



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Ανίχνευση Αντικειμένων - Χαρακτηριστικών  
με Οπτικές Τεχνικές  
σε Συστήματα Βιντεοεπιτήρησης**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

ΤΟΥ

**ΝΙΚΟΥΛΗ ΠΕΡΙΚΛΗ**

**Επιβλέπων :** Ευάγγελος Β. Χριστοφόρου  
Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2020

Η σελίδα αυτή είναι σκόπιμα λευκή.



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ  
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

# Ανίχνευση Αντικειμένων - Χαρακτηριστικών με Οπτικές Τεχνικές σε Συστήματα Βιντεοεπιτήρησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

**ΝΙΚΟΥΛΗ ΠΕΡΙΚΛΗ**

**Επιβλέπων :** Ευάγγελος Β. Χριστοφόρου  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 25<sup>η</sup> Σεπτεμβρίου 2020.

.....  
Καθηγητής  
Χριστοφόρου Ευάγγελος

.....  
Επίκουρος Καθηγητής  
Τσαραμπάρης Παναγιώτης

.....  
Ομότιμος Καθηγητής  
Θεοδώρου Νικόλαος

Αθήνα, Σεπτέμβριος 2020

(Υπογραφή)

.....

**ΝΙΚΟΥΛΗΣ ΠΕΡΙΚΛΗΣ**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© 2020 – All rights reserved

## Περίληψη

Τα συστήματα βιντεοεπιτήρησης [1] έχουν πλέον θέση στην καθημερινή μας ζωή, καθώς η χρήση τους σε ορισμένες εφαρμογές είναι αναγκαία, ενώ σε άλλες επαυξάνει την χρηστικότητα συγκεκριμένων σκοπών, όπως σε θέματα υγείας, ασφαλείας κ.α. Η παρούσα διπλωματική εργασία εστιάζει σε τεχνικές επεξεργασίας δεδομένων εικόνας που συλλέγονται από συστήματα βιντεοεπιτήρησης σε περιπτώσεις όπου δεν περιορίζεται η χρηστικότητά τους.

Στην παρούσα διπλωματική θα εξεταστούν και θα περιγραφούν, οι σημαντικές τεχνικές ανίχνευσης προσώπου, η ανίχνευση κίνησης καθώς και η ανίχνευση μηκών κύματος οπτικού πεδίου. Οι τεχνικές αυτές μπορούν να θεωρηθούν βασικές για την επίλυση πολυπλοκότερων προβλημάτων σε θέματα επεξεργασίας εικόνας.

Το πρόβλημα της ανίχνευσης προσώπου [2] (face detection) είναι αρκετά δημοφιλές, αλλά απαιτητικό, και παραμένει ερευνητικά ανοιχτό. Οι λόγοι που καθιστούν τη λύση του συγκεκριμένου προβλήματος δύσκολη είναι πολλοί, με πιο σημαντικούς, την χρωματική ποικιλία, τις διαφορετικές στάσεις του σώματος, τις φωτομετρικές παραμορφώσεις, το γεγονός ότι συνήθως επιθυμούμε να γίνει η ανίχνευση σε χώρους που υπάρχει συνωστισμός και άλλα.

Το πρόβλημα ανίχνευσης κίνησης βασίζεται στην σύγκριση δύο ή περισσότερων πλαισίων, ενώ το πρόβλημα της ανίχνευσης μηκών κύματος οπτικού πεδίου βασίζεται σε μαθηματικά μοντέλα μετατροπής του RGB σε τρισδιάστατες μαθηματικές αναπαραστάσεις.

Αναπτύχθηκαν συνολικά τρία λογισμικά, ένα για την αναγνώριση και την κάλυψη στοιχείων προσώπου, ένα για την ανίχνευση κινούμενου σώματος, και ένα λογισμικό φιλτραρίσματος των μηκών κύματος του οπτικού φάσματος. Τα προγράμματα δοκιμάστηκαν από web – camera υπολογιστή και είναι γραμμένα στη γλώσσα προγραμματισμού Python [3], με τη χρήση κυρίως της βιβλιοθήκης αλγορίθμων επεξεργασίας εικόνας και όρασης υπολογιστών OpenCV [4].

**Λέξεις Κλειδιά:** <<βιντεοεπιτήρηση, python, openCV>>

## Abstract

Video surveillance systems [1] have a place in our daily lives, as their use in some applications is necessary, while in others, it increases the usefulness of specific purposes, such as health, safety, etc. The dissertation focuses on image data processing techniques, collected by video surveillance systems in cases where their usefulness is not limited.

The current dissertation examines and describes the most important techniques of face detection, motion detection and optical wavelength detection. These techniques can be considered essential for solving more complex image processing problems.

The problem of face detection [2] is quite popular (familiar) and demanding, but it remains open for research. The reasons that make the solution of this problem difficult are many, the most important ones being: the colour variety, the different postures, the photometric distortions, the fact that we usually want to be detected in crowded places and more.

The motion detection problem is based on the comparison of two or more frames, while the optical wavelength wave detection problem is based on mathematical models for converting RGB to 3D mathematical representations.

Three algorithms have been developed in total, one for face recognition and coverage, one for moving body detection, and one for optical spectrum wave filtering software. The programs were tested by a web-camera computer, written in the Python programming language [3] using mainly the library of OpenCV computer image processing and vision algorithms [4].

**Keywords:**<< video surveillance, python, openCV>>

## **Ευχαριστίες**

Αρχικά, θα ήθελα να ευχαριστήσω τον Καθηγητή κ. Ευάγγελο Χριστοφόρου για την ευκαιρία που μου έδωσε να ασχοληθώ με το συγκεκριμένο τομέα της επιστήμης των υπολογιστών στο Εργαστήριο Ηλεκτρονικών Αισθητήρων.

Επίσης θα ήθελα να ευχαριστήσω τον διδάκτορα Σπυρίδωνα Αγγελόπουλο, χωρίς τις κατευθύνσεις και οδηγίες του οποίου, η εκπόνηση της διπλωματικής αυτής εργασίας δε θα μπορούσε να ολοκληρωθεί.

Τέλος, ένα μεγάλο ευχαριστώ στην οικογένειά μου, στους φίλους μου, και την καθηγήτρια Αγγλικών μου την κ. Έλενα, για τη συνεχή τους στήριξη όλο το διάστημα των σπουδών, καθώς επίσης και την ίδια την Σχολή για τα ωραιότερα συναισθήματα και αναμνήσεις που μου πρόσφερε κατά την διάρκεια των σπουδών μου.

Νικούλης Περικλής

Η σελίδα αυτή είναι σκόπιμα λευκή.



---

## Πίνακας Περιεχομένων

<b>1</b>	<b>Εισαγωγή</b> .....	<b>1</b>
1.1	Συστήματα Βιντεοεπιτήρησης.....	2
1.2	Επεξεργασία Δεδομένων από Συστήματα Βιντεοεπιτήρησης.....	2
1.3	Συνεισφορά.....	7
1.4	Οργάνωση κειμένου.....	8
<b>2</b>	<b>Υπόβαθρο</b> .....	<b>11</b>
2.1	Τεχνικά Χαρακτηριστικά Συστημάτων Βιντεοεπιτήρησης.....	11
2.2	Η Γλώσσα Προγραμματισμού Python.....	16
2.3	Η Βιβλιοθήκη OpenCV.....	19
2.4	Η Βιβλιοθήκη DLib.....	22
2.5	Ο Διαχειριστής Πακέτων Anaconda.....	23
2.6	Τεχνικές Ανίχνευσης Αντικειμένων.....	23
	2.6.1 Ανίχνευση Προσώπου με Ταξινομητές Χαρακτηριστικών (HaarCascade) στο OpenCV.....	25
	2.6.2 Ανίχνευση Προσώπου βασισμένο με DeepLearning στο OpenCV ..	34
	2.6.3 Ανίχνευση Προσώπου με Ιστόγραμμα Προσανατολισμένων Κλίσεων (HoG) στο Dlib.....	36
	2.6.4 Ανίχνευση Προσώπου με DeepLearning στο Dlib.....	42
	2.6.5 Ανίχνευση Κίνησης με Σύγκριση Frames.....	45
	2.6.6 Ανίχνευση Μηκών Κύματος Οπτικού Φάσματος.....	50
<b>3</b>	<b>Αλγόριθμοι Επεξεργασίας Εικόνας</b> .....	<b>61</b>
3.1	Αλγόριθμοι Επεξεργασίας Εικόνας.....	61
	3.1.1 Αλγόριθμος Αναγνώρισης και Κάλυψης Στοιχείων Προσώπου.....	61

---

3.1.2	Αλγόριθμος Αναγνώρισης Κίνησης.....	65
3.1.3	Αλγόριθμος Αναγνώρισης Μηκών Οπτικού Φάσματος.....	69
3.2	Σύνοψη συγκρίσεων.....	72
<b>4</b>	<b>Επίλογος.....</b>	<b>83</b>
4.1	Σύνοψη και συμπεράσματα.....	84
4.2	Μελλοντικές επεκτάσεις.....	84
<b>5</b>	<b>Βιβλιογραφία.....</b>	<b>87</b>

---

# 1

## ***Εισαγωγή***

Εδώ και αρκετά χρόνια, παρατηρείται μια τρομακτική αύξηση στη χρήση συστημάτων βιντεοεπιτήρησης, τόσο σε ιδιωτικούς όσο και σε δημόσιους χώρους με ποικίλες εφαρμογές, από μία απλή επιτήρηση χώρου μέχρι και την εξειδικευμένη επιτήρηση ασθενούς.

Οι ψηφιακές εικόνες και το βίντεο γίνονται όλο και περισσότερο σημαντικές στην εποχή των πολυμέσων. Με τη σειρά της, η ανίχνευση αντικειμένων (object detection) είναι μια από τις κατηγορίες της τεχνολογίας υπολογιστών, η οποία συνδέεται με την επεξεργασία εικόνας και την υπολογιστική όραση και η οποία αλληλοεπιδρά μέσω της ανίχνευσης αντικειμένων μιας συγκεκριμένης ομάδας που μπορεί να αποτελείται από ανθρώπινα πρόσωπα, κτίρια, δέντρα, αυτοκίνητα, κ.λπ.

Σκοπός της παρούσας διπλωματικής, είναι η ανάπτυξη αλγοριθμικών τεχνικών επεξεργασίας δεδομένων εικόνας, από συστήματα βιντεοεπιτήρησης για την ασφάλεια δεδομένων προσωπικού χαρακτήρα, καθώς και την στοχευμένη επεξεργασία δεδομένων, σε περιπτώσεις που δεν περιορίζεται η χρηστικότητα των εφαρμογών τους.

Παρακάτω αναπτύσσονται οι βασικές έννοιες των συστημάτων βιντεοεπιτήρησης καθώς και τα εργαλεία που χρησιμοποιούνται για την ανάπτυξή τους.

---

## **1.1 Συστήματα Βιντεοεπιτήρησης**

Συστήματα βιντεοεπιτήρησης [5], ονομάζονται τα συστήματα εκείνα, που είναι μόνιμα εγκατεστημένα σε ένα χώρο και έχουν τη δυνατότητα λήψης ή/και μετάδοσης εικόνας ή και ήχου προς οθόνες προβολής ή μηχανήματα καταγραφής (όπου οι κάμερες μπορούν να συνδέονται με την οθόνη μέσω του μηχανήματος καταγραφής είτε απευθείας, είτε μέσω δικτύου/διαδικτύου) ή ακόμα και σε άλλα συστήματα αισθητήρων προκειμένου να συνεισφέρουν με τα δεδομένα τους σε μια πολυπλοκότερη εφαρμογή. Η συνηθέστερη περίπτωση ενός συστήματος είναι τα κλειστά κυκλώματα τηλεόρασης.

Η εγκατάσταση των συστημάτων βιντεοεπιτήρησης αποσκοπεί κυρίως στην ασφάλεια:

- ❖ Προσωπικού
- ❖ Υλικού
- ❖ Εγκαταστάσεων

## **1.2 Επεξεργασία Δεδομένων από Συστήματα**

### ***Βιντεοεπιτήρησης***

Το ανθρώπινο σύστημα όρασης [6] έχει την ικανότητα να ανιχνεύει και να εντοπίζει με εξαιρετική ακρίβεια πρόσωπα και αντικείμενα που βρίσκονται μέσα στο οπτικό πεδίο του ματιού. Έτσι, μέσω του σταδίου αυτού, ο εγκέφαλος είναι σε θέση να εξακριβώνει την ταυτότητα του ανθρώπου ή του αντικειμένου που κοιτάει ή να αποκωδικοποιεί συναισθήματα, καθώς και άλλα χαρακτηριστικά που αποστέλλονται μέσω της εικόνας. Εύκολα κανείς καταλαβαίνει ότι οι άνθρωποι έχουν μια θαυμάσια δυνατότητα στο να ταυτοποιούν πρόσωπα και αντικείμενα από ότι οι μηχανές. Επομένως, η ανίχνευση προσώπου διαδραματίζει σημαντικό ρόλο, καθώς αποτελεί το πρώτο βήμα σε αλγορίθμους που σχετίζονται με την ευθυγράμμιση προσώπων (face alignment), τη μοντελοποίηση προσώπων (face

---

modell<sup>i</sup>ng),την αναγνώριση προσώπου (face recognition),την αναγνώριση της έκφρασης προσώπου (facial expression recognition) την αλληλεπίδραση ανθρώπου-υπολογιστή κ.α. Με άλλα λόγια η ανίχνευση προσώπων αποτελεί μια τεχνική εντοπισμού της θέσης και του μεγέθους του προσώπου σε μια δεδομένη εικόνα.

Το κυριότερο πρόβλημα που έχει κανείς να επιλύσει όσον αφορά την ανίχνευση προσώπων είναι η κατάλληλη επιλογή του χρωματικού μοντέλου για κατάτμηση του χρώματος δέρματος, τα οποία μοντέλα είναι αρκετά, και το καθένα έχει το δικά του χαρακτηριστικά. Η ανίχνευση των εκφράσεων του προσώπου εξαρτάται σε μεγάλο βαθμό από τα χαρακτηριστικά της εικόνας και επηρεάζεται σε μεγάλο ποσοστό από τον θόρυβο και τις συνθήκες φωτισμού που επικρατούν κατά την διάρκεια λήψης της εικόνας. Οπότε τα προβλήματα που αντιμετωπίζει η ανίχνευση προσώπων μπορούν να αποδοθούν στους παρακάτω παράγοντες:

❖ Στάση σώματος-Πόζα (Pose):

Σε ένα σύστημα παρακολούθησης, η κάμερα τοποθετείται σχεδόν πάντα σε τέτοια θέση όπου οι άνθρωποι δεν μπορούν την προσεγγίσουν. Αν όμως η κάμερα βρίσκεται σε πιο υψηλή θέση τότε τα πρόσωπα των ανθρώπων καταγράφονται από διαφορετικές γωνίες. Όλο αυτό αποτελεί την απλούστερη περίπτωση εφαρμογών παρακολούθησης μεγάλων πόλεων. Στην περίπτωση που οι άνθρωποι περνάνε χωρίς να εστιάζουν το βλέμμα τους στον φακό της κάμερας, το πρόβλημα ανίχνευσης γίνεται αρκετά απαιτητικό μιας και οι Αρχές δεν είναι σε θέση να εντοπίσουν τις παράνομες συμπεριφορές. Επίσης οι εικόνες προσώπων ποικίλουν λόγω της διαφορετικής θέσης που έχει το πρόσωπο ως προς τον φακό της κάμερας, όπως είναι οι εμπρόσθιες εικόνες προσώπων, οι εικόνες με πρόσωπα περιστραμμένα κατά 45°, και οι εικόνες προφίλ. Ακόμα, χαρακτηριστικά του προσώπου όπως η μύτη και τα μάτια μπορούν να επικαλύπτονται σε μικρό ή μεγάλο ποσοστό με αποτέλεσμα να δυσχεραίνουν την ανίχνευση.

---

❖ Έκφραση προσώπου (Facial expression):

Η έκφραση προσώπου είναι ένα από τα στοιχεία εκείνα που χαρακτηρίζουν την ιδιοσυγκρασία του ανθρώπου, αφού μέσω αυτού εκδηλώνονται τα συναισθήματα. Δηλαδή η έκφραση προσώπου σχετίζεται με την «εικόνα» που είναι ζωγραφισμένη στο πρόσωπο του. Αισθήματα όπως η ευτυχία και ο θυμός επηρεάζουν τον μορφασμό του προσώπου, καθώς ο άνθρωπος που χαμογελάει έχει διαφορετική έκφραση προσώπου από εκείνον που είναι θυμωμένος. Άρα γίνεται κατανοητό ότι οι εκφράσεις προσώπων επηρεάζουν άμεσα την εμφάνιση του προσώπου σε μια εικόνα.

❖ Αποκλεισμοί (Occlusion):

Υπάρχουν περιπτώσεις όπου στις εικόνες τα πρόσωπα καλύπτονται από άλλα αντικείμενα. Για παράδειγμα εικόνες που περιέχουν περισσότερα από ένα πρόσωπα, όπου κάποια πρόσωπα μπορεί να καλύπτουν κάποια άλλα.

❖ Προσανατολισμός εικόνας (Image orientation):

Ο προσανατολισμός της εικόνας εξαρτάται σε μεγάλο βαθμό από τον τρόπο λήψης τους. Δηλαδή οι εικόνες των προσώπων είναι διαφορετικές κάθε φορά για διαφορετικές γωνίες περιστροφής του οπτικού άξονα της κάμερας.

❖ Συνθήκες φωτογράφισης:

Στην περίπτωση που σχηματίζεται μια εικόνα, οι συνθήκες φωτισμού που επικρατούν όπως η ένταση του φακού και τα χαρακτηριστικά της κάμερας (φακός, ανταπόκριση αισθητήρα) επηρεάζουν την εμφάνιση του προσώπου.

❖ Χαρακτηριστικά προσώπου (Facial feature):

Στην κατηγορία αυτή ανήκουν χαρακτηριστικά όπως είναι τα γυαλιά, το μουστάκι, ένα σημάδι στο πρόσωπο και άλλα. Χαρακτηριστικά που όπως είναι λογικό σε πολλές περιπτώσεις αρκετά από αυτά παρουσιάζουν αρκετές διαφοροποιήσεις ως προς το σχήμα, το μέγεθος και το χρώμα.

---

❖ Μέγεθος προσώπου (face size):

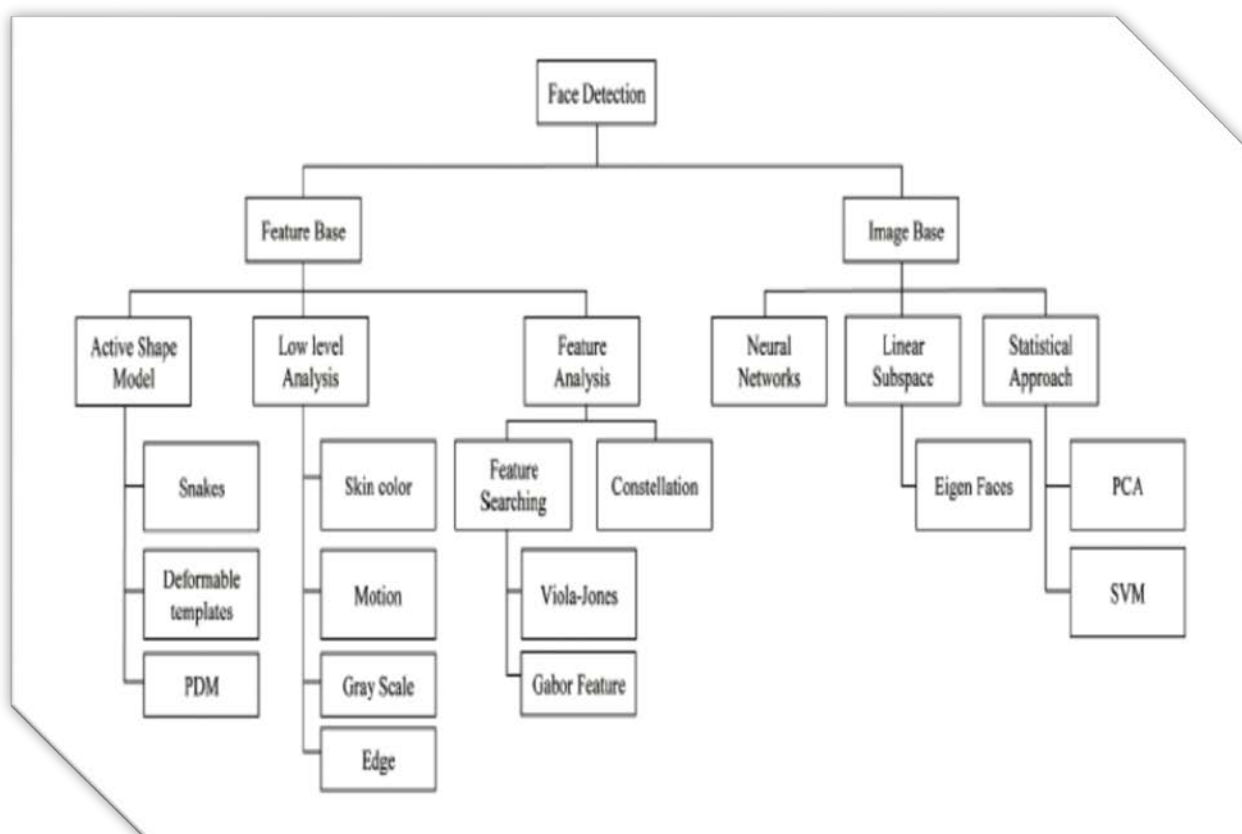
Το μέγεθος του ανθρώπινου προσώπου ποικίλει. Αυτό δεν σημαίνει ότι μόνο οι διαφορετικοί άνθρωποι έχουν διαφορετικό μέγεθος προσώπου, αλλά έχει διαπιστωθεί ότι πρόσωπα που είναι κοντά στην κάμερα εμφανίζονται μεγαλύτερα από όταν βρίσκονται μακριά.

❖ Φωτεινότητα (Illumination):

Η φωτεινότητα είναι ένας σημαντικός παράγοντας στον καθορισμό της ποιότητας της εικόνας και επιδρά αρκετά στην εκτίμηση της εικόνας, και πιο ειδικά στην ανίχνευση προσώπου. Σχετίζεται άμεσα με τον φωτισμό και αξίζει να αναφερθεί ότι οι διαφορετικές συνθήκες φωτισμού επηρεάζουν το πώς φαίνονται τα πρόσωπα. Για παράδειγμα κατά την λήψη του προφίλ ενός ανθρώπου, η μεριά του προσώπου που είναι κοντά στο φακό είναι πιο φωτεινή από την άλλη μεριά [4].

Αναφέρθηκε παραπάνω πως η ανίχνευση προσώπου είναι μια τεχνική εντοπισμού προσώπου σε μια δεδομένη εικόνα. Στην εικόνα αυτή ανιχνεύονται τα χαρακτηριστικά γνωρίσματα του προσώπου με αποτέλεσμα οποιαδήποτε άλλα αντικείμενα όπως δέντρα, κτίρια, αυτοκίνητα, να αγνοούνται παντελώς. Μπορεί να θεωρηθεί ως μια ειδική περίπτωση ανίχνευσης αντικειμένου ή μιας γενικότερης ομάδας. Δηλαδή σκοπός των ανιχνευτών είναι να εντοπίσουν τη θέση και το μέγεθος των αντικειμένων που ανήκουν σε μια ομάδα με κοινά χαρακτηριστικά. Η ανίχνευση προσώπου (face detection) θεωρείται ως πιο γενική περίπτωση εντοπισμού προσώπου (face localization), γιατί ο εντοπισμός προσώπου έχει στόχο την εύρεση των θέσεων και των μεγεθών ενός γνωστού αριθμού προσώπων μιας δεδομένης εικόνας και συνήθως πρόκειται για εικόνες που απεικονίζουν ένα πρόσωπο. Γενικά υπάρχουν δύο τύποι προσεγγίσεων για να ανιχνεύσουν τα χαρακτηριστικά του προσώπου σε μια δεδομένη εικόνα. Πρόκειται για τις προσεγγίσεις με βάση τα χαρακτηριστικά (feature-based approach), και με βάση την εμφάνιση (image-based approach). Η προσέγγιση με βάση τα χαρακτηριστικά έχει στόχο την εξαγωγή των χαρακτηριστικών μιας δοθείσας εικόνας και να τα αντιστοιχήσει με βάση την γνώση που έχει για τα χαρακτηριστικά του προσώπου.

Αντίθετα η προσέγγιση με βάση την εμφάνιση προσπαθεί να πραγματοποιήσει την καλύτερη αντιστοίχιση μεταξύ των εικόνων-εκπαίδευσης (training images) με τις εικόνες δοκιμής (test images). Τέλος, στο *σχήμα 1.1* απεικονίζονται οι μέθοδοι που περιεγράφηκαν μαζί με τις περαιτέρω κατηγορίες που τις απαρτίζουν [7].



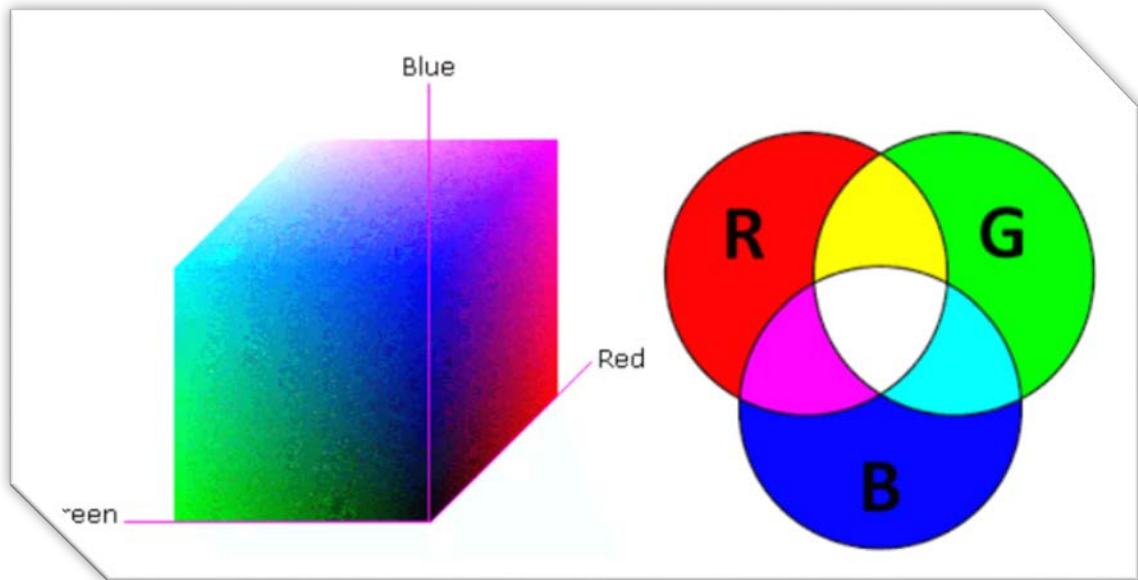
*Σχήμα 1.1 Τεχνικές Ανίχνευσης Προσώπων*

Η ανίχνευση κίνησης από συστήματα βιντεοεπιτήρησης, βασίζεται σε σύγκριση ακολουθιών καρέ εικόνας βίντεο και αποτελεί επίσης μια θεμελιώδης τεχνική, ιδιαίτερα όταν χρησιμοποιείται σε συνδυασμό με άλλες τεχνικές.

Το χρώμα αποτελεί το πιο σημαντικό χαρακτηριστικό για την ανίχνευση ενός αντικειμένου. Η επεξεργασία χρώματος είναι αρκετά πιο γρήγορη από την επεξεργασία άλλων χαρακτηριστικών γνωρισμάτων και παρουσιάζει αμεταβλητότητα με τον προσανατολισμό, ανάλογα με τις συνθήκες φωτισμού που επικρατούν. Γενικά, υπάρχουν πολλά μοντέλα χρώματος. Το πιο γνωστό χρωματικό



μοντέλο είναι η αναπαράσταση σε RGB, όπου το κόκκινο (red), πράσινο (green) και το μπλέξ (blue) συνδυάζονται μεταξύ τους για να αναπαράγουν μια ευρεία σειρά χρωμάτων όπως στο παρακάτω σχήμα 1.2.



Σχήμα 1.2 Χώρος RGB

### 1.3 Συνεισφορά

Στο πλαίσιο της παρούσας διπλωματικής, γίνεται μια παρουσίαση των υπάρχουσών τεχνικών για την αναγνώριση και ανίχνευση προσώπων και αντικειμένων, της ανίχνευσης κινούμενου σώματος, της αναγνώρισης και ανίχνευσης μηκών κύματος οπτικού φάσματος καθώς και μια συγκριτική αξιολόγηση των τεχνικών για την αναγνώριση προσώπων. Αναλυτικότερα η συνεισφορά της παρούσας διπλωματικής εργασίας συνοψίζεται ως εξής:

α. Θα εξεταστούν συστήματα βιντεοεπιτήρησης και τύποι αυτών που χρησιμοποιούνται σήμερα.

β. Θα προσδιοριστούν τα προβλήματα τους πάνω σε θέματα προστασίας δεδομένων προσωπικού χαρακτήρα και επεξεργασίας μεμονωμένων δεδομένων εικόνας.

---

γ. Θα μελετηθούν έπειτα τεχνικές επεξεργασίας εικόνας, καθώς και σύγχρονοι και αποδοτικοί αλγόριθμοι, μέσω των οποίων αυτές μπορούν να εφαρμοστούν σε πολλά συστήματα βιντεοεπιτήρησης.

δ. Θα προσδιοριστούν οι τεχνικές λεπτομέρειες για την υλοποίηση της λειτουργίας κάλυψης στοιχείων προσώπου, ανίχνευσης κινούμενου σώματος και ανίχνευσης μηκών κύματος οπτικού φάσματος.

ε. Θα γίνει χρήση αρκετών πηγών πάνω σε παρεμφερή θέματα για την άντληση περισσότερων τεχνικών λεπτομερειών.

Τέλος θα αναλυθούν τα πλεονεκτήματα και τα μειονεκτήματα των χαρακτηριστικών των παραπάνω τεχνικών λεπτομερειών.

## **1.4 Οργάνωση κειμένου**

Η παρούσα διπλωματική εργασία αναπτύσσεται συνολικά σε έξι κεφάλαια σε κάθε ένα εκ των οποίων παρουσιάζονται αναλυτικά τα παρακάτω θέματα:

α. Στο κεφάλαιο 2 αναπτύσσεται το βασικό θεωρητικό υπόβαθρο που απαιτεί η εν λόγω διπλωματική εργασία για την υλοποίηση των εφαρμογών που θα παρουσιαστούν παρακάτω. Συγκεκριμένα περιγράφεται η Γλώσσα Προγραμματισμού Python που χρησιμοποιήθηκε για την υλοποίηση αυτών των εφαρμογών, ο διαχειριστής πακέτων Anaconda καθώς και οι βιβλιοθήκες OpenCV[3] και DLib [8] που χρησιμοποιήθηκαν ως πηγές άντλησης των κάτωθι αλγορίθμων:

➤ Ανίχνευση Προσώπου με Ταξινομητές Χαρακτηριστικών (Haar Cascade) στο OpenCV

➤ Ανίχνευση Προσώπου βασισμένη σε DeepLearning στο OpenCV

➤ Ανίχνευση Προσώπου με Ιστόγραμμα Προσανατολισμένων Κλίσεων (HoG) στο Dlib

➤ Ανίχνευση Προσώπου με Συγκεντρωτικά Νευρωνικά Δίκτυα στο Dlib

- 
- Ανίχνευση Κίνησης με Σύγκριση Πλαισίων (Frames Comparison)
  - Ανίχνευση Μηκών Κύματος Οπτικού Φάσματος

και η περιγραφική ανάλυση αυτών των, καθώς και η πλατφόρμα Anaconda [9] πάνω στην οποία μεταγλωττίστηκαν και εκτελέστηκαν τα προγράμματά μας.

β. Στο κεφάλαιο 3 αναπτύσσονται οι τρεις εφαρμογές που γράφτηκαν για την υλοποίηση της παρούσας διπλωματικής καθώς και η λεπτομερής λειτουργία αυτών.

γ. Τέλος στα κεφάλαια 5 και 6 παρουσιάζονται ο επίλογος και η σχετική βιβλιογραφία αντίστοιχα της παρούσας διπλωματικής εργασίας.

---

Η σελίδα αυτή είναι σκόπιμα λευκή.

---

# 2

## ***Υπόβαθρο***

Στην παρούσα διπλωματική εργασία για την υλοποίηση των τριών εφαρμογών που παρουσιάζονται παρακάτω χρησιμοποιήθηκε σαν κύρια γλώσσα προγραμματισμού η γλώσσα Python, μια διερμηνευόμενη υψηλού επιπέδου γλώσσα προγραμματισμού, με χρήση των βιβλιοθηκών OpenCV και Dlib, βιβλιοθήκες ανοιχτού κώδικα που είναι ελεύθερα διαθέσιμες και είναι γραμμένες σε C/C++, τρέχουν σε Linux, Windows, Mac OSX και παρέχουν διεπαφές με Python καθώς και με άλλες γλώσσες προγραμματισμού όπως Ruby, Matlab κλπ.

Τα προγράμματα δοκιμάστηκαν σε λειτουργικό σύστημα Windows μέσω της πλατφόρμας Anaconda.

Παρακάτω γίνεται μια εισαγωγή στα τεχνικά εργαλεία και ανάλυση των αλγορίθμων που χρησιμοποιήθηκαν στην παρούσα εργασία προκειμένου να κατανοήσουμε σε μεγαλύτερο βαθμό τον τρόπο λειτουργίας των εφαρμογών που θα παρουσιαστούν σε επόμενο κεφάλαιο.

### ***2.1 Τεχνικά Χαρακτηριστικά Συστημάτων Βιντεοεπιτήρησης***

Κατά την αναζήτηση ενός συστήματος βιντεοεπιτήρησης, παρατηρείται ότι στην αγορά εμφανίζεται μία πληθώρα συστημάτων καταγραφής, με διαφορετικές δυνατότητες και χαρακτηριστικά το καθένα.

---

Οι δύο πιο διαδεδομένες τεχνολογίες είναι τα:

α. NVR (Network Video Recorder)

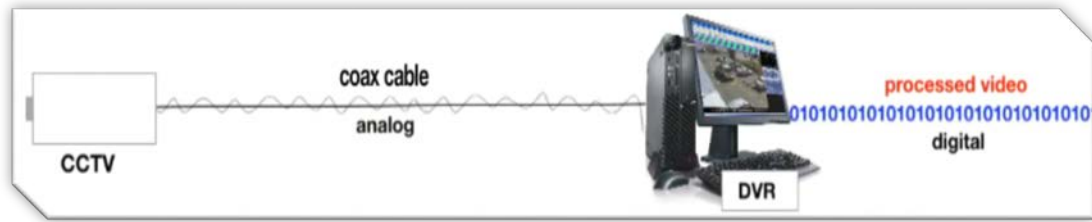
Οι δικτυακές συσκευές εγγραφής βίντεο καταγράφουν μέσω καμερών IP ασύρματα ή με την χρήση καλωδίου δικτύου. Προσφέρουν εικόνα υψηλής ευκρίνειας, καθώς η επεξεργασία της γίνεται τόσο από τις IP κάμερες όσο και από το καταγραφικό, ενώ τα καλώδια ethernet, μεταφέρουν τα δεδομένα με μεγαλύτερη ταχύτητα, άρα και η απεικόνιση είναι λεπτομερέστερη. Οι κάμερες συνδέονται με το NVR, είτε μέσω ενός διανομέα δικτύου είτε μέσω ρούτερ και σε ορισμένες περιπτώσεις απευθείας στην συσκευή εγγραφής. Με τη λήψη βίντεο από τις IP κάμερες γίνεται η μεταφορά στο NVR [10] για αποθήκευση ή για απομακρυσμένη επιτήρηση.



Σχήμα2.1 Network Video Recorder

β. DVR (Digital Video Recorder)

Οι ψηφιακές συσκευές εγγραφής βίντεο καταγράφουν σε ψηφιακή μορφή και αποθηκεύουν τα αρχεία σε δίσκο, μονάδα USB, κ.α. Η εικόνα μεταφέρεται από τις κάμερες μέσω αναλογικών καλωδίων στο DVR [10], όπου γίνεται και η κωδικοποίησή της. Πλέον κυκλοφορούν στην αγορά DVR νέας τεχνολογίας, που συναγωνίζονται ισάξια σε ποιότητα τα συστήματα NVR.



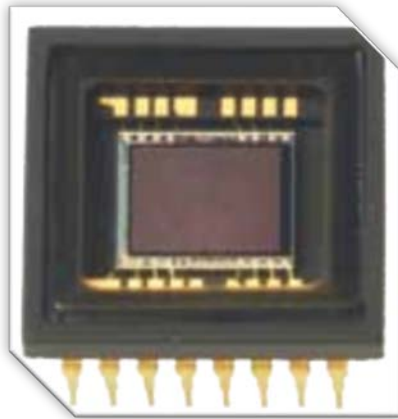
Σχήμα 2.2 Digital Video Recorder

Η βασική διαφορά επομένως των δύο συστημάτων καταγραφής που αναφέρονται παραπάνω, είναι ο τρόπος σύνδεσής τους με τις κάμερες και η επεξεργασία του οπτικού υλικού που λαμβάνουν από τις κάμερες. Τα NVR λειτουργούν και λαμβάνουν οπτικό υλικό δικτυακά, ενώ τα DVR ψηφιακά.

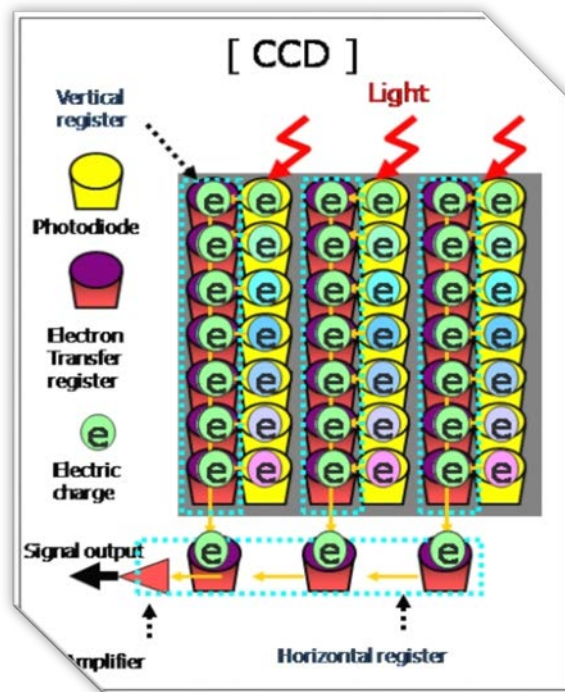
Όλα τα συστήματα βιντεοεπιτήρησης χρησιμοποιούν δύο αισθητήρες εικόνας που τοποθετούνται στις κάμερες και μετατρέπουν το φως σε ηλεκτρονικά σήματα. Οι τύποι αυτοί είναι:

α. CCD αισθητήρες

Οι πρώτες ψηφιακές φωτογραφικές μηχανές χρησιμοποίησαν αισθητήρες CCD [10] (Charged Coupling Devices) για να μετατρέψουν τις εικόνες από αναλογικά σήματα φωτός σε ψηφιακά pixels . Οι αισθητήρες CCD φτιάχνονται μέσα από μια ειδική διαδικασία παραγωγής που επιτρέπει αυτή τη μετατροπή να πραγματοποιηθεί στο τσιπ χωρίς παραμόρφωση. Μέσα από αυτή η διαδικασία δημιουργούνται αισθητήρες υψηλής ποιότητας που παράγουν εξαιρετικά αποτελέσματα στην εικόνα της κάμερας. Ωστόσο, επειδή απαιτούν ειδική κατασκευή, έχουν μεγαλύτερο κόστος κατασκευής από ότι οι νεότεροι αισθητήρες CMOS.



Σχήμα 2.3 Charged Coupling Devices



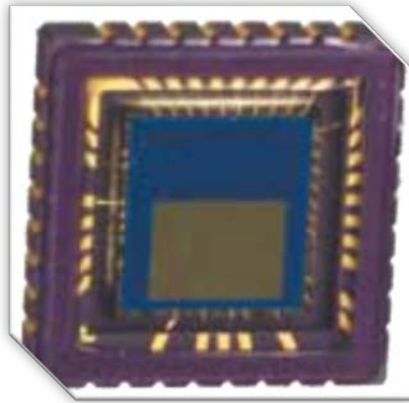
Σχήμα 2.4 CCD (Charged Coupling Devices)

β. CMOS αισθητήρες

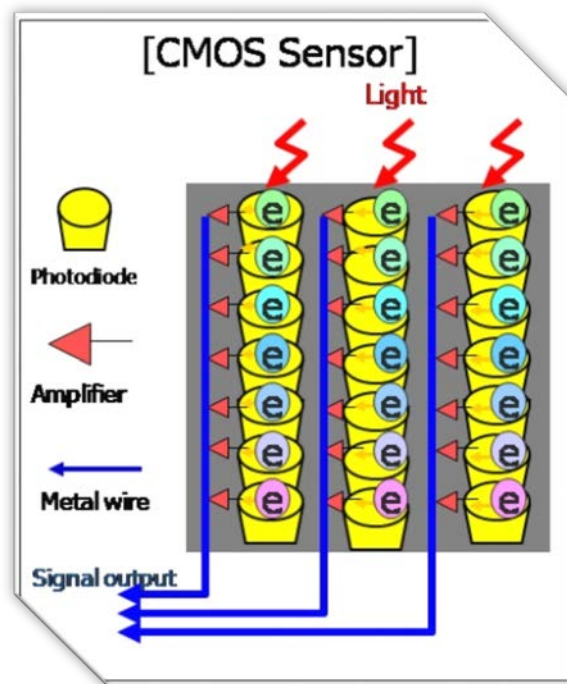
Οι αισθητήρες CMOS [10] (Complementary Metal Oxide Semiconductor) χρησιμοποιούν τρανζίστορ σε κάθε pixel για την μεταφορά φορτίου μέσω των κλασικών καλωδίων. Αυτό προσφέρει ευελιξία, επειδή κάθε pixel αντιμετωπίζεται ξεχωριστά. Η διαδικασία που χρησιμοποιείται για να κατασκευαστεί ένας αισθητήρας CMOS, είναι η ίδια με τη δημιουργία ενός



μικροσίπ. Επειδή είναι πιο εύκολο να παραχθούν, οι αισθητήρες CMOS έχουν χαμηλότερο κόστος παραγωγής απ' ό,τι οι αισθητήρες CCD.

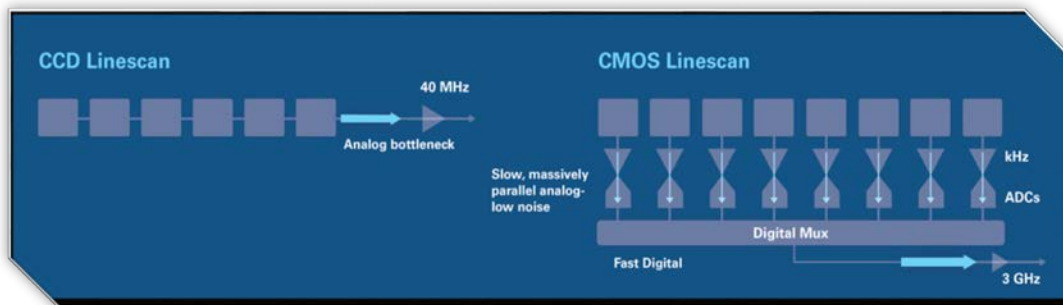


Σχήμα 2.5 Complementary Metal Oxide Semiconductor



Σχήμα 2.6 CMOS (Complementary Metal Oxide Semiconductor)

Η μεγαλύτερη διαφορά τους, πλην του ότι η τεχνολογία CMOS ήρθε μετά από τους αισθητήρες CCD και είναι φθηνότερη η κατασκευή τους, είναι ότι οι αισθητήρες CCD δημιουργούν υψηλής ποιότητας εικόνες με χαμηλό θόρυβο (κόκκους). Οι εικόνες στους αισθητήρες CMOS τείνουν να είναι με υψηλότερο θόρυβο.



Σχήμα 2.7 CCD – CMOS Linescan

Τόσο στα συστήματα τύπου DVR όσο και στα συστήματα NVR, ανεξαρτήτως τύπου αισθητήρα που χρησιμοποιούν, η εικόνα που αποστέλλεται στα καταγραφικά μπορεί να επεξεργαστεί περαιτέρω πριν την αποθήκευσή της, την προβολή της ή ακόμα και την αποστολή της σε άλλα συστήματα αισθητήρων. Η επεξεργασία αυτή μπορεί να πραγματοποιηθεί κεντρικά (κεντρική επεξεργασία) και για τα δύο συστήματα αλλά και τοπικά μόνο για τα συστήματα NVR (καταναμεμμένη επεξεργασία).

## 2.2 Η Γλώσσα Προγραμματισμού Python

Η γλώσσα προγραμματισμού Python είναι μια διερμηνευόμενη, υψηλού επιπέδου γλώσσα με δυναμική σημασιολογία (semantics). Η φιλοσοφία της ενθαρρύνει την αναγνωσιμότητα του κώδικα και έχει μια αρκετά μεγάλη κύρια βιβλιοθήκη (standard library).

Ανάμεσα στα κύρια χαρακτηριστικά της είναι:

- ❖ Εύκολη
  - Εκμάθηση
  - Αναγνωσιμότητα
  - Συντήρηση
- ❖ Γρήγορη Ανάπτυξη Εφαρμογών
- ❖ Διερμηνευόμενη

- 
- ❖ Πολύ υψηλού επιπέδου δομές δεδομένων
  - ❖ Επεκτάσιμη
  - ❖ Ανοικτού Κώδικα
  - ❖ Λειτουργεί σχεδόν παντού
  - ❖ Δεν παρουσιάζει το φαινόμενο των segmentation faults
  - ❖ Αυτόματη διαχείριση μνήμης

Η Python είναι μια γλώσσα προγραμματισμού στην οποία διευκολύνεται η συντήρηση του κώδικα και η επεκτασιμότητα του. Όλα τα παραπάνω συνηγορούν στην δυνατότητα της Python να επιτρέπει την ταχύτερη ανάπτυξη εφαρμογών ειδικά σε σχέση με άλλες γλώσσες χαμηλότερου επιπέδου (πχ C, C++) ενώ λέγεται ότι συνήθως τα προγράμματα σε Python είναι 3 – 5 φορές μικρότερα σε σχέση με τα αντίστοιχα σε Java. Όσο πιο υψηλού επιπέδου μια γλώσσα προγραμματισμού είναι, τόσο πιο κοντά στην σκέψη του ανθρώπου βρίσκεται. Αυτό σημαίνει ότι είναι πιο εύκολο να γραφτούν προγράμματα σε υψηλού επιπέδου γλώσσες (υψηλό επίπεδο αφαίρεσης) και συνήθως λειτουργούν σε περισσότερες πλατφόρμες. Αυτό όμως γίνεται θυσιάζοντας μέρος της ταχύτητας προς την εκτέλεση προγραμμάτων. Στις μέρες μας παρατηρείται μια σταδιακή στροφή από γλώσσες που επικεντρώνονταν στην απόδοση των προγραμμάτων (efficiency), να επικεντρώνουν στην απόδοση του προγραμματιστή (productivity).

Υπάρχουν δυο είδη προγραμμάτων που ασχολούνται με την μετατροπή του προγράμματος από γλώσσα προγραμματισμού που είναι κοντά στον άνθρωπο, σε γλώσσα μηχανής. Αυτά είναι οι διερμηνείς (interpreters) και οι μεταφραστές (compilers). Οι διερμηνείς μετατρέπουν γραμμή προς γραμμή τον πηγαίο κώδικα του προγράμματος μας σε γλώσσα μηχανής και τον εκτελούν άμεσα ενώ οι μεταφραστές πρέπει να μετατρέψουν όλο το πρόγραμμα σε γλώσσα μηχανής και στην συνέχεια αυτό μπορεί να εκτελεστεί. Υπάρχει η ειδική περίπτωση που χρησιμοποιείται από την γλώσσα προγραμματισμού εικονική μηχανή (virtual machine) όπου εκτελείται ο κώδικας (όπως συμβαίνει και με την Python). Πριν μετατραπεί σε γλώσσα μηχανής, που καταλαβαίνει τελικά ο υπολογιστής, μετατρέπεται σε μια ενδιάμεση γλώσσα (bytecode). Η τελική μορφή του κώδικα

---

που έχει μεταφραστεί και μπορεί πλέον να εκτελεστεί, ονομάζεται αντικειμενικός κώδικας (object code). Η κύρια βιβλιοθήκη περιλαμβάνει τα πάντα από σύγχρονη επεξεργασία έως συμπιεσμένα αρχεία. Επειδή ο κώδικας της έχει γραφτεί από πολλούς έξυπνους ανθρώπους, είναι πολύ γρήγορος για τις περισσότερες εφαρμογές που θα χρειαστεί κάποιος.

Οι ευκολίες που παρέχονται επίσης, είναι πολύ σημαντικές καθώς καλύπτουν ένα ευρύ φάσμα πιθανών προβλημάτων που μπορεί να αντιμετωπίσει κανείς, αποφεύγοντας έτσι την ανάγκη για κάποιον να προσπαθεί να ανακαλύψει από την αρχή τον τροχό. Η ίδια η γλώσσα είναι επεκτάσιμη καθώς ένα βασικό σύνολο της γλώσσας αποτελεί τον πυρήνα της, ενώ όλα τα υπόλοιπα είναι αρθρώματα (modules) που επεκτείνουν την λειτουργικότητα της, γεγονός που σε συνδυασμό με το ότι είναι ανοικτού κώδικα την βοηθάει να μην μένει στάσιμη, αλλά να παρακολουθεί πάντα τις εξελίξεις (παραδείγματα η έκδοση 3 της Python που είναι και το αντικείμενο πραγμάτευσης του παρόντος οδηγού). Το ότι παίζει σχεδόν παντού, δεν αναφέρεται μόνο σε λειτουργικά συστήματα όπου παίζει σε όλες τις κύριες πλατφόρμες (πχ Windows, Linux/Unix, OS/2, Mac, Amiga). Αναφέρεται ακόμα και σε άλλες γλώσσες προγραμματισμού, όπως η Java, όπου μέσω της Python μπορούν να χρησιμοποιηθούν βιβλιοθήκες της Java, του NET για το οποίο υπάρχει η πρόσφατη υλοποίηση της Iron Python από την Microsoft. Ακόμα, μπορούν να γραφτούν σε κώδικα C/C++ και στην συνέχεια να δημιουργηθούν αρθρώματα (modules) μέσω των οποίων, ο τελικός χρήστης του κώδικα δεν θα καταλαβαίνει καμία διαφορά σε σχέση με τον υπόλοιπο κώδικα Python. Επιπρόσθετα, η γλώσσα είναι πλέον ώριμη. Υπάρχει από τα τέλη της δεκαετίας του 1980, και σε αυτή την πορεία του χρόνου, πολλοί την έχουν υιοθετήσει και έχουν δημιουργηθεί πολλές βιβλιοθήκες για αυτή. Επίσης, έχει ξεπεράσει παιδικές ασθένειες και συμπεριλαμβάνει πολλά χαρακτηριστικά όπως αυτά προβλήθηκαν μέσα από τις ανάγκες των χρηστών της.

Στην Python δεν εντοπίζονται τα segmentation faults. Σε αντίστοιχες περιπτώσεις, ο διερμηνευτής της Python ενημερώνει με μια εξαίρεση που πετάει (γίνεται throw) και πλέον γίνεται γνωστό σε ποια γραμμή υπάρχει το πρόβλημα ώστε να αντιμετωπιστεί. Η αυτόματη διαχείριση μνήμης σημαίνει πως δεν

---

χρειάζεται να εξεταστεί πλέον για το πότε θα ελευθερωθεί η μνήμη που δεσμεύεται όταν παράγονται αντικείμενα. Επίσης, η Python αντιλαμβάνεται πότε το ίδιο αντικείμενο αναφέρεται πάνω από μια φορές, και έτσι δεν το αποθηκεύει στη μνήμη αν δεν χρειάζεται. Η τεχνική αυτή ονομάζεται μέτρηση αναφορών (reference counting).

Οι τρεις εφαρμογές που παρουσιάζονται στην παρούσα διπλωματική εργασία χρησιμοποιούν την πλατφόρμα εργασίας Anaconda μέσω της οποίας εκτελούνται τα προγράμματα που είναι γραμμένα σε Python. Το Anaconda είναι μια πλατφόρμα για την επιστήμη των δεδομένων και την εκμάθηση μηχανών για τις γλώσσες προγραμματισμού Python και R. Έχει σχεδιαστεί για να κάνει τη διαδικασία δημιουργίας και διανομής έργων απλή, σταθερή και αναπαραγωγίσιμη μεταξύ των συστημάτων και είναι διαθέσιμη σε Linux, Windows και OSX. Το Anaconda αποτελεί μια πλατφόρμα βασισμένη στην Python που περιλαμβάνει μεγάλα πακέτα της επιστήμης δεδομένων, συμπεριλαμβανομένων των pandas, scikit-learn, SciPy, NumPy και της πλατφόρμας μηχανικής μάθησης της Google, TensorFlow. Έρχεται σε συνδυασμό με το conda (ένα εργαλείο εγκατάστασης pip), το Navigator Anaconda για GUI και το spyder για ένα IDE.

### **2.3 Η Βιβλιοθήκη OpenCV**

Η OpenCV είναι μια βιβλιοθήκη C/C++ με λειτουργίες σχετικές με την όραση στον υπολογιστή σε πραγματικό χρόνο και είναι συμβατή με C++, C, Python, Java κ.α.

Η OpenCV είναι μία βιβλιοθήκη ανοιχτού κώδικα που είναι ελεύθερα διαθέσιμη. Τρέχει σε Linux, Windows και Mac OSX. Η OpenCV αναπτύχθηκε από ερευνητική πρωτοβουλία της Intel Research για την προώθηση των εφαρμογών με υψηλές απαιτήσεις CPU. Υπάρχει ενεργή ανάπτυξη για διεπαφές με Python, Ruby, Matlab και άλλες γλώσσες. Η OpenCV σχεδιάστηκε για υπολογιστική αποτελεσματικότητα και με ιδιαίτερη έμφαση στις εφαρμογές πραγματικού χρόνου.

---

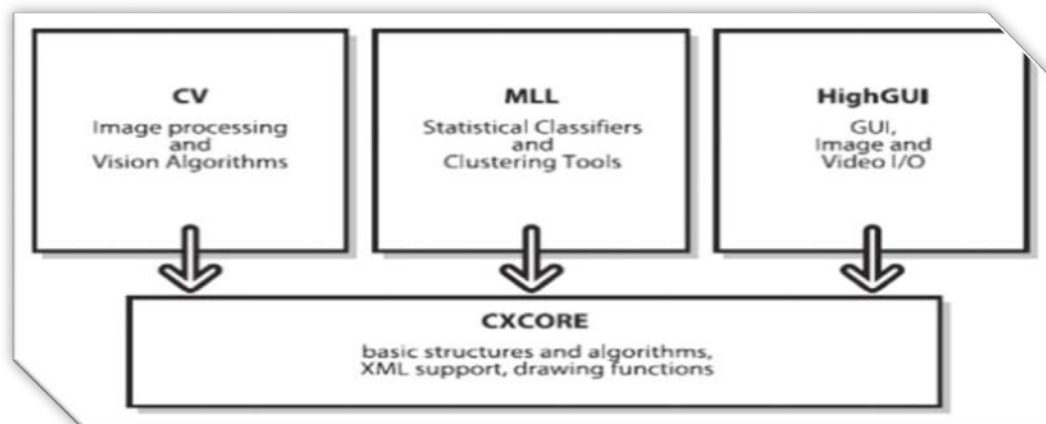
Είναι γραμμένη σε βελτιστοποιημένη C και μπορεί να επωφεληθεί από πολυπύρηνους επεξεργαστές. Ένας από τους στόχους της OpenCV είναι να παρέχει στο χρήστη απλές στη χρήση υποδομές στην τεχνητή όραση που βοηθάει τους ανθρώπους να χτίσουν γρήγορα, αρκετά εξελιγμένες εφαρμογές όρασης. Η βιβλιοθήκη περιέχει πάνω από 500 συναρτήσεις που εκτείνονται σε πολλούς τομείς της όρασης, συμπεριλαμβανομένων της εργοστασιακής επιθεώρησης προϊόντων, ιατρικής απεικόνισης, ασφάλειας, διεπαφή με χρήστη, βαθμονόμηση κάμερας, στερεοφωνική όραση και ρομποτική.

Επειδή η τεχνητή όραση και η μηχανική μάθηση συχνά πηγαίνουν χέρι-χέρι, η OpenCV επίσης περιλαμβάνει μια πλήρη, γενικού σκοπού Βιβλιοθήκη Μηχανικής Μάθησης (MLL). Αυτή η υποβιβλιοθήκη επικεντρώνεται στη στατιστική αναγνώριση προτύπων και ομαδοποίηση. Η MLL είναι ιδιαίτερα χρήσιμη για οπτικές διεργασίες που είναι ο πυρήνας της αποστολής της OpenCV, αλλά μπορούν να χρησιμοποιηθούν για οποιοδήποτε πρόβλημα μηχανική εκπαίδευσης.

Οι περισσότεροι επιστήμονες της πληροφορικής και προγραμματιστές γνωρίζουν κάποια πτυχή του ρόλου που παίζει η τεχνητή όραση. Αλλά λίγοι άνθρωποι γνωρίζουν όλους τους τρόπους με τους οποίους χρησιμοποιείται η τεχνητή όραση. Για παράδειγμα, οι περισσότεροι άνθρωποι κάπως γνωρίζουν για τη χρήση του σε επιτήρηση, και πολλοί επίσης γνωρίζουν ότι χρησιμοποιείται όλο και περισσότερο για εικόνες και βίντεο στο διαδίκτυο. Λίγοι έχουν δει τη χρήση της τεχνητής όρασης σε παιχνίδια. Επίσης λίγοι άνθρωποι συνειδητοποιούν ότι οι περισσότερες εναέριες και street-map εικόνες (όπως το Google' s Street της Google) κάνουν χρήση της προσαρμογής κάμερας και τεχνικές συρραφής εικόνων. Μερικοί γνωρίζουν εξειδικευμένες εφαρμογές για παρακολούθηση ασφαλείας, για επανδρωμένα ιπτάμενα οχήματα ή για αναλύσεις βιοϊατρικής.

Η OpenCV σε γενικές γραμμές διαρθρώνεται σε πέντε κύρια μέρη, τέσσερα από τα οποία παρουσιάζονται στο *σχήμα 2.8*.

- ❖ Η CV περιέχει την βασική επεξεργασία εικόνας και τους αλγόριθμους τεχνητής όρασης υψηλότερου επιπέδου.
- ❖ Η ML είναι η βιβλιοθήκη μάθησης μηχανής, που περιλαμβάνει πολλούς στατιστικούς ταξινομητές και εργαλεία ομαδοποίησης (clustering).
- ❖ Η HighGUI περιέχει ρουτίνες εισόδου/εξόδου και τις συναρτήσεις για την αποθήκευση και φόρτωση βίντεο και εικόνων και η CXCore περιέχει τις βασικές δομές δεδομένων και το περιεχόμενο.
- ❖ Επίσης υπάρχει η CvAux, η οποία περιέχει περιοχές που δεν χρησιμοποιούνται (ενσωματωμένο HMM αναγνώρισης προσώπου) αλλά και πειραματικούς αλγόριθμους (κατάτμηση φόντου, πρόσοψης).



Σχήμα 2.8 Διάρθρωση OpenCV

Οι τομείς εφαρμογής της OpenCV περιλαμβάνουν:

- ❖ Σύστημα αναγνώρισης προσώπου
- ❖ Αλληλεπίδραση ανθρώπου-υπολογιστή
- ❖ Ρομποτική κίνηση
- ❖ Αναγνώριση αντικειμένων
- ❖ Κατασκευή από κίνηση
- ❖ Παρακολούθηση κίνησης

- 
- ❖ Εργαλεία δισδιάστατων και τρισδιάστατων χαρακτηριστικών
  - ❖ Στερεοσκοπική όραση: αντίληψη του βάθους από δύο κάμερες

Για την υποστήριξη μερικών εφαρμογών, η OpenCV περιλαμβάνει μια στατική βιβλιοθήκη μηχανικής μάθησης που περιλαμβάνει:

- ❖ Ενίσχυση
- ❖ Δέντρα απόφασης
- ❖ Αλγόριθμος μέγιστης πρόβλεψης
- ❖ Αλγόριθμος k-κοντινότερων γειτόνων
- ❖ Ταξινομητής Bayes
- ❖ Τεχνητά νευρωνικά δίκτυα
- ❖ Random Forest
- ❖ Μηχανές Διανυσμάτων Στήριξης

## **2.4 Η Βιβλιοθήκη DLib**

Η DLib είναι μια βιβλιοθήκη / εργαλειοθήκη C ++ που περιέχει αλγόριθμους μηχανικής μάθησης, συμπεριλαμβανομένης της όρασης υπολογιστή και διαθέτει Python API.

Η DLib είναι ένα σύγχρονο εργαλείο C ++ που περιέχει αλγόριθμους μηχανικής μάθησης και εργαλεία για τη δημιουργία σύνθετου λογισμικού στην C ++ για την επίλυση πραγματικών προβλημάτων στον κόσμο. Χρησιμοποιείται τόσο στον ιδιωτικό κλάδο όσο και στον ακαδημαϊκό χώρο σε ένα ευρύ φάσμα τομέων, όπως η ρομποτική, οι ενσωματωμένες συσκευές, τα κινητά τηλέφωνα και τα μεγάλα υπολογιστικά περιβάλλοντα υψηλής απόδοσης. Η αδειοδότηση ανοικτού κώδικα της DLib επιτρέπει τη χρήση της σε οποιαδήποτε εφαρμογή, δωρεάν.



---

## 2.5 Ο Διαχειριστής Πακέτων *Anaconda*

Ο διαχειριστής πακέτων *Anaconda* είναι μια διανομή ελεύθερης και ανοικτής πηγής των γλωσσών προγραμματισμού *Python* και *R* για την επιστημονική πληροφορική (επιστήμη των δεδομένων, εφαρμογές μηχανικής μάθησης, επεξεργασία δεδομένων μεγάλης κλίμακας, προγνωστικά αναλύσεις κλπ.), με στόχο την απλοποίηση της διαχείρισης πακέτων ανάπτυξης. Οι εκδόσεις των πακέτων διαχειρίζονται το σύστημα διαχείρισης πακέτων *conda*. Η κατανομή *Anaconda* περιλαμβάνει πακέτα δεδομένων επιστήμης κατάλληλα για *Windows*, *Linux* και *MacOS*.

## 2.6 Τεχνικές Ανίχνευσης Αντικειμένων

Στην ενότητα αυτή, θα εξεταστούν οι βασικές τεχνικές ανίχνευσης προσώπου, συμπεριλαμβανομένων των *HaarCascade*, του *DeepLearning* στο *OpenCV*, των παραθύρων *HOG* και του *DeepLearningCNNs*. Επίσης θα αναφερθούν οι αλγόριθμοι ανίχνευσης κινούμενου σώματος και αναγνώρισης οπτικού φάσματος.

Τόσο η *OpenCV* όσο και η *Dlib* χρησιμοποιούνται για κάθε είδους ανάλυση εικόνων και βίντεο, όπως:

- ❖ αναγνώριση και ανίχνευση προσώπου,
- ❖ ανάγνωση πινακίδων,
- ❖ επεξεργασία φωτογραφιών,
- ❖ προηγμένη ρομποτική όραση,
- ❖ οπτική αναγνώριση χαρακτήρων και πολλά άλλα.

Όπως αναφέρθηκε παραπάνω η παρούσα διπλωματική εργασία εστιάζει σε τρεις εφαρμογές εξ αυτών, ανίχνευσης προσώπου, κίνησης και μηκών κύματος οπτικού φάσματος.

---

❖ **Απαιτήσεις:**

Η χρήση της OpenCV, ως κύρια βιβλιοθήκη των εφαρμογών που θα αναλυθούν παρακάτω, ορισμένες από τις οποίες θα απαιτήσουν επίσης τη χρήση της βιβλιοθήκης DLib.

Η εγκατάσταση των παραπάνω βιβλιοθηκών καθώς και πρόσθετων πακέτων όπως NumPy, Imutils και Pandas μπορεί να γίνει μέσω, ενός διαχειριστή πακέτων, του pip. Το pip είναι ένα εργαλείο που επιτρέπει να εγκατασταθεί και να διαχειριστεί πρόσθετες βιβλιοθήκες και εξαρτήσεις που δεν διανέμονται ως μέρος της συνηθισμένης βιβλιοθήκης. Το πρόγραμμα εγκατάστασης Python εγκαθιστά μαζί και τον διαχειριστή πακέτων pip, οπότε είναι έτοιμο για περαιτέρω χρήση.

Μετά την εγκατάσταση της Python εγκαθίσταται μέσω του διαχειριστή πακέτων την OpenCV και την DLib μέσω των παρακάτω εντολών.

<code>pip install opencv-python</code>
--

<code>pip install dlib</code>
-------------------------------

Έπειτα γίνεται εγκατάσταση των παρακάτω χρήσιμων πακέτων:

➤ Το NumPy είναι η βασική βιβλιοθήκη για την επιστήμη της πληροφορικής στη Python. Παρέχει αντικείμενα πολυδιάστατης συστοιχίας υψηλής απόδοσης και εργαλεία για την επεξεργασία αυτών.

➤ Το Imutils είναι ένα πακέτο που περιλαμβάνει μια σειρά λειτουργιών διευκόλυνσης του OpenCV+ που εκτελούν βασικές εργασίες όπως μετάφραση, περιστροφή, αλλαγή μεγέθους και σκελετοποίησης.

➤ Το Pandas είναι μια βιβλιοθήκη Python ανοικτού κώδικα, με άδεια BSD, που παρέχει δομές δεδομένων υψηλής απόδοσης και εύκολης χρήσης

---

και εργαλεία ανάλυσης δεδομένων για τη γλώσσα προγραμματισμού Python. Η Python με το Pandas χρησιμοποιείται σε ένα ευρύ φάσμα τομέων, συμπεριλαμβανομένων των ακαδημαϊκών και εμπορικών τομέων, των οικονομικών, των στατιστικών, των αναλυτικών στοιχείων κλπ.

pipinstallnumpy
pipinstallimutils
pipinstallpandas

### **2.6.1 Ανίχνευση Προσώπου με Ταξινομητές Χαρακτηριστικών**

#### **(HaarCascade) στο OpenCV**

Ο ανιχνευτής προσώπων HaarCascade ήταν ο τελευταίος τεχνολογίας στην ανίχνευση προσώπων για πολλά χρόνια από το 2001, όταν εισήχθη από τη Viola και τον Jones. Έχουν σημειωθεί πολλές βελτιώσεις τα τελευταία χρόνια. Το OpenCV έχει πολλά μοντέλα (.xml) βασισμένα στο Haar τα οποία διατίθενται ελεύθερα στο διαδίκτυο.

```
Python
faceCascade = cv2.CascadeClassifier('./haarcascade_frontalface_default.xml')
faces = faceCascade.detectMultiScale(frameGray)
for face in faces:
    x1, y1, w, h = face
    x2 = x1 + w
    y2 = y1 + h
```

Το παραπάνω απόσπασμα κώδικα φορτώνει το αρχείο μοντέλου haar και το εφαρμόζει σε μια εικόνα κλίμακας του γκρι. Η έξοδος είναι μια λίστα που περιέχει τα ανιχνευμένα πρόσωπα. Κάθε μέλος της λίστας είναι και πάλι μια λίστα με 4

---

στοιχεία που υποδεικνύουν τις συντεταγμένες (x, y) της πάνω αριστερής γωνίας και το πλάτος και το ύψος της ανιχνευμένης όψης.

Ο ταξινομητής Cascade, ή συγκεκριμένα ο καταρράκτης των ενισχυμένων ταξινομητών που εργάζονται με χαρακτηριστικά όπως το haar, είναι μια ειδική περίπτωση της μάθησης του συνόλου, που ονομάζεται ενίσχυση. Συνήθως βασίζεται σε ταξινομητές Adaboost (και σε άλλα μοντέλα όπως Real Adaboost [12], GentleAdaboost ή Logitboost). Οι ταξινομητές Cascade εκπαιδεύονται σε μερικές εκατοντάδες εικόνες δείγματος εικόνας που περιέχουν το αντικείμενο που θέλουμε να εντοπίσουμε και άλλες εικόνες που δεν περιέχουν αυτές τις εικόνες.

Η ανίχνευση προσώπου σε με εικόνα γίνεται με έναν αλγόριθμο, που ονομάζεται πλαίσιο ανίχνευσης αντικειμένων Viola-Jones, που περιλαμβάνει όλα τα βήματα που απαιτούνται για την ανίχνευση ζωντανών προσώπων:

- ❖ Χαρακτηριστικά Προερχόμενα από Μοντέλα Haar (Haar Feature Selection, features derived from Haar wavelets)
- ❖ Δημιουργία Ολοκληρωματικής Εικόνας (Createintegralimage)
- ❖ Εκπαίδευση Ταξινομητών (AdaboostTraining)
- ❖ Διαδοχικοί Ταξινομητές (CascadingClassifiers)

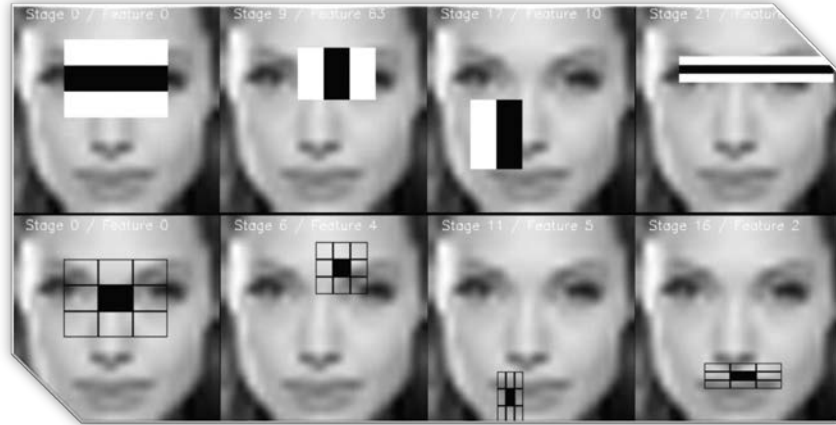
### 1. Χαρακτηριστικά Προερχόμενα από Μοντέλα Haar

Υπάρχουν μερικά κοινά χαρακτηριστικά που εντοπίζονται στα πιο κοινά ανθρώπινα πρόσωπα:

- μια περιοχή μαύρου ματιού σε σύγκριση με τα άνω μάγουλα
- μια φωτεινή περιοχή γέφυρας μύτης σε σύγκριση με τα μάτια
- κάποια συγκεκριμένη θέση των ματιών, του στόματος, της μύτης

κ.λπ.

Τα χαρακτηριστικά γνωρίσματα ονομάζονται Haar χαρακτηριστικά. Η διαδικασία εξαγωγής χαρακτηριστικών θα έχει την μορφή όπως αυτή φαίνεται στο *σχήμα 2.9*.



Σχήμα 2.9 Διαδικασία Εξαγωγής Χαρακτηριστικών

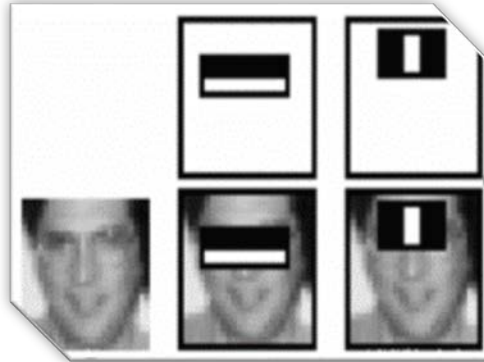
Σε αυτό το παράδειγμα, το πρώτο χαρακτηριστικό μετρά τη διαφορά έντασης μεταξύ της περιοχής των ματιών και μιας περιοχής κατά μήκος των άνω μάγουλων. Η τιμή χαρακτηριστικού υπολογίζεται απλά αθροίζοντας τα εικονοστοιχεία στην μαύρη περιοχή και αφαιρώντας τα εικονοστοιχεία στην άσπρη περιοχή.

$$Rectangle\ Feature = \sum (pixels_{black\ area}) - \sum (pixels_{white\ area}) \quad (2.1)$$

Στη συνέχεια, εφαρμόζουμε αυτό το ορθογώνιο ως συνελικτικό πυρήνα, πάνω από ολόκληρη την εικόνα μας. Αυτό εφαρμόζεται για όλες τις δυνατές διαστάσεις και θέσεις κάθε πυρήνα. Μια απλή εικόνα 24 x 24 θα οδηγήσει συνήθως σε πάνω από 160.000 χαρακτηριστικά, καθένα από τα οποία είναι ένα άθροισμα/αφαίρεση των τιμών των εικονοστοιχείων. Θα ήταν υπολογιστικά αδύνατο για ανίχνευση ζωντανών προσώπων. Η διαδικασία όμως επιτυγχάνεται μέσω των παρακάτω τεχνικών:

➤ Η 1<sup>η</sup> τεχνική βασίζεται στην παρατήρηση ότι εφόσον η καλή περιοχή έχει αναγνωριστεί από ένα ορθογώνιο, είναι άχρηστο να “τρέχουμε” το παράθυρο σε μια εντελώς διαφορετική περιοχή της εικόνας. Αυτό μπορεί να επιτευχθεί από την τεχνική Adaboost που θα αναλύσουμε παρακάτω.

➤ Η 2<sup>η</sup> τεχνική υπολογίζει τα χαρακτηριστικά ορθογωνίου χρησιμοποιώντας την ενσωματωμένη αρχική εικόνα, η οποία είναι πιο γρήγορη.



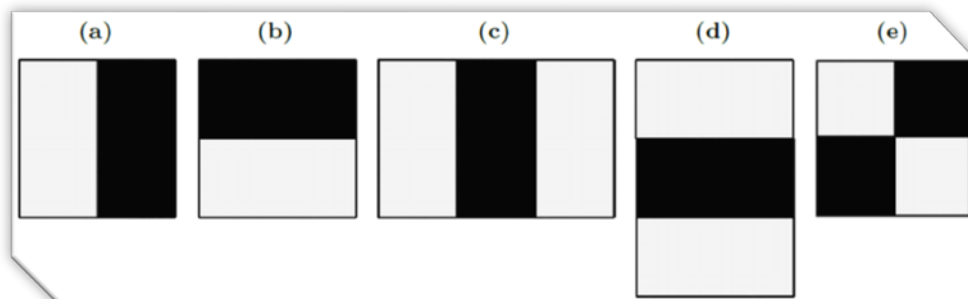
Σχήμα 2.10 Διαδικασία Εξαγωγής Χαρακτηριστικών

Υπάρχουν διάφοροι τύποι ορθογωνίων που μπορούν να εφαρμοστούν για την εξαγωγή χαρακτηριστικών Haar:

➤ Το χαρακτηριστικό δύο ορθογωνίων είναι η διαφορά μεταξύ του αθροίσματος των εικονοστοιχείων μέσα σε δύο ορθογώνιες περιοχές, που χρησιμοποιούνται κυρίως για την ανίχνευση των άκρων (a, b)

➤ Το χαρακτηριστικό τριών ορθογωνίων υπολογίζει το άθροισμα εντός δύο εξωτερικών ορθογωνίων που αφαιρούνται από το άθροισμα σε ένα κεντρικό ορθογώνιο που χρησιμοποιείται κυρίως για την ανίχνευση γραμμών (c, d)

➤ Το χαρακτηριστικό τεσσάρων ορθογωνίων υπολογίζει τη διαφορά μεταξύ διαγώνιων ζευγών ορθογωνίου (e)



Σχήμα 2.11 Τύποι Ορθογωνίων Χαρακτηριστικών Haar

---

Τώρα που έχουν επιλεχθεί τα χαρακτηριστικά, στη συνέχεια εφαρμόζονται στο σετ εκπαιδευτικών εικόνων χρησιμοποιώντας την ταξινόμηση Adaboost, η οποία συνδυάζει ένα σύνολο αδύναμων ταξινομητών για να δημιουργήσει ένα ακριβές μοντέλο συνόλου. Με 200 χαρακτηριστικά (αντί για 160.000 αρχικά), επιτυγχάνεται ακρίβεια 95%.

## 2. Δημιουργία Ολοκληρωματικής Εικόνας

Ο υπολογισμός των χαρακτηριστικών του ορθογωνίου σε στυλ συνάθροισης πυρήνα μπορεί να είναι πολύ μεγάλος. Για το λόγο αυτό, οι Viola και Jones, πρότειναν μια ενδιάμεση αναπαράσταση για την εικόνα: την ολοκληρωμένη εικόνα. Ο ρόλος της αναπόσπαστης εικόνας είναι να επιτρέπει σε κάθε ορθογώνιο άθροισμα να υπολογίζεται απλά, χρησιμοποιώντας μόνο τέσσερις τιμές.

Ας υποθέσουμε ότι θέλουμε να προσδιορίσουμε τα χαρακτηριστικά ενός ορθογωνίου σε ένα δεδομένο εικονοστοιχείο με συντεταγμένες  $(x, y)$ . Η ολοκληρωμένη εικόνα του εικονοστοιχείου στο άθροισμα των εικονοστοιχείων πάνω και αριστερά του δεδομένου εικονοστοιχείου θα ήταν:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2.2)$$

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

Όταν υπολογίζεται η ολοκληρωτική εικόνα, υπάρχει μια μορφή επανάληψης που απαιτεί μόνο ένα πέρασμα πάνω από την αρχική εικόνα εφαρμόζοντας τις παρακάτω σχέσεις:

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2.3)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y) \quad (2.4)$$

---

όπου  $s(x, y)$  είναι το συσσωρευτικό άθροισμα σειρών και:

$$s(x - 1) = 0 \quad (2.5)$$

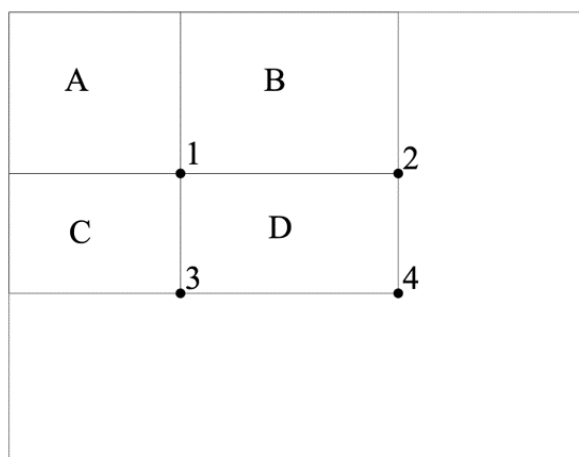
$$ii(-1, y) = 0 \quad (2.6)$$

Το παραπάνω μπορεί να φανεί χρήσιμο αν ληφθεί υπόψη ότι για να υπολογιστεί το άθροισμα των εικονοστοιχείων σε μια περιοχή D θα πρέπει να έχουν οριστεί άλλες 3 περιοχές: A, B και C:

- Η τιμή της ολοκληρωμένης εικόνας στο σημείο 1 είναι το άθροισμα των εικονοστοιχείων στο ορθογώνιο A
- Η τιμή στο σημείο 2 είναι A + B
- Η τιμή στο σημείο 3 είναι A + C
- Η τιμή στο σημείο 4 είναι A + B + C + D

Επομένως, το άθροισμα των εικονοστοιχείων στην περιοχή D μπορεί απλά να υπολογιστεί ως:  $(4 + 1) - (2 + 3)$ .

Επομένως, υπολογίστηκε η τιμή μέσα σε ένα ορθογώνιο χρησιμοποιώντας μόνο 4 αναφορές συστοιχιών.

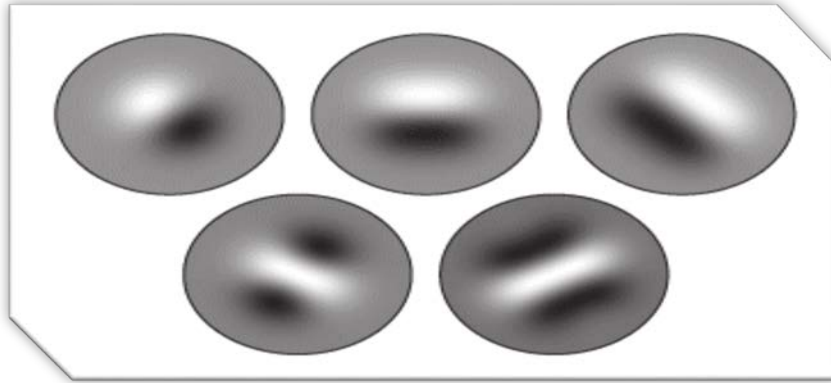


Σχήμα 2.12 Υπολογισμός Εικονοστοιχείων



---

Οι ορθογώνιες περιοχές εικονοστοιχείων απλοποιούν τις πράξεις και επαρκούν για την ανίχνευση προσώπου. Υπάρχουν σήμερα κατευθυνόμενα φίλτρα, όπως φαίνονται στο παρακάτω *σχήμα 2.13*, τα οποία τείνουν να είναι πιο ευέλικτα όταν πρόκειται για πολυπλοκότερα προβλήματα ανίχνευσης.



*Σχήμα 2.13 Κατευθυνόμενα Φίλτρα*

### **3. Εκπαίδευση Ταξινομητών**

Με δεδομένο ένα σύνολο ετικετών εκπαιδευτικών εικόνων (θετικών ή αρνητικών), το Adaboost χρησιμοποιείται για:

- Την επιλογή ενός μικρού συνόλου χαρακτηριστικών
- και για την εκπαίδευση του ταξινομητή

Δεδομένου ότι τα περισσότερα χαρακτηριστικά μεταξύ των 160.000 υποτίθεται ότι δεν έχουν καμία σχέση, ένας ασθενής αλγόριθμος μάθησης γύρω από τον οποίο δημιουργείται ένα μοντέλο ώθησης έχει σχεδιαστεί για να επιλέγει το χαρακτηριστικό ενιαίου ορθογωνίου που χωρίζει τα καλύτερα αρνητικά και θετικά παραδείγματα.

### **4. Διαδοχικοί Ταξινομητές**

Αν και η διαδικασία που περιεγράφηκε παραπάνω είναι αρκετά αποδοτική, παραμένει ένα σημαντικό ζήτημα. Σε μια εικόνα, το μεγαλύτερο μέρος της εικόνας είναι μια μη προσανατολισμένη περιοχή. Το να δοθεί ίση σημασία σε κάθε περιοχή της εικόνας δεν έχει νόημα, αφού πρέπει να γίνει εστίαση, κυρίως

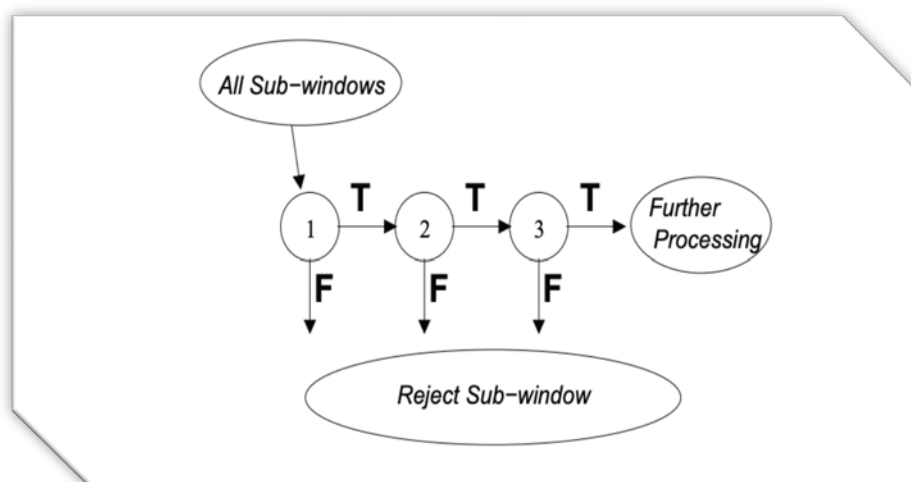
στις περιοχές που είναι πιθανότερο να περιέχουν ένα πρόσωπο. Οι Viola και Jones πέτυχαν ένα αυξημένο ποσοστό ανίχνευσης μειώνοντας ταυτόχρονα τον χρόνο υπολογισμού με τη χρήση των ταξινομητών Cascading [2].

Η βασική ιδέα είναι να απορριφθούν τα δευτερεύοντα παράθυρα που δεν περιέχουν πρόσωπα. Δεδομένου ότι σκοπός είναι να προσδιοριστεί σωστά το πρόσωπο, πρέπει να ελαχιστοποιηθούν τα ψευδή αρνητικά ποσοστά, δηλαδή τα υπο-παράθυρα που δεν περιέχουν ένα πρόσωπο και έχουν αναγνωριστεί ως τέτοια.

Έτσι μια σειρά ταξινομητών εφαρμόζονται σε κάθε υπο-παράθυρο και οι ταξινομητές αυτοί αποτελούν απλά δέντρα απόφασης:

- εάν ο πρώτος ταξινομητής είναι θετικός, θα προχωρήσουμε στο δεύτερο
- εάν ο δεύτερος ταξινομητής είναι θετικός, θα προχωρήσουμε στο τρίτο

Κάθε αρνητικό αποτέλεσμα σε κάποιο σημείο οδηγεί σε απόρριψη του δευτερεύοντος παραθύρου, καθώς πιθανώς περιέχει ένα πρόσωπο. Ο αρχικός ταξινομητής εξαλείφει τα περισσότερα αρνητικά παραδείγματα με χαμηλό υπολογιστικό κόστος και οι ακόλουθοι ταξινομητές εξαλείφουν πρόσθετα αρνητικά αλλά απαιτούν περισσότερη υπολογιστική προσπάθεια.



Σχήμα 2.14 Σχηματική αναπαράσταση ενός καταρράκτη ανίχνευσης

---

Οι ταξινομητές εκπαιδεύονται χρησιμοποιώντας το Adaboost και προσαρμόζουν το κατώτατο όριο για να ελαχιστοποιήσουν το ψευδές ποσοστό. Κατά την εκπαίδευση αυτού του μοντέλου, οι μεταβλητές είναι οι εξής:

- Αριθμός σταδίων ταξινομητή
- Αριθμό χαρακτηριστικών σε κάθε στάδιο
- Κατώφλι κάθε φάσης

Στο OpenCV, όλο αυτό το μοντέλο είναι ήδη προ-εκπαιδευμένο για την ανίχνευση προσώπου.



Σχήμα 2.15 Αρχικό Πλαίσιο



Σχήμα 2.16 Τελικό Πλαίσιο

#### ❖ **Adaptative Boosting**

Το Adaptative Boosting [13] είναι μια μετα-αλγοριθμική μηχανική μάθηση που διατυπώθηκε από τον Yoan Freund και Robert Schapire , οι οποίοι κέρδισαν το 2003 το βραβείο Γκέντελ για την εργασία τους. Μπορεί να χρησιμοποιηθεί σε συνδυασμό με πολλούς άλλους τύπους αλγορίθμων μάθησης για τη βελτίωση της απόδοσης. Η έξοδος των άλλων αλγορίθμων μάθησης («αδύναμοι μαθητές») συνδυάζεται σε ένα σταθμισμένο άθροισμα που αντιπροσωπεύει την τελική έξοδο του ενισχυμένου ταξινομητή.

---

Το AdaBoost είναι προσαρμοστικό με την έννοια ότι οι επόμενοι αδύναμοι μαθητές τροποποιούνται υπέρ αυτών των περιπτώσεων που δεν έχουν ταξινομηθεί από προηγούμενους ταξινομητές. Το AdaBoost είναι ευαίσθητο σε θορυβώδη δεδομένα και ακραίες τιμές. Σε ορισμένα προβλήματα μπορεί να είναι λιγότερο ευαίσθητο στο πρόβλημα της υπερφόρτωσης από άλλους αλγορίθμους μάθησης. Οι μεμονωμένοι μαθητές μπορεί να είναι αδύναμοι, αλλά όσο η απόδοση του καθενός είναι ελαφρώς καλύτερη από την τυχαία εικασία, το τελικό μοντέλο μπορεί να αποδειχθεί ότι συγκλίνει σε έναν ισχυρό μαθητή.

### **2.6.2 Ανίχνευση Προσώπου βασισμένο με *DeepLearning* στο *OpenCV***

Αυτό το μοντέλο συμπεριλήφθηκε στο OpenCV από την έκδοση 3.3 και ύστερα. Βασίζεται στον ανιχνευτή Single-Shot-Multibox και χρησιμοποιεί την αρχιτεκτονική ResNet-10. Το μοντέλο εκπαιδεύτηκε χρησιμοποιώντας εικόνες διαθέσιμες από τον ιστό. Το OpenCV παρέχει 2 μοντέλα για αυτόν τον ανιχνευτή προσώπου.

- ❖ Floating point 16 version of the original caffe implementation (5.4 MB)
- ❖ 8-bit quantized version using Tensorflow (2.7 MB)

Έχουν συμπεριληφθεί και τα δύο μοντέλα στον παρακάτω κώδικα:

Python
<pre>DNN = "TF" if DNN == "CAFFE":     modelFile = "res10_300x300_ssd_iter_140000_fp16.caffemodel"     configFile = "deploy.prototxt"     net = cv2.dnn.readNetFromCaffe(configFile, modelFile) else:     modelFile = "opencv_face_detector_uint8.pb"     configFile = "opencv_face_detector.pbtxt"     net = cv2.dnn.readNetFromTensorflow(modelFile, configFile)</pre>

---

Γίνεται φόρτωση του απαιτούμενου μοντέλου χρησιμοποιώντας τον παραπάνω κώδικα. Αν πρέπει να γίνει χρήση του μοντέλου FloatingPoint του Caffe, τότε εφαρμόζονται τα αρχεία caffemodel και prototxt. Διαφορετικά, θα γίνει χρήση του κβαντισμένου μοντέλου tensorflow. Η βασική διαφορά έγκειται στον τρόπο ανάγνωσης των δικτύων για το Caffe και το Tensorflow.

```
Python
blob = cv2.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123],
False, False)
net.setInput(blob)
detections = net.forward()
bboxes = []
for i in range(detections.shape[2]):
confidence = detections[0, 0, i, 2]
if confidence > conf_threshold:
x1 = int(detections[0, 0, i, 3] * frameWidth)
y1 = int(detections[0, 0, i, 4] * frameHeight)
x2 = int(detections[0, 0, i, 5] * frameWidth)
y2 = int(detections[0, 0, i, 6] * frameHeight)
```

Στον παραπάνω κώδικα, η εικόνα μετατρέπεται σε ένα πλαίσιο και διαβιβάζεται μέσω του δικτύου χρησιμοποιώντας τη λειτουργία forward. Οι ανιχνεύσεις εξόδου είναι ένας πίνακας 4-D, όπου:

- ❖ Η τρίτη διάσταση επαναλαμβάνεται πάνω από τα ανιχνευμένα πρόσωπα. (i είναι ο iterator πάνω στον αριθμό των προσώπων)

- ❖ Η τέταρτη διάσταση περιέχει πληροφορίες σχετικά με το πλαίσιο οριοθέτησης και βαθμολογία για κάθε πρόσωπο. Για παράδειγμα, οι ανιχνεύσεις [0,0,0,2] δίνουν τη βαθμολογία εμπιστοσύνης για την πρώτη όψη και οι ανιχνεύσεις [0,0,0,3: 6] δίνουν το πλαίσιο οριοθέτησης.

Οι συντεταγμένες εξόδου του πλαισίου οριοθέτησης κανονικοποιούνται μεταξύ  $[0,1]$ . Επομένως, οι συντεταγμένες θα πρέπει να πολλαπλασιάζονται με το ύψος και το πλάτος της αρχικής εικόνας για να έχετε το σωστό πλαίσιο οριοθέτησης στην εικόνα.



Σχήμα 2.17 Αρχικό Πλαίσιο



Σχήμα 2.18 Τελικό Πλαίσιο

### 2.6.3 Ανίχνευση Προσώπου με Ιστόγραμμα Προσανατολισμένων

#### Κλίσεων (HoG) στο Dlib

Πρόκειται για ένα ευρέως χρησιμοποιούμενο μοντέλο ανίχνευσης προσώπου, που βασίζεται σε χαρακτηριστικά HoG και SVM. Το μοντέλο είναι χτισμένο από 5 φίλτρα HOG - μπροστά, αριστερά, δεξιά, πίσω, αλλά και περιστρεφόμενη. Το μοντέλο έρχεται ενσωματωμένο στο ίδιο το αρχείο κεφαλίδας.

Το σύνολο δεδομένων που χρησιμοποιείται για την κατάρτιση, αποτελείται από 2825 εικόνες που λαμβάνονται από το σύνολο δεδομένων LFW και επισημαίνονται χειρωνακτικά από τον DavisKing, τον συντάκτη της Dlib. Τα αρχεία αυτά υπάρχουν διαθέσιμα στο διαδίκτυο.

---

```
Python
```

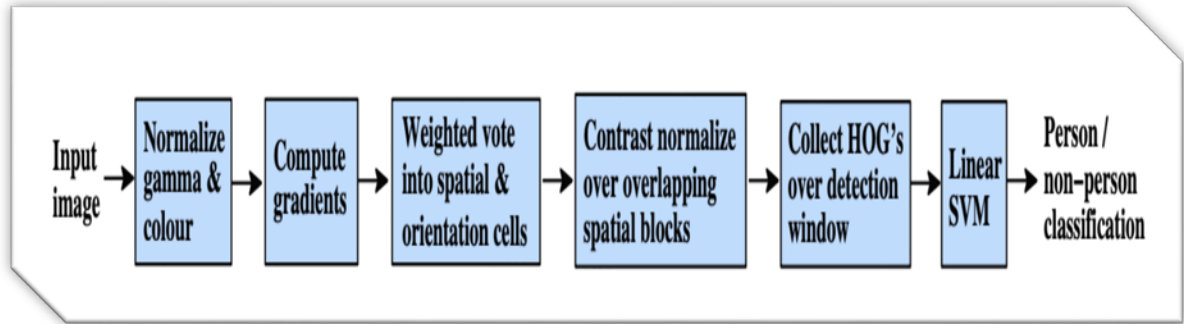
```
hogFaceDetector = dlib.get_frontal_face_detector()
faceRects = hogFaceDetector(frameDlibHogSmall, 0)
for faceRect in faceRects:
    x1 = faceRect.left()
    y1 = faceRect.top()
    x2 = faceRect.right()
    y2 = faceRect.bottom()
```

Στον παραπάνω κώδικα, φορτώνεται πρώτα ο ανιχνευτής προσώπου. Στη συνέχεια, η εικόνα περνάει μέσω του ανιχνευτή. Το δεύτερο επιχείρημα είναι ο αριθμός των φορών που πρέπει να αναβαθμιστεί η εικόνα. Όσο πιο υψηλός είναι, τόσο καλύτερες είναι οι πιθανότητες ανίχνευσης μικρότερων προσώπων. Ωστόσο, η αναβάθμιση της εικόνας θα έχει σημαντικό αντίκτυπο στην ταχύτητα υπολογισμού. Η έξοδος έχει τη μορφή μιας λίστας επιφανειών με τις συντεταγμένες (x, y) των διαγώνιων γωνιών.

Η ιδέα πίσω από το HOG είναι να εξαγάγει τα χαρακτηριστικά σε ένα διάνυσμα και να τον τροφοδοτήσει σε έναν αλγόριθμο ταξινόμησης, που θα αξιολογήσει εάν υπάρχει ένα πρόσωπο (ή οποιοδήποτε αντικείμενο το εκπαιδεύσουμε να αναγνωρίζει στην πραγματικότητα) σε μια περιοχή ή όχι .

Τα χαρακτηριστικά που εξάγονται είναι η κατανομή (ιστογράμματα) των διευθύνσεων κλίσεων (προσανατολισμένες κλίσεις) της εικόνας. Οι κλίσεις είναι συνήθως μεγάλες γύρω από τις άκρες και τις γωνίες και μας επιτρέπουν να εντοπίζουμε αυτές τις περιοχές.

Η διαδικασία για την ανίχνευση ανθρώπινου σώματος και η αλυσίδα ανίχνευσης είναι η ακόλουθη:



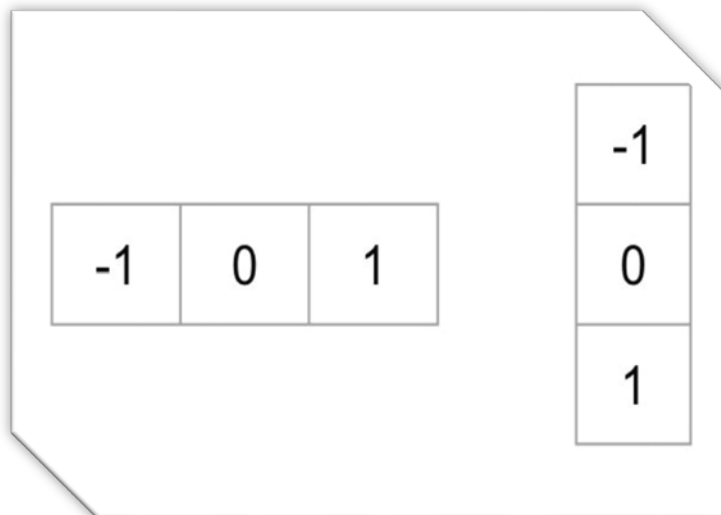
Σχήμα 2.19 Στάδια Διαδικασίας Ανίχνευσης Προσώπου

### 1. Προ επεξεργασία

Πρώτα απ' όλα, οι εικόνες εισόδου πρέπει να έχουν το ίδιο μέγεθος (περικοπή και αναδιάταξη εικόνων). Η τεχνική αυτή απαιτεί αναλογία διαστάσεων 1:2, έτσι ώστε οι διαστάσεις των εικόνων εισόδου να είναι για παράδειγμα 64x128 ή 100x200.

### 2. Υπολογισμός κλίσεων εικόνας

Το πρώτο βήμα είναι να υπολογιστούν οι οριζόντιες και κάθετες κλίσεις της εικόνας, εφαρμόζοντας τους ακόλουθους πυρήνες:



Σχήμα 2.20 Υπολογισμός Κλίσεων



---

Η κλίση μιας εικόνας συνήθως απομακρύνει τις μη απαραίτητες πληροφορίες. Η κλίση της εικόνας που εξετάστηκε παραπάνω μπορεί να βρεθεί με τον παρακάτω τρόπο στην Python:

```
Python
gray = cv2.imread('images/face_detect_test.jpeg', 0)im = np.float32(gray) / 255.0#
Calculate gradient
gx = cv2.Sobel(im, cv2.CV_32F, 1, 0, ksize=1)
gy = cv2.Sobel(im, cv2.CV_32F, 0, 1, ksize=1)
mag, angle = cv2.cartToPolar(gx, gy, angleInDegrees=True)
plt.figure(figsize=(12,8))
plt.imshow(mag)
plt.show()
```

### 3. Υπολογισμός HOG

Χωρίζεται η εικόνα στη συνέχεια σε 8x8 κελιά για να υπάρξει μια συμπαγή αναπαράσταση και να γίνει το HOG πιο ανθεκτικό στο θόρυβο. Στη συνέχεια, υπολογίζεται ένα HOG για καθένα από αυτά τα κύτταρα.

Για να υπολογιστεί η κατεύθυνση κλίσης μέσα σε μια περιοχή χτίζεται απλά ένα ιστόγραμμα ανάμεσα στις 64 τιμές των οδηγιών κλίσης (8x8) και το μέγεθος τους (άλλες 64 τιμές) μέσα σε κάθε περιοχή. Οι κατηγορίες του ιστότοπου αντιστοιχούν στις γωνίες κλίσης, από 0 έως 180 °. Αυτές είναι 9 κατηγορίες συνολικά: 0 °, 20 °, 40 ° ... 160 °.

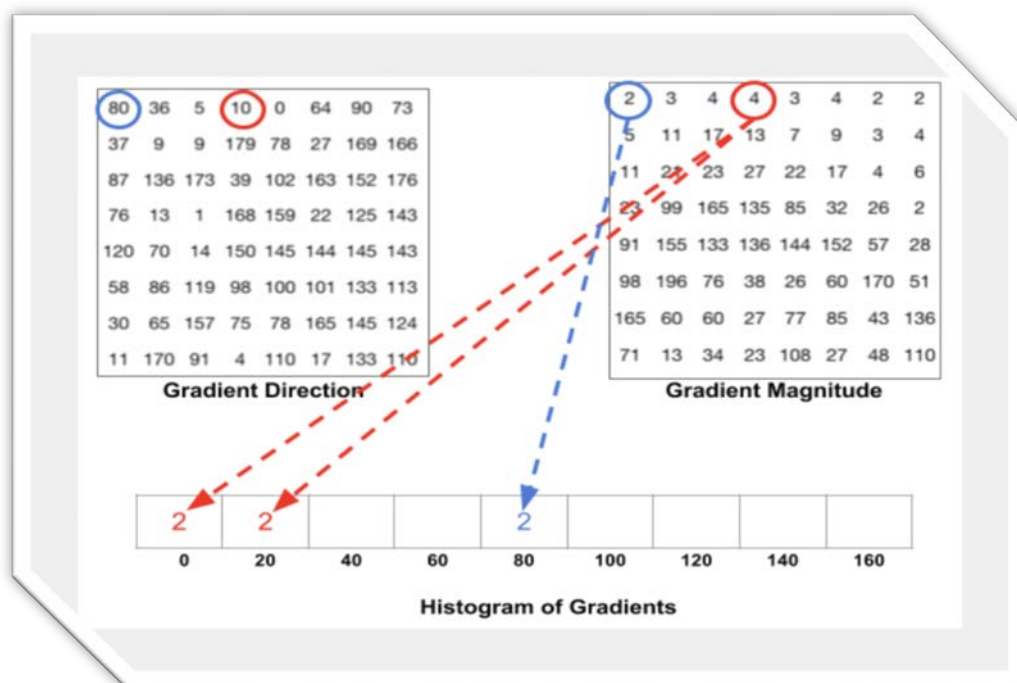
Ο παραπάνω κώδικας δίνει δύο πληροφορίες:

- Κατεύθυνση κλίσης
- Μέγεθος κλίσης

Όταν χτίζεται το HOG, υπάρχουν τρεις υποκατηγορίες:

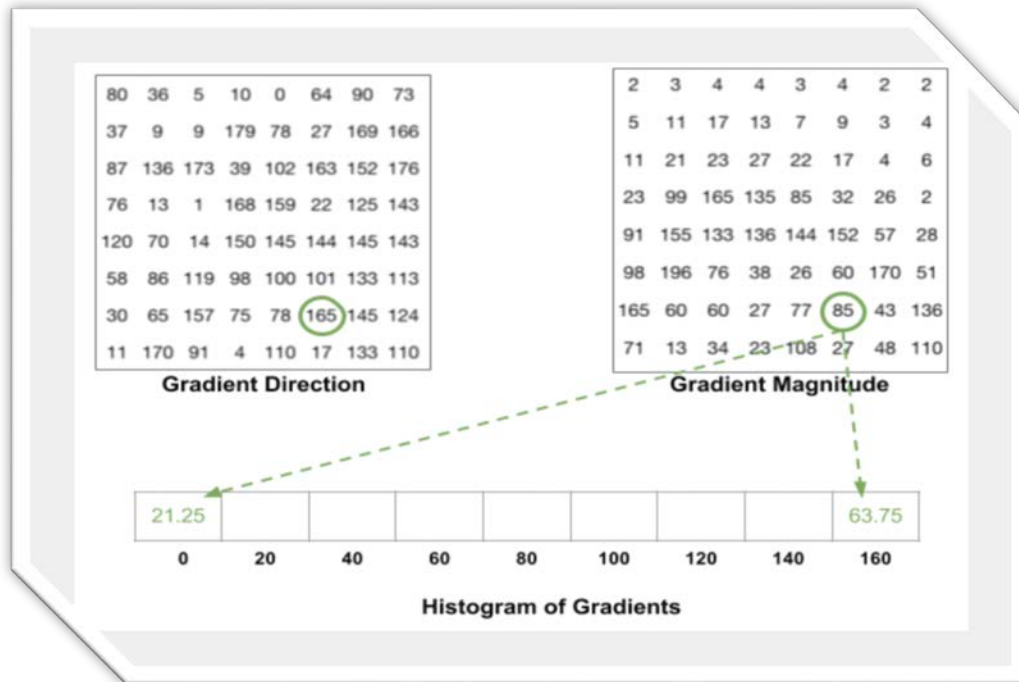
➤ Η γωνία είναι μικρότερη από  $160^\circ$  και όχι στα μισά μεταξύ των δυο τάξεων. Σε αυτή την περίπτωση, η γωνία θα προστεθεί στη σωστή κατηγορία του HOG

➤ Η γωνία είναι μικρότερη από  $160^\circ$  και ακριβώς μεταξύ 2 τάξεων. Σε αυτή την περίπτωση, θεωρείται η ίση συνεισφορά στις 2 πλησιέστερες τάξεις και διαιρείται το μέγεθος στο μισό.

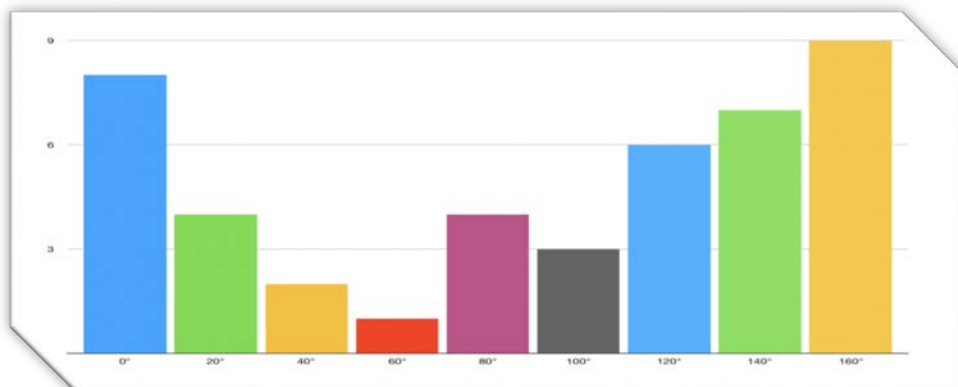


Σχήμα 2.21 Ιστόγραμμα Κλίσεων

➤ Η γωνία είναι μεγαλύτερη από  $160^\circ$ . Σε αυτή την περίπτωση, θεωρείται ότι το εικονοστοιχείο συνέβαλε αναλογικά σε  $160^\circ$  και σε  $0^\circ$ .



Σχήμα 2.22 Ιστόγραμμα Κλίσεων



Πίνακας 2.1 Υπολογισμός HOG

#### 4. Αποκλεισμός της Κανονικοποίησης

Έπειτα, μπορεί να εφαρμοστεί ένα μπλοκ 16x16 προκειμένου να ομαλοποιηθεί η εικόνα και να γίνει αναλλοίωτο για παράδειγμα στο φωτισμό. Αυτό επιτυγχάνεται απλά διαιρώντας κάθε τιμή του HOG μεγέθους 8x8 με το πρότυπο L2 του HOG του μπλοκ 16x16 που το περιέχει, πράγμα που είναι στην πραγματικότητα ένα απλό διάνυσμα μήκους  $9 * 4 = 36$ .

Τέλος, όλοι οι φορείς 36x1 συνενώνονται σε ένα μεγάλο φορέα και έχουμε το χαρακτηριστικό μας διάνυσμα, στο οποίο μπορεί να εκπαιδευτεί ένας soft SVM ταξινομητής ( $C = 0,01$ ).



Σχήμα 2.23 Αρχικό Πλαίσιο



Σχήμα 2.24 Τελικό Πλαίσιο

#### 2.6.4 Ανίχνευση Προσώπου με *DeepLearning* στο *Dlib*

Αυτή η μέθοδος χρησιμοποιεί ανιχνευτή αντικειμένου μέγιστου περιθωρίου (MMOD) με χαρακτηριστικά που βασίζονται στο CNN. Η διαδικασία κατάρτισης για αυτήν τη μέθοδο είναι πολύ απλή και δεν χρειάζεστε μεγάλο όγκο δεδομένων για να εκπαιδεύσουμε έναν προσαρμοσμένο ανιχνευτή αντικειμένων.

Το μοντέλο μπορεί να μεταφορτωθεί από το χώρο αποθήκευσης μοντέλων της *Dlib*. Χρησιμοποιεί ένα σύνολο δεδομένων που έχει επισημανθεί χειροκίνητα από τον συντάκτη της, *DavisKing*, που αποτελείται από εικόνες από διάφορα σύνολα δεδομένων όπως το *ImageNet*, το *PASCAL VOC*, το *VGG*, το *WIDER*, το *Scrub* για το πρόσωπο. Περιέχει 7220 εικόνες.

```
Python
```

```
dnnFaceDetector=dlib.cnn_face_detection_model_v1("./mmod_human_face_detector.dat")
```

---

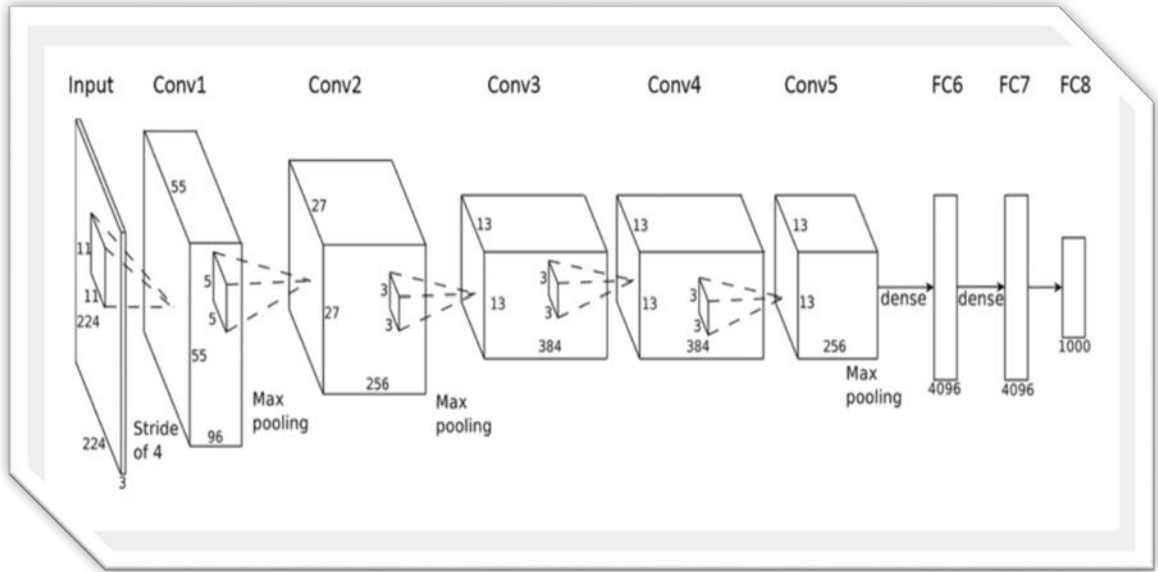
```
faceRects = dnnFaceDetector(frameDlibHogSmall, 0)
for faceRect in faceRects:
    x1 = faceRect.rect.left()
    y1 = faceRect.rect.top()
    x2 = faceRect.rect.right()
    y2 = faceRect.rect.bottom()
```

Ο κώδικας είναι παρόμοιος με τον ανιχνευτή HoG εκτός από το ότι σε αυτή την περίπτωση φορτώνουμε το μοντέλο ανίχνευσης προσώπου CNN. Επίσης, οι συντεταγμένες βρίσκονται μέσα σε ένα αντικείμενο `rect`.

Αυτή η τελευταία μέθοδος βασίζεται σε Convolutional Neural Networks (CNN). Επίσης, εφαρμόζει ένα πρωτόκολλο σχετικά με την ανίχνευση αντικειμένων Max-Margin (MMOD) για βελτιωμένα αποτελέσματα.

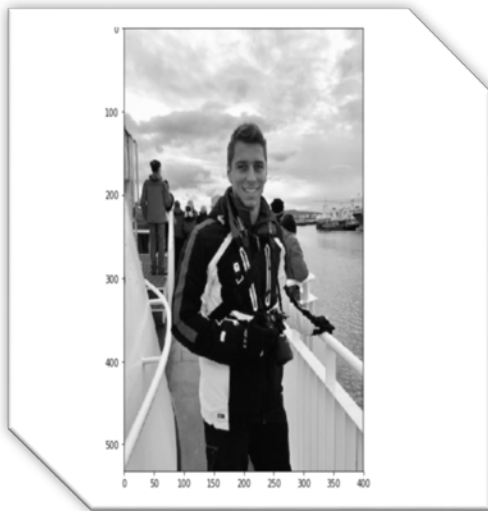
Το συνεργατικό νευρωνικό δίκτυο (CNN) είναι ένα νευρωνικό δίκτυο τροφοδοσίας προς τα εμπρός, το οποίο χρησιμοποιείται κυρίως για όραση στον υπολογιστή. Προσφέρει μια αυτοματοποιημένη προ επεξεργασία εικόνας καθώς και ένα πυκνό τμήμα νευρωνικού δικτύου. Τα CNN είναι ειδικοί τύποι νευρωνικών δικτύων για την επεξεργασία δεδομένων που μοιάζει με πλέγμα. Η αρχιτεκτονική του CNN εμπνέεται από τον οπτικό φλοιό των ζώων.

Σε προηγούμενες προσεγγίσεις, ένα μεγάλο μέρος του έργου ήταν να επιλεγούν τα φίλτρα προκειμένου να δημιουργηθούν τα χαρακτηριστικά για να εξαχθούν όσο το δυνατόν περισσότερες πληροφορίες από την εικόνα. Με την άνοδο της βαθιάς μάθησης και τις μεγαλύτερες ικανότητες υπολογισμού, αυτό το έργο μπορεί πλέον να αυτοματοποιηθεί. Το όνομα των CNNs προέρχεται από το γεγονός ότι περιστρέφεται η αρχική εικόνα με μια σειρά φίλτρων. Η παράμετρος που θα πρέπει να επιλεγεί είναι ο αριθμός των φίλτρων που πρέπει να εφαρμοστούν, καθώς και η διάσταση των φίλτρων αυτών. Η διάσταση του φίλτρου ονομάζεται και μήκος του βήματος. Τυπικές τιμές για το βήμα είναι μεταξύ 2 και 5.



Σχήμα 2.25 Συνεργατικό Νευρωνικό Δίκτυο (CNN)

Η έξοδος του CNN σε αυτή τη συγκεκριμένη περίπτωση είναι μια δυαδική ταξινόμηση, η οποία παίρνει την τιμή 1 αν υπάρχει πρόσωπο, 0 αλλιώς.



Σχήμα 2.26 Αρχικό Πλαίσιο



Σχήμα 2.27 Τελικό Πλαίσιο

---

### 2.6.5 Ανίχνευση Κίνησης με Σύγκριση Frames

Σε περιπτώσεις που πρέπει να μελετηθούν ακολουθίες βίντεο και όχι εικόνες, ο κατάλληλος τρόπος για να εντοπιστεί ένα αντικείμενο που κινείται είναι ο εντοπισμός μετατόπισης θέσης βάση προηγούμενων στιγμιότυπων εικόνων καρέ.

Ο πιο γρήγορος τρόπος για τον διαχωρισμό της κίνησης (segmentation motion) είναι η τεχνική της ανάλυσης διαφορών ανά καρέ. Η συγκεκριμένη τεχνική έχει την δυνατότητα να μπορεί να ξεχωρίζει το κινούμενο προσκήνιο ανεξάρτητα από το περιεχόμενο του παρασκηνίου. Σε κινούμενα περιγράμματα (silhouettes), η εξαγωγή χαρακτηριστικών όπως το πρόσωπο και μέρη του σώματος προέρχονται από τον ορισμό ενός κάτω φλοιού για την συσσωρευμένη διαφορά των καρέ. Μάλιστα οι Yao & Cham στην εργασία τους πρότειναν μια ικανή μέθοδο να εκτιμά τις παραμέτρους κίνησης του ανθρώπινου κεφαλιού που προκύπτουν από μια ακολουθία βίντεο χρησιμοποιώντας μια γραμμική επαναληπτική διαδικασία τριών στρωμάτων (three – layer linear iterative process). Επιπρόσθετα κάνοντας χρήση της τεχνικής της συσσωρευμένης διαφοράς των καρέ μπορούν να εντοπιστούν και χαρακτηριστικά γνωρίσματα, εκτός από την ανίχνευση του προσώπου. Στην εργασία των Crowley και Berard, η ανίχνευση και ο σωστός εντοπισμός των ματιών προϋποθέτει την μέτρηση των οριζόντιων και κάθετων μετατοπίσεων μεταξύ δύο γειτονικών υποψήφιας περιοχών που έχουν ληφθεί από την τεχνική των διαφορών των καρέ.

Μια άλλη τεχνική μέτρησης της κίνησης επιτυγχάνεται μέσω της εκτίμησης της κίνησης των περιγραμμάτων εικόνας. Από την προκύπτουσα σύγκριση με την μέθοδο της διαφοράς των καρέ, προκύπτει ότι τα κινούμενα περιγράμματα παρουσιάζουν πιο αξιόπιστα αποτελέσματα όταν η κίνηση είναι ασήμαντη. Ο Mc Kenna για να μπορέσει να ανιχνεύσει τα κινούμενα όρια των προσώπων χρησιμοποίησε ένα χώρο - χρονικό φίλτρο Gauss. Η διαδικασία περιέχει την συνέλιξη της γκρι εικόνας  $I(x,y)$  με την δεύτερης τάξης παράγωγο του τελεστή ακμών  $m(x,y,t)$  ο οποίος καθορίζεται από το γκαουσιανό φίλτρο  $G(x,y,t)$  και οι οποίες σχέσεις είναι οι εξής:

$$G(x, y, t) = u \left( \frac{a}{\pi} \right)^{\frac{3}{2}} e^{-a(x^2+y^2+u^2t^2)} \quad (2.7)$$

$$m(x, y, t) = - \left( V^2 + \frac{1}{u^2} \frac{d^2}{dt^2} \right) G(x, y, t) \quad (2.8)$$

Όπου  $u$ , ο παράγοντας κλίμακας χρόνου και  $a$  το πλάτος του φίλτρου. Η δεύτερη τάξης παράγωγος του τελεστή ακμών συνελίσσεται με διαδοχικά καρέ από μια ακολουθία εικόνας, μέσω της ακόλουθης σχέσης:

$$S(x, y, t) = m(x, y, t)I(x, y, t) \quad (2.9)$$

Η συνέλιξη  $S(x,y,t)$  περιέχει μηδενικές διελεύσεις που αποδεικνύουν την άμεση ένδειξη κινούμενων ακμών στην γκρι εικόνα  $I(x,y,t)$ . Οι θέσεις των εντοπισμένων μηδενικών διελεύσεων έπειτα συ-στοιχίζονται για να διαπιστωθεί τελικά η τοποθεσία των κινούμενων αντικειμένων.

Γενικά υπάρχουν πιο εξελιγμένες μέθοδοι ανάλυσης κίνησης που έχουν χρησιμοποιηθεί σε πιο πρόσφατες μελέτες ανίχνευσης προσώπου. Αντίθετα με τις μεθόδους που στηρίζονται στην αναγνώριση κινούμενων ακμών και περιοχών, οι συγκεκριμένες μέθοδοι στηρίζονται στην ακριβή εκτίμηση της ταχύτητας φωτεινότητας των περιοχών η οποία ονομάζεται οπτική ροή. Επειδή δε η εκτίμηση βασίζεται σε πρότυπα μικρού εύρους κίνησης, εκδηλώνει μεγάλη ευαισθησία σε μεγάλη κίνηση. Ενδεικτική εργασία όπου χρησιμοποιείται η οπτική ροή είναι η εργασία του Lee που την εφάρμοσε για να μπορέσει να μετρήσει την κίνηση του προσώπου. Ο Lee εκμεταλλεύεται την πληροφορία της κίνησης προκειμένου το κινούμενο πρόσωπο μελέτης να μπορεί να διαχωριστεί σε μια ακολουθία εικόνων. Η οπτική ροή μοντελοποιείται και περιγράφεται από την εξίσωση περιορισμού της ροής εικόνας που δίνεται από την ακόλουθη σχέση:

$$I_x V_x + I_y V_y + I_t = 0 \quad (2.10)$$



---

Όπου  $I_x$ ,  $I_y$  και  $I_t$  είναι οι χωροχρονικές παράγωγοι της εντάσεως εικόνας και  $x$  και  $y$  είναι οι ταχύτητες της εικόνας.

Με την επίλυση της παραπάνω εξίσωσης ως προς τα  $x$  και  $y$  μπορεί να βρεθεί ένα πεδίο οπτικής ροής που περιέχει τις τροχιές των κινούμενων εικονοστοιχείων. Έτσι οι περιοχές που αντιστοιχούν σε διαφορετικές τροχιές κίνησης μπορούν να ταξινομηθούν σε περιοχές κίνησης και μη.

Παρατηρώντας την εξίσωση (2.10) διαπιστώνεται ότι περιλαμβάνει δυο αγνώστους, οπότε θα πρέπει να τεθούν κάποιοι επιπλέον περιορισμοί. Η επιλογή των επιπλέον περιορισμών αποτελεί μια σημαντική εξέταση για την εκτίμηση της οπτικής ροής και έχει προταθεί από πολλούς ερευνητές στην ανάλυση κίνησης.

Ο Lee πρότεινε την εφαρμογή ενός αλγορίθμου γραμμικής ομαδοποίησης που αποτελεί μια προέκταση του αλγορίθμου του Schunk. Ο συγκεκριμένος τροποποιημένος αλγόριθμος έχει το χαρακτηριστικό ότι είναι αρκετά γρήγορος σε σχέση με την αρχική του μορφή. Θέτοντας ένα κατώφλι για τις ταχύτητες των εικόνων, μπορούμε να αποκτήσουμε τις κινούμενες περιοχές του προσώπου. Για την εξαγωγή της περιοχής του προσώπου εφαρμόζεται ένας αλγόριθμος μορφής έλλειψης καθώς οι περιοχές εξαγωγής δεν είναι σε θέση να ορίσουν την περιοχή του προσώπου.

Η Pythοn επιτρέπει την ανίχνευση κίνησης και την αποθήκευση του χρονικού διαστήματος αυτής. Για την υποστήριξη μερικών εφαρμογών, η OpenCV περιλαμβάνει μια στατική βιβλιοθήκη μηχανικής μάθησης που περιλαμβάνει τις βιβλιοθήκες: OpenCV και Pandas.

Οι εικόνες που δημιουργεί η παραπάνω τεχνική για την ανίχνευση κίνησης αναλύονται παρακάτω:

---

❖ Γκρι Πλαίσιο:

Στο Γκρίζο πλαίσιο η εικόνα είναι λίγο θολή και σε κλίμακα του γκρι. Αυτό γίνεται γιατί σε γκρι εικόνες, υπάρχει μόνο μία τιμή έντασης, ενώ σε RGB (Κόκκινο, Πράσινο και Μπλε) οι τρεις εικόνες είναι τρεις τιμές έντασης. Έτσι είναι εύκολο να υπολογιστεί η διαφορά έντασης της κλίμακας του γκρι.



Σχήμα 2.28 Πλαίσιο Γκρι

❖ Πλαίσιο διαφορών:

Το πλαίσιο διαφορών δείχνει τη διαφορά εντάσεων του πρώτου πλαισίου στο τρέχον πλαίσιο.



Σχήμα 2.29 Πλαίσιο Διαφορών

---

❖ Πλαίσιο κατωφλίου:

Εάν η διαφορά έντασης για ένα συγκεκριμένο εικονοστοιχείο είναι μεγαλύτερη από 30 (όπως στο παράδειγμα των σχημάτων) τότε αυτό το εικονοστοιχείο θα είναι λευκό και εάν η διαφορά είναι μικρότερη από 30 ότι το εικονοστοιχείο θα είναι μαύρο.



Σχήμα 2.30 Πλαίσιο Κατωφλίου

❖ Πλαίσιο χρώματος:

Σε αυτό το πλαίσιο μπορούν να φανούν οι έγχρωμες εικόνες σε χρωματικό πλαίσιο μαζί με το πράσινο περίγραμμα γύρω από τα κινούμενα αντικείμενα.

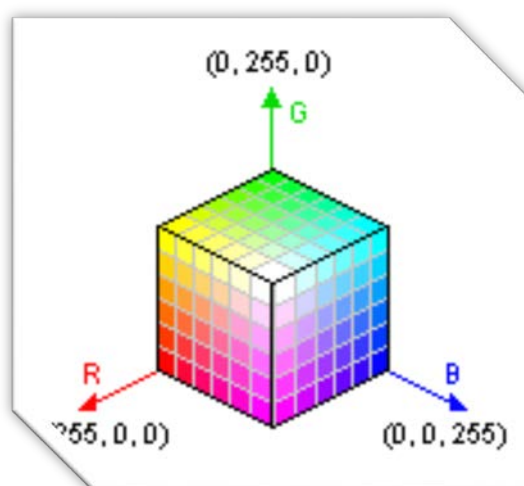


Σχήμα 2.31 Πλαίσιο Χρώματος

---

## 2.6.6 Ανίχνευση Μηκών Κύματος Οπτικού Φάσματος

Το χρώμα αποτελεί το πιο σημαντικό χαρακτηριστικό για την ανίχνευση ενός αντικειμένου. Η χρήση του χρώματος σαν χαρακτηριστικό για την παρακολούθηση αντικειμένων παρουσιάζει αρκετά πλεονεκτήματα. Η επεξεργασία χρώματος είναι αρκετά πιο γρήγορη από την επεξεργασία άλλων χαρακτηριστικών γνωρισμάτων και παρουσιάζει αμεταβλητότητα με τον προσανατολισμό, ανάλογα με τις συνθήκες φωτισμού που επικρατούν. Το χρώμα παρουσιάζει παραπάνω διαστάσεις και έχει παρατηρηθεί ότι σχήματα που παρουσιάζουν γκρι πληροφορία εμφανίζονται αρκετά διαφορετικά σε έναν χρωματικό χώρο. Γενικά υπάρχουν πολλά μοντέλα χρώματος. Το πιο γνωστό χρωματικό μοντέλο είναι η αναπαράσταση σε RGB, όπου το κόκκινο (red), πράσινο (green) και το μπλέ (blue) συνδυάζονται μεταξύ τους για να αναπαράγουν μια ευρεία σειρά χρωμάτων όπως φαίνεται στο *σχήμα 2.32*.



Σχήμα 2.32 Αναπαράσταση RGB

Ο κύριος στόχος του μοντέλου αναπαράστασης RGB είναι η παρουσίαση και εμφάνιση των εικόνων σε ηλεκτρονικά συστήματα όπως είναι η τηλεόραση και οι υπολογιστές αν και έχει χρησιμοποιηθεί σε συμβατικές φωτογραφίες. Πριν από την χρήση του στους υπολογιστές, το μοντέλο ήταν ήδη γνωστό αφού σε αυτό στηριζόταν η ανθρώπινη αντίληψη για τα χρώματα [13]. Λόγω του ότι η κύρια διακύμανση στην εμφάνιση του δέρματος επηρεάζεται σε μεγάλο βαθμό από τις συνθήκες φωτισμού προτιμώνται τα κανονικοποιημένα RGB χρώματα, καθώς είναι

---

σε θέση να φιλτράρονται ως αποτέλεσμα του φωτισμού. Οπότε οι κανονικοποιημένες τιμές του RGB δίνονται από τις ακόλουθες σχέσεις:

$$r = \frac{R}{R + G + B} \quad (2.11)$$

$$g = \frac{G}{R + G + B} \quad (2.12)$$

$$b = \frac{B}{R + G + B} \quad (2.13)$$

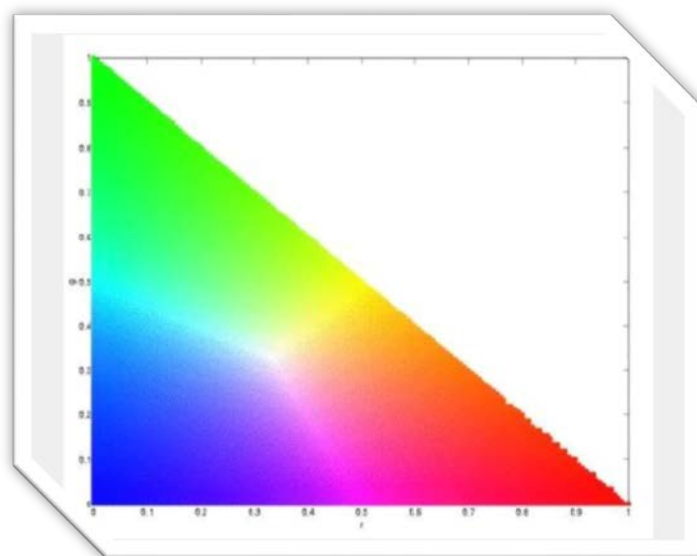
Παρατηρώντας τις παραπάνω εξισώσεις φαίνεται ότι:

$$r + g + b = 1 \quad (2.14)$$

δηλαδή είναι κανονικοποιημένα και οπότε μπορεί να γίνει παράληψη της τρίτης συνιστώσας αφού μπορεί να γραφεί σαν:

$$b = 1 - r - g \quad (2.15)$$

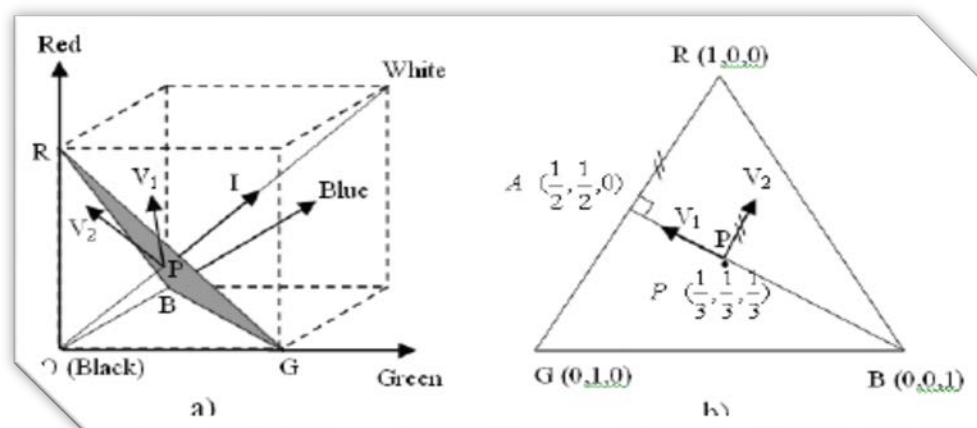
Αυτό έχει σαν επακόλουθο τη μείωση των διαστάσεων του χώρου (rgspace).



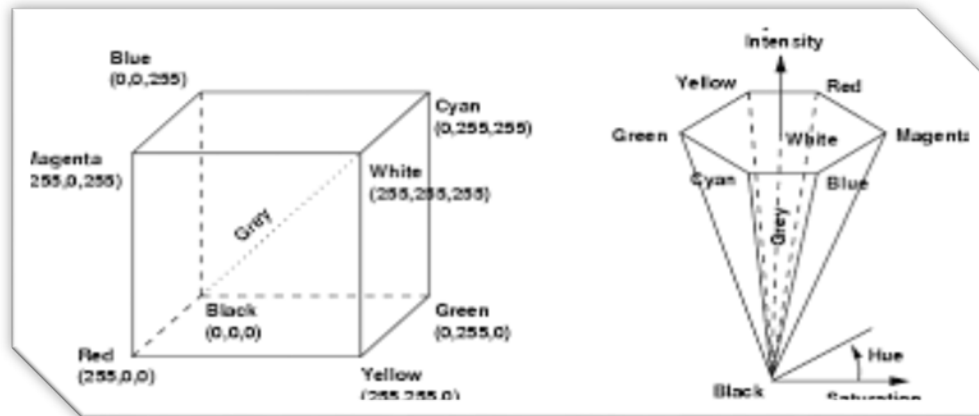
Σχήμα 2.33 RGSspace

Οι υπόλοιπες διαστάσεις που απομένουν ονομάζονται καθαρά χρώματα (pure colors) καθώς η εξάρτηση των  $r$  και  $g$  στη φωτεινότητα είναι αρκετά μειωμένη καθώς τα μεγέθη αυτά είναι κανονικοποιημένα. Μια ενδιαφέρουσα ιδιότητα του κανονικοποιημένου χώρου (rgspace) είναι ότι σε επιφάνειες ματ που αγνοείται το φως του περιβάλλοντος, παρουσιάζει αμεταβλητότητα σε περιστροφές προσανατολισμού σχετικές με την πηγή του φωτός [14]. Για αυτό και αρκετοί ερευνητές στη έρευνα που κάνουν για την ανίχνευση του χρώματος του δέρματος στηρίζονται στον κανονικοποιημένο αυτό χώρο [15].

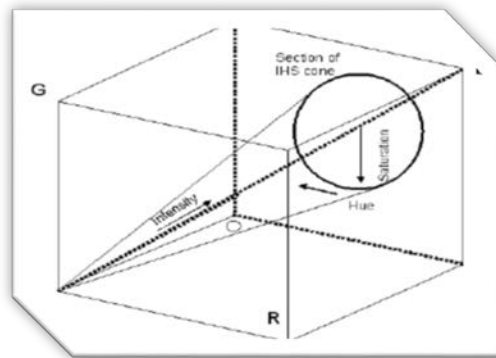
Εκτός από τον RGB χώρο υπάρχουν και άλλοι χρωματικοί χώροι που χρησιμοποιούνται στην ανίχνευση προσώπων. Ένας τέτοιος χώρος είναι ο χώρος HIS (H = Hue = Χρώμα, I = Intensity = Ένταση, Saturation = Κορεσμός), όπως φαίνεται στα παρακάτω σχήματα, που παρουσιάζει αρκετά πλεονεκτήματα σε σχέση με άλλους χρωματικούς χώρους σε ότι μας παρέχει μεγάλη διασπορά ανάμεσα στις χρωματικές περιοχές των χαρακτηριστικών περιοχών του προσώπου. Οπότε με βάση την ιδιότητα που έχει μπορεί να εφαρμοσθεί σε εικόνες και να συμβάλει στην εξαγωγή χαρακτηριστικών όπως είναι τα φρύδια, τα μάτια και τα χείλια. Ο συγκεκριμένος χρωματικός χώρος συνδέεται αρκετά στενά με την ανθρώπινη αντιληπτικότητα για το χρώμα και εφαρμόζεται σε συστήματα κατάτμησης προσώπου.



Σχήμα 2.34 Υπολογισμός Χώρου Αναπαράστασης HIS

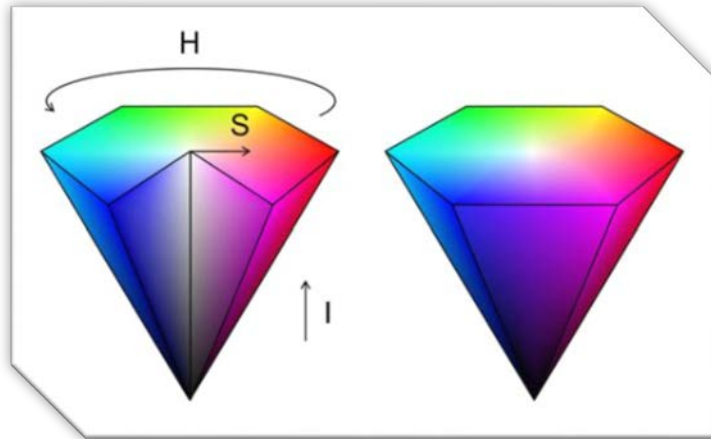


Σχήμα 2.35 Υπολογισμός Χώρου Αναπαράστασης HIS



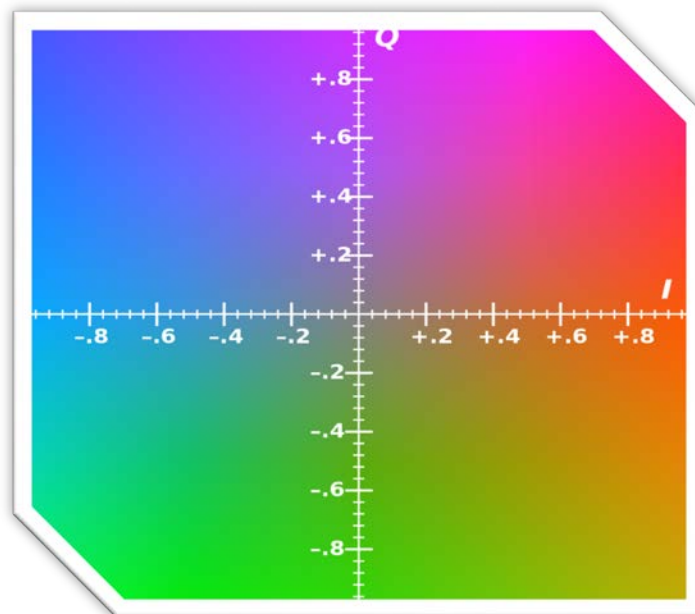
Σχήμα 2.36 Υπολογισμός Χώρου Αναπαράστασης HIS

Ένα άλλος χώρος αναπαράστασης των χρωμάτων που χρησιμοποιείται αρκετά είναι ο YIQ (Y = Luma = Φωτεινότητα, I = Απόχρωση, Q = Χρωματική Καθαρότητα) όπως φαίνεται στο σχήμα 2.37. Ο συγκεκριμένος χώρος χρησιμοποιείται αρκετά στην ανίχνευση των προσώπων [16].



Σχήμα 2.37 Χώρος Αναπαράσταση HIS

Βασικά μέσω της μετατροπής από τον χώρο RGB στον YIQ χρωματικό χώρο βρέθηκε ότι η συνιστώσα I που απεικονίζει την διακύμανση του χρώματος από το πορτοκαλί έως το κυανό βελτιώνει την περιοχή των αποχρώσεων[17]. Επιπρόσθετα η παραπάνω μετατροπή επιτρέπει την ανίχνευση μικρών προσώπων χωρίς να επηρεάζεται από το υπόβαθρο των άλλων χρωμάτων.

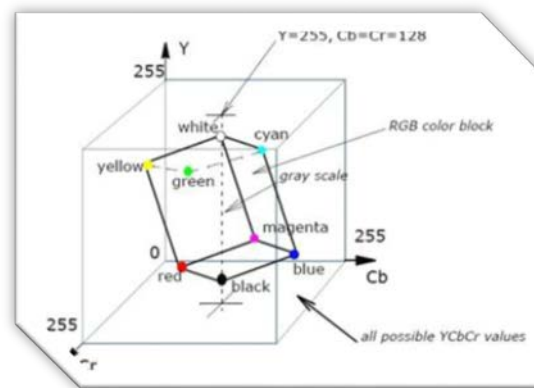


Σχήμα 2.38 Χώρος Αναπαράσταση YIQ

Ο χρωματικός χώρος YCbCr χρησιμοποιείται αρκετά στην ανίχνευση προσώπων[18]. Πρόκειται για την οικογένεια των χρωμάτων αναπαράστασης που



χρησιμοποιούνται στα τηλεοπτικά και ψηφιακά συστήματα φωτογραφίας. Το χρώμα αναπαρίσταται από τη Luma (Luminance=φωτεινότητα) και το οποίο κατασκευάζεται από το σταθμισμένο άθροισμα των τριών χρωματικών συνιστωσών του RGB, όπως φαίνεται στην σχέση που ακολουθεί: Το Y είναι η συνιστώσα της φωτεινότητας (luma) και το Cb και Cr είναι οι συνιστώσες που αναπαριστούν την διαφορά της luma από την κόκκινη και την μπλέ συνιστώσα αντίστοιχα όπως φαίνεται στο *σχήμα 2.39* [19].



*Σχήμα 2.39 Χώρος Αναπαράστασης YCbCr*

Η Y που συμβολίζει την φωτεινότητα του YCbCr είναι ανεξάρτητη από το χρώμα με αποτέλεσμα να μπορεί να χρησιμοποιηθεί για να λυθεί το πρόβλημα μεταβολής φωτεινότητας. Από την σύγκριση του συγκεκριμένου χώρου χρωμάτων και του κανονικοποιημένου rg, για την ανίχνευση αντικειμένων προτιμάται ο YCbCr καθώς με βάση την ανθρώπινη αντίληψη παρουσιάζει ομοιομορφία και χρησιμοποιείται σε τεχνικές συμπίεσης βίντεο (MPEG, JPEG). Στην εργασία των Rahman και Tonmoy το χρωματικό ιστόγραμμα με βάση τις συνιστώσες Cr και Cb δείχνει ότι το χρώμα προσώπου κατέχει ένα μικρό σύμπλεγμα στο ιστόγραμμα του χώρου YCbCr. Η κατανομή μπορεί να αναπαρασταθεί:

$$\text{Γκαουσιανό Μοντέλο } N(m, C) \text{ με} \quad (2.16)$$

$$\text{Μέση Τιμή } m = E\{x\} \text{ όπου } x = (r \ b)T \quad (2.17)$$

$$\text{Πίνακας Συνδιασποράς: } C = E\{(x - m)(x - m)T\} \quad (2.18)$$

---

Με την χρήση της Γκαουσιανής κατανομής να υπολογιστεί η πιθανότητα να ανήκει το εικονοστοιχείο μιας εικόνας στην περιοχή χρώματος. Οπότε η πιθανότητα ενός εικονοστοιχείου αφού προηγουμένως έχει μετασχηματιστεί από τον rg χώρο στον YCbCr υπολογίζεται από την παρακάτω σχέση:

$$P(r, b) = \exp[-0.5(x - m)^T \{x - m\}]^{-1} \quad (20), \quad (2.19)$$

όπου  $\chi = (r \ b)^T$

Εκτός από το χρωματικό μοντέλο HIS που αναφέρθηκε πιο πάνω χρησιμοποιείται και το μοντέλο HSV όπου (H = Hue, S = Saturation, V = Value, L = Lightness) [20]. Πιο αναλυτικά η συνιστώσα Hue ορίζει το χρώμα, η συνιστώσα Saturation καθορίζει τον κορεσμό του χρώματος και η συνιστώσα Value ορίζει την ένταση του χρώματος. Δηλαδή πρόκειται για εναλλακτική χρωματική αναπαράσταση του χρώματος. Πρόκειται για παραμορφώσεις του RGB χώρου όπου η μετατροπή τους στο χώρο HSV γίνεται μέσω ενός μη-γραμμικού μετασχηματισμού. Το πιο σημαντικό πλεονέκτημα που παρουσιάζει είναι ότι στο πρόβλημα ανίχνευσης δέρματος, δίνει την δυνατότητα στους χρήστες να διευκρινίσουν τα όρια του χρώματος του δέρματος με όρους κορεσμού (saturation) και χρώματος (Hue).

Οι Alabassi και Moldoveanu προτείνουν σε μια εργασία τους την ανίχνευση προσώπων στηριζόμενοι στο χρώμα του δέρματος κάνοντας χρήση των χρωματικών χώρων RGB, HSV, YCbCr. Αρχικά γίνεται εισαγωγή μιας εικόνας όπου γίνεται επανακαθορισμός του μεγέθους και κατάλληλη διόρθωση στο φωτισμό διότι το χρώμα του δέρματος επηρεάζεται από τις εκάστοτε συνθήκες φωτισμού που επικρατούν. Κατόπιν, η νέα εικόνα καταμηνύεται βασιζόμενη στο χρώμα του δέρματος χρησιμοποιώντας έναν συνδυασμό κατάτμησης των χρωματικών χώρων RGB και HSV που ακολουθείται από την μετατροπή του στο νέο χρωματικό χώρο κατάτμησης YCbCr χρησιμοποιώντας το ελλειπτικό πρότυπο. Όμως η παρακολούθηση των ανθρώπινων προσώπων, χρησιμοποιώντας σαν χαρακτηριστικό γνώρισμα το χρώμα παρουσιάζει αρκετά προβλήματα γιατί η

---

αναπαράσταση του χρώματος ενός προσώπου που λαμβάνεται από μια φωτογραφική μηχανή επηρεάζεται από πολλούς παράγοντες, όπως το φως του περιβάλλοντος, την μετακίνηση του αντικειμένου και άλλα. Τέλος, εκτός από τους παραπάνω χρωματικούς χώρους υπάρχουν και αρκετοί άλλοι.

Ο Terrillon πραγματοποίησε μια μελέτη όπου παρουσίαζε διάφορους χρησιμοποιημένους χρωματικούς χώρους [21]. Στην έρευνα αυτή έκανε συγκρίσεις των χώρων TSL (T = Απόχρωση, S = Κορεσμός, L = Λαμπρότητα), μοντελοποιώντας τις κατανομές των χρωμάτων των αντικειμένων είτε με μια συνάρτηση πυκνότητας πιθανότητας Gauss είτε με μίξη πολλών πυκνοτήτων πιθανότητας Gauss. Το συμπέρασμα που προέκυψε είναι ότι το χρώμα που αναπαρίσταται σε κανονικοποιημένους χρωματικούς χώρους με την χρήση της κατανομής Gauss αποδίδει πολύ καλά σε σχέση με μη κανονικοποιημένους χώρους όπου απαιτείται η χρήση μίξης πυκνοτήτων πιθανότητας. Ακόμα αποδείχτηκε ότι ο κανονικοποιημένος χώρος TSL δίνει τα καλύτερα αποτελέσματα και το πιο σημαντικό είναι ότι η ανίχνευση προσώπου καθορίζεται από τον βαθμό επικάλυψης των κατανομών δέρματος και μη σε ένα δοσμένο χώρο, που σημαίνει ότι υπάρχει μεγάλη εξάρτηση από το πλήθος δειγμάτων δέρματος και μη που θα εκπαιδευτούν.

Σε αυτό το OpenCV με το Python tutorial, πρόκειται να αναλυθεί πως δημιουργείται ένα είδος φίλτρου, ανατρέχοντας τις δυαδικές λειτουργίες, όπου θα φιλτραριστούν, ένα συγκεκριμένο χρώμα προσπαθώντας να φανεί απλά. Εναλλακτικά, θα μπορεί να φιλτραριστεί ένα συγκεκριμένο χρώμα και στη συνέχεια να αντικατασταθεί σε ένα αρχικό πλαίσιο.

Για να φιλτραριστεί υπάρχουν μερικές επιλογές. Αρχικά, απαιτείται να μετατραπούν τα χρώματα σε HSV, το οποίο είναι "Hue Saturation Value". Αυτό βοηθάει στον εντοπισμό ενός πιο συγκεκριμένου χρώματος, με βάση την απόχρωση και το εύρος κορεσμού, με μια διαφορά αξίας. Για παράδειγμα, εάν χρειαστεί να δημιουργηθούν πραγματικά φίλτρα με βάση τις τιμές του BGR.

---

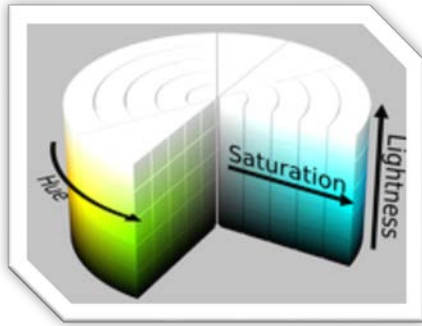
❖ **Χώρος HSL και HSV**

Οι HSL, *σχήμα 2.40*, ( απόχρωση, κορεσμός, ελαφρότητα ) και HSV, *σχήμα 2.41*, (απόχρωση, κορεσμός, αξία) [22] αποτελούν εναλλακτικές αναπαραστάσεις του έγχρωμου μοντέλου RGB που σχεδιάστηκε στη δεκαετία του 1970 από τους ερευνητές γραφικών υπολογιστών για να ευθυγραμμιστούν περισσότερο με τον τρόπο που οι ανθρώπινες όψεις αντιλαμβάνονται τις ιδιότητες χρωματισμού.

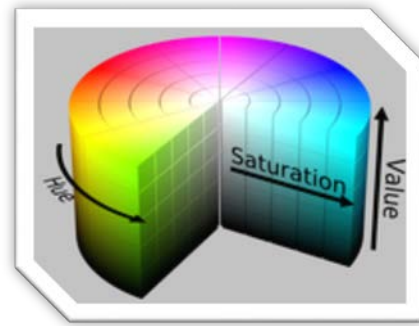
Σε αυτά τα μοντέλα, τα χρώματα κάθε απόχρωσης είναι διατεταγμένα σε μια ακτινική φέτα, γύρω από έναν κεντρικό άξονα ουδέτερων χρωμάτων που κυμαίνεται από το μαύρο στο κάτω μέρος έως το λευκό στην κορυφή. Η αναπαράσταση του HSV διαμορφώνει τον τρόπο με τον οποίο συνδυάζονται χρώματα διαφορετικών χρωμάτων, με την διάσταση κορεσμού να μοιάζει με διάφορες αποχρώσεις βαμμένων χρωμάτων και τη διάσταση της αξίας που μοιάζει με το μείγμα αυτών των χρωμάτων με ποικίλες ποσότητες μαύρου ή λευκού χρώματος. Το μοντέλο HSL επιχειρεί να μοιάζει περισσότερο με μοντέλα αντίληψης χρώματος όπως NCS ή Munsell, τοποθετώντας πλήρως κορεσμένα χρώματα γύρω από ένα κύκλο με τιμή ελαφρότητας  $1/2$ , όπου η τιμή φωτεινότητας 0 ή 1 είναι πλήρως μαύρη ή άσπρη, αντίστοιχα.

Οι HSL και HSV είναι και οι δύο κυλινδρικές γεωμετρίες, όπως φαίνεται στα παρακάτω σχήματα, με απόχρωση, τη γωνιακή τους διάσταση, ξεκινώντας από το κόκκινο πρωτεύον σε  $0^\circ$ , περνώντας μέσα από το πράσινο πρωτεύον σε  $120^\circ$  και το κυανό πρωτεύον σε  $240^\circ$  και στη συνέχεια τυλίγοντας πίσω στο κόκκινο σε  $360^\circ$ .

Σε κάθε γεωμετρία, ο κεντρικός κατακόρυφος άξονας περιλαμβάνει τα ουδέτερα, αχρωματικά ή γκρι χρώματα, που κυμαίνονται από το μαύρο με την ελαφρότητα 0 ή την τιμή 0, τον πυθμένα, με το λευκό με την ελαφρότητα 1 ή την τιμή 1, την κορυφή.

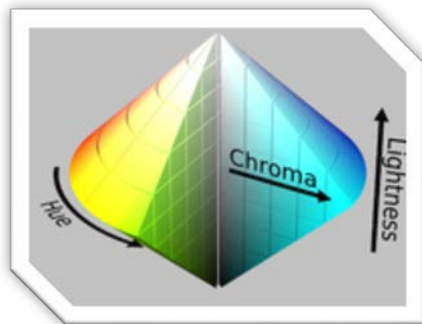


Σχήμα 2.40 Κυλινδρική Γεωμετρία HSL

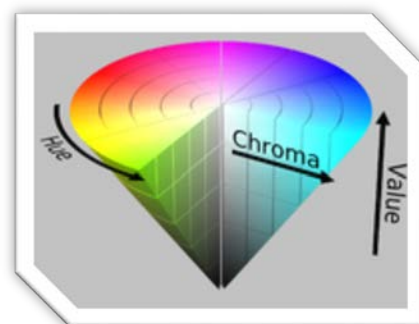


Σχήμα 2.41 Κυλινδρική Γεωμετρία HSV

Και στις δύο γεωμετρίες, τα πρόσθετα πρωτεύοντα και δευτερεύοντα χρώματα - μεσαία, κίτρινα, πράσινα, κυανό, μπλε και ματζέντα - και γραμμικά μείγματα μεταξύ γειτονικών ζευγών αυτών, μερικές φορές αποκαλούμενα καθαρά χρώματα, είναι διατεταγμένα γύρω από την εξωτερική άκρη του κυλίνδρου με κορεσμό 1. Αυτά τα κεκορεσμένα χρώματα έχουν ελαφρότητα 0.5 σε HSL, ενώ στον HSV έχουν τιμή 1. Η ανάμειξη αυτών των καθαρών χρωμάτων με τις μαύρες που παράγουν αποκαλούμενες αποχρώσεις - καθιστά τον κορεσμό αμετάβλητο. Στην HSL, ο κορεσμός παραμένει αμετάβλητος με τη χρώση με λευκά και μόνο τα μείγματα με μαύρους και λευκούς τόνους έχουν κορεσμό μικρότερο από 1. Στον HSV, η χρώση μόνο μειώνει τον κορεσμό.



Σχήμα 2.42

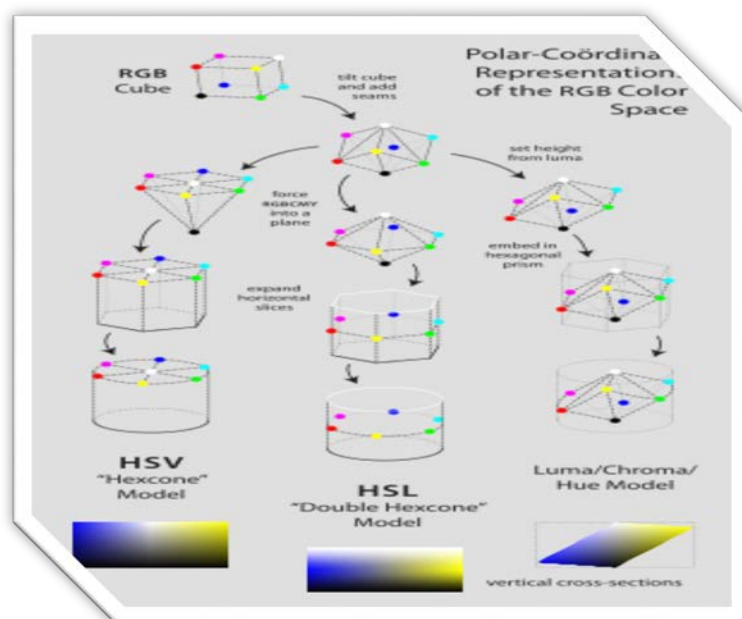


Σχήμα 2.43

Εάν το χρώμα της γραφικής παράστασης και η ελαφρότητα HSL ή η τιμή HSV έναντι χρώματος (εύρος τιμών RGB) και όχι ο κορεσμός (χρώματος πάνω από το μέγιστο χρώμα για αυτή τη φέτα), το προκύπτουν στερεό είναι bicone ή κώνος αντίστοιχα. Τέτοια διαγράμματα συχνά ισχυρίζονται ότι εκπροσωπούν άμεσα

την HSL ή την HSV, με την διάσταση χρώματος να χαρακτηρίζεται με σύγχυση ως "κορεσμός".

Επειδή αυτοί οι ορισμοί του κορεσμού στους οποίους τα πολύ σκοτεινά (και στα δύο μοντέλα) ή πολύ ελαφρά (σε HSL) σχεδόν ουδέτερα χρώματα θεωρούνται πλήρως κορεσμένα (για παράδειγμα, από το κάτω μέρος δεξιά στον κύλινδρο HSL σε φέτες ή από την κορυφή δεξιά) -συμφωνία με την διαισθητική έννοια της καθαρότητας του χρώματος, συχνά αντλείται ένα κωνικό ή δικλωνικό στερεό, με αυτό που το ονομάζεται chroma ως την ακτινική του διάσταση (ίσο με το εύρος των τιμών RGB), αντί του κορεσμού (όπου ο κορεσμός είναι ίσος με το χρώμα πάνω από το μέγιστο χρώμα σε εκείνο το κομμάτι του (δι) κώνου). Τέτοια διαγράμματα συνήθως χαρακτηρίζουν αυτή την ακτινική διάσταση "κορεσμό", θολώνουν ή διαγράφουν τη διάκριση μεταξύ κορεσμού και χρώματος. Όπως περιγράφεται παρακάτω, το computing chroma είναι ένα χρήσιμο βήμα για την παραγωγή κάθε μοντέλου. Επειδή ένα τέτοιο ενδιάμεσο μοντέλο με διαστάσεις απόχρωσης, χρώματος και HSV ή ελαφρότητας HSL παίρνει το σχήμα ενός κώνου ή δικωνικού, ο HSV ονομάζεται συχνά "**πρότυπο hexcone**" ενώ το HSL ονομάζεται συχνά "μοντέλο bi-hexcone".



Σχήμα 2.44 Computing Chroma

---

# 3

## ***Αλγόριθμοι Επεξεργασίας Εικόνας***

Σ' αυτό το κεφάλαιο θα αναλυθούν τρεις αλγοριθμικές τεχνικές που θα αναπτυχθούν στην παρούσα διπλωματική εργασία. Οι αλγόριθμοι αυτοί βασίζονται σε τεχνικές που αναλύθηκαν στο προηγούμενο κεφάλαιο και αφορούν συγκεκριμένα:

- ❖ Αλγόριθμος ανίχνευσης και κάλυψης στοιχείων προσώπου
- ❖ Αλγόριθμος ανίχνευσης κίνησης
- ❖ Αλγόριθμος ανίχνευσης μηκών κύματος οπτικού πεδίου

### ***3.1 Αλγόριθμοι Επεξεργασίας Εικόνας***

#### ***3.1.1 Αλγόριθμος Αναγνώρισης και Κάλυψης Στοιχείων Προσώπου***

Στην παράγραφο αυτή θα αναλυθεί ο κώδικας που γράφτηκε για την ανίχνευση και κάλυψη προσώπου χρησιμοποιώντας την τεχνική με Ιστόγραμμα Προσανατολισμένων Κλίσεων (HoG) στο Dlib, γιατί έγινε φανερό και σε προηγούμενο κεφάλαιο, η τεχνική αυτή παρουσιάζει αρκετά καλά αποτελέσματα.

Για την ακριβή ανίχνευση προσώπου έγινε χρήση της βιβλιοθήκης dlib της οποίας η εισαγωγή γίνεται όπως παρακάτω. Επίσης, εισάγεται η βασική βιβλιοθήκη OpenCV και η βοηθητική συνάρτηση imutils, για να διευκολύνουν βασικές λειτουργίες επεξεργασίας εικόνας όπως η μετάφραση, η περιστροφή, η αλλαγή

---

μεγέθους, ο σκελετισμός και η απεικόνιση των εικόνων Matplotlib με το OpenCV και την Python.

- ❖ Εισαγωγή βιβλιοθηκών και συναρτήσεων OpenCV, Dlib, Imutils:

```
import dlib
import cv2
import imutils
```

- ❖ Λήψη βίντεο:

```
stream = cv2.VideoCapture(0)
```

- ❖ Δημιουργία μεταβλητής ανίχνευσης προσώπου:

```
detector = dlib.get_frontal_face_detector()
```

- ❖ Δημιουργία πλαισίου ανίχνευσης προσώπου:

```
def draw_border(img, pt1, pt2, color, thickness, r, d):
    x1, y1 = pt1
    x2, y2 = pt2

    # Κορυφή πάνω αριστερά:
    cv2.line(img, (x1 + r, y1), (x1 + r + d, y1), color, thickness)
    cv2.line(img, (x1, y1 + r), (x1, y1 + r + d), color, thickness)
    cv2.ellipse(img, (x1 + r, y1 + r), (r, r), 180, 0, 90, color, thickness)

    # Κορυφή πάνω δεξιά:
    cv2.line(img, (x2 - r, y1), (x2 - r - d, y1), color, thickness)
    cv2.line(img, (x2, y1 + r), (x2, y1 + r + d), color, thickness)
    cv2.ellipse(img, (x2 - r, y1 + r), (r, r), 270, 0, 90, color, thickness)

    # Κορυφή κάτω αριστερά:
    cv2.line(img, (x1 + r, y2), (x1 + r + d, y2), color, thickness)
```



---

```
cv2.line(img, (x1, y2 - r), (x1, y2 - r - d), color, thickness)
cv2.ellipse(img, (x1 + r, y2 - r), (r, r), 90, 0, 90, color, thickness)

# Κορυφή κάτω δεξιά:
cv2.line(img, (x2 - r, y2), (x2 - r - d, y2), color, thickness)
cv2.line(img, (x2, y2 - r), (x2, y2 - r - d), color, thickness)
cv2.ellipse(img, (x2 - r, y2 - r), (r, r), 0, 0, 90, color, thickness)
```

- ❖ Εισαγωγή στιγμιότυπων εικόνας βίντεο:

```
while True:
    (grabbed, frame) = stream.read()
```

- ❖ Επεξεργασία κλίμακας πλαισίων – δημιουργία πλαισίου κλίμακας γκρι:

```
frame = imutils.resize(frame, width=700)
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

- ❖ Αντίγραφα πλαισίων για επεξεργασία διαφάνειας:

```
overlay = frame.copy()
output = frame.copy()
```

- ❖ Ορισμός τιμής διαφάνειας:

```
alpha = 0.5
```

- ❖ Ανίχνευση προσώπου στο γκρι πλαίσιο:

```
face_rects = detector(gray, 0)
```

- ❖ Βρόχος ανίχνευσης προσώπων:

```
for i, d in enumerate(face_rects):
    x1, y1, x2, y2, w, h = d.left(), d.top(), d.right() + 1, d.bottom() + 1, d.width(),
    d.height()
```

---

❖ Δημιουργία περιγράμματος στα ανιχνευμένα πρόσωπα με ταυτόχρονη θόλωση των χαρακτηριστικών:

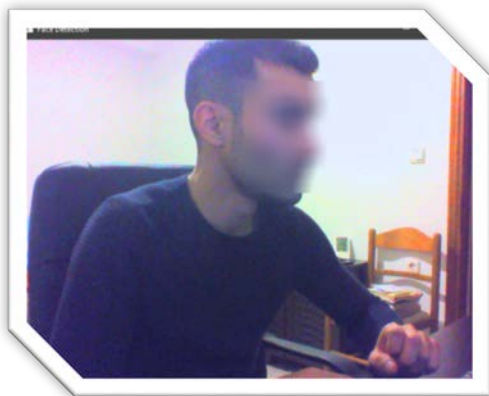
```
blur = cv2.blur(output[y1:y2, x1:x2], (30, 30))
output[y1:y2, x1:x2] = blur
# Εμφάνιση περιγράμματος:
cv2.imshow("Face Detection", output)
key = cv2.waitKey(1) & 0xFF
```

❖ Έξοδος προγράμματος:

```
key = cv2.waitKey(1) & 0xFF
ifkey == ord("q"):
break
```

❖ Απελευθέρωση μνήμης βίντεο και κλείσιμο όλων των παραθύρων:

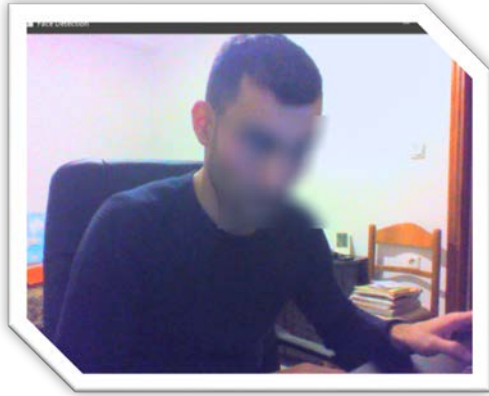
```
cv2.destroyAllWindows()
stream.stop()
```



Σχήμα 3.1 Πλάγια Όψη 1



Σχήμα 3.2 Πλάγια Όψη 2



Σχήμα 3.3 Κάτω Όψη



Σχήμα 3.4 Εμπρόσθια Όψη

### 3.1.2 Αλγόριθμος Αναγνώρισης Κίνησης

Στην παράγραφο αυτή θα αναλυθεί ο κώδικας που γράφτηκε για την ανίχνευση κίνησης χρησιμοποιώντας την τεχνική σύγκρισης ακολουθιών πλαισίων.

- ❖ Εισαγωγή βιβλιοθηκών OpenCV, Time, Datetime και Pandas:

```
import cv2 , time , pandas  
from datetime import datetime
```

- ❖ Δημιουργία μεταβλητής static\_back:

```
static_back = None
```

- ❖ Δημιουργία κενής λίστας motion\_list για τα κινούμενα αντικείμενα και λίστα time χρόνου κίνησης:

```
motion_list = [ None , None ]  
time = []
```

- ❖ Αρχικοποίηση του DataFrame, μια στήλη είναι η ώρα έναρξης και η άλλη στήλη είναι η ώρα λήξης:

```
df = pandas.DataFrame(columns = ["Start" , "End"])
```

- 
- ❖ Λήψη βίντεο:

```
video = cv2.VideoCapture(0)
```

- ❖ Βρόχος επεξεργασίας στοίβας της εικόνας ως βίντεο:

```
while True:  
    check , frame = video.read()  
    motion = 0
```

- ❖ Μετατροπή έγχρωμης εικόνας σε εικόνα gray\_scale:

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

- ❖ Μετατροπή εικόνας κλίμακας γκρι σε GaussianBlur έτσι ώστε η αλλαγή να μπορεί να βρεθεί εύκολα:

```
gray = cv2.GaussianBlur(gray, (9, 9), 2)
```

- ❖ Στην πρώτη επανάληψη αναθέτουμε την τιμή του static\_back στο πρώτο μας πλαίσιο:

```
if static_back is None:  
    static_back = gray  
    continue
```

- ❖ Διαφορά μεταξύ στατικού φόντου και του τρέχοντος πλαισίου (το οποίο είναι GaussianBlur):

```
diff_frame = cv2.absdiff(static_back, gray)
```

- ❖ Εάν η αλλαγή μεταξύ του στατικού φόντου και του τρέχοντος πλαισίου είναι μεγαλύτερη από 30, θα εμφανιστεί λευκό χρώμα (255):

```
thresh_frame = cv2.threshold(diff_frame, 55, 255, cv2.THRESH_BINARY)[1]  
thresh_frame = cv2.dilate(thresh_frame, None, iterations=2)
```

- 
- ❖ Εύρεση περιγράμματος κινούμενου αντικειμένου:

```
(_,cnts,_) = cv2.findContours(thresh_frame.copy(), cv2.RETR_EXTERNAL,  
cv2.CHAIN_APPROX_SIMPLE)  
for contour in cnts:  
    if cv2.contourArea(contour) < 50:  
        continue  
    motion = 1  
    (x,y,w,h) = cv2.boundingRect(contour)
```

- ❖ Δημιουργία πράσινου ορθογώνιου πλαισίου γύρω από το κινούμενο αντικείμενο:

```
blur = cv2.blur(frame[y:y+h, x:x+w], (35, 35))  
frame[y:y+h, x:x+w] = blur
```

- ❖ Προσθήκη κατάστασης κίνησης:

```
motion_list.append(motion)  
motion_list = motion_list[-2:]
```

- ❖ Προσάρτηση χρόνου έναρξης κίνησης:

```
#appending Start time of motion  
if motion_list[-1] == 1 and motion_list[-2] == 0:  
    time.append(datetime.now())  
if motion_list[-1] == 0 and motion_list[-2] == 1:  
    time.append(datetime.now())
```

- ❖ Προβολή εικόνας σε μορφή gray\_scale:

```
# cv2.imshow("Gray Frame", gray)
```

- ❖ Προβολή διαφοράς στο τρέχον πλαίσιο staticframe:

```
# cv2.imshow("Difference Frame", diff_frame)
```

---

❖ Προβολή ασπρόμαυρης εικόνας στην οποία όπου η διαφορά έντασης είναι μεγαλύτερη από 30 θα εμφανιστεί λευκή:

```
# cv2.imshow("Threshold Frame", thresh_frame)
```

❖ Προβολή πλαισίου με περίγραμμα κίνησης αντικειμένου:

```
#displaying color frame with contour of motion of object  
cv2.imshow("Color Frame", frame)
```

❖ Με το πλήκτρο q σταματά η λειτουργία του προγράμματος ακόμα και αν κινείται κάτι όταν προσαρτηθεί η ώρα λήξης της κίνησης:

```
key = cv2.waitKey(1)  
if key == ord('q'):  
    if motion == 1:  
time.append(datetime.now())  
    break
```

❖ Προσθήκη χρόνου κίνησης στο DataFrame και δημιουργία αρχείου csv, στον οποίο θα αποθηκευτεί ο χρόνος των κινήσεων:

```
for i in range(0, len(time), 2):  
    df = df.append({"Start":time[i], "End":time[i+1]}, ignore_index=True)  
df.to_csv("Time_of_movements.csv")
```

❖ Απελευθέρωση μνήμης βίντεο και κλείσιμο όλων των παραθύρων:

```
video.release()  
cv2.destroyAllWindows()
```



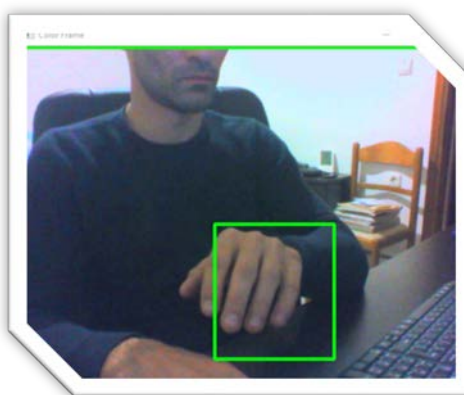
Σχήμα 3.5 Ασπρόμαυρο Πλαίσιο



Σχήμα 3.6 Γκρι Πλαίσιο



Σχήμα 3.7 Πλαίσιο Διαφορών



Σχήμα 3.8 Τελικό Πλαίσιο

### 3.1.3 Αλγόριθμος Αναγνώρισης Μηκών Οπτικού Φάσματος

Στην παράγραφο αυτή θα αναλυθεί ο κώδικας που γράφτηκε για την ανίχνευση μηκών κύματος οπτικού πεδίου χρησιμοποιώντας την μετατροπή του χώρου RGB σε χώρο HSV όπως περιγράψαμε σε προηγούμενο κεφάλαιο.

- ❖ Εισαγωγή απαραίτητων βιβλιοθηκών – στοιχείων:

```
import cv2  
import numpy as np
```

- ❖ Εισαγωγή εικόνων βίντεο καρέ:

```
cap = cv2.VideoCapture(0)
```

- 
- ❖ Βρόχος επανάληψης:

```
while(1):
```

- ❖ Διαβάζουμε ένα – ένα τα πλαίσια:

```
_, frame = cap.read()  
hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
```

- ❖ Μετατροπή BGR σε HSV:

```
_, frame = cap.read()  
hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
```

- ❖ Καθορισμός του μπλε χρώματος σε HSV:

```
hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)  
lower_blue = np.array([110,50,50])  
upper_blue = np.array([130,255,255])
```

- ❖ Κατώτατο όριο της εικόνας HSV για να ληφθούν μόνο μπλε χρώματα:

```
mask = cv2.inRange(hsv, lower_blue, upper_blue)
```

- ❖ Σύγκριση μάσκας με αρχική εικόνα:

```
res = cv2.bitwise_and(frame,frame, mask= mask)
```

- ❖ Εμφάνιση πλαισίων σύγκρισης και τελικής εικόνας:

```
cv2.imshow('frame',frame)  
cv2.imshow('mask',mask)  
cv2.imshow('res',res)
```

- ❖ Έξοδος προγράμματος:

```
k = cv2.waitKey(5) & 0xFF
```



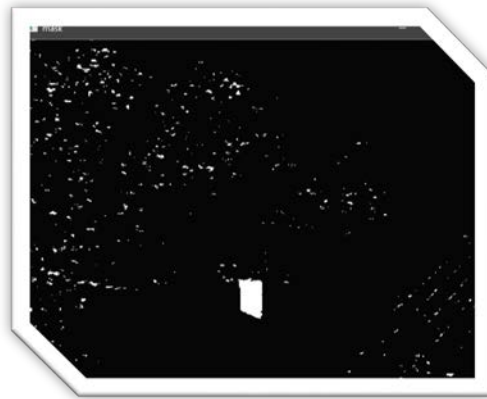
```
if k == 27:  
    break
```

❖ Απελευθέρωση μνήμης βίντεο και κλείσιμο όλων των παραθύρων:

```
cv2.destroyAllWindows()  
cap.release()
```



Σχήμα 3.9 Αρχικό Πλαίσιο



Σχήμα 3.103.9 Πλαίσιο Μάσκας



Σχήμα 3.11 Τελικό Πλαίσιο

---

## 3.2 Σύνοψη συγκρίσεων

### 1. Ανίχνευση Προσώπου με Ταξινομητές Χαρακτηριστικών (HaarCascade) στο OpenCV

➤ **Πλεονεκτήματα:**

- ✓ Λειτουργεί σχεδόν σε πραγματικό χρόνο στο CPU
- ✓ Απλή αρχιτεκτονική
- ✓ Εντοπίζει πρόσωπα σε διαφορετικές κλίμακες

➤ **Μειονεκτήματα:**

- ✓ Το κύριο μειονέκτημα αυτής της μεθόδου είναι ότι δίνει πολλές ψευδείς προβλέψεις.
- ✓ Δεν λειτουργεί σε μη μετωπικές εικόνες.
- ✓ Δεν λειτουργεί υπό την απόφραξη

### 2. Ανίχνευσης Προσώπου βασισμένο με Deep Learning στο OpenCV

➤ **Πλεονεκτήματα:**

- ✓ Η πιο ακριβής από τις τέσσερις μεθόδους
- ✓ Λειτουργεί σε πραγματικό χρόνο στο CPU
- ✓ Ανίχνευση προσανατολισμού προσώπου - πάνω, κάτω, αριστερά, δεξιά, πλάγια όψη κλπ.
- ✓ Λειτουργεί ακόμη και κάτω από σημαντική απόφραξη
- ✓ Εντοπίζει πρόσωπα σε διάφορες κλίμακες (ανιχνεύει μεγάλα καθώς και μικροσκοπικά πρόσωπα)

Ο ανιχνευτής με βάση το DNN, ξεπερνά όλα τα μειονεκτήματα του ανιχνευτή βασισμένο σε χαρακτηριστικά Haar, χωρίς να διακυβεύεται κανένα όφελος που παρέχει η Haar. Δεν εντοπίστηκε κανένα σημαντικό μειονέκτημα για αυτή τη μέθοδο, εκτός από το ότι είναι πιο αργό από το επόμενο ανιχνευτή προσώπου βασισμένο στο Dlib HoG.

---

### 3. Ανίχνευσης Προσώπου με Ιστόγραμμα Προσανατολισμένων Κλίσεων (HoG) στο Dlib

➤ **Πλεονεκτήματα:**

- ✓ Ταχύτερη μέθοδος στο CPU
- ✓ Λειτουργεί πολύ καλά για μετωπικά και ελαφρώς μη μετωπικά πρόσωπα
- ✓ Μοντέλο ελαφρού βάρους σε σύγκριση με τα άλλα τρία.
- ✓ Λειτουργεί κάτω από μικρή απόφραξη

Βασικά, αυτή η μέθοδος λειτουργεί στις περισσότερες περιπτώσεις εκτός από μερικές όπως συζητείται παρακάτω.

➤ **Μειονεκτήματα:**

- ✓ Το κύριο μειονέκτημα είναι ότι δεν ανιχνεύει μικρά πρόσωπα καθώς είναι εκπαιδευμένο για ελάχιστο μέγεθος προσώπου  $80 \times 80$ . Έτσι, θα πρέπει να βεβαιωθείτε ότι το μέγεθος του προσώπου θα πρέπει να είναι μεγαλύτερο από αυτό της εφαρμογής σας. Μπορείτε, ωστόσο, να εκπαιδεύσετε το δικό σας ανιχνευτή προσώπου για πρόσωπα μικρότερου μεγέθους.
  - ✓ Το περιθωριακό κουτί αποκλείει μερικές φορές μέρος του μέσου και ακόμη και μέρος του πηγουνιού μερικές φορές.
  - ✓ Δεν λειτουργεί πολύ καλά κάτω από σημαντική απόφραξη
  - ✓ Δεν λειτουργεί για πλευρικό πρόσωπο και ακραίες μη μετωπικές επιφάνειες, όπως κοιτάζοντας προς τα κάτω ή προς τα πάνω.

### 4. Ανίχνευση Προσώπου με Συγκεντρωτικά Νευρωνικά Δίκτυα στο Dlib

➤ **Πλεονεκτήματα:**

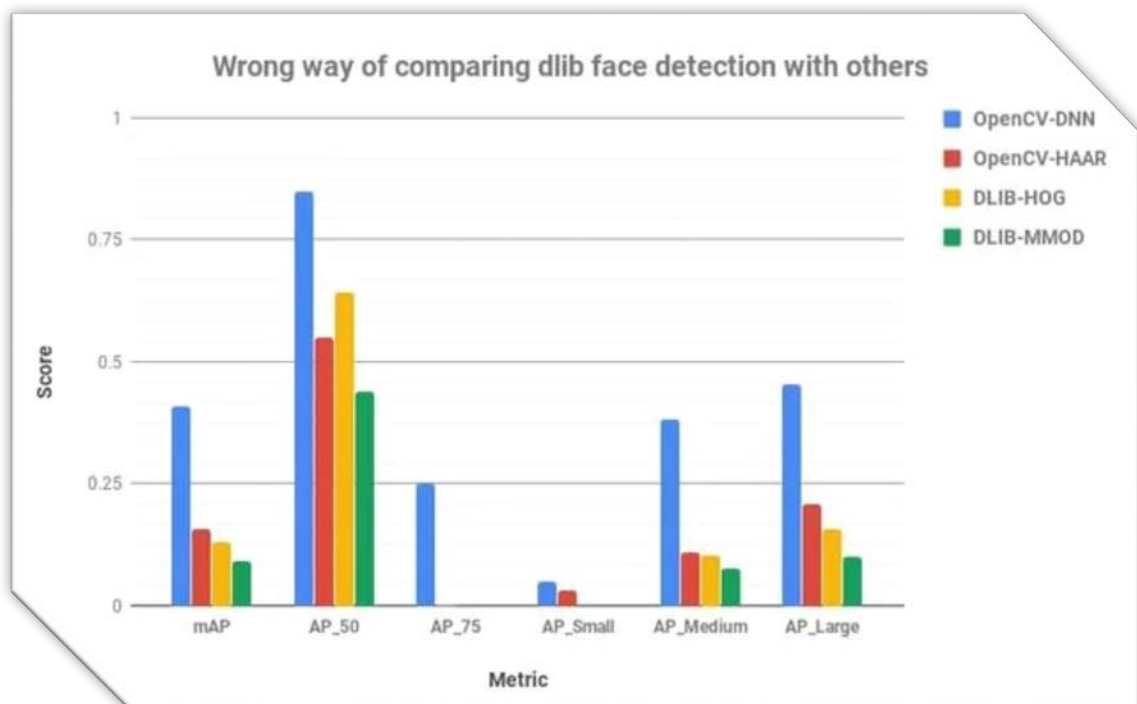
- ✓ Έργα για διαφορετικούς προσανατολισμούς προσώπου
- ✓ Στερεό σε απόφραξη
- ✓ Λειτουργεί πολύ γρήγορα στη GPU
- ✓ Πολύ εύκολη διαδικασία κατάρτισης

➤ **Μειονεκτήματα:**

- ✓ Πολύ αργά στη CPU
- ✓ Δεν ανιχνεύει μικρά πρόσωπα καθώς είναι εκπαιδευμένο για ελάχιστο μέγεθος όψης  $80 \times 80$ . Έτσι, θα πρέπει να βεβαιωθούμε ότι το μέγεθος του προσώπου θα πρέπει να είναι μεγαλύτερο από αυτό της εφαρμογής μας. Μπορούμε, ωστόσο, να εκπαιδύσουμε το δικό μας ανιχνευτή προσώπου για πρόσωπα μικρότερου μεγέθους.

- ✓ Το κιβώτιο οριοθέτησης είναι ακόμη μικρότερο από τον ανιχνευτή HoG.

5. **Συγκρίσεις Ακρίβειας Τεχνικών Ανίχνευσης Προσώπου**



Πίνακας 3.1 Διάγραμμα Σύγκρισης Τεχνικών Ανίχνευσης Προσώπου

- ✓ AP\_50 = Ακρίβεια όταν η επικάλυψη μεταξύ GroundTruth και προβλεπόμενου πλαισίου οριοθέτησης είναι τουλάχιστον 50% (IoU = 50%)

---

✓ AP\_75 = Ακρίβεια όταν η αλληλοεπικάλυψη μεταξύ της GroundTruth και του προβλεπόμενου πλαισίου οριοθέτησης είναι τουλάχιστον 75% (IoU = 75%)

✓ AP\_Small = Μέση ακρίβεια για πρόσωπα μικρού μεγέθους (Μέσος όρος IoU = 50% έως 95%)

✓ AP\_medium = Μέση ακρίβεια για πρόσωπα μεσαίου μεγέθους (Μέσος όρος IoU = 50% έως 95%)

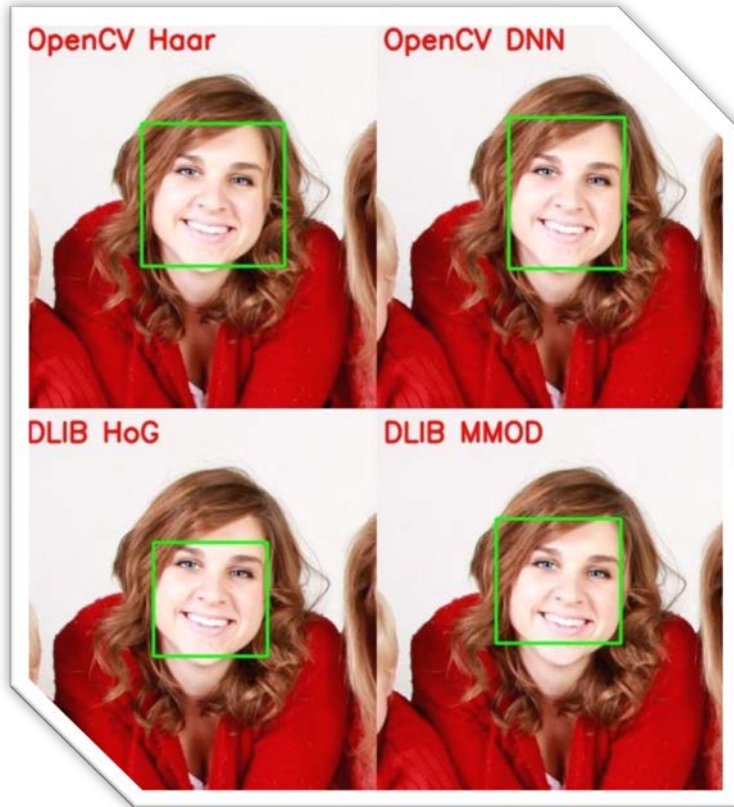
✓ AP\_Large = Μέση ακρίβεια για πρόσωπα μεγάλου μεγέθους (Μέσος όρος IoU = 50% έως 95%)

✓ mAP = Μέση ακρίβεια μεταξύ διαφορετικών IoU (Μέσος όρος IoU = 50% έως 95%)

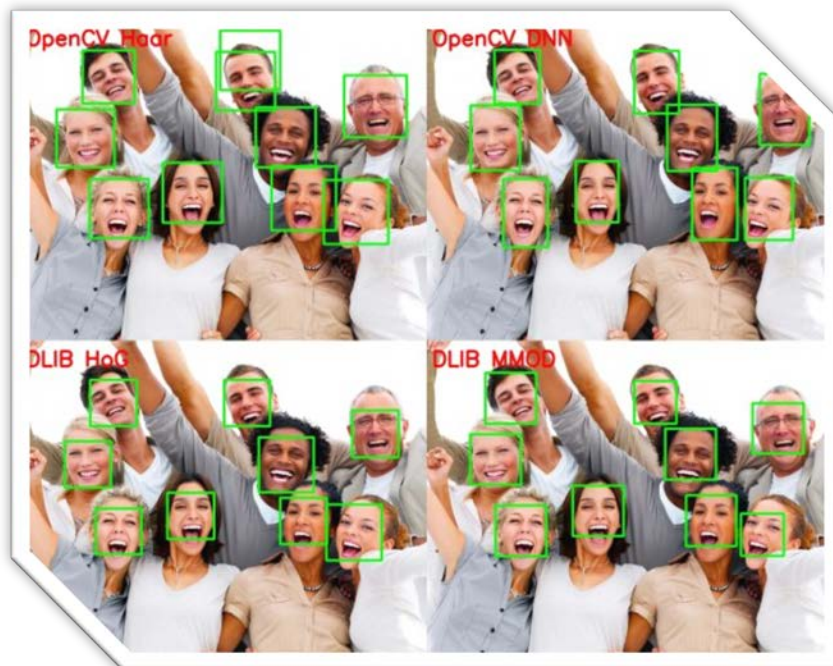
Οι κυριότεροι λόγοι χαμηλότερων ποσοστών για τις τεχνικές μέσω της βιβλιοθήκης DLib είναι οι εξής:

✓ Ο κύριος λόγος είναι ότι το dlib εκπαιδεύτηκε χρησιμοποιώντας τυποποιημένα σύνολα δεδομένων. Έτσι, βρέθηκε ότι ακόμη και όταν ανιχνεύονται τα πρόσωπα, τα πλαίσια είναι αρκετά διαφορετικά από αυτά των Haar ή OpenCV-DNN. Ήταν μικρότερα και συχνά κομμένα τμήματα του μετώπου και του πηγουνιού όπως φαίνεται παρακάτω. Αυτό σημαίνει μόνο ότι τα μοντέλα Dlib είναι σε θέση να ανιχνεύσουν περισσότερα πρόσωπα από αυτά του Haar, αλλά τα μικρότερα πλαίσια οριοθέτησης του dlib χαμηλώσουν τους AP\_75 και άλλους αριθμούς.

✓ Ο δεύτερος λόγος είναι ότι η Dlib δεν είναι σε θέση να ανιχνεύσει μικρά πρόσωπα, τα οποία παρασύρουν τους αριθμούς.



Σχήμα 3.10 Μετωπική ανίχνευση



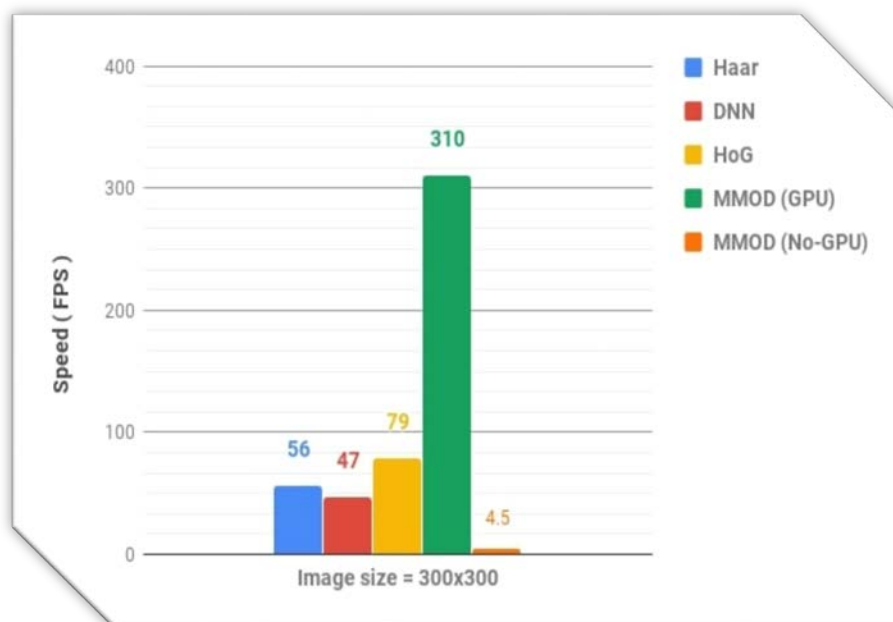
Σχήμα 3.11 Ανίχνευση πολλαπλών προσώπων

Έγινε χρήση μιας εικόνας  $300 \times 300$  για τη σύγκριση των παραπάνω τεχνικών. Ο ανιχνευτής MMOD μπορεί να τρέξει σε μια GPU, αλλά η υποστήριξη για GPU NVIDIA στο OpenCV εξακολουθεί να μην υπάρχει. Επομένως, αξιολογούνται οι μέθοδοι, μόνο στην CPU και επίσης, αναφέρεται το αποτέλεσμα για το MMOD στη GPU καθώς και για την CPU.

Τα προγράμματα δοκιμάστηκαν σε υλικό:

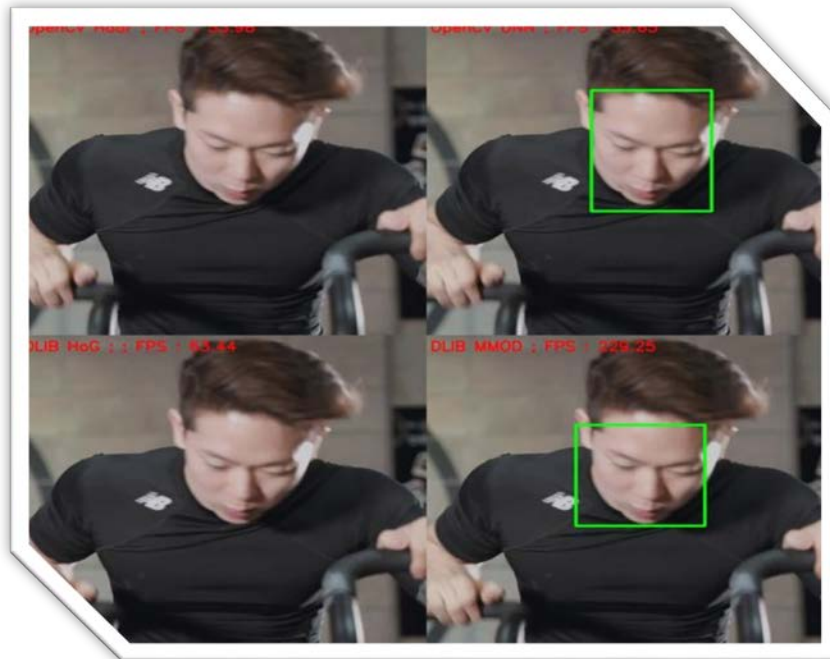
- Επεξεργαστής: Intel Core i7
- RAM: 16 GB
- GPU: NVIDIA GTX 1080 Ti με μνήμη RAM 11 GB
- OS: Windows 10
- Γλώσσα προγραμματισμού: Python

Όπως διαπιστώνεται, για την εικόνα αυτού του μεγέθους, όλες οι μέθοδοι εκτελούνται σε πραγματικό χρόνο, εκτός από το MMOD. Ο ανιχνευτής MMOD είναι πολύ γρήγορος σε μια GPU, αλλά είναι πολύ αργός σε μια CPU.



Πίνακας 3.2 Διάγραμμα χρόνου απόκρισης

Η μη μετωπική μπορεί να κοιτάζει δεξιά, αριστερά, πάνω, κάτω. Και πάλι, βεβαιώνουμε ότι το μέγεθος του προσώπου είναι μεγαλύτερο από 80×80. Ακολουθούν μερικά παραδείγματα.



Σχήμα 3.12 Κάτω ανίχνευση



Σχήμα 3.13 Πλάγια ανίχνευση

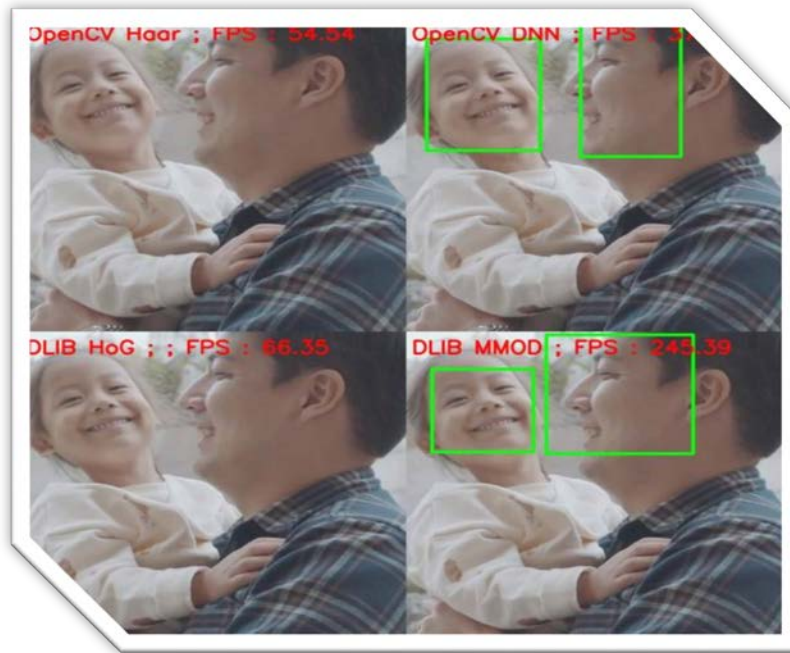




Σχήμα 3.14 Πλάγια κάτω αντίχνευση



Σχήμα 3.15 Κάτω αριστερή αντίχνευση



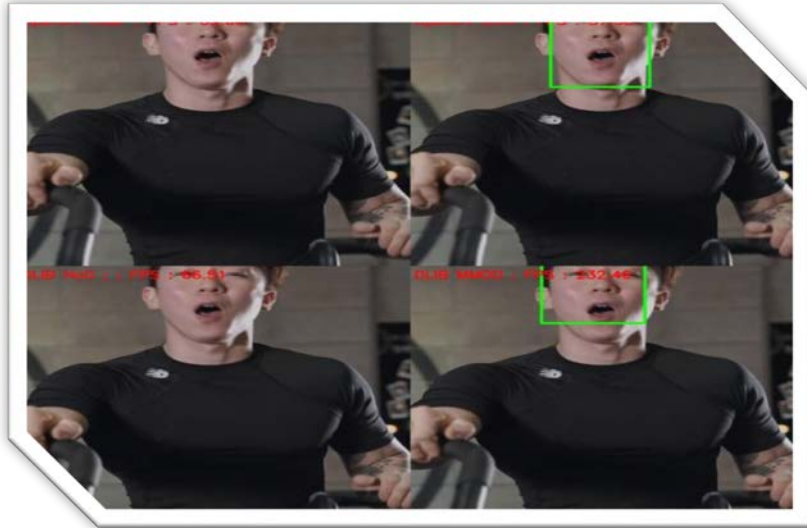
Σχήμα 3.16 Πολλαπλή ανίχνευση

Όπως αναμενόταν, ο ανιχνευτής με βάση το Haar αποτυγχάνει εντελώς. Ο ανιχνευτής βασισμένος στο HoG ανιχνεύει πρόσωπα για πρόσωπα αριστερής ή δεξιάς εμφάνισης (δεδομένου ότι εκπαιδεύτηκε σε αυτά) αλλά όχι τόσο ακριβή όσο οι ανιχνευτές DNC με βάση το OpenCV και το Dlib.

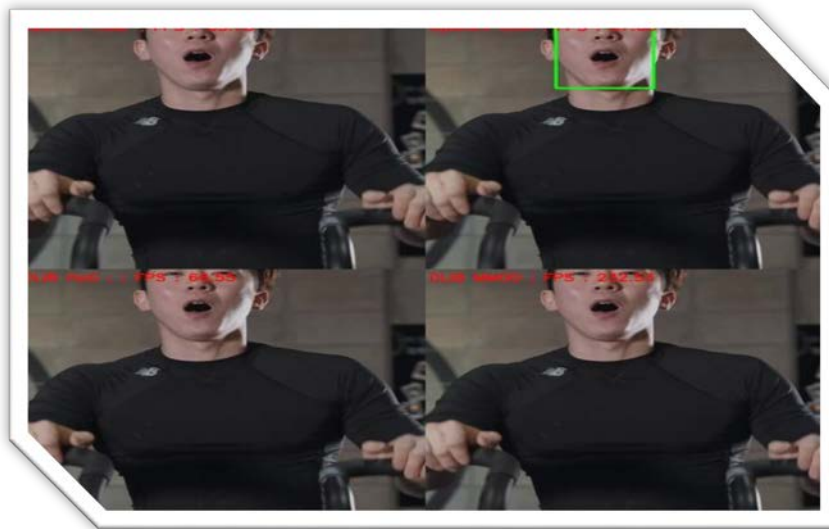
Παρακάτω θα παρουσιαστούν μερικά παραδείγματα εκτέλεσης ανίχνευσης υπό την απόφραξη από άλλα αντικείμενα.



Σχήμα 3.17 Ανίχνευση υπό κάλυψη 1



Σχήμα 3.18 Ανίχνευση υπό κάλυψη 2



Σχήμα 3.19 Ανίχνευση υπό κάλυψη 3

Και πάλι, οι μέθοδοι DNN ξεπερνούν τα υπόλοιπα δύο, με το OpenCV-DNN ελαφρώς καλύτερο από το Dlib-MMOD. Αυτό συμβαίνει κυρίως επειδή τα χαρακτηριστικά του CNN είναι πολύ πιο ισχυρά από τα χαρακτηριστικά HoG ή Haar.

---

Η σελίδα αυτή είναι σκόπιμα λευκή.

---

# 4

## **Επίλογος**

Νωρίτερα αναφέρθηκε ότι πολλές εφαρμογές όπως η αναγνώριση προσώπου, η παρακολούθηση ανθρώπων μέσω κάμερας, η αλληλεπίδραση ανθρώπων υπολογιστή απαιτούν σαν προκαταρκτικό στάδιο την ανίχνευση προσώπου. Με άλλα λόγια οι παραπάνω εφαρμογές προϋποθέτουν ότι ο εντοπισμός του προσώπου είναι προκαθορισμένος και διαθέσιμος για το επόμενο βήμα. Γενικά υπάρχουν πολλά προβλήματα που σχετίζονται με το πρόβλημα της ανίχνευσης προσώπου και μάλιστα είναι από τα πιο σημαντικά θέματα που έχει να αντιμετωπίσει η επεξεργασία εικόνας.

Το πρόβλημα της ανίχνευσης προσώπου μελετάται σε έντονο βαθμό τα τελευταία 30 χρόνια, λόγω των ποικίλων εφαρμογών που έχει στους τομείς που έχουν να κάνουν με την ασφάλεια των πολιτών. Στις αρχές του 1970 που ξεκίνησε η μελέτη του, οι τεχνικές που χρησιμοποιήθηκαν είχαν ανθρωπομετρικό χαρακτήρα, ενώ τα τελευταία χρόνια έχουν προταθεί μέθοδοι που στηρίζονται κατά βάση στην αναγνώριση προτύπων για την ανίχνευση προσώπων σε εικόνες και σε βίντεο.

Η ανίχνευση προσώπων και αντικειμένων καθώς και η ανίχνευση μηκών κύματος οπτικού πεδίου, είναι προβλήματα που απασχολούν ιδιαίτερα την επιστημονική κοινότητα τα τελευταία χρόνια, με μεθόδους και τεχνικές να εμφανίζονται με εξαιρετικά μεγάλο ρυθμό και θεωρούνται ακόμα ανοικτά πεδία έρευνας της επιστήμης της πληροφορικής.

---

## **4.1 Σύνοψη και συμπεράσματα**

Στην παρούσα διπλωματική εργασία αναλύθηκαν βασικές τεχνικές επεξεργασίας εικόνας βίντεο, τεχνικές από τις οποίες επιλύονται πολυπλοκότερα προβλήματα για στοχευμένη επεξεργασία δεδομένων. Ερευνήθηκαν οι επικρατέστερες τέσσερις τεχνικές ανίχνευσης προσώπων με τις μεθόδους DNN να ξεπερνούν τις υπόλοιπες δύο.

Έπειτα, εξετάστηκε η τεχνική ανίχνευσης κίνησης, μια τεχνική που βασίζεται στην σύγκριση ακολουθιών πλαισίων.

Τέλος έγινε μια μεγάλη ανάλυση στην τεχνική ανίχνευσης μηκών κύματος οπτικού πεδίου, παραθέτοντας πολλά χρωματικά μοντέλα και τις ιδιαιτερότητες αυτών με αποδοτικότερο όλων το HSV.

Τα παραπάνω προβλήματα αποτελούν προκλήσεις για τους σύγχρονους μηχανικούς, όπου τα μέσα επιτήρησης γίνονται ολοένα και περισσότερο κομμάτι της ζωής μας, αναπτύσσοντας γρήγορους και αξιόπιστους αλγόριθμους επεξεργασίας δεδομένων.

## **4.2 Μελλοντικές επεκτάσεις**

Η ανίχνευση προσώπου, όπως προαναφέρθηκε, είναι το πρώτο επίπεδο σε πολλά συστήματα επεξεργασίας προσώπου. Επίσης η αναγνώριση και ανίχνευση αντικειμένων καθώς και μηκών κύματος οπτικού φάσματος είναι προβλήματα και αυτά με ανοιχτό ερευνητικό πεδίο. Τα προβλήματα αυτά μπορούν να θεωρηθούν βασικά στην επίλυση πολυπλοκότερων για την ανίχνευση αντικειμένων με συγκεκριμένα χαρακτηριστικά και ιδιότητες.

Σύγχρονες τεχνικές βασίζονται πλέον στα νευρωνικά δίκτυα. Αρχικά χρησιμοποιείται ένα νευρωνικό δίκτυο για την εξαγωγή υποψήφιων θέσεων ενός αντικειμένου και στη συνέχεια οι θέσεις αυτές ταξινομούνται στην κατάλληλη κλάση και εξάγεται το αντίστοιχο παράθυρο με τη χρήση κάποιου άλλου συνελκτικού κυρίως νευρωνικού δικτύου. Τα τελευταία χρόνια εμφανίστηκε η τάση

---

της εξαγωγής της κλάσης και του παραθύρου ταυτόχρονα με τον προσδιορισμό των υποψήφιων θέσεων των αντικειμένων. Οι ανιχνευτές μονής λήψης, όπως ονομάζονται, φαίνεται να δίνουν της κατεύθυνση της ανίχνευσης αντικειμένων στο μέλλον, καθώς παρουσιάζουν αρκετά καλή ακρίβεια σε μικρό χρόνο εκτέλεσης.

Παράλληλα, λόγω της αυξημένης χρήσης συσκευών περιορισμένης επεξεργαστικής ισχύς, εμφανίζεται η ανάγκη περιορισμού των απαιτήσεων που χρειάζονται οι παραπάνω μέθοδοι τόσο σε μνήμη όσο και σε επεξεργαστική ισχύ. Το νευρωνικό δίκτυο Mobile Net σχεδιάστηκε με τέτοιο τρόπο ώστε η μέθοδος ανίχνευσης αντικειμένου που το χρησιμοποιεί να μπορεί να τρέξει με μια συσκευή κινητής τηλεφωνίας. Επομένως, φαίνεται πως ενώ στον τομέα της ακρίβειας του αποτελέσματος της ανίχνευσης έχει επιτευχθεί ένα ικανοποιητικό ποσοστό, το μέλλον προϋποθέτει την προσαρμογή των μεθόδων σε συσκευές με μικρότερες δυνατότητες όπως κινητά τηλέφωνα και tablets με τη χρήση «ελαφριών» νευρωνικών δικτύων.

---

Η σελίδα αυτή είναι σκόπιμα λευκή.



---

# 5

## **Βιβλιογραφία**

- [1] C. Akrivopoulou, *Human Rights and Risks in Digital Era. Globalization and the Effects of Information Technologies, USA, 2012.*
- [2] E. A. C. Lazar, *Practical Implementation of Face Detection by Using Matlab Cascade Object Detecto, IEEE, 2015.*
- [3] A. Downey, *Think Python, 2nd Edition, Massachusetts, 2015.*
- [4] T. Mezijin, *Learning OpenCV, Burundi, 2005.*
- [5] J. Petersen, *Handbook of Surveillance Technologies, 3rd Edition, New York, 2012.*
- [6] A. Berbar & A. Kandeel, *Faces and Facial Features Detection in Color Images, London, 2006.*
- [7] F. Macwan, *Face Detection Approaches: A Survey, International Journal of Innovative Research in Science, Engineering and Technology, Vol.3, Gujarat, 2014.*
- [8] B. Stroustrup, *Programming: Principles and Practice Using C++, 2nd Edition, United States, 2014.*
- [9] D. Watkins, *Opensource.com, Red Hat, 2018. [Ηλεκτρονικό]. Available: <https://opensource.com/article/18/4/getting-started-anaconda-python>.*
- [10] K. Herman, *CCTV Surveillance: Video Practices and Technology, Butterworth-Heinemann, 2nd Edition, London, 2006.*

- 
- [11] S. Lomheim, *CMOS/CCD Sensors and Camera Systems, 2nd Edition, Vol. PM172, 2007.*
- [12] F. Tan, *Face Detection in complex background based on skin color features and improved Adaboost algorithms, IEEE, Shanghai, 2010.*
- [13] R. Schapire, *Improved Boosting Algorithms Using Confidence-rated Predictions, 1999.*
- [14] J. Yang & A. Yang, *Areal-time face tracker, IEEE, Florida, 1996.*
- [15] W. Skarbek & A. Koschan, *Colour image segmentation-a survey, Technical University of Berlin, 1994.*
- [16] I. Craw, J. Lewthwaite & Brown, *A som based approach to skin detection with application in real time system, In Proc. Of the British Machine Vision Conference, 2001.*
- [17] G. Wei & I. K. Sethi, *Face detection for image annotation, 1999.*
- [18] Y. Dai & Y. Nakano, *Face-texture model based on sgld and its application, 1996.*
- [19] A. Albiol, *A simple and efficient face detection algorithm foe video database applications, International Conference on Image Processing, 2000.*
- [20] C. Garcia & G. Tziritas, *Face detection using quantized skin color regions,merging and wavelet packet analysis, IEEE, 1999.*
- [21] L. Yang, *Multiple-face tracking system for general region-of-interest video coding, International Conference on Image Processing, 2000.*
- [22] C. Terrillon, H. Fukamachi, & S. Akamatsu, *Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images, IEEE, 2000.*
- [23] Wikipedia, *Wikipedia.com, 2011, [Ηλεκτρονικό]. Available: [http://en.wikipedia.org/wiki/RGB\\_color\\_model](http://en.wikipedia.org/wiki/RGB_color_model).*
- [24] M. Agoston, *Computer Graphics and Geometric Modeling: Implementation and*

---

*Algorithms, London, 2006.*

[25] *J. D. Foley, Computer Graphics: Principles and Practice, 2nd Edition, California, 1995.*

[26] *N. Αγγελιδάκης, Εισαγωγή στον προγραμματισμό με την Python, Ηράκλειο, 2015.*

[27] *P. Viola & M. Jones, Robust Real-time Object Detection, Vol. 57, Canada, 2004.*