



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΒΙΟΜΗΧΑΝΙΚΗΣ ΣΧΕΔΙΑΣΗΣ & ΠΑΡΑΓΩΓΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Εφαρμογές Ρομποτικής Όρασης»

Επιβλέπων Καθηγητής: κος Νικολάου Γρηγόριος

**ΓΚΑΓΚΑΡΑΣ ΣΠΥΡΙΔΩΝ - ΓΕΩΡΓΙΟΣ (Α.Μ. 41686)
ΤΕΡΖΑΚΗΣ ΠΑΝΑΓΙΩΤΗΣ (Α.Μ. 41712)**

Αθήνα

2018

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/Η κάτωθι υπογεγραμμένος/η Γεωργίας Σουρίδου-Γεωργίου Τριζάκης Παναγιώτης του Σταύρου, φοιτητής του Τμήματος Μηχανικών Διαχείρισης και Λογιστικής και Πληροφορικής του Πανεπιστημίου Δυτικής Αττικής, πριν αναλάβω την εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

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

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

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα, σε περίπτωση που το Ίδρυμα του έχει απονεμίσει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασή της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση Π.Ε με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού βμήνου από την ημερομηνία ανάθεσής της.

Ο Δηλών Τριζάκης Παναγιώτης Ημερομηνία 28-06-2018

Ο Δηλών Γεωργίας Σουρίδου-Γεωργίου

Γ

Περίληψη

Στην παρούσα εργασία παρουσιάζονται δύο συστήματα ρομποτικής όρασης. Το πρώτο σύστημα αφορά την ανίχνευση και την αναγνώριση προσώπου σε πραγματικό χρόνο. Το δεύτερο σύστημα αφορά την ανίχνευση και τον έλεγχο της στάθμης υγρού σε μπουκάλι καθώς και τον υπολογισμό των χιλιοστογραμμαρίων του υγρού που περιέχονται κάθε φορά σε αυτό. Στόχος και στις δύο εφαρμογές είναι η μέγιστη δυνατή αποτελεσματικότητα με το χαμηλότερο δυνατό κόστος. Στο πρώτο μέρος της πτυχιακής εργασίας αναλύονται οι απαραίτητες θεωρητικές έννοιες στις οποίες βασίστηκαν οι συγγραφείς για την τεκμηρίωση της έρευνάς τους. Συγκεκριμένα, αξιοποιήθηκαν οι μέθοδοι των Viola & Jones στην ανίχνευση προσώπου, τα Τοπικά Δυαδικά Πρότυπα στην αναγνώριση προσώπου, ενώ η ανίχνευση της στάθμης έγινε με βάση την Εικόνα Ακρών. Στη συνέχεια, παρουσιάζονται τα υλικά που αξιοποιήθηκαν στην έρευνα και ο σκοπός χρήσης τους. Παρατίθενται οι κώδικες που αναπτύχθηκαν για τις δύο εφαρμογές ρομποτικής όρασης και σχολιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν μέσω της βιβλιοθήκης OpenCV και της γλώσσας προγραμματισμού Python. Στην ενότητα των αποτελεσμάτων, παρουσιάζονται φωτογραφίες από την εφαρμογή των δύο κωδίκων και επεξηγούνται. Έμφαση δίνεται στα συμπεράσματα από την εφαρμογή των κωδίκων. Συζητούνται οι προϋποθέσεις επιτυχίας της έρευνας, οι δυσκολίες που συναντήθηκαν καθώς και οι τρόποι με τους οποίους αυτές ξεπεράστηκαν. Τέλος, αναφέρονται κάποιοι τομείς στους οποίους θα μπορούσαν να έχουν εφαρμογή τα συστήματα της πτυχιακής εργασίας.

Λέξεις κλειδιά: Ρομποτική Όραση, OpenCV, Python, Ανίχνευση Προσώπου, Αναγνώριση Προσώπου, Καταρράκτης Ταξινομητών, Μέθοδος Viola & Jones, Μέθοδος Τοπικών δυαδικών προτύπων, Έλεγχος Στάθμης, Ανίχνευση Ακρών με Canny, Πιθανοτικός Μετασχηματισμός Γραμμής Hough.

Abstract

In the present study two applications of robotic vision are presented. The first system aims at face detection and recognition in real time. The second system aims at the detection and control of the liquid level in a bottle as well as the estimation of the liquid's milligrams that are contained in it. The basic goal for both applications is the maximum efficiency with the lowest possible cost. In the first part of this diploma study, the necessary theoretical concepts, on which the authors documented their research, are explored. More specifically, the authors deployed the method of Viola & Jones for face detection, the Local Binary Patters for face recognition and the control of the liquid level was achieved through the Edge Image. Then, the materials used and their goal of use is presented. The codes that were developed for both systems are cited. Also, the functions used through the OpenCV library and the programming language Python are explored. In the section of Results, photos of the implementation of the codes are presented and explained. Emphasis is placed on the conclusions from the implementation of the codes. The conditions that would ensure the success of this research are discussed along with the difficulties faced and the ways that they were surpassed. Finally, some applications to which the systems developed in this diploma research could be applied are mentioned.

Key Words: Robotic Vision, OpenCV, Python, Face Detection, Face Recognition, Cascade Classifier, Viola & Jones Method, Local Binary Patters, Control of Liquid Level, Edge Detection, Canny, Probabilistic Hough Lines Transform

Πίνακας Περιεχομένων

Περίληψη	2
Abstract	4
Κατάλογος Εικόνων	8
Ευχαριστίες	10
Εισαγωγή	12
Κεφάλαιο 1: Ανίχνευση Και Αναγνώριση Προσώπου	16
1.1. Μέθοδοι Ανίχνευσης	16
1.1.1. Εισαγωγή	16
1.1.2. Ταξινομήσεις Των Μεθόδων Ανίχνευσης	16
1.1.3. Μέθοδοι Ανίχνευσης Βασισμένες Στην Εμφάνιση	18
1.2. Μέθοδοι Αναγνώρισης	20
1.2.1. Εισαγωγή	20
1.2.2. Διαδικασία – Πορεία Αναγνώρισης Προσώπων	21
1.2.3. Ταξινόμηση Των Μεθόδων Αναγνώρισης Προσώπων	22
1.2.4. Επισκόπηση Των Βασικότερων Μεθόδων Ανίχνευσης Προσώπων	23
Κεφάλαιο 2: Ανίχνευση Προσώπου Κατά Viola & Jones Και Αναγνώριση Κατά LBPH Με Χρήση OpenCV Και Python	28
2.1. Μέθοδος Κατά Viola & Jones	28
2.1.1. Εισαγωγή	28
2.1.2. Δημιουργία Ολοκληρωτικής Εικόνας	29
2.1.3. Υπολογισμός Χαρακτηριστικών	30
2.1.4. Επιλογή Χαρακτηριστικών – Εκπαίδευση Του Ταξινομητή Με Τον Adaboost	30
2.1.5. Ταξινόμηση Με Έναν Καταρράκτη (Cascade) Ταξινομητών	32
2.1.6. Μέθοδος Κατά Lienhart	34
2.2. Η Μέθοδος Των Τοπικών Δυναδικών Προτύπων	34
2.3. Βιβλιοθήκη OpenCV	37
2.4. Γλώσσα Προγραμματισμού Python	37
2.5. Ανίχνευση Και Αναγνώριση Προσώπου Με Την OpenCV Και Python.....	38
Κεφάλαιο 3: Ανίχνευση Στάθμης Υγρού Με Χρήση OpenCV Και Python	41
3.1. Εισαγωγή	41

3.2. Επιλογή Περιοχής Ενδιαφέροντος (ROI)	42
3.3. Εφαρμογή Της Συνάρτησης Canny Edge Detector	42
3.3.1. Μείωση Του Θορύβου (Noise Reduction)	43
3.3.2. Εύρεση Της Κλίσης Έντασης Της Εικόνας (Finding Intensity Gradient Of The Image)	44
3.3.3. Μη-Μέγιστη Συμπίεση (Non-Maximum Suppression).....	45
3.3.4. Υστέρηση Κατωφλίων (Hysteresis Thresholding)	45
3.4. Εφαρμογή Του Hough Lines Transform	46
Κεφάλαιο 4: Σκοπός Πτυχιακής Εργασίας	50
Κεφάλαιο 5: Μεθοδολογία Έρευνας	51
5.1. Υλικά	51
5.1.1. Raspberry Pi 3 Model B	51
5.1.2. Raspberry Pi Camera	52
5.1.3. Εικόνες Προσώπων	52
5.1.4. Κατασκευή Για Ανίχνευση Στάθμης Υγρού	53
5.2. Κώδικας Ανίχνευσης Και Αναγνώρισης Προσώπου	53
5.2.1. Δημιουργία Βάσης Δεδομένων Προσώπων	54
5.2.2. Προετοιμασία Βάσης Για Την Εκπαίδευση Του LBP Classifier	57
5.2.3. Εκπαίδευση Και Ανίχνευση-Αναγνώριση Προσώπου Σε Πραγματικό Χρόνο	59
5.3. Κώδικας Ανίχνευσης Και Ελέγχου Στάθμης Υγρού Σε Μπουκάλι	63
Κεφάλαιο 6: Αποτελέσματα και Συμπεράσματα.....	67
6.1. Αποτελέσματα και Συμπεράσματα Για Την Ανίχνευση Και Αναγνώριση Προσώπων	67
6.1.1. Αποτελέσματα	67
6.1.2. Συμπεράσματα.....	70
6.2. Αποτελέσματα-Συμπεράσματα Για Την Ανίχνευση Και Έλεγχο Στάθμης Υγρού Σε Μπουκάλι	71
6.2.1. Αποτελέσματα	71
6.2.2 Συμπεράσματα.....	73
Κεφάλαιο 7: Συζήτηση – Μελλοντική Έρευνα.....	77
Βιβλιογραφία.....	79

Κατάλογος Εικόνων

Εικόνα 1.1. Διάγραμμα Διαδικασίας Αναγνώρισης Προσώπου	21
Εικόνα 2.1. Η Ολοκληρωτική Εικόνα	29
Εικόνα 2.2. Αναπαράσταση Της Ολοκληρωτικής Εικόνας	29
Εικόνα 2.3. Χαρακτηριστικά Haar Των Viola & Jones	30
Εικόνα 2.4. Τα Δύο Πιο Χρήσιμα Χαρακτηριστικά	31
Εικόνα 2.5. Σχηματική Αναπαράσταση Ενός Cascade Ταξινομητή	31
Εικόνα 2.6. Η Διαδικασία Υπολογισμού Των Τιμών Των Pixel Με Τα Τοπικά Δυναδικά Πρότυπα.....	35
Εικόνα 2.7. Κυκλικά Τοπικά Δυναδικά Πρότυπα.....	35
Εικόνα 2.8. Η Κατασκευή Των Τοπικών Ιστογραμμάτων	36
Εικόνα 2.9. Το Λογότυπο Της OpenCV.....	37
Εικόνα 2.10. Το Λογότυπο Της Python	38
Εικόνα 3.1. Εφαρμογή Του Gaussian Blur	44
Εικόνα 3.2. Μη-Μέγιστη Συμπίεση	45
Εικόνα 3.3. Κατηγοριοποίηση Ακρών Με Βάση Τις Τιμές Κατωφλίου.....	46
Εικόνα 3.4. Εικόνα Ακρών (Edge Image).....	46
Εικόνα 3.5. Αναπαράσταση Γραμμής Στο Πολικό Σύστημα Αξόνων	47
Εικόνα 3.6. Τρεις Καμπύλες Που Διασταυρώνονται Σε Ένα Σημείο	47
Εικόνα 3.7. Η Διαδικασία Εύρεσης Γραμμών	48
Εικόνα 3.8. Ανίχνευση Γραμμών Με Τον Προοδευτικό Πιθανοτικό Μετασχηματισμό Γραμμών Hough	48
Εικόνα 5.1. Raspberry Pi 3 Model B.....	51
Εικόνα 5.2. Raspberry Pi Camera.....	52
Εικόνα 5.3. Δείγματα Εικόνων Για Την Αναγνώριση Προσώπου	52
Εικόνα 5.4. Κατασκευή Για Ανίχνευση και Έλεγχο Στάθμης Υγρού.....	53
Εικόνα 5.5. Βάση Δεδομένων.....	54
Εικόνα 6.1.Ανίχνευση-Αναγνώριση Και Με Τους Δύο Συγγραφείς Γνωστούς Στην Βάση .67	
Εικόνα 6.2.Ανίχνευση-Αναγνώριση Με Γνωστό Πρόσωπο Στην Βάση Τον Ένα Συγγραφέα (Πάνο).....	70

Εικόνα 6.3.Ανίχνευση-Αναγνώριση Με Γνωστό Πρόσωπο Στην Βάση Τον Ένα Συγγραφέα (Σπύρο)	70
Εικόνα 6.4.Ανίχνευση-Αναγνώριση Και Με Τους Δύο Συγγραφείς Ως Άγνωστα Πρόσωπα	70
Εικόνα 6.5.Ανίχνευση Αλλά Μη Αναγνώριση Εκτός Ζώνης Αναγνώρισης	70
Εικόνα 6.6. Λανθασμένη Ανίχνευση Με Ορίσματα detectMultiScale(gray, 1.1, 1).....	71
Εικόνα 6.7. Αλλαγή τιμής conf – Εύρος Ζώνης Αναγνώρισης Με Σταθερό Όρισμα rec=cv2.face.LBPHFaceRecognizer_create ().....	71
Εικόνα 6.8. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Άδειο	72
Εικόνα 6.9. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Χαμηλό	72
Εικόνα 6.10. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Σχεδόν Χαμηλό	73
Εικόνα 6.11. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Κανονικό	
Εικόνα 6.12. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Σχεδόν Γεμάτο	73
Εικόνα 6.13. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Γεμάτο, Σταματήστε.....	73
Εικόνα 6.14. Εικόνα Ακρών Με Εφαρμογή Φίλτρου Gauss Και Χωρίς	74
Εικόνα 6.15. Λανθασμένη Ανίχνευση Πίσω Γραμμής Επιφάνειας	75
Εικόνα 6.16. Μη Ανίχνευση Της Στάθμης Του Νερού	76
Εικόνα 6.17. Σωστή Ανίχνευση	76

Ευχαριστίες

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή μου, κύριο Νικολάου Γρηγόριο, για την επιστημονική του καθοδήγηση, η οποία συνέβαλε τα μέγιστα στην επιτυχή ολοκλήρωση της παρούσας εργασίας, και για τη συνεργασία μας στο διάστημα αυτό. Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου για την υποστήριξη που μου προσφέρουν και για την εμπιστοσύνη που μου δείχνουν συνεχώς. Ακόμη, θέλω να ευχαριστήσω ιδιαίτερα τη σύντροφό μου, Μαρία, για τη μεγάλη υποστήριξη που μου παρείχε και τη σημαντική της βοήθεια. Τέλος, ευχαριστώ ιδιαίτερα τον κύριο Σπύρο Γκαγκάρα για την ανταλλαγή ιδεών, τη συνέπεια και την άψογη συνεργασία μας κατά την εκπόνηση της παρούσας πτυχιακής εργασίας.

Τερζάκης Παναγιώτης

Θα ήθελα να ευχαριστήσω το ΑΕΙ ΠΕΙΡΑΙΑ Τ.Τ. στο οποίο ολοκλήρωσα τις σπουδές μου για τις πολύτιμες γνώσεις που μου παρείχε. Επίσης, θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή μου κύριο Νικολάου Γρηγόριο για την σημαντική του βοήθεια κατά την διάρκεια της συνεργασίας μας. Ένα μεγάλο ευχαριστώ ανήκει και στην οικογένεια μου για την στήριξη που μου παρείχε κατά την διάρκεια των σπουδών μου. Τέλος, θα ήθελα να ευχαριστήσω ιδιαίτερα τον κύριο Τερζάκη Παναγιώτη για την άψογη συνεργασία μας καθ' όλη την διάρκεια της πτυχιακής εργασίας και για τις πολύτιμες γνώσεις με τις οποίες και ο ίδιος συνέβαλε στην ολοκλήρωσή της.

Γκαγκάρας Σπυρίδων – Γεώργιος

Εισαγωγή

Η παρούσα εργασία τοποθετείται ερευνητικά στον κλάδο της Ρομποτικής Όρασης (Robotic Vision). Η Ρομποτική Όραση είναι ένας ταχέως αναπτυσσόμενος ερευνητικός τομέας με ποικίλες εφαρμογές τόσο σε επίπεδο καθημερινής ζωής όσο και σε βιομηχανικό επίπεδο.

Η Ρομποτική Όραση σχετίζεται με μεθόδους όρασης που στη συνέχεια στοχεύουν στην εκτέλεση ενεργειών από κάποιο ρομπότ ή μικροϋπολογιστή. Οι μέθοδοι όρασης αναφέρονται στον συνδυασμό αφενός της χρήσης μιας ψηφιακής κάμερας, η οποία αποτελεί και τον κυριότερο αισθητήρα των αντίστοιχων συστημάτων, και αφετέρου υπολογιστικών αλγορίθμων. Το σύστημα Ρομποτικής Όρασης αφού συλλέξει τα οπτικά δεδομένα μέσω της ψηφιακής κάμερας, τα επεξεργάζεται με διάφορες μεθόδους επεξεργασίας εικόνων ή εικονοσειρών. Έτσι, γίνεται δυνατή η εκτέλεση ενεργειών με βάση τα οπτικά δεδομένα.

Σύμφωνα με τον Τζαφέστα (2003), ένας ορισμός της Ρομποτικής Όρασης είναι ο εξής: «Όραση των Ρομπότ ή Ρομποτική Όραση είναι η διεργασία εξαγωγής, χαρακτηρισμού, παράστασης και ερμηνείας πληροφορίας, η οποία προέρχεται από εικόνες του τρισδιάστατου ρομποτικού κόσμου». Η διεργασία αυτή διαρθρώνεται σε 6 βήματα:

1. Αίσθηση: λήψη εικόνων
2. Προεπεξεργασία: μείωση θορύβου και ανάδειξη των λεπτομερειών των εικόνων
3. Κατάτμηση: χωρισμός εικόνων σε αντικείμενα ενδιαφέροντος
4. Περιγραφή: υπολογισμός ιδιοτήτων εικόνων (π.χ. μέγεθος, σχήμα)
5. Αναγνώριση: εύρεση ταυτότητας των αντικειμένων που απεικονίζονται
6. Ερμηνεία: απόδοση σημασίας στα αναγνωρισμένα αντικείμενα [7].

Πρωταρχικός σκοπός της Ρομποτικής Όρασης είναι η δημιουργία ενός μοντέλου του πραγματικού κόσμου που θα βασίζεται σε ψηφιακές εικόνες, καθιστώντας έτσι εφικτή τη λήψη αποφάσεων και εκτέλεση ενεργειών από το ρομπότ, ανάλογα με το προς επίλυση πρόβλημα.

Η Ρομποτική Όραση, ως όρος, συνδέεται στενά με τη Μηχανική Όραση όσο και με την Υπολογιστική Όραση. Η Υπολογιστική Όραση σχετίζεται με την εξαγωγή χαρακτηριστικών από εικόνες. Παράδειγμα της Υπολογιστικής Όρασης είναι η ανίχνευση αντικειμένων μέσα σε εικόνες. Από την άλλη πλευρά, η Μηχανική Όραση σχετίζεται

περισσότερο με συγκεκριμένες εφαρμογές και όχι τόσο με τις τεχνικές μέσω των οποίων αυτές επιτυγχάνονται. Δηλαδή, η Μηχανική Όραση αφορά, για παράδειγμα, την βιομηχανική χρήση του αυτόματου ελέγχου και της πλοήγησης ρομπότ. Επομένως, συνδέεται με την Υπολογιστική Όραση, εφόσον χρησιμοποιεί τεχνικές της, όπως η επεξεργασία εικόνας.

Όπως γίνεται αντιληπτό, η Ρομποτική Όραση ενσωματώνει τεχνικές τόσο της Υπολογιστικής όσο και της Μηχανικής Όρασης, αφού, όπως αναφέρθηκε και στην αρχή του κεφαλαίου, συνδυάζει τη χρήση εικόνων με την αξιοποίηση υπολογιστικών αλγορίθμων για την ανάπτυξη συγκεκριμένων ρομποτικών εφαρμογών.

Η Ρομποτική και η Υπολογιστική Όραση ως τομείς αρχίζουν να αποκτούν ερευνητικό ενδιαφέρον από τη δεκαετία του 1970. Από τότε, βασικός στόχος ήταν η δημιουργία ενός μοντέλου του τρισδιάστατου κόσμου μέσω εικόνων. Έγινε, μάλιστα, γρήγορα σαφές πως αυτό θα ήταν ένα δύσκολο και απαιτητικό έργο. Στη δεκαετία του 1980 η έρευνα εστίασε στις προηγμένες μαθηματικές μεθόδους που θα επέτρεπαν την ποιοτική ανάλυση των εικόνων που λαμβάνουν τα ψηφιακά ρομποτικά συστήματα. Στη δεκαετία του 1990 οι μέθοδοι της Ρομποτικής Όρασης εξελίχθηκαν περαιτέρω με τη συμβολή της Φυσικής και της Οπτικής. Επίσης, αρχίζουν να χρησιμοποιούνται τεχνικές βασισμένες στη Στατιστική, όπως η τεχνική της ανάλυσης κυρίων συνιστωσών (principal component eigenface analysis) για την αναγνώριση προσώπων. Τέλος, στη δεκαετία του 2000 έως σήμερα, υπάρχει ευρεία αξιοποίηση των γραφικών του υπολογιστή στις εφαρμογές Ρομποτικής και Υπολογιστικής Όρασης και γενικότερα παρατηρείται η ανάπτυξη όλο και συνθετότερων αλγορίθμων (π.χ. Νευρωνικά Δίκτυα).

Από τις πολυάριθμες έρευνες που έχουν πραγματοποιηθεί στη διάρκεια αυτών των δεκαετιών, έχουν αναπτυχθεί, λοιπόν, πολλές εφαρμογές Ρομποτικής Όρασης. Ορισμένες από αυτές είναι οι εξής:

- Η ανίχνευση, αναγνώριση και παρακολούθηση αντικειμένων και προσώπων
- Η αυτόματη πλοήγηση
- Η χαρτογράφηση άγνωστων χώρων
- Η οπτική μέτρηση
- Ο αυτόματος έλεγχος
- Τα συστήματα παρακολούθησης
- Οι Εφαρμογές Άμυνας και Παρακολούθησης σε συστήματα ασφαλείας
- Η ασφάλεια στην αυτοκίνηση (π.χ. εντοπισμός απροσδόκητων εμποδίων ή πεζών)

- Ο έλεγχος ποιότητας σε βιομηχανικές εφαρμογές
- Η υποστήριξη εργασιών σε γραμμή παραγωγής (π.χ. απόρριψη ελαττωματικών εξαρτημάτων)
- Το visual servoing.

Μία, λοιπόν, βασική εφαρμογή Ρομποτικής Όρασης είναι η ανίχνευση και αναγνώριση αντικειμένων και προσώπων. Βασιζόμενοι σε αυτό, στην παρούσα εργασία θα εστιάσουμε σε ένα σύστημα ανίχνευσης και αναγνώρισης προσώπων, καθώς και στην ανίχνευση και έλεγχο της στάθμης υγρού με τη χρήση μόνο μιας κάμερας και του μικροεπεξεργαστή Raspberry Pi.

Κεφάλαιο 1 : Ανίχνευση Και Αναγνώριση Προσώπου

1.1 Μέθοδοι Ανίχνευσης

1.1.1. Εισαγωγή

Η ανίχνευση προσώπου (face detection) είναι το πρώτο στάδιο στην ευρύτερη διαδικασία της αναγνώρισης προσώπου. Το σύστημα ανίχνευσης προσώπου καθορίζει σε μεγάλο βαθμό την αποτελεσματικότητα ολόκληρου του συστήματος αναγνώρισης προσώπου δεδομένου ότι αποτελεί το πρώτο βήμα της διαδικασίας [2].

Έχει φανεί ερευνητικά, από τις δεκαετίες μελετών και τις ποικίλες μεθόδους που έχουν αναπτυχθεί, ότι η ανίχνευση προσώπου είναι ένα δύσκολο και πολύπλοκο προς επίλυση πρόβλημα. Η αξιοπιστία του συστήματος ανίχνευσης καθορίζεται από την δυνατότητα του ανιχνευτή να εντοπίζει όλα τα ανθρώπινα πρόσωπα που απεικονίζονται σε μια εικόνα ανεξάρτητα από τα χαρακτηριστικά τους ή τις όποιες άλλες συνθήκες (π.χ. φωτισμού) επικρατούν στην εικόνα. Αναλυτικά, ο ανιχνευτής πρέπει να εντοπίσει τα πρόσωπα ανεξαρτήτως:

- Της θέσης τους
- Του προσανατολισμού τους στο χώρο
- Της στάσης τους (pose)
- Του μεγέθους/κλίμακας τους
- Της έκφρασής τους
- Της ύπαρξης άλλων δοκιμών στοιχείων (π.χ. μαλλιά)
- Του φωτισμού της εικόνας
- Του περιεχομένου της εικόνας
- Και τέλος των συνθηκών αποτύπωσης της εικόνας (π.χ. ανάλυση) [21].

1.1.2. Ταξινομήσεις Των Μεθόδων Ανίχνευσης Προσώπου

Έχουν προταθεί διαφορετικές ταξινομήσεις των μεθόδων ανίχνευσης προσώπου, επειδή δεν υπάρχουν κοινώς αποδεκτά κριτήρια ταξινόμησης. Οι δύο βασικότερες ταξινομήσεις παρουσιάζονται στην ενότητα αυτή.

Η πρώτη ταξινόμηση βασίζεται στις συνθήκες λήψης των εικόνων (scenario). Έτσι, διακρίνεται η ανίχνευση σε ελεγχόμενο περιβάλλον, η ανίχνευση με βάση έγχρωμες εικόνες και η ανίχνευση με βάση κινούμενες εικόνες/βίντεο. Η ανίχνευση σε ελεγχόμενο περιβάλλον είναι η πιο απλή περίπτωση, εφόσον οι φωτογραφίες λαμβάνονται σε συγκεκριμένες συνθήκες φωτισμού και φόντου. Στην ανίχνευση με βάση έγχρωμες εικόνες αξιοποιείται το χρώμα της ανθρώπινης επιδερμίδας για τον εντοπισμό προσώπων. Ωστόσο, μειονεκτήματα της προσέγγισης αυτής είναι πρώτον ότι επηρεάζεται από τις συνθήκες φωτισμού και δεύτερον ότι το χρώμα του ανθρώπινου δέρματος είναι ένα φάσμα που μπορεί να εκτείνεται από το λευκό έως το μαύρο. Τέλος, στην ανίχνευση με βάση κινούμενες εικόνες/βίντεο αξιοποιείται η κίνηση για τον εντοπισμό των ανθρώπινων μορφών. Σε κάποιες μεθόδους, μάλιστα, αξιοποιείται και το ανοιγοκλείσιμο των ματιών για την ανίχνευση προσώπων [22].

Η δεύτερη ταξινόμηση βασίζεται στο έργο των Yang, Kriegman & Ahuja και είναι ευρέως αποδεκτή. Οι μέθοδοι ανίχνευσης διακρίνονται σε:

1. Μέθοδοι βασισμένες στη γνώση (knowledge-based methods): αξιοποιείται από τους ερευνητές η γνώση για τα χαρακτηριστικά του ανθρώπινου προσώπου (π.χ. θέσεις και αποστάσεις ματιών, στόματος).
2. Προσεγγίσεις των αναλλοίωτων χαρακτηριστικών (feature invariant approaches): οι μέθοδοι αυτοί εντοπίζουν τα δομικά χαρακτηριστικά του προσώπου (φρύδια, μάτια, στόμα κτλ.) ανεξάρτητα από τη θέση λήψης ή το φωτισμό και κατασκευάζουν στατιστικά μοντέλα μορφών.
3. Μέθοδοι συνταιριάσματος υποδειγμάτων (template matching methods): αξιοποιούνται προκαθορισμένα τυπικά υποδείγματα για να ανιχνεύσουν τα χαρακτηριστικά ενός προσώπου. Το μειονέκτημα της προσέγγισης αυτής αναφέρεται στο ότι τα υποδείγματα είναι προκαθορισμένα και έτσι δε λαμβάνονται υπόψη αλλαγές στην κλίμακα, στον προσανατολισμό κτλ.
4. Μέθοδοι βασισμένες στην εμφάνιση (appearance-based methods): αξιοποιούνται υποδείγματα προσώπου, τα οποία όμως δεν είναι προκαθορισμένα όπως στην προηγούμενη προσέγγιση, αλλά προέρχονται από ένα σύνολο εικόνων εκπαίδευσης, αντιπροσωπευτικών των ποικίλων τρόπων εμφάνισης ενός προσώπου (π.χ. παραλλαγές στην κλίμακα, στη θέση, στην έκφραση κτλ.) [21].

1.1.3. Μέθοδοι Ανίχνευσης Βασισμένες Στην Εμφάνιση

Έχει φανεί ερευνητικά ότι οι πιο αποτελεσματικές μέθοδοι ανίχνευσης προσώπου είναι όσες βασίζονται στην εμφάνιση (appearance-based). Οι εικόνες εκπαίδευσης που αποτελούν τα υποδείγματα προσώπου προέρχονται από φωτογραφίες που έχουν ληφθεί από όλες τις δυνατές θέσεις και κλίμακες.

Στη συνέχεια οι φωτογραφίες σαρώνονται από ένα υπό-παράθυρο ανίχνευσης. Η ανίχνευση κρίνεται από την ταξινόμηση του κάθε δείγματος φωτογραφίας από το υπό-παράθυρο σαν πρόσωπο ή μη-πρόσωπο. Η απόφαση αυτή βασίζεται σε ένα σύνολο δειγμάτων προσώπων και μη-προσώπων που εκπαιδεύουν έναν ταξινομητή χρησιμοποιώντας στατιστικές μεθόδους εκπαίδευσης[2].

Οι κυριότερες μέθοδοι ανίχνευσης προσώπων είναι οι εξής:

- Μέθοδος Eigenfaced-based: Οι Sirovich & Kirby ανέπτυξαν μία μέθοδο ανίχνευσης προσώπου χρησιμοποιώντας την Ανάλυση Κυρίων Συνιστωσών (Principal Component Analysis – PCA). Ο στόχος τους είναι η αναπαράσταση ενός προσώπου ως ένα σύστημα συντεταγμένων. Αργότερα, άλλοι ερευνητές αξιοποίησαν τη μεθόδό τους για να αναπτύξουν μια μέθοδο αναγνώρισης προσώπου.
- Νευρωνικά δίκτυα (Neural networks): Οι Rowley et al. ανέπτυξαν μια εξελιγμένη νευρωνική προσέγγιση στην ανίχνευση προσώπων. Το σύστημά τους ενσωματώνει γνώση προσώπου σε ένα νευρωνικό δίκτυο το οποίο είναι σχεδιασμένο να κοιτάει σε παράθυρα 20×20 pixels. Ένα πρόβλημα που προκύπτει από τις τεχνικές σκαναρίσματος παραθύρου είναι οι επικαλυπτόμενοι εντοπισμοί, ωστόσο οι ερευνητές έχουν αναπτύξει τεχνικές για να αντιμετωπίσουν αυτό το πρόβλημα. Για να βελτιωθεί επιπλέον την απόδοση, εκπαιδεύονται πολλαπλά νευρωνικά δίκτυα και η έξοδος συνδυάζεται με μια στρατηγική διαίτησίας.
- Μέθοδος Viola & Jones: Χρήση των κυματιδίων τύπου Haar σε ταξινομητές ενός-μόνο χαρακτηριστικού που εκπαιδεύονται με τον αλγόριθμο AdaBoost. Η μέθοδος αυτή έχει αποδειχθεί ιδιαίτερα αποτελεσματική στην ανίχνευση προσώπου. Θα αναλυθεί διεξοδικότερα στο επόμενο κεφάλαιο της παρούσας μελέτης, καθώς είναι και η μέθοδος που έχει αξιοποιηθεί στην έρευνα μας.
- Μέθοδος Lienhart et al. : οι Lienhart et al. βελτίωσαν τη μέθοδο των Viola & Jones με τη χρήση ενός μεγάλου συνόλου περιστραμμένων κατά 45° χαρακτηριστικών Haar

αυξάνοντας έτσι την αποτελεσματικότητα του συστήματος, προσθέτοντας νέες πληροφορίες για την ανίχνευση προσώπων.

- Μέθοδος Li et al. : Οι Li et al. προχώρησαν στη χρήση ενός μεγάλου συνόλου περιστραμμένων χαρακτηριστικών Haar και ενός πυραμιδοειδούς ανιχνευτή για τη διαχείριση των στρωφών εκτός του επιπέδου της εικόνας.
- Μέθοδος HOG : Η μέθοδος HOG (Histogram of Oriented Gradients) αναπτύχθηκε αρχικά από τον Robert McConnell, αλλά έγινε ευρύτερα διαδεδομένη ως μέθοδος HOG κάποια χρόνια αργότερα από τους Dalal & Triggs. Ο HOG είναι ένας περιγραφέας χαρακτηριστικών. Η βασική ιδέα του HOG είναι ότι ένα αντικείμενο οποιουδήποτε σχήματος στην εικόνα μπορεί να περιγραφεί από την κατανομή της έντασης των κλίσεων ή από την κατεύθυνση των γωνιών. Αυτό μπορεί να γίνει διαιρώντας την εικόνα σε μικρότερα τμήματα που χαρακτηρίζονται ως κελιά (cells). Σε κάθε ένα από αυτά εξάγεται ένα ιστόγραμμα κλίσης σύμφωνα με την κατεύθυνση των γωνιών των pixel. Πλεονέκτημα της μεθόδου HOG και ότι λειτουργεί με βάση κελιά είναι ότι διατηρούνται αμετάβλητοι τόσο οι γεωμετρικοί και φωτομετρικοί σχηματισμοί όσο και η κατεύθυνση των αντικειμένων. [3, 22].

Επιπλέον, αναφέρονται επιγραμματικά και άλλες, χρονολογικά παλαιότερες, μέθοδοι:

- Χρήση μη γραμμικών μηχανών ανυσματικής στήριξης (Support Vector Machines – SVM) που αναπτύχθηκε από τους Osuna et al. το 1997.
- Χρήση κυματιδίων Haar με SVM ταξινομητή (Haar Wavelets): που αναπτύχθηκε από τους Papageorgiou & Poggio το 1998.
- Χρήση Γκαουσιανών κατανομών και πολυστρωματικών αντιλήπτρων (multilayer perceptron), που αναπτύχθηκε από τους Sung & Poggio το 1998.
- Χρήση στατιστικής για την τοπική εμφάνιση και εκμάθηση με AdaBoost που αναπτύχθηκε από τους Schneiderman & Kanade το 2000 [3, 22].

Συμπερασματικά, οι μέθοδοι βασισμένες στην εμφάνιση χρησιμοποιούν κάποια κοινά χαρακτηριστικά. Αρχικά, χρησιμοποιούν θετικά και αρνητικά δείγματα προσώπων ως εικόνες εκπαίδευσης και παράλληλα χρησιμοποιούν έναν ταξινομητή, τον οποίο εκπαιδεύουν στην ανίχνευση προσώπων. Επίσης, διαθέτουν συγκεκριμένο τρόπο αναπαράστασης των προσώπων και αντίστοιχα συγκεκριμένη στρατηγική αναζήτησης στο χώρο της εικόνας [2].

1.2 Μέθοδοι Αναγνώρισης

1.2.1 Εισαγωγή

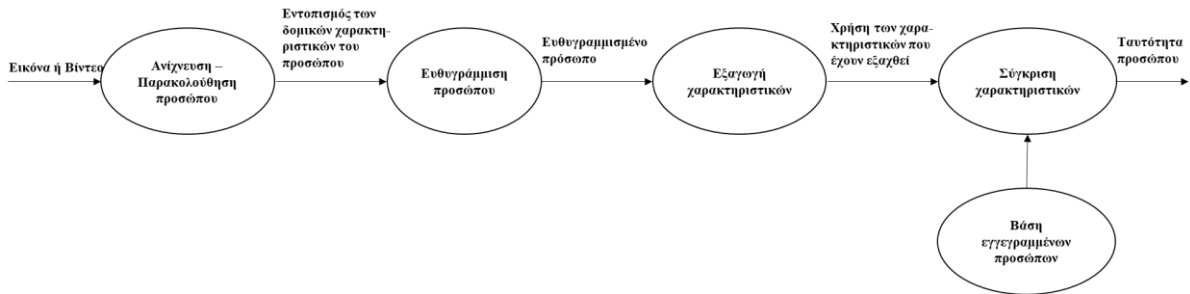
Η αναγνώριση προσώπου (face recognition) είναι ένα πρόβλημα αναγνώρισης οπτικών προτύπων (visual pattern recognition problem). Ένα πρόσωπο που παρουσιάζεται σε διαφορετικές συνθήκες έκφρασης, φωτισμού κτλ. και αναπαρίσταται ως ένα τρισδιάστατο αντικείμενο, πρέπει να αναγνωριστεί με βάση εικόνες προσώπων που έχουν αποθηκευτεί σε κάποια βάση δεδομένων. Βέβαια, οι περισσότερες εφαρμογές αναγνώρισης βασίζονται και σε δισδιάστατη αναπαράσταση των προσώπων, ενώ η τρισδιάστατη αναπαράσταση συνηθίζεται να χρησιμοποιείται σε περιπτώσεις όπου το επίπεδο ασφαλείας είναι υψηλότερο [2].

Το πρώτο αυτόματο σύστημα αναγνώρισης προσώπου αναπτύχθηκε από τον Kanade το 1973 στη διδακτορική του διατριβή «Picture processing system by computer complex and recognition of human faces». Στη συνέχεια, μετά από ένα χρονικό διάστημα αδράνειας, σημαντική πρόοδο πέτυχαν οι Sirovich & Kirby με τη μέθοδο Ανάλυσης Κυρίων Συνιστωσών (Principal Component Analysis – PCA). Έκτοτε εισήχθησαν σημαντικές καινοτόμες μέθοδοι όπως η μέθοδος των Eigenfaces, η μέθοδος Fisherface, η Γραμμική Διακρίνουσα Ανάλυση (Linear Discriminant Analysis – LDA), η μέθοδος του αλγορίθμου AdaBoost και άλλες.

Πλέον στις ημέρες μας, οι εφαρμογές αναγνώρισης προσώπου είναι πολυάριθμες. Η αναγνώριση προσώπου αξιοποιείται σε εφαρμογές ελέγχου πρόσβασης σε κτήρια ή σε περιοχές ασφαλείας. Το πρόσωπο καταγράφεται από μία κάμερα και τελικά η είσοδος επιτρέπεται μόνο αν υπάρξει ταύτιση του προσώπου αυτού με τα αποθηκευμένα πρόσωπα στη βάση δεδομένων. Επίσης, η αναγνώριση προσώπου έχει αρχίσει να υποκαθιστά τους κλασικούς τρόπους επαλήθευσης ταυτότητας χρηστών σε διάφορα ηλεκτρονικά μέσα (π.χ. ηλεκτρονικές τραπεζικές συναλλαγές), αυξάνοντας έτσι τα επίπεδα ασφαλείας, δεδομένου ότι για παράδειγμα οι κωδικοί χρήστη μπορεί να υποκλαπούν. Ακόμη, η αναγνώριση προσώπου μέσα από συστήματα παρακολούθησης μπορεί να εξυπηρετήσει την επιβολή του νόμου. Τέλος, ορισμένες ενδιαφέρουσες εφαρμογές θεωρούνται η ανακατασκευή του προσώπου από υπολειπόμενα στοιχεία και ο υπολογισμός των αλλαγών που επήλθαν με την πάροδο του χρόνου [26, 2].

1.2.2. Διαδικασία – Πορεία Αναγνώρισης Προσώπων

Ένα σύστημα αναγνώρισης προσώπου αποτελείται από τέσσερα βασικά υποσυστήματα, όπως φαίνεται στην εικόνα 1.1.



Εικόνα 1.1. Διάγραμμα Διαδικασίας Αναγνώρισης προσώπου [2]

Το πρώτο υποσύστημα είναι αυτό της *ανίχνευσης προσώπου* (face detection), που, όπως έχει αναφερθεί, στοχεύει στον έλεγχο ύπαρξης προσώπων σε μία εικόνα. Στην περίπτωση του βίντεο τα πρόσωπα που ανιχνεύονται πρέπει να ανιχνεύονται διαρκώς σε κάθε πλαίσιο, με ένα σύστημα παρακολούθησης προσώπου (face tracking). Στη συνέχεια μέσω του εντοπισμού – παρακολούθησης προσώπου (face localization), εντοπίζονται με ακρίβεια τα δομικά χαρακτηριστικά (π.χ. μάτια, μύτη, στόμα, περίγραμμα προσώπου).

Η *ευθυγράμμιση του προσώπου* (face normalization) είναι το επόμενο στάδιο. Για την ευθυγράμμιση χρησιμοποιούνται τα δομικά χαρακτηριστικά που έχουν ανιχνευθεί. Έτσι, τα πρόσωπα κανονικοποιούνται τόσο ως προς τις γεωμετρικές τους ιδιότητες όπως το μέγεθος ή ο προσανατολισμός όσο και ως προς τις φωτομετρικές τους ιδιότητες όπως ο φωτισμός ή η κλίμακα του γκρι.

Κατά τη φάση *εξαγωγής χαρακτηριστικών* (feature extraction), η οποία εφαρμόζεται πάνω στο ευθυγραμμισμένο πρόσωπο, εξάγονται τα χαρακτηριστικά που θα επιτρέψουν τη σύγκριση των προσώπων. Στο υποσύστημα της *σύγκρισης προσώπων* (face matching), τα χαρακτηριστικά που έχουν εξαχθεί συγκρίνονται με τα χαρακτηριστικά των προσώπων που υπάρχουν στη βάση εγγεγραμμένων προσώπων. Εάν έχει βρεθεί ταύτιση με αρκετή αξιοπιστία, τότε εξακριβώνεται η ταυτότητα του προς αναζήτηση προσώπου. Διαφορετικά, εάν δεν επιτευχθεί ταύτιση, η ταυτότητα του προς αναζήτηση προσώπου παραμένει άγνωστη [2].

1.2.3. Ταξινόμηση Μεθόδων Αναγνώρισης Προσώπου

Αντίστοιχα με τις μεθόδους ανίχνευσης προσώπου, έτσι και για τις μεθόδους αναγνώρισης προσώπου έχουν προταθεί διαφορετικές ταξινομήσεις. Η κυριότερη ταξινόμηση είναι η εξής:

1. Ολιστικές μέθοδοι αναγνώρισης (holistic matching methods): Στην ολιστική προσέγγιση ολόκληρη η περιοχή του προσώπου λαμβάνεται υπόψη και επεξεργάζεται από το σύστημα αναγνώρισης. Ένα από τα πιο χαρακτηριστικά παραδείγματα ολιστικής μεθόδου είναι η μέθοδος Eigenfaces.
2. Μέθοδοι βασισμένες στα χαρακτηριστικά (feature – based methods): Στις μεθόδους βασισμένες στα χαρακτηριστικά εντοπίζονται τα δομικά χαρακτηριστικά του προσώπου (π.χ. μάτια, μύτη, στόμα) και εισάγονται σε έναν ταξινομητή. Οι μέθοδοι της προσέγγισης αυτής διακρίνονται σε τέσσερις υπό-κατηγορίες:
 1. Μέθοδοι βασισμένες στη γεωμετρία (geometry-based methods): Στις μεθόδους αυτές χρησιμοποιούνται χαρακτηριστικά που εξάγονται με βάση το μέγεθος και τη θέση των δομικών χαρακτηριστικών του προσώπου.
 2. Μέθοδοι βασισμένες στα υποδείγματα (template-based methods): Στις μεθόδους αυτές χρησιμοποιούνται χαρακτηριστικά που εξάγονται με βάση προσχεδιασμένα υποδείγματα. Ένα παράδειγμα τέτοιας μεθόδου είναι το Παραμορφώσιμο Πρότυπο του Yuille.
 3. Μέθοδοι βασισμένες στην εμφάνιση (appearance-based methods): Στις μεθόδους αυτές κάθε χαρακτηριστικό που εξάγεται από την εικόνα μπορεί να αξιοποιηθεί ως χαρακτηριστικό που θα οδηγήσει στην αναγνώριση. Δηλαδή, δεν περιορίζονται μόνο σε χαρακτηριστικά όπως τα μάτια και η μύτη. Επίσης, οι μέθοδοι αυτοί είναι πιο αποδοτικές, επειδή διατηρούν τις χρήσιμες πληροφορίες και απορρίπτουν τις πλεονάζουσες. Παραδείγματα είναι οι μέθοδοι της Ανάλυσης Κυρίων Συνιστωσών (PCA) και της Ανάλυσης Ανεξάρτητων Συνιστωσών (ICA).
 4. Μέθοδοι βασισμένες στο χρώμα (color-based methods): Στις μεθόδους αυτές η αναγνώριση του προσώπου βασίζεται στην ανίχνευση του χρώματος του δέρματος. Έτσι διαχωρίζεται η περιοχή του προσώπου από τα υπόλοιπα στοιχεία της εικόνας. Στη συνέχεια, κάθε περιοχή του προσώπου που διαφέρει χρωματικά από το χρώμα του δέρματος είναι υποψηφία για την περιοχή των ματιών και του στόματος, τα οποία χρησιμοποιούνται τελικά για την αναγνώριση.

3. Υβριδικές μέθοδοι αναγνώρισης (hybrid methods): Οι υβριδικές μέθοδοι αξιοποιούν στοιχεία τόσο από τις ολιστικές μεθόδους όσο και από τις μεθόδους βασισμένες σε χαρακτηριστικά. Συνήθως, μάλιστα, χρησιμοποιούν τρισδιάστατες εικόνες. Το πλεονέκτημα των τρισδιάστατων εικόνων είναι ότι παρέχουν περισσότερες πληροφορίες για το προς αναγνώριση πρόσωπο, όπως για παράδειγμα οι καμπύλες της εσοχής των ματιών, το σχήμα του πηγουνιού ή του μετώπου κτλ. Η πορεία αναγνώρισης με εικόνες τριών διαστάσεων συνήθως περιλαμβάνουν τα εξής βήματα: ανίχνευση – θέση (καθορισμός της θέσης και του προσανατολισμού του κεφαλιού) – μέτρηση (μετρήσεις των γωνιών του προσώπου) – αναπαράσταση (κατασκευή μιας αριθμητικής αναπαράστασης του προσώπου) – σύγκριση (σύγκριση των δεδομένων με τη βάση εγγεγραμμένων προσώπων) [13, 17].

1.2.4. Επισκόπηση Των Βασικότερων Μεθόδων Αναγνώρισης Προσώπου

Η μέθοδος της *Ανάλυσης Κυρίων Συνιστωσών* (Principal Component Analysis – PCA) είναι κατά βάση μία στατιστική μέθοδος, η οποία στοχεύει στη μείωση των μεταβλητών. Χρησιμοποιείται σε περιπτώσεις όπου συγκεντρώνεται ένα μεγάλο πλήθος μεταβλητών και θεωρείται ότι κάποιες από αυτές είναι πλεονάζουσες. Κάποιες μεταβλητές θεωρούνται πλεονάζουσες καθώς μετρούν – περιγράφουν την ίδια έννοια – χαρακτηριστικό. Έτσι, μπορούν να μειωθούν σε ένα μικρότερο πλήθος κυρίων συνιστωσών, οι οποίες θα περιγράφουν με πιο συνοπτικό τρόπο το ίδιο χαρακτηριστικό. Αρχικά, οι Sirovich & Kirby αξιοποίησαν την τεχνική αυτή για την ανίχνευση του προσώπου, ενώ οι Turk & Pentland την επέκτειναν στην αναγνώριση προσώπου.

Οι κύριες συνιστώσες, στην περίπτωση της αναγνώρισης προσώπου, είναι τα μικρότερα σετ χαρακτηριστικών του προσώπου που εξάγονται από τις εικόνες εκπαίδευσης του συστήματος και έχει επικρατήσει να ονομάζονται «Eigenfaces». Ο όρος Eigenfaces ουσιαστικά αναφέρεται στο μετασχηματισμό της δομής του προσώπου σε ένα ορθοκανονικό σύνολο αξόνων στις διευθύνσεις όπου τα δεδομένα παρουσιάζουν τη μέγιστη συνδιακύμανση. Οι εικόνες προσώπων του συνόλου εκπαίδευσης προβάλλονται στον υποχώρο προσώπου (face space) με σκοπό να υπολογιστεί ένα σύνολο βαρών που περιγράφει τη συμμετοχή κάθε διανύσματος στον υποχώρο προσώπου. Για να γίνει ταυτοποίηση μιας νέας εικόνας πρέπει να προβληθεί το διάνυσμα που την αναπαριστά στον υποχώρο προσώπου και να βρεθεί το αντίστοιχο σύνολο βαρών. Συγκρίνοντας τα βάρη της νέας εικόνας με το

σύνολο βαρών των εικόνων εκπαίδευσης μπορεί να γίνει η ταυτοποίηση της και συνεπώς η αναγνώριση του προσώπου [2, 3].

Τα πλεονεκτήματα της μεθόδου είναι ότι είναι σχετικά απλή συγκρινόμενη με άλλες μεθόδους, ότι τα δεδομένα που εισάγονται στο σύστημα είναι αξιοποιήσιμα χωρίς ιδιαίτερη προ-επεξεργασία και ότι δεν απαιτείται γνώση της γεωμετρίας του προσώπου. Επίσης, ενδιαφέρον στοιχείο είναι ότι ο συνδυασμός των Eigenfaces μπορεί να αξιοποιηθεί στην ανακατασκευή κάθε αρχικής εικόνας από το σετ εκπαίδευσης. Τα μειονεκτήματα της μεθόδου είναι ότι επηρεάζεται πολύ από τις αλλαγές της κλίμακας – μεγέθους των προσώπων, ότι η αξιοπιστία της αναγνώρισης μειώνεται όταν οι συνθήκες φωτισμού δεν είναι ιδανικές και ότι απαιτεί ένα μεγάλο χρονικό διάστημα εκπαίδευσης του συστήματος, επειδή είναι μέθοδος βασισμένη στην εμφάνιση [14].

Η μέθοδος της *Γραμμικής Διακρινουσας Ανάλυσης* (Linear Discriminant Analysis – LDA) ή αλλιώς μέθοδος *Fisherface*, είναι μία μέθοδος που στοχεύει στο να ανακαλύψει το γραμμικό συνδυασμό χαρακτηριστικών τα οποία διαχωρίζουν καλύτερα δύο ή περισσότερες κλάσεις προσώπων. Η λειτουργία του αλγορίθμου βασίζεται σε μείωση των διαστάσεων του προβλήματος, διαφέρει όμως από την PCA στο ότι επιχειρεί να μοντελοποιήσει τη διαφορά μεταξύ των κατηγοριών των δεδομένων.

Τα πλεονεκτήματα της μεθόδου είναι ότι σε αντίθεση με τη PCA η αξιοπιστία της δεν επηρεάζεται από τις συνθήκες φωτισμού, καθώς και ότι οι αλγόριθμοι της μεθόδου αυτοί είναι πιο αποτελεσματικοί στην αναπαράσταση των αντικειμένων – προσώπων με έμφαση στα χαρακτηριστικά που τα διαφοροποιούν. Ωστόσο, μειονέκτημα της μεθόδου είναι ότι απαιτεί μεγάλο αριθμό δειγματικών εικόνων προκειμένου να υπολογιστούν οι κλάσεις και να μπορέσει να λειτουργήσει σωστά ο αλγόριθμος αναγνώρισης [2, 14].

Η μέθοδος της *Ανάλυσης Ανεξάρτητων Συνιστωσών* (Independent Component Analysis – ICA) είναι επίσης μια γενίκευση της μεθόδου PCA και στοχεύει στον υπολογισμό μιας στατιστικά ανεξάρτητης διανυσματικής βάσης. Η μέθοδος αυτή ορίζει ένα γενικευμένο μοντέλο των παρατηρούμενων δεδομένων, τα οποία συνήθως δίνονται με τη μορφή μίας βάσης δεδομένων από δείγματα. Στο μοντέλο αυτό τα δεδομένα θεωρούνται γραμμικά συνδυαστικά παράγωγα κάποιων αγνώστων μεταβλητών, καθώς και ο τρόπος συνδυασμού των μεταβλητών αυτών είναι άγνωστος. Οι μεταβλητές αυτές ονομάζονται ανεξάρτητες συνιστώσες των παρατηρούμενων δεδομένων. Για την αναγνώριση προσώπου έχουν προταθεί δύο αρχιτεκτονικές βασισμένες στη μέθοδο ICA. Η πρώτη αρχιτεκτονική παράγει

στατιστικά ανεξάρτητες εικόνες βάσης, ενώ η δεύτερη αρχιτεκτονική παράγει στατιστικά ανεξάρτητους συντελεστές. Παρότι η μέθοδος ανάλυσης ανεξάρτητων συνιστωσών εμφανίζει αρκετά πλεονεκτήματα, βασικό μειονέκτημα είναι η πολυπλοκότητά της σε σχέση με την PCA [12, 28].

Μια άλλη μέθοδος είναι οι *Μηχανές Διανυσμάτων Υποστήριξης* (Support Vector Machines – SVM). Οι μηχανές διανυσμάτων υποστήριξης αποτελούν ένα σύνολο σχεσιακών μεθόδων εκπαίδευσης με επίβλεψη για σκοπούς ταξινόμησης. Αντιμετωπίζουν τα δεδομένα εισόδου σαν δύο σύνολα διανυσμάτων σε ένα n -διάστατο χώρο και κατασκευάζουν μία διαχωριστική υπερεπιφάνεια σ' αυτό το χώρο, η οποία μεγιστοποιεί το περιθώριο μεταξύ των δύο συνόλων των δεδομένων. Οι μηχανές διανυσμάτων υποστήριξης είναι, λοιπόν, μέθοδος δυαδικής ταξινόμησης. Το πρόβλημα της αναγνώρισης προσώπου είναι, ωστόσο, ένα πρόβλημα n -κλάσεων. Απαιτείται, επομένως, μια διαφορετική προσέγγιση. Έχουν αναπτυχθεί δύο στρατηγικές. Η πρώτη στρατηγική «ενός προς όλα», απαιτεί την εκπαίδευση μηχανών οι οποίες θα ξεχωρίζουν την κάθε μία κλάση από όλες τις υπόλοιπες. Στη δεύτερη στρατηγική οι κλάσεις χωρίζονται ανά ζευγάρια και εκπαιδεύονται μηχανές που διαχωρίζουν τις δύο κλάσεις μεταξύ τους. Σε επόμενο στάδιο, αυτές οργανώνονται σε δέντρα απόφασης και η αναγνώριση επιτυγχάνεται μέσα από τον σταδιακό αποκλεισμό των κλάσεων [11, 28].

Μια εξέλιξη στις μεθόδους αναγνώρισης είναι τα *Πιθανοθεωρητικά Νευρωνικά Δίκτυα Βασισμένα σε Αποφάσεις* (Probabilistic Neural Network – PNN). Τα δίκτυα είναι έτσι σχεδιασμένα ώστε να μοντελοποιούν τη συνάρτηση φυσικής λογαριθμικής ομοιότητας. Ο βέλτιστος τρόπος για τη χρήση των δικτύων αυτών στην αναγνώριση προσώπων είναι η τεχνική διάκρισης μίας κλάσης από όλες τις υπόλοιπες. Έτσι δημιουργείται ένα δίκτυο για κάθε πρόσωπο, ενώ όλα συνδέονται με ένα ευρύτερο δίκτυο, το οποίο καθορίζει τη κλάση που επικρατεί με βάση τις εξόδους των υποδικτύων. Το πλεονέκτημα της μεθόδου είναι ο σύντομος χρόνος εκπαίδευσης των δικτύων. Μάλιστα, όταν προστίθεται ένα νέο άτομο αρκεί να εκπαιδευτεί μόνο ένα νέο δίκτυο που θα αφορά το συγκεκριμένο πρόσωπο και να προσαρμοστεί ελάχιστα η εκπαίδευση των υπολοίπων [5].

Η μέθοδος των *Τοπικών Δυαδικών Προτύπων*: Η μέθοδος LBP (Local Binary Pattern Histograms – LBP) αξιοποιείται για την αναγνώριση προσώπων μέσω της κατασκευής ιστογραμμάτων χαρακτηριστικών για κάθε πρόσωπο σε μια βάση δεδομένων. Τα ιστογράμματα προκύπτουν από τον χωρισμό της εικόνας σε τμήματα και την εξαγωγή ενός δυαδικού αριθμού για κάθε ένα τμήμα. Η μέθοδος αυτή θα αναλυθεί διεξοδικότερα στο

επόμενο κεφάλαιο της παρούσας μελέτης, καθώς είναι και η μέθοδος που έχει αξιοποιηθεί στην έρευνα μας [24].

Επίσης, το τελευταίο χρονικό διάστημα έχουν αναπτυχθεί *τρισδιάστατες μέθοδοι αναγνώρισης προσώπου*, οι οποίες, όπως αναφέρθηκε στην προηγούμενη ενότητα, εκμεταλλεύονται περισσότερες πληροφορίες για το προς αναγνώριση πρόσωπο, όπως για παράδειγμα οι καμπύλες της εσοχής των ματιών, το σχήμα του πηγουνιού ή του μετώπου κτλ. χωρίς να επηρεάζονται, μάλιστα, από τις διακυμάνσεις του φωτισμού, τις διάφορες θέσεις του κεφαλιού, τις εκφράσεις του προσώπου και το φόντο της εικόνας. Ορισμένες τεχνικές που έχουν αναπτυχθεί είναι οι εξής:

- **Συστήματα σκαναρίσματος:** Εταιρείες έχουν παράγει laser σαρωτές προσώπου που παράγουν αξιόπιστες τρισδιάστατες αναπαραστάσεις προσώπων. Ωστόσο, το κόστος σάρωσης είναι πολύ μεγάλο.
- **Δομημένα συστήματα φωτός:** Αυτές οι τεχνικές αξιοποιούν τις τεχνικές της στερεοσκοπικής όρασης. Πλεονέκτημα θεωρείται ότι απαιτούν απλά τη χρήση κάμερας και ενός συστήματος προβολής.
- **Συστήματα στερεοσκοπικής όρασης:** Αυτές οι τεχνικές στοχεύουν στην εξαγωγή τρισδιάστατης πληροφορίας από δυο ή περισσότερες δυσδιάστατες εικόνες του ίδιου αντικειμένου ή προσώπου που λαμβάνονται από διαφορετικές γωνίες. Το μειονέκτημα των τεχνικών αυτών σχετίζεται με τη σχετικά λεία υφή του ανθρώπινου προσώπου, καθώς τα συστήματα αυτά δεν έχουν καταφέρει μέχρι στιγμής να παράγουν ένα αξιόπιστο σχήμα επιφάνειας λείων αντικειμένων [25].

Επιπλέον, αναφέρονται επιγραμματικά και άλλες μέθοδοι:

- **Μέθοδοι Kernel:** είναι μια γενίκευση των γραμμικών μεθόδων, όπου η αναπαράσταση του προσώπου στον υποχώρο προσώπου δεν χρειάζεται να είναι γραμμική.
- Τα μοντέλα ενεργού σχήματος (Active Shape Models): αναπαριστούν τη φυσική και κατ' επέκταση υψηλότερου επιπέδου εμφάνιση των χαρακτηριστικών.
- Τα μοντέλα Hidden Markov: είναι ένα στατιστικό εργαλείο που χρησιμοποιήθηκε για την αναγνώριση προσώπου σε συνδυασμό με τη μέθοδο των νευρωνικών δικτύων.
- Τα κυματίδια Gabor: είναι φίλτρα που αξιοποιούνται σε συνδυασμό με τα πεδία Fourier για την ανάλυση των πληροφοριών. Έχουν αξιοποιηθεί και αυτά στη μέθοδο των νευρωνικών δικτύων [28].

Τέλος, αξίζει να αναφερθεί ότι τα τελευταία χρόνια γίνεται προσπάθεια αναβάθμισης και εξέλιξης των μεθόδων ώστε να αυξηθεί η αξιοπιστία τους στην αναγνώριση προσώπων από βίντεο. Η βασική πρόκληση είναι ότι τα βίντεο που προέρχονται από κάμερες παρακολούθησης είναι χαμηλής ποιότητας και οι εικόνες των προσώπων αρκετά μικρές [28].

Κεφάλαιο 2 : Ανίχνευση Προσώπου Κατά Viola & Jones Και Αναγνώριση Κατά LBPH Με Χρήση OpenCV Και Python

2.1. Μέθοδος Κατά Viola & Jones

2.1.1. Εισαγωγή

Οι Paul Viola & Michael Jones ανέπτυξαν μία νέα μέθοδο ανίχνευσης αντικειμένων και προσώπων, οι οποίες παρουσιάστηκαν το 2001 στα άρθρα τους «Robust real-time face detection» και «Rapid Object Detection using a boosted cascade of simple features». Ο βασικός τους στόχος ήταν να αναπτύξουν μια μέθοδο, η οποία θα είναι τόσο αποτελεσματική όσο και ταχεία [20, 19].

Η μέθοδος αυτή χαρακτηρίζεται από τρεις βασικές έννοιες, που τη διαφοροποιούν από προηγούμενες μεθόδους. Αυτές οι έννοιες αποτελούν τις κύριες αρχές της λειτουργίας της και συμβάλλουν στην υψηλή αποτελεσματικότητά της. Οι νέες αυτές έννοιες είναι οι εξής:

1. Η ολοκληρωτική εικόνα (integral image): είναι ένας νέος τρόπος αναπαράστασης των προσώπων και επιτρέπει τον γρήγορο υπολογισμό των χαρακτηριστικών (features) από τον ανιχνευτή.
2. Ένας απλός αλλά αποτελεσματικός ταξινομητής που χρησιμοποιεί έναν μικρό αριθμό από κρίσιμα χαρακτηριστικά και ενισχύεται από τον αλγόριθμο μάθησης AdaBoost.
3. Ο συνδυασμός όλο και πιο σύνθετων ταξινομητών σε μια δομή καταρράκτη (cascade), η οποία αυξάνει την ταχύτητα του ανιχνευτή εστιάζοντας την προσοχή σε περιοχές της εικόνας, στις οποίες η πιθανότητα να υπάρχει πρόσωπο είναι μεγάλη [20, 19].

Η μέθοδος αναπτύσσεται σε τέσσερα στάδια:

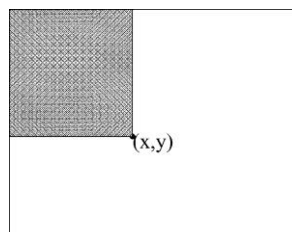
1. Δημιουργία της ολοκληρωτικής εικόνας
2. Υπολογισμός χαρακτηριστικών
3. Επιλογή χαρακτηριστικών – Εκπαίδευση του ταξινομητή με τον AdaBoost
4. Ταξινόμηση με έναν καταρράκτη ταξινομητών.

2.1.2. Δημιουργία Της Ολοκληρωτικής Εικόνας

Η ολοκληρωτική εικόνα (integral image) είναι ένας νέος τρόπος αναπαράστασης των προσώπων που εξυπηρετεί τον γρήγορο υπολογισμό των χαρακτηριστικών. Η ολοκληρωτική εικόνα (Εικόνα 2.1.) έχει τιμή στη θέση (x, y) που ορίζεται ως το άθροισμα των εντάσεων των pixel του ορθογωνίου από την πάνω αριστερή κορυφή $(0, 0)$ έως και την κάτω δεξιά κορυφή (x, y) . Ο τύπος που την περιγράφει είναι ο εξής :

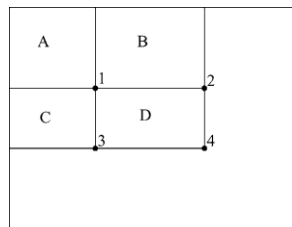
$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2.1)$$

όπου, $ii(x, y)$ είναι η ολοκληρωτική εικόνα, $i(x, y)$ είναι η αρχική εικόνα.



Εικόνα 2.1. Η ολοκληρωτική εικόνα [20]

Το άθροισμα των σημείων ενός ορθογωνίου της ολοκληρωτικής εικόνας μπορεί να υπολογιστεί από τις τέσσερις κορυφές του. Για παράδειγμα, στην Εικόνα 2.2. για ένα ορθογώνιο D που ορίζεται από τα σημεία 1, 2, 3, 4 το άθροισμα των pixel μπορεί να υπολογιστεί με τέσσερις σχέσεις πινάκων. Η τιμή της ολοκληρωτικής εικόνας στη θέση 1 είναι το άθροισμα των pixel στο τετράγωνο A . Η τιμή στη θέση 2 είναι $A+B$, στη θέση 3 είναι $A+C$, και στη θέση 4 είναι $A + B + C + D$. Το άθροισμα μέσα στο D μπορεί να υπολογιστεί ως $4+1-(2+3)$ [20, 19].

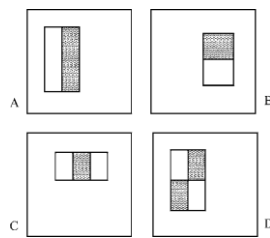


Εικόνα 2.2. Αναπαράσταση της ολοκληρωτικής εικόνας[20]

2.1.3. Υπολογισμός Χαρακτηριστικών

Το σύστημα ανίχνευσης ταξινομεί τις εικόνες βάσει απλών βαθμωτών χαρακτηριστικών (features). Τα χαρακτηριστικά είναι όμοια με τις συναρτήσεις βάσης τύπου Haar (Haar basis) και εφαρμόζονται σε ασπρόμαυρες εικόνες.

Τα χαρακτηριστικά των Viola & Jones ονομάζονται «χαρακτηριστικά Haar» και είναι ορθογώνια σχήματα με λευκές και μαύρες περιοχές. Υπάρχουν 3 είδη χαρακτηριστικών: δύο ορθογωνίων – τριών ορθογωνίων και τεσσάρων ορθογωνίων. Η τιμή των χαρακτηριστικών με δύο ορθογώνια είναι η διαφορά μεταξύ του αθροίσματος των pixel της λευκής και της μαύρης περιοχής. Η τιμή των χαρακτηριστικών με τρία ορθογώνια είναι η διαφορά μεταξύ του αθροίσματος των δύο λευκών περιοχών και της μαύρης περιοχής. Η τιμή των χαρακτηριστικών με τέσσερα ορθογώνια είναι η διαφορά μεταξύ διαγώνιων ζευγαριών λευκών και μαύρων περιοχών.



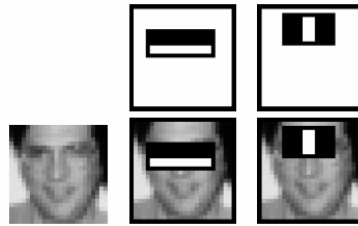
Εικόνα 2.3. Χαρακτηριστικά Haar των Viola & Jones [20]

Τα ορθογώνια χαρακτηριστικά αυτά υπολογίζονται με υψηλές ταχύτητες με τη χρήση της ολοκληρωτικής εικόνας, που παρουσιάστηκε στην προηγούμενη ενότητα. Το πλήθος των χαρακτηριστικών που δημιουργούνται για δείγματα προσώπων 24×24 pixel είναι περίπου 45.000, δηλαδή ένας πολύ μεγάλος αριθμός. Έτσι, είναι απαραίτητη η επιλογή των κυριότερων χαρακτηριστικών [20, 19].

2.1.4. Επιλογή Χαρακτηριστικών – Εκπαίδευση Του Ταξινομητή Με Τον Adaboost

Όπως αναφέρθηκε, το πλήθος των ορθογωνίων χαρακτηριστικών που δημιουργούνται είναι πολύ μεγάλο. Έτσι, ο τελικός ταξινομητής πρέπει να αποτελείται μόνο από τα πιο κρίσιμα και αποτελεσματικά χαρακτηριστικά. Για την επιλογή αυτών των κρίσιμων χαρακτηριστικών χρησιμοποιείται ένας αλγόριθμος μάθησης, ο AdaBoost. Ο AdaBoost (Adaptive Boosting) είναι ένας δημοφιλής αλγόριθμος προσαρμοστικής ενίσχυσης, που

χρησιμοποιείται από τους Viola & Jones τόσο για την επιλογή των κρίσιμων χαρακτηριστικών όσο και για την εκπαίδευση του ταξινομητή.



Εικόνα 2.4. Τα δύο πιο χρήσιμα χαρακτηριστικά (τα μάτια και η διαφορά τους από την άνω περιοχή των μάγουλων και δεύτερον η διαφορά μεταξύ της περιοχής των ματιών και της γέφυρας της μύτης) [20]

Αρχικά, ο AdaBoost επιλέγει τα πλέον χρήσιμα χαρακτηριστικά από το σύνολο χαρακτηριστικών. Στη συνέχεια, κατασκευάζει αδύναμους ταξινομητές τόσους όσα και τα χρήσιμα χαρακτηριστικά, οι οποίοι στηρίζονται σε ένα μόνο από αυτά τα χαρακτηριστικά. Τέλος, συνδυάζει τους αδύναμους ταξινομητές σε έναν ισχυρό ταξινομητή, ο οποίος προκύπτει από το άθροισμα των ειδικών βαρών των αδύναμων ταξινομητών.

Ο αδύναμος ταξινομητής περιγράφεται από τη σχέση:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } pf(x) < p\theta \\ 0 & \text{otherwise} \end{cases} \quad (2.2)$$

όπου $h(x, f, p, \theta)$ είναι ο αδύναμος ταξινομητής, f το χαρακτηριστικό, θ το κατώφλι και p η πιθανότητα που δείχνει την κατεύθυνση της ανισότητας. Η τιμή του κατωφλίου ορίζεται έτσι ώστε για κάθε χαρακτηριστικό να ταξινομείται λάθος ο μικρότερος δυνατός αριθμός δειγμάτων [20,19].

Η διαδικασία κατασκευής του ισχυρού ταξινομητή με βάση τους Viola & Jones [20,19] από τον AdaBoost είναι η εξής:

1. Είσοδος: Εικόνες Εκπαίδευσης (x_i, y_i) , $i = 1 \dots N$ με $y_i = 0,1$ για αρνητικά (εικόνες που δεν περιέχουν πρόσωπο ή το προς ανίχνευση αντικείμενο) και θετικά (εικόνες που περιέχουν πρόσωπο ή το προς ανίχνευση αντικείμενο) παραδείγματα αντίστοιχα.
2. Αρχικοποίηση: Βάρη, όπου m αρνητικά και l θετικά παραδείγματα.

$$\omega_{1,i} = \frac{1}{2m}, \frac{1}{2l} \quad (2.3)$$

3. Για $t=1 \dots T$ γίνεται:

- Κανονικοποίηση των βαρών

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}} \quad (2.4)$$

- Επιλογή του καλύτερου αδύναμου ταξινομητή ανάλογα με το σφάλμα

$$\epsilon_t = \min_{f,p,\theta} \sum_i w_i |h(x_i, f, p, \theta) - y_i| \quad (2.5)$$

- Ορισμός του $h_t(x) = h(x, f_t, p_t, \theta_t)$ όπου f_t, p_t, θ_t οι τιμές που ελαχιστοποιούν το σφάλμα
- Ενημέρωση των βαρών

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i} \quad \text{και} \quad \beta_t = \frac{\epsilon_t}{1 - \epsilon_t} \quad (2.6)$$

όπου $e_i = 0$ εάν το δείγμα x_i έχει ταξινομηθεί σωστά, ενώ $e_i = 1$ εάν το δείγμα x_i δεν έχει ταξινομηθεί σωστά.

4. Ο τελικός ισχυρός ταξινομητής είναι:

$$h(x) = \begin{cases} 1, & \sum_{t=1}^T a_t h_t(x) \geq 0.5 \sum_{t=1}^T a_t \\ 0, & \text{διαφορετικά} \end{cases} \quad (2.7)$$

$$a_t = \log\left(\frac{1}{\beta_t}\right)$$

2.1.5. Ταξινόμηση Με Έναν Καταρράκτη (Cascade) Ταξινομητών

Όπως αναφέρθηκε στην εισαγωγή της παρούσας ενότητας, στόχος των Viola & Jones ήταν η ανάπτυξη ενός γρήγορου και αποτελεσματικού συστήματος ανίχνευσης και αναγνώρισης προσώπων. Γενικά, σε μια εικόνα, ακόμα και όταν περιλαμβάνονται περισσότερα από ένα πρόσωπα, ο μεγαλύτερος αριθμός παραθύρων υπολογισμού δεν περιλαμβάνει πρόσωπα. Έτσι, ο εντοπισμός των προσώπων είναι πιο χρονοβόρα διαδικασία από την απόρριψη μη – προσώπων.

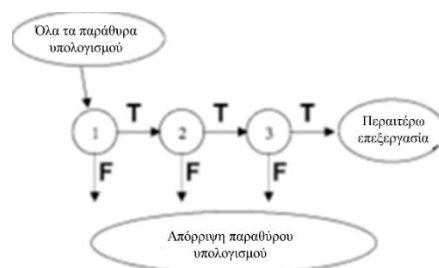
Για το λόγο αυτό, οι Viola & Jones ακολούθησαν μια διαφορετική πορεία. Εισήγαγαν την έννοια του cascade ταξινομητή ή αλλιώς καταρράκτη ταξινομητών. Η βασική παραδοχή

είναι ότι μπορούν να κατασκευαστούν απλοί και ταυτόχρονα αποτελεσματικοί ταξινομητές που απορρίπτουν πολλά από τα αρνητικά παράθυρα υπολογισμού, ενώ ανιχνεύουν σχεδόν όλα τα θετικά παράθυρα υπολογισμού.

Ο ταξινομητής cascade, λοιπόν, αποτελείται από σειριακά τμήματα, καθένα από τα οποία είναι ένας ισχυρός ταξινομητής και εκπαιδεύεται από τον AdaBoost. Η εκπαίδευση του καταρράκτη ταξινομητή χρησιμοποιώντας τον AdaBoost καθορίζει τον αριθμό των επιπέδων του καταρράκτη ταξινομητών, τον αριθμό των χαρακτηριστικών σε κάθε επίπεδο και το κατώφλι σε κάθε επίπεδο. Κάθε επιμέρους ταξινομητής καθορίζει εάν ένα παράθυρο υπολογισμού είναι πρόσωπο ή μη-πρόσωπο. Κάθε παράθυρο που ταξινομείται ως μη-πρόσωπο απορρίπτεται και δεν επεξεργάζεται περισσότερο από τους συνθετότερους ταξινομητές του επόμενου επιπέδου. Αντίθετα, κάθε παράθυρο που ταξινομείται ως πιθανό πρόσωπο συνεχίζει στο επόμενο επίπεδο επεξεργασίας.

Επομένως, οι απλοί και λιγότερο χρονοβόροι ταξινομητές χρησιμοποιούνται για να απορρίψουν την πλειονότητα των παραθύρων υπολογισμού ως αρνητικά, προτού χρησιμοποιηθούν οι πιο σύνθετοι και περισσότερο χρονοβόροι ταξινομητές που θα επεξεργαστούν τις πιο πολύπλοκες περιπτώσεις παραθύρων υπολογισμού και θα επιτύχουν χαμηλά επίπεδα εσφαλμένων θετικών ανιχνεύσεων. Έτσι, η περισσότερη υπολογιστική ισχύς αναλώνεται σε περιοχές της εικόνας που πιθανόν περιέχουν πρόσωπα και η ταχύτητα του συστήματος αυξάνεται.

Όπως φαίνεται και στην Εικόνα 2.5. η διαδικασία ανίχνευσης μοιάζει με ένα δέντρο απόφασης (decision tree). Ένα θετικό αποτέλεσμα από τον ταξινομητή πρώτου επιπέδου πυροδοτεί την επεξεργασία από τον ταξινομητή δεύτερου επιπέδου, του οποίου το θετικό αποτέλεσμα πυροδοτεί την επεξεργασία από τον ταξινομητή τρίτου επιπέδου κ.ό.κ. Ένα αρνητικό αποτέλεσμα από τον ταξινομητή οποιουδήποτε επιπέδου σταματά την περαιτέρω επεξεργασία του παραθύρου υπολογισμού απορρίπτοντάς το [20,19].



Εικόνα 2.5. Σχηματική αναπαράσταση ενός cascade ταξινομητή [20]

2.1.6. Μέθοδος Κατά Lienhart

Οι Lienhart et al. εξέλιξαν την μέθοδο των Viola & Jones εισάγοντας τις εξής βελτιώσεις:

1. Ένα εκτεταμένο σύνολο χαρακτηριστικών Haar: τα οποία είναι περιστραμμένα κατά 45°, τα οποία βελτιώνουν την αποτελεσματικότητα του συστήματος, παρέχοντας περισσότερες πληροφορίες.
2. CART δέντρα ως αδύναμοι ταξινομητές: οι αδύναμοι ταξινομητές του Lienhart είναι Δέντρα Ταξινόμησης και Παλινδρόμησης (Classification and Regression Trees-CART) με έως 4 χαρακτηριστικά. Η χρήση τους οδήγησε σε μεγαλύτερη αποδοτικότητα του συστήματος σε σχέση με τους αδύναμους ταξινομητές ενός χαρακτηριστικού.
3. Παραλλαγή του AdaBoost: αξιοποιήθηκε η Gentle μορφή της AdaBoost, όπου σε κάθε γύρο της ενίσχυσης προστίθεται ο ταξινομητής που ταξινομεί ορθότερα τα δείγματα εκπαίδευσης με τη χρήση του αποτελεσματικότερου CART δένδρου [15].

2.2. Η Μέθοδος Των Τοπικών Δυαδικών Προτύπων

Η μέθοδος των Τοπικών Δυαδικών Προτύπων (Local Binary Patterns Histograms - LBPH) αναπτύχθηκε από τους Ahonen et al. και έχει αποδειχθεί αποτελεσματική, ιδιαίτερα εξαιτίας του ότι μπορεί να επιτύχει υψηλά ποσοστά αναγνώρισης ακόμα και σε όχι ιδανικές συνθήκες φωτισμού.

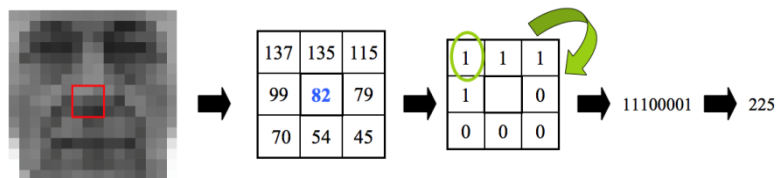
Ο LBP περιγραφέας αναπτύχθηκε αρχικά για την περιγραφή της υφής των εικόνων και βασίζεται στην ιδέα ότι δεν είναι απαραίτητο να εξετάζεται μια εικόνα ως σύνολο, αλλά αρκεί να βρεθούν τοπικά χαρακτηριστικά [29].

Ο πρωτότυπος τελεστής αποδίδει μια τιμή σε κάθε pixel της εικόνας ως εξής:

$$\text{LBP}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c) \quad (2.8)$$

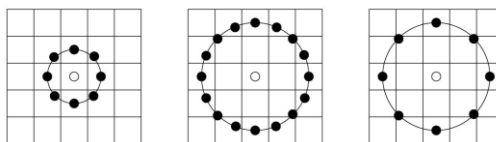
Αρχικά, η grayscale εικόνα χωρίζεται σε επιμέρους τμήματα – block μεγέθους 3 x 3. Υπολογίζεται έπειτα η τιμή του κεντρικού pixel (με βάση την έντασή του), χρησιμοποιείται ως κατώφλι και συγκρίνεται με τα γειτονικά. Εάν η τιμή των γειτονικών pixel είναι μεγαλύτερη από του κεντρικού, τους αποδίδεται η τιμή 1, ενώ εάν είναι μικρότερη τους αποδίδεται η τιμή 0. Το αποτέλεσμα είναι ένας δυαδικός αριθμός. Αυτή η διαδικασία ακολουθείται σε ολόκληρη την εικόνα και έτσι τελικά έχουμε μία λίστα με τοπικά δυαδικά πρότυπα. Στη συνέχεια, οι δυαδικοί αριθμοί μετατρέπονται σε δεκαδικούς.

Επόμενο βήμα είναι η κατασκευή ενός ιστογράμματος αυτών των δεκαδικών τιμών για κάθε εικόνα σε μια βάση δεδομένων [8, 24]. Τα ιστογράμματα θα χρησιμοποιηθούν στη συνέχεια για την αναγνώριση των προσώπων.



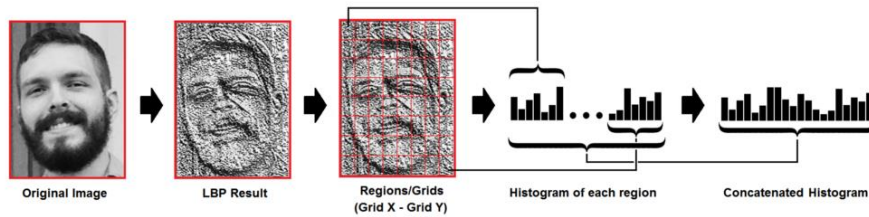
Εικόνα 2.6. Η διαδικασία υπολογισμού των τιμών των pixel με τα Τοπικά Δυαδικά Πρότυπα [24]

Μία επέκταση της μεθόδου, είναι τα Κυκλικά Τοπικά Δυαδικά Πρότυπα (Circular Local Binary Patterns) τα οποία χρησιμοποιούν γειτονικά pixel διαφορετικών μεγεθών – κλίμακας. Με κέντρο του κύκλου το κεντρικό pixel P, μπορούν να οριστούν γειτονικά pixel τα οποία βρίσκονται σε απόσταση από το κεντρικό ίση με την ακτίνα R που έχει καθοριστεί κάθε φορά [8].



Εικόνα 2.7. Κυκλικά Τοπικά Δυαδικά Πρότυπα [8]

Επίσης, και τα τοπικά ιστογράμματα έχουν εξελιχθεί. Πέρα από το βασικό ιστόγραμμα έχει αναπτυχθεί το Χωρικά Ενισχυμένο Ιστόγραμμα (Spatially Enhanced Histogram), το οποίο επιτρέπει την κωδικοποίηση τόσο της εμφάνισης των προσώπων όσο και τις χωρικές σχέσεις μεταξύ των περιοχών των προσώπων. Κατασκευάζεται, λοιπόν, ένα ιστόγραμμα για κάθε περιοχή του προσώπου και τα επιμέρους ιστογράμματα συνδυάζονται για να δώσουν το Χωρικά Ενισχυμένο Ιστόγραμμα όλης της εικόνας [8].



Εικόνα 2.8. Η Κατασκευή των Τοπικών Ιστογραμμάτων [41]

Στη συνέχεια, ακολουθεί η εκπαίδευση του αλγορίθμου. Σε μία βάση δεδομένων εισάγονται εικόνες των προσώπων που το σύστημα θα πρέπει να αναγνωρίζει και ορίζεται μία ταυτότητα (ID) για καθένα από αυτά. Εάν υπάρχουν περισσότερες από μία εικόνες του ίδιου προσώπου πρέπει, βέβαια, να έχουν όλες το ίδιο ID. Ο αλγόριθμος θα κατασκευάσει, με τη μεθοδολογία που περιγράφηκε παραπάνω, τα ιστογράμματα των προσώπων και θα τα αποθηκεύσει μαζί με την ταυτότητά τους.

Έτσι, όταν το σύστημα αναγνώρισης θα τεθεί σε εφαρμογή, ο αλγόριθμος θα κατασκευάσει ένα ιστόγραμμα για την νέα εικόνα εισόδου, με σκοπό να το συγκρίνει με τα ήδη αποθηκευμένα και να αποφασίσει εάν υπάρχει ταύτιση ή όχι. Η τελική απόφαση για την ύπαρξη ταύτισης θα παρθεί με βάση κάποια κριτήρια. Ένα από αυτά τα κριτήρια, το οποίο προτείνεται από τους Ahonen et al. είναι το στατιστικό κριτήριο χ^2 (chi square):

$$\chi_w^2(\mathbf{x}, \boldsymbol{\xi}) = \sum_{j,i} w_j \frac{(x_{i,j} - \xi_{i,j})^2}{x_{i,j} + \xi_{i,j}}, \quad (2.9)$$

Όπου \mathbf{x} και $\boldsymbol{\xi}$ είναι τα δύο χωρικά ενισχυμένα ιστογράμματα που συγκρίνονται, ο δείκτης i είναι η i -οστή περιοχή του ιστογράμματος που αντιστοιχεί στην j -οστή περιοχή του προσώπου και w_j το βάρος για την περιοχή j στην οποία έχει αποδοθεί μεγάλη σημασία εξαιτίας των πληροφοριών που περιέχει (αφορά συνήθως την περιοχή των ματιών, αφού αυτή έχει φανεί ερευνητικά ότι είναι η πλέον σημαντική για την αναγνώριση προσώπων).

Τελικά, ο αλγόριθμος δίνει ως αποτέλεσμα την ταυτότητα (ID) του προσώπου με το περισσότερο ταυτόσημο ιστόγραμμα. Επίσης, δίνει την εκτιμώμενη απόσταση (μεταξύ των δύο ιστογραμμάτων), η οποία καθορίζει το επίπεδο εμπιστοσύνης (confidence) της μέτρησης. Το επίπεδο εμπιστοσύνης σε συνδυασμό με το κατώφλι που ορίζει ο χειριστής μπορούν να χρησιμοποιηθούν για να γίνει αυτόματα εκτίμηση του αν ο αλγόριθμος αναγνώρισε τελικά σωστά το πρόσωπο. Εάν το επίπεδο εμπιστοσύνης είναι μικρότερο από το κατώφλι θεωρείται ότι η αναγνώριση έγινε σωστά [8, 24, 41].

2.3. Βιβλιοθήκη OpenCV

Η OpenCV (Open Source Computer Vision Library) είναι μια βιβλιοθήκη λογισμικού με κύριο στόχο την επεξεργασία βίντεο και εικόνας πραγματικού χρόνου και την ανάπτυξη εφαρμογών υπολογιστικής, και κατά συνέπεια ρομποτικής, όρασης. Αναπτύχθηκε από την εταιρεία Intel και διατίθεται δωρεάν ως ελεύθερο λογισμικό. Έτσι, χρησιμοποιείται ευρέως τόσο από εταιρίες όσο και από ερευνητικές ομάδες.

Η βιβλιοθήκη περιλαμβάνει μια ολοκληρωμένη σειρά αλγορίθμων υπολογιστικής όρασης και εκμάθησης μηχανών. Οι αλγόριθμοι αυτοί μπορούν να χρησιμοποιηθούν για την ανίχνευση και αναγνώριση προσώπων ή αντικειμένων, για την αναζήτηση παρόμοιων εικόνων σε μια βάση δεδομένων, για την ταξινόμηση ανθρώπινων ενεργειών σε βίντεο, για την παρακολούθηση κινούμενων αντικειμένων, για την κατασκευή τρισδιάστατων μοντέλων των αντικειμένων, για την παρακολούθηση των κινήσεων του ματιού, για την αναγνώριση σκηνικού σε εικόνες και σε άλλες πολλές εξειδικευμένες εφαρμογές ρομποτικής όρασης όπως η ιατρική απεικόνιση, η επιθεώρηση της παραγωγής σε εργοστασιακές μονάδες, τα συστήματα ασφαλείας, η στερεοσκοπική όραση κτλ. Μάλιστα, ένα από τα πλεονεκτήματά της είναι η ταχύτητά της, εφόσον μπορεί να επεξεργάζεται εικόνες και βίντεο σε πραγματικό χρόνο εξυπηρετώντας έτσι την αποτελεσματικότητα των αναπτυσσόμενων εφαρμογών.

Η βιβλιοθήκη αρχικά ήταν γραμμένη σε γλώσσα C αλλά πλέον μπορεί να χρησιμοποιηθεί με γλώσσες όπως C#, Python, Ruby και Java και προγράμματα όπως το Matlab. Μπορεί να υποστηρίξει όλα τα ευρέως διαδεδομένα λειτουργικά συστήματα όπως τα Windows, τα Linux και τα MacOS [40, 1].



Εικόνα 2.9. Το λογότυπο της OpenCV [40]

2.4. Γλώσσα Προγραμματισμού Python

Η Python είναι μια δυναμική γλώσσα προγραμματισμού υψηλού επιπέδου, γενικού σκοπού και ανοικτού πηγαίου κώδικα (open source). Αναπτύχθηκε στις αρχές της δεκαετίας

του 1990 από τον Guido van Rossum. Είναι ενδιαφέρον ότι το όνομά της βασίστηκε σε μία ομάδα κωμικών, και συγκεκριμένα τους “Monty Python”. Η Python χρησιμοποιείται με μεγάλη επιτυχία στην ανάπτυξη εφαρμογών σε όλα τα ευρέως διαδεδομένα λειτουργικά συστήματα [6].

Η πλέον πρόσφατη έκδοση της Python είναι η 3.0, η οποία κυκλοφόρησε το Δεκέμβριο του 2008. Ωστόσο, ακόμη χρησιμοποιείται ευρέως και η έκδοση 2.7. Το τέλος ζωής για την Python 2.7 είχε προγραμματιστεί για το 2015, ωστόσο δόθηκε παράταση μέχρι το 2020 εξαιτίας του ότι ένα μεγάλο μέρος του υπάρχοντος κώδικα είναι δύσκολο να μεταφερθεί στην επόμενη έκδοση [27].

Βασικά χαρακτηριστικά της Python είναι:

- Ο αντικειμενοστρεφής προγραμματισμός (object-oriented programming)
- Ο προστακτικός προγραμματισμός (imperative programming)
- Ο συναρτησιακός προγραμματισμός (functional programming)
- Ο διαδικαστικός προγραμματισμός (procedural styles programming)
- Ο δυναμικός καθορισμός των τύπων δεδομένων
- Η αυτοματοποιημένη διαχείριση της μνήμης
- Η μεγάλη ενσωματωμένη βιβλιοθήκη.

Ένα βασικό πλεονέκτημα της συγκεκριμένης γλώσσας είναι η έμφαση που έχει δοθεί στην εύκολη ανάγνωση και σύνταξη των κωδίκων, χαρακτηριστικό το οποίο επιτρέπει στους προγραμματιστές να χρησιμοποιούν λιγότερες γραμμές για τον κώδικά τους, σε σχέση με άλλες γλώσσες προγραμματισμού όπως η C++ και η Java[4].



Εικόνα 2.10. Το λογότυπο της Python [42]

2.5. Ανίχνευση Και Αναγνώριση Προσώπου Με Την OpenCV Και Python

Η ανίχνευση προσώπου με την OpenCV γίνεται με τη χρήση συγκεκριμένων μεθόδων ταξινόμησης. Η OpenCV παρέχει δύο ταξινομητές:

1. Τον Haar Classifier, που βασίζεται στη θεωρία των Viola & Jones
2. Τον LBP Classifier, που βασίζεται στη θεωρία των Τοπικών Δυαδικών Προτύπων.

Και οι δύο ταξινομητές έχουν πλεονεκτήματα και μειονεκτήματα. Ο Haar Classifier χαρακτηρίζεται από υψηλή ακρίβεια ανίχνευσης και χαμηλά ποσοστά ψευδώς θετικών ανιχνεύσεων. Όμως, απαιτεί αρκετά μεγάλο χρόνο εκπαίδευσης και η αποδοτικότητά του περιορίζεται σε όχι ιδανικές συνθήκες φωτισμού. Από την άλλη, ο LBP Classifier είναι υπολογιστικά απλός και απαιτεί σχετικά μικρότερο χρόνο εκπαίδευσης, ενώ είναι και ισχυρός απέναντι σε μεταβολές του φωτισμού. Ωστόσο, δεν είναι το ίδιο ακριβής με τον Haar Classifier ενώ έχει και υψηλά ποσοστά ψευδώς θετικών ανιχνεύσεων [41, 39].

Ο Haar Classifier της OpenCV παρέχει αρκετούς αλγόριθμους ανίχνευσης, που είναι αποθηκευμένοι σε αρχεία .xml, ανάλογα με το στόχο του κάθε ερευνητή:

- Ανίχνευση των ματιών (π.χ. haarcascade_eye.xml)
- Ανίχνευση του ανθρώπινου σώματος (π.χ. haarcascade_fullbody.xml, haarcascade_lowerbody.xml, haarcascade_upperbody.xml)
- Ανίχνευση του προσώπου σε προφίλ (π.χ. haarcascade_profileface.xml)
- Ανίχνευση του προσώπου κατά μέτωπο (π.χ. haarcascade_frontalface.xml, haarcascade_frontalface_alt.xml, haarcascade_frontalface_alt2.xml, haarcascade_frontalface_alt_tree.xml, haarcascade_frontalface_default.xml)
- Ανίχνευση του χαμόγελου κτλ. [39]

Αντίστοιχα, και η αναγνώριση προσώπου με την OpenCV γίνεται με τη χρήση συγκεκριμένων μεθόδων. Η OpenCV παρέχει 3 αλγόριθμους για αναγνώριση προσώπου:

1. Τον Eigenfaces Face Recognizer
2. Τον Fisherfaces Face Recognizer
3. Τον LBPH Face Recognizer (Local Binary Patterns Histograms).

Και οι τρεις αλγόριθμοι εμφανίζουν δυνάμεις και αδυναμίες. Οι αλγόριθμοι των Eigenfaces και των Fisherfaces επιτυγχάνουν πολύ υψηλά ποσοστά αναγνώρισης, ωστόσο απαιτούν αρκετά καλές συνθήκες φωτισμού και περίπου 8 + - 1 φωτογραφίες για το κάθε άτομο της βάσης εγγεγραμμένων προσώπων [29, 30].

Με βάση τα κριτήρια αυτά, για την παρούσα εργασία αξιοποιήθηκαν:

- Η γλώσσα προγραμματισμού Python λόγω της ευκολίας χρήσης της (π.χ. συντομία κώδικα, πλούσια βιβλιοθήκη) και του συνδυασμού της με τη βιβλιοθήκη OpenCV
- Ο Haar Classifier εφόσον κύριος στόχος είναι η ανίχνευση προσώπων με αυξημένα ποσοστά ακρίβειας και όχι πρωταρχικά η ταχύτητα του συστήματος
- Ο αλγόριθμος LBPH για την αναγνώριση προσώπων.

Κεφάλαιο 3: Ανίχνευση Στάθμης Υγρού Με Χρήση OpenCV & Python

3.1. Εισαγωγή

Η ανίχνευση της στάθμης υγρών είναι μία πολύ σημαντική διεργασία και με πολλές εφαρμογές τόσο σε βιομηχανικό επίπεδο (π.χ. ανίχνευση στάθμης σε δεξαμενές πετρελαίου, ανίχνευση στάθμης σε γραμμές παραγωγής πλήρωσης μπουκαλιών ή άλλων δοχείων) όσο και επίπεδο εργαστηρίων χημείας [23, 10].

Η ανίχνευση της στάθμης υγρών είναι ένα ερευνητικό πρόβλημα που μελετάται εδώ και δεκαετίες, με αποτέλεσμα να έχουν διαμορφωθεί αρκετές διαφορετικές προσεγγίσεις. Αρχικά, η κύρια μέθοδος που χρησιμοποιούνταν ήταν οι PID Ελεγκτές, εξαιτίας της ευχρηστίας τους και της ποικιλίας των εφαρμογών τους [18]. Ωστόσο, με την ανάπτυξη της υπολογιστικής όρασης νέες προσεγγίσεις έχουν αρχίσει να κερδίζουν έδαφος. Το βασικό πλεονέκτημα των μεθόδων που στηρίζονται σε οπτικά δεδομένα είναι η ασφάλεια που παρέχουν κυρίως όταν εστιάζουμε σε εφαρμογές σε βιομηχανικό επίπεδο. Με άλλα λόγια, οι μέθοδοι υπολογιστικής όρασης είναι περισσότερο ασφαλείς γιατί μπορούν να εφαρμοστούν εξ αποστάσεως και έτσι, παραδείγματος χάριν, να μην εκθέτουν το ανθρώπινο δυναμικό σε κίνδυνο σε περιπτώσεις εκρήξεων, υψηλών θερμοκρασιών, αναθυμιάσεων κτλ. [10].

Μπορούμε να διακρίνουμε τις εξής προσεγγίσεις ανίχνευσης με υπολογιστική όραση:

1. Ανίχνευση στάθμης υγρών με βάση υπολογιστική όραση και εξωτερικά πρόσθετα: έχουν χρησιμοποιηθεί διαφορετικών ειδών πρόσθετα, όπως πολύχρωμες επιπλέουσες χάντρες (των οποίων η πυκνότητα τους επιτρέπει να επιπλέον ακριβώς στην επιφάνεια του υγρού) ή ακτίνες λέιζερ. Αυτά τα πρόσθετα αναγνωρίζονται με ευκολία στην εικόνα και η θέση τους χρησιμοποιείται για τον καθορισμό της στάθμης του υγρού.
2. Ανίχνευση στάθμης με βάση το χρώμα: οι μέθοδοι αυτοί εφαρμόζονται όταν τα προς εξέταση υγρά έχουν κάποιο χαρακτηριστικό χρώμα. Η ανίχνευση βασίζεται στην εύρεση της περιοχής της εικόνας που ανταποκρίνεται στο συγκεκριμένο χρώμα. Προφανώς, απαιτείται ο καθορισμός ενός χρώματος φόντου και έτσι οι μέθοδοι αυτοί μπορούν να αξιοποιηθούν για κάθε αδιαφανές υγρό.
3. Ανίχνευση στάθμης με βάση την εικόνα ακρών (edge image): αυτή η προσέγγιση θεωρείται η πιο αποτελεσματική ειδικά όταν αφορά διαφανή υγρά. Οι μέθοδοι αυτοί

χρησιμοποιούν την εικόνα ακρών (edge-image) για να αναγνωρίσουν την ευθεία γραμμή της επιφάνειας του υγρού [23].

4. Ανίχνευση στάθμης υγρών με τη συμβολή νευρωνικών δικτύων [18].

Επομένως, όπως αναφέραμε, η επικρατέστερη προσέγγιση πλέον είναι η ανίχνευση με βάση την edge image. Η ανίχνευση της στάθμης του υγρού γίνεται σε 3 στάδια:

1. Επιλογή της Περιοχής Ενδιαφέροντος (Region of Interest – ROI)
2. Εφαρμογή της συνάρτησης Canny για την εύρεση των ακρών της εικόνας
3. Εφαρμογή του μετασχηματισμού γραμμών Hough (Hough lines transform) για την εύρεση της μεγαλύτερου μήκους οριζόντιας ευθείας γραμμής, η οποία θα αντιστοιχεί στην επιφάνεια του υγρού [18].

3.2. Επιλογή Περιοχής Ενδιαφέροντος (ROI)

Η Περιοχή Ενδιαφέροντος (Region of Interest – ROI) είναι ένα τμήμα της εικόνας στο οποίο πρόκειται να γίνει κάποια επεξεργασία – να εφαρμοστεί κάποια διαδικασία. Στις εφαρμογές υπολογιστικής όρασης το ROI ορίζεται ως τα όρια της περιοχής που περιέχει το προς εξέταση αντικείμενο. Μάλιστα, μέσα σε μία Περιοχή Ενδιαφέροντος μπορεί να περιέχονται περισσότερα ανεξάρτητα Σημεία Ενδιαφέροντος (Points of Interest – POIs) [38].

Η βιβλιοθήκη της OpenCV παρέχει έτοιμη συνάρτηση επιλογής περιοχών ενδιαφέροντος. Ωστόσο, υπάρχουν και εναλλακτικοί τρόποι επιλογής ενός τέτοιου παραθύρου, όπως ο καθορισμός ενός συγκεκριμένου σχήματος παραθύρου επιλογής από τους χρήστες (π.χ. τετράγωνο).

3.3. Εφαρμογή Της Συνάρτησης Canny Edge Detector

Η Ανίχνευση Ακρών Canny (Canny Edge Detection) είναι ένας δημοφιλής αλγόριθμος ανίχνευσης ακρών. Αναπτύχθηκε από τον John F. Canny το 1986 [9]. Η τεχνική εύρεσης των ακρών της εικόνας είναι μια μέθοδος που επιτρέπει αφενός τον καθορισμό των δομικών πληροφοριών των αντικειμένων που περιέχονται σε μια εικόνα, αφετέρου μειώνει δραματικά την ποσότητα των πληροφοριών που πρέπει να επεξεργαστεί ένα σύστημα προκειμένου να ολοκληρώσει κάποιες διεργασίες (π.χ. στη συγκεκριμένη περίπτωση την ανίχνευση της στάθμης υγρού) [35].

Ο αλγόριθμος Canny Edge Detector έχει επικρατήσει χάρη στα τρία κριτήρια απόδοσης, τα οποία έχει καταφέρει να πληροί:

1. Υψηλά ποσοστά ανίχνευσης: Στόχος είναι να υπάρχει χαμηλή πιθανότητα να ανιχνευθεί ένα σημείο ψευδώς ως άκρη (χαμηλό ποσοστό λάθους) και ταυτόχρονα να είναι χαμηλή και η πιθανότητα να μην αναγνωριστεί ένα πραγματικό σημείο άκρης ως τέτοιο (υψηλό ποσοστό ανίχνευσης).
2. Καλή τοπικοποίηση (good localization): Τα σημεία που χαρακτηρίζονται άκρες από τον τελεστή πρέπει να είναι όσο το δυνατόν πιο κοντά στις πραγματικές άκρες της εικόνας.
3. Μία απόκριση ανά άκρη (one response to a single edge): Αυτό το κριτήριο, αν και σχετίζεται με τα υψηλά ποσοστά ανίχνευσης, μπορεί να διατυπωθεί αυτόνομα, ώστε να εστιάζει στις ψευδείς ανιχνεύσεις λόγω θορύβου ή διακυμάνσεων του χρώματος της αρχικής εικόνας [9].

Ο αλγόριθμος της Canny Edge Detector είναι ένας αλγόριθμος που διαρθρώνεται στα εξής στάδια:

1. Μείωση του Θορύβου (Noise Reduction)
2. Εύρεση της κλίσης έντασης της εικόνας (Finding Intensity Gradient of the image)
3. Μη-μέγιστη συμπίεση (Non-maximum Suppression)
4. Υστέρηση κατωφλίων (Hysteresis Thresholding) [32].

3.3.1. Μείωση Του Θορύβου (Noise Reduction)

Επειδή η ανίχνευση των ακρών είναι επιρρεπής στο θόρυβο, είναι απαραίτητο, σαν πρώτο βήμα, να αφαιρεθεί ο θόρυβος από την εικόνα. Αυτό επιτυγχάνεται με την εφαρμογή κάποιων φίλτρων στην αρχική εικόνα. Η Canny χρησιμοποιεί το Γκαουσιανό Φίλτρο (Gaussian Filter) [32].

Στην επεξεργασία εικόνας, το Gaussian Filter ή αλλιώς Gaussian Blur χρησιμοποιείται για τη μείωση του θορύβου στο στάδιο προ-επεξεργασίας της εικόνας. Θεωρείται ως το πιθανόν χρησιμότερο φίλτρο, αν και δεν είναι το γρηγορότερο. Το Γκαουσιανό Φίλτρο είναι ένα χαμηλοπερατό φίλτρο (low-pass filter) που χρησιμοποιεί μια Γκαουσιανή συνάρτηση για να υπολογίσει το μετασχηματισμό που πρέπει να εφαρμοστεί σε κάθε pixel της εικόνας:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.1.)$$

Όπου x η απόσταση του σημείου από τον οριζόντιο άξονα, y η απόσταση του σημείου από τον κάθετο άξονα και σ η τυπική απόκλιση της Γκαουσιανής Κατανομής [1, 36].



Εικόνα 3.1. Εφαρμογή του Gaussian Blur [1]

Συνήθως, η OpenCV εφαρμόζει ένα Gaussian filter 5 x 5, όπως το παρακάτω παράδειγμα [33]:

$$K = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad (3.2.)$$

3.3.2. Εύρεση Της Κλίσης Έντασης Της Εικόνας (Finding Intensity Gradient Of The Image)

Η εικόνα που έχει φιλτραριστεί και έχει εξομαλυνθεί φιλτράρεται στη συνέχεια και με ένα Sobel Kernel σε οριζόντια και κάθετη κατεύθυνση για να υπολογιστεί η πρώτη παράγωγος σε οριζόντια G_x και κάθετη κατεύθυνση G_y .

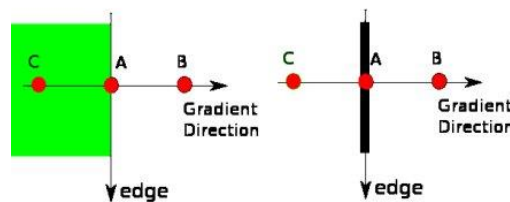
Από αυτά, μπορεί να βρεθεί η κλίση των ακρών (σε γωνίες 0° , 45° , 90° , 135°) και η κατεύθυνσή της για κάθε pixel [32]:

$$\begin{aligned} \text{Edge_Gradient } (G) &= \sqrt{G_x^2 + G_y^2} \\ \text{Angle } (\theta) &= \tan^{-1} \left(\frac{G_y}{G_x} \right) \end{aligned} \quad (3.3.)$$

3.3.3. Μη-Μέγιστη Συμπίεση (Non-Maximum Suppression)

Μετά την απόκτηση των πληροφοριών για το μέγεθος της κλίσης και την κατεύθυνσή της, πραγματοποιείται μία πλήρης σάρωση της εικόνας για να αφαιρεθούν όλα τα pixel που δε γίνεται να αποτελούν άκρες. Για το σκοπό αυτό, σε κάθε pixel ελέγχεται εάν η τιμή του είναι η τοπική μέγιστη σε σχέση με τα γειτονικά του στην κατεύθυνση της κλίσης.

Σύμφωνα με την Εικόνα 3.2, το σημείο A βρίσκεται πάνω σε άκρη. Τα σημεία B και C βρίσκονται στην κατεύθυνση της κλίσης. Ελέγχεται, λοιπόν, το σημείο A σε σχέση με τα σημεία B και C για να φανεί εάν διαμορφώνουν ένα τοπικό μέγιστο. Εάν ναι, επεξεργάζονται σε επόμενο επίπεδο, αλλιώς συμπίεζονται στο 0. Το αποτέλεσμα αυτής της διαδικασίας είναι μια δυαδική εικόνα (binary image) με άκρες [32, 9].



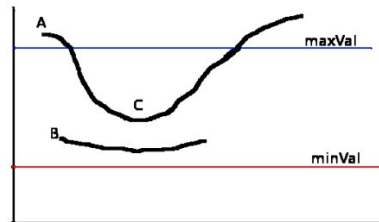
Εικόνα 3.2. Μη-μέγιστη Συμπίεση [32]

Η διαδικασία της μη-μέγιστης συμπίεσης είναι μία τεχνική λέπτυνσης των ακρών (edge thinning technique). Οι άκρες που εξάγονται μέσω του υπολογισμού της τιμής κλίσης είναι σχετικά «θολές». Για να ικανοποιείται το τρίτο κριτήριο απόδοσης, δηλαδή για να υπάρχει μόνο μία απόκριση για κάθε σημείο άκρης, η μη-μέγιστη συμπίεση θέτει σε κάθε pixel την τιμή 0 εκτός από τα τοπικά μέγιστα, τα pixel δηλαδή με την υψηλότερη αλλαγή στην τιμή έντασης [35].

3.3.4. Υστέρηση Κατωφλίων (Hysteresis Thresholding)

Στο στάδιο αυτό κρίνεται εάν οι άκρες που έχουν εντοπιστεί είναι όντως πραγματικές άκρες. Για το λόγο αυτό είναι απαραίτητες δύο τιμές κατωφλίων, η μέγιστη (maxVal) και η ελάχιστη (minVal). Οι τιμές των κατωφλίων ρυθμίζονται εμπειρικά. Η OpenCV προτείνει μία αναλογία 2:1 έως 3:1 ανάμεσα στις τιμές των δύο κατωφλίων, ενώ ο σωστός καθορισμός τους θα καθορίσει τελικά και το πόσο υψηλό θα είναι το ποσοστό σωστών ανιχνεύσεων αλλά και το ποσοστό ψευδών ανιχνεύσεων [33].

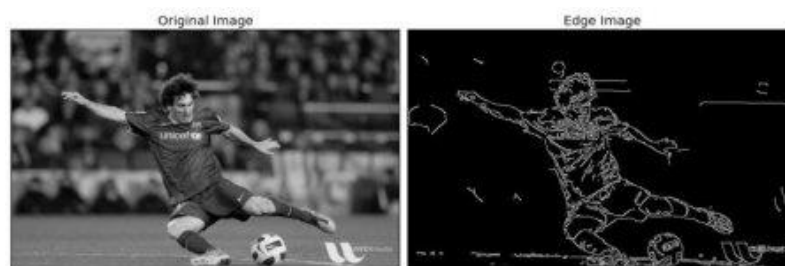
Κάθε άκρη με ένταση κλίσης μεγαλύτερη από τη μέγιστη τιμή κατωφλίου επιβεβαιώνεται ότι είναι άκρη. Κάθε άκρη με ένταση κλίσης μικρότερη από την ελάχιστη τιμή κατωφλίου επιβεβαιώνεται ότι δεν είναι άκρη και έτσι απορρίπτεται. Κάθε άκρη με ένταση κλίσης ανάμεσα στις τιμές των δύο κατωφλίων κατηγοριοποιείται τελικά ως άκρη εάν συνδέεται με μία άλλη άκρη με τιμή μεγαλύτερη του μέγιστου κατωφλίου, ενώ απορρίπτεται εάν δε συνδέεται με μία άλλη άκρη με τιμή μεγαλύτερη του μέγιστου κατωφλίου (βλ. Εικόνα 3.3.).



Εικόνα 3.3. Κατηγοριοποίηση Ακρών με βάση τις Τιμές Κατωφλίου [32]

Επομένως, με τον τρόπο αυτό απορρίπτονται και όλα τα pixel που έχουν λανθασμένα αναγνωριστεί ως άκρες εξαιτίας του εναπομείναντος θορύβου ή διακυμάνσεων του χρώματος στην αρχική εικόνα και σχηματίζεται το τελικό αποτέλεσμα της Εικόνας Ακρών (Edge Image), όπως φαίνεται στην Εικόνα 3.4.[32, 9].

Συμπερασματικά, το πιο σημαντικό χαρακτηριστικό της Canny Edge Detector είναι ότι προσπαθεί να συγκεντρώσει όλα τα υποψήφια pixel ακρών σε ένα ενιαίο περίγραμμα. Αυτό επιτυγχάνεται, όπως αναλύθηκε, από το στάδιο της Υστέρησης Κατωφλίων, κατά το οποίο εξετάζεται η σύνδεση των pixel ακρών ώστε να φανεί ποια είναι όντως άκρες και ποια τελικά όχι [1].

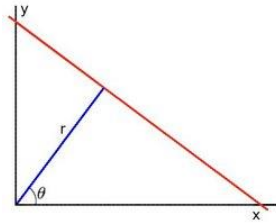


Εικόνα 3.4. Εικόνα Ακρών (Edge Image) [32]

3.4. Εφαρμογή Του Hough Lines Transform

Ο Μετασχηματισμός Hough (Hough Transform) αναπτύχθηκε αρχικά για χρήση του σε πειράματα φυσικής. Οι Duda & Hart τον εισήγαγαν αργότερα στην επεξεργασία εικόνας και στην υπολογιστική όραση. Ο Μετασχηματισμός Hough είναι μια μέθοδος για την εύρεση

γραμμών, κύκλων και άλλων απλών μορφών σε μία εικόνα [1]. Στην παρούσα εργασία η έμφαση δίνεται στον Μετασχηματισμό Γραμμών Hough (Hough Lines Transform) δεδομένου ότι η ανίχνευση αφορά την επιφάνεια υγρού, που είναι ευθεία γραμμή.



Εικόνα 3.5. Αναπαράσταση Γραμμής στο Πολικό Σύστημα Αξόνων [34]

Η βασική θεωρητική θέση του Μετασχηματισμού Γραμμών Hough είναι πως κάθε σημείο – pixel σε μία δυαδική εικόνα μπορεί να είναι τμήμα κάποιας γραμμής. Μία γραμμή στο χώρο της εικόνας μπορεί να εκφραστεί με δύο μεταβλητές είτε στο Καρτεσιανό σύστημα αξόνων (Cartesian Coordinate System) είτε στο Πολικό σύστημα αξόνων (Polar Coordinate System) (βλ. Εικόνα 3.5.). Ο Μετασχηματισμός Hough αξιοποιεί το Πολικό σύστημα αξόνων και διατυπώνει την εξίσωση της γραμμής ως εξής:

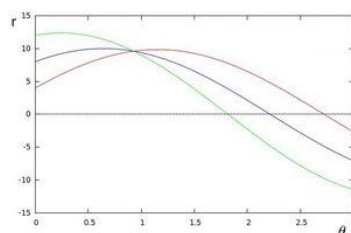
$$y = \left(-\frac{\cos \theta}{\sin \theta} \right) x + \left(\frac{r}{\sin \theta} \right) \quad (3.4.)$$

$$r = x \cos \theta + y \sin \theta$$

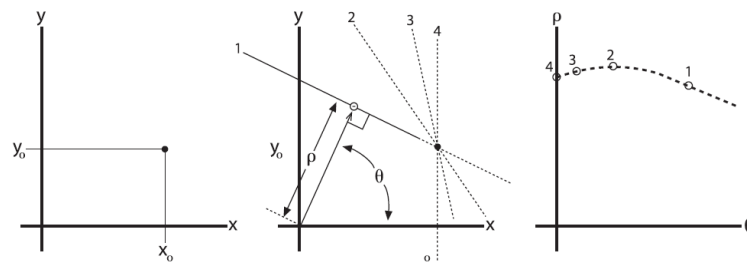
Έτσι, για κάθε σημείο (x_0, y_0) μπορούμε να ορίσουμε τις πιθανές γραμμές που διέρχονται από αυτό, όπου (r_θ, θ) είναι κάθε γραμμή που διέρχεται από το σημείο (x_0, y_0) :

$$r_\theta = x_0 \cdot \cos \theta + y_0 \cdot \sin \theta \quad (3.4.)$$

Για κάθε σημείο (x_0, y_0) εάν σχεδιαστούν οι γραμμές που μπορούν να διέρχονται από αυτό σχηματίζεται μια ημιτονοειδής κυματομορφή. Η ίδια διαδικασία γίνεται για κάθε σημείο της εικόνας. Εάν οι καμπύλες δύο ή περισσότερων σημείων διασταυρώνονται στο πλάνο θ - r σημαίνει πως αυτά τα σημεία ανήκουν στην ίδια γραμμή.



Εικόνα 3.6. Τρεις καμπύλες που διασταυρώνονται σε ένα σημείο. Οι συντεταγμένες τους είναι οι παράμετροι (θ, r) της γραμμής (x_0, y_0) , (x_1, y_1) και (x_2, y_2) στην οποία ανήκουν [34]

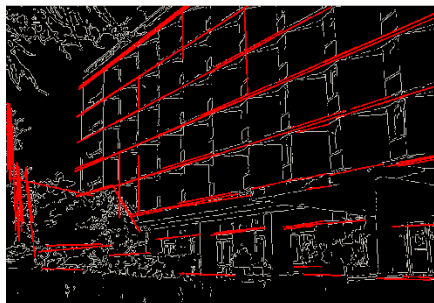


Εικόνα 3.7. Η διαδικασία εύρεσης γραμμών [1]

Επομένως, η μέθοδος του Μετασχηματισμού Hough Lines ανιχνεύει γραμμές βρίσκοντας τον αριθμό των διασταυρώσεων μεταξύ καμπυλών. Όσες περισσότερες καμπύλες διασταυρώνονται τόσα περισσότερα σημεία έχει η γραμμή που αναπαρίσταται από αυτές τις διασταυρώσεις. Ταυτόχρονα, για να ανιχνευθεί μία γραμμή απαιτείται και μία τιμή κατωφλίου, η οποία θα καθορίζει τον ελάχιστο αριθμό διασταυρώσεων που απαιτούνται για να θεωρηθεί ότι σχηματίζεται μία γραμμή [32, 34].

Η OpenCV εφαρμόζει δύο τύπους Μετασχηματισμού Γραμμών Hough:

1. Τον Κλασικό Μετασχηματισμό Γραμμών Hough (Standard Hough Lines Transform)
2. Τον Προοδευτικό Πιθανοτικό Μετασχηματισμό Γραμμών Hough (Progressive Probabilistic Hough Lines Transform).



Εικόνα 3.8. Ανίχνευση Γραμμών με τον Προοδευτικό Πιθανοτικό Μετασχηματισμό Γραμμών Hough [31]

Η διαφορά μεταξύ των δύο μεθόδων έγκειται κυρίως στο ότι ο Πιθανοτικός Μετασχηματισμός είναι πιο αποτελεσματικός και γρήγορος από τον Κλασικό αλγόριθμο και δίνει ως αποτέλεσμα τα άκρα των ανιχνευμένων γραμμών [1].

Ο Πιθανοτικός Μετασχηματισμός υποστηρίζει ότι μπορεί να επιτύχει ανάλογα αποτελέσματα με τον Κλασικό Μετασχηματισμό χρησιμοποιώντας μόνο ένα τμήμα των συνολικών σημείων της εικόνας. Το μέγεθος του δείγματος που θα χρησιμοποιηθεί είναι ανεξάρτητο από τα δεδομένα αλλά εξαρτάται από τη γνώση του ερευνητή για το περιεχόμενο των εικόνων. Όσο πιο άγνωστες είναι οι ιδιότητες των αντικειμένων της εικόνας τόσο

μεγαλύτερο πρέπει να είναι το δείγμα των σημείων, ώστε να μην επηρεαστεί η ακρίβεια και η αποτελεσματικότητα του αλγορίθμου.

Ειδικότερα, ο Προοδευτικός Πιθανοτικός Μετασχηματισμός επιχειρεί να μειώσει ακόμη περισσότερο το χρόνο υπολογισμού εκμεταλλευόμενος τη διαφορά μεταξύ του δείγματος σημείων που χρησιμοποιεί για την ανίχνευση γραμμών και του αριθμού των σημείων που είναι απαραίτητο να «υποστηρίζουν» την ύπαρξη μιας γραμμής [16].

Κεφάλαιο 4: Σκοπός Πτυχιακής Εργασίας

Η παρούσα εργασία, όπως αναφέρθηκε και στην Εισαγωγή, τοποθετείται ερευνητικά στο πεδίο της Ρομποτικής Όρασης. Επιλέξαμε να δώσουμε έμφαση σε περισσότερες από μία εφαρμογές Ρομποτικής Όρασης, τόσο λόγω των ποικίλων ενδιαφερόντων μας, όσο και εξαιτίας της επιθυμίας να αναδείξουμε τον μεγάλο αριθμό εφαρμογών που υπάρχουν και μπορούν να αναπτυχθούν μάλιστα με απλά, ευρέως γνωστά και σχετικά χαμηλού κόστους υλικά.

Έτσι, πρώτος στόχος που τέθηκε ήταν η ανίχνευση και η αναγνώριση προσώπων. Οι δύο αυτές διαδικασίες αποτελούν τη βάση πολυάριθμων άλλων εφαρμογών, ορισμένες από τις οποίες παρουσιάστηκαν στο Κεφάλαιο 2. Σκοπός μας ήταν να αναπτύξουμε ένα σύστημα ανίχνευσης και αναγνώρισης προσώπων με τη μέθοδο των Viola & Jones ως προς την ανίχνευση και τη μέθοδο LBPH ως προς την αναγνώριση, το οποίο θα αναγνωρίζει τα πρόσωπα των συγγραφέων, με το χαμηλότερο δυνατό κόστος. Αξιοποιήθηκε μόνο ένας μικρο-υπολογιστής και μία κάμερα του Raspberry Pi, τα οποία είναι σχετικά οικονομικά, και η βιβλιοθήκη OpenCV σε συνδυασμό με τη γλώσσα προγραμματισμού Python, τα οποία διατίθεται δωρεάν στο διαδίκτυο.

Δεύτερος στόχος που τέθηκε ήταν η ανίχνευση και ο έλεγχος της στάθμης υγρού σε μπουκάλι. Αυτή η εφαρμογή αξιοποιεί μεν τα ίδια υλικά, τα οποία χρησιμοποιήσαμε και στην ανίχνευση και αναγνώριση προσώπων, αλλά διαφοροποιείται μεθοδολογικά από αυτή. Αξιοποιήσαμε διαφορετικές μεθόδους και συγκεκριμένα την ανίχνευση της στάθμης μέσω των αλγορίθμων Canny Edge Detector και Probabilistic Hough Line Transform. Ο έλεγχος της στάθμης έγινε με τη μέτρηση των pixel, διαδικασία η οποία θα παρουσιαστεί αναλυτικά στη μεθοδολογία της έρευνας. Σκοπός μας ήταν να αναπτύξουμε ένα σύστημα ανίχνευσης και ελέγχου της στάθμης του υγρού μέσα σε ένα μπουκάλι, καθώς ένα τέτοιο σύστημα μπορεί να αξιοποιηθεί σε γραμμές παραγωγής σε βιομηχανικό επίπεδο. Οι προϋποθέσεις επιτυχίας μιας τέτοιας εφαρμογής συζητώνται επίσης στη μεθοδολογία της έρευνας.

Κεφάλαιο 5: Μεθοδολογία Έρευνας

5.1. Υλικά

Στην ενότητα αυτή παρουσιάζονται τα υλικά που χρησιμοποιήθηκαν στην παρούσα εργασία.

5.1.1. Raspberry Pi 3 Model B

Το Raspberry Pi είναι μία σειρά μικρο-υπολογιστών που αναπτύχθηκε από τον ομώνυμο οργανισμό της Μεγάλης Βρετανίας, με αρχικό σκοπό τη διδασκαλία βασικών αρχών της επιστήμης των υπολογιστών στα σχολεία. Όμως, πλέον, οι χρήσεις του είναι ποικίλες, από εφαρμογές σε οικιακό επίπεδο (π.χ. μετατροπή ενσύρματου εκτυπωτή σε ασύρματο, μετατροπή τηλεόρασης σε «έξυπνη» τηλεόραση) μέχρι περισσότερες προηγμένες εφαρμογές όπως η σύνδεσή του με διάφορους αισθητήρες (π.χ. θερμοκρασίας, κίνησης) και η οδήγηση ρομποτικών συστημάτων (π.χ. βραχιόνων) [37].



Εικόνα 5.1. Raspberry Pi 3 Model B [44]

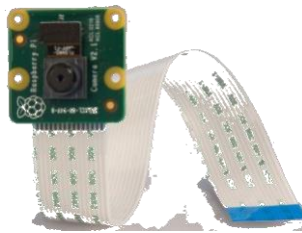
Έχουν κυκλοφορήσει διαφορετικές εκδόσεις και μοντέλα, με πλέον πρόσφατο το Raspberry Pi 3 Model B, το οποίο χρησιμοποιείται στην εργασία. Ο μικρο-υπολογιστής αυτός αποδίδει καλύτερα σε συνδυασμό με ένα λειτουργικό σύστημα που βασίζεται στο Linux. Τα τεχνικά χαρακτηριστικά του Raspberry Pi 3 Model B είναι τα εξής:

- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
- 1GB RAM
- BCM43438 wireless LAN and Bluetooth Low Energy (BLE)
- 40-pin extended GPIO
- 4 USB 2 ports
- 4 Pole stereo output and composite video port

- Full size HDMI
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source up to 2.5A [44].

5.1.2. Raspberry Pi Camera

Ο μικρο-υπολογιστής Raspberry Pi είναι συμβατός με την αντίστοιχη κάμερά του, τη Raspberry Pi Camera με Sony IMX219 8-megapixel αισθητήρα. Η κάμερα αυτή έχει χρησιμοποιηθεί σε ποικίλες εφαρμογές. Επίσης, κυκλοφορεί σε δύο εκδόσεις, την έκδοση που αφορά τη σύλληψη του ορατού φωτός και την έκδοση που αφορά τη σύλληψη υπέρυθρου φωτός [43].



Εικόνα 5.2. Raspberry Pi Camera [43]

5.1.3. Εικόνες Προσώπων

Όσον αφορά τον πρώτο στόχο της εργασίας αυτής, την ανίχνευση και αναγνώριση προσώπων, για την ανίχνευση αξιοποιήθηκαν οι έτοιμες βιβλιοθήκες της OpenCV που έχουν εκπαιδεύσει τους ταξινομητές Cascade στην ανίχνευση προσώπων. Για το έργο της αναγνώρισης προσώπων, χρησιμοποιήθηκαν εικόνες των προσώπων των συγγραφέων της εργασίας καθώς και τα πρόσωπα της βάσης δεδομένων.



Εικόνα 5.3. Δείγματα Εικόνων για την Αναγνώριση Προσώπων

5.1.4. Κατασκευή Για Ανίχνευση Και Έλεγχο Στάθμης Υγρού

Για τον δεύτερο στόχο της εργασίας, την ανίχνευση και τον έλεγχο της στάθμης υγρού σε ένα μπουκάλι, κατασκευάσαμε μία αυτοσχέδια κατασκευή, η οποία αποτελείται από τα εξής υλικά:

1. Κουτί από χαρτόνι
2. Λευκό χαρτόνι (για να δημιουργηθεί ένα σταθερό background)
3. 3 Στήλες με 4 φώτα LED η καθεμία και οι βάσεις τους
4. Βάση στήριξης για το Raspberry Pi και την κάμερά του
5. Βάση από χαρτόνι για το μπουκάλι (ώστε να είναι σταθερό σε συγκεκριμένο ύψος)
6. Γυάλινο μπουκάλι κυλινδρικού σχήματος χωρητικότητας 750ml
7. Τσάι.



Εικόνα 5.4. Κατασκευή Για Ανίχνευση και Έλεγχο Στάθμης Υγρού

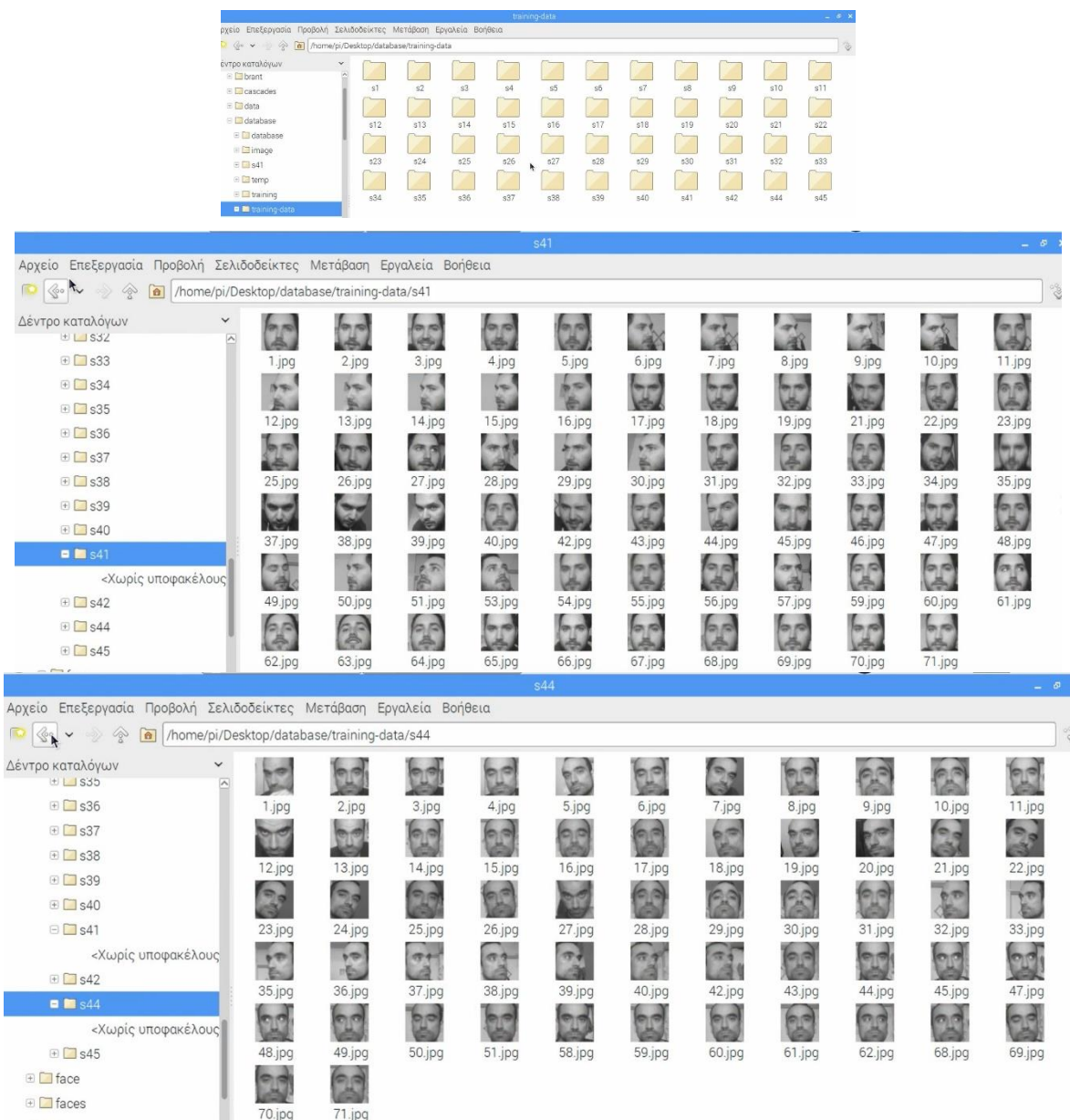
5.2. Κώδικας Ανίχνευσης Και Αναγνώρισης Προσώπου

Στην ενότητα αυτή παρουσιάζεται ο κώδικας που αναπτύχθηκε για την ανίχνευση και αναγνώριση προσώπου. Το σύστημα ανίχνευσης και αναγνώρισης προσώπου αναπτύχθηκε σε τρία στάδια:

1. Δημιουργία βάσης δεδομένων προσώπων
2. Προετοιμασία Βάσης Για Την Εκπαίδευση του LBP Classifier
3. Εκπαίδευση Και Ανίχνευση – Αναγνώριση σε Πραγματικό Χρόνο.

5.2.1. Δημιουργία Βάσης Δεδομένων Προσώπων

Η βάση δεδομένων προσώπων παρέχεται έτοιμη διαδικτυακά όπως φαίνεται στην συνέχεια <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>. Σε αυτή προστέθηκαν τα πρόσωπα των συγγραφέων της εργασίας. Η βάση δεδομένων περιέχει φακέλους με ονομασία «s1» έως «s45». Οι σαράντα τρεις φακέλοι περιέχουν εικόνες άγνωστων προσώπων. Η χρησιμότητά τους θα εξηγηθεί στη συνέχεια. Οι φάκελοι «s41» και «s44» περιέχουν εικόνες των προσώπων των συγγραφέων, οι οποίοι θα είναι τα πρόσωπα προς αναγνώριση. Οι παρακάτω εικόνες (Εικόνα 5.5.) δείχνουν τι περιέχει η βάση δεδομένων.



Εικόνα 5.5. Βάση Δεδομένων

Όσον αφορά τον κώδικα για την βάση δεδομένων έχει ως σκοπό τη λήψη φωτογραφιών των προσώπων που θέλουν να καταχωρηθούν στην βάση και την δημιουργία του αντίστοιχου φακέλου τους καθώς και τον έλεγχο ύπαρξης φακέλου με το ίδιο όνομα. Το όνομα του φακέλου είναι ένας ακεραίος αριθμός που αντιστοιχεί στην ετικέτα του προσώπου (id) και χρησιμεύει όπως θα δούμε στην συνέχεια στην αναγνώριση του προσώπου. Σε περίπτωση εύρεσης φακέλου με την ίδια ονομασία υπάρχει σφάλμα και ο χρήστης πρέπει να εισάγει άλλον αριθμό.

Για τον κώδικα χρησιμοποιήθηκαν οι εξής βασικές εντολές της OpenCV και της Python:

1. `os.chdir(path)`: Αλλάζει τον τρέχοντα κατάλογο εργασίας στη δεδομένη διαδρομή.
 - `path`: Η δεδομένη διαδρομή.
2. `os.makedirs(name)`: Δημιουργεί κατάλογο (φάκελο). Εάν ήδη υπάρχει φάκελος με τον ίδιον αριθμό προκαλείται σφάλμα.
 - `name`: Το όνομα του φακέλου.
3. `face_cascade=cv2.CascadeClassifier (path)`: Συνάρτηση της OpenCV από την οποία καλείται ο ταξινομητής Haar.
 - `path`: Το όνομα του αρχείου από το οποίο έχει τοποθετηθεί ο ταξινομητής Haar.
4. `gray=cv2.cvtColor (src,color)`: Συνάρτηση της OpenCV που μετατρέπει την εικόνα σε κλίμακα χρώματος.
 - `src`: Εικόνα πραγματικού χρόνου.
 - `color`: Το χρώμα που επιθυμούμε να έχουμε στην εικόνα.
5. `faces=face_cascade.detectMultiScale(src)`: Συνάρτηση της OpenCV που χρησιμοποιείται για την ανίχνευση προσώπων σε πραγματικό χρόνο και διαφορετικό μέγεθος.
 - `src`: Εικόνα από την οποία ανιχνεύονται τα αντικείμενα (πρόσωπα). Πρέπει να είναι σε κλίμακα του γκρι.

Ο κώδικας με βάση με τον οποίο δημιουργήθηκε η βάση δεδομένων είναι ο εξής:

```
# -*- coding: utf-8 -*-  
#Υπάρχει Λόγω Των Ελληνικών Λέξεων Στο Πρόγραμμα  
#Δημιουργία Module  
from picamera.array import PiRGBArray  
from picamera import PiCamera  
import time  
import cv2  
import os
```



```

#Ρύθμιση Παραμέτρων Camera Pi
camera = PiCamera()
camera.resolution = (320, 240 )
camera.framerate = 30
rawCapture = PiRGBArray( camera, size=( 320, 240 ) )

#Δημιουργία Αντικειμένου Ανίχνευσης Και Αρίθμηση Φακέλου (id)
face_cascade = cv2.CascadeClassifier( '/home/pi/opencv-
3.3.0/data/haarcascades/haarcascade_frontalface_alt2.xml' )

fps=0
folder=True

#Βρόχος Ελέγχου Για Την Ύπαρξη Του Ίδιου Φακέλου
while folder==True:
    id=raw_input('ΔΩΣΤΕ id')

    #Διαδρομή Βάσης Δεδομένων
    os.chdir('/home/pi/Desktop/database/training-data')

    #Ελεγχος Για Ύπαρξη Του Ίδιου Φακέλου (Σφάλμα) Αλλιώς Δημιουργία Με Το id Που Έχουμε
    Δώσει
    try:
        os.makedirs('s'+str(id))
        break
    except OSError:
        print 'ΥΠΑΡΧΕΙ Ο ΦΑΚΕΛΟΣ---ΠΑΡΑΚΑΛΩ ΔΩΣΤΕ ΑΛΛΟ ΝΟΥΜΕΡΟ'

#Καταχώριση Φακέλου Στην Βάση Δεδομένων
snum=0
os.chdir('/home/pi/Desktop/database/training-data/s'+str(id))

#Βρόχος Για Την Ανίχνευση Και Καταγραφή Των Προσώπων Στην Βάση
for frame in camera.capture_continuous( rawCapture, format="bgr", use_video_port=True ):

    #Δημιουργία Πλαισίου Και Μετατροπή Εικόνας Σε Κλίμακα Του Γκρι
    image = frame.array
    gray = cv2.cvtColor( image, cv2.COLOR_BGR2GRAY )

    #Κάλεσμα Συνάρτησης Ανίχνευσης Προσώπου
    faces = face_cascade.detectMultiScale( gray )

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

        #Καταγραφή Προσώπου Στον Φάκελο Με Αριθμό id Και Ονομασία Εικόνας Τον Αριθμό
        snum
        snum=snum+1
        cv2.imwrite('/home/pi/Desktop/database/training-
        data/s'+str(id)+'/'+str(snum)+'.jpg',gray[y:y+h,x:x+w])

        #Κατασκευή Πλαισίου Για Το Πρόσωπο
        cv2.rectangle(image,(x,y),(x+w,y+h),(255,255,0),2)
        cv2.waitKey(100)
        fps=fps+1

```

```

#Εμφάνιση Αποτελεσμάτων
cv2.imshow( "Frame", image )
cv2.waitKey(1)

#Χρόνος Καθυστέρησης Ενός Δευτερολέπτου Για Την Λήψη Της Φωτογραφίας Και Εμφάνιση
Του Αριθμού Φωτογραφίας
time.sleep(1)
print snum

#Διακοπή Προγράμματος Σε Εβδομήντα Φωτογραφίες (snum)
if (snum>70):
    break

#Κλείσιμο Camera Pi
rawCapture.truncate( 0 )

camera.stop_preview()
camera.close()

```

5.2.2. Προετοιμασία Βάσης Για Την Εκπαίδευση του LBP Classifier

Η ανίχνευση των προσώπων επιτυγχάνεται με τους ήδη εκπαιδευμένους ταξινομητές Cascade της OpenCV, διαδικασία η οποία παρουσιάστηκε και στη βιβλιογραφική ανασκόπηση της εργασίας. Για την αναγνώριση των προσώπων, θα πρέπει να δημιουργηθούν οι κατάλληλες συνθήκες έτσι ώστε να επιτευχθεί η εκπαίδευση του συστήματος και κατ' επέκταση η σωστή αναγνώριση προσώπου. Αυτό τον στόχο εξυπηρετεί ο κώδικας αυτής της ενότητας. Ο κώδικας δημιουργεί δύο κενές λίστες που θα περιέχουν, μετά την επεξεργασία των δεδομένων, τα πρόσωπα με τις αντίστοιχες ετικέτες (id). Αφού ορισθούν οι λίστες αυτές καταχωρούνται σε μια νέα λίστα όλες οι διαδρομές που αφορούν την βάση δεδομένων. Για κάθε διαδρομή επεξεργάζονται οι φακέλοι που ξεκινούν με το γράμμα «s» (αυτοί περιέχουν τα πρόσωπα) και εντός αυτών γίνεται μετατροπή κάθε εικόνας προσώπου σε σειρά gray (διάνυσμα- λίστα σε πολυάριθμες συστοιχίες) για να μπορούν να επεξεργαστούν στον κώδικα αναγνώρισης και στην συνέχεια προστίθενται στη λίστα προσώπων. Τέλος προστίθεται και στην λίστα των ετικετών (id) η ετικέτα που αντιστοιχεί στο πρόσωπο που έχει επεξεργαστεί εντός του φακέλου όπου υπήρχε, δηλαδή τον αριθμό του φακέλου.

Οι βασικότερες εντολές που χρησιμοποιήθηκαν από τις βιβλιοθήκες της Python και της OpenCV είναι οι εξής :

1. `img_way.startswith (str)`: Ελέγχει εάν η συμβολοσειρά ξεκινά με `str`.
 - `str`: Συμβολοσειρά ελέγχου.

2. `os.listdir (path)`: Επιστρέφει μια λίστα που περιέχει τα ονόματα των καταχωρήσεων στον κατάλογο που δίνεται στην διαδρομή.
 - `path`: Η διαδρομή που δίνεται.
3. `img_way.replace (old,new)`: Επιστρέφει ένα αντίγραφο της συμβολοσειράς στην οποία έχουν αντικατασταθεί τα παλαιά περιστατικά με τα νέα.
 - `old`: Παλαιό περιστατικό προς αντικατάσταση.
 - `new`: Η νέα περίπτωση που θα αντικαταστήσει την παλαιά.
4. `imgNr=np.array (src , data type)`: Μετατρέπει την εικόνα εισόδου σε σειρά array (λίστες σε πολυάριθμες συστοιχίες). Είναι απαραίτητη για τη λειτουργία του αναγνωριστή.
 - `src`: Εικόνα Εισόδου.
 - `data type`: Αριθμητικός τύπος με όριο ακέραιους αριθμούς από 0 έως 255.

Όσον αφορά την ανίχνευση των προσώπων στις εικόνες των φακέλων, χρησιμοποιήθηκαν οι ίδιες συναρτήσεις όπως στην βάση δεδομένων.

Ο κώδικας με βάση τον οποίο έγινε η προετοιμασία και η εκπαίδευση του συστήματος είναι ο εξής:

```
# -*- coding: utf-8 -*-
#Υπάρχει Λόγω Των Ελληνικών Λέξεων Στο Πρόγραμμα
#Ορισμός Module
import cv2,os
import numpy as np

#Δημιουργία Αντικειμένου Ανίχνευσης Μέσω Της OpenCV
haar=cv2.CascadeClassifier('/home/pi/opencv3.3.0/data/haarcascades/haarcascade_frontalface_alt2.xml')

#Συνάρτηση Εκπαίδευσης
def training(way):

    #Δημιουργία Κενών Λιστών Προσώπων Και Ετικετών
    face_list=[]
    Id_list=[]

    #Δημιουργία Λίστας Από Τις Διαδρομές Όλων Των Αρχείων Που Υπάρχουν Στον Φάκελο
    folder_list= os.listdir(way)

    #Βρόχος Επανάληψης Όλων Των Διαδρομών Παίρνοντας Το id Και Εικόνα
    for img_way in folder_list:

        #Έλεγχος Φακέλου Εάν Ξεκινά Με Το Γράμμα s
        if not img_way.startswith("s"):
            continue;
```

```

#Αφαίρεση Του Γράμματος s Και Αντικατάσταση Με Το Κενό ' ', Καταχώριση Αριθμού
Φακέλου Στην Μεταβλητή ident
ident = int(img_way.replace("s", ""))

#Προσθήκη Στην Διαδρομή Τον Φάκελο Που Ξεκινά Από s
person_way = way + "/" + img_way
img_name = os.listdir(person_way)

#Βρόχος Επανάληψης Που Αφορά Τις Εικόνες Προσώπων Στον Φάκελο
for name in img_name:

    #Ελεγχος Εάν Η Εικόνα Ξεκινά Με Τελεία
    if name.startswith("."):
        continue;

    #Διαδρομή Κάθε Εικόνας Και Διάβασμα Στο Φάκελο Που Έχει Ήδη Βρεθεί Και Μετατροπή
    Σε Κλίμακα Του Γκρι
    img_dir = person_way + "/" + name
    img = cv2.imread(img_dir)
    gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

    #Μετατροπή Εικόνας Σε Σειρά Array
    imgNr=np.array(gray,'uint8')

    #Ανίχνευση Προσώπου Εικόνας
    faces=haar.detectMultiScale(imgNr)

    #Προσθήκη Προσώπων (Σε Σειρά Array) Και Ετικετών id Στις Αντίστοιχες Λίστες Μετά Την
    Ανίχνευση
    for (x,y,w,h) in faces:
        face_list.append(imgNr[y:y+h,x:x+w])
        Id_list.append(ident)

#Επιστροφή Των Λιστών
return face_list,Id_list

#Κάλεσμα της Συνάρτησης Και Καταχώριση Των Λιστών Στα face, Id
face,Id = training('/home/pi/Desktop/database/training-data')

```

5.2.3. Εκπαίδευση Και Ανίχνευση – Αναγνώριση Προσώπου σε Πραγματικό Χρόνο

Εφόσον έχει προετοιμαστεί η βάση και έχουν γίνει οι κατάλληλες ενέργειες για την εκπαίδευση φτάνουμε στο τελικό στάδιο, αυτό της ανίχνευσης και αναγνώρισης σε πραγματικό χρόνο. Για να πραγματοποιηθεί αυτό, στον κώδικα δημιουργούμε τον αναγνωριστή (recognizer) και στην συνέχεια τον εκπαιδεύουμε με τα πρόσωπα που είναι καταχωρημένα στη βάση και τις αντίστοιχες ετικέτες (id). Ύστερα αφού ανιχνευθεί πρόσωπο εντός του παραθύρου πραγματοποιούμε με την βοήθεια του αναγνωριστή μια πρόβλεψη. Κατά την πρόβλεψη παίρνουμε την αξιοπιστία του αποτελέσματος (conf) και την ετικέτα (id)

του προσώπου. Με βάση την αξιοπιστία δημιουργούμε μια ζώνη αναγνώρισης, έτσι ώστε η αναγνώριση να γίνεται εντός αυτής για την επίτευξη σωστού αποτελέσματος. Διαφορετικά γίνεται μόνο ανίχνευση και ζητείται από το προς αναγνώριση άτομο να τοποθετηθεί εντός της ζώνης αναγνώρισης. Τέλος εντός της ζώνης γίνεται και έλεγχος των ετικετών (id) και εάν υπάρχει ταύτιση με αυτών της βάσης εμφανίζεται εάν το πρόσωπο είναι γνωστό ή όχι.

Εδώ χρησιμοποιούνται τα πολλά πρόσωπα που υπάρχουν στην βάση για τον ρόλο του αγνώστου. Αυτά μας βοηθούν ώστε να έχουμε διαφορετική ετικέτα (id) κάθε φορά που εισέρχεται ένα άγνωστο πρόσωπο εντός της ζώνης αναγνώρισης.

Για τον κώδικα χρησιμοποιήσαμε τις εξής συναρτήσεις της OpenCV και της Python:

Αρχικά καλούμε τον ανιχνευτή Haar.

Στην συνέχεια έχουμε τις εντολές που ακολουθούν.

1. `rec=cv2.face.LBPHFaceRecognizer_create (radius, neighbors, grid_x, grid_y):`
Δημιουργεί τον αναγνωριστή τοπικού δυαδικού μοτίβου.
 - `radius`: Η ακτίνα κατασκευής του κυκλικού δυαδικού μοτίβου γύρω από το κεντρικό εικονοστοιχείο.
 - `neighbors`: Αριθμός των σημείων δειγματοληψίας για την κατασκευή του κυκλικού τοπικού δυαδικού μοτίβου. Όσο μεγαλύτερος είναι ο αριθμός τόσο μεγάλο το υπολογιστικό κόστος.
 - `grid_x`: Αριθμός κελιών οριζόντιας κατεύθυνσης. Όσο περισσότερα κύτταρα υπάρχουν τόσο λεπτότερο το πλέγμα και αντίστοιχα τόσο μεγαλύτερο το πλήθος των διαστάσεων του διανύσματος των χαρακτηριστικών που προκύπτουν.
 - `grid_y`: Αριθμός κελιών κάθετης κατεύθυνσης. Όσο περισσότερα κύτταρα υπάρχουν τόσο λεπτότερο το πλέγμα και αντίστοιχα τόσο μεγαλύτερο το πλήθος των διαστάσεων του διανύσματος των χαρακτηριστικών που προκύπτουν.
2. `rec.train (src, labels):` Εκπαιδεύει έναν αναγνωριστή LBP με δεδομένα εικόνων και σχετικών ετικετών (id).
 - `src`: Εικόνες εκπαίδευσης, δηλαδή τα πρόσωπα που θέλουμε να μάθει. Τα δεδομένα πρέπει να δίνονται σαν διάνυσμα (σειρά array-πολυάριθμες συστοιχίες).
 - `labels`: Οι ετικέτες (id) που αντιστοιχούν στις εικόνες. Πρέπει να είναι ακέραιος αριθμός.
3. `faces=haar.detectMultiScale (src, scaleFactor, minNeighbors):` Ανιχνεύει αντικείμενα (πρόσωπα) στην εικόνα εισόδου.

- src: Εικόνα για ανίχνευση προσώπου.
 - scaleFactor: Παράμετρος που καθορίζει πόσο μειώνεται το μέγεθος της εικόνας σε κάθε κλίμακα εικόνας.
 - minNeighbors: Παράμετρος που καθορίζει τον αριθμό των γειτόνων που κάθε υποψήφιο ορθογώνιο θα πρέπει να διατηρήσει.
4. rec.predict (src): Προβλέπει μια ετικέτα (id) και σχετική εμπιστοσύνη για μια δεδομένη εικόνα εισόδου.
- src: Δείγμα εικόνας για να πάρουμε μια πρόβλεψη.

Ο κώδικας εκπαίδευσης - ανίχνευσης και αναγνώρισης είναι ο εξής:

```
# -*- coding: utf-8 -*-
#Υπάρχει Λόγω Των Ελληνικών Λέξεων Στο Πρόγραμμα
# Ορισμός Module
from picamera.array import PiRGBArray
from picamera import PiCamera
import time
import cv2
import cv2.face
import train_final

#Ρύθμιση Παραμέτρων Camera Pi
camera = PiCamera()
camera.resolution = (320, 240 )
camera.framerate = 30
rawCapture = PiRGBArray( camera, size=( 320, 240) )

#Ορισμός Αντικειμένων Ανίχνευσης Και Αναγνώρισης Μέσω Της OpenCV
haar = cv2.CascadeClassifier( '/home/pi/opencv-
3.3.0/data/haarcascades/haarcascade_frontalface_alt2.xml' )
rec=cv2.face.LBPHFaceRecognizer_create(3,9,1,1)

#Εκπαίδευση Του Αναγνωριστή Για Την Αναγνώριση Προσώπου Από Την Βάση Δεδομένων
(Training) Μέσω Της OpenCV
rec.train(train_final.face,train_final.np.array(train_final.Id))

#Βρόχος Συνεχούς Λειτουργίας Της Camera
for frame in camera.capture_continuous( rawCapture, format="bgr", use_video_port=True ):

    #Ορισμός Πλαισίου Εικόνας Και Μετατροπή Σε Κλίμακα Του Γκρι
    image = frame.array
    gray = cv2.cvtColor( image, cv2.COLOR_BGR2GRAY )

    #Συνάρτηση Ανίχνευσης Προσώπων Σε Πραγματικό Χρόνο Μέσω Της OpenCV
    faces = haar.detectMultiScale( gray,1.2,6)

    #Βρόχος Ελέγχου Για Την Ανίχνευση Και Εξαγωγή Χαρακτηριστικών Προσώπου
    for (x,y,w,h) in faces:

        #Δημιουργία Τετραγώνου (Πλαίσιο) Στο Ανιχνευμένο Πρόσωπο
```

```

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

#Σύγκριση Διανυσμάτων Εικόνας Από Την Βάση Με Πρόσωπο Πραγματικού Χρόνου Και
Επιστροφή Της Ετικέτας Και Αξιοπιστίας Του
id,conf=rec.predict(gray[y:y+h,x:x+w])

#Δημιουργία Ζώνης Αναγνώρισης Βάση Της Αξιοπιστίας Και Εμφάνιση Αποτελεσμάτων
Βάση Της Ετικέτας
if conf<0.2 :
    print 'ΕΚΤΟΣ ΖΩΝΗΣ ΑΝΑΓΝΩΡΙΣΗΣ---ΠΑΡΑΚΑΛΩ ΠΙΟ ΠΙΣΩ'
    id=""
elif conf>0.4:
    print 'ΕΚΤΟΣ ΖΩΝΗΣ ΑΝΑΓΝΩΡΙΣΗΣ---ΠΑΡΑΚΑΛΩ ΠΙΟ ΜΠΡΟΣΤΑ'
    id=""
elif (id==41) :

    id='PANOS'
    print 'ΓΝΩΣΤΟ ΠΡΟΣΩΠΟ --> ΠΑΝΟΣ'

elif (id==44) :

    id='SPYROS'
    print 'ΓΝΩΣΤΟ ΠΡΟΣΩΠΟ --> ΣΠΥΡΟΣ'

else:

    id='UNKNOWN'
    print 'ΑΓΝΩΣΤΟ ΠΡΟΣΩΠΟ ΣΤΗ ΖΩΝΗ ΑΝΑΓΝΩΡΙΣΗΣ'

#Εμφάνιση Αποτελεσμάτων Σε Πραγματικό Χρόνο
cv2.putText(image,str(id) ,(x,(y+h)+15),cv2.FONT_HERSHEY_SIMPLEX,(0.7),(0,255,255))

cv2.imshow( "Frame", image )
key=cv2.waitKey(1) & 0xFF

#Τερματισμός Προγράμματος Με Την Πληκτρολόγηση Του Γράμματος q,Κλείσιμο Όλων Των
Παραθύρων
if key == ord('q'):
    break

rawCapture.truncate( 0 )

camera.stop_preview()
camera.close()
cv2.destroyAllWindows()

```

5.3. Κώδικας Ανίχνευσης Και Ελέγχου Στάθμης Υγρού Σε Μπουκάλι

Στην ενότητα αυτή παρουσιάζεται ο κώδικας που αναπτύχθηκε για την ανίχνευση και τον έλεγχο της στάθμης υγρού (στην παρούσα εργασία, τσάι) σε γυάλινο μπουκάλι 750 ml (η πραγματική χωρητικότητα χωρίς τον λαιμό του μπουκαλιού ήταν 722 ml) . Για την ανίχνευση χρησιμοποιήθηκαν οι εξής συναρτήσεις της OpenCV:

1. `gauss=cv2.GaussianBlur(gray,(sigmaX, sigmaY), borderType)`
 - `gray`: η εικόνα εισόδου, σε κλίμακα του γκρι, στην οποία θα εφαρμοστεί το φίλτρο Gaussian Blur
 - `sigmaX – sigma Y`: καθορίζουν την τυπική απόκλιση των αξόνων X και Y και την ευαισθησία στον θόρυβο για κάθε άξονα. Πρέπει να είναι περιττοί αριθμοί.
 - `borderType`: το μέγεθος προβολής των εικονοστοιχείων
2. `edges1=cv2.Canny(image, double threshold1, double threshold2, int apertureSize = 3)`
 - `image`: η εικόνα εισόδου σε κλίμακα του γκρι, στην οποία έχει ήδη εφαρμοστεί το φίλτρο
 - `double threshold1`: η χαμηλή τιμή κατωφλίου για τη διαδικασία υστέρησης κατωφλίων
 - `double threshold2`: η υψηλή τιμή κατωφλίου για τη διαδικασία υστέρησης κατωφλίων
 - `int apertureSize = 3`: το μέγεθος του πυρήνα Sobel που χρησιμοποιείται για να βρει την κλίση της εικόνας
3. `lines = cv2.HoughLinesP(dst, rho, theta, threshold, minLinLength, maxLineGap)`
 - `dst`: η εικόνα εξόδου από τον ανιχνευτή ακρών Canny
 - `rho`: η ανάλυση της παραμέτρου r σε pixel. Χρησιμοποιούμε 1 pixel
 - `theta`: η ανάλυση της παραμέτρου θ σε ακτίνια
 - `threshold`: ο ελάχιστος αριθμός διασταυρώσεων για να ανιχνευθεί μία γραμμή
 - `minLinLenght`: ο ελάχιστος αριθμός σημείων που μπορούν να σχηματίσουν μια γραμμή (οι υπόλοιπες γραμμές απορρίπτονται)
 - `maxLineGap`: το μέγιστο κενό μεταξύ δύο σημείων ώστε να θεωρηθεί ότι ανήκουν στην ίδια γραμμή
4. `cv2.line(img, pt1, pt2, color, thickness)`
 - `img`: η εικόνα στην οποία θα σχεδιαστεί η γραμμή
 - `pt1 – pt2`: το πρώτο και τελευταίο σημείο που αποτελούν τη γραμμή
 - `color`: το χρώμα της γραμμής που θα σχεδιαστεί

- thickness: το πάχος της γραμμής που θα σχεδιαστεί
5. cv2.rectangle (src , pt1, pt2, color, thickness)
- src: Εικόνα
 - pt1: Η κορυφή του ορθογωνίου
 - pt2: Η κορυφή του ορθογωνίου απέναντι από το pt1
 - color: Χρώμα ορθογωνίου (εικόνα σε κλίμακα του γκρι)
 - thickness: Πάχος γραμμών που σχεδιάζουν το ορθογώνιο

Όσον αφορά τον υπολογισμό της στάθμης του υγρού στο μπουκάλι, επιτυγχάνεται μέσω απλών πράξεων μέτρησης των pixel της εικόνας. Αρχικά, καθορίστηκε ένα ορθογώνιο παράθυρο μέσα στο οποίο πραγματοποιείται η ανίχνευση και ο έλεγχος της στάθμης (δηλαδή, η Περιοχή Ενδιαφέροντος). Επίσης, στον πάτο του μπουκαλιού σχηματίσαμε μια μαύρη γραμμή, η οποία χρησιμεύει ως σημείο αναφοράς για το άδειο μπουκάλι.

Στη συνέχεια βρήκαμε πως το ορθογώνιο παράθυρο καθορίζεται από τα σημεία (270,90) και (350,373). Συγκεκριμένα στον άξονα των y, ο οποίος μας ενδιαφέρει, υπολογίσαμε πως μέσα στο ορθογώνιο παράθυρο περιέχονται 283 pixel (κάνοντας την αφαίρεση των δύο κορυφών: 373-90). Όμως, και η μαύρη γραμμή που είχαμε σχηματίσει ως σημείο αναφοράς για το άδειο μπουκάλι υπολογίσαμε πως καλύπτει 2 pixel του ορθογωνίου παραθύρου. Άρα, τα συνολικά pixel που φαίνεται ότι μπορεί να καταλαμβάνει το υγρό είναι 281.

Επομένως, γνωρίζουμε ότι το μπουκάλι είναι πλήρως γεμάτο στα 281 pixel. Επίσης, έχοντας μετρήσει με δοσομετρητή την ποσότητα υγρού στην οποία το μπουκάλι είναι πλήρως γεμάτο βρήκαμε πως αυτή είναι 720 ml. Με βάση αυτά τα δύο στοιχεία, μπορούμε να υπολογίσουμε τα ml που περιέχονται σε ένα μόνο pixel με την απλή μέθοδο των τριών:

Pixel	ml	
281	720	
1	x	(5.1)

$$x = 720 / 281 = 2,5622 \text{ ml}$$

Άρα, σε κάθε pixel περιέχονται 2,5622 ml υγρού.

Όμως, επειδή έχουμε παραβλέψει 2 pixel λόγω της μαύρης γραμμής ως σημείου αναφοράς, στον τελικό τύπο πρέπει να τα προσθέσουμε και αυτά.

$$2 \text{ pixel} * 2,5622 \text{ ml} = 5,1244 \text{ ml} \quad (5.2)$$

Ο τελικός τύπος προκύπτει από την αφαίρεση του κορυφαίου σημείου του ορθογωνίου παραθύρου (281 pixel) από τα pixel των σημείων της γραμμής (x1[1]) της στάθμης του υγρού, που υπολογίζεται από το πρόγραμμα και είναι ο εξής:

$$ml = ((281-(x1[1])) * 2,5622) + 5,1244 \quad (5.3)$$

Ο τελικός κώδικας ανίχνευσης και ελέγχου της στάθμης υγρού είναι ο εξής:

```
# -*- coding: utf-8 -*-
#Υπάρχει Λόγω Των Ελληνικών Λέξεων Στο Πρόγραμμα
#Εισαγωγή Βιβλιοθηκών
import cv2
import math
from picamera import PiCamera
from picamera.array import PiRGBArray

#Ρύθμιση Παραμέτρων Camera Pi
camera = PiCamera()
camera.resolution = ( 640, 480 )
camera.framerate = 30
rawCapture = PiRGBArray( camera, size=( 640, 480 ) )

#Δημιουργία Συνάρτησης Για Την Συνεχή Λειτουργία Της Κάμερας Και Του Προγράμματος
def liquid_level():
    for frame in camera.capture_continuous( rawCapture, format="bgr", use_video_port=True ):

        #Δημιουργία Πλαισίου Εικόνας Και Μετατροπή Σε Κλίμακα Του Γκρι
        image = frame.array
        gray = cv2.cvtColor( image, cv2.COLOR_BGR2LAB )

        #Κατασκευή Ορθογωνίου Στο Οποίο Τοποθετείται Το Μπουκάλι Για Την Επίτευξη Της
        #Ανίχνευσης
        cv2.rectangle(image,(270,90),(350,373),(255,255,0),2)

        #Εφαρμογή Φίλτρου Gaussianblur Στην Εικόνα Κλίμακας Γκρι Και Του Ανιχνευτή Ακρών Canny
        gauss=cv2.GaussianBlur(gray,(105,5),0)
        edges1=cv2.Canny(gauss[90:373,290:325], 50, 80,apertureSize=3)

        #Ανίχνευση Σημείων Ευθείας Γραμμής Που Αντιστοιχεί Στην Στάθμη Του Υγρού
        lines = cv2.HoughLinesP(edges1, 1, math.pi/2, 2, None, 20, 30)

        if lines is not None:

            #Ανάθεση Τιμών Στα x1,x2 Που Αντιστοιχούν Στο Αρχικό Και Τελικό Σημείο
            #Γραμμής Για Τον Σχεδιασμό Της
            for i in lines[0]:
                x1 = (i[0],i[1])
```

```

x2 = (i[2],i[3])

# Υπολογισμός ml
ml=((281-(x1[1]))*2.5622)+5.1244

#Ελεγχος Στάθμης Για Άδειο Μπουκάλι
if x1[1]>=280:
    print 'LEVEL---EMPTY'
    cv2.putText(image,'empty',(280,230),cv2.FONT_HERSHEY_SIMPLEX,(0.7),(0,0,0))
else:

    #Σχεδιασμός Γραμμής Στάθμης Υγρού Και Εμφάνιση Γραμμής Και ml Στην Εικόνα Σε
    #Πραγματικό Χρόνο
    cv2.line(edges1,x1,x2,(255,0,0),2)
    cv2.line(image,(x1[0]+275,x1[1]+90),(x2[0]+310,x2[1]+90),(255,0,0),2)
    cv2.putText(image,str((int(ml),'ml')),(x2[0]+165),(x2[1]+100),cv2.FONT_
HERSHEY_SIMPLEX,(0.4),(0,255,255))

#Ελεγχός Στάθμης Υγρού Στο Μπουκάλι
if 50<x1[1]<235:
    print 'LEVEL---NORMAL'
elif 7<=x1[1]<=50:
    print 'LEVEL---ALMOST FULL'
elif x1[1]<7:
    print 'LEVEL---FULL-STOP'
elif 235<=x1[1]<=265:
    print 'LEVEL---ALMOST LOW'
elif 265<x1[1]<281:
    print 'LEVEL---LOW'

#Εμφάνιση Πλαισίων
cv2.imshow('Video', image)
cv2.imshow('Video1',edges1)

#Τερματισμός Προγράμματος Με Την Πληκτρολόγηση Του Q
key=cv2.waitKey(1) & 0xFF

if key == ord('q'):
    break
    rawCapture.truncate(0)
    camera.stop_preview()
    camera.close()
    cv2.destroyAllWindows()

#Κάλεσμα Συνάρτησης
liquid_level()

```

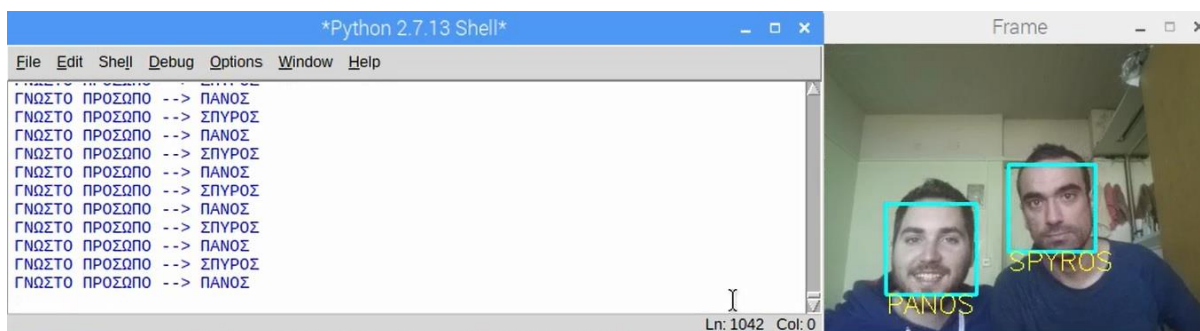
Κεφάλαιο 6: Αποτελέσματα - Συμπεράσματα

6.1. Αποτελέσματα – Συμπεράσματα Για Την Ανίχνευση Και Αναγνώριση Προσώπου

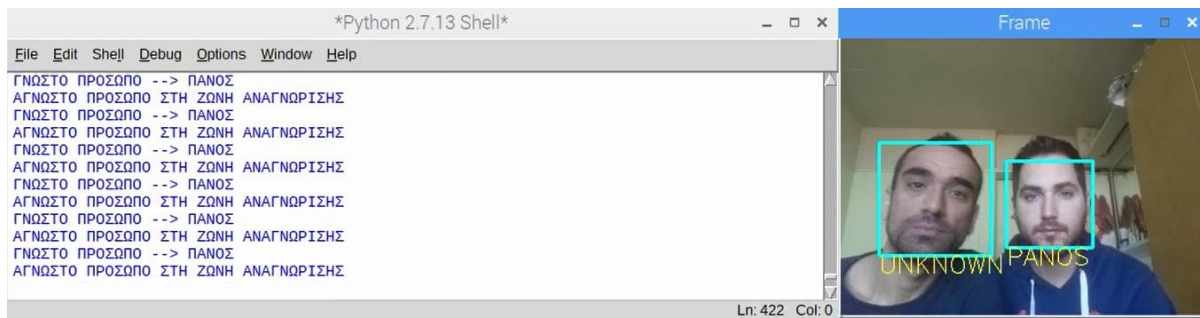
6.1.1 Αποτελέσματα

Αρχικά, στην ενότητα αυτή θα παρουσιαστούν τα αποτελέσματα από την εφαρμογή του κώδικα της ενότητας 5.2. Παρακάτω παρατίθενται φωτογραφίες από τις διαφορετικές περιπτώσεις ανίχνευσης και αναγνώρισης προσώπων των συγγραφέων . Στο αριστερό μέρος της εικόνας τρέχει το πρόγραμμα τυπώνοντας τα εξής μηνύματα: Α) της αναγνώρισης ή μη των προσώπων που έχουν καταχωρηθεί στην βάση (γνωστό ή άγνωστο πρόσωπο) Β) της ένταξης των προσώπων στην ζώνη αναγνώρισης και της ενημέρωσής τους για τις κατάλληλες ενέργειες που πρέπει να γίνουν (μετακίνηση πιο πίσω ή πιο μπροστά). Στο δεξί μέρος της εικόνας φαίνονται τα ανιχνευμένα πρόσωπα εντός των ορθογωνίων πλαισίων και κάτω από αυτά ο χαρακτηρισμός της αναγνώρισης που έχει επιτευχθεί σε πραγματικό χρόνο. Συγκεκριμένα είχαν οριστεί οι εξής συνθήκες ανίχνευσης - αναγνώρισης:

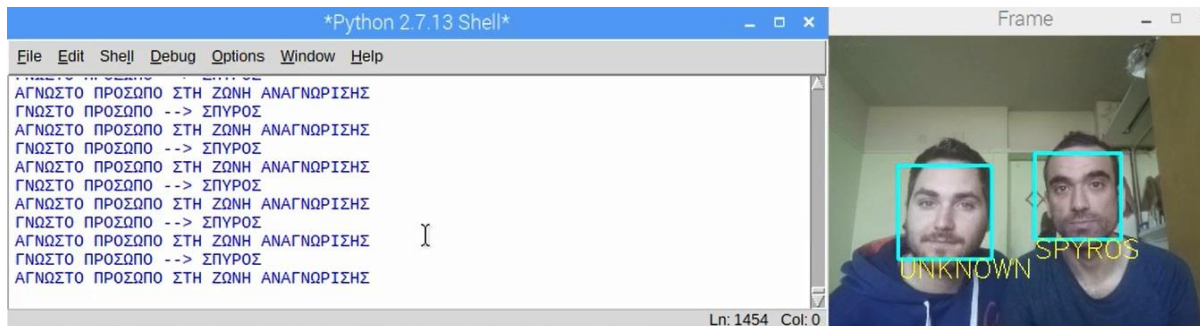
1. Οι δύο συγγραφείς ως γνωστά πρόσωπα (PANOS,SPYROS)
2. Ο ένας συγγραφέας (Σπύρος) ως άγνωστο πρόσωπο και ο άλλος συγγραφέας (Πάνος) ως γνωστό πρόσωπο (UNKNOWN,PANOS)
3. Ο ένας συγγραφέας (Πάνος) ως άγνωστο πρόσωπο και ο άλλος συγγραφέας (Σπύρος) ως γνωστό πρόσωπο (UNKNOWN,SPYROS)
4. Και οι δύο συγγραφείς ως άγνωστα πρόσωπα (UNKNOWN, UNKNOWN)
5. Μετακίνηση των συγγραφέων πιο μακριά ή πιο κοντά από τη ζώνη αναγνώρισης.



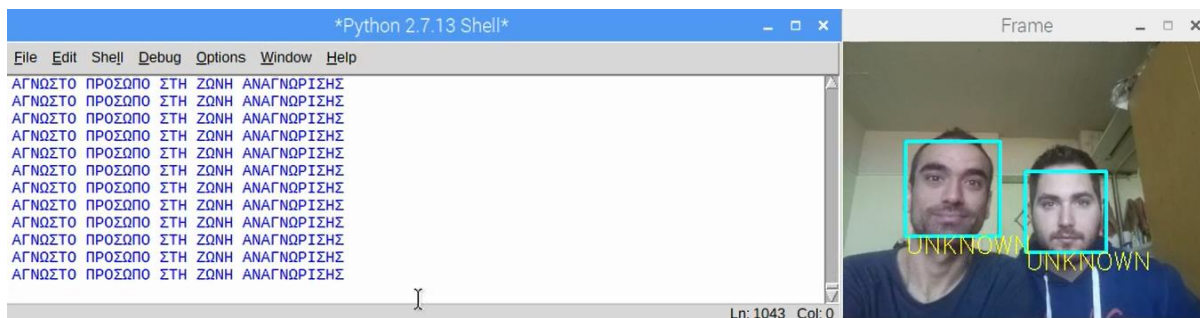
Εικόνα 6.1.Ανίχνευση-Αναγνώριση Και Με Τους Δύο Συγγραφείς Γνωστούς Στην Βάση



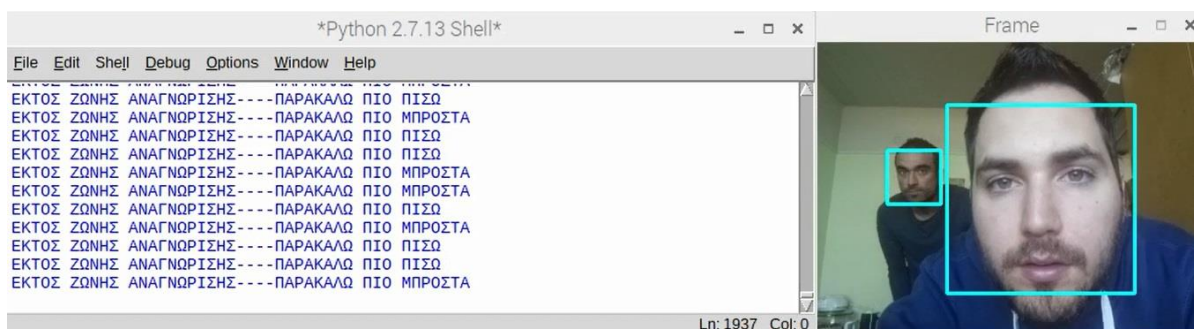
Εικόνα 6.2.Ανίχνευση-Αναγνώριση Με Γνωστό Πρόσωπο Στην Βάση Τον Ένα Συγγραφέα (Πάνο)



Εικόνα 6.3.Ανίχνευση-Αναγνώριση Με Γνωστό Πρόσωπο Στην Βάση Τον Ένα Συγγραφέα (Σπύρο)



Εικόνα 6.4.Ανίχνευση-Αναγνώριση Και Με Τους Δύο Συγγραφείς Ως Άγνωστα Πρόσωπα



Εικόνα 6.5.Ανίχνευση Αλλά Μη Αναγνώριση Εκτός Ζώνης Αναγνώρισης

6.1.2. Συμπεράσματα

Κατά την προσπάθεια κατασκευής του κώδικα και εφαρμογής του διαπιστώσαμε πως υπάρχουν συγκεκριμένες προϋποθέσεις επιτυχίας. Οι βασικότερες ήταν το μέγεθος της

ανάλυσης εικόνας της κάμερας που θα χρησιμοποιούσαμε σε πραγματικό χρόνο, ο αριθμός και το μέγεθος των εικόνων προς αναγνώριση που υπάρχουν στην βάση δεδομένων, η απόσταση των προσώπων από την κάμερα, οι τιμές των παραμέτρων στις συναρτήσεις της OpenCV που χρησιμοποιήσαμε καθώς και η τεράστιας σημασίας παράμετρος αυτή του φωτισμού.

Όσον αφορά το μέγεθος ανάλυσης της εικόνας της κάμερας παρατηρήθηκε πως όσο πιο μικρή είναι το πιο καλύτερα αποτελέσματα έχουμε στην αναγνώριση των προσώπων. Αυτό οφείλεται κυρίως στον φθινό εξοπλισμό που χρησιμοποιούμε κάνοντας το σύστημα να μην λειτουργεί σωστά τόσο στην αποδοτικότητα (μεγάλη ανάλυση – πιο αργό το πρόγραμμα) όσο και στην αναγνώριση των προσώπων (συχνότερα σφάλματα ανίχνευσης).

Επίσης, για τις εικόνες που έχουμε στην βάση δεδομένων θα πρέπει η ανάλυση τους να αντιστοιχεί με αυτήν της κάμερας που χρησιμοποιούμε στο σύστημα για καλύτερα αποτελέσματα και αξιοπιστία. Παρατηρήθηκε ότι εάν οι φωτογραφίες ληφθούν από κάμερα καλύτερης ανάλυσης προκύπτει ανικανότητα αναγνώρισης προσώπων που έχουν καταχωρηθεί στην βάση δεδομένων. Επιπλέον, ένας ακόμα λόγος για τον οποίο επιλέγουμε χαμηλή ανάλυση στην εικόνα της κάμερας είναι ότι οι έτοιμες φωτογραφίες που βρίσκονται στην βάση δεδομένων για τα άγνωστα πρόσωπα είναι εικόνες χαμηλής ανάλυσης. Σε περίπτωση επιλογής μεγάλης ανάλυσης στην εικόνα της κάμερας είναι αδύνατη η αναγνώριση των άγνωστων προσώπων καθώς γίνεται ταύτιση αυτών με κάποιο από τα γνωστά πρόσωπα της βάσης.

Ο αριθμός των εικόνων προσώπων που χρησιμοποιούνται για την αναγνώριση (συγγραφείς) θα πρέπει να είναι μεγάλος έτσι ώστε κατά την πειραματική φάση η ταύτιση τους με τις αντίστοιχες ετικέτες τους να είναι πιο σίγουρη. Επίσης, σημαντικό είναι ότι οι φωτογραφίες για την αναγνώριση θα πρέπει να ληφθούν κατά μέτωπο (και όχι προφίλ) διότι χρησιμοποιούμε αυτόν τον τύπο ανιχνευτή, αλλιώς υπάρχει ο κίνδυνος η ανίχνευση να είναι δύσκολο να επιτευχθεί ή εάν γίνεται τα αποτελέσματα που λαμβάνουμε να είναι λανθασμένα. Τέλος, μετά την λήψη των εικόνων αυτές θα πρέπει να μετατραπούν σε χρώμα κλίμακας του γκρι που είναι απαραίτητο στοιχείο για την λειτουργία του κώδικα καθώς επίσης και στην αλλαγή του μεγέθους των εικόνων σε μικρότερο μέγεθος για πιο γρήγορη εκπαίδευση του συστήματος.

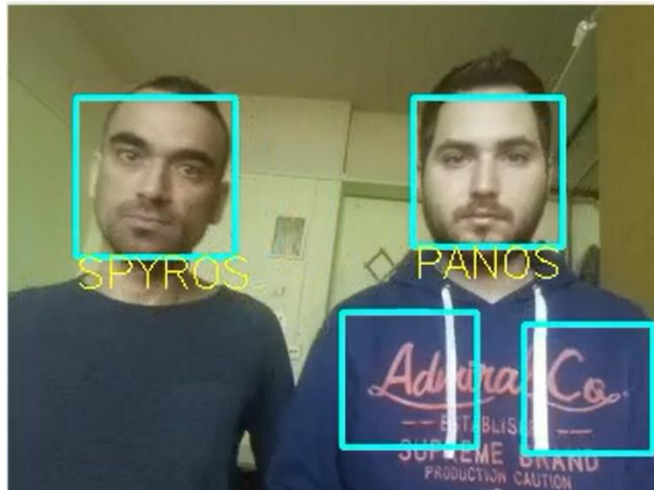
Ο φωτισμός παίζει καθοριστικό ρόλο για την σωστή λειτουργία του συστήματος. Αρχικά, καταλαβαίνουμε πως όσο καλύτερος είναι ο φωτισμός τόσο περισσότερο αξιόπιστα

αποτελέσματα λαμβάνουμε. Έτσι έχουμε πιο «καθαρές» φωτογραφίες στην λήψη και κατά την επεξεργασία τους εξάγουμε καλύτερα χαρακτηριστικά γνωρίσματα και κατ' επέκταση πιο ασφαλή συμπεράσματα. Επιπλέον, ο φωτισμός επηρεάζει και την αξιοπιστία αναγνώρισης (conf) που επιστρέφει το σύστημα, αφού λιγότερος φωτισμός σημαίνει μεγαλύτερο νούμερο αξιοπιστίας (conf) και άρα λιγότερο αξιόπιστο αποτέλεσμα, καθώς και αλλαγή στο εύρος λειτουργίας της ζώνης αναγνώρισης.

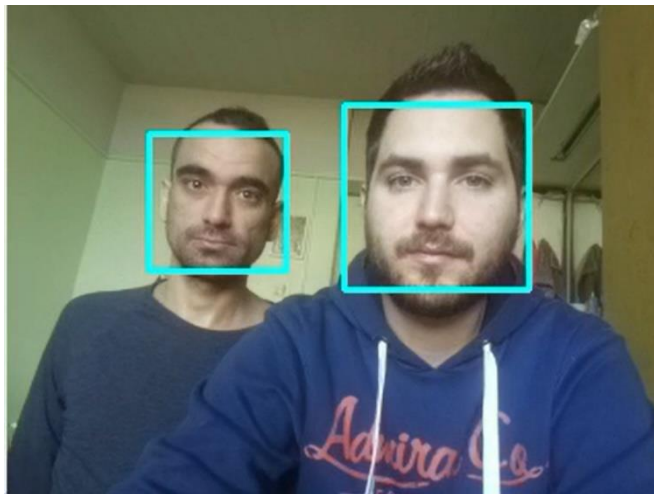
Για την επίτευξη καλύτερων αποτελεσμάτων και την αποφυγή σφαλμάτων εκμεταλλευτήκαμε την αξιοπιστία (conf) που επιστρέφεται από το σύστημα δημιουργώντας μια ζώνη αναγνώρισης. Εντός αυτής πραγματοποιείται η αναγνώριση των προσώπων συγκρίνοντας τις ετικέτες της πρόβλεψης με αυτών της βάσης δεδομένων και εμφανίζονται τα αντίστοιχα αποτελέσματα (γνωστός ή άγνωστος). Ακόμα ο καθορισμός του εύρους γίνεται από την αξιοπιστία που επιστρέφει το σύστημα από την πρόβλεψη του και αιτία της δημιουργίας της είναι ότι παρατηρήθηκε πως στην συγκεκριμένη απόσταση από την κάμερα έχουμε πιο αξιόπιστα αποτελέσματα, ενώ όσο πιο μακριά ή όσο πιο κοντά είναι κάποιο πρόσωπο λόγω του φθηνού εξοπλισμού που διαθέτουμε η πρόβλεψη γίνεται πιο δύσκολη και πιο πολύπλοκη και συνεπώς υπάρχει σοβαρό ενδεχόμενο για εσφαλμένα αποτελέσματα. Στο σημείο αυτό αναφέρεται ξανά πως όταν ένα πρόσωπο βρίσκεται εκτός της ζώνης αναγνώρισης γίνεται απλή ανίχνευση και ενημέρωση με κατάλληλο μήνυμα ώστε το άτομο που προσπαθεί να αναγνωριστεί να εισέλθει εντός αυτής.

Για την εκπαίδευση του συστήματος είναι απαραίτητο να υπάρχει στην βάση δεδομένων ένας μεγάλος αριθμός από άγνωστα πρόσωπα έτσι ώστε η ετικέτα που προκύπτει από την πρόβλεψη του συστήματος να αντιστοιχίζεται με αυτά τα πρόσωπα πετυχαίνοντας τον στόχο αναγνώρισης αγνώστου προσώπου και αποφυγή σφάλματος με τα γνωστά πρόσωπα που έχουν καταχωρηθεί στην βάση.

Τελειώνοντας αυτήν την ενότητα θα πρέπει να αναλύσουμε ακόμη μια προϋπόθεση επιτυχίας. Οι τιμές των παραμέτρων που επιλέχθηκαν στις συναρτήσεις της OpenCV για την ανίχνευση και αναγνώριση των προσώπων έγιναν πειραματικά αναλόγως το περιβάλλον εργασίας, το φωτισμό και τον εξοπλισμό. Όπως φαίνεται και στις Εικόνες 6.6. και 6.7. οι διαφορετικοί αριθμοί που εισάγουμε εντός των συναρτήσεων αυτών μπορεί να επηρεάσουν τόσο τη σωστή ανίχνευση όσο και τη πρόβλεψη – αξιοπιστία του συστήματος.



Εικόνα 6.6. Λανθασμένη Ανίχνευση Με Ορίσματα detectMultiScale(gray, 1.1, 1)



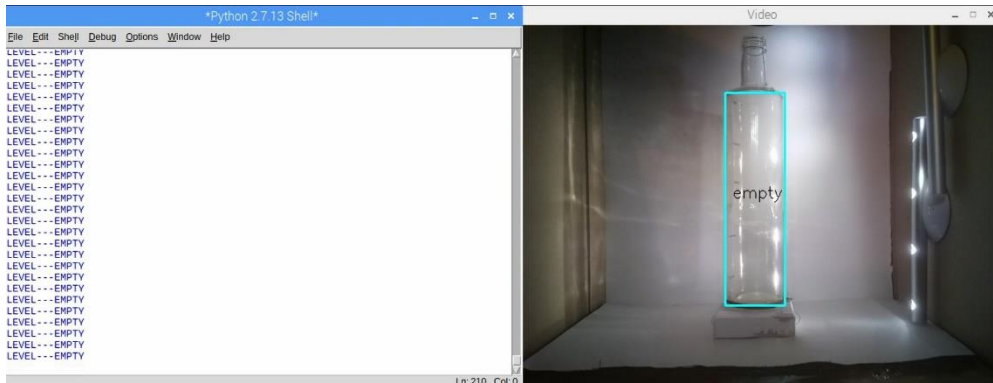
Εικόνα 6.7. Αλλαγή τιμής conf – Εύρος Ζώνης Αναγνώρισης Με Σταθερό Όρισμα rec=
cv2.face.LBPHFaceRecognizer_create ()

6.2. Αποτελέσματα – Συμπεράσματα Για Την Ανίχνευση Και Έλεγχο Στάθμης Υγρού Σε Μπουκάλι

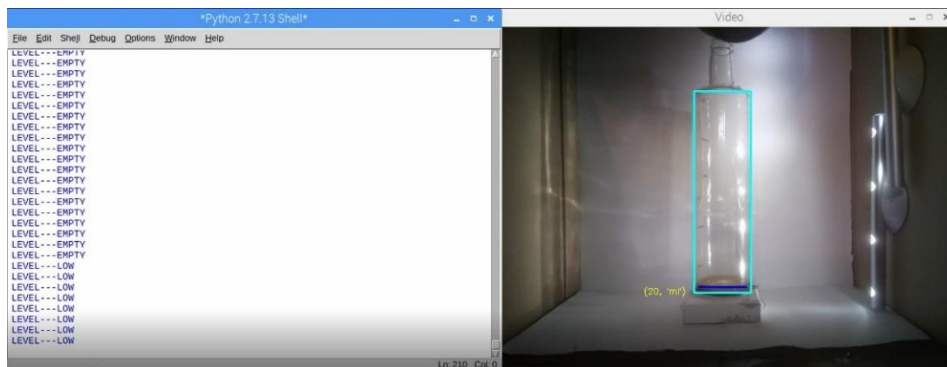
6.2.1. Αποτελέσματα

Αρχικά, στην ενότητα αυτή θα παρουσιαστούν τα αποτελέσματα από την εφαρμογή του κώδικα της ενότητας 5.3. Παρακάτω παρατίθενται φωτογραφίες από τις διαφορετικές περιπτώσεις πλήρωσης του μπουκαλιού. Στο αριστερό μέρος της εικόνας τρέχει το πρόγραμμα τυπώνοντας το επίπεδο της στάθμης του υγρού. Στο δεξί μέρος της εικόνας φαίνεται το μπουκάλι το οποίο γεμίζει και τυπώνεται τόσο η γραμμή στάθμης που έχει δημιουργήσει το πρόγραμμα όσο και τα ml σε πραγματικό χρόνο. Συγκεκριμένα είχαν οριστεί τα εξής επίπεδα στάθμης:

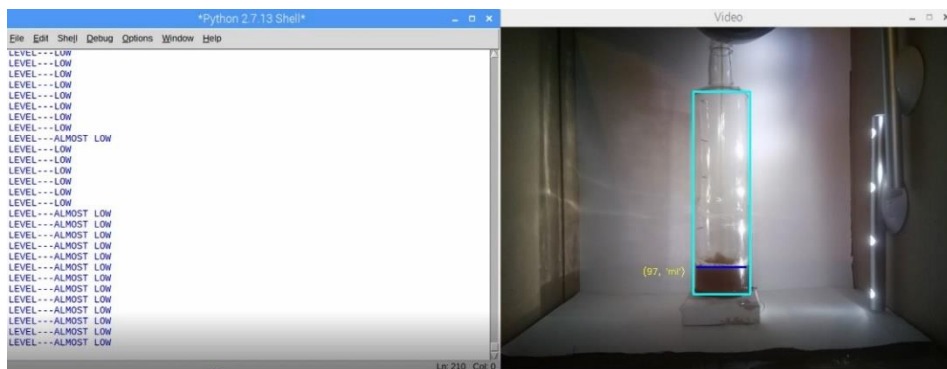
1. Άδειο (Empty)
2. Χαμηλό (Low)
3. Σχεδόν Χαμηλό (Almost Low)
4. Κανονικό (Normal)
5. Σχεδόν Γεμάτο (Almost Full)
6. Γεμάτο – Σταματήστε (Full – Stop)



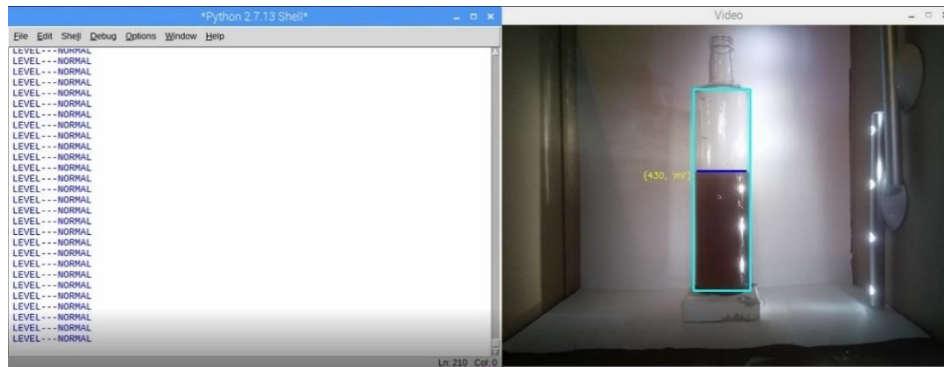
Εικόνα 6.8. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Άδειο



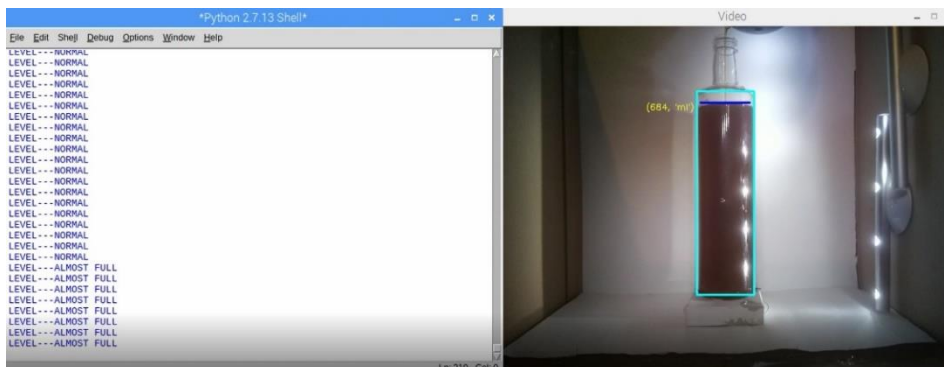
Εικόνα 6.9. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Χαμηλό



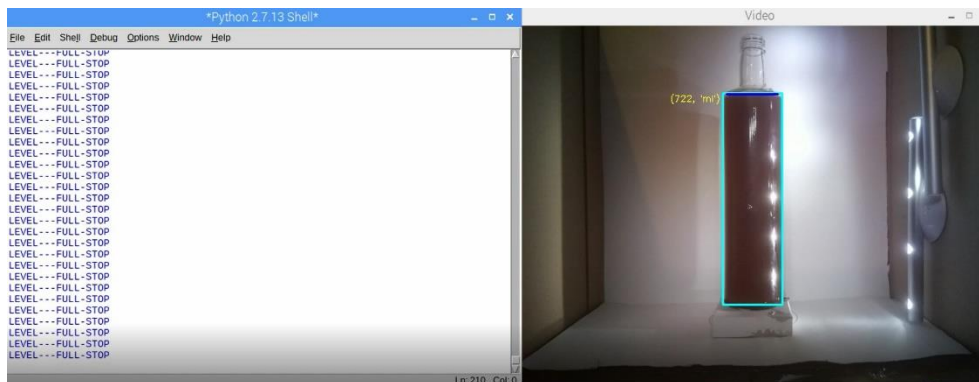
Εικόνα 6.10. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Σχεδόν Χαμηλό



Εικόνα 6.11. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Κανονικό



Εικόνα 6.12. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Σχεδόν Γεμάτο



Εικόνα 6.13. Ανίχνευση Και Έλεγχος Στάθμης Υγρού Σε Μπουκάλι – Επίπεδο: Γεμάτο, Σταματήστε

6.2.2. Συμπεράσματα

Κατά την προσπάθεια κατασκευής του κώδικα και εφαρμογής του διαπιστώσαμε πως υπάρχουν συγκεκριμένες προϋποθέσεις επιτυχίας. Οι βασικότερες ήταν ο σταθερός φωτισμός, το σταθερό φόντο (background) και η σταθερή θέση του μπουκαλιού και της κάμερας, καθώς και η μεταξύ τους απόσταση.

Όσον αφορά το φωτισμό, θεωρούμε ότι διαδραματίζει τον καθοριστικότερο ρόλο, διότι σε περίπτωση αλλαγής του, επηρεάζεται ο θόρυβος της εικόνας. Όσο εντονότερος είναι

ο φωτισμός τόσο αυξάνεται και ο θόρυβος. Αυτό δυσχεραίνει την ανίχνευση, επειδή σχηματίζονται περισσότερες γραμμές στην εικόνα ακρών. Η δυσκολία αυτή αντιμετωπίζεται μέσω της εφαρμογής ενός φίλτρου Gauss. Η βελτίωση είναι εμφανής στην εικόνα 6.14.



Εικόνα 6.14. Εικόνα Ακρών Με Εφαρμογή Φίλτρου Gauss Και Χωρίς

Ο φωτισμός επηρεάζει και την ανίχνευση της γραμμής στάθμης ανάλογα με τη γωνία στην οποία το φως συναντά το μπουκάλι. Με άλλα λόγια, διαπιστώσαμε πως σε συγκεκριμένες γωνίες δεν ήταν δυνατή η σωστή ανίχνευση της γραμμής, ακόμα και αν αλλάζαμε τις υπόλοιπες σημαντικές παραμέτρους (π.χ. θέση μπουκαλιού ή απόσταση κάμερας και μπουκαλιού). Αυτό ενδεχομένως οφείλεται και στο σχήμα του μπουκαλιού, καθώς σε κάποιες περιπτώσεις ανιχνευόταν η πίσω γραμμή της επιφάνειας.

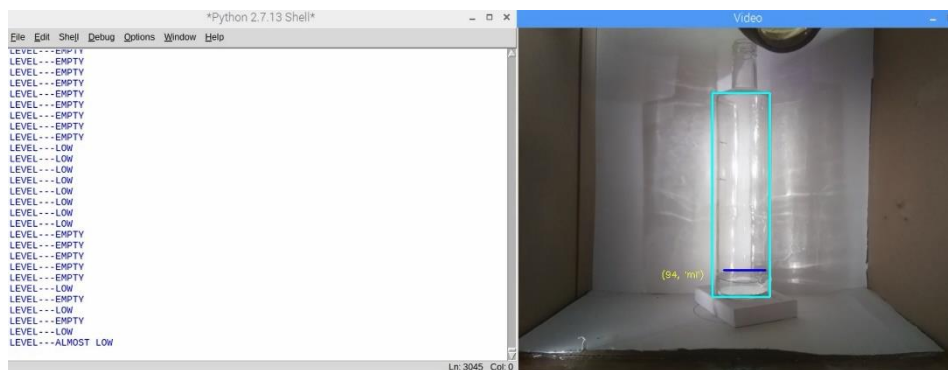
Επίσης, απαραίτητη προϋπόθεση είναι το σταθερό φόντο (background). Μετά από δοκιμές παρατηρήσαμε ότι ένα πολύ καλό χρώμα φόντου ήταν το λευκό, καθώς δημιουργεί αντίθεση με το σκουρόχρωμο υγρό, το τσάι, το οποίο χρησιμοποιήσαμε και έτσι εξυπηρετεί την ανίχνευση της γραμμής στάθμης. Εξάλλου, χωρίς το σταθερό φόντο αυξανόταν και ο θόρυβος της εικόνας. Αυτό επηρέαζε σε μεγάλο βαθμό την ανίχνευση, διότι το σύστημα ανίχνευε περισσότερες άκρες και γραμμές και δεν μπορούσε να εντοπίσει πάντα τη σωστή γραμμή, αυτή που αντιστοιχούσε στη γραμμή στάθμης του υγρού.

Αντίστοιχα, βασικές παράμετροι για τη σωστή λειτουργία του προγράμματος ήταν η θέση του μπουκαλιού (απόσταση και ύψος) σε σχέση με τη θέση της κάμερας. Εκτός του ότι και τα δύο έπρεπε να είναι σταθερά, έπρεπε να βρίσκονται και στην ίδια ευθεία (έστω και κατά προσέγγιση) ώστε να μην ανιχνεύεται η πίσω γραμμή εξαιτίας της διάθλασης. Επίσης, η σταθερή θέση των δύο αυτών αντικειμένων ήταν απαραίτητη και για το τμήμα του κώδικα που αφορούσε τον υπολογισμό των ml υγρού. Αυτό οφείλεται στο ότι υπολογίστηκε ένα σταθερό ορθογώνιο παραλληλόγραμμο παράθυρο μέσα στο οποίο γινόταν ο υπολογισμός των

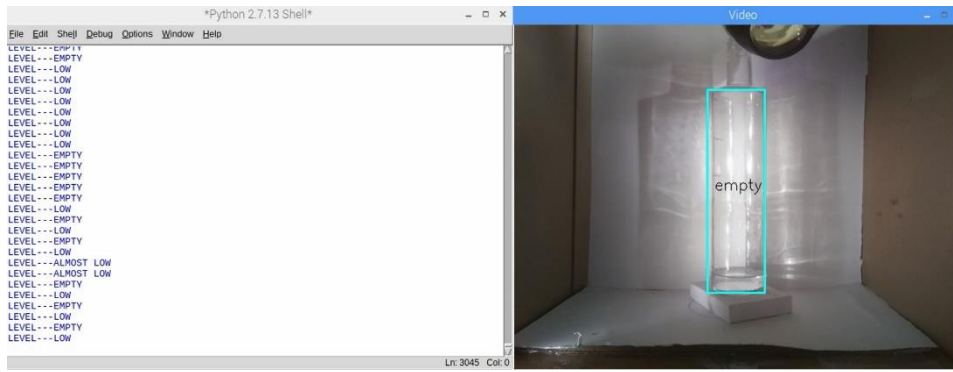
pixel. Εάν άλλαζε η θέση του μπουκαλιού ή της κάμερας, το παράθυρο αυτό δε θα περιλάμβανε όλο το μπουκάλι ή θα περιλάμβανε και χώρο εκτός του μπουκαλιού (ανάλογα με το αν θα μετακινούνταν σε κοντινότερη ή μακρινότερη απόσταση αντίστοιχα) και έτσι δε θα ήταν σωστός ο υπολογισμός των ml.

Εκτός από τις προϋποθέσεις επιτυχίας που παρουσιάστηκαν, θα θέλαμε να δώσουμε έμφαση και σε ορισμένες επιπλέον δυσκολίες που συναντήσαμε, όταν προσπαθήσαμε να ανιχνεύσουμε τη στάθμη του υγρού σε ένα διαφανές υγρό, όπως το νερό, σε αντίθεση με το σκουρόχρωμο υγρό, όπως το τσάι. Στο σκουρόχρωμο υγρό η στάθμη ανιχνεύθηκε ευκολότερα σε σχέση με το νερό, διότι είναι περισσότερο εμφανής.

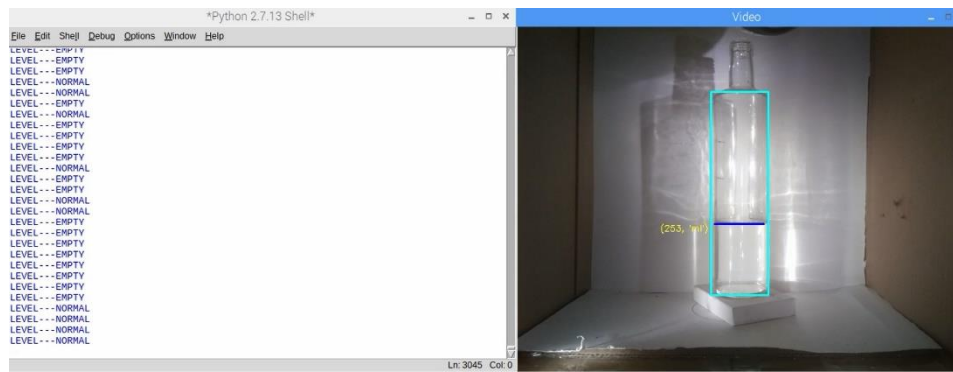
Συγκεκριμένα, ο κώδικάς μας κατάφερνε μόνο σε ορισμένες περιπτώσεις να ανιχνεύσει τη γραμμή στάθμης, ενώ σε κάποιες άλλες δεν ήταν τόσο αποτελεσματικός, καθώς ανίχνευε την πίσω γραμμή επιφάνειας ή δεν μπορούσε να εντοπίσει τη γραμμή στάθμης. Αυτό οφείλεται κυρίως στο φωτισμό. Δοκιμάζοντας τόσο πιο απαλό όσο και πιο έντονο φωτισμό, διαπιστώσαμε ότι τα αποτελέσματα που παίρναμε δεν ήταν συνεπή. Άλλες φορές ο απαλός φωτισμός εξυπηρετούσε την ανίχνευση, ενώ άλλες όχι. Αντίστοιχα και με εντονότερο φωτισμό μπορούσαμε να ανιχνεύσουμε τη γραμμή, ωστόσο και πάλι δεν υπήρχε σταθερότητα στην ανίχνευση κάθε φορά που έτρεχε ο κώδικας. Αλλάζοντας και την παράμετρο του σταθερού φόντου, διαπιστώσαμε ότι για το διαφανές υγρό, καλύτερο χρώμα φόντου ήταν το μαύρο, χωρίς ωστόσο να εξασφαλίζει την ανίχνευση της στάθμης κάθε φορά για αυτό το λόγο παρέμεινε σταθερό το λευκό φόντο. Στις Εικόνες 6.15, 6.16, 6.17 αποτυπώνεται η δυσκολία ανίχνευσης της στάθμης του νερού. Τέλος, αξίζει να σημειωθεί πως καθοριστικό ρόλο διαδραματίζει η αξία του εξοπλισμού. Ο σχετικά φθηνός εξοπλισμός που αξιοποιήθηκε θεωρείται πως επηρέασε αρνητικά την επίτευξη του επιθυμητού αποτελέσματος.



Εικόνα 6.15. Λανθασμένη Ανίχνευση Πίσω Γραμμής Επιφάνειας



Εικόνα 6.16. Μη Ανίχνευση Της Στάθμης Του Νερού



Εικόνα 6.17. Σωστή Ανίχνευση

Κεφάλαιο 7: Συζήτηση – Μελλοντική Έρευνα

Στο συγκεκριμένο κεφάλαιο θα αναφερθούμε σε μελλοντικές εφαρμογές που θα μπορούσαν να εφαρμοστούν τα συστήματα που αναπτύχθηκαν στην παρούσα εργασία καθώς και στους περιορισμούς της έρευνας.

Όσον αφορά το σύστημα ανίχνευσης και αναγνώρισης προσώπων μελλοντικές εφαρμογές που θα μπορούσαν να χρησιμοποιηθεί είναι σε συστήματα ασφαλείας, για παράδειγμα στον έλεγχο των ατόμων που εισέρχονται σε διάφορων ειδών εγκαταστάσεις (τράπεζες, μουσεία, ιδρύματα κ.α.). Επιπλέον, μια πολύ χρήσιμη εφαρμογή είναι αυτή των ρομποτικών συστημάτων όπου θα εξυπηρετούν κυρίως άτομα με δυσκολίες (όρασης, κίνησης) ή ηλικιωμένους επιτρέποντας την είσοδο μόνο στα επιθυμητά πρόσωπα, όπως γιατροί, οικογενειακό περιβάλλον κ.τ.λ.

Στην περίπτωση του συστήματος ελέγχου στάθμης υγρού μια πολύ βασική εφαρμογή είναι αυτή της γραμμής παραγωγής βιομηχανικού περιβάλλοντος. Με το σύστημα ανίχνευσης υπολογιστικής όρασης θα επιτυγχάνεται ο έλεγχος της στάθμης του υγρού και θα λαμβάνονται αποφάσεις. Εάν το επίπεδο πλήρωσης του μπουκαλιού είναι το επιθυμητό, αυτό θα μπορεί να συνεχίζει στην επόμενη φάση της παραγωγής. Εάν όχι, θα μπορούν να εκτελεστούν οι κατάλληλες ενέργειες (π.χ. απόρριψη μπουκαλιού, επιστροφή στη προηγούμενη φάση παραγωγής για σωστή πλήρωση). Ακόμα το σύστημα που αναπτύξαμε θα μπορούσε να έχει εφαρμογή σε εργαστήρια χημείας. Εκεί σε πραγματικό χρόνο θα γίνεται έλεγχος της στάθμης των υγρών στα δοχεία και στη συνέχεια σωστή και ασφαλής ανάμειξη των διάφορων χημικών στοιχείων. Είναι εύκολο να κατανοήσει κάποιος πως η ανάμειξη σωστών ποσοτήτων από διάφορα χημικά στοιχεία ή ενώσεις επηρεάζει τόσο το τελικό επιθυμητό αποτέλεσμα των ερευνητών όσο και την ασφάλεια των εργαζομένων στα εργαστήρια.

Τέλος, αναφέρονται οι βασικοί περιορισμοί της έρευνάς μας. Για την ύπαρξη καλύτερων αποτελεσμάτων και αξιοπιστίας των προηγούμενων συστημάτων και εφαρμογών είναι αναγκαίες οι εξής βελτιώσεις:

- Καλύτερος εξοπλισμός, όπως η κάμερα, ο επεξεργαστής – υπολογιστική ισχύς, το περιβάλλον εργασίας
- Σωστός και σταθερός φωτισμός

- Αυστηρότητα κριτηρίων, όπως στις αποστάσεις γωνιών θέασης
- Συνδυασμός μεθόδων, όπως η αναγνώριση προσώπου μαζί με την αναγνώριση των ματιών, με σκοπό την επίτευξη μεγαλύτερης αξιοπιστίας
- Συνδυασμός του συστήματος ελέγχου στάθμης υγρού με αυτοματοποιημένο σύστημα για λήψη σωστών αποφάσεων.

Βιβλιογραφία

BIBΛΙΑ

1. Bradski, G. & Kaehler, A. *'Learning OpenCV'*. O' Reilly, 2008.
2. Li, S. & Jain, A. *'Handbook of Face Recognition'*, Second Edition. Springer-Verlag, London, 2011.
3. Szeliski, R. *'Computer Vision: Algorithms and Applications'*. Springer, 2010.
4. Δημητριάδης, Σ. *'Εισαγωγή στον Προγραμματισμό με Python - Μάθημα στο Διαδίκτυο'*. Σημειώσεις Σεμιναρίου, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης – Δομή Δια Βίου Μάθησης, 2015.
5. Διαμαντάρας, Κ. *'Τεχνητά Νευρωνικά Δίκτυα'*. Κλειδάριθμος, 2007.
6. Μαγκούτης, Κ. & Νικολάου, Χ. *'Εισαγωγή Στον Αντικειμενοστραφή Προγραμματισμό Με Python: Μια Προσέγγιση Από Την Πλευρά Των Υπολογιστικών Συστημάτων'*. Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα www.kallipos.gr , 2015.
7. Τζαφέστας, Σ., *'Ρομποτική: Ανάλυση, Έλεγχος, Σχεδιασμός, Προγραμματισμός, Αίσθηση'*. ΕΜΠ, Αθήνα, 2003.

ΑΡΘΡΑ

8. Ahonen, T. & Pietikainen, M. *'Face Description with Local Binary Patterns: Application to Face Recognition'*. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, Volume 28 (12), 2006.
9. Canny, J. *'A Computational Approach to Edge Detection'*. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, Volume 8 (6), 1986.
10. Dongdong, G. & Na L. *'New Liquid Level Measurement Method Based on Randomized Hough Transform'*. *IEEE*, 2017.
11. Heisele, B., Ho, P. & Poggio, T. *'Face Recognition with Support Vector Machines: Global versus Component-based Approach'*. *IEEE*, 2002.
12. Hyvärinen, A., & Oja, E. *'Independent component analysis: algorithms and applications'*. *Neural Networks*, Volume 13 (4-5), pp.411–430, 2000.
13. Jain, N., Sharma, N., Kamboj, N. & Kakani, A. *'Analysis of Different Methods for Face Recognition'*. *International Journal of Innovative Computer Science & Engineering*, Volume 1(2), pp.18-21, 2014.

14. Kumar, S. & Kaur, H. 'Face Recognition Techniques: Classification and Comparisons'. *International Journal of Information Technology and Knowledge Management*, Volume 5 (2), pp. 361-363, 2012.
15. Lienhart, R., Kuranov, A. & Pisarevsky, V. 'Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection'. *MRL Technical Report*, 2002.
16. Matas, J. 'Robust Detection of Lines Using the Progressive Probabilistic Hough Transform'. *Computer Vision and Image Understanding*, Volume 78, pp.119–137, 2000.
17. Parmar, D. & Mehta, B. 'Face Recognition Methods & Applications'. *International Journal of Computer Technology & Applications*, Volume 4 (1), pp.84-86, 2013.
18. Vassiljeva, K., Teplyakov, A. & Petlenkov, E. 'NN-ANARX Model based Control of Liquid Level using Visual Feedback'. *IEEE*, 2015.
19. Viola, P. & Jones. M. 'Rapid Object Detection using a Boosted Cascade of Simple Features'. *IEEE*, 2001.
20. Viola, P. & Jones. M. 'Robust Real-Time Face Detection'. *International Journal of Computer Vision*, Volume 57(2), pp.137–154, 2004.
21. Yang, M., Kriegman, D. & Ahuja, N. 'Detecting Faces in Images: A Survey'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

ΕΡΓΑΣΙΕΣ – ΔΙΑΤΡΙΒΕΣ

22. De Carrera, F. & Marquez, I. 'Face Recognition Algorithms'. Universidad del Pais Vasco, 2010.
23. Eppel, S. & Kachman, T. 'Computer Vision-Based Recognition Of Liquid Surfaces And Phase Boundaries In Transparent Vessels, With Emphasis On Chemistry Applications'. Cornell University Library, 2014.
24. Lopez, L.S. 'Local Binary Patterns applied to Face Detection and Recognition'. Universitat Politecnica de Catalunya, 2010.
25. Tibbalds, A. 'Three Dimensional Human Face Acquisition for Recognition'. Ph.D. Thesis. Trinity College, University of Cambridge, Cambridge, 1998.
26. Τσαπατσούλης, Ν. 'Προηγμένες Τεχνικές Αναγνώρισης Προσώπων και Ανάλυσης Εκφράσεων'. Διδακτορική Διατριβή. Εθνικό Μετσόβιο Πολυτεχνείο (ΕΜΠ). Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Αθήνα, 2000.

ΠΗΓΕΣ ΑΠΟ ΤΟ ΔΙΑΔΙΚΤΥΟ

27. <http://legacy.python.org/dev/peps/pep-0373/> προσπελάστηκε στις 16/12/2017
28. <http://www.face-rec.org/algorithms/> προσπελάστηκε στις 20/12/2017
29. https://docs.opencv.org/2.4/modules/contrib/doc/face_recognition/tutorial.html#local-binary-patterns-histograms προσπελάστηκε στις 27/12/2017
30. https://docs.opencv.org/2.4/modules/contrib/doc/face_recognition/tutorial.html#face-recognition-with-opencv προσπελάστηκε στις 12/01/2018
31. https://docs.opencv.org/2.4/modules/imgproc/doc/feature_detection.html προσπελάστηκε στις 14/01/2018
32. https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_canny/py_canny.html?highlight=canny προσπελάστηκε στις 13/01/2018
33. https://docs.opencv.org/3.0-beta/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.html?highlight=canny προσπελάστηκε στις 13/01/2018
34. https://docs.opencv.org/3.0-beta/doc/tutorials/imgproc/imgtrans/hough_lines/hough_lines.html?highlight=probabilistic%20hough προσπελάστηκε στις 14/01/2018
35. https://en.wikipedia.org/wiki/Canny_edge_detector προσπελάστηκε στις 13/01/2018
36. https://en.wikipedia.org/wiki/Gaussian_blur προσπελάστηκε στις 15/01/2018
37. https://en.wikipedia.org/wiki/Raspberry_Pi προσπελάστηκε στις 20/01/2018
38. https://en.wikipedia.org/wiki/Region_of_interest προσπελάστηκε στις 12/01/2018
39. <https://github.com/informramiz/Face-Detection-OpenCV> προσπελάστηκε στις 10/01/2018
40. <https://opencv.org/about.html> προσπελάστηκε στις 15/12/2017
41. <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b> προσπελάστηκε στις 27/12/2017
42. <https://www.python.org/about/> προσπελάστηκε στις 16/12/2017
43. <https://www.raspberrypi.org/blog/new-8-megapixel-camera-board-sale-25/> προσπελάστηκε στις 20/01/2018
44. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/> προσπελάστηκε στις 20/01/2018