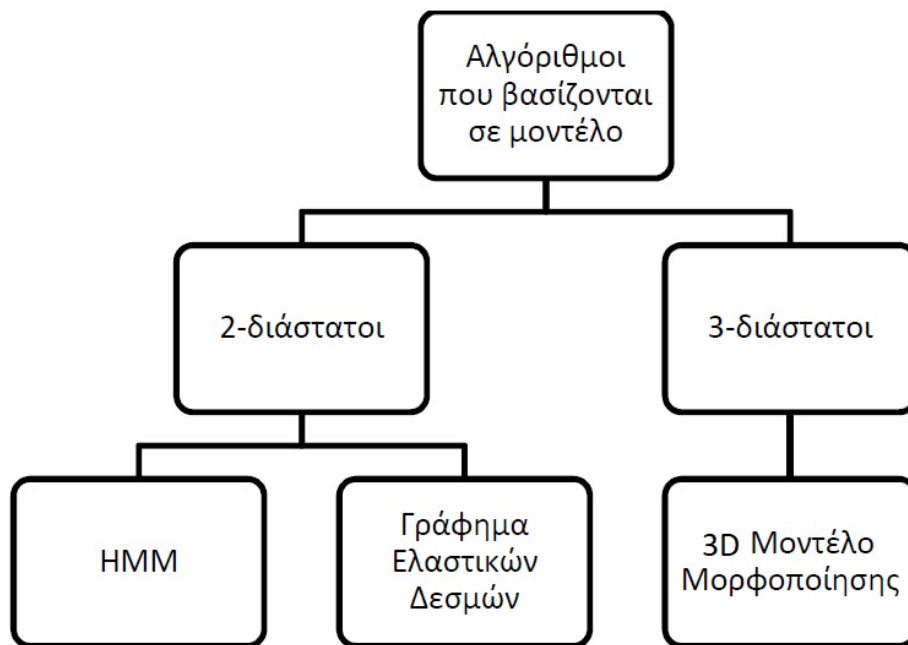
Είναι δυνατό να εξαχθούν οι πρώτες q ($1 \leq q \leq m$) μη γραμμικές Κύριες Συνιστώσες (ιδιοδιανύσματα v^{ϕ}) χρησιμοποιώντας τη συνάρτηση πυρήνα χωρίς να είναι απαραίτητη η εκτέλεση της ιδιαίτερα δαπανηρής πράξης της προβολής των δειγμάτων στον χώρο υψηλής διάστασης \mathbb{R}^l . Οι πρώτες q συνιστώσες αντιστοιχούν στις πρώτες q ιδιοτιμές της $m \times m$ K . Στην αναγνώριση προσώπων όπου κάθε x αναπαριστά μία εικόνα οι εξαγόμενες μη γραμμικές Κύριες Συνιστώσες καλούνται ιδιοπρόσωπα του πυρήνα (Kernel Eigenfaces).

vi. Kernel Linear Discriminant Analysis

Όπως και στην περίπτωση του Kernel PCA, στον Αλγόριθμο πυρήνα με Ανάλυση Γραμμικής Διακρίνουσας (Kernel LDA) γίνεται η υπόθεση ότι τα δείγματα $\Phi(x)$ που προβάλλονται είναι κεντρικά (centered) στον χώρο \mathbb{R}^l . Οι εξισώσεις διατυπώνονται με τέτοιο τρόπο ώστε να χρησιμοποιούνται αποκλειστικά εσωτερικά γινόμενα κατά τη γραμμική διακρίνουσα ανάλυση.

2. ΑΛΓΟΡΙΘΜΟΙ ΒΑΣΙΖΟΜΕΝΟΙ ΣΕ ΜΟΝΤΕΛΑ ΔΥΟ ΚΑΙ ΤΡΙΩΝ ΔΙΑΣΤΑΣΕΩΝ

Οι W. Bledsoe και H. Chan [11] [12] [13] έπαιξαν κύριο ρόλο τη δεκαετία του 1960 όταν θεωρήθηκαν εφευρέτες των βασιζόμενα σε μοντέλα συστημάτων αναγνώρισης προσώπων. Οι αλγόριθμοι αυτοί, κατασκευάζουν ένα μοντέλο που βασίζεται στην γεωμετρία του ανθρώπινου προσώπου έτσι ώστε να είναι ικανό να περιγράψει τις διαφορές ανάμεσα στα άτομα. Πιο πρόσφατα συστήματα που βασίζονται σε μοντέλα είναι αυτά των γραφημάτων Ελαστικών Δεσμών (Elastic Bunch Matching) [14] -δημιουργήθηκαν το 1997- το σύστημα που βασίζεται σε κρυμμένα πρότυπα Markov (Hidden Markov Model) [15] το οποίο βελτιώθηκε το 1998 [16] όπως και περισσότερο σύνθετα συστήματα 3D μορφοποίησης. Οι κυριότεροι αλγόριθμοι που βασίζονται σε μοντέλα φαίνονται στην παρακάτω εικόνα:



Εικόνα 3.5 Διάγραμμα κυριότερων αλγορίθμων

i. Hidden Markov Models

Σύμφωνα με τους F. Samaria, S. Young [17] και Rabiner. L R [18] τα Hidden Markov Models είναι σύνολο στατιστικών προτύπων που

χρησιμοποιούνται για να χαρακτηρίσουν τις στατιστικές ιδιότητες ενός σήματος. Ένα κρυμμένο πρότυπο Markov αποτελείται από τα εξής:

► Τον αριθμό των καταστάσεων n του μοντέλου. Αν και οι καταστάσεις είναι κρυφές, για πολλές πρακτικές εφαρμογές συνήθως υπάρχει κάποια φυσική σημασία που συνδέεται με τις καταστάσεις ή με τα σύνολα των καταστάσεων του μοντέλου. Οι καταστάσεις είναι συνδεδεμένες έτσι, ώστε οποιαδήποτε κατάσταση να μπορεί να φτάσει τις υπόλοιπες καταστάσεις. Σημειώνεται ότι S είναι το σύνολο των καταστάσεων, με διάνυσμα $S = \{S_1, S_2, \dots, S_N\}$. Ο αριθμός των πλαισίων (μήκος ακολουθίας παρατηρήσεων) είναι T και η κατάσταση του μοντέλου μια χρονική στιγμή t δίνεται ως $q_t \in S, 1 \leq t \leq T$.

► Το M συμβολίζει τον αριθμό των διαφορετικών συμβόλων παρατήρησης. Ας πάρουμε ως παράδειγμα το απλό πείραμα με τη ρίψη νομίσματος, τα σύμβολα παρατήρησης αυτού είναι μόνο κορώνα ή γράμματα. Συμβολίζουμε με V το σύνολο των πιθανών συμβόλων παρατήρησης οπότε ισχύει $V = \{V_1, V_2, \dots, V_M\}$.

► Ο πίνακας κατάστασης μεταφοράς πιθανότητας $P = \{P_{ij}\}$, δείχνει τη πιθανότητα μετατροπής κατάστασης από την S_i στην S_j :

$$P_{ij} = P[q_{t+1} = S_j | q_t = S_i]$$

$$1 \leq i, j \leq N$$

Για την περίπτωση που οποιαδήποτε κατάσταση μπορεί να φτάσει οποιαδήποτε άλλη κατάσταση σε ένα βήμα, υπάρχει ο περιορισμός $a_{ij} > 0 \forall i, j$.

► Ο πίνακας πιθανότητας παρατήρησης συμβόλου είναι $B = \{b_j(k)\}$, όπου:

$$b_j(k) = P[\theta_t = v_k | q_t = S_j], \quad 1 \leq j \leq N, 1 \leq k \leq M$$

όπου θ_t το σύμβολο που παρατηρείται σε χρόνο t .

► Η αρχική κατανομή κατάστασης συμβολίζεται ως $\pi = \{\pi_i\}$, όπου:

$$\pi_i = P[q_1 = S_j], \quad 1 \leq i \leq N$$

$$\pi_i = 0 \quad \text{και} \quad \sum_{i=1}^N \pi_i = 1$$

Δίνοντας κατάλληλες τιμές για τα N , M , A , B και π , ο HMM μπορεί να χρησιμοποιηθεί ως γεννήτρια για να δώσει μια ακολουθία παρατηρήσεων:

$$O = O_1, O_2, \dots, O_T$$

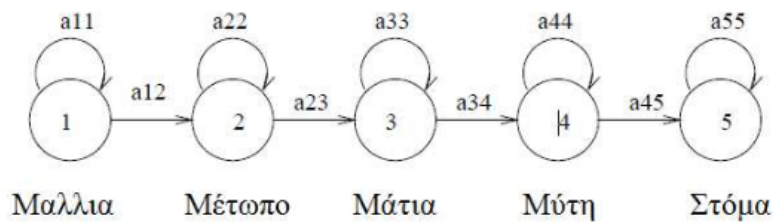
Όπου κάθε παρατήρηση O_T είναι ένα από τα σύμβολα του V και T ο αριθμός των παρατηρήσεων σε μια ακολουθία.

Όμως, σε ένα HMM με συνεχή πυκνότητα, οι καταστάσεις χαρακτηρίζονται από συνεχόμενες συναρτήσεις πυκνότητας παρατήρησης. Η πιο γενική μορφή αυτής -της συνάρτησης πυκνότητας πιθανότητας- είναι:

$$b_i(O) = \sum_{k=1}^M c_{ik} N(O, \mu_{ik}, U_{ik}), \quad 1 \leq i \leq N$$

Όπου c_{ik} είναι ο συντελεστής μείξης για την k -ιοστή μίξη στην κατάσταση i . Χωρίς όμως να χυθεί η γενικότητα το $N(O, \mu_{ik}, U_{ik})$ είναι μια συνάρτηση πυκνότητας πιθανότητας Gauss με μέσο διάνυσμα μ_{ik} και πίνακα συνδιακύμανσης U_{ik} .

Τα Hidden Markov Models έχουν επιτυχώς χρησιμοποιηθεί και για την αναγνώριση ομιλίας όπου η πληροφορία είναι μονοδιάστατη. Όμως η ανάπτυξη σε ένα HMM δύο διαστάσεων είναι σύνθετη υπολογιστικά. Για αυτόν το λόγο, για κατά πρόσωπο εικόνες οι σημαντικές περιοχές του προσώπου (μαλλιά, μέτωπο, μάτια, μύτη, στόμα) έρχονται με μια φυσική κατάσταση από πάνω προς τα κάτω. Όλες οι περιοχές του προσώπου αντιστοιχίζονται σε μια κατάσταση σε ένα μονοδιάστατο στοιχισμένο από αριστερά στα δεξιά HMM.



Εικόνα 3.6: Εξαγωγή χαρακτηριστικών

ii. Ταίριασμα Ελαστικών Δεσμών

Βασικό σκεπτικό της μεθόδου Elastic Bunch Matching, είναι ότι είναι μια δομή που φανερώνει τις εικόνες προσώπων ως 2D απεικονίσεις τρισδιάστατων «αντικειμένων».

Ο τρόπος με τον οποίο λειτουργεί το EBM είναι με τη βοήθεια γραφημάτων με συγκεκριμένα ονόματα κωδικών-ετικετών, οι ακμές κατονομάζονται με πληροφορίες αποστάσεων, και οι κόμβοι ονομάζονται με αποκρίσεις κυματομορφών (wavelet) που σχηματίζουν συστάδες (jets). Τα αποθηκευμένα μοντέλα γραφημάτων εφάπτονται με ταίριασμα στις νέες εικόνες για να σχηματίσουν γραφήματα εικόνων τα οποία στην συνέχεια συμμετέχουν στην δημιουργία της βάσης. Τα wavelets με τον τρόπο που χρησιμοποιούνται μένουν ανεπηρέαστα σε μικρές μετακινήσεις και αλλαγές φωτισμού.

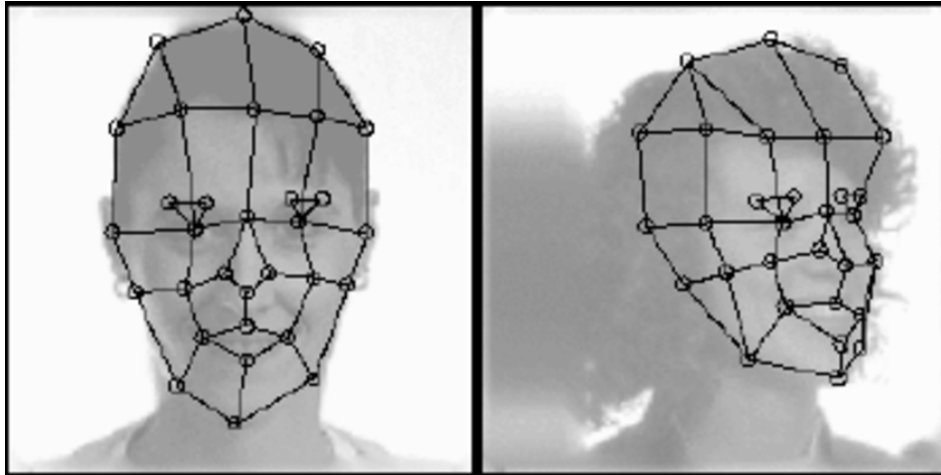
Ο μετασχηματισμός κυματομορφής Gabor (τα Gabor wavelets) λοιπόν, βοηθάει στην αναπαράσταση των τοπικών χαρακτηριστικών, ενώ το σύνολο των συντελεστών της συνέλιξης για πυρήνες με διαφορετικές κατευθύνσεις και συχνότητες σε κάθε στοιχείο ονομάζεται jet.

Για τα πρόσωπα ορίζονται μια σειρά πρότυπων σημείων, όπως οι κόρες των ματιών, τα άκρα των χειλιών, η άκρη της μύτης κ.α. Το γράφημα αυτό μπορεί να «ταιριάξει» σε οποιοδήποτε πρόσωπο μιας και βασίζεται σε συγκεκριμένα πρότυπα σημεία. Για τον προσδιορισμό των πρότυπων σημείων σε νέα πρόσωπα είναι αναγκαία μια γενική αναπαράσταση του ανθρώπινου προσώπου. Η αναπαράσταση πρέπει να καλύπτει μια σειρά διαφοροποιήσεων που μπορεί να εμφανιστούν ανάμεσα σε διαφορετικά πρόσωπα, όπως για παράδειγμα διαφορετικό σχήμα ματιών, μύτης, κλπ. Για το σκοπό αυτό δημιουργούνται συστάδες (bunch) γραφημάτων προσώπου (Face Bunch Graph) ενός συνόλου ατομικών μοντέλων. Η διαδικασία του ταιριάσματος μπορεί να χωριστεί σε τέσσερα βήματα:

- Βήμα 1^ο Προσέγγιση της θέσης του προσώπου.
- Βήμα 2^ο Βελτίωση της θέσης και του μεγέθους

- Βήμα 3ο Βελτίωση της θέσης και εύρεση του λόγου 2 διαστάσεων (aspect ratio)
- Βήμα 4ο Τοπικοί μετασχηματισμοί

Το γράφημα του αποτελέσματος ονομάζεται γράφημα εικόνας και αποθηκεύεται σαν μία αναπαράσταση της συγκεκριμένης εικόνας.



Εικόνα 3.7: Παράδειγμα γραφημάτων

Μετά τον υπολογισμό των πρότυπων γραφημάτων των εικόνων της βάσης και του γραφήματος εικόνας των εικόνων ελέγχου η αναγνώριση επιτυγχάνεται με την σύγκριση του γραφήματος εικόνας με τα γραφήματα της βάσης και επιλογή της εικόνας με τη μεγαλύτερη τιμή ομοιότητας.

iii. Τρισδιάστατο Μοντέλο Μορφοποίησης

Τα 3D Morphable Models ακολουθούν μια συγκεκριμένη στρατηγική κατά την οποία επιδιώκεται τόσο η εξαγωγή ολοκληρωμένων σχημάτων όσο και η εκτίμηση των υφών ως αμετάβλητα χαρακτηριστικά. Επιπρόσθετα, επωφελούνται από το γεγονός ότι οι αλλαγές στην εικόνα λήψης και στο φωτισμό είναι λιγότερο πολύπλοκες στον τρισδιάστατο χώρο από ότι στις εικόνες. Συνοπτικά, αναθέτονται σε ένα απλό μοντέλο τα προβλήματα των διαφοροποιήσεων -της γωνίας λήψης των εικόνων και του φωτισμού-, με

κόστος την πιθανή εξαγωγή μεγαλύτερου όγκου πληροφορίας από την εικόνα από ότι είναι αναγκαίο για την αναγνώριση.

Αυτό το 3D μοντέλο λοιπόν, αφορά ένα διανυσματικό χώρο τρισδιάστατων σχημάτων και υφών και απαρτίζεται από ένα σύνολο εικόνων. Το μοντέλο εντοπίζει τις μεταβολές καθώς και τις κοινές ιδιότητες του συνόλου αυτού. Ορίζονται διανύσματα, για το σχήμα και την υφή, έτσι ώστε κάθε γραμμικός συνδυασμός των εικόνων να είναι ένα πραγματικό πρόσωπο, αν και μόνο αν αυτοί οι γραμμικοί συνδυασμοί έχουν μικρή τυπική απόκλιση από τους μέσους τους.

Κατά τη μετατροπή των εικόνων σε διανύσματα σχήματος και υφής, είναι σημαντικό να καθοριστεί μία αντιστοιχία σημείου προς σημείο κάθε εικόνας με ένα πρόσωπο έτσι ώστε οι διαστάσεις των διανυσμάτων να περιγράφουν το ίδιο σημείο (για παράδειγμα η άκρη της μύτης) σε όλα τα πρόσωπα. Έπειτα, εφαρμόζεται ο αλγόριθμος ανάλυσης σε Κύριες Συνιστώσες (PCA) για να εκτιμηθούν οι πιθανότητες διασποράς των προσώπων γύρω από τους μέσους και αντικαθίστανται τα διανύσματα με το ορθογώνιο σύνολο των ιδιοδιανυσμάτων.

Πλέον, είναι δυνατός ο υπολογισμός μιας έγχρωμης εικόνας. Η εικόνα βέβαια εξαρτάται από πολλές παραμέτρους (για παράδειγμα τρισδιάστατη περιστροφή, γωνία φωτός, χρωματική αντίθεση κ.λπ.). Όλες οι μεταβλητές εκτιμώνται ταυτόχρονα σε έναν ανάλυση-από-σύνθεση (analysis-by-synthesis) βρόχο. Ο κύριος στόχος της ανάλυσης είναι να βρεθούν οι παράμετροι που αυξάνουν όσο το δυνατό περισσότερο την ομοιότητα μεταξύ της νέας εικόνας και της πρωτότυπης.

Οι παράμετροι της εικόνας ανακτώνται αυτόματα, αρχίζουν από μία κατά πρόσωπο στάση στο κέντρο της εικόνας. Επιπλέον, προστίθενται όροι κανονικοποίησης στη συνάρτηση που «τιμωρούν» τις λύσεις που είναι μακριά από το μέσο σε όρους σχήματος, υφής ή των παραμέτρων. Παρακάτω βλέπουμε ανακατασκευασμένες εικόνες ελέγχου.



Εικόνα 3.8 Παράδειγμα βάσης Δεδομένων

Παραπάνω βλέπουμε την ανακατασκευή του τρισδιάστατου σχήματος και υφής κάποιων προσώπων από την βάση δεδομένων FERET. Στη 1^η σειρά είναι οι πρωτότυπες εικόνες που δόθηκαν. Στη 2^η σειρά ο αλγόριθμος ανακτά το φωτισμό και η στάση• ενώ στη 3^η σειρά βλέπουμε το πρόσωπο από μια άλλη γωνία.

Με την ολοκλήρωση της ανάλυσης των παραπάνω αλγορίθμων εξάγονται διάφορα σημαντικά συμπεράσματα. Η απόδοση των αλγορίθμων είναι άμεσα συνδεδεμένη με την δομή των βάσεων στις οποίες εφαρμόζονται. Πιο συγκεκριμένα οι εικόνες της κάθε βάσης διαφέρουν ως προς το κενό (μονόχρωμο) υπόβαθρο, παρουσιάζουν διαφορές στον φωτισμό και στην γωνία λήψης των εικόνων και εμφανίζουν διαφορές στην χρονική στιγμή λήψης των εικόνων. Βέβαια, κάποιοι αλγόριθμοι είναι περισσότερο ευαίσθητοι από άλλους.

3. ΕΠΙΛΕΓΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ

Στις προηγούμενες ενότητες μελετήθηκαν αρκετοί βασικοί. Επιπροσθέτως, θα μπορούσαν να μελετηθούν και να υλοποιηθούν οι αλγόριθμοι που μελετούνται στην παρούσα εργασία, και να εφαρμοστούν στις ίδιες βάσεις προσώπων έτσι ώστε να μπορεί να εφαρμοσθεί απευθείας σύγκριση της απόδοσης τους. Έχοντας μελετήσει λοιπόν παραπάνω διάφορους αλγόριθμους, αποφασίστηκε πως η βέλτιστη λύση για τη παρούσα εργασία είναι να χρησιμοποιηθεί αλγόριθμος βασιζόμενος σε μοντέλο δύο διαστάσεων.

Οι περισσότεροι αλγόριθμοι αναγνώρισης προσώπου είναι εξαιρετικά ευαίσθητοι στις συνθήκες φωτισμού, οπότε αν το σύστημα εκπαιδευόταν να αναγνωρίζει ένα άτομο όταν βρίσκεται σε σκοτεινό δωμάτιο, πιθανότατα δεν θα το αναγνωρίσει σε φωτεινό δωμάτιο και αντιστρόφως. Αυτό το πρόβλημα αναφέρεται ως «lumination dependent».

Λόγω αυτού του προβλήματος («lumination dependent») και για λόγους απλότητας της υλοποίησης, το μοντέλο που χρησιμοποιήθηκε για την αναγνώριση προσώπου στο σύστημα είναι ο PCA (Eigenfaces). Γίνεται μετατροπή έγχρωμων εικόνων σε κλίμακα του γκρι ('grayscale') και ύστερα εφαρμόζεται η εξίσωση ιστογράμματος [25] ως μια πολύ απλή μέθοδος αυτόματης τυποποίησης της φωτεινότητας και αντίθεσης των εικόνων του προσώπου. Στο επόμενο κεφάλαιο υπάρχουν περισσότερες λεπτομέρειες για τον επιλεγμένο αλγόριθμο, κάποια προβλήματά του, όπως επίσης και πως θα μπορούσε να έχει καλύτερα αποτελέσματα.

IV. ΚΕΦΑΛΑΙΟ 4

1. ΤΟ OpenCV ΚΑΙ Ο ΕΠΙΛΕΓΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ

a. OpenCV

Το Open source Computer Vision [19] είναι μια δωρεάν βιβλιοθήκη υπολογιστικής όρασης (computer vision). Ανήκει στην Intel [20], δημιουργήθηκε από τον Gary Bradski και κυκλοφόρησε το 2000 γραμμένο σε C/C++ . Το OpenCV [21] υποστηρίζει πολλά μοντέλα μηχανικής μάθησης όπως τα TensorFlow, Torch, PyTorch (ύστερα από μετατροπή σε μοντέλο ONNX2 -Open Neural Network Exchange-) και Caffe.

Η αναγνώριση προσώπου αποτελεί ένα ισχυρό πεδίο έρευνας από τη δεκαετία του '90, αλλά εξακολουθεί να απέχει πολύ από μια αξιόπιστη μέθοδο ελέγχου ταυτότητας χρήστη. Κάθε χρόνο αναπτύσσονται όλο και περισσότερες τεχνικές αναγνώρισης, η τεχνική Eigenface θεωρείται η πιο απλή μέθοδος για ακριβή αναγνώριση προσώπου, αλλά πολλές άλλες (πολύ πιο περίπλοκες) μέθοδοι ή συνδυασμοί πολλαπλών μεθόδων είναι ελαφρώς πιο ακριβείς.

Το OpenCV έχει ένα ενσωματωμένο ανιχνευτή προσώπου που δουλεύει με περίπου 90-95% ακρίβεια όταν χρησιμοποιούνται εικόνες προσώπων που κοιτάνε τη κάμερα. Δυστυχώς όμως, όταν πρέπει να αναγνωρίσει πρόσωπο από κάποια γωνία τότε τα πράγματα δυσκολεύουν αρκετά, διότι απαιτείται μια 3D εκτίμηση προσώπου. Κάποιοι από τους μεγαλύτερους παράγοντες για τη καλύτερη δυνατή αναγνώριση προσώπου είναι η κατάλληλη φωτεινότητα σε μια εικόνα, ένα πρόσωπο με γυαλιά ή ακόμη και όταν μια εικόνα είναι θολή• όλοι αυτοί οι παράγοντες μπορούν να αυξήσουν τη δυσκολία ανίχνευσης προσώπου. Η αναγνώριση προσώπου όμως είναι λιγότερο αξιόπιστη από την ανίχνευση, με ακρίβεια 50-70%.

Το OpenCV έχει τόσες πολλές δυνατότητες που μπορεί να φαίνονται υπερβολικές στην αρχή. Η σωστή κατανόηση του τρόπου λειτουργίας των μεθόδων του είναι σημαντική για την επίτευξη καλών αποτελεσμάτων κατά

την χρήση του. Ευτυχώς, λίγα στοιχεία χρειάζονται να είναι γνωστά τα οποία είναι και αρκετά. Η λειτουργία του OpenCV που θα χρησιμοποιηθεί για την αναγνώριση προσώπου περιλαμβάνεται σε διάφορες ενότητες. Μια σύντομη περιγραφή των βασικών στοιχείων είναι τα παρακάτω:

- ▶ Το CXCORE περιέχει βασικούς ορισμούς τύπων δεδομένων, γραμμικές άλγεβρες και στατιστικές μεθόδους, τις λειτουργίες επιμονής και τους χειριστές σφαλμάτων. Επίσης, οι λειτουργίες γραφικών για τη σχεδίαση εικόνων βρίσκονται επίσης εδώ.
- ▶ Το CV περιέχει μεθόδους επεξεργασίας εικόνας και βαθμονόμησης κάμερας. Οι υπολογιστικές συναρτήσεις γεωμετρίας βρίσκονται επίσης εδώ.
- ▶ Στη τεκμηρίωση του OpenCV αναφέρεται ότι το CVAUX περιέχει ξεπερασμένο και πειραματικό κώδικα. Ωστόσο, οι απλούστερες διεπαφές για την αναγνώριση προσώπων βρίσκονται σε αυτήν την ενότητα. Ο κώδικας που βρίσκεται πίσω από αυτές τις διεπαφές είναι εξειδικευμένος για την αναγνώριση προσώπου και χρησιμοποιείται ευρέως για το σκοπό αυτό.
- ▶ Το ML περιέχει διεπαφές μηχανικής μάθησης.
- ▶ Το HighGUI περιέχει τις βασικές διεπαφές I/O(εισόδου-εξόδου).
- ▶ Το CVCAM περιέχει διεπαφές για πρόσβαση σε βίντεο μέσω του DirectX σε πλατφόρμες 32-bit των Windows.

Οι περισσότεροι πόροι για την αναγνώριση προσώπων προορίζονται για βασικά Νευρωνικά Δίκτυα, τα οποία συνήθως δεν λειτουργούν όπως και το Eigenfaces. Δυστυχώς, υπάρχουν μόνο μερικές βασικές εξηγήσεις για καλύτερο τύπο αναγνώρισης προσώπου από το Eigenfaces, όπως αναγνώριση από βίντεο και άλλες τεχνικές στο Face Recognition Homepage [22], στη σελίδα της Wikipedia “3D αναγνώρισης προσώπου” [23] αλλά και στη σελίδα Active Appearance Models [24]. Για περισσότερες πρόσφατες τεχνικές, υπάρχουν μερικά ερευνητικά έγγραφα από το CVPR και άλλα συνέδρια.

2. Eigenfaces

Στο τρίτο κεφάλαιο αποφασίστηκε πως ο αλγόριθμος που αρμόζει καλύτερα σε αυτή την εργασία είναι ο Eigenfaces. Μπορεί βέβαια να είναι “απλός” αλγόριθμος, υπάρχει όμως πληθώρα παραγόντων που επηρεάζουν την ακρίβεια του συστήματος αυτού, εάν λάβουμε υπόψιν την ποιότητα της εικόνας. Για αυτό το λόγο είναι εξαιρετικά σημαντικό να εφαρμοστούν διάφορες τεχνικές προ-επεξεργασίας εικόνας για να τυποποιηθούν οι εικόνες που παρέχονται σε ένα σύστημα αναγνώρισης προσώπου.

Υπάρχουν βέβαια πολλά άλλα ζητήματα εκτός του «lumination dependent», όπως ότι το πρόσωπο πρέπει επίσης να βρίσκεται σε πολύ συνεπή θέση εντός των εικόνων (όπως τα μάτια να έχουν τις ίδιες συντεταγμένες pixel), σταθερό μέγεθος, μαλλιά και μακιγιάζ, αισθήματα(με χαμόγελο, θυμωμένος, κ.λπ.), θέση των φώτων (προς τα αριστερά ή πάνω, κ.λπ.). Για αυτό τον λόγο είναι πολύ σημαντικό να χρησιμοποιούνται καλά φίλτρα προ-επεξεργασίας εικόνας πριν από την εφαρμογή αναγνώρισης προσώπου.

Θα ήταν καλό επίσης, να αφαιρεθούν τα pixels γύρω από το πρόσωπο που δεν χρησιμοποιούνται, για παράδειγμα με τη βοήθεια μιας ελλειπτικής μάσκας για να δείξουμε μόνο την εσωτερική περιοχή του προσώπου και όχι το φόντο των μαλλιών και της εικόνας, καθώς αλλάζουν περισσότερο από ό,τι το πρόσωπο.

Για καλύτερα αποτελέσματα, θα μπορούσαν να εφαρμοστούν περισσότερα στάδια επεξεργασίας όπως η ανίχνευση περιγράμματος, ανίχνευση κίνησης κ.λπ. Επίσης, ο κώδικας (βλέπε παράρτημα) αλλάζει το μέγεθος της εικόνας σε τυπικό μέγεθος, αλλά αυτό μπορεί να αλλάξει την αναλογία διαστάσεων του προσώπου. Το OpenCV χρησιμοποιεί έναν αλγόριθμο μηχανικής μάθησης που ονομάζεται HAAR Cascade Classifier και χρησιμοποιείται για την αναγνώριση αντικειμένων σε μια εικόνα ή βίντεο, προτάθηκε από τον Paul Viola και τον Michael Jones [26].

3. ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ HAAR CASCADE

Δίνοντας μια εικόνα, η οποία μπορεί να προέρχεται από ένα αρχείο ή από βίντεο, ο ανιχνευτής προσώπου εξετάζει κάθε θέση εικόνας και το ταξινομεί ως "Πρόσωπο" ή "Όχι πρόσωπο". Η ταξινόμηση προϋποθέτει μια σταθερή κλίμακα για το πρόσωπο, έστω 50×50 pixels. Εφόσον τα πρόσωπα σε μια εικόνα μπορεί να είναι μικρότερα ή μεγαλύτερα από αυτή τη κλίμακα, ο ταξινομητής περνάει την εικόνα αρκετές φορές, για να αναζητήσει πρόσωπα. Αυτό μπορεί να φαίνεται ως τεράστια ποσότητα επεξεργασίας, αλλά χάρη σε αλγοριθμικά τεχνάσματα, η ταξινόμηση είναι αρκετά γρήγορη, ακόμα και όταν εφαρμόζεται σε διάφορες κλίμακες.

Ο ταξινομητής χρησιμοποιεί δεδομένα που είναι αποθηκευμένα σε ένα αρχείο XML [haarcascade_frontalface_default.xml]. Ο Haar Cascade είναι ταξινομητής που χρησιμοποιείται για την ανίχνευση συγκεκριμένων αντικειμένων από την πηγή. Με άλλα λόγια το παραπάνω αρχείο είναι ένας Haar ταξινομητής σχεδιασμένος από το OpenCV για την ανίχνευση προσώπων. Ο ταξινομητής είναι διαθέσιμος προς όλους στο Github⁴. Ένας Haar ταξινομητής λειτουργεί εκπαιδύοντας τον ταξινομητή σε χιλιάδες αρνητικές εικόνες με τη θετική εικόνα να τοποθετείται πάνω τους. Η λήψη του OpenCV, περιλαμβάνει και τέσσερα XML αρχεία, ένα για ανίχνευση προσώπου, ένα για ανίχνευση προφίλ προσώπου όπως και -άλλα τρία αρχεία XML που δεν αφορούν το πρόσωπο- ένα για αναγνώριση ολόκληρου σώματος, ένα για το πάνω μέρος του σώματος και ένα για το κάτω μέρος του σώματος. Θα πρέπει να «δείξουμε» στον ταξινομητή που είναι το αρχείο δεδομένων που θέλουμε να χρησιμοποιήσει.

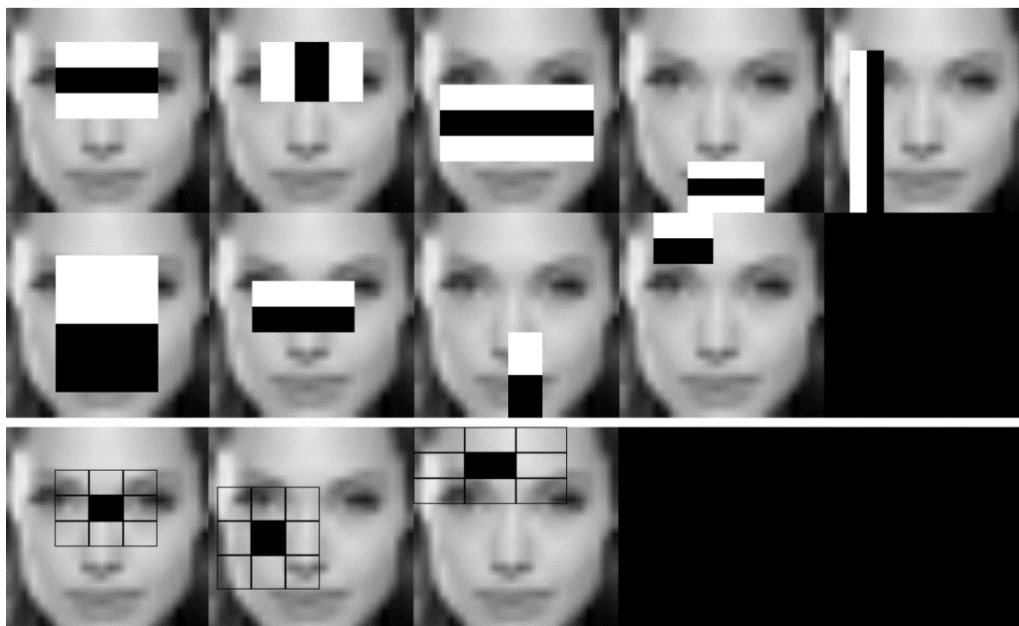
Είναι πολύ εύκολο να χρησιμοποιηθεί μια ροή κάμερας ως είσοδος στο σύστημα αναγνώρισης προσώπου αντί για μια λίστα αρχείων για δύο λόγους, πρώτον διότι το μόνο που χρειάζεται να γίνει είναι να τραβηχτούν πλαίσια από τη κάμερα αντί για αρχεία και δεύτερον το πρόγραμμά μας εκτελείται για πάντα έως ότου ο χρήστης θελήσει να το σταματήσει, αντί να τρέχει μέχρι να

⁴ GitHub: Εξυπηρετητής που φιλοξενεί αποθετήρια και μια εύχρηστη διεπαφή ιστού. Επίσης παρέχει έλεγχο πρόσβασης και πολλές δυνατότητες συνεργασίας, όπως ανίχνευση σφαλμάτων, αιτήματα χαρακτηριστικών, διαχείριση εργασιών, και wikis για κάθε project.

τελειώσει η λίστα αρχείων. Παρακάτω υπάρχει ένα παράδειγμα του μοντέλου HAAR χρησιμοποιώντας πρόσωπο σε διαστάσεις 24x24:



Εικόνα 4.1 Κάθε στάδιο αποθηκεύεται ως εικόνα για μελλοντική επικύρωση χαρακτηριστικών.



Εικόνα 4.2: Οι παραπάνω εικόνες δημιουργήθηκαν για το OpenCV3 από τον Steven Puttemans.

Συνοψίζοντας, για να έχουμε σωστή αναγνώριση από αυτόν τον αλγόριθμο θα πρέπει να γίνει περικοπή εικόνων όσο πιο στενά στα χαρακτηριστικά του προσώπου γίνεται. Αυτός ο ταξινομητής μπορεί να χειριστεί μόνο αντικείμενα (που είναι προς αναγνώριση) τα οποία είναι σε ορθογώνια των οποίων οι

άξονες είναι ευθυγραμμισμένοι με την εικόνα (χωρίς δηλαδή περιστρεφόμενα ορθογώνια). Φυσικά ο HAAR είναι “ιδανικός” για ανθρώπινα πρόσωπα διότι πάντα βρίσκονται σε σωστή θέση και όχι ανάποδα. Αλλά σε περίπτωση που θα θέλαμε να αναγνωρίσουμε και ανεστραμμένα πρόσωπα τότε θα χρειαζόμασταν και πάλι πρόσωπα στη σωστή τους θέση (όπως στην εικόνα 4.1) και ύστερα θα γινόταν 2 φορές έλεγχος, ο πρώτος με την αρχική εικόνα και ο δεύτερος με την εικόνα αναστρεμμένη.

Οπότε και στη περίπτωση αυτού του συστήματος ασφαλείας, όλες οι εικόνες που θα δοθούν στον αλγόριθμο πρέπει να είναι ευθυγραμμισμένες έτσι ώστε το πρόσωπο (στις εικόνες) να έχει περίπου τον ίδιο προσανατολισμό και κλίμακα. Στο επόμενο κεφάλαιο, αναλύονται τα υλικά και το λογισμικό που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος.

V. ΚΕΦΑΛΑΙΟ 5

1. ΥΛΙΚΑ ΚΑΙ ΛΟΓΙΣΜΙΚΟ

a. Raspberry Pi και PiNoIR

Το Raspberry Pi (εικόνα 5.1) ανήκει στην ομάδα υπολογιστών-κάρτα γενικής χρήσης, συνήθως με λειτουργικό σύστημα Linux και έχουν δυνατότητα εκτέλεσης πολλαπλών προγραμμάτων. Είναι ένα ολοκληρωμένο υπολογιστικό σύστημα με μέγεθος 85mm x 56mm x 17mm και μόλις 50 γραμμάρια.

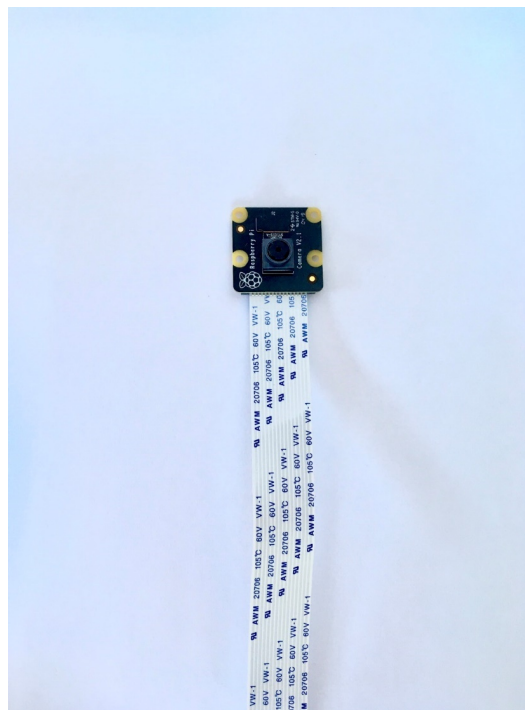


Εικόνα 5.1: Μέγεθος του Raspberry Pi 3B+

Η έκδοση που χρησιμοποιήθηκε για να διεξαχθούν όλα τα πειράματα ήταν η Raspberry Pi 3 Model B+. Παρά το μικρό μέγεθός του, το Raspberry Pi διαθέτει επεξεργαστή Quad Core 1.4 GHz 64-Bit, 1GB RAM, 4 θύρες USB 2.0 για σύνδεση με πληκτρολόγιο, ποντίκι και άλλα περιφερειακά, παρέχει

διασύνδεση κάμερας (CSI), θύρα Ethernet, WiFi, Bluetooth 4.1, έξοδο HDMI, 40 ακροδέκτες γενικής χρήσης για σύνδεση με άλλα ηλεκτρονικά και περιφερειακά (pin GPIO), έξοδο ήχου mini jack και microUSB υποδοχή για να την τροφοδοσία του. Για να λειτουργήσει χρειάζεται τροφοδοσία 5V 2.5A και μια κάρτα micro SD με τουλάχιστον 4GB (εικόνα 5.3), στην οποία θα εγκατασταθεί το λειτουργικό σύστημα (Raspbian). Επιπλέον, χρειάστηκε ένας προσαρμογέας για σύνδεση της κάρτας micro SD στον υπολογιστή για να εγκατασταθεί το λειτουργικό σύστημα, ένα καλώδιο HDMI, σύνδεση σε internet/καλώδιο Ethernet, ένα ποντίκι, ένα πληκτρολόγιο και μια οθόνη.

Ως κάμερα χρησιμοποιήθηκε η PiNoIR με υψηλή ανάλυση 8 megapixels fixed focus (3.280 x 2.464 pixel) και καλή ποιότητα χρώματος, υποστηρίζει 1080p, 720p60 & VGA90. Βασίζεται στον αισθητήρα της Sony IMX219PQ CMOS ο οποίος διαθέτει φίλτρο υπέρυθρων και παρέχει ανάλυση εικόνας. Η PiNoIR είναι κατάλληλη και για νυχτερινή χρήση. Για να συνδεθεί η κάμερα στο Raspberry Pi συνδέουμε τη κορδέλα 15 εκατοστών που υπάρχει στην υποδοχή της μονάδας με τη θύρα σειριακής διασύνδεσης της κάμερας (CSI) (εικόνα 5.2).



Εικόνα 5.2: Raspberry PiNoIR κάμερα.

b. Υπόλοιπα Υλικά

Μαζί με τα παραπάνω χρειάστηκαν βέβαια άλλα 4 βασικά αντικείμενα. Αρχίζοντας με την ηλεκτρομαγνητική κλειδαριά• αν και είναι πολύ εύκολη στη χρήση ένα πρόβλημα που εμφανίστηκε ήταν ότι η κλειδαριά για να λειτουργήσει σωστά χρειάζεται από 7V έως 12V ενώ το σύστημα μπορεί να προσφέρει από 3,3V έως και 5V. Για την επίλυση αυτού του προβλήματος χρησιμοποιήθηκε εξωτερική πηγή ρεύματος συνεχούς τάσης που μπορεί να προσφέρει έως και 12V.

Ύστερα με τη βοήθεια ενός ρελέ των 5V (το οποίο χρησιμοποιείται για τον έλεγχο κυκλωμάτων υψηλής τάσης με τη βοήθεια σημάτων χαμηλής τάσης αλλά και για τον έλεγχο κυκλωμάτων υψηλού ρεύματος με τη βοήθεια σημάτων χαμηλού ρεύματος), συνδέουμε το σύστημα. Τέλος χρησιμοποιήθηκε μια οθόνη 5 ιντσών η οποία συνδέθηκε με το Raspberry Pi (εικόνες 5.3, 5.4, 5.5, 5.6).



Εικόνα 5.3: Solenoid Lock / Ηλεκτρομαγνητική κλειδαριά



Εικόνα 5.4: Εξωτερική πηγή ρεύματος 3V-12V



Εικόνα 5.5: Κάρτα μνήμης 8G



Εικόνα 5.5: Ρελέ 5V



Εικόνα 5.6: Οθόνη 5 ιντσών.

3. ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ

Το Raspberry Pi κάνει όλες τις βασικές λειτουργίες που κάνει και ένας σύγχρονος υπολογιστής, επιτρέποντας σε έναν μη έμπειρο χρήστη να περιηγηθεί στο διαδίκτυο, να μεταδίδει το περιεχόμενό του και να κάνει τα πρώτα του βήματα στον προγραμματισμό. Ορισμένες βασικές χρήσεις αυτής της μικρής συσκευής είναι οι παρακάτω:

- ▶ Προγραμματισμός: Το Raspberry Pi είναι ο ιδανικός τρόπος για να ξεκινήσουμε στον τομέα του προγραμματισμού. Υπάρχουν πολλές γλώσσες με τις οποίες μπορούμε να ασχοληθούμε αλλά το Raspbian παρέχει περιβάλλον ανάπτυξης κώδικα σε γλώσσα Python που ονομάζεται Thonny IDE.
- ▶ Personal Computer (PC): Το Raspberry μπορεί να χρησιμοποιηθεί ως σταθερός υπολογιστής αλλά και ως φορητός. Μπορεί να έχει περιορισμένες δυνατότητες έχει όμως επαρκή δύναμη για να καλύψει τις περισσότερες ανάγκες του μέσου χρήστη.
- ▶ Παιχνίδια: Όταν μιλάμε για απλά παιχνίδια που δεν χρειάζονται προηγμένο υλικό, το Raspberry είναι πολύ καλή επιλογή. Μπορούμε να παίξουμε retro παιχνίδια ή και να φτιάξουμε δικό μας server.
- ▶ Λειτουργικό Σύστημα: Μπορούμε εύκολα να εγκαταστήσουμε και να χρησιμοποιήσουμε οποιοδήποτε λειτουργικό σύστημα υπάρχει στην αγορά.

d. Λειτουργικό Σύστημα

Το λειτουργικό σύστημα που χρησιμοποιήθηκε είναι το Raspbian το οποίο βασίζεται στο Debian το οποίο ανήκει στο Linux (λειτουργικό σύστημα που αποτελείται από δωρεάν λογισμικά). Το Raspbian έχει διάφορες εκδόσεις όπως το Raspbian Buster και Raspbian Stretch. Πλέον, είναι το κύριο λειτουργικό σύστημα όλων των Raspberry Pi και έχει μετονομαστεί σε Raspberry Pi Os. Δημιουργήθηκε από τους Mike Thompson και Peter Green [27].

VI. ΚΕΦΑΛΑΙΟ 6

1. ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

Ως πρώτο βήμα για την υλοποίηση αυτού του συστήματος ήταν η συλλογή των δεδομένων με την βοήθεια των οποίων εκπαιδεύτηκε ο ταξινομητής. ύστερα γράφτηκε ο κώδικας σε γλώσσα Python χρησιμοποιώντας τον προ-εκπαιδευμένο ταξινομητή του OpenCV που αναφέρθηκε παραπάνω. Αν και το OpenCV περιέχει πάρα πολλούς προ-εκπαιδευμένους ταξινομητές -για πρόσωπο, μάτια, χαμόγελο κ.λπ.- ο ταξινομητής που χρησιμοποιήθηκε είναι ειδικός στο να εντοπίζει πρόσωπα και μπορεί να βρεθεί δωρεάν στο GitHub⁵. Αποθηκεύουμε το αρχείο αναγνώρισης προσώπου ως “haarcascade_frontalface_default.xml”.

Επόμενο βήμα ήταν η συλλογή των υλικών. Όπως προαναφέρθηκε χρησιμοποιήθηκε το Raspberry Pi 3 μοντέλο B+, η κάμερα Raspberry PiNoIR, σωληνοειδής κλειδαριά, εξωτερική πηγή συνεχούς ρεύματος, ένα ρελέ (ένας ηλεκτρικός διακόπτης που ανοίγει και κλείνει ένα ηλεκτρικό κύκλωμα κάτω από τον έλεγχο ενός άλλου ηλεκτρικού κυκλώματος) των 5V και καλώδια. Χρειάστηκε εξωτερική πηγή ενέργειας των 7,5V και ρελέ των 5V (διότι το Raspberry Pi έχει τη δυνατότητα να παρέχει μόνο έως 3.3V) για τη σωστή λειτουργία του συστήματος, δηλαδή έτσι ώστε η κλειδαριά να μπορέσει να λειτουργήσει.

a. Εγκατάσταση κάμερας

Το PiNoIR είναι μία από τις πολλές μονάδες κάμερας που μπορούν να χρησιμοποιηθούν για να ενσωματώσουμε στο Raspberry Pi. Αυτή η κάμερα δεν είναι ίδια με την τυπική φωτογραφική μηχανή ημέρας. Το PiNoIR έχει λειτουργίες που διαθέτει μια κανονική μονάδα κάμερας, αλλά έχει μια μεγάλη

διαφορά: δεν διαθέτει φίλτρο υπέρυθρων. Αυτό σημαίνει και το όνομα NoIR-
No infrared (Χωρίς υπέρυθρες).

Δεδομένου των παραπάνω λοιπόν, η Pi NoIR δεν έχει την ίδια ποιότητα με μια υπέρυθρη κάμερα κατά τη διάρκεια της ημέρας. Όσον αφορά τα περιβάλλοντα χαμηλού φωτισμού, ωστόσο, η κάμερα NoIR βλέπει πολύ καλύτερα, καθιστώντας την μια καλή κάμερα για εφαρμογές όπως συστήματα νυχτερινής παρακολούθησης για παράδειγμα.

Για να εγκαταστήσουμε τη κάμερα στο Raspberry χρειάζεται να έχουμε εγκαταστήσει το Raspbian. Αφού εγκατασταθεί το λειτουργικό σύστημα πρέπει να ελέγξουμε αν το Raspbian διαθέτει ήδη έκδοση για τη λειτουργία της Picamera, την οποία θα χρειαστούμε για να χρησιμοποιήσουμε το Pi NoIR. Για να ελέγξουμε εάν υπάρχει έκδοση Picamera που είναι ήδη εγκατεστημένη στο λειτουργικό σύστημα (Raspbian) που διαθέτουμε, πρέπει να εισάγουμε στο τερματικό του Raspberry Pi τις παρακάτω εντολές:

```
python -c "import picamera"  
python3 -c "import picamera"
```

Αν η κάμερα υπάρχει ήδη στο σύστημα δεν θα υπάρξει κάποιο σφάλμα στο τερματικό. Εάν δεν υπάρχει, θα μας εμφανίσει ανάλογο μήνυμα. Για να εγκαταστήσουμε τη κάμερα βάζουμε τις ακόλουθες εντολές στο τερματικό:

```
sudo apt-get update  
sudo apt-get install python-picamera python3-picamera
```

Ύστερα, είμαστε έτοιμοι να συνδέσουμε τη κάμερα στο Raspberry.

b. Εγκατάσταση Βιβλιοθήκης OpenCV

Αφού γίνουν τα παραπάνω βήματα, το βασικό επόμενο βήμα είναι να ολοκληρωθεί σωστά η εγκατάσταση της βιβλιοθήκης OpenCV στο Raspberry Pi. Παρακάτω είναι τα 11 βήματα που χρειάστηκαν για να ολοκληρωθεί η εγκατάσταση επιτυχώς:

- ▶ Επέκταση συστήματος αρχείων(Filesystem)

```
>sudo raspi-config
```

Επιλογές για προχωρημένους → A1 Ανάπτυξη συστήματος αρχείων
→ "Enter"

Στη συνέχεια, κάνουμε επανεκκίνηση το Raspberry Pi χρησιμοποιώντας την ακόλουθη εντολή

```
>sudo shutdown -r now
```

- ▶ Απελευθέρωση χώρου

Το προεπιλεγμένο λειτουργικό σύστημα θα χρησιμοποιήσει περίπου το 15% χώρου εάν χρησιμοποιείτε κάρτα μνήμης 32GB. Αν όμως χρησιμοποιείτε κάρτα μνήμης 8GB, τότε σίγουρα μπορεί να χρειαστεί το 50% του χώρου. Επομένως, είναι καλύτερο να αφαιρεθούν ορισμένα πακέτα που δεν χρησιμοποιούνται όπως το LibreOffice και η μηχανή Wolfram για να ελευθερώσουμε λίγο χώρο στο Raspberry Pi.

```
>sudo apt-get purge wolfram-engine
```

```
>sudo apt-get purge libreoffice*
```

```
>sudo apt-get clean
```

```
>sudo apt-get autoremove
```

- ▶ Εγκατάσταση Εξαρτήσεων

(i) Ενημέρωση υπαρχόντων πακέτων

>*sudo apt-get update*

>*sudo apt-get upgrade*

(ii) Σε περίπτωση που υπάρξει σφάλμα εκτελούμε τη παρακάτω εντολή

>*sudo apt-get upgrade --fix-missing*

▶ Επανεκκίνηση του Raspberry Pi

>*sudo shutdown -r now*

▶ Εγκατάσταση πακέτου εικόνας I/O

>*sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev -y*

▶ Εγκατάσταση πακέτου βίντεο I/O

>*sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev -y*

>*sudo apt-get install libxvidcore-dev libx264-dev -y*

▶ Εγκατάσταση βιβλιοθήκης GTK για βασικά γραφικά σε windows

>*sudo apt-get install libgtk2.0-dev libgtk-3-dev -y*

▶ Εγκατάσταση πακέτων βελτιστοποίησης (περιέχουν αναβαθμισμένες λειτουργίες πινάκων για το OpenCV)

```
>sudo apt-get install libatlas-base-dev gfortran -y
```

I) Εγκατάσταση Python 3 και βιβλιοθήκης Numpy

```
> sudo apt-get install python3 python3-setuptools python3-dev -y
```

```
>wget https://bootstrap.pypa.io/get-pip.py
```

```
>sudo python3 get-pip.py
```

```
>sudo pip3 install numpy
```

II) Εγκατάσταση του OpenCV 3.4

```
>cd ~
```

```
>wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.4.0.zip
```

```
>unzip opencv.zip
```

```
>wget -O opencv_contrib.zip https://github.com/Itseez/opencv_contrib/archive/3.4.0.zip
```

```
>unzip opencv_contrib.zip
```

III) Μεταγλώττιση και εγκατάσταση του OpenCV3.4.0 για το Python3

```
>cd opencv-3.4.0
```

```
>mkdir build
```

```
>cd build
```

```
>cmake -D CMAKE_BUILD_TYPE=RELEASE \
```

```
>D CMAKE_INSTALL_PREFIX=/usr/local \  
  
>D BUILD_opencv_java=OFF \  
  
>D BUILD_opencv_python2=OFF \  
  
>D BUILD_opencv_python3=ON \  
  
>D PYTHON_DEFAULT_EXECUTABLE=$(which python3) \  
  
>D INSTALL_C_EXAMPLES=OFF \  
  
>D INSTALL_PYTHON_EXAMPLES=ON \  
  
>D BUILD_EXAMPLES=ON \  
  
>D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-3.4.0/  
modules \  
  
>D WITH_CUDA=OFF \  
  
>D BUILD_TESTS=OFF \  
  
>D BUILD_PERF_TESTS= OFF ..
```

Έτοιμο για μεταγλώττιση.

Με τη παρακάτω εντολή χρησιμοποιούμε 4 επεξεργαστές του Raspberry pi για τη μεταγλώττιση:

```
>make -j4
```

IV) Εγκατάσταση της έκδοσης στο Raspberry Pi

```
>sudo make install
```

```
>sudo ldconfig
```

V) Δοκιμή του OpenCV 3.4.0

```
>>> import cv2  
>>> cv2.__version__ '3.4.0'
```

Υπάρχει περίπτωση να προκύψει πρόβλημα με την έκδοση του Python 3 που έχει εγκαταστήσει αυτόματα το OpenCV, ο τρόπος αντιμετώπισης αυτού του προβλήματος είναι να διαγραφεί η έκδοση του OpenCV και Python.

```
>pip3 uninstall opencv-python
```

και στη συνέχεια τα εγκατέστησα πάλι με την εξής εντολή:

```
>pip3 install opencv-python == 3.4.6.27
```

(Για παράδειγμα από το τερματικό του Raspberry Pi βλέπε παράρτημα εικόνα 4.)

2. ΣΥΝΔΕΣΕΙΣ

Όπως προαναφέρθηκε το Raspberry Pi μπορεί να μας προσφέρει μέσω των ακροδεκτών γενικής χρήσης (GPIO pin) έως και 3.3V. Για αυτό το λόγο έπρεπε να βρεθεί κάποιος τρόπος που θα έλυne το θέμα πηγής ρεύματος. Εφόσον χρειάζονται τουλάχιστον 7V έως το πολύ 12V, ο μοναδικός τρόπος που μπορεί να υλοποιηθεί το σύστημα είναι με τη βοήθεια βύσματος (που προσφέρει έως και 12V) και ενός ρελέ 5V.

Κατευθείαν βέβαια έρχεται στο μυαλό ένα σημαντικό πρόβλημα: Σε περίπτωση διακοπής ρεύματος δεν θα μπορούμε να χρησιμοποιήσουμε αυτό το σύστημα. Κι όμως, αυτό ίσχυε και για το Raspberry Pi γιατί χρειάζεται ενέργεια για να λειτουργήσει. Μια απλή λύση είναι να φτιάξουμε δική μας πηγή με μπαταρίες και όχι μέσω άμεσης σύνδεσης βύσματος με υποδοχή. Επειδή ένας από τους σκοπούς αυτής της διπλωματικής είναι να δούμε πόσο εύκολα μπορούμε να φτιάξουμε ένα τόσο ιδιαίτερο σύστημα, η πιο γρήγορη

λύση αυτού το προβλήματος είναι να αποκτηθεί ένα βύσμα που δίνει έως και 12V συνεχούς ενέργειας (DC).

Αφού λοιπόν βρεθεί η κατάλληλη πηγή συνεχούς ενέργειας, όπως στη περίπτωση αυτή το βύσμα, ύστερα πρέπει να το συνδέσουμε με ένα μικρό ρελέ των 5V έτσι ώστε να λειτουργήσει η κλειδαριά.

Όπως βλέπουμε και στο διάγραμμα (βλέπε παράρτημα πρόσθετη εικόνα 2), η κλειδαριά έχει συνδεθεί με:

- ▶ Εξωτερική πηγή ενέργειας που προσφέρει έως και 12V.
- ▶ Ένα ρελέ, το οποίο στέλνει σήμα στην κλειδαριά να λειτουργήσει.

Όπως βλέπουμε και στο διάγραμμα συνεχίζοντας με τον ρελέ, συνδέθηκε το VCC στα 5V και το GND στο GND του Raspberry Pi. Στη συνέχεια, συνδέθηκε ο ακροδέκτης σήματος του ρελέ στο GPIO 26 του Raspberry Pi.

Στην άλλη πλευρά του ρελέ, έχει συνδεθεί η πηγή ισχύος αρνητικής μορφής (DC) στη πηγή της ηλεκτρομαγνητικής κλειδαριάς. Η θετική μορφή συνδέεται από τη πηγή ενέργειας DC στο ρελέ και τέλος συνδέθηκε ο ρελέ κανονικά με τη θετική μορφή της ηλεκτρομαγνητικής κλειδαριάς.

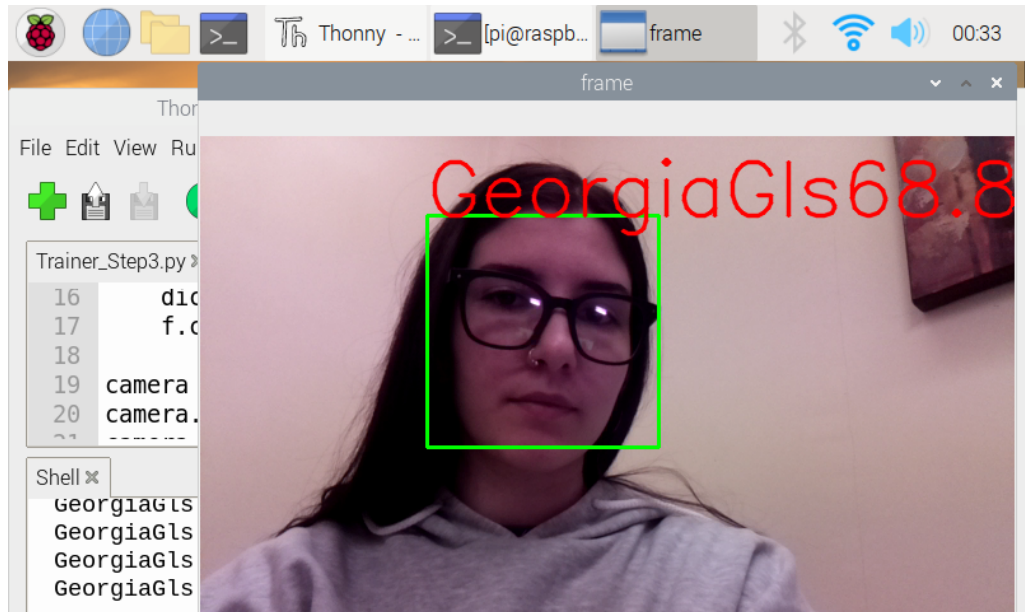
3. ΔΟΚΙΜΗ ΣΥΣΤΗΜΑΤΟΣ

Αφού ολοκληρώθηκαν επιτυχώς τα παραπάνω βήματα, έγινε η δοκιμή του συστήματος. Εκτελώντας το για πρώτη φορά, δημιουργήθηκε φάκελος με 30 εικόνες του προσώπου μου κοιτώντας ευθέως την κάμερα. Ύστερα, έγινε λήψη εικόνων φορώντας γυαλιά οράσεως.

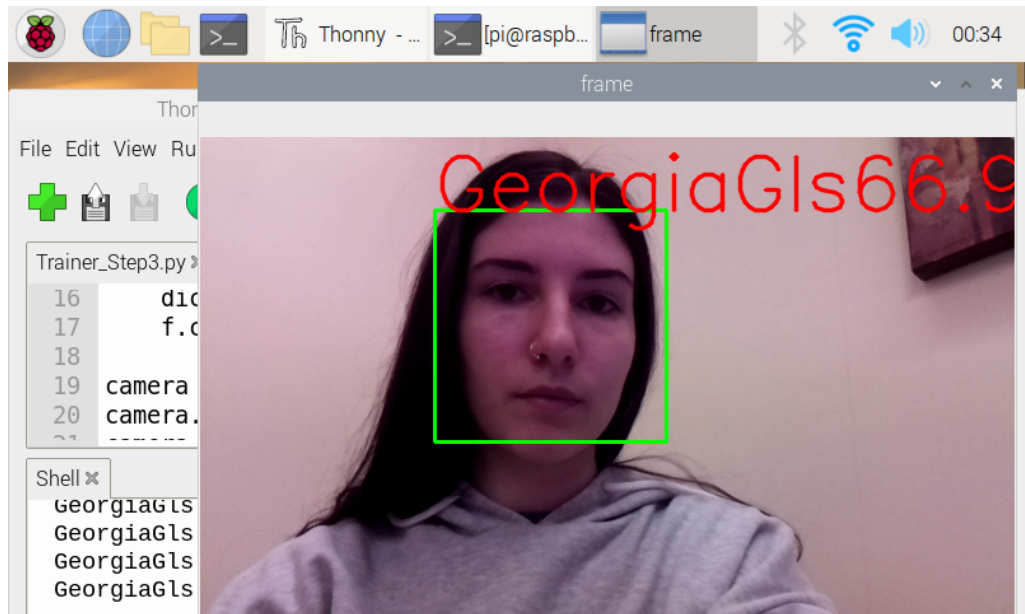


Εικόνα 6.5.1: Δείγμα εικόνων από το αρχείο που δημιουργήθηκε με όνομα GeorgiaGls για την αναγνώριση προσώπου μου με γυαλιά. Βλέπε παράρτημα για δείγμα εικόνων χωρίς γυαλιά από το αρχείο Georgia

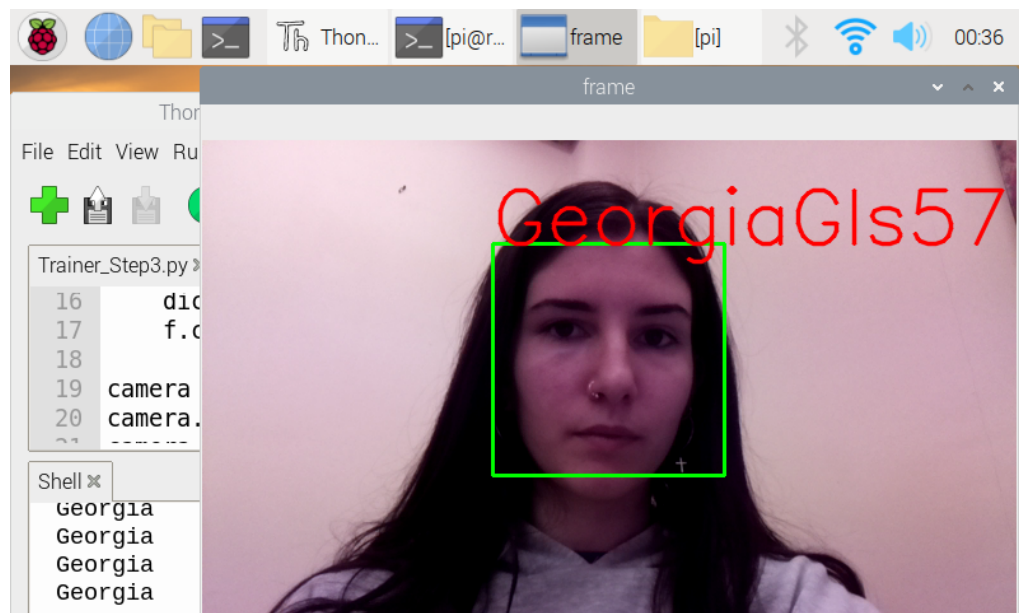
Οι φάκελοι που περιέχουν τις εικόνες προσώπου μου ονομάστηκαν GeorgiaGls (συντόμευση για GeorgiaGlasses) για αναγνώριση του προσώπου μου με γυαλιά οράσεως και Georgia χωρίς γυαλιά. Στη πρώτη δοκιμή, χρησιμοποιώντας πρόσωπα άγνωστα το σύστημα, όπως ήταν αναμενόμενο, ήταν αρκετά γρήγορη η ανίχνευση προσώπου αλλά η αναγνώριση του φαινόταν να μην είναι δυνατή. Ύστερα, κοιτάζοντας την κάμερα ευθέως -φορώντας τα γυαλιά οράσεως- και έχοντας τα μαλλιά πίσω από τους ώμους το σύστημα έφτασε έως και 68% - 70% ακρίβεια ενώ έπεσε στο 64% χωρίς τα γυαλιά (εικόνες 6.5.2 και 6.5.3), ενώ με τα μαλλιά μπροστά στο πρόσωπο το σύστημα δυσκολευόταν να εντοπίσει σωστά τα χαρακτηριστικά του προσώπου με αποτέλεσμα να έχει χαμηλή επιτυχία 57,7% (εικόνα 6.5.4).



Εικόνα 6.5.2: Αναγνώριση προσώπου με 68,9% επιτυχία.

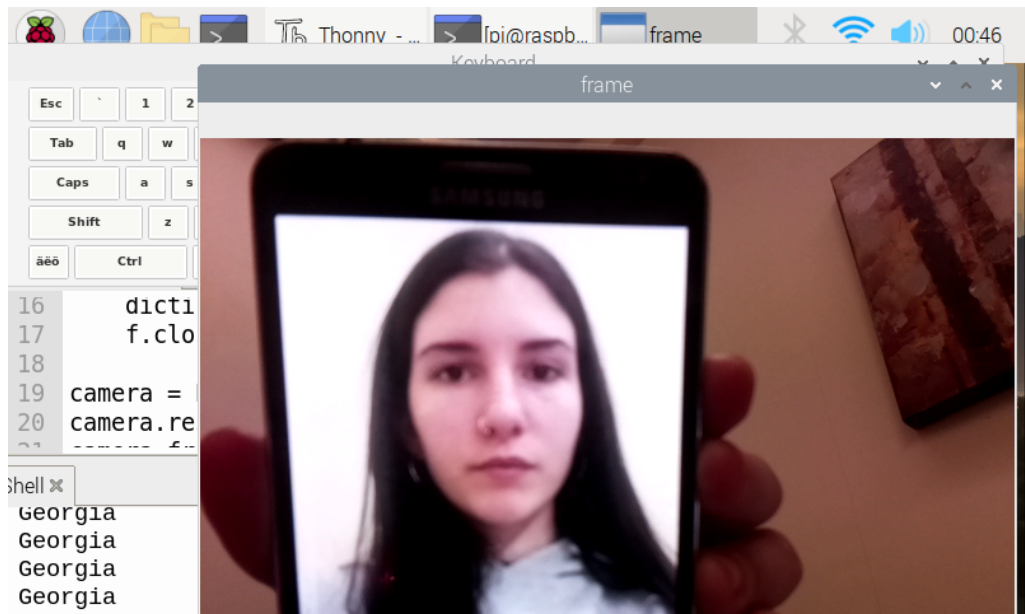


Εικόνα 6.5.3: Αναγνώριση προσώπου με 66,9% επιτυχία.

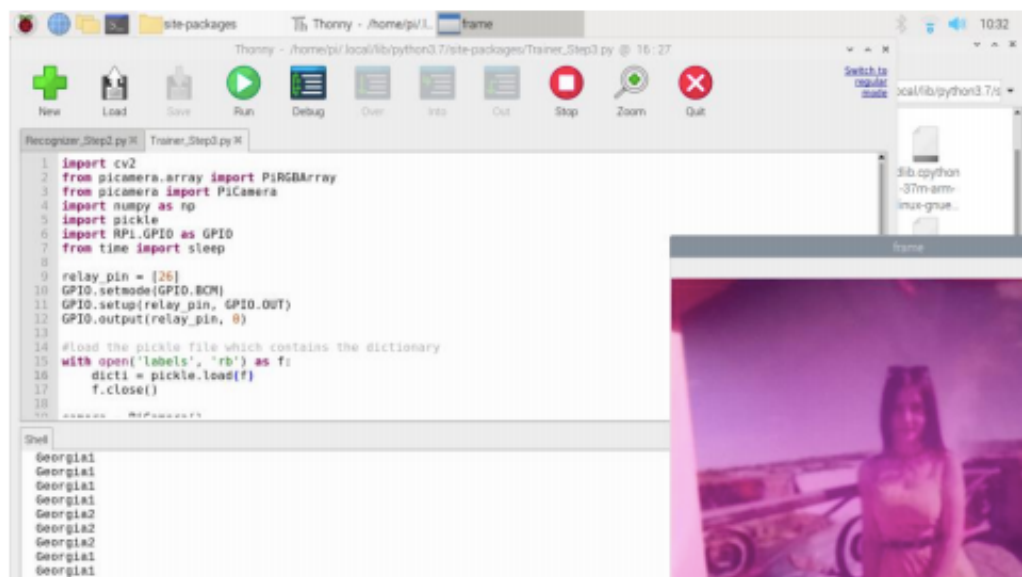


Εικόνα 6.5.4: Αναγνώριση προσώπου με 57,7% επιτυχία.

Την επόμενη φορά, επιχειρήθηκαν δοκιμές του συστήματος χρησιμοποιώντας εικόνες του προσώπου μου με τη βοήθεια ενός κινητού. Το σύστημα όπως αναλύθηκε νωρίτερα, θα έπρεπε να δουλέψει έστω και με μικρό ποσοστό ακρίβειας. Στις περιπτώσεις που δεν φορούσα γυαλιά και στις προσπάθειες με χρήση αναλογικών φωτογραφιών μου, το σύστημα δεν κατάφερε να αναγνωρίσει το πρόσωπο στις εικόνες ακόμη και με τη βοήθεια περισσότερης φωτεινότητας. Φυσικά αυτά τα αποτελέσματα μπορεί να είναι το επιθυμητό, για λόγους ασφαλείας (κυρίως), δεν είναι όμως το αναμενόμενο. Δείχνοντας στο σύστημα εικόνα χωρίς γυαλιά (εικόνα 6.5.4) αλλά και αναλογικές φωτογραφίες (εικόνα 6.5.5) λοιπόν, το σύστημα δεν μπόρεσε να αναγνωρίσει με επιτυχία το πρόσωπο

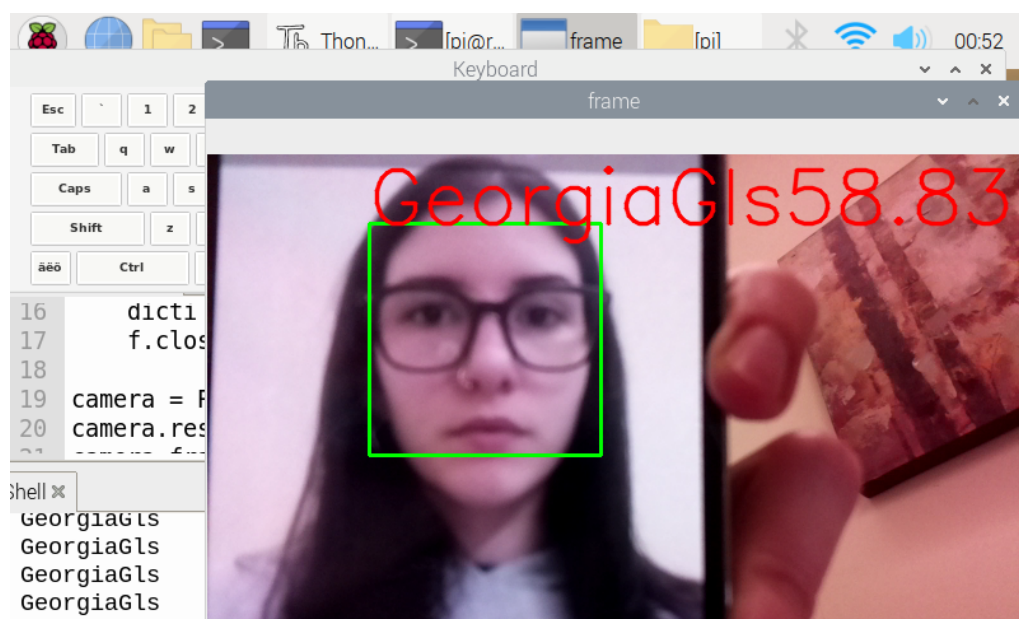


Εικόνα 6.5.4: Προσπάθειες αναγνώρισης προσώπου με χρήση εικόνες από κινητό, πρόσωπο χωρίς γυαλιά



Εικόνα 6.5.5: Προσπάθειες αναγνώρισης προσώπου με χρήση αναλογικής εικόνας

Όταν όμως το πρόσωπο είχε γυαλιά το σύστημα κατάφερε να αναγνωρίσει το πρόσωπο με περισσότερη ευκολία (εικόνα 6.5.6) αλλά χωρίς αυτό να σημαίνει ότι η ακρίβεια ήταν αρκετά καλή, αφού το σύστημα και πάλι δυσκολευόταν να εντοπίσει το πρόσωπο μου κάνοντας την ηλεκτρομαγνητική κλειδαριά να ανοιγοκλείνει συνεχώς. Πιστεύω πως ακόμη και χωρίς γυαλιά θα ήταν εύκολο στο σύστημα να αναγνωρίσει το πρόσωπο ίσως όμως υπό άλλες συνθήκες (πχ ακόμη καλύτερος φωτισμός, τοποθέτηση μαλλιών, ανάλυση εικόνας και άλλα).



Εικόνα 6.5.6: Προσπάθειες αναγνώρισης προσώπου με χρήση εικόνες από κινητό, πρόσωπο με γυαλιά

Ένα συμπέρασμα των παραπάνω προσπαθειών είναι ότι μπορεί το σύστημα να απέτυχε στην αναγνώριση λόγω φωτισμού ή ακόμη και κάμερας διότι, όπως φαίνεται στην εικόνα 6.5.5, η ποιότητα της εικόνας δεν είναι αρκετά καλή για να συμβεί αυτό που περιμέναμε, δηλαδή το σύστημα να λειτουργήσει κανονικά. Βέβαια όπως και με εικόνες από το κινητό, αν υπήρχε καλύτερη φωτεινότητα ή η εικόνα ήταν πιο κοντινή (δηλαδή εικόνα μόνο προσώπου ή έστω πορτρέτο) τότε ίσως το αποτέλεσμα να ήταν διαφορετικό.

Για να φτιάξουμε όμως ένα σύστημα άκρως ασφαλές και 3D χρειάζονται αισθητήρες που είναι δύσκολο να βρεθούν, ακριβοί αλλά και δύσκολο να υλοποιηθούν διότι είναι αρκετά περίπλοκοι.

Υπάρχουν 3D αισθητήρες σε συγκεκριμένα κινητά που είναι ικανά για αναγνώριση προσώπου. Για τη λήψη πληροφοριών σχετικά με το σχήμα ενός προσώπου (όπως το περίγραμμα των ματιών, η μύτη, το πηγούνι και γενικά ιδιαίτερα χαρακτηριστικά που βρίσκονται πάνω σε πρόσωπο) οι αισθητήρες προβάλλουν δομημένο φως στο πρόσωπο του χρήστη. Υπάρχουν τουλάχιστον 12 αισθητήρες που τοποθετούνται μέσα σε τσιπ (CMOS) και κάθε ένας από αυτούς λαμβάνει από ένα διαφορετικό μέρος του φάσματος. Φυσικά η ασφάλεια δεν είναι το μόνο πλεονέκτημα 3D αναγνώρισης, άλλο ένα βασικό είναι ότι οι αλλαγές φωτισμού δεν επηρεάζουν στην αναγνώριση προσώπου, κάτι που στο σύστημα αυτό (το οποίο είναι 2D) έχει πολύ μεγάλο ρόλο διότι όσο πιο καθαρή και φωτεινή είναι η εικόνα του προσώπου του χρήστη τόσο καλύτερο θα είναι το αποτέλεσμα.

Εφόσον δεν είναι εύκολο να χρησιμοποιηθεί αισθητήρας για τρισδιάστατη αναγνώριση μια όχι και τόσο ασφαλής λύση για το σύστημα ώστε να μη μπορεί να χρησιμοποιηθεί εικόνα μας είναι η χρήση κάμερας χαμηλής ποιότητας (όπως στη περίπτωση του συστήματος αυτής της εργασίας). Πιστεύω αυτός είναι ο κύριος λόγος που το σύστημα δεν μπόρεσε να αναγνωρίσει τις εικόνες του προσώπου με τόση επιτυχία, (με και χωρίς γυαλιά) από κινητό αλλά και από αναλογικές εικόνες.

VII. ΚΕΦΑΛΑΙΟ 7

1. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ ΠΕΡΑΙΤΕΡΩ ΑΝΑΠΤΥΞΗΣ

Το σύστημα ασφάλειας που δημιουργήθηκε σίγουρα δεν είναι αρκετό ώστε να μπορεί να χρησιμοποιηθεί με σκοπό την βέλτιστη ασφάλεια ενός χώρου. Όμως, αναμφισβήτητα μπορεί να χρησιμοποιηθεί για πιο απλούς σκοπούς όπως για λιγότερο σημαντικούς χώρους που θα θέλαμε να κρατήσουμε ασφαλή.

Για να δημιουργήσουμε ένα ακόμη πιο ασφαλές σύστημα που να πετυχαίνει καλύτερη ασφάλεια (μεγαλύτερη από 70%), ένα 3D σύστημα θα ήταν η βέλτιστη επιλογή. Για παράδειγμα αν είχαμε υλικά όπως το Face ID της Apple, το οποίο δεν μπορεί να ξεγελασθεί με εικόνες -όπως μπορεί το σύστημα που δημιουργήθηκε στη διπλωματική- αλλά όπως και άλλα συστήματα ασφαλείας, για παράδειγμα το Face-unlock των Android, η αναγνώριση προσώπου θα ήταν πολύ πιο γρήγορη και χωρίς ο φωτισμός να θεωρείται πρόβλημα. Δυστυχώς όμως αυτά τα υλικά είτε δεν δίνονται εύκολα στο ευρύ κοινό ή είναι πολύ ακριβά.

Η παρούσα διπλωματική εργασία, όπως αναφέρθηκε αρχικά, έχει σκοπό την παροχή του τρόπου και των μέσων που απαιτούνται για την υλοποίηση ενός μικρού συστήματος ασφάλειας, χρησιμοποιώντας φθηνά υλικά και τεχνολογίες ανοιχτού κώδικα. Το σύστημα που δημιουργήθηκε είναι μόνο ένα μικρό παράδειγμα των δυνατοτήτων που διαθέτουμε με αυτές τις τεχνολογίες. Από χώρους ασφάλειας έως και αυτοματοποίηση κάποιων καθημερινών εργασιών μας (γνωστό και ως IoT). Εν ολίγης, το σύστημα αυτό δεν είναι άριστο και ούτε αρκετά αξιόπιστο• οι δυνατότητες που έχουμε με τα υλικά που χρησιμοποιήθηκαν αλλά και με το λογισμικό που δημιουργήθηκε είναι απεριόριστες και πολύ καινοτόμες.

Ως μελλοντική μελέτη, θα μπορούσε να κριθεί σκόπιμη η υλοποίηση και εφαρμογή των βασικών αλγορίθμων αναγνώρισης προσώπου, της παρούσας εργασίας, που όπως περιγράφεται και στο τρίτο κεφάλαιο, θα ήταν καλύτερο να δοκιμαστούν στις ίδιες βάσεις προσώπων για πιο έγκυρα αποτελέσματα.

VIII. ΠΑΡΑΡΤΗΜΑ

1. ΚΩΔΙΚΑΣ ΡΥΤΗΘΝ

i. Face Recognition

Για το πρώτο κώδικα που αφορά την ανίχνευση προσώπου και αποθήκευση εικόνων(του προσώπου), πρώτο μου βήμα ήταν να εισάγω τα απαιτούμενα πακέτα.

```
#We import the required packages

import cv2

from picamera.array import PiRGBArray

from picamera import PiCamera

import numpy as np

import os

import sys
```

Ύστερα αρχικοποίησα τη κάμερα και προσάρμοσα την ανάλυση σε 30fps (640, 480).

```
camera = PiCamera()

camera.resolution = (640,480)

camera.framerate=30
```

Απέφυγα τη μετατροπή από JPEG σε OpenCV επειδή αυτό θα επιβράδυνε το σύστημα και φόρτωσα το XML αρχείο «haarcascade» το οποίο χρησιμοποιείται για ανίχνευση προσώπων.

```
rawCapture = PiRGBArray(camera, size=(640,480))

faceCascade=cv2.CascadeClassifier("haarcascade_frontalface_de
fault.xml");
```

Ζητάω όνομα για το φάκελο που δημιουργείται. Χρησιμοποιώντας τη λειτουργία «capture_continuous» διαβάζει τα καρέ από τη κάμερα του Raspberry Pi.

```
name=input("What's their Name?")

dirName="./images/"+name

print(dirName)

if not os.path.exists(dirName):

    os.makedirs(dirName)

    print("Directory Created")

else:

    print("Name already exists")

    sys.exit()

count = 1

for frame in camera.capture_continuous(rawCapture,
format="bgr", use_video_port=True):

    if count>30:

        break
```

Αποκτώ πρόσβαση στον ακατέργαστο πίνακα «NumPy» (μέσω του .array) και κάνω μετατροπή του πλαισίου σε κλίμακα του γκρι (“grayscale”) [εικόνα μου].

```
frame = frame.array

gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

Παίρνουμε ορθογώνιες συντεταγμένες της περιοχής του προσώπου. Χρησιμοποίησα αυτές τις συντεταγμένες για να «εξάγω» το πρόσωπο από τις εικόνες και το αποθηκεύω στο φάκελο που δημιούργησα νωρίτερα. Αποθηκεύονται 30 εικόνες.

```
faces = faceCascade.detectMultiScale(gray, scaleFactor =
1.5, minNeighbors = 5)

for (x, y, w, h) in faces:
```



```

        roiGray = gray[y:y+h, x:x+w]

        fileName = dirName + "/" + name + str(count) + ".jpg"

        cv2.imwrite(fileName, roiGray)

        cv2.imshow("face", roiGray)

        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0),
2)

        count += 1

```

Το `cv2.imshow()` χρησιμοποιείται για να εμφανίσει την εικόνα στην οθόνη. Το «`cv2.waitKey()`» περιμένει μέχρι να λάβει ένα χαρακτήρα από το πληκτρολόγιο και ύστερα συνεχίζει.

```

        cv2.imshow('frame', frame)

        key = cv2.waitKey(1)

```

Καθαρίζουμε τη ροή για τη προετοιμασία του επόμενου καρέ καλώντας το `truncate(0)`. Η ανίχνευση προσώπων και η συλλογή δεδομένων έχουν πλέον ολοκληρωθεί.

```

#we clear the stream in preparation for the next frame by
calling truncate(0)

        rawCapture.truncate(0)

        if key==27:

            break

cv2.destroyAllWindows()

```

ii. Training Recognizer

Ο δεύτερος κώδικας αφορά την εκπαίδευση του recognizer. Αρχικά εισάγουμε τις βιβλιοθήκες

```
import os
import numpy as np
from PIL import Image
import cv2
import pickle
```

Με χρήση του LBPH (Local Binary Patterns Histograms) κάνουμε την αναγνώριση προσώπου. Περιέχεται μέσα στο πακέτο του OpenCV. Το φορτώνουμε ως εξής

```
faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
recognizer = cv2.face.LBPHFaceRecognizer_create()
```

Η πρώτη γραμμή έχει το μονοπάτι του τρέχοντος αρχείου ενώ παρακάτω είναι το αρχείο που περιέχει τις εικόνες που τραβήχτηκαν νωρίτερα.

```
baseDir = os.path.dirname(os.path.abspath(__file__))
#directory where the image directories are present.
imageDir = os.path.join(baseDir, "images")

currentId = 1
labelIds = {}
yLabels = []
xTrain = []
```

Γίνεται μετάβαση σε κάθε αρχείο εικόνων που έχω δημιουργήσει και αναζητά την εικόνα μου. Αν βρεθεί εικόνα που να ταιριάζει τη δικιά μου τότε γίνεται μετατροπή της σε πίνακα NumPy.

```
for root, dirs, files in os.walk(imageDir):
    print(root, dirs, files)
    for file in files:
        print(file)
        if file.endswith("png") or file.endswith("jpg"):
            path = os.path.join(root, file)
            label = os.path.basename(root)
            print(label)

            if not label in labelIds:
```

```

        labelIds[label] = currentId
        print(labelIds)
        currentId += 1

    id_ = labelIds[label]
    pilImage = Image.open(path).convert("L")
    imageArray = np.array(pilImage, "uint8")

```

Γίνεται μια δεύτερη ανίχνευση προσώπου για να σιγουρευτώ ότι υπάρχουν οι σωστές εικόνες.

```

    #perform the face detection again to make sure we have the
    right images
        faces = faceCascade.detectMultiScale(imageArray,
        scaleFactor=1.1, minNeighbors=5)
        for (x, y, w, h) in faces:
            roi = imageArray[y:y+h, x:x+w]
            xTrain.append(roi)
            yLabels.append(id_)

```

Αποθήκευση λεξικού, το οποίο περιέχει τα ονόματα των αρχείων που δημιουργώ.

```

    #store the dictionary which contains the directory names and
    label IDs
    with open("labels", "wb") as f:
        pickle.dump(labelIds, f)
        f.close()

```

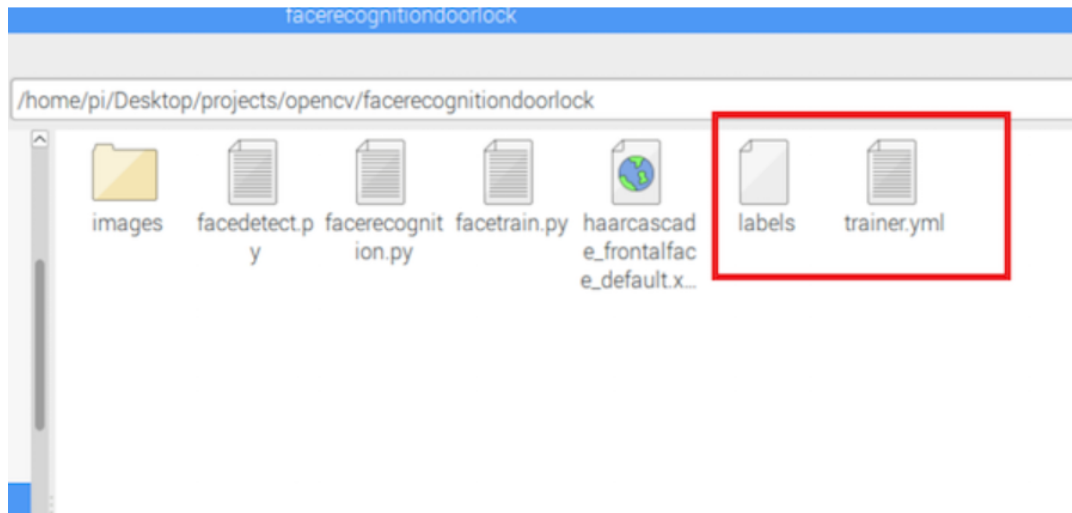
Εκπαίδευση των δεδομένων και αποθήκευση του αρχείου.

```

recognizer.train(xTrain, np.array(yLabels))
recognizer.save("trainer.yml")
print(labelIds)

```

Αυτός ο κώδικας δημιουργεί ένα αρχείο trainer.yml και ονομάζει τα αρχεία που χρησιμοποιούμε στον κωδικό αναγνώρισης.



iii. Recognizer

Ο στόχος αυτού του κώδικα είναι να ελέγξει αν το πρόσωπο στην κάμερα ταιριάζει με κάποιο από τα πρόσωπα που έχουν αποθηκευτεί στο αρχείο «images». Αν βρεθεί εικόνα που να ταιριάζει τότε το Raspberry Pi θα δώσει σήμα στο ρελέ.

Ξεκινάω όπως και πριν, εισάγοντας τις βιβλιοθήκες και αρχικοποιώντας κάποια τιμές.

```
import cv2
from picamera.array import PiRGBArray
from picamera import PiCamera
import numpy as np
import pickle
import RPi.GPIO as GPIO
from time import sleep
relay_pin = [26]
GPIO.setmode(GPIO.BCM)
GPIO.setup(relay_pin, GPIO.OUT)
GPIO.output(relay_pin, 0)
```

Αρχικά φορτώνω το αρχείο «pickle» το οποίο περιέχει το λεξικό (Το Pickle χρησιμοποιείται για σειριοποίηση και απο-σειριοποίηση δομών αντικειμένων Python, που ονομάζονται επίσης marshalling ή flattening).

```
with open('labels', 'rb') as f:
    dicti = pickle.load(f)
    f.close()

camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 30
rawCapture = PiRGBArray(camera, size=(640, 480))
```

Φορτώνω το ταξινομητή που θα ανιχνεύει τα πρόσωπα, τον recognizer που θα προβλέπει τα πρόσωπα όπως και τα εκπαιδευμένα δεδομένα.

```
faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer.read("trainer.yml")

font = cv2.FONT_HERSHEY_SIMPLEX
```

Παρακάτω ο κώδικας διαβάζει το πλαίσιο, το μετατρέπει σε κλίμακα του γκρι (grayscale) και ψάχνει για το πρόσωπο στις εικόνες.

```
for frame in camera.capture_continuous(rawCapture,
format="bgr", use_video_port=True):
    frame = frame.array
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(gray, scaleFactor =
1.5, minNeighbors = 5)
    for (x, y, w, h) in faces:
        roiGray = gray[y:y+h, x:x+w]
```

Ο recognizer θα μας δώσει το όνομα του προσώπου (σύμφωνα με το όνομα που δώσαμε εμείς νωρίτερα στο πρώτο κώδικα) και το ποσοστό εμπιστοσύνης. Το ποσοστό εμπιστοσύνης αποτελεί κρίσιμο συστατικό των συστημάτων ανίχνευσης και σύγκρισης προσώπων. Συστήματα όπως αυτά κάνουν προβλέψεις για το εάν ένα πρόσωπο υπάρχει σε μια εικόνα ή ταιριάζει με ένα

πρόσωπο σε μια άλλη εικόνα, με αντίστοιχο επίπεδο εμπιστοσύνης στην πρόβλεψη, θα μπορούσε κανείς να πει ότι είναι ένα ποσοστό ακρίβειας).

```
id_, conf = recognizer.predict(roiGray)
for name, value in dicti.items():
    if value == id_:
        print(name)
```

Παρακάτω ελέγγω αν το ποσοστό εμπιστοσύνης είναι κάτω από 70. Αν ναι τότε η κλειδαριά θα ανοίξει, αλλιώς θα παραμείνει κλειστή. Τέλος, δημιουργώ ένα ορθογώνιο στην αρχική εικόνα και γράφω το όνομα πάνω από το ορθογώνιο.

```
if conf <= 70:
    GPIO.output(relay_pin, 1)
    cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255,
0), 2)
    cv2.putText(frame, name + str(conf), (x, y),
font, 2, (0, 0, 255), 2, cv2.LINE_AA)

else:
    GPIO.output(relay_pin, 0)

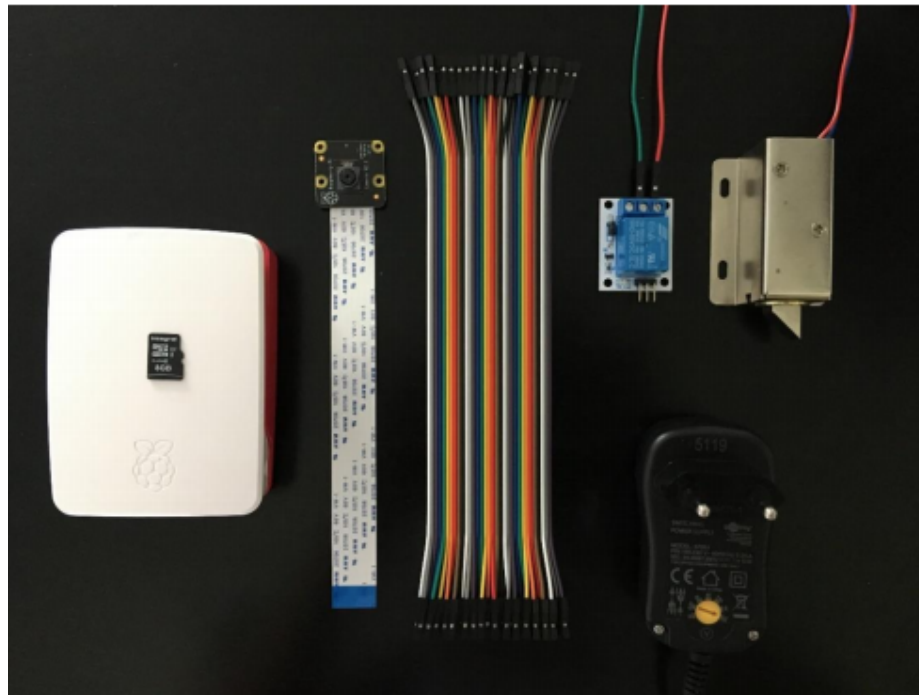
cv2.imshow('frame', frame)
key = cv2.waitKey(1)

rawCapture.truncate(0)

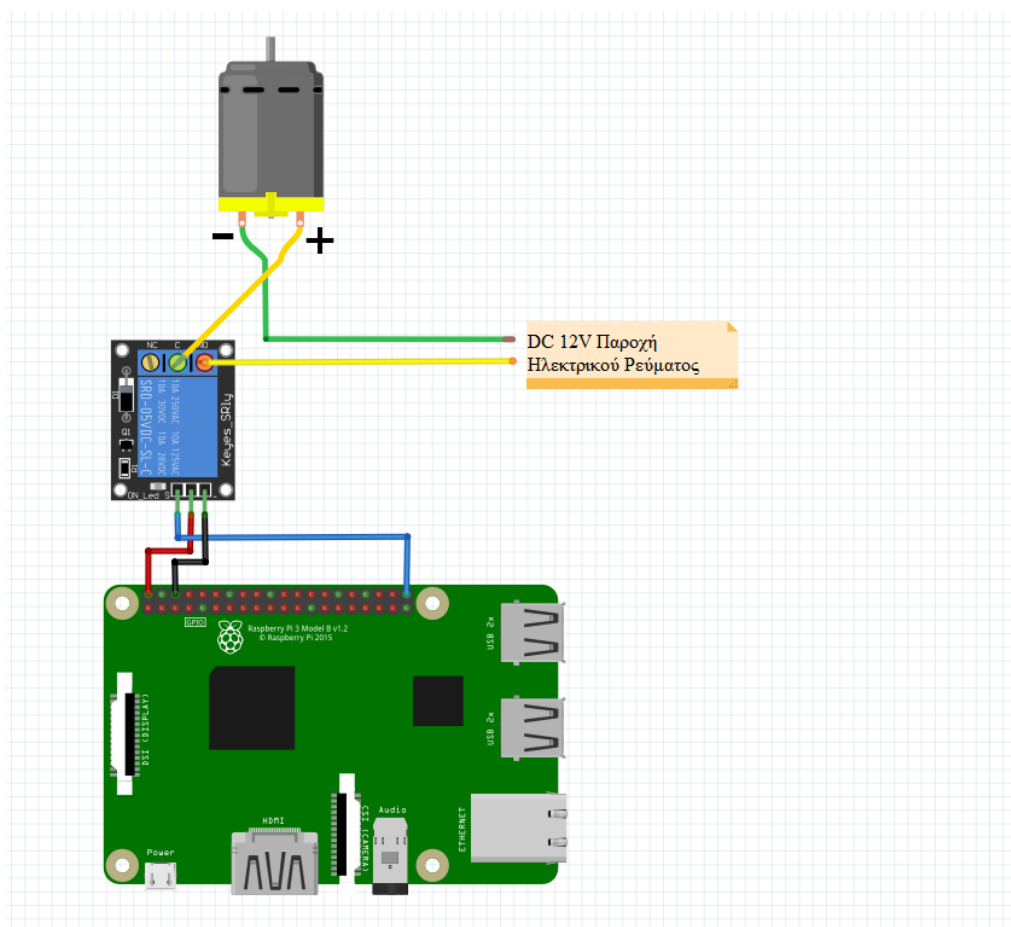
if key == 27:
    break

cv2.destroyAllWindows()
```

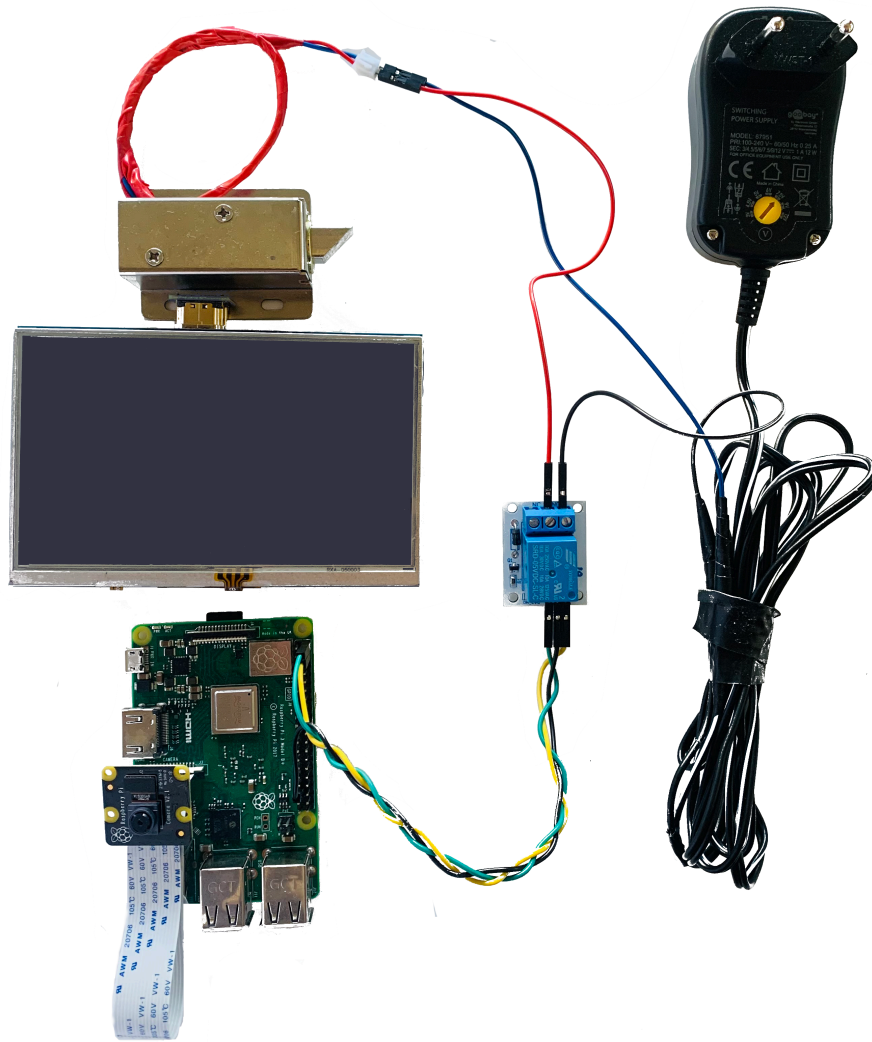
2. ΕΙΚΟΝΕΣ



Πρόσθετη εικόνα 1: Συγκέντρωση υλικών



Πρόσθετη Εικόνα 2: Διάγραμμα συνδέσεων συστήματος.



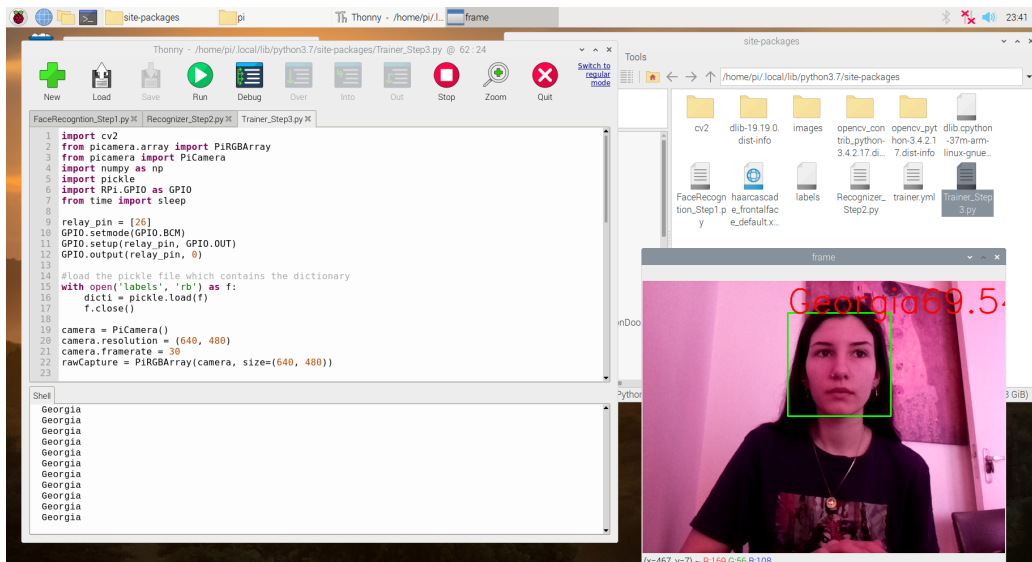
Πρόσθετη εικόνα 3: Το τελικό project δείχνοντας τις συνδέσεις.


```
pi@raspberrypi:~$ pip3 install opencv-python==3.4.6.27
Defaulting to user installation because normal site-packages is not writeable
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting opencv-python==3.4.6.27
  Downloading https://www.piwheels.org/simple/opencv-python/opencv_python-3.4.6.27-cp37-cp37m-linux_armv7l.whl (9.6 MB)
    |-----| 9.6 MB 38 kB/s
Requirement already satisfied: numpy>=1.16.2 in /usr/local/lib/python3.7/dist-packages (from opencv-python==3.4.6.27) (1.18.1)
Installing collected packages: opencv-python
Successfully installed opencv-python-3.4.6.27
pi@raspberrypi:~$
```

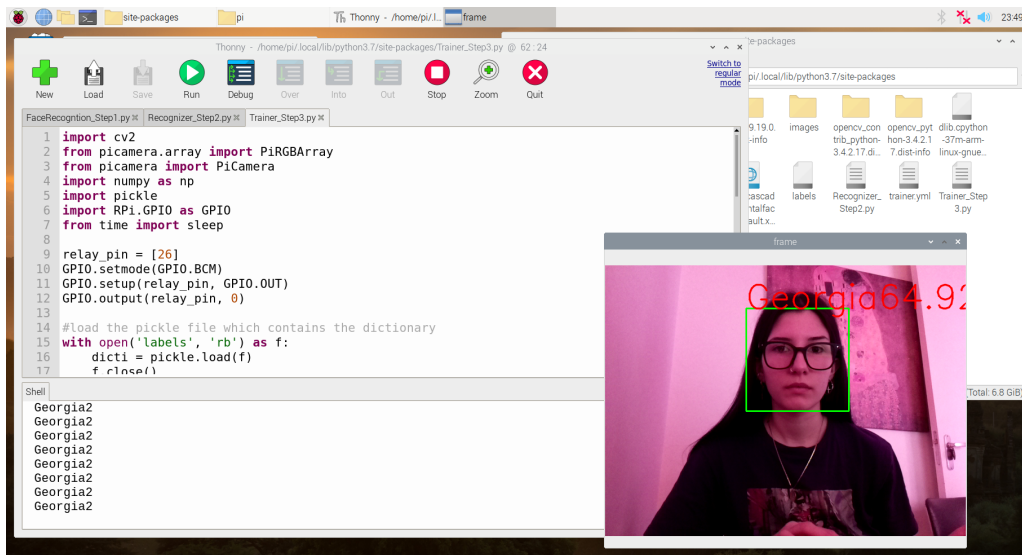
Πρόσθετη εικόνα 4: Επιθυμητό αποτέλεσμα στο τερματικό του Raspberry Pi.



Πρόσθετη εικόνα 5: Παράδειγμα συνόλου 30 εικόνων στον φάκελο Georgia



Πρόσθετη εικόνα 6: Επιτυχία αναγνώρισης 69,5% με τα μαλλιά κάτω.



Πρόσθετη εικόνα 6: Επιτυχία αναγνώρισης 64,92% με τα μαλλιά κάτω ενώ φορούσα γυαλιά οράσεως.

3. ΟΡΟΙ

¹OpenCV: Δημιουργήθηκε το 1999 από τον Gary Bradski.

²ONNX: Είναι μία συλλογή από προ-εκπαιδευμένα State-of-the-Art μοντέλα μηχανικής μάθησης.

IX. ΒΙΒΛΙΟΓΡΑΦΙΑ

1. G. Huang, M. Mattar, H. Lee, and E. G. Learned-Miller, "Learning to align from scratch," in *Advances in Neural Information Processing Systems*, 2012.
2. Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
3. F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", 2015.
4. T. Hassner, S. Harel, E. Paz, and R. Enbar, "Effective face frontalization in unconstrained images", 2014.
5. OpenCV Face Recognition <https://www.pyimagesearch.com/2018/09/24/opencv-face-recognition/>
6. "Μελέτη Αλγορίθμων Αναγνώρισης Προσώπου, Υλοποίηση, Εφαρμογή και Σύγκριση της απόδοσής τους" Πανεπιστήμιο Μακεδονίας Μπακάλης Αθανάσιος, Ιούνιος 2010
7. M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.
8. A. Papoulis and S U Pillai, *Probability, Random Variables and Stochastic Processes with Errata Sheet*, 4th ed.: McGraw-Hill, 2002.
9. Μπακάλης Αθανάσιος, "Μελέτη Αλγορίθμων Αναγνώρισης Προσώπου, Υλοποίηση, Εφαρμογή και Σύγκριση της απόδοσής τους" Πανεπιστήμιο Μακεδονίας Οικονομικών και Κοινωνικών Επιστημών, 2010
10. Independent Component Analysis: Algorithms and ApplicationsAapo Hyvärinen and Erkki OjaNeural Networks Research CentreHelsinki University of Technology P.O. Box 5400, FIN-02015 HUT, Finland
11. W Bledsoe, "The Model Method in Facial Recognition," Panoramic Research, Inc, Palo Alto, California., Technical Report PRI 15, 1964.

12. W. Bledsoe and H. Chan, "A Man-Machine Facial Recognition System-Some Preliminary Results," Panoramic Research, Inc, Palo Alto, California., Technical Report PRI 19A, 1965.
13. W. Bledsoe, "Semiautomatic Facial Recognition," Stanford Research Institute, Menlo Park, California, Technical Report SRI Project 6693, 1968.
14. L. Wiscott, J.M. Fellous, N. Kuiger, and C. von Der Malrsburg, "Face recognition by elastic bunch graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence In Pattern Analysis and Machine Intelligenc*, vol. 19, no. 7, pp. 775-779, July 1997.
15. F. Samaria and S. Young, "HMM based architecture for face identification," *Image and Computer Vision*, vol. 12, pp. 537-583, October 1994.
16. V. Nefian and H. Hayes III, "Hidden Markov Models for face recognition," in *IEE International Conference on Acoustic speech and Signal Processing*, 1998.
17. F. Samaria and S. Young, "HMM based architecture for face identification," *Image and Computer Vision*, vol. 12, pp. 537-583, October 1994.
18. Rabiner. L R 'A tutorial on hidden Markov models and selected applications in speech recognition', Proc. IEEE, Vol 77 No 2 (January 1989) pp 257-286
19. Computer Vision Papers, <http://www.cvpapers.com/>
20. *Open Source Computer Vision Library Reference Manual-intel; Gary Bradski & Adrian Kaehler O'Reilly, 2008*
21. Shervin Emami, Valentin Petrit Suciu: "Face Recognition Using OpenCV", 2012
22. Face Recognition Homepage, <http://www.face-rec.org/algorithms/>
23. Wikipedia, Three-dimensional face recognition, http://en.wikipedia.org/wiki/Three-dimensional_face_recognition

24. Wikipedia, Active appearance model, http://en.wikipedia.org/wiki/Active_appearance_model
25. Wikipedia, Histogram equalization, http://en.wikipedia.org/wiki/Histogram_equalization
26. Paul Viola and Michael Jones in their paper "Rapid Object Detection using a Boosted **Cascade** of Simple Features" in 2001
27. Βικιπαίδεια <https://en.wikipedia.org/wiki/Raspbian>