

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

Σκοπός του πρώτου μέρους αποτελεί η δημιουργία ενός συστήματος παρακολούθησης προσώπου και χεριών. Συγκεκριμένα, θα γίνεται ανίχνευση μέσω μιας πιθανοτικής κατανομής, διαχωρισμός των περιοχών δέρματος που ανιχνεύθηκαν και στη συνέχεια tracking με τον αλγόριθμο Lucas-Kanade.

1.1 Ανίχνευση Δέρματος Προσώπου και Χεριών

Αρχικά, από ένα αρχείο με δείγματα από δέρμα ανθρώπου, που φαίνεται παρακάτω, έγινε η εκπαίδευση μιας διδιάστατης Γκαουσιανής κατανομής της μορφής:

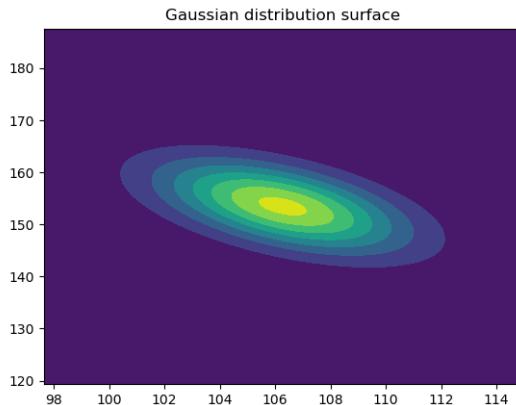
$$P(\mathbf{c} = \text{skin}) = \frac{1}{\sqrt{|\Sigma|(2\pi)^2}} e^{-\frac{1}{2}(\mathbf{c}-\mu)\Sigma^{-1}(\mathbf{c}-\mu)'} \quad \text{Skin samples}$$

υπολογίζοντας το διάνυσμα μέσης τιμής $\mu = [\mu_{C_b} \ \mu_{C_r}]^\top$ και τον πίνακα συνδιακύμανσης $\Sigma = \begin{bmatrix} \sigma_{C_b C_b} & \sigma_{C_b C_r} \\ \sigma_{C_r C_b} & \sigma_{C_r C_r} \end{bmatrix}$.



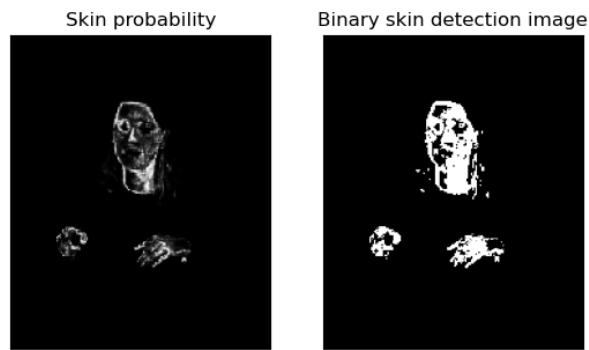
Σχήμα 1: Δείγματα δέρματος για την εκπαίδευση της Γκαουσιανής κατανομής

Παρακάτω φαίνεται η επιφάνεια της Γκαουσιανής με βάση τη μέση τιμή και διακύμανση που προέκυψαν:



Σχήμα 2: Επιφάνεια Γκαουσιανής κατανομής για ανίχνευση δέρματος

Έπειτα, για την ανίχνευση των περιοχών του προσώπου και των χεριών δημιουργήθηκε η δυαδική εικόνα δέρματος με κατωφλιοποίηση πάνω στην εικόνα πιθανότητας:



Σχήμα 3: Εικόνα πιθανότητας και δυαδική εικόνα δέρματος

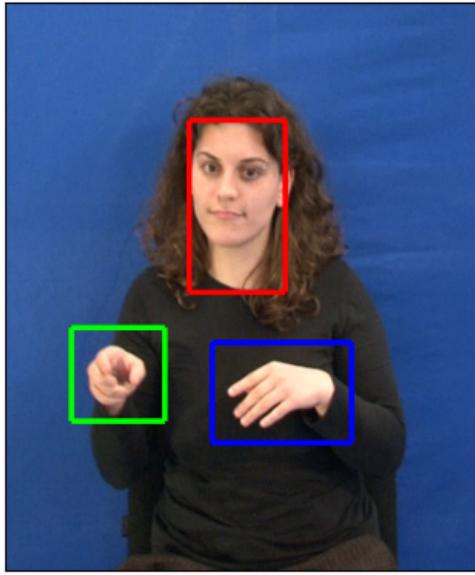
Παρατηρούμε ότι οι περιοχές δεν είναι ομοιογενείς και γι' αυτό το λόγο, κάνουμε μια μορφολογική επεξεργασία της εικόνας. Για κάλυψη των τρυπών, εφαρμόζουμε opening με ένα πολύ μικρό δίσκο και για εξάλειψη των μικρών περιοχών, εφαρμόζουμε closing με ένα μεγάλο δίσκο:



Σχήμα 4: Μορφολογική επεξεργασία δυαδικής εικόνας δέρματος

Στη συνέχεια, έγινε labeling των 3 ενιαίων περιοχών και ορίστηκαν οι οριακές συντεταγμένες των κουτιών που θα περιβάλλουν κάθε περιοχή, οπότε προέκυψε η ακόλουθη ανίχνευση για το πρώτο frame:

Head and Hands Detection



Σχήμα 5: Ανίχνευση κεφαλιού και χεριών στο πρώτο frame

1.2 Παρακολούθηση Προσώπου και Χεριών

Αφού έχει γίνει η ανίχνευση του προσώπου και των χεριών, για να γίνει το tracking κατά τη διάρκεια του βίντεο, όταν εφαρμόσουμε τον αλγόριθμο Lucas-Kanade για να υπολογίσουμε την οπτική ροή κάποιων σημείων ενδιαφέροντος, μεταξύ δύο διαδοχικών frame και με βάση αυτή, να μετακινούμε το κουτί που περιβάλλει την κάθε περιοχή.

1.2.1 Υλοποίηση του Αλγόριθμου των Lucas-Kanade

Η υλοποίηση του αλγορίθμου έγινε με υπολογισμό της οπτικής ροής που προκύπτει από την παρακάτω εξίσωση, η οποία ελαχιστοποιεί το τετραγωνικό σφάλμα μεταξύ δύο διαδοχικών εικόνων:

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

όπου

$$A(x) = [A_1(\mathbf{x}) \quad A_2(\mathbf{x})] = \left[\frac{\partial I_{n-1}(\mathbf{x} + \mathbf{d}_i)}{\partial x} \quad \frac{\partial I_{n-1}(\mathbf{x} + \mathbf{d}_i)}{\partial y} \right]$$

$$E(x) = I_n(\mathbf{x}) - I_{n-1}(\mathbf{x} + \mathbf{d}_i)$$

Το G_ρ υποδηλώνει Γκαουσιανό πυρήνα τυπικής απόκλισης ρ , το ϵ είναι μια μικρή σταθερά, το \mathbf{d}_i το διάνυσμα οπτικής ροής, ενώ τα I_{n-1}, I_n υποδηλώνουν το ζεύγος στο οποίο υπολογίζεται η οπτική ροή.

Η παραπάνω εξίσωση επαναλαμβάνεται αρκετές φορές μέχρι τη σύγκλιση του $\mathbf{d}_{i+1} = \mathbf{d}_i + \mathbf{u}$. Συγκεκριμένα, παίρνουμε την I_n εικόνα και κάνουμε ανίχνευση χαρακτηριστικών (features). Ξεκινάμε με μία αρχική εκτίμηση \mathbf{d}_0 για την οπτική ροή, παίρνουμε τις interpolated εικόνες του I_{n-1} frame και των παραγώγων της και εφαρμόζουμε το

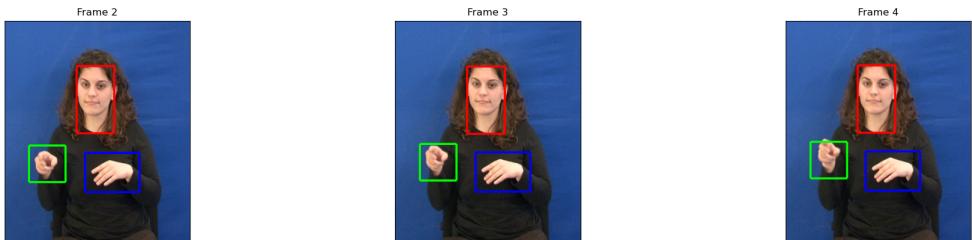
Γκαουσιανό πυρήνα. Κρατάμε τις οπτικές ροές μόνο των features που κάναμε track και ανανεώνουμε την εκτίμηση με το διάνυσμα \mathbf{u} που υπολογίζουμε με βάση την παραπόνω εξίσωση.

Ως αρχική εκτίμηση \mathbf{d}_0 δώσαμε μηδενική και με πειραματισμό του αριθμού των επαναλήψεων, συμπεράναμε ότι τα περισσότερα features δε χρειάστηκαν πάνω από 100-150 επαναλήψεις ενώ κάποια χρειάστηκαν έως και 200-250. Το κριτήριο σύγκλισης που χρησιμοποιήσαμε ήταν η ευκλείδια νόρμα του διανύσματος $\mathbf{u}(\mathbf{x})$ να πέσει κάτω από ένα κατώφλι της τάξης του 10^{-1} .

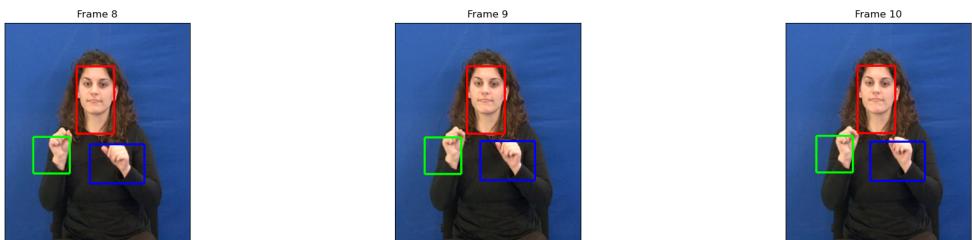
Πειραματιστήκαμε, επίσης, με διαφορετικές τιμές της τυπικής απόκλισης ρ του Γκαουσιανού παραθύρου και παρατηρήσαμε ότι με μικρές τιμές, τα διανύσματα οπτικής ροής είχαν μικρότερο μέτρο από το επιυμητό, παρ' όλο που ίσως ήταν πιο ακριβή σε κατεύθυνση και συνεπώς η μετακινήσεις των χουτιών δεν ήταν αρκετές για να ακολουθηθεί η τροχιά. Το αντίστροφο παρατηρήσαμε με την σταθερά ϵ , που όσο μεγαλύτερη τιμή δίναμε, τόσο εκτός πέφταμε στο tracking.

Οι παράμετροι που χρησιμοποιήσαμε ήταν $\rho = 5$, $\epsilon = 0.01$ και παρακάτω φαίνονται μερικά αποτελέσματα από το tracking:

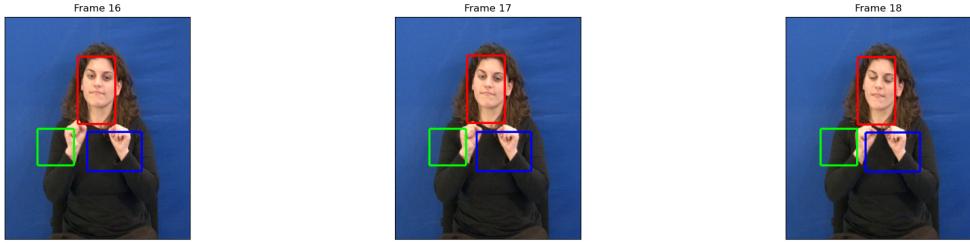
Στα παραδοτέα αρχεία υπάρχουν GIFs με ολόκληρο το tracking στο φάκελο με όνομα plots.



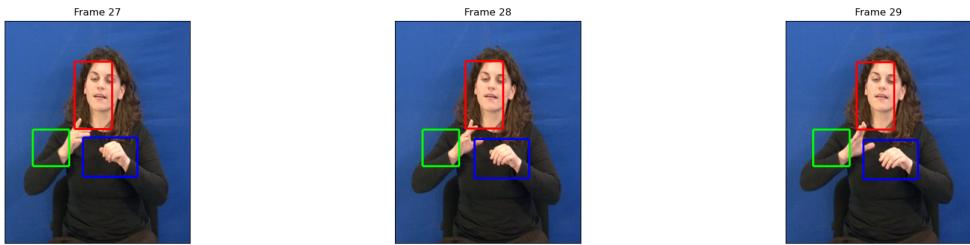
Σχήμα 6: Lucas-Kanade για τα frames 2-4



Σχήμα 7: Lucas-Kanade για τα frames 8-10



Σχήμα 8: Lucas-Kanade για τα frames 16-18



Σχήμα 9: Lucas-Kanade για τα frames 27-29

1.2.2 Υπολογισμός της Μετατόπισης των Παραθύρων από τα Διανύσματα Οπτικής Ροής

Για να μετακινήσουμε τα χουτιά, χρησιμοποιήσαμε ένα χριτήριο ενέργειας των διανυσμάτων οπτικής ροής των features που ανιχνεύσαμε, με σκοπό να πετύχουμε μέγιστη ακρίβεια και να απορρίψουμε outliers. Συγκεκριμένα, υπολογίσαμε την ενέργεια κάθε διανύσματος:

$$\|\mathbf{d}\|^2 = d_x^2 + d_y^2$$

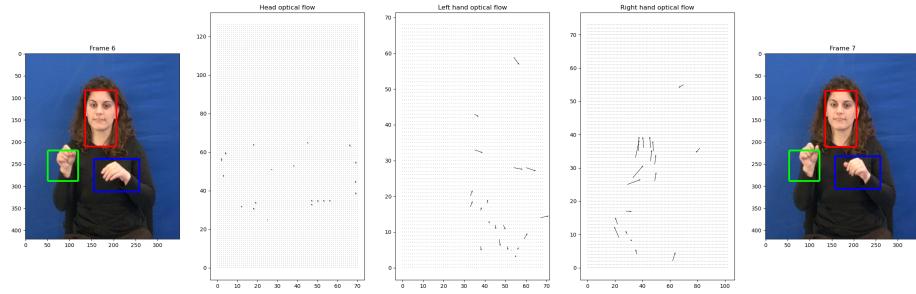
και κρατήσαμε μόνο εκείνα τα features τα οποία είχαν ενέργεια μεγαλύτερη ενός ποσοστού της μέγιστης ενέργειας που βρέθηκε. Έτσι, παίρνοντας το μέσο όρο των διανυσμάτων που κρατήσαμε, προέκυψε το ενιαίο διάνυσμα, κατά το οποίο μετακινούταν κάθε φορά το χουτί.

Το καλύτερο αποτέλεσμα που πήραμε επιτεύχθηκε με κατώφλι 95% της μέγιστης ενέργειας, ενώ με μικρότερα κατώφλια η μετακίνηση των χουτιών ήταν μικρότερη από την επιθυμητή, λόγω πιθανής επίδρασης outliers και σημείων με ομοιόμορφη ή επίπεδη υφή.

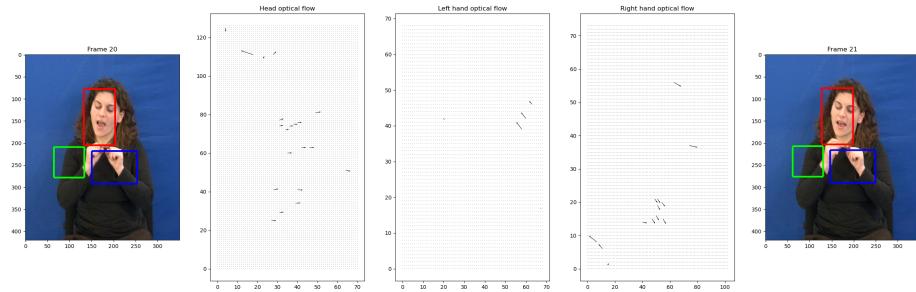
Τέλος, χρησιμοποιώντας ως διάνυσμα μετατόπισης του χουτιού, απλά τον μέσο όρο των αρχικών διανυσμάτων οπτικής ροής, η μετατόπιση ήταν και πάλι μικρότερη και όχι τόσο ακριβής όσο την επιθυμητή και η παρακολούθηση ήταν πολύ χειρότερη.

Παρακάτω φαίνονται μερικά αποτελέσματα από τα διανύσματα οπτικής ροής:

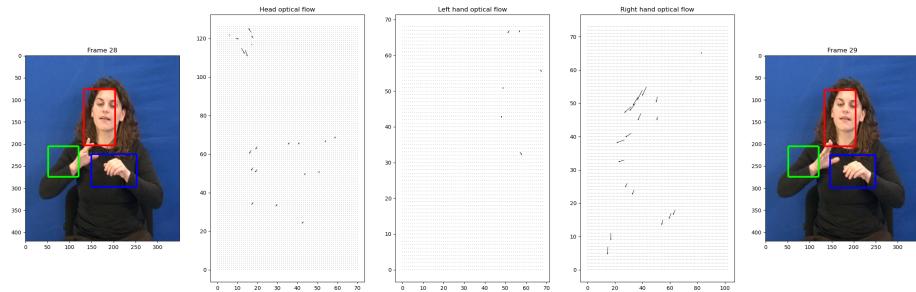
Στα παραδοτέα αρχεία υπάρχουν GIFs με τα quiver ολόκληρου του tracking στο φάκελο με όνομα plots.



Σχήμα 10: Διανύσματα ροής Lucas-Kanade για τα frames 6, 7



Σχήμα 11: Διανύσματα ροής Lucas-Kanade για τα frames 20, 21



Σχήμα 12: Διανύσματα ροής Lucas-Kanade για τα frames 28, 29

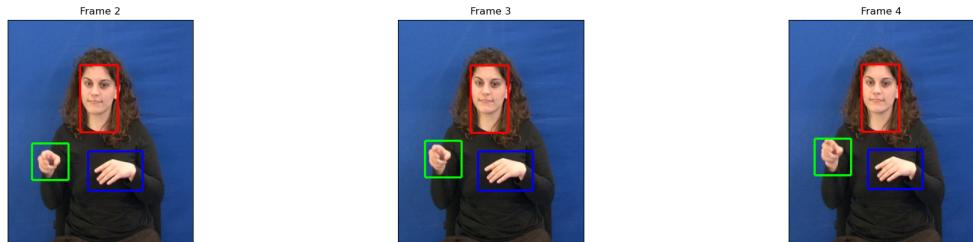
1.2.3 Πολυ-Κλιμακωτός Υπολογισμός Οπτικής Ροής

Γενικά, παρατηρούμε ότι τα αποτελέσματα του παραπάνω αλγορίθμου δεν είναι και πολύ ικανοποιητικά, καθώς το tracking χάνεται μετά από πιο μεγάλες και απότομες κινήσεις. Γι' αυτό, υλοποιούμε τον πολυκλιμακωτό (multi-scale) Lucas-Kanade αλγόριθμο, ο οποίος δημιουργεί Γκαουσιανές πυραμίδες με ύψος όσες κλίμακες ορίζουμε, για τα δύο frame που κάνει το tracking.

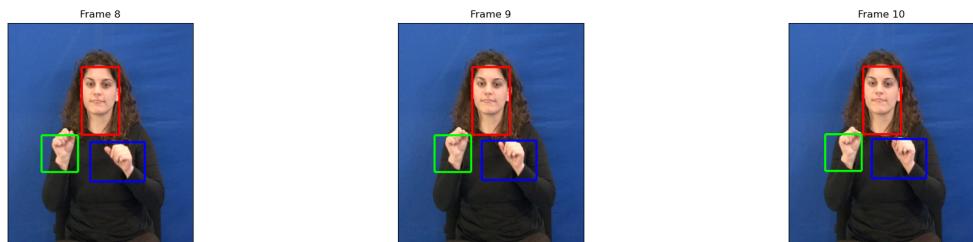
Συγκεκριμένα, μια πυραμίδα με L επίπεδα έχει ως κατώτατο επίπεδο την αρχική εικόνα $I^0 = I$ και τα επόμενα $L - 1$ επίπεδα προκύπτουν από διαδοχικό Gaussian

φιλτράρισμα με παράθυρο της επιλογής μας και υποδειγματοληψία κατά παράγοντα 2 ανα διάσταση σε κάθε επίπεδο. Επομένως, η εικόνα I^M στο επίπεδο M έχει μέγειος $(n_x^M, n_y^M) = (n_x^{M-1}/2, n_y^{M-1}/2)$. Ο αλγόριθμος εφαρμόζεται στην πυραμίδα ξεκινώντας από την κορυφή, έστω επίπεδο L , πάλι με μια αρχική εκτίμηση. Υπολογίζει την υπολοιπόμενη οπτική ροή \mathbf{d}^L μετά από σύγχλιση όπως και στο μονοκλιμακωτό και προωθεί την οπτική ροή των δύο frames στο προηγούμενο επίπεδο $L - 1$. Έπειτα, επαναλαμβάνεται η ίδια διαδικασία, δηλαδή interpolation με μηδενικά για διπλασιασμό των διαστάσεων και μετά υπολογισμός της \mathbf{d}^{L-1} , μέχρι να φτάσουμε στο επίπεδο 0.

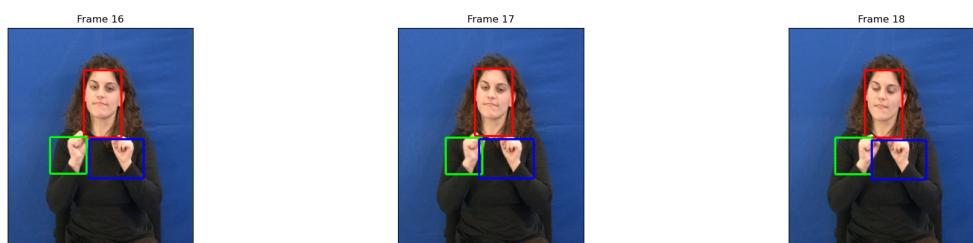
Χρησιμοποιήσαμε ίδιες τιμές για τις παραμέτρους ρ , ϵ ενώ τώρα δημιουργήσαμε πυραμίδες με 4 επίπεδα, δηλαδή είχαμε 4 κλίμακες. Παρακάτω φαίνονται τα αποτελέσματα από το tracking και τα quivers στα ίδια frames:



Σχήμα 13: Multi-scale Lucas-Kanade για τα frames 2-4



Σχήμα 14: Multi-scale Lucas-Kanade για τα frames 8-10



Σχήμα 15: Multi-scale Lucas-Kanade για τα frames 16-18

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

Σκοπός του δεύτερου μέρους αποτελεί ο εντοπισμός χωρο-χρονικών σημείων ενδιαφέροντος από βίντεο 3 κλάσεων (walking,running,boxing), στη συνέχεια εξαγωγή χωρο-χρονικών χαρακτηριστικών και τέλος, αναγνώριση ανθρωπίνων δράσεων.

2.1 Χωρο-χρονικά Σημεία Ενδιαφέροντος

Τα χωρο-χρονικά σημεία ενδιαφέροντος θα εντοπισθούν με δύο μεθόδους, τη μέθοδο Harris και τη μέθοδο Gabor.

2.1.1 3Δ Ανιχνευτής Harris

Ο ανιχνευτής Harris χρησιμοποιεί το παρακάτω 3Δ κριτήριο γωνιότητας, για να εντοπίσει γωνίες σε κάθε frame:

$$H(x, y, t) = \det(M(x, y, t)) - k \cdot \text{trace}^3(M(x, y, t))$$

όπου

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

- $g(x, y, t; s\sigma, s\tau)$: ένας 3Δ Γκαουσιανός πυρήνας με κλίμακες $s\sigma, s\tau$
- L_x, L_y, L_t : οι μερικές παράγωγοι του βίντεο ως προς x, y, t αντίστοιχα
- k : μια παράμετρος κατωφλίου για το κριτήριο γωνιότητας

Για την υλοποίηση του παραπάνω αλγορίθμου, αρχικά εφαρμόσαμε Γκαουσιανό φιλτράρισμα κλίμακας σ στις χωρικές διαστάσεις και κλίμακας τ στη διάσταση του χρόνου. Στη συνέχεια, υπολογίσαμε τις μερικές παραγώγους L_x, L_y, L_t και σχηματίσαμε τα στοιχεία του παραπάνω 3×3 πίνακα.

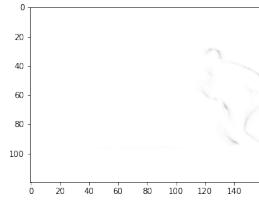
Τέλος, εφαρμόσαμε ξανά Γκαουσιανό φιλτράρισμα, αυτή τη φορά με κλίμακες $s\sigma$ και $s\tau$, αντίστοιχα και υπολογίσαμε το 3Δ κριτήριο γωνιότητας, δηλαδή του ίχνους και της ορίζουσας με element-wise πολλαπλασιασμούς.

Παρακάτω, παρατίθενται κάποια αποτελέσματα του αλγορίθμου για επιλεγμένα βίντεο από κάθε κλάση με εμφάνιση των 8 “καλύτερων” γωνιών σε κάθε frame:

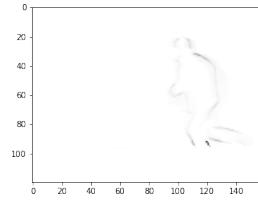
Στα παραδοτέα αρχεία υπάρχουν mp4 βίντεο που παρουσιάζουν τα σημεία ενδιαφέροντος για όλα τα frame με όλους τους συνδυασμούς ανιχνευτών-δράσεων στο φάκελο με όνομα `videos_with_interest_points`.

Βίντεο με τρέξιμο: **person04_running_d4_uncomp.avi**

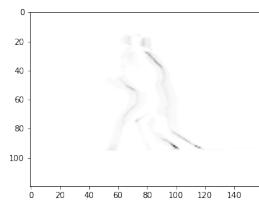
Παράμετροι: $\sigma = 4, \tau = 1.5, s = 2, k = 0.005$



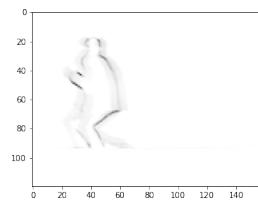
(α') Frame 15



(β') Frame 20

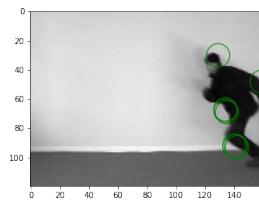


(γ') Frame 25

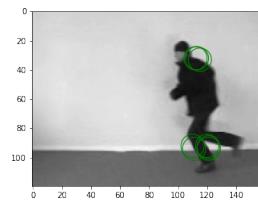


(δ') Frame 30

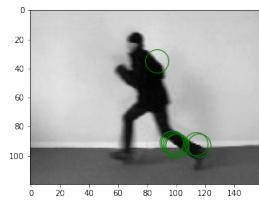
Σχήμα 22: Κριτήριο γωνιότητας Harris σε running



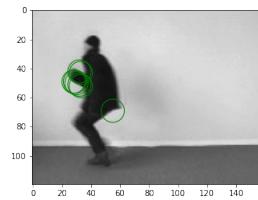
(α') Frame 15



(β') Frame 20



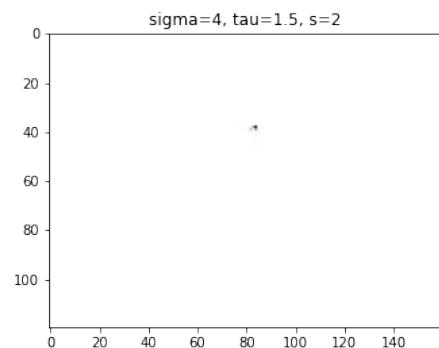
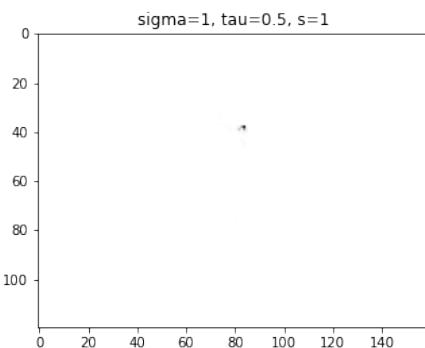
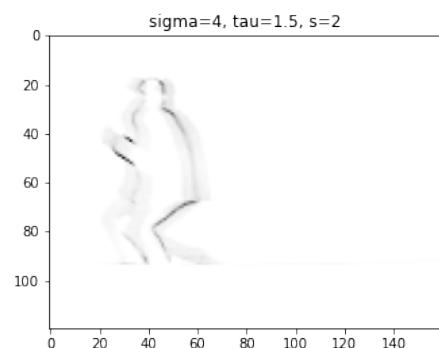
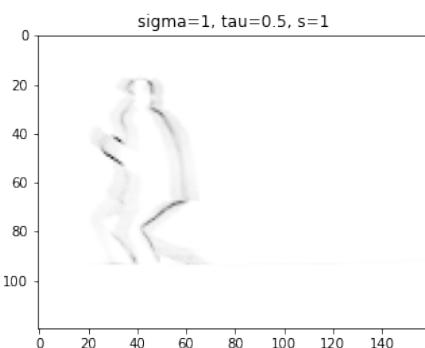
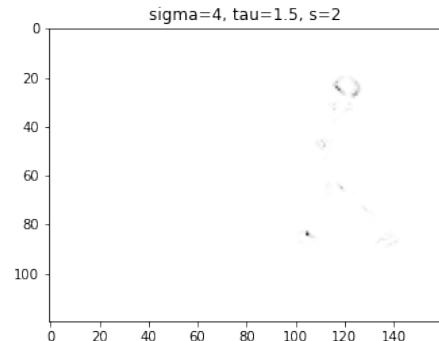
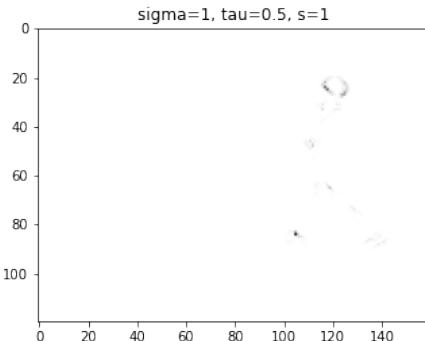
(γ') Frame 25



(δ') Frame 30

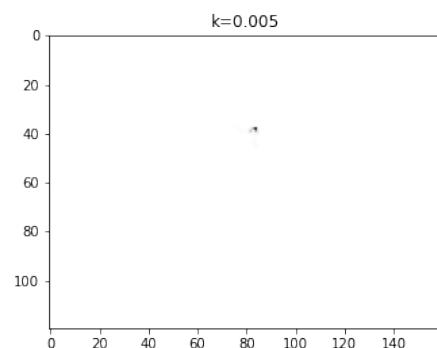
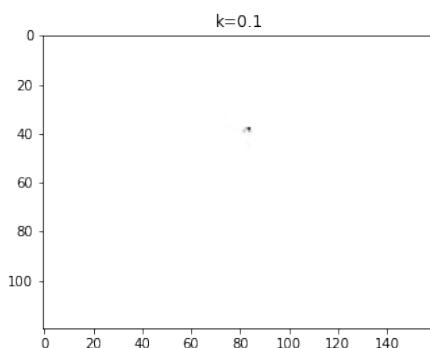
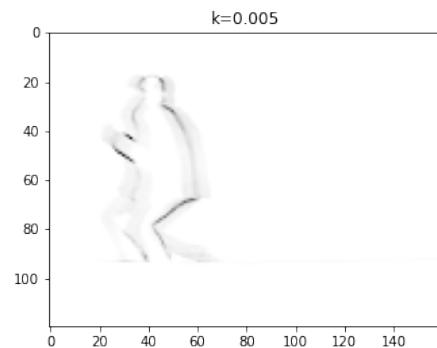
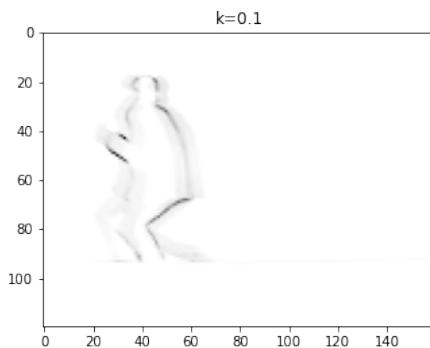
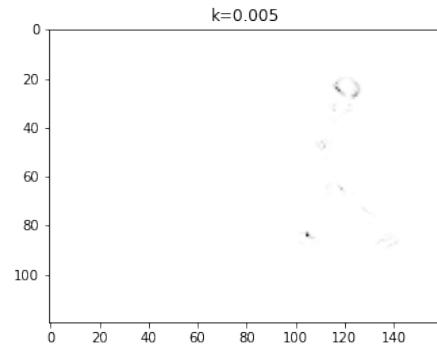
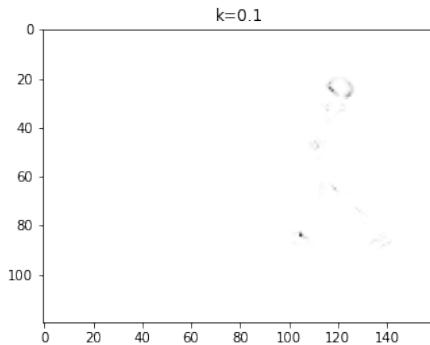
Σχήμα 23: 3Δ Ανίχνευση Harris σε running

Με αλλαγή των κλιμάκων σε $\sigma = 0.5, \tau = 0.5, s = 1$ τα χριτήρια γωνιότητας φαίνεται να μην αλλάζουν σχεδόν καθόλου. Παρακάτω φαίνονται ζευγάρια του ίδιου frame με το αριστερό διάγραμμα στις καινούργιες κλίμακες και το δεξί στις αρχικές:



Δεν παρουσιάζουμε τα frames με σημειωμένα τα σημεία, καθώς θα ήταν τα ίδια αλλά με κύκλους μικρότερης ακτίνας.

Πειραματιζόμενοι και με την παράμετρο για το χατώφλι σε $k = 0.1$, πάλι δεν παρατηρήσαμε αλλαγές που φαίνονται με το μάτι:



Συνεπώς, συμπεραίνουμε πως ο 3Δ ανιχνευτής Harris δίνει αρκετά καλά αποτελέσματα, αφού ανιχνεύει σημεία ενδιαφέροντος του ανθρώπου και στις τρεις ανθρώπινες δράσεις (π.χ. κεφάλι, χέρια, πόδια) και δίνει πρακτικά το ίδιο αποτέλεσμα για διαφορετικές τιμές των παραμέτρων.

2.1.2 Ανιχνευτής Gabor

Ο ανιχνευτής Gabor φιλτράρει το αρχικά το βίντεο στις χωρικές διαστάσεις με 2Δ Γκαουσιανό πυρήνα κλίμακας σ και στη συνέχεια το φιλτράρει στη διάσταση του χρόνου με τα γνωστά φίλτρα Gabor:

$$h_{\text{ev}}(t; \tau, \omega) = \cos(2\pi t\omega)e^{(-\frac{t^2}{2\tau^2})}, \quad h_{\text{odd}}(t; \tau, \omega) = \sin(2\pi t\omega)e^{(-\frac{t^2}{2\tau^2})}$$

όπου $\omega = 4/\tau$.

Για τον προσδιορισμό των σημείων ενδιαφέροντος, χρησιμοποιείται ένα κριτήριο σημαντικότητας που προκύπτει από την τετραγωνική ενέργεια εξόδου:

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

Για την υλοποίηση του αλγορίθμου, φιλτράμε με Γκαουσιανό πυρήνα τις χωρικές διαστάσεις του αρχικού βίντεο και στη συνέχεια ορίσαμε διανύσματα μεγέθους $[-2\tau, 2\tau]$ για να πάρουμε τις κρουστικές αποχρίσεις των φίλτρων.

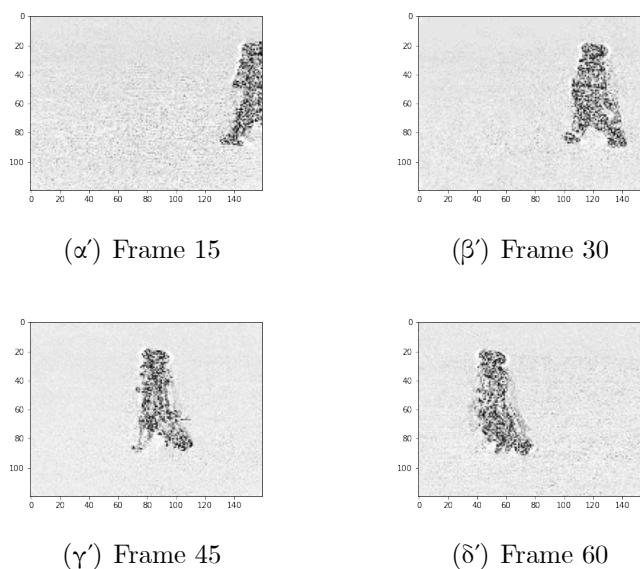
Έπειτα, κανονικοποιήσαμε τις κρουστικές με L1 νόρμα, και τις συνελίξαμε με τη χρονική διάσταση του βίντεο ζεχωριστά, οπότε προέκυψαν δύο .

Τετραγωνίζοντας και μετά αθροίζοντας τα αποτελέσματα, προέκυψε το κριτήριο σημαντικότητας H .

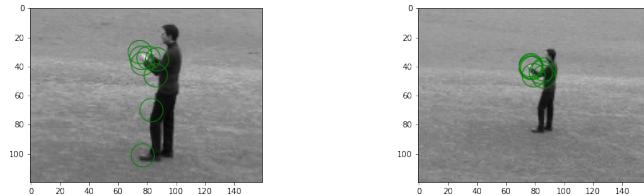
Παρακάτω, παρατίθενται κάποια αποτελέσματα του αλγορίθμου για επιλεγμένα βίντεο από κάθε κλάση με εμφάνιση των 8 “καλύτερων” γωνιών σε κάθε frame:

Βίντεο με περπάτημα: **person13_walking_d3_uncomp.avi**

Παράμετροι: $\sigma = 4, \tau = 1.5$

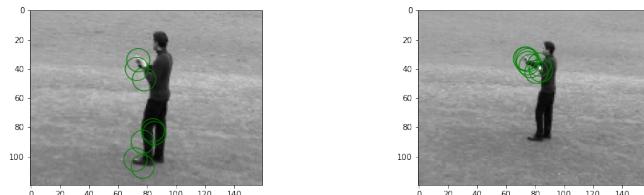


Σχήμα 26: Κριτήριο σημαντικότητας Gabor σε walking



(α') Frame 40

(β') Frame 80

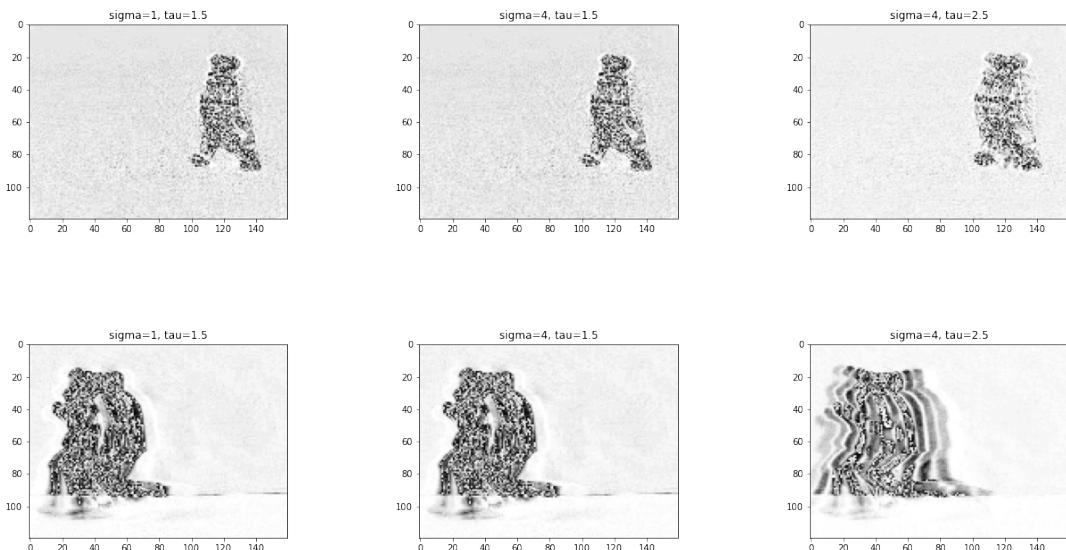


(γ') Frame 120

(δ') Frame 160

Σχήμα 31: Ανίχνευση Gabor σε boxing

Πειραματιζόμενοι με τις παραμέτρους, αρχικά αλλάζοντας το σ , τα αποτελέσματα ήταν πάνω κάτω και πάλι τα ίδια με την αρχική κλίμακα, ενώ αλλάζοντας το τ , συμπεραίνουμε από το αποτέλεσμα ότι με μεγαλύτερο παράθυρο των φίλτρων, ο αλγόριθμος μπορεί να ανιχνεύσει πιο μεγάλη κίνηση:



Συμπεράσματα: Ο ανιχνευτής Gabor δίνει καλύτερα αποτελέσματα από τον Harris, αφού εντοπίζει τις κινήσεις που κάνει ο άνθρωπος, αντί απλώς να ανιχνεύει τις γωνίες του κάθε frame. Συγκεκριμένα, παρατηρούμε ότι στα βίντεο του walking και του running ο Gabor ανιχνεύει ότι κινείται όλο το σώμα (και ο κορμός), ενώ ο Harris κυρίως τα άκρα και το κεφάλι. Στο βίντεο του boxing, παρατηρούμε καλή απόδοση και από τους δύο, όπου ανιχνεύονται ορθά οι κινήσεις των χεριών και λίγες εσφαλμένα στα πόδια. Σε ότι αφορά την ανίχνευση με διαφορετικές κλίμακες, δεν παρατηρήσαμε ιδιαίτερη διαφορά, μόνο ότι η οπτικοποίηση των σημείων ίσως είναι ακριβέστερη.

2.2 Χωρο-χρονικοί Ιστογραφικοί Περιγραφητές

Στο μέρος 2.2 χρησιμοποιήσαμε ως συναρτήσεις κριτηρίου τους ανιχνευτές Harris 3D και Gabor για εύρεση σημείων ενδιαφέροντος από το προηγούμενο ερώτημα και δημιουργήσαμε τους ιστογραφικούς περιγραφητές HOG (Histogram of Oriented Gradients) και HOF (Histogram of Oriented Flow), οι οποίοι βασίζονται στην κατευθυντική παράγωγο και την οπτική ροή αντίστοιχα. Για την υλοποίηση του HOG/HOF περιγραφητή, έγινε απλά συνένωση των χαρακτηριστικών που εξάγουν οι δύο προηγούμενοι, οπότε προέκυπτε ίδιο αριθμός περιγραφητών με διπλάσιο πλήθος χαρακτηριστικών.

Αρχικά, υπολογίσαμε με την συνάρτηση την κατευθυντική παράγωγο ως προς x και y για κάθε frame του βίντεο καθώς και την οπτική ροή σε κάθε frame. Στην συνέχεια, γύρω από τις περιοχές των σημείων ενδιαφέροντος του κάθε στιγμιοτύπου, που εντοπίσαμε με τις μεθόδους Harris 3D και Gabor, εξάγαμε τα ιστογράμματα του εκάστοτε τοπικού περιγραφητή.

Εδώ να επισημάνουμε ότι χρησιμοποιήσαμε μια περιοχή $[4\sigma \times 4\sigma]$ pixel γύρω από το σημείο και αριθμό bins ίσο με 4. Επίσης η κλίμακα της γκαουσιανής κατανομής που χρησιμοποιήθηκε για τον εντοπισμό των σημείων ήταν $\sigma = 4$.

Οι υλοποιήσεις των HOG, HOF βρίσκεται στα αντίστοιχα .py αρχεία.

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

Στο τελευταίο μέρος, ψα χρησιμοποιήσουμε τα χαρακτηριστικά που εξάγουν οι παραπάνω περιγραφητές, ώστε να κάνουμε κατηγοριοποίηση των βίντεο σε 3 κλάσεις.

2.3.1 Δημιουργία train και test set

Τα training και test set, που χρησιμοποιήσαμε ήταν αρκετά ισορροπημένα, αφού περιείχαν συνολικά 36 βίντεο με 12 από κάθε κλάση και 12 με 4 από κάθε κλάση, αντίστοιχα.

2.3.2 Ιστογράμματα Bag of Visual Words

Με την τεχνική του Bag of Visual Words, δημιουργήσαμε ένα οπτικό λεξικό με 400 “λέξεις” (τα κεντροειδή του k-means). Έγινε πειραματισμός, τόσο με την έτοιμη συνάρτηση που μας δόθηκε, όσο και με τη δική μας υλοποίηση που βρίσκεται στα αρχεία `bagOfVisualWords.py` και `histogramFunction.py`.

2.3.3 Ταξινομητής Support Vector Machine

Στη συνέχεια κάναμε κατηγοριοποίηση με χρήση ενός Support Vector Machine, δημιουργώντας labels για τα βίντεο του training και του test set, ανάλογα με το περιεχόμενό τους.

2.3.4 Αποτελέσματα Κατηγοριοποίησης

Τα καλύτερα, αλλά και αρκετά αντιπροσωπευτικά, αποτελέσματα που πήραμε για όλους τους συνδυασμούς φαίνονται στον παρακάτω πίνακα:

Detector	Descriptor	Accuracy	Our BoVW Accuracy
Harris	HOG	66.7%	66.7%
	HOF	91.7%	91.7%
	HOG/HOF	91.7%	91.7%
Gabor	HOG	83.3%	91.7%
	HOF	83.3%	75.0%
	HOG/HOF	100.0%	100.0%

Πίνακας 1: Ποσοστά επιτυχίας της κατηγοριοποίησης για τους συνδυασμούς ανιχνευτών-περιγραφητών-υλοποιήσεων BoVW

Συμπεράσματα: Είναι φανερό ότι ο καλύτερος συνδυασμός είναι ο ανιχνευτής Gabor και ο περιγραφητής HOG/HOF, αφού έχει την καλύτερη ακρίβεια (σχεδόν πάντα 100%, όχι τόσο παράλογο για ένα τόσο μικρό dataset).

Συγκρίνοντας τους ανιχνευτές Harris και Gabor, καλύτερα overall αποτελέσματα δίνει ο Gabor, καθώς φαίνεται να έχει ικανοποιητική ακρίβεια και με τους δύο απλούς περιγραφητές (HOG, HOF), ενώ ο συνδυασμός Harris-HOG δίνει αρκετά μέτρια αποτελέσματα, παρ' όλο που δίνει πολύ καλά με HOF.

Σε ό,τι αφορά τους περιγραφητές, ο HOF δίνει καλύτερα αποτελέσματα, χυρίως, γιατί ο HOG με απλό υπολογισμό παραγώγου, είναι πολύ δύσκολο να περιγράψει σύνθετες κινήσεις, όπως τρέξιμο και περπάτημα. Αντιθέτως ο HOF, έχοντας παραπάνω πληροφορία, είναι ικανός να κάνει καλύτερη περιγραφή μιας δράσης. Τέλος ο HOG/HOF, που είναι συνδυασμός των παραπάνω, είναι αναμενόμενο να έχει τα καλύτερα αποτελέσματα, καθώς έχει χαρακτηριστικά που π.χ. είναι και ανεξάρτητα σε περιστροφές (HOG), αλλά και πληροφορία για οπτική ροή (HOF).

2.3.5 Πειραματισμός με train/test sets

Πειραματιζόμενοι με, όχι τόσο ισορροπημένα σύνολα εκπαίδευσης και δοκιμής, παρατηρήσαμε χειρότερη επίδοση από τον αλγόριθμο. Συγκεκριμένα, χρησιμοποιήσαμε 2 walking και 3 running 6 βίντεο boxing στο test set, οπότε στο training set 14 walking, 13 running, 10 boxing. Παρακάτω φαίνονται τα αποτελέσματα:

Detector	Descriptor	Accuracy
Harris	HOG	36.7%
	HOF	63.6%
	HOG/HOF	81.8%
Gabor	HOG	72.7%
	HOF	81.8%
	HOG/HOF	90.9%

Πίνακας 2: Ποσοστά επιτυχίας με μη-ισορροπημένο σύνολο εκπαίδευσης/δοκιμής

Τα αποτελέσματα που πήραμε είναι φανερά χειρότερα, γεγονός το οποίο οφείλεται ότι το λεξικό που δημιουργούμε προκύπτει από δεδομένα με λιγότερα βίντεο boxing ($10 < 12$), οπότε όταν δοκιμάζεται σε αυτά, πόσο μάλλον όταν είναι και περισσότερα ($6 > 4$), η ακρίβεια θα είναι πολύ χειρότερη. Αναμένουμε πως ο αλγόριθμος ταξινομεί τα βίντεο με walking και running με μεγαλύτερη ακρίβεια.

Αναφορές

- [1] S. Baker and I. Matthews, *Lucas-Kanade 20 years on: A unifying framework*. I. J. of CV, 2004.
- [2] J. Shi and C. Tomasi, *Good Features to Track* IEEE CCVPR, 1994
- [3] R. Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed., 2021