2η Εργαστηριακή Άσκηση
Εκτίμηση Οπτικής Ροής ($Optical\ Flow$) και Εξαγωγή Χαρακτηριστικών σε Βίντεο για Αναγνώριση Δράσεων

Κωνσταντίνος Κωστόπουλος ΑΜ: 03117043
Ανδρέας Βεζάκης ΑΜ: 03117186

2 Ιουνίου 2021

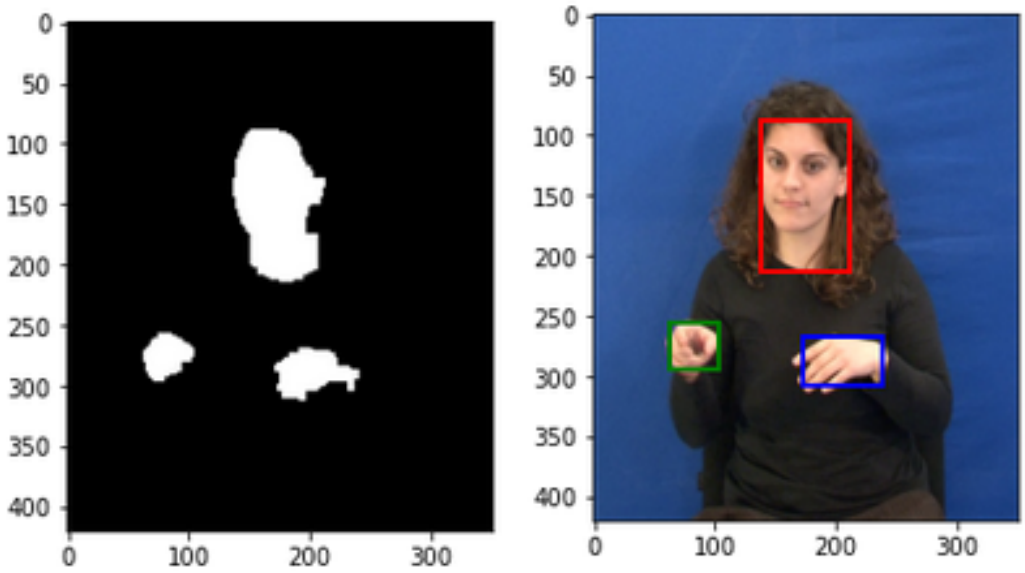
Μέρος 1: Παρακολούθηση Προσώπου και Χεριών με Χρήση της Μεθόδου Οπτικής Ροής των Lucas-Kanade

1.1 Ανίχνευση Δέρματος Προσώπου και Χεριών

Στο 1ο μέρος γίνεται ο υπολογισμός των διαστάσεων και της θέσης του ζητούμενου πλαισίου. Αυτό γίνεται με την ανίχνευση των σημείων του δέρματος στο 1ο $frame$ του βίντεο. Θεωρούμε ότι το χρώμα του δέρματος μοντελοποιείται με δισδιάστατη $Gaussian$ κατανομή:

$$P(c = skin) = \frac{1}{\sqrt{|\Sigma|(2\pi)^2}} e^{-1/2(e-\mu)\Sigma^{-1}(c-\mu)'}$$

Μετατρέπουμε το δείγμα των σημείων δέρματος από το δειγματικό χώρο RGB στο χώρο $YCbCr$, και στη συνέχεια με τις έτοιμες συναρτήσεις την $numpy$, ($numpy.mean$ και $numpy.cov$), υπολογίζουμε το διάνυσμα μέσης τιμής μ=[μ_{C_b}, μ_{C_r}] (όπου μ_{c_b} και μ_{C_r} οι μέσες τιμές για τα κανάλια C_b και C_r αντίστοιχα) καθώς και τον $2x2$ πίνακα συνδιακύμανσης Σ. Έπειτα, χρησιμοποιώντας την εικόνα πιθανοτήτων και κατώφλι 0.0001 παράγουμε μια εικόνα με δυαδικές τιμές, όπου τα λευκά σημεία αντιπροσωπεύουν τα σημεία δέρματος και τα μαύρα τα υπόλοιπα. Σε αυτό το σημείο το δέρμα που θα βρίσκαμε θα παρουσίαζε πολλές οπές, για την εξάλειψη αυτών επεξεργαζόμαστε την εικόνα με μορφολογικά φίλτρα. Ειδικότερα, εφαρμόζουμε $opening$ με μικρό δομικό στοιχείο και στη συνέχεια $closing$ με μεγαλύτερο. Τέλος, χρησιμοποιώντας τις έτοιμες συναρτήσεις της $python$, $addpatch$ και $patches.Rectangle$ σημειώνουμε με τετράγωνο τις περιοχές του δέρματος.



Σχήμα 1: Ανίχνευση Δέρματος

1.2 Παρακολούθηση Προσώπου και Χεριών

Στο δεύτερο μέρος της άσκησης στόχος είναι η παρακολούθηση του προσώπου που εντοπίσαμε στο πρώτο $frame$, στα επόμενα $frames$ του βίντεο. Θα χρησιμοποιήσουμε τον αλγόριθμο $Lucas$-$Kanade$ για την εύρεση της οπτικής ροής σε κάθε πλαίσιο ανάλυσης.

1.2.1 Υλοποίηση του Αλγόριθμου των Lucas-Kanade

Το πεδίο οπτικής ροής -d, φέρνει σε αντιστοιχία δυο διαδοχικά $frames$, έτσι ώστε:

$$I_n(x) = I_{n-1}(x + d)$$

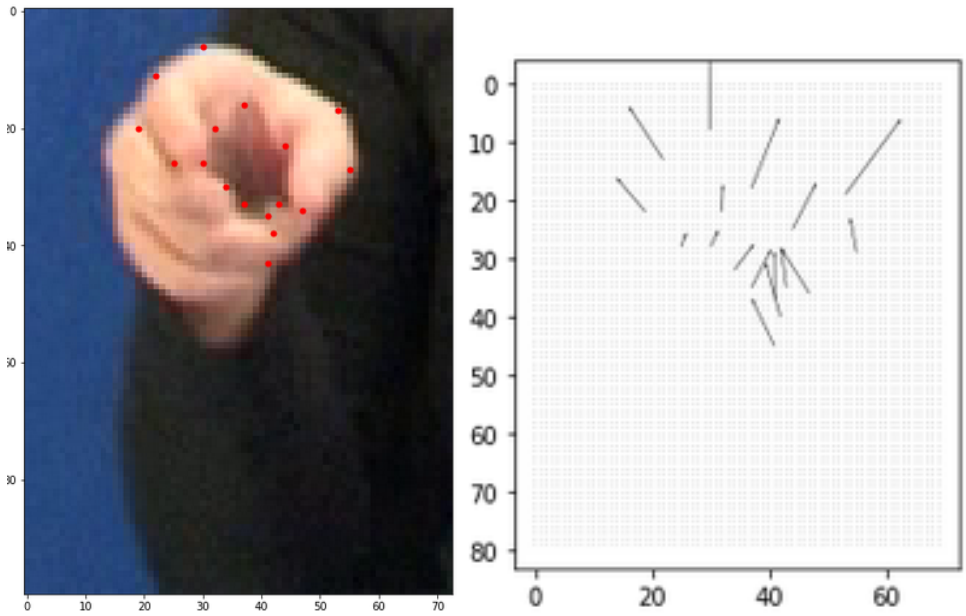
Ο αλγόριθμος υπολογίζει την οπτική ροή σε κάθε σημείο της εικόνας x με τη μέθοδο των ελαχίστων τετραγώνων, με βασική υπόθεση ότι το d είναι σταθερό σε ένα μικρό παράθυρο γύρω από το σημείο και ελαχιστοποιώντας το τετραγωνικό σφάλμα:

$$J_x(d) = \int G_\rho(x - x')[I_n(x') - I_{n-1}(x' + d)]^2 dx'$$

Όπου $G\rho(x)$ είναι $Gaussian$ με τυπική απόκλιση ρ.
Έστω ότι έχουμε μια εκτίμηση d_i για την ροή και θέλουμε να την βελτιώσουμε κατά u, δηλαδή $d_{i+1} = d_i + u$. Αναπτύσσοντας κατά $Taylor$ το $I_{n-1}(x + d) = I_{n-1}(x + d_i + u)$ και λύνοντας με βάση το τετραγωνικό σφάλμα βρίσκουμε την λύση:

$$u(x) = \begin{bmatrix} (G_\rho * A_1^2)(x) + \epsilon & (G_\rho * (A_1 A_2))(x) \\ (G_\rho * A_1 A_2)(x) & (G_\rho * A_2^2)(x) + \epsilon \end{bmatrix}^{-1} \begin{bmatrix} (G_\rho * A_1 E)(x) \\ (G_\rho * A_2 E)(x) \end{bmatrix}$$

Αυτόν τον εφαρμόζουμε μόνο για τα σημεία ενδιαφέροντος (γωνίες) (τα οποία εντοπίζουμε με την έτοιμη συνάρτηση της $cv2$ την $goodFeaturesToTrack$) και η επαναληπτική διαδικασία υπολογισμού του πεδίου οπτικής ροής επαναλαμβάνεται μέχρι να φτάσουμε σε σύγκλιση. Ως αρχική συνθήκη θεωρούμε d_0 =0.



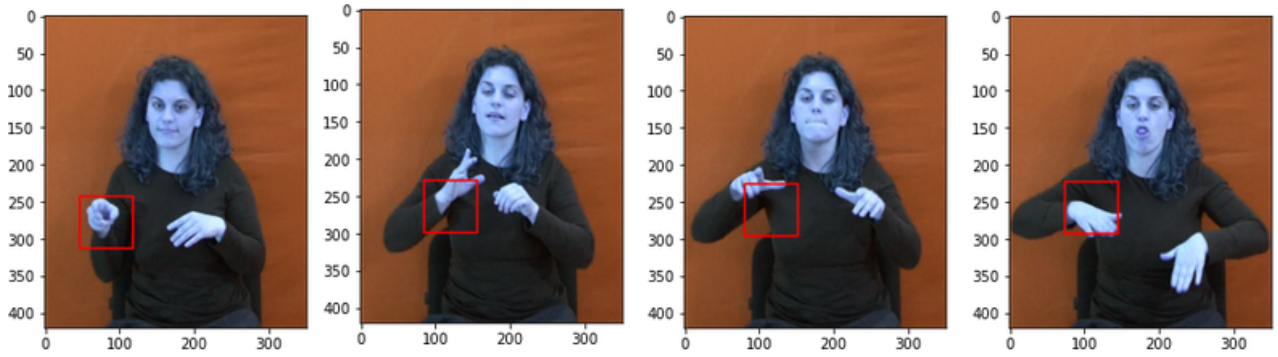
Σχήμα 2: α) Ανίχνευση Σημείων Ενδιαφέροντος β) Εφαρμογή του αλγορίθμου $Lucas\ Kanade$ για την ίδια εικόνα μετατοπισμένη προς τα πάνω.

Παρατηρούμε ότι η αύξηση του ε ελάχιστα έως καθόλου επηρεάζει το αποτέλεσμα μας, ωστόσο η επιλογή της τιμής της απόκλισης της $Gaussian$ παίζει καθοριστικό ρόλο για το σωστό εντοπισμό του πλαισίου. Βέβαια, η επιλογή των βέλτιστων τιμών είναι δύσκολη να γίνει με γυμνό μάτι εξαιτίας των πολύ μικρών διαφορών.

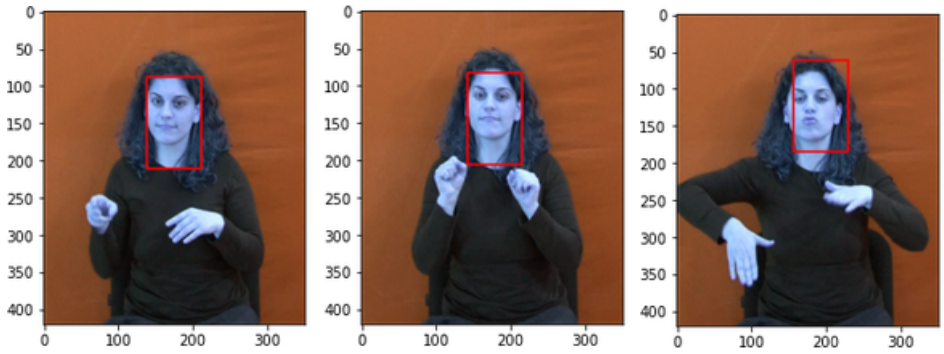
1.2.2 Υπολογισμός της Μετατόπισης των Παραθύρων από τα Διανύσματα Οπτικής Ροής

Η ενέργεια του πεδίου οπτικής ροής για ένα συγκεκριμένο $pixel$ ορίζεται ως το τετράγωνο του μέτρου του διανύσματος οπτικής ροής στο $pixel$ αυτό. Το τελικό διάνυσμα μετατόπισης του ορθογωνίου μπορεί να υπολογιστεί από την ενέργεια του πεδίου οπτικής ροής, αν πάρουμε υπόψιν τις

πιο σημαντικές συνιστώσες και υπολογίσουμε τη μέση τιμή τους. Οι σημαντικότερες συνιστώσες υπολογίζονται με χρήση ενός κατωφλίου, το οποίο ορίζει τι ποσοστό της μέγιστης ενέργειας πρέπει να έχει ένα $pixel$ προκειμένου να θεωρηθεί ότι συνεισφέρει ουσιαστικά. Ακολουθούν ενδεικτικά αποτελέσματα για διάφορα $frames$:



Σχήμα 3: Εντοπισμός της κίνησης του αριστερού χεριού.



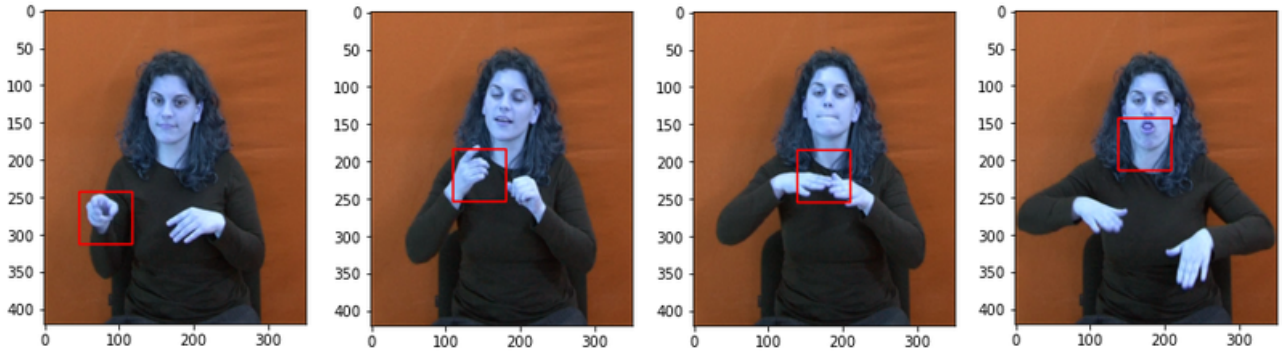
Σχήμα 4: Εντοπισμός της κίνησης του κεφαλιού.

Παρατηρούμε ότι ο αλγόριθμος μας καταφέρνει να ακολουθήσει το χέρι, όμως σε μερικές περιπτώσεις δεν ¨πιάνει¨ ολόκληρο το χέρι, και σε απότομες κινησεις ο εντοπισμός της κίνησης δεν είναι τόσο καλός. Ακόμη, οι κινήσεις του κεφαλιού είναι πολύ μικρές και επομένως το τετράγωνο παραμένει πάνω του.

1.2.3 Πολυ-Κλιμακωτός Υπολογισμός Οπτικής Ροής

Παρ' όλο που ο παραπάνω αλγόριθμος δίνει ικανοποιητικά αποτελέσματα, από τις παρατηρήσεις αντιλαμβανόμαστε ότι παρουσιάζει αδυναμίες. Για αυτό θα υλοποιήσουμε την πολυκλιμακωτή έκδοση του αλγορίθμου μας.
Σε αυτήν την έκδοση ο αλγόριθμος εκτελείται σε δύο στάδια. Στο πρώτο στάδιο λαμβάνονται τα πλαίσια I_{n-1} και I_n προς υπολογισμό της οπτικής ροής. Για αυτά τα πλαίσια κατασκευάζεται μια πυραμίδα η βάση της οποίας αποτελείται από τα ίδια πλαίσια. Όσο ανεβαίνουμε στην πυραμίδα τα δύο πλαίσια μειώνονται στο μισό μέγεθος τους. Το πλήθος των βαθμίδων της πυραμίδας βρίσκεται στην ευχέρεια του χειριστή. Στο δεύτερο στάδιο εκτελείται μια αναδρομή ξεκινώντας από την κορυφή της πυραμίδας. Εφαρμόζεται η μονοκλιμακωτή έκδοση του αλγορίθμου όπως ακριβώς περιγράφηκε παραπάνω. Το διάνυσμα εξόδου του μονοκλιμακωτού αλγορίθμου θα τεθεί ως αρχική συνθήκη στην επόμενη βαθμίδα της πυραμίδας. Αξίζει να σημειωθεί ωστόσο, ότι ξεκινώντας από την κορυφή της πυραμίδας, το τρέχον διάνυσμα οπτικής ροής αντιστοιχεί σε πλαίσιο

υποδιπλάσιας κλίμακας από εκείνη της επόμενης βαθμίδας. Για αυτό κατά την μετάβαση στην επόμενη βαθμίδα οφείλουμε αρχικά να διπλασιάσουμε το μέγεθος του διανύσματος d προκειμένου η ανάλυση να γίνει σε ίδια κλίμακα.
Παρατηρούμε ότι ο πολυκλιμακωτός αλγόριθμος μας αν και στην αρχή και για αρκετά $frames$ εντόπιζε το χέρι με καλή ακρίβεια, στο τέλος ¨χάθηκε¨. Ακολουθούν μερικά αποτελέσματα του αλγόριθμου μας.



Σχήμα 5: Εντοπισμός της κίνησης του αριστερού χεριού με τον πολυκλιμακωτό $Lucas\ Kanade$.

Μέρος 2: Εντοπισμός Χωρο-χρονικών Σημείων Ενδιαφέροντος και Εξαγωγή Χαρακτηριστικών σε Βίντεο Ανθρωπίνων Δράσεων

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η επίλυση του προβλήματος κατηγοριοποίησης βίντεο που περιέχουν ανθρώπινες δράσεις. Η παρακάτω ανάλυση βασίζεται στην εξαγωγή χωρο- χρονικών χαρακτηριστικών μέσω δύο διαφορετικών ανιχνευτών ($Harris$ και $Gabor$). Τα χαρακτηριστικά αυτά συνεισφέρουν στους τοπικούς περιγραφητές (θα τους αναλύσουμε στην επόμενη ενότητα), η συλλογή των οποίων είναι ικανή να αναπαραστήσει τη στατιστική κατανομή τους. Όπως αναφέρθηκε θα χρησιμοποιήσουμε δύο ανιχνευτές χωρο-χρονικών σημείων ενδιαφέροντος, $Harris\ Detector$ και $Gabor\ Detector$.

2.1 Χωρο-χρονικά Σημεία Ενδιαφέροντος

2.1.1 Υλοποίηση του ανιχνευτή Harris στις 3 διαστάσεις

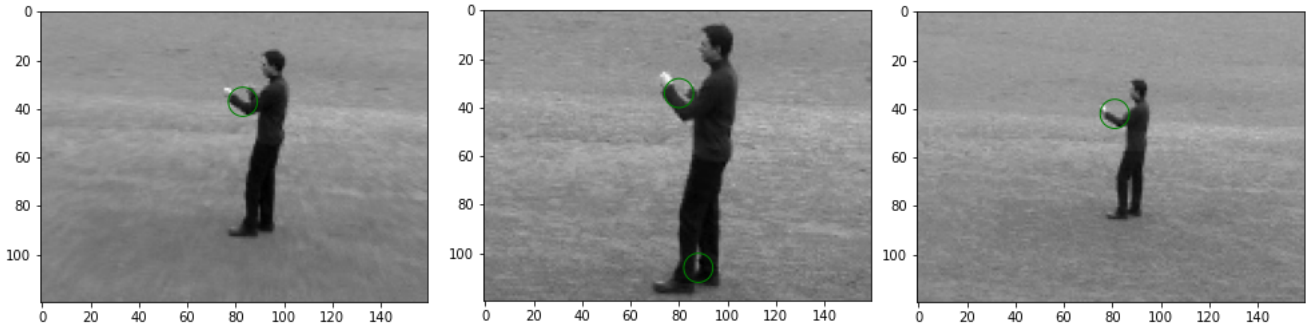
Ο $3D$ ανιχνευτής $Harris$ προκύπτει από τον ανιχνευτή γωνιών $Harris$-$Stephens$, αν προσθέσουμε στον $2D$ δομικό τανυστή και τη χρονική παράγωγο. Έτσι, ο $3x3$ πίνακας M προκύπτει:

$$M(x, y, t, \sigma, \tau) = g(x, y, t, s\sigma, s\tau) * \begin{pmatrix} L_x^2 & L_xL_y & L_xL_t \\ L_xL_y & L_y^2 & L_yL_t \\ L_xL_t & L_yL_t & L_t^2 \end{pmatrix}$$

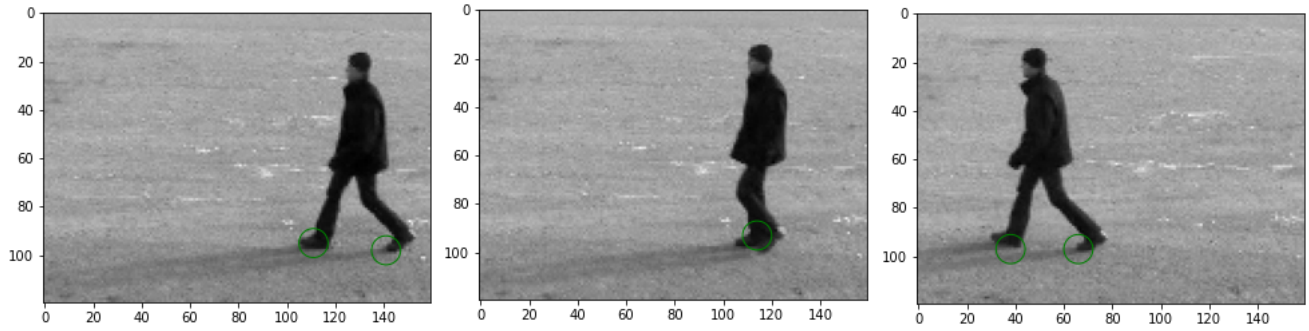
όπου έχει προκύψει από $3D$ φιλτράρισμα με γκαουσιανές συναρτήσεις των κατευθυντικών παραγώγων σε χωρική κλίμακα σ και χρονική κλίμακα τ. Αρχικά λαμβάνουμε τις κατευθυντικές παραγώγους είτε μέσω ένος πυρήνα κεντρικών διαφορών $[-101]^T$ είτε στο δικό μας αλγόριθμο από έτοιμες συναρτήσεις της $python$ και στη συνέχεια εφαρμόζουμε συνέλιξη ως προς κάθε διάσταση. Ομοίως επεκτείνουμε και το κριτήριο γωνιότητας. Το κριτήριο γωνιότητας λαμβάνει τη μορφή:

$$H(x,y,t) = det(M(x,y,t)) - k * trace^3(M(x,y,t))$$

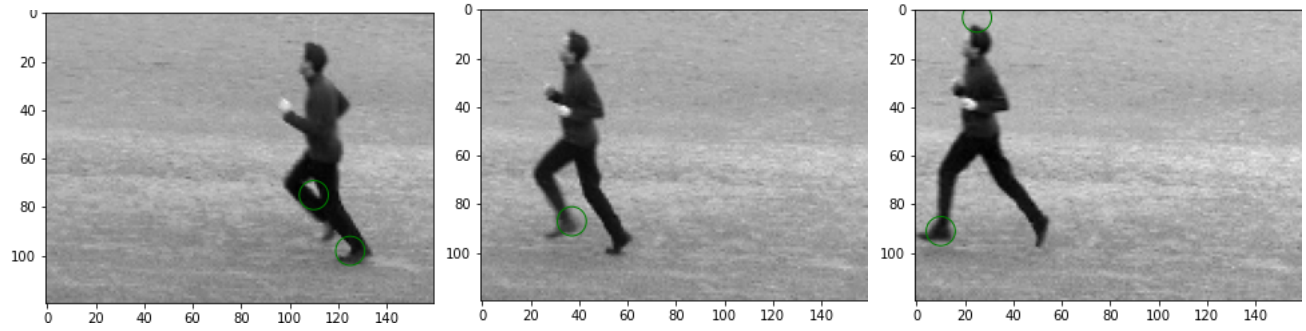
Έπειτα, ομοίως με τον αλγόριθμο του πρώτου εργαστηρίου εφαρμόζουμε τα κριτήρια για την επιλογή των σημείων ενδιαφέροντος.



Σχήμα 6: Εντοπισμός σημείων ενδιφέροντων για βίντεο μποξ.



Σχήμα 7: Εντοπισμός σημείων ενδιφέροντων για βίντεο βαδίσματος.



Σχήμα 8: Εντοπισμός σημείων ενδιφέροντων για βίντεο τρεξίματος.

2.1.2 Υλοποίηση του ανιχνευτή Gabor

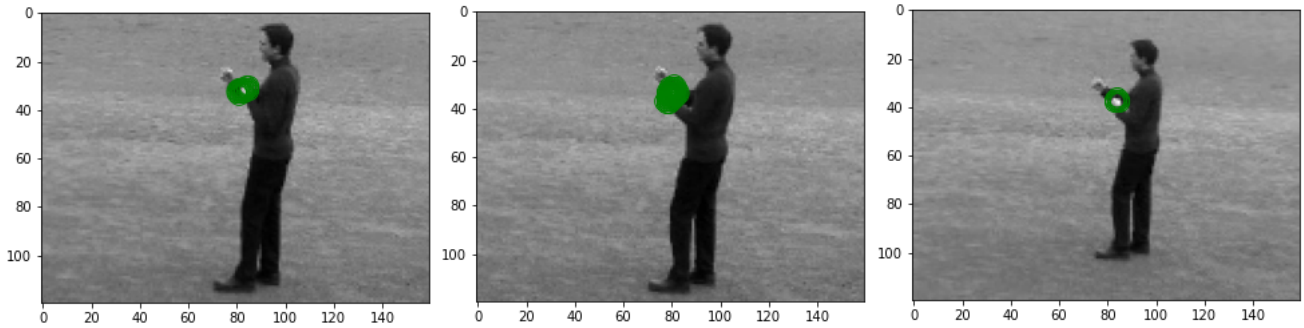
Ο ανιχνευτής *Gabor* βασίζεται στο χρονικό φιλτράρισμα του βίντεο μέσω ενός περιττού και ενός άρτιου φίλτρου *Gabor* στο διάστημα [-2τ,2τ] . Τα φίλτρα αυτά είναι της μορφής:

$$h_{ev}(t,\tau,\omega) = cos(2\pi t\omega)exp(-t^2/2\tau^2)$$
$$h_{od}(t,\tau,\omega) = sin(2\pi t\omega)exp(-t^2/2\tau^2)$$

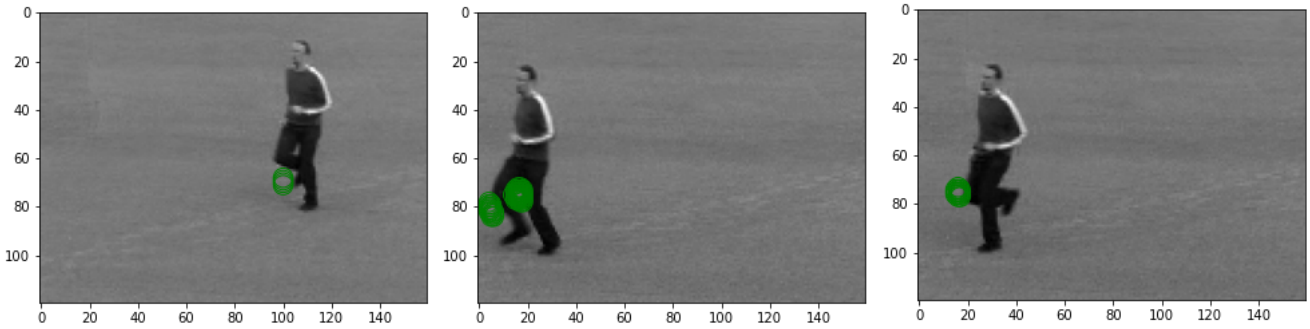
Επίσης, η συχνότητα ω σε κάθε φίλτρο σχετίζεται με την χρονική κλίμακα τ μέσω της σχέσης ω = 4/τ.
Εφαρμόζοντας ξεχωριστά χρονικό φιλτράρισμα με τα δύο παραπάνω *Gabor* φίλτρα, λαμβάνουμε την επέκταση του κριτηρίου τετραγωνικής ενέργειας (κριτήριο σημαντικότητας):

$$H(x,y,t) = (I(x,y,t) * g * h_{ev})^2 + (I(x,y,t) * g * h_{od})^2$$

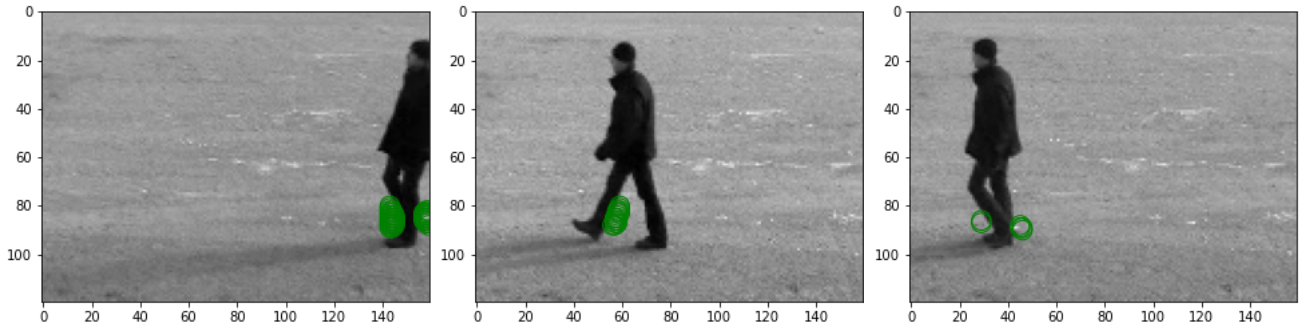
Για τον συγκεκριμένο ανιχνευτή όπως αναφέρει η εκφώνηση επιστρέφουμε τα 500 σημεία με τις μεγαλύτερες τιμές του κριτηρίου σημαντικότητας. Στη συνέχεια επεκτείνουμε τον ανιχνευτή σε πολυκλιμακωτό ακριβώς όπως κάναμε και στο πρώτο εργαστήριο.



Σχήμα 9: Εντοπισμός σημείων ενδιφέροντων για βίντεο με μποξ.



Σχήμα 10: Εντοπισμός σημείων ενδιφέροντων για βίντεο τρεξίματος.



Σχήμα 11: Εντοπισμός σημείων ενδιφέροντων για βίντεο βαδίσματος.

Παρατηρούμε ότι και οι δύο ανιχνευτές καταφέρνουν να εντοπίσουν την κίνηση στα σημεία ενδιαφέροντος (χέρια και πόδια). Ωστόσο, είναι προφανές ότι ο ανιχνευτής $Gabor$ είναι αρκετά πιο ακριβής, αφού σπάνια ¨χάνει¨ σημεία ενδιαφέροντος σε $frame$ στο οποίο εμφανίζεται άνθρωπος. Αντίθετα, τα σημεία ενδιαφέροντος που εντοπίζει ο ανιχνευτής $Harris$ είναι αρκετά πιο ¨αραιά¨, τόσο ως προς το πλήθος των πλαισίων όσο και ως προς την πυκνότητα σε κάθε πλαίσιο. Γενικά, και οι δύο ανιχνευτές ανιχνεύουν ευκολότερα ¨έντονες¨ κινήσεις (πχ. κίνηση γονάτων και αστραγάλων στο τρέξιμο) σε σχέση με πιο αργές κινήσεις (πχ. βάδισμα). Ωστόσο, είναι ικανοί να ανιχνεύσουν και απειροστές κινήσεις σε ορισμένες περιπτώσεις (πχ. μικρή μετακίνηση των ποδιών των ανθρώπων στα βίντεο του μποξ).

2.2 Χωρο-χρονικοί Ιστογραφικοί Περιγραφητές

Γύρω από τα σημεία ενδιαφέροντος που υπολογίσαμε χρησιμοποιούμε χωρο-χρονικούς περιγραφητές οι οποίοι βασίζονται στον υπολογισμό ιστογραμμάτων της HOG (κατευθυντική παράγωγος), και της HOF (οπτική ροή). Οι περιγραφητές αυτοί είναι ικανοί να ταξινομήσουν την κίνηση του ανρθώπου.

Για τον υπολογισμό της $TVL1$ οπτικής ροής σε κάθε πίξελ χρησιμοποιούμε την $DualTVL1$ $OpticalFlowcreate$ Ακόμη, μας δίνεται η συνάρτηση $orientationhistogram$ με την οποία υπολογίζουμε τους τοπικούς περιγραφητές, HOG όταν δίνουμε κατευθυντικές παραγώγους και HOF όταν δίνουμε την κατεύθυνση ροής. Συνενώνοντας τους δύο περιγραφητές δημιουργούμε τον HOG/HOF περιγραφητή.

2.3 Κατασκευή Bag of Visual Words και χρήση Support Vector Machines για την ταξινόμηση δράσεων

Αρχικά διαχωρίζουμε το σύνολο των βίντεο σε σύνολο εκπάιδευσης ($train\ set$) και σύνολο δοκιμής ($test\ set$) με βάση το txt αρχείο που μας δίνεται στο συμπληρωματικό υλικό. Έπειτα, κάνουμε απλή εφαρμογή των συναρτήσεων που μας δίνονται εξάγουμε το $accuracy$ των αλγορίθμων μας.

Μετά από αρκετές δοκιμές, παρατηρούμε ότι το $accuracy$ του $Harris$ αλγορίθμου για HOG κυμαίνεται στο 83%-91.6% για HOF κυμαίνεται στο 75%-83% και για HOG/HOF κυμαίνεται στο 83%-90%.
Ακόμη, το $accuracy$ του $Gabor$ ανιχνευτή για HOG κυμαίνεται στο 83% για HOF κυμαίνεται στο 66%-83% και για HOG/HOF κυμαίνεται στο 66%-83%