

Ηλιόπουλος Γεώργιος:

03118815

grgslpls1@gmail.com

Σερλής Εμμανουήλ Αναστάσιος:

03118125

manosserlis@gmail.com

Θέμα: Εκτίμηση οπτικής Ροής (Optical Flow) Εξαγωγή Χαρακτηριστικών σε Βίντεο για Αναγνώριση Δράσεων, Συνένωση Εικόνων (Image Stitching)

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

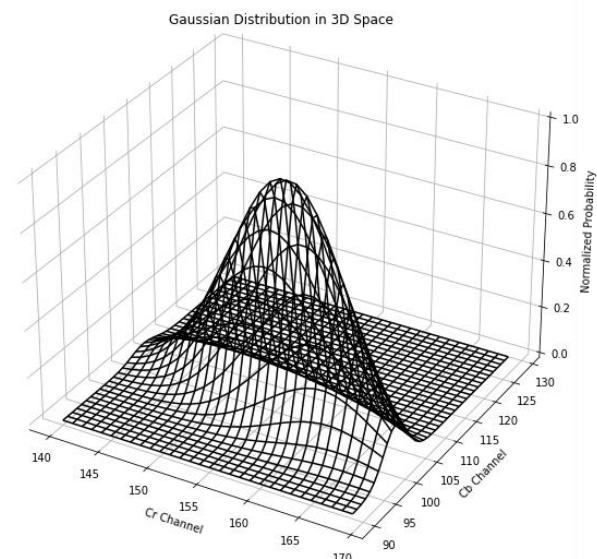
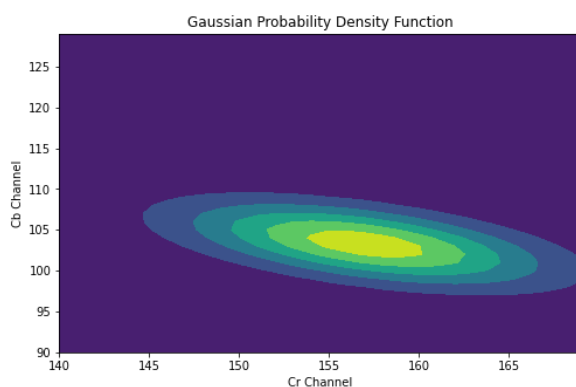
Στόχος του μέρους 1 είναι η υλοποίηση ενός συστήματος παρακολούθησης προσώπου και χεριών, που περιλαμβάνει α) την ανίχνευση των περιοχών ενδιαφέροντος και β) την παρακολούθησή τους σε μία ακολουθία από frames μέσω των υπολογιζόμενων διανυσμάτων οπτικής ροής (από τον αλγόριθμο Lukas Kanade)

1.1 Ανίχνευση Δέρματος και Προσώπου

Αρχικά, καλούμαστε να βρούμε τα διανύσματα μέσης τιμής και συνδιακύμανσης της Gaussian κατανομής που μοντελοποιεί το χρώμα του δέρματος στον χώρο YCbCr. Χρησιμοποιώντας το δοθέν αρχείο skinSamplesRGB.mat λαμβάνουμε τα εξής διανύσματα:

```
mu = [157.0460157126824, 103.27048260381594]
cov = [[ 44.19103128 -11.9310385 ]
       [-11.9310385  11.19574811]]
```

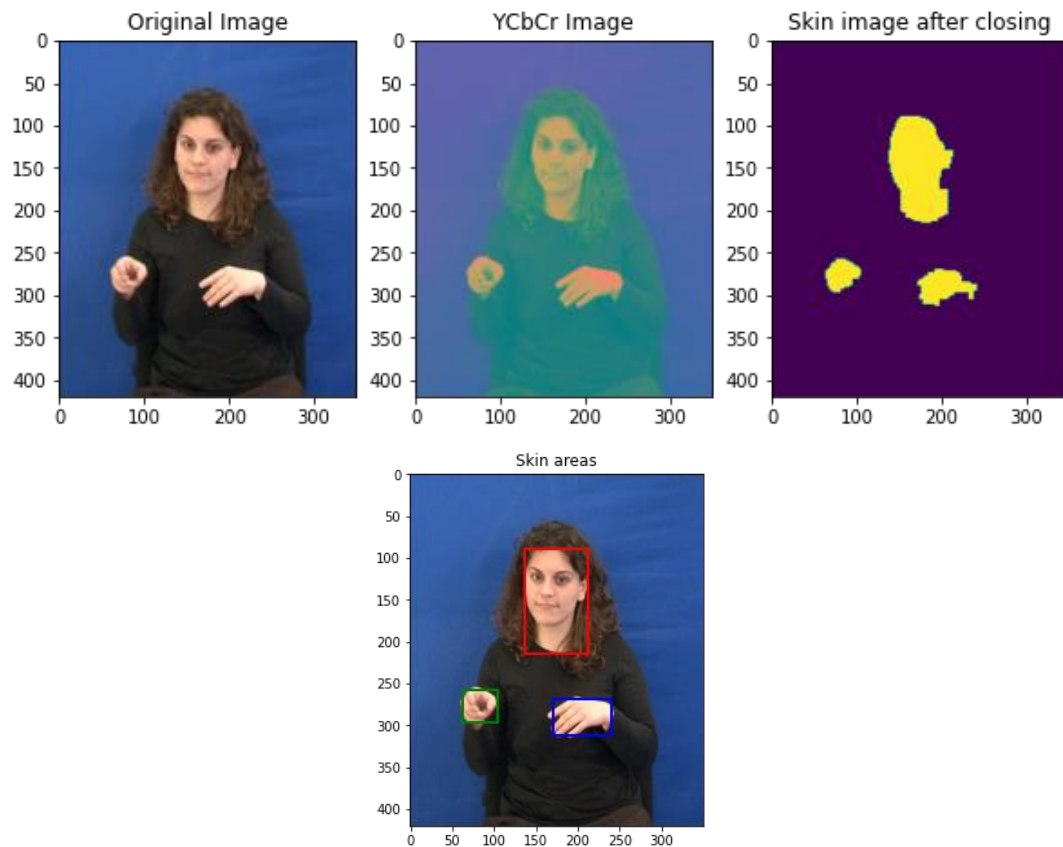
Παρακάτω φαίνονται και τις συναρτήσεις πυκνότητας πιθανότητας(σε 2D και σε 3D Space)



Στην συνέχεια, υλοποιούμε τον ζητούμενο αλγόριθμο `bounds(I,mu,cov)`. Συγκεκριμένα, μετατρέπουμε την BGR εικόνα σε YCrCb image προτού εφαρμόσουμε την γκαουσιανή κατανομή για threshold ανίχνευσης ίσο με 0.0001. Έπειτα, εφαρμόζουμε διαδοχικά opening με [5x5] kernel και closing με [10x10] kernel, για την δημιουργία της τελικής συνεκτικής εικόνας η οποία

θα περιλαμβάνει τα skin areas. Τέλος, χρησιμοποιούμε την εντολή `add_patch` για την οπτική απεικόνιση των `bounding_boxes` στην αρχική εικόνα, ενώ ταυτόχρονα αποθηκεύουμε και τις ζητούμενες τιμές `[x,y,height,width]` για κάθε περιοχή δέρματος.

Παρακάτω παρουσιάζονται τα ενδιάμεσα και τελικά αποτελέσματα της διαδικασίας για την 1η εικόνα του δοθέντος set από frames:



Head Coords: [137, 89, 75, 125]

Left Hand Coords: [63, 257, 41, 39]

Right Hand Coords: [171, 269, 69, 43]

1.2 Παρακολούθηση Προσώπου και Χεριών

Συγκρίνουμε τις τιμές αρχικοποίησης που βρήκαμε για τα `bounding_boxes` με τις τιμές που δίνονται

- Πρόσωπο: [138, 88, 73, 123]
- Αριστερό χέρι: [47, 243, 71, 66]
- Δεξί χέρι: [162, 264, 83, 48]

Παρά τις σχετικά μικρές διαφορές μεταξύ των τιμών μας και των δοθέντων, θα ακολουθήσουμε τις τιμές της εκφώνησης ώστε να ανεξαρτητοποιηθεί η ανάλυση των μερών 1.1 και 1.2 καθώς και τυχόν σφάλματα στην ανάλυση του 1.1.

1.2.1 Υλοποίηση του Αλγορίθμου των Lukas-Kanade

Πραγματοποιήθηκε υλοποίηση σε Python της υπορουτίνας `lk(I1,I2,r0,e)` για τον υπολογισμό της οπτικής ροής μεταξύ των εισαγόμενων εικόνων με βάση τα βήματα της εκφώνησης. Σημειώνεται ότι επιλέξαμε πλήθος επαναλήψεων 100, ενώ για την εξαγωγή των features χρησιμοποιήσαμε την `cv2.goodFeaturesToTrack()` των Shi και Tomashi. Η οπτικοποίηση των αποτελεσμάτων έγινε μέσω της εντολής `quiver` που δίνεται στις οδηγίες

1.2.2 Υπολογισμός της Μετατόπισης των Παραθύρων από τα Διανύσματα Οπτικής Ροής

Πραγματοποιήθηκε υλοποίηση της υπορουτίνας `speed(dx,dy,thr)`, η οποία αρχικά υπολογίζει την μέση τετραγωνική μετατόπιση ως εξής:

$$d = d_x^2 + d_y^2$$

Και στην συνέχεια κάναμε επιλογή των τιμών $[dx, dy]$ για τις οποίες ισχύει ότι $d_x^2 + d_y^2 > d \cdot thr$, με $thr \in [0,1]$.

Τέλος, λαμβάνουμε ως έξοδο της συνάρτησης την μέση οπτική ροή ανά συντεταγμένη, η οποία δίνεται από τις σχέσεις

$$d_{x_{mean}} = \frac{1}{N} \sum d_{x_i}$$
$$d_{y_{mean}} = \frac{1}{N} \sum d_{y_i}$$

1.2.3 Πολυ-Κλιμακωτός Υπολογισμός Οπτικής Ροής

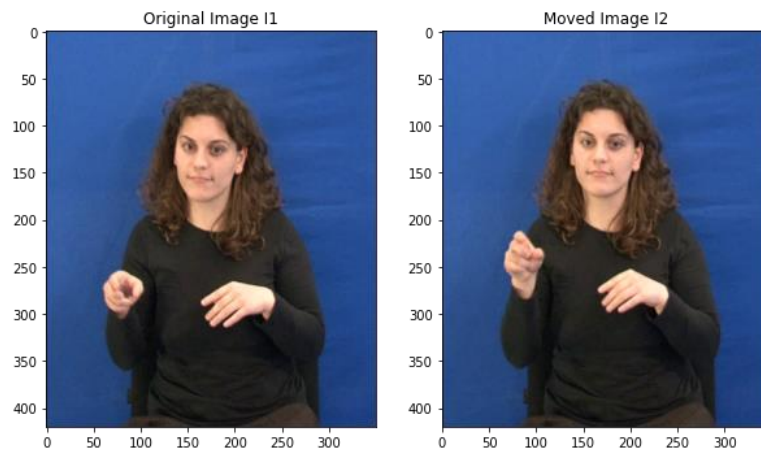
Επεκτείνουμε την υλοποίηση του ερωτήματος 1.2.1 για πολλαπλές κλίμακες. Η υπορουτίνα `multi_lk(I1,I2,r0,e,scales)` αρχικά κάνει `downsampling` τις αρχικές εικόνες τόσες φορές όσα είναι και τα ζητούμενα `scales`. Σημειώνεται ότι πριν από κάθε υποδειγματοληψία-μέσω της εντολής `cv2.pyrDown()`-γίνεται βαθυπερατό gaussian φιλτράρισμα για αποφυγή φαινομένων `aliasing`.

Στην συνέχεια, ξεκινάμε από την οπτική ροή των υποδειγματοληπτημένων εικόνων οι οποίες αποτελούν αρχική συνθήκη για την κλήση της υπορουτίνας `lk()` για τον υπολογισμό των νέων οπτικών ροών. Σημειώνεται ότι πριν την μετάβαση στην επόμενη κλίμακα, οι οπτικές ροές της προηγούμενης γίνονται `reshaped` στις διαστάσεις της επόμενης κλίμακας και πολλαπλασιάζονται με 2.

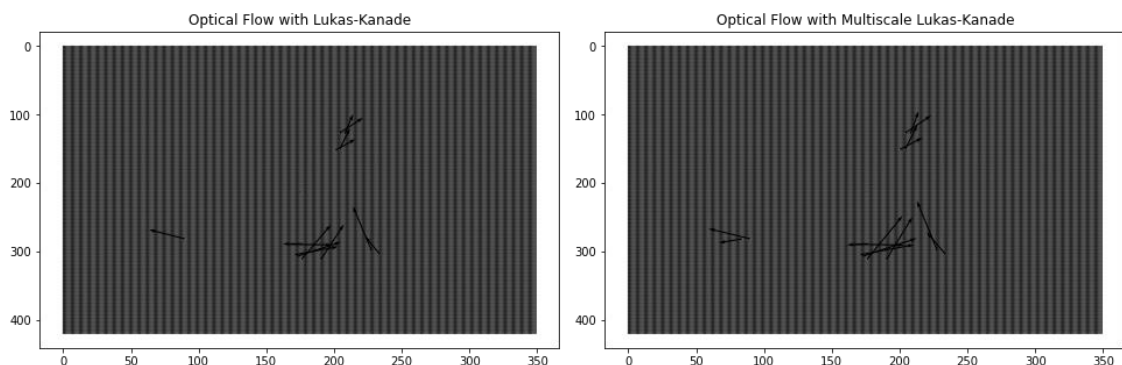
1.2.4 Παρουσίαση αποτελεσμάτων μεθόδων

A. Έλεγχος υπορουτίνων για τα Frames 1 και 5

Αρχικά, θα γίνει αναλυτική παρουσίαση των αποτελεσμάτων των άνωθι συναρτήσεων για τις εικόνες 1 και 5 του βίντεο στις οποίες η βασική διαφορά έγκειται στην κίνηση του αριστερού-όπως το βλέπουμε εμείς-χεριού προς τα πάνω και δεξιά.



Παρατίθενται τα συνολικά αποτελέσματα για σταθερές τιμές παραμέτρων $\rho_0=5$
 $e=0.1$, $scales=5$ και $thr=0.5$



```
dx_mean with lk = -3.1042756210297524
dy_mean with lk = 5.110704538777008
dx_mean with Multi_scale_lk = -3.755219651306578
dy_mean with Multi_scale_lk = 6.079692032172258
```

Παρατηρούμε ότι και οι 2 αλγόριθμοι δίνουν ορθό αποτέλεσμα μιας και τα διανύσματα οπτικής ροής που προκύπτουν έχουν-στην πλειονότητά τους-κατεύθυνση αυτήν την απεικονιζόμενης μετατόπισης. Παρόλα αυτά, η πολυκλιμακωτή έκδοση Lukas-Kanade όχι μόνο βρίσκει ορισμένες ακόμα ακμές οπτικής ροής αλλά και οδηγεί σε κατ'απόλυτον μεγαλύτερες τιμές μέσης μετατόπισης ανά άξονα.

Τα παραπάνω μας οδηγούν στο συμπέρασμα ότι ο πολυκλιμακωτός αλγόριθμος είναι πιο αξιόπιστος για την ανίχνευση της κίνησης μεταξύ 2 frames, κάτι που θα επαληθευτεί παρακάτω και κατά το motion traction σε όλο το video.

B. Ανάλυση παραμέτρων $\rho_0, e, thr, scales$

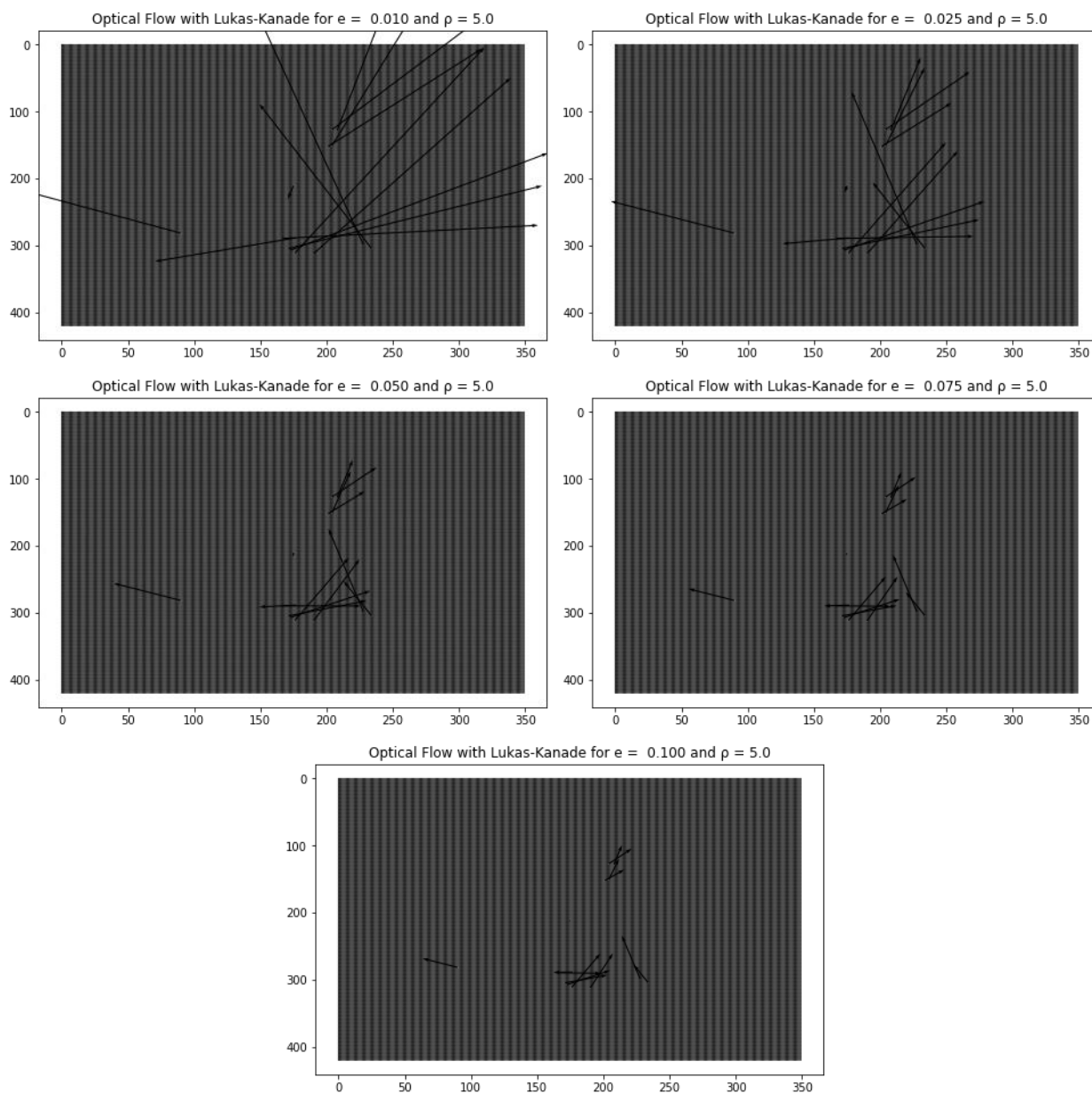
Στην συνέχεια θα τρέξουμε τις υπορουτίνες `lk()`, `speed()` και `multi_lk()` για πολλαπλές τιμές των παραμέτρων. Συγκεκριμένα, δημιουργήσαμε τα εξής 4 διανύσματα

- $\rho_0=[1,2,3,4,5]$
- $e=[0.01,0.025,0.05,0.075,1]$
- $scales=[1,2,3,4,5]$
- $thr=[0.001,0.1,0.2,0.5,0.75]$

Σημειώνεται ότι για λόγους συντομίας η ανάλυση για τις παραμέτρους ρ_0, e θα γίνει στον μονοκλιμακωτό lk, για την scales στον πολυκλιμακωτό lk ενώ για το threshold στην υπορουτίνα speed()

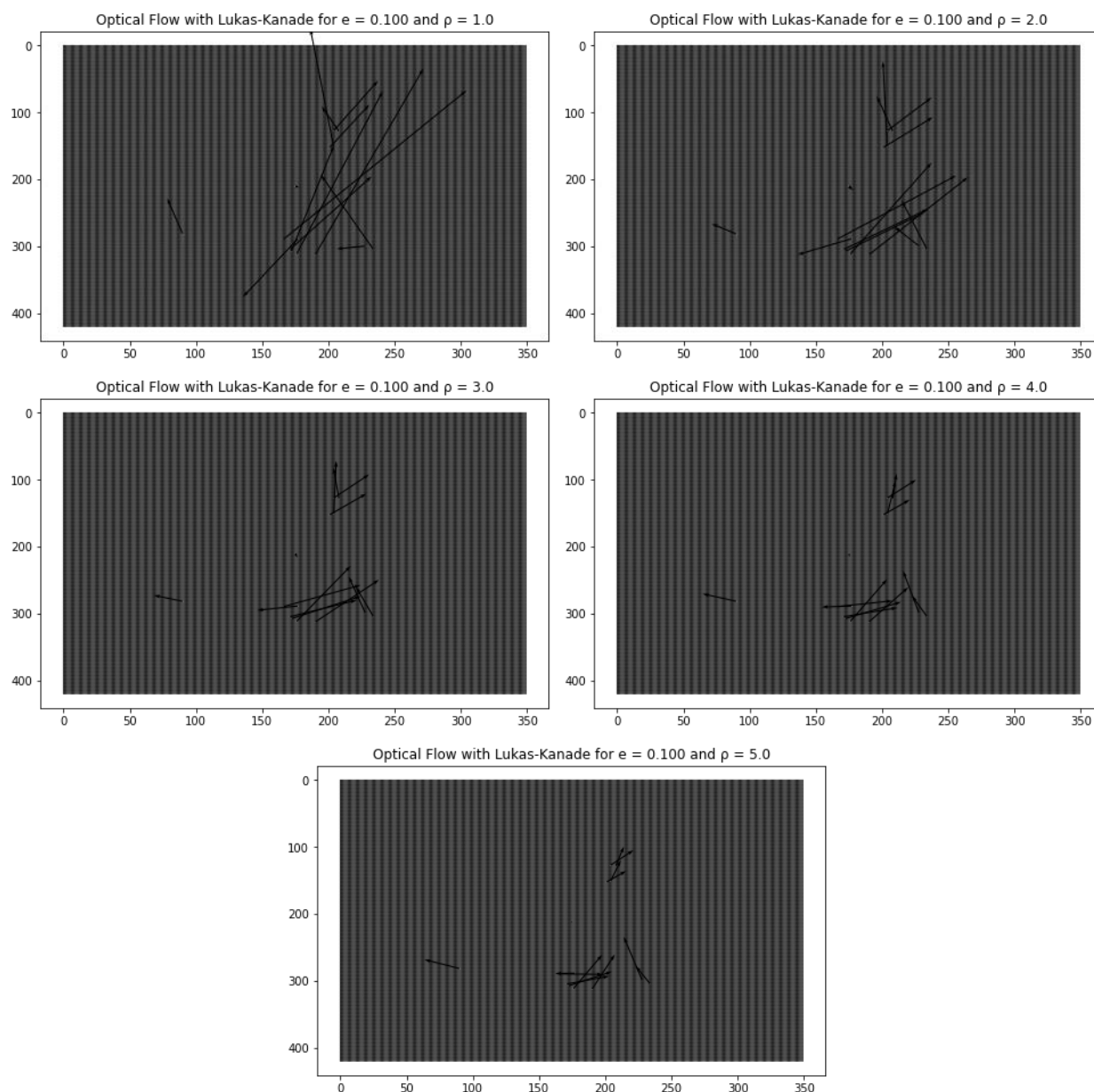
B1. Αποτελέσματα ανάλυσης για την σταθερά κανονικοποίησης e

Η εν λόγω σταθερά έχει σκοπό να βελτιώσει το αποτέλεσμα σε επίπεδες περιοχές με μειωμένη υφή και άρα μειωμένη πληροφορία για τον υπολογισμό της οπτικής ροής. Άρα, αναμένουμε ότι αύξηση της σταθεράς θα οδηγεί σε πιο “ευσταθή” αποτελέσματα, γεγονός που επαληθεύεται παρακάτω.



B2. Αποτελέσματα ανάλυσης για την παράμετρο ρ_0 της Gaussian συνέλιξης:

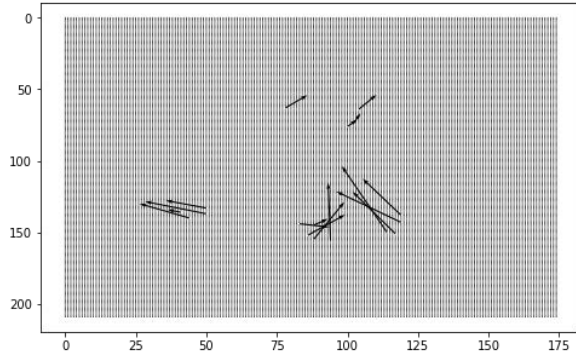
Στα αποτελέσματα που ακολουθούν, παρατηρούμε ότι-ομοίως με την παράμετρο e -μικρές τιμές της παραμέτρου ρ_0 μειώνουν την αξιοπιστία των αποτελεσμάτων της μεθόδου, μιας και προκύπτουν υπερβολικά μεγάλες ακμές που δεν αντικατοπτρίζουν την πραγματική μετατόπιση από το ένα frame στο επόμενο. Αντίθετα, μεγαλύτερες τιμές (όπως $\rho_0=4$ ή $\rho_0=5$) οδηγούν σε πολύ πιο ευκρινή αποτελέσματα:



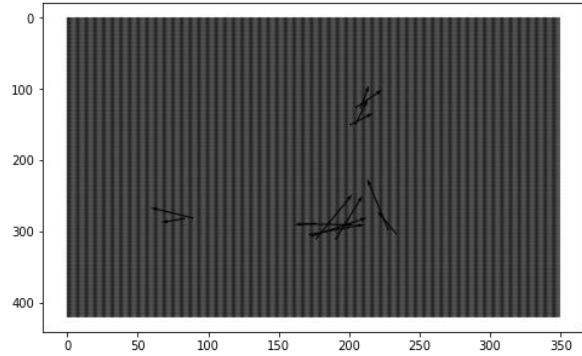
B3. Αποτελέσματα ανάλυσης για την παράμετρο scales:

Αλλάζοντας την παράμετρο scales τον πολυκλιμακωτό Lukas-Kanade, παρατηρήσαμε την αισθητή διαφοροποίηση μεταξύ των αποτελεσμάτων για $scales=1$ και $scales>1$. Αξίζει να σημειωθεί,ωστόσο, ότι οι διαφορές των διανυσμάτων ροής για $scales>1$ είναι πρακτικά αμελητέες, γεγονός που υποδηλώνει την σύγκλιση του πολυκλιμακωτού Lukas-Kanade για $scales=2$.

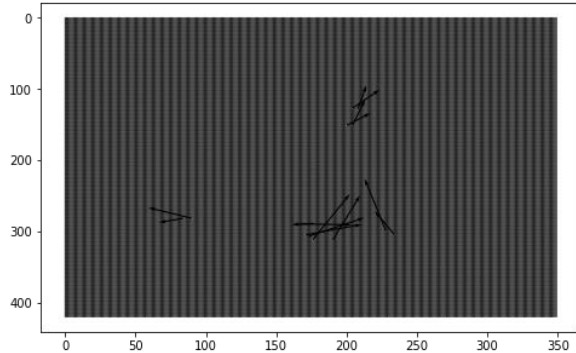
Optical Flow with Multiscale Lukas-Kanade for $e = 0.100$, $\rho = 5.0$ and scales = 1.000



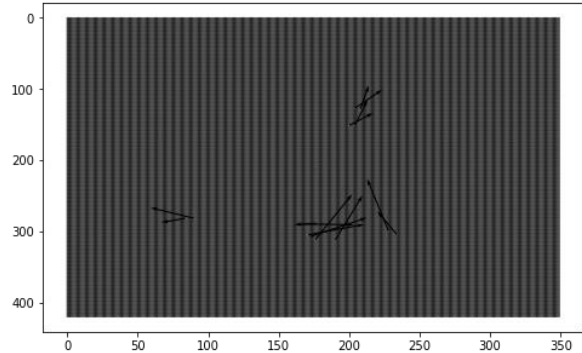
Optical Flow with Multiscale Lukas-Kanade for $e = 0.100$, $\rho = 5.0$ and scales = 2.000



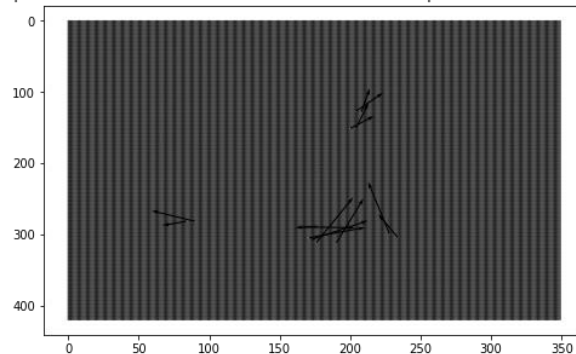
Optical Flow with Multiscale Lukas-Kanade for $e = 0.100$, $\rho = 5.0$ and scales = 3.000



Optical Flow with Multiscale Lukas-Kanade for $e = 0.100$, $\rho = 5.0$ and scales = 4.000



Optical Flow with Multiscale Lukas-Kanade for $e = 0.100$, $\rho = 5.0$ and scales = 5.000



B4. Αποτελέσματα ανάλυσης για την παράμετρο threshold:

Τέλος, εφαρμόσαμε την μέθοδο υπολογισμού της μέσης μετατόπισης για τις διάφορες τιμές της τιμής threshold. Παρατηρήσαμε ότι-τόσο σε μία όσο και σε πολλαπλές κλίμακες-αύξηση της παραμέτρου οδηγεί σε κατ'απόλυτον αύξηση της μέσης μετατόπισης ανά άξονα.

```
e = 0.100 , ρ = 5.000 , threshold = 0.001 , dx = -1.740 , dy = 3.966
e = 0.100 , ρ = 5.000 , threshold = 0.100 , dx = -1.877 , dy = 4.304
e = 0.100 , ρ = 5.000 , threshold = 0.200 , dx = -2.491 , dy = 4.734
e = 0.100 , ρ = 5.000 , threshold = 0.500 , dx = -3.104 , dy = 5.111
e = 0.100 , ρ = 5.000 , threshold = 0.750 , dx = -4.684 , dy = 5.430
```

```
e = 0.100 , ρ = 5.000 , scales = 5.000 , threshold = 0.001 , dx = -1.658 , dy = 4.242
e = 0.100 , ρ = 5.000 , scales = 5.000 , threshold = 0.100 , dx = -1.781 , dy = 4.573
e = 0.100 , ρ = 5.000 , scales = 5.000 , threshold = 0.200 , dx = -2.955 , dy = 5.582
e = 0.100 , ρ = 5.000 , scales = 5.000 , threshold = 0.500 , dx = -3.755 , dy = 6.080
e = 0.100 , ρ = 5.000 , scales = 5.000 , threshold = 0.750 , dx = -5.621 , dy = 6.497
```


Γ. Motion Traction of Frames

Τελευταίο κομμάτι της ανάλυσης περιλαμβάνει την εφαρμογή των άνωθεν αλγορίθμων για την ανίχνευση της μετατόπισης των διαφόρων περιοχών ενδιαφέροντος. Συγκεκριμένα, υλοποιήσαμε την υπορουτίνα `motion_traction(parts,algo)`, της οποίας οι είσοδοι κωδικοποιούνται ως εξής:

- Part = 0 -> Left Hand
- Part = 1 -> Right Hand
- Part = 2 -> Head

- Algo = 0 -> Lukas-Kanade
- Algo = 1 -> MultiScale Lukas-Kanade

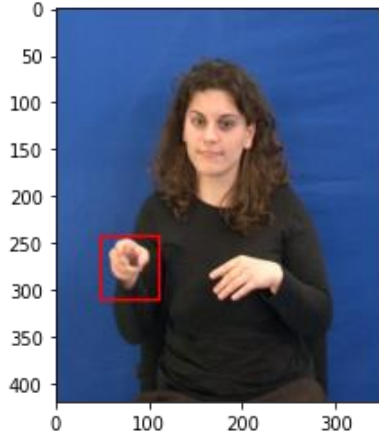
Εν περιλήψει, η ρουτίνα υπολογίζει την οπτική ροή μεταξύ δύο διαδοχικών frames, καθώς και την μέση μετατόπιση μέσω της `speed()`, μετατοπίζοντας εν συνεχεία αναλόγως το προηγούμενο `boundingBox`.

Σημειώνεται ότι χρησιμοποιήθηκαν παράμετροι $\rho_0=7$, $\text{thr}=0.5$, $e=0.1$, $\text{scales}=5$. Η αύξηση της παραμέτρου του γκαουσιανού φίλτραρίσματος ρ_0 έγκειται στο γεγονός ότι για μικρότερες τιμές του ρ_0 λαμβάναμε αισθητά χειρότερα αποτελέσματα, με τα `boundingBoxes` να αποκλίνουν από την περιοχή ενδιαφέροντος μετά από λίγα frames.

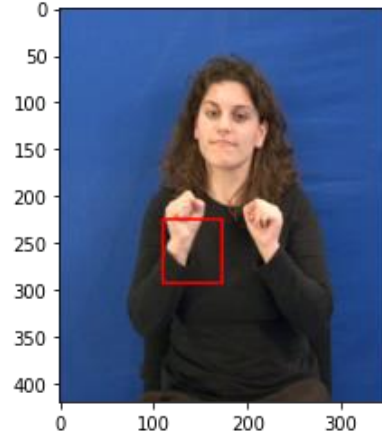
Επιπλέον, λόγω του μεγάλου μεγέθους των δοθέντων `heights` και `widths` για κάθε περιοχή δέρματος, πραγματοποιήθηκε εντός της ρουτίνας μία μείωση των εν λόγω παραμέτρων κατά 3 για τα χέρια και κατά 2 για το κεφάλι, γεγονός που οδήγησε σε βελτιστοποίηση των αποτελεσμάτων μας.

Παρουσιάζονται τα αποτελέσματα για τις συναρτήσεις `motion_traction(0,0)` και `motion_traction(0,1)` και για τα frames 1. Από αυτά παρατηρούμε το αναμενόμενο: ότι παρά την σχετική ακρίβεια και των 2 αλγορίθμων για την ανίχνευση των κινήσεων του αριστερού χεριού, ο πολυκλιμακωτός Lukas-Kanade κρατάει “επαφή” με το αριστερό χέρι καθόλη την διάρκεια του video, κάτι που αδυνατεί να καταφέρει ο απλός Lukas-kanade, συγκεκριμένα στα τελευταία frames.

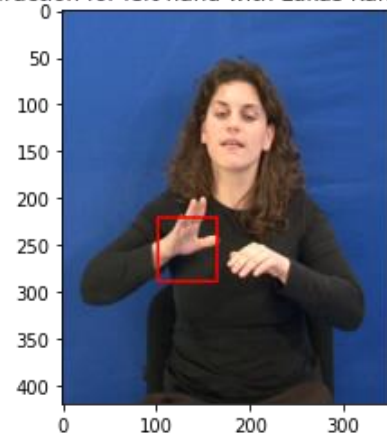
Motion Traction for left hand with Lukas Kanade,frame 1



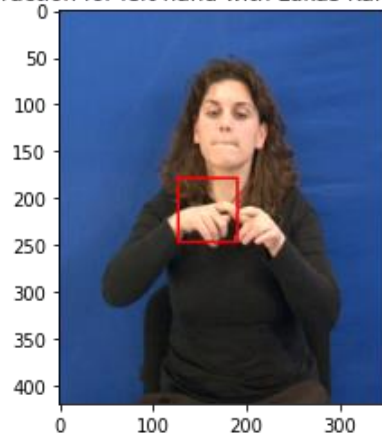
Motion Traction for left hand with Lukas Kanade,frame 15



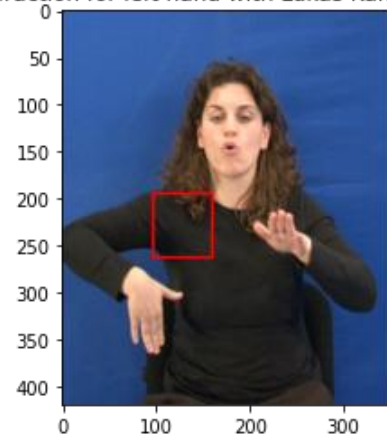
Motion Traction for left hand with Lukas Kanade,frame 30



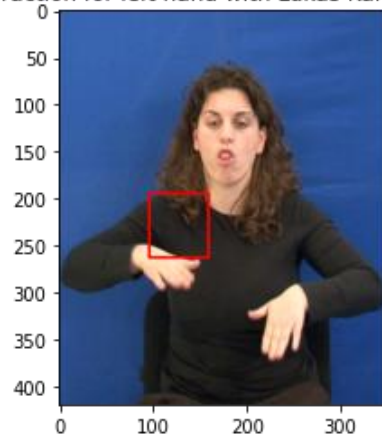
Motion Traction for left hand with Lukas Kanade,frame 45



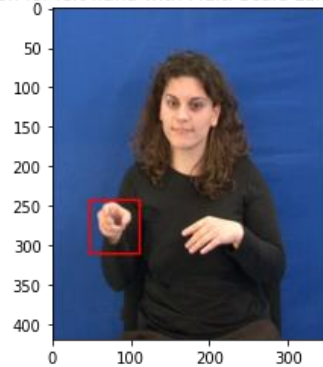
Motion Traction for left hand with Lukas Kanade,frame 60



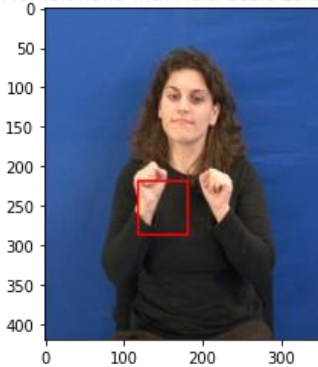
Motion Traction for left hand with Lukas Kanade,frame 64



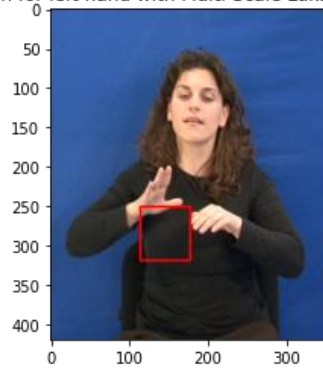
Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 1



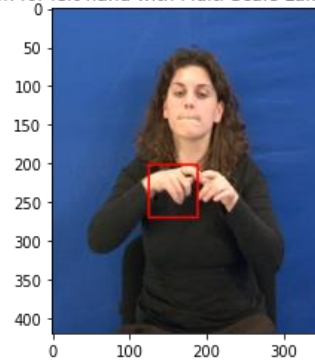
Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 15



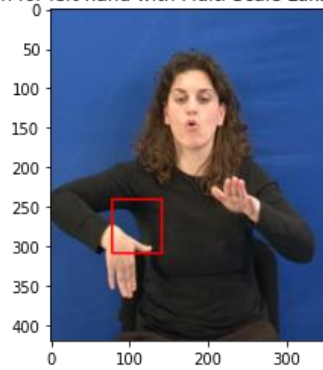
Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 30



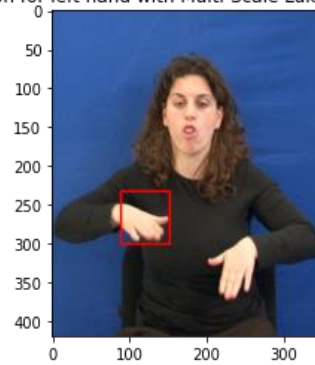
Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 45



Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 60



Motion Traction for left hand with Multi-Scale Lukas Kanade,frame 64



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

Στο μέρος 2 του εργαστηριακού project #2 θα προσπαθήσουμε να κατηγοριοποιήσουμε βίντεο σε 3 κατηγορίες ανθρωπίνων δράσεων: σε τρέξιμο, περπάτημα και μποξ. Για να γίνει αυτό αρχικά εξάγουμε χωρο-χρονικά χαρακτηριστικά από τα βίντεο. Οι τοπικές αναπαραστάσεις περιγράφουν το προς παρατήρηση αντικείμενο με μια σειρά από τοπικούς περιγραφητές που υπολογίζονται σε γειτονιές ανιχνευθέντων σημείων ενδιαφέροντος. Τελικά, η συλλογή των τοπικών χαρακτηριστικών ενσωματώνεται σε μια τελική αναπαράσταση global representation (π.χ. bag of visual words) ικανή να αναπαραστήσει τη στατιστική κατανομή τους και να προχωρήσει στα επόμενα στάδια της αναγνώρισης.

2.1 Χωροχρονικά Σημεία Ενδιαφέροντος

Στην παρούσα άσκηση θα υλοποιήσουμε δύο ανιχνευτές: 1) Harris detector και 2) Gabor detector.

2.1.1) Ο ανιχνευτής Harris αποτελεί μία προέκταση σε 3 διαστάσεις του ανιχνευτή γωνιών Harris-Stephens που υλοποιήσαμε στην 1^η εργαστηριακή άσκηση προσθέτοντας έναν 2Δ τανυστή και τη χρονική παράγωγο. Αρχικά για κάθε voxel του βίντεο υπολογίζουμε τον ακόλουθο πίνακα M .

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

Στην ουσία ομαλοποιούμε χωροχρονικά με γκαουσιανά φιλτραρίσματα (χωρική κλίμακα σ και χρονική κλίμακα τ) το βίντεο. Τις παραγώγους τις υπολογίζουμε εφαρμόζοντας συνέλιξη με τον πυρήνα κεντρικών διαφορών $[-1 \ 0 \ 1]^T$ προσαρμοσμένο κάθε φορά στην κατάλληλη διάσταση. Πιο συγκεκριμένα για παράδειγμα το L_x υπολογίστηκε ως εξής:

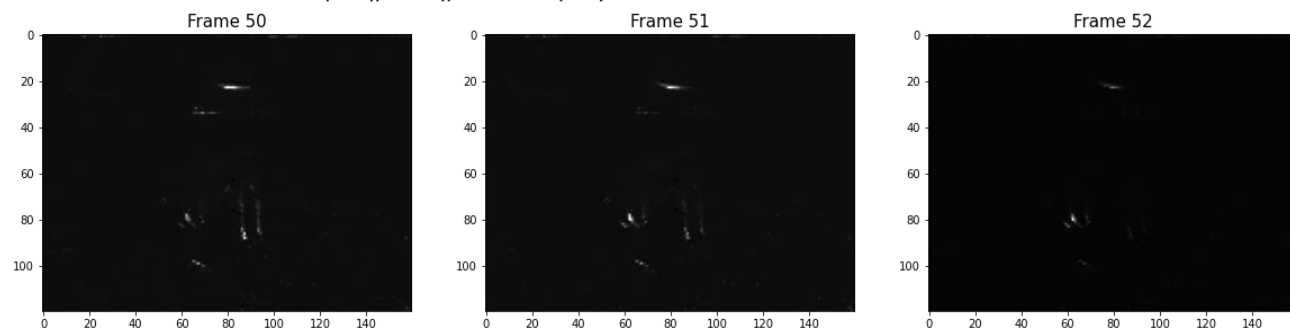
$$dx = [-1 \ 0 \ 1]^T, L_x = \text{convolve}(\text{Input}, dx)$$

Στη συνέχεια υπολογίστηκε το κριτήριο σημαντικότητας ως:

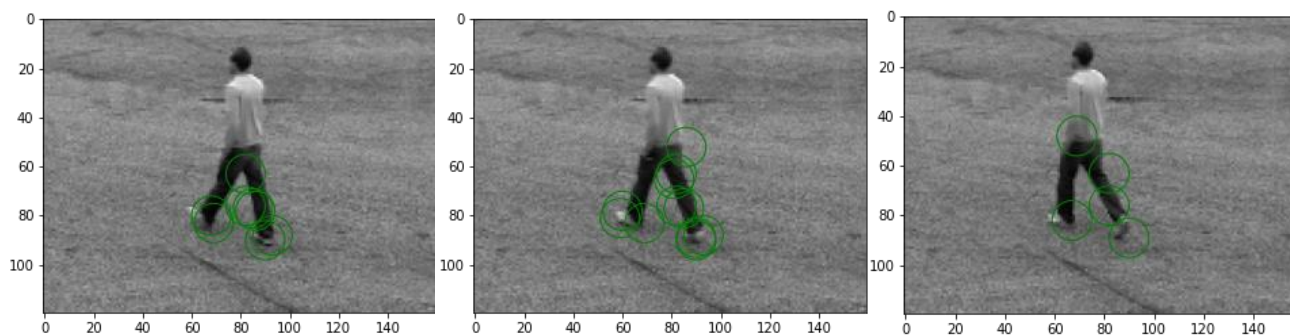
$$H(x, y, t) = \det(M(x, y, t)) - k \cdot \text{trace}^3(M(x, y, t))$$

Εν τέλει υπολογίσαμε τα σημεία ενδιαφέροντος ως τα τοπικά μέγιστα του κριτηρίου H . Για την απεικόνιση χρησιμοποιήθηκε η δοσμένη συνάρτηση `show_detection`. Ακολουθούν κάποια αντιπροσωπευτικές εικόνες των αποτελεσμάτων του Harris Detector για κάθε μία από τις κλάσεις των βίντεο.

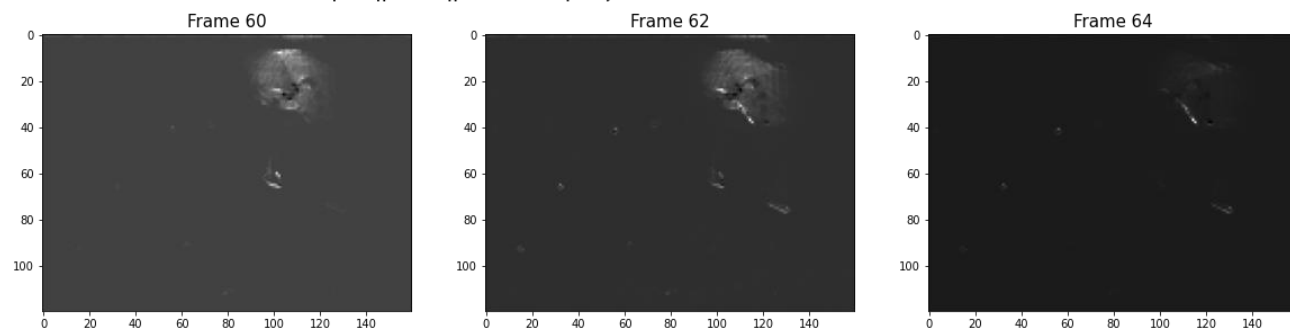
Κριτήριο Σημαντικότητας H - walk video 1 - Harris Detector



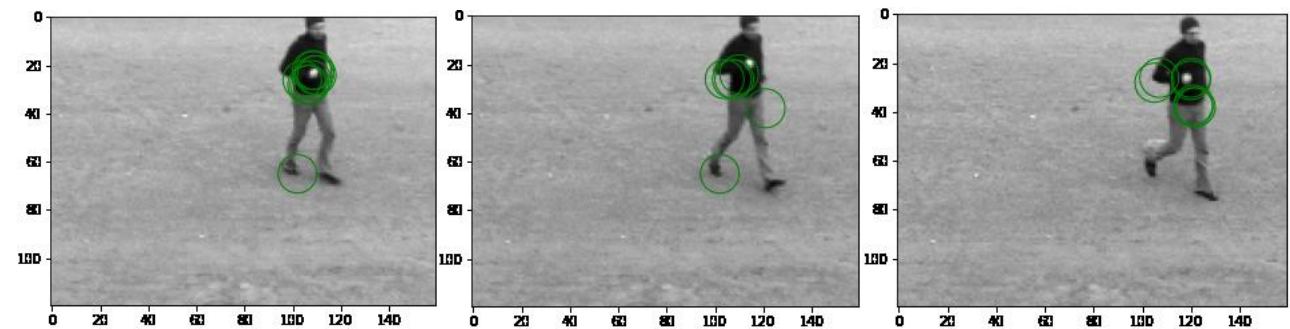
Ανιχνευθέντα σημεία



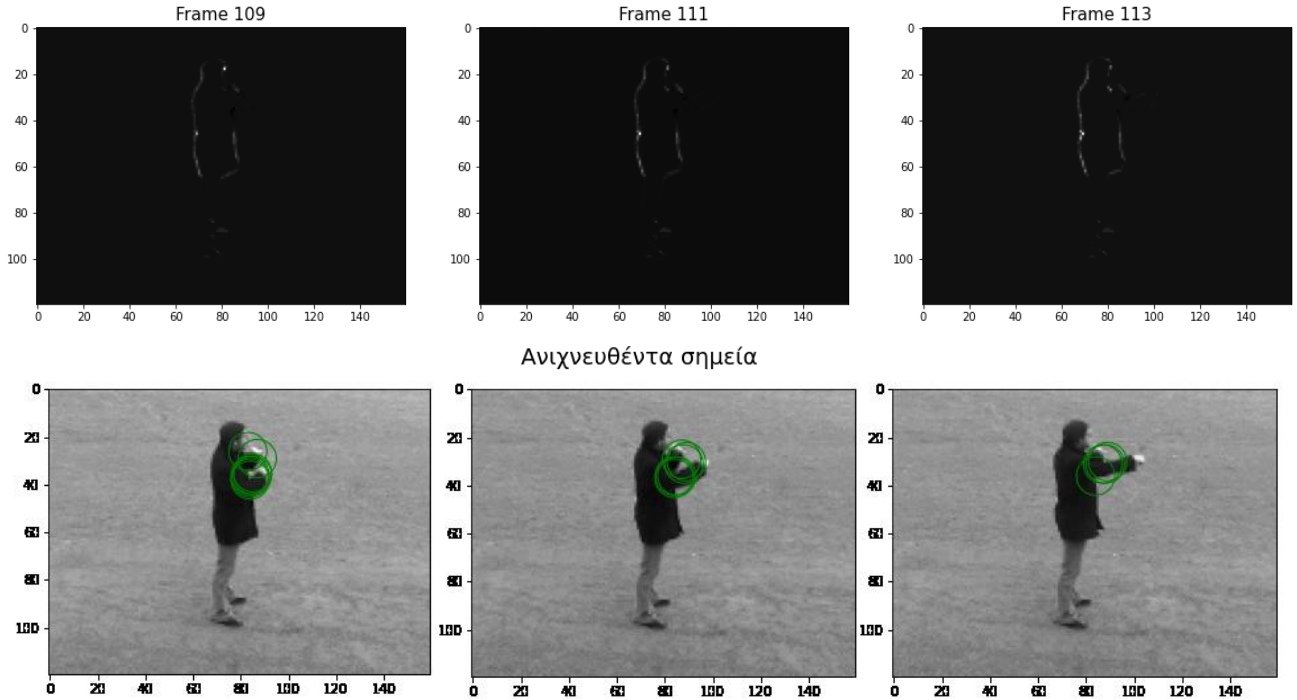
Κριτήριο Σημαντικότητας H - run video 1 - Harris Detector



Ανιχνευθέντα σημεία



Κριτήριο Σημαντικότητας H - box video 1 - Harris Detector



2.1.2) Ο ανιχνευτής Gabor βασίζεται στο χρονικό φιλτράρισμα του βίντεο με ένα ζεύγος Gabor φίλτρων αφού πρώτα έχει υποστεί εξομάλυνση στις χωρικές διαστάσεις μέσω ενός 2D γκαουσιανού πυρήνα $g(x, y; \sigma)$. Τα Gabor φίλτρα ορίζονται ως εξής:

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

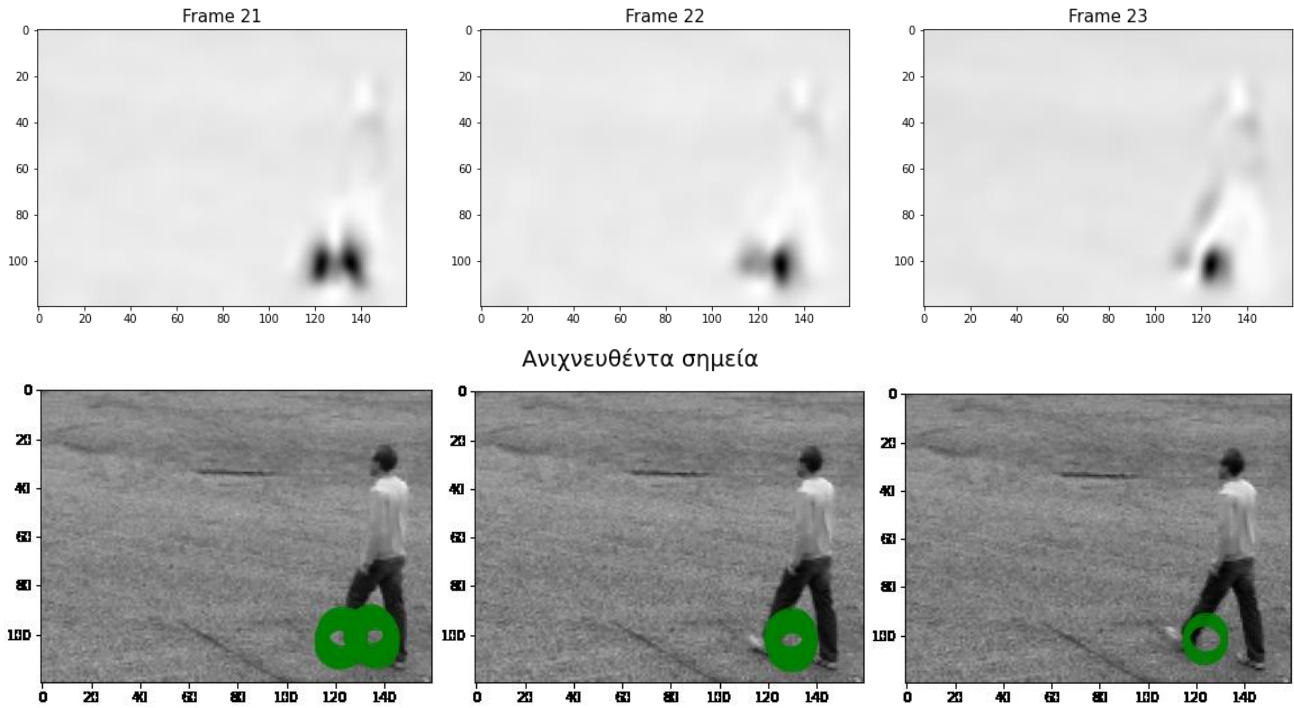
$$h_{od}(t; \tau, \omega) = \sin(2\pi t\omega) \exp(-t^2/2\tau^2)$$

Η συχνότητα του Gabor φίλτρου συνδέεται με την χρονική κλίμακα μέσω της σχέσης $\omega = 4/\tau$. Αφού κανονικοποιήσουμε τα h_{ev} και h_{od} με την L_1 νόρμα μπορούμε να εξάγουμε το κριτήριο σημαντικότητας από τη σχέση:

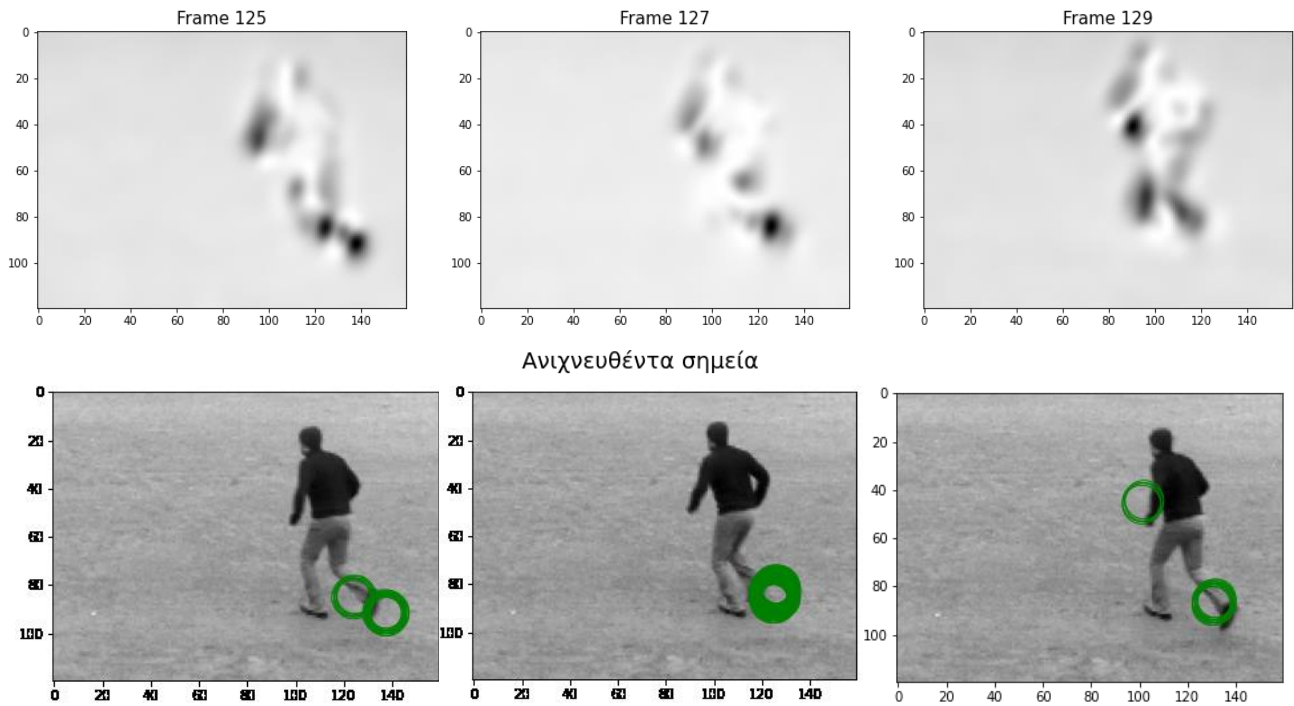
$$H(x, y, t) = (I(x, y, t) * g * h_{ev})^2 + (I(x, y, t) * g * h_{od})^2$$

Όπως και πριν υπολογίσαμε τα σημεία ενδιαφέροντος ως τα τοπικά μέγιστα του κριτηρίου H . Για την απεικόνιση χρησιμοποιήθηκε η δοσμένη συνάρτηση `show_detection`. Ακολουθούν κάποια αντιπροσωπευτικές εικόνες των αποτελεσμάτων του Harris Detector για κάθε μία από τις κλάσεις των βίντεο.

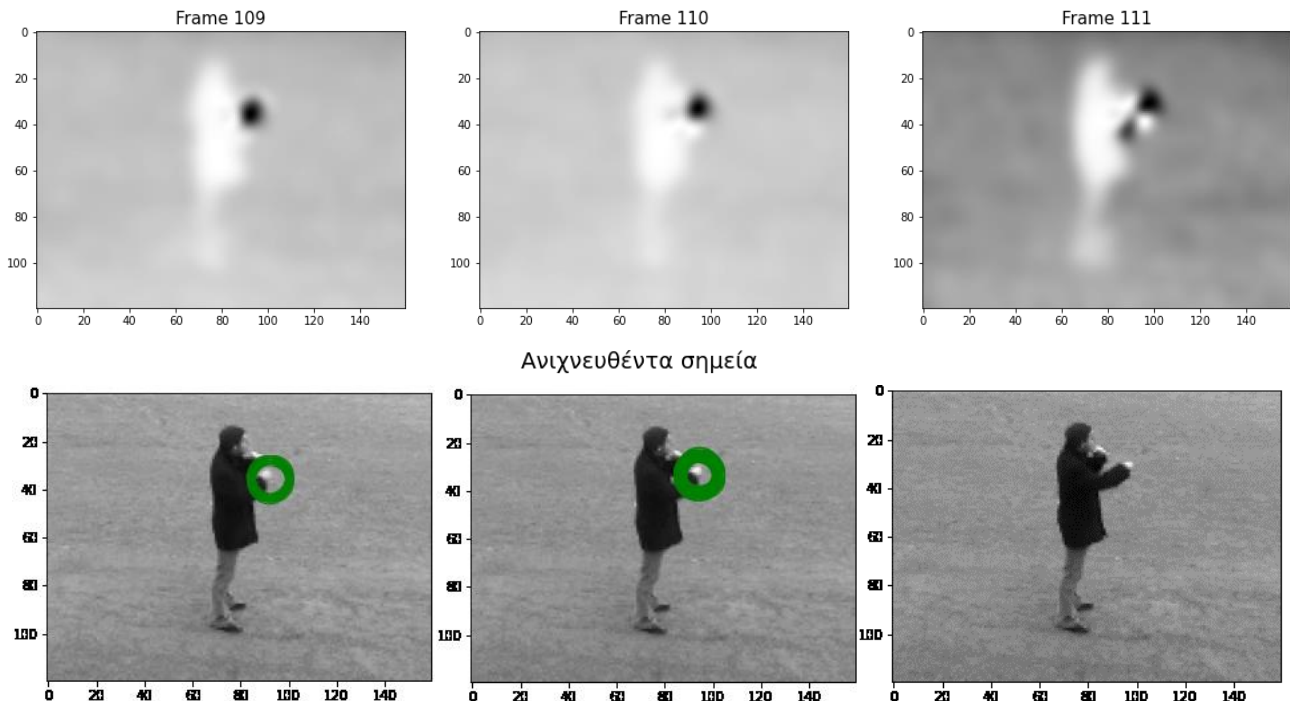
Κριτήριο Σημαντικότητας H - walk video 1 - Gabor Detector



Κριτήριο Σημαντικότητας H - run video 1 - Gabor Detector



Κριτήριο Σημαντικότητας H - box video 1 - Gabor Detector



Παρατηρώντας όλα τα εξαγόμενα βίντεο με τα εντοπισμένα σημεία ενδιαφέροντος παρατηρούμε πως ο ανιχνευτής Harris εντοπίζει σχεδόν κάθε φορά την κίνηση στα βίντεο ενώ ο ανιχνευτής Gabor εντοπίζει ένα μικρό μέρος των κινήσεων. Ωστόσο ο Gabor ανιχνευτής είναι πιο εύστοχος από τον Harris. Αυτό συμβαίνει καθώς ο ανιχνευτής Harris εντοπίζει κυρίως γωνίες στα frame των βίντεο όπως φαίνεται και στα παραπάνω παραδειγματικά frames. Χαρακτηριστικό παράδειγμα είναι πως εντοπίζει τη γωνία εντός των ποδιών στο τρέξιμο και το περπάτημα και το λυγισμένο αγκώνα στο μποξ. Αντίθετα στον Gabor ανιχνευτή τα σημεία που εντοπίζονται είναι πιο στοχευμένα και αυτό φαίνεται και από τα παραπάνω παραδείγματα όπου οι πράσινοι κύκλοι είναι πιο έντονοι, δηλαδή πολλά εντοπίζονται πολλά σημεία στην γειτονιά που όντως υπάρχει κίνηση.

2.2 Χωρο-χρονικοί Ιστογραφικοί Περιγραφητές

Σε αυτό το βήμα κατασκευάζουμε τις συναρτήσεις που εξάγουν τους περιγραφητές HOG, HOF και έναν συνδυασμό HOG και HOF.

Αρχικά υπολογίζουμε το διάνυσμα κλίσης (gradient) και την TLV1 οπτική ροή σε κάθε pixel που θα χρειαστούν για τους περιγραφητές. Οι περιγραφητές υπολογίζονται με την δοθείσα συνάρτηση `orientation_histogram` η οποία δέχεται ως είσοδο τα G_x , G_y που ορίζουν ένα διανυσματικό πεδίο. Για τον HOG τα G_x , G_y υπολογίζονται από το διάνυσμα κλίσης αλλά για τον HOF απαιτείται ένα πυκνό πεδίο οπτικής ροής και για αυτό παίρνουμε τα G_x , G_y από την TLV1 οπτική ροή. Ο περιγραφητής συνδυασμού των HOG/HOF προκύπτει με συνένωση των δύο.

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

Σε αυτό το βήμα γίνεται η κατηγοριοποίηση των βίντεο σε κατηγορίες με χρήση BoVW αναπαραστάσεων βασισμένων στα HOG/HOF χαρακτηριστικά η εξαγωγή των οποίων περιγράφεται παραπάνω.

2.3.1) Για τον διαχωρισμό των αρχείων σε δύο σύνολα (εκπαίδευσης και δοκιμής) με αντίστοιχα labels χρησιμοποιήσαμε κώδικα που παίρνει ως είσοδο ένα txt αρχείο ώστε εύκολα κανείς να μπορεί να αλλάξει τον διαμοιρασμό.

2.3.2) Για την τελική αναπαράσταση χρησιμοποιήθηκε η συνάρτηση που μας δόθηκε `bag_of_words`

2.3.3) Αφού έχουμε εξάγει αναπαραστήσεις για όλα τα βίντεο εκπαιδεύουμε το μοντέλο χρησιμοποιώντας την συνάρτηση `svm_train_test`.

2.3.4) Στη συνέχεια ακολουθούν για κάθε συνδυασμό ανιχνευτών/περιγραφητών η ακρίβεια και αναλυτικοί πίνακες που δείχνουν ως τι αναγνωρίστηκε κάθε δράσης.

detector: Harris descriptor: HOG	detector: Harris descriptor: HOF																																																																														
The accuracy for the combination is 58.333%	The accuracy for the combination is 33.333%																																																																														
Here is a table with detailed results:	Here is a table with detailed results:																																																																														
<table><tr><th>Video Name</th><th>Correct</th><th>Prediction</th></tr><tr><td>person07_walking_d4_uncomp.avi</td><td>walk</td><td>run</td></tr><tr><td>person08_walking_d2_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person11_walking_d1_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person20_walking_d3_uncomp.avi</td><td>walk</td><td>run</td></tr><tr><td>person06_boxing_d1_uncomp.avi</td><td>box</td><td>run</td></tr><tr><td>person07_boxing_d3_uncomp.avi</td><td>box</td><td>run</td></tr><tr><td>person11_boxing_d2_uncomp.avi</td><td>box</td><td>box</td></tr><tr><td>person15_boxing_d4_uncomp.avi</td><td>box</td><td>run</td></tr><tr><td>person10_running_d2_uncomp.avi</td><td>run</td><td>run</td></tr><tr><td>person20_running_d1_uncomp.avi</td><td>run</td><td>run</td></tr><tr><td>person24_running_d3_uncomp.avi</td><td>run</td><td>run</td></tr><tr><td>person25_running_d4_uncomp.avi</td><td>run</td><td>run</td></tr></table>	Video Name	Correct	Prediction	person07_walking_d4_uncomp.avi	walk	run	person08_walking_d2_uncomp.avi	walk	walk	person11_walking_d1_uncomp.avi	walk	walk	person20_walking_d3_uncomp.avi	walk	run	person06_boxing_d1_uncomp.avi	box	run	person07_boxing_d3_uncomp.avi	box	run	person11_boxing_d2_uncomp.avi	box	box	person15_boxing_d4_uncomp.avi	box	run	person10_running_d2_uncomp.avi	run	run	person20_running_d1_uncomp.avi	run	run	person24_running_d3_uncomp.avi	run	run	person25_running_d4_uncomp.avi	run	run	<table><tr><th>Video Name</th><th>Correct</th><th>Prediction</th></tr><tr><td>person07_walking_d4_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person08_walking_d2_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person11_walking_d1_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person20_walking_d3_uncomp.avi</td><td>walk</td><td>walk</td></tr><tr><td>person06_boxing_d1_uncomp.avi</td><td>box</td><td>walk</td></tr><tr><td>person07_boxing_d3_uncomp.avi</td><td>box</td><td>walk</td></tr><tr><td>person11_boxing_d2_uncomp.avi</td><td>box</td><td>walk</td></tr><tr><td>person15_boxing_d4_uncomp.avi</td><td>box</td><td>walk</td></tr><tr><td>person10_running_d2_uncomp.avi</td><td>run</td><td>walk</td></tr><tr><td>person20_running_d1_uncomp.avi</td><td>run</td><td>walk</td></tr><tr><td>person24_running_d3_uncomp.avi</td><td>run</td><td>walk</td></tr><tr><td>person25_running_d4_uncomp.avi</td><td>run</td><td>walk</td></tr></table>	Video Name	Correct	Prediction	person07_walking_d4_uncomp.avi	walk	walk	person08_walking_d2_uncomp.avi	walk	walk	person11_walking_d1_uncomp.avi	walk	walk	person20_walking_d3_uncomp.avi	walk	walk	person06_boxing_d1_uncomp.avi	box	walk	person07_boxing_d3_uncomp.avi	box	walk	person11_boxing_d2_uncomp.avi	box	walk	person15_boxing_d4_uncomp.avi	box	walk	person10_running_d2_uncomp.avi	run	walk	person20_running_d1_uncomp.avi	run	walk	person24_running_d3_uncomp.avi	run	walk	person25_running_d4_uncomp.avi	run	walk
Video Name	Correct	Prediction																																																																													
person07_walking_d4_uncomp.avi	walk	run																																																																													
person08_walking_d2_uncomp.avi	walk	walk																																																																													
person11_walking_d1_uncomp.avi	walk	walk																																																																													
person20_walking_d3_uncomp.avi	walk	run																																																																													
person06_boxing_d1_uncomp.avi	box	run																																																																													
person07_boxing_d3_uncomp.avi	box	run																																																																													
person11_boxing_d2_uncomp.avi	box	box																																																																													
person15_boxing_d4_uncomp.avi	box	run																																																																													
person10_running_d2_uncomp.avi	run	run																																																																													
person20_running_d1_uncomp.avi	run	run																																																																													
person24_running_d3_uncomp.avi	run	run																																																																													
person25_running_d4_uncomp.avi	run	run																																																																													
Video Name	Correct	Prediction																																																																													
person07_walking_d4_uncomp.avi	walk	walk																																																																													
person08_walking_d2_uncomp.avi	walk	walk																																																																													
person11_walking_d1_uncomp.avi	walk	walk																																																																													
person20_walking_d3_uncomp.avi	walk	walk																																																																													
person06_boxing_d1_uncomp.avi	box	walk																																																																													
person07_boxing_d3_uncomp.avi	box	walk																																																																													
person11_boxing_d2_uncomp.avi	box	walk																																																																													
person15_boxing_d4_uncomp.avi	box	walk																																																																													
person10_running_d2_uncomp.avi	run	walk																																																																													
person20_running_d1_uncomp.avi	run	walk																																																																													
person24_running_d3_uncomp.avi	run	walk																																																																													
person25_running_d4_uncomp.avi	run	walk																																																																													

detector:	Harris
descriptor:	HOF and HOG
The accuracy for the combination is 83.333%	
Here is a table with detailed results:	
Video Name	Correct Prediction
person07_walking_d4_uncomp.avi	walk walk True
person08_walking_d2_uncomp.avi	walk walk True
person11_walking_d1_uncomp.avi	walk walk True
person20_walking_d3_uncomp.avi	walk run False
person06_boxing_d1_uncomp.avi	box box True
person07_boxing_d3_uncomp.avi	box walk False
person11_boxing_d2_uncomp.avi	box box True
person15_boxing_d4_uncomp.avi	box box True
person10_running_d2_uncomp.avi	run run True
person20_running_d1_uncomp.avi	run run True
person24_running_d3_uncomp.avi	run run True
person25_running_d4_uncomp.avi	run run True

detector:	Gabor	detector:	Gabor
descriptor:	HOG	descriptor:	HOF
The accuracy for the combination is 50.000%		The accuracy for the combination is 33.333%	
Here is a table with detailed results:		Here is a table with detailed results:	
Video Name	Correct Prediction	Video Name	Correct Prediction
person07_walking_d4_uncomp.avi	walk run False	person07_walking_d4_uncomp.avi	walk walk True
person08_walking_d2_uncomp.avi	walk run False	person08_walking_d2_uncomp.avi	walk walk True
person11_walking_d1_uncomp.avi	walk run False	person11_walking_d1_uncomp.avi	walk walk True
person20_walking_d3_uncomp.avi	walk run False	person20_walking_d3_uncomp.avi	walk walk True
person06_boxing_d1_uncomp.avi	box box True	person06_boxing_d1_uncomp.avi	box walk False
person07_boxing_d3_uncomp.avi	box box True	person07_boxing_d3_uncomp.avi	box walk False
person11_boxing_d2_uncomp.avi	box run False	person11_boxing_d2_uncomp.avi	box walk False
person15_boxing_d4_uncomp.avi	box run False	person15_boxing_d4_uncomp.avi	box walk False
person10_running_d2_uncomp.avi	run run True	person10_running_d2_uncomp.avi	run walk False
person20_running_d1_uncomp.avi	run run True	person20_running_d1_uncomp.avi	run walk False
person24_running_d3_uncomp.avi	run run True	person24_running_d3_uncomp.avi	run walk False
person25_running_d4_uncomp.avi	run run True	person25_running_d4_uncomp.avi	run walk False

```

detector:      Gabor
descriptor:   HOF

The accuracy for the combination is 33.333%

Here is a table with detailed results:

Video Name                                Correct Prediction
person07_walking_d4_uncomp.avi           walk    walk    True
person08_walking_d2_uncomp.avi           walk    walk    True
person11_walking_d1_uncomp.avi           walk    walk    True
person20_walking_d3_uncomp.avi           walk    walk    True
person06_boxing_d1_uncomp.avi            box     walk    False
person07_boxing_d3_uncomp.avi            box     walk    False
person11_boxing_d2_uncomp.avi            box     walk    False
person15_boxing_d4_uncomp.avi            box     walk    False
person10_running_d2_uncomp.avi           run     walk    False
person20_running_d1_uncomp.avi           run     walk    False
person24_running_d3_uncomp.avi           run     walk    False
person25_running_d4_uncomp.avi           run     walk    False

```

detector: Gabor
descriptor: HOF and HOG

The accuracy for the combination is 91.667%

Here is a table with detailed results:

Video Name	Correct	Prediction	
person07_walking_d4_uncomp.avi	walk	walk	True
person08_walking_d2_uncomp.avi	walk	walk	True
person11_walking_d1_uncomp.avi	walk	run	False
person20_walking_d3_uncomp.avi	walk	walk	True
person06_boxing_d1_uncomp.avi	box	box	True
person07_boxing_d3_uncomp.avi	box	box	True
person11_boxing_d2_uncomp.avi	box	box	True
person15_boxing_d4_uncomp.avi	box	box	True
person10_running_d2_uncomp.avi	run	run	True
person20_running_d1_uncomp.avi	run	run	True
person24_running_d3_uncomp.avi	run	run	True
person25_running_d4_uncomp.avi	run	run	True

Και συγκεντρωτικά:

<div> <div>ανιχνευτής</div> <div>περιγραφητής</div> </div>	Harris	Gabor
HOG	58.333%	50%
HOF	33.333%	33.333%
HOG / HOF	83.333%	91.667%

Παρατηρούμε λοιπόν πως γενικά οι περιγραφητές μόνοι τους δεν πετυχαίνουν ιδιαίτερα υψηλά ποσοστά αλλά ο συνδυασμός των δύο βγάζει αρκετά υψηλό accuracy. Συγκεκριμένα ο περιγραφής HOF και για τους δύο ανιχνευτές αναγνωρίζει όλα τα βίντεο ως walk. Αυτό το πρόβλημα ίσως λυθεί αν βάλουμε λιγότερα βίντεο με label walk στο training set όπως φαίνεται στη συνέχεια.

detector: Harris descriptor: HOF The accuracy for the combination is 50.000% Here is a table with detailed results:				detector: Gabor descriptor: HOF The accuracy for the combination is 43.750% Here is a table with detailed results:			
Video Name	Correct	Prediction		Video Name	Correct	Prediction	
person07_walking_d4_uncomp.avi	walk	walk	True	person07_walking_d4_uncomp.avi	walk	run	False
person08_walking_d2_uncomp.avi	walk	walk	True	person08_walking_d2_uncomp.avi	walk	run	False
person11_walking_d1_uncomp.avi	walk	walk	True	person11_walking_d1_uncomp.avi	walk	run	False
person13_walking_d3_uncomp.avi	walk	walk	True	person13_walking_d3_uncomp.avi	walk	box	False
person14_walking_d4_uncomp.avi	walk	walk	True	person14_walking_d4_uncomp.avi	walk	box	False
person18_walking_d4_uncomp.avi	walk	walk	True	person18_walking_d4_uncomp.avi	walk	run	False
person20_walking_d3_uncomp.avi	walk	walk	True	person20_walking_d3_uncomp.avi	walk	run	False
person25_walking_d4_uncomp.avi	walk	walk	True	person25_walking_d4_uncomp.avi	walk	box	False
person06_boxing_d1_uncomp.avi	box	walk	False	person06_boxing_d1_uncomp.avi	box	box	True
person07_boxing_d3_uncomp.avi	box	walk	False	person07_boxing_d3_uncomp.avi	box	box	True
person11_boxing_d2_uncomp.avi	box	walk	False	person11_boxing_d2_uncomp.avi	box	box	True
person15_boxing_d4_uncomp.avi	box	walk	False	person15_boxing_d4_uncomp.avi	box	box	True
person10_running_d2_uncomp.avi	run	walk	False	person10_running_d2_uncomp.avi	run	box	False
person20_running_d1_uncomp.avi	run	walk	False	person20_running_d1_uncomp.avi	run	run	True
person24_running_d3_uncomp.avi	run	walk	False	person24_running_d3_uncomp.avi	run	run	True
person25_running_d4_uncomp.avi	run	walk	False	person25_running_d4_uncomp.avi	run	run	True

Βλέπουμε πως αφήνοντας 8 (από τα 11 που ήταν αρχικά) βίντεο για walk στο training test το accuracy αυξάνεται για τον περιγραφητή HOF. Για τον συνδυασμό Gabor/HOF πλέον αναγνωρίζει και άλλα είδη δράσεων αλλά για τον συνδυασμό Harris/HOF και πάλι τα αναγνωρίζει όλα ως walk.

Μέρος 3ο: Συνένωση Εικόνων(Image Stitching) για Δημιουργία Πανοράματος

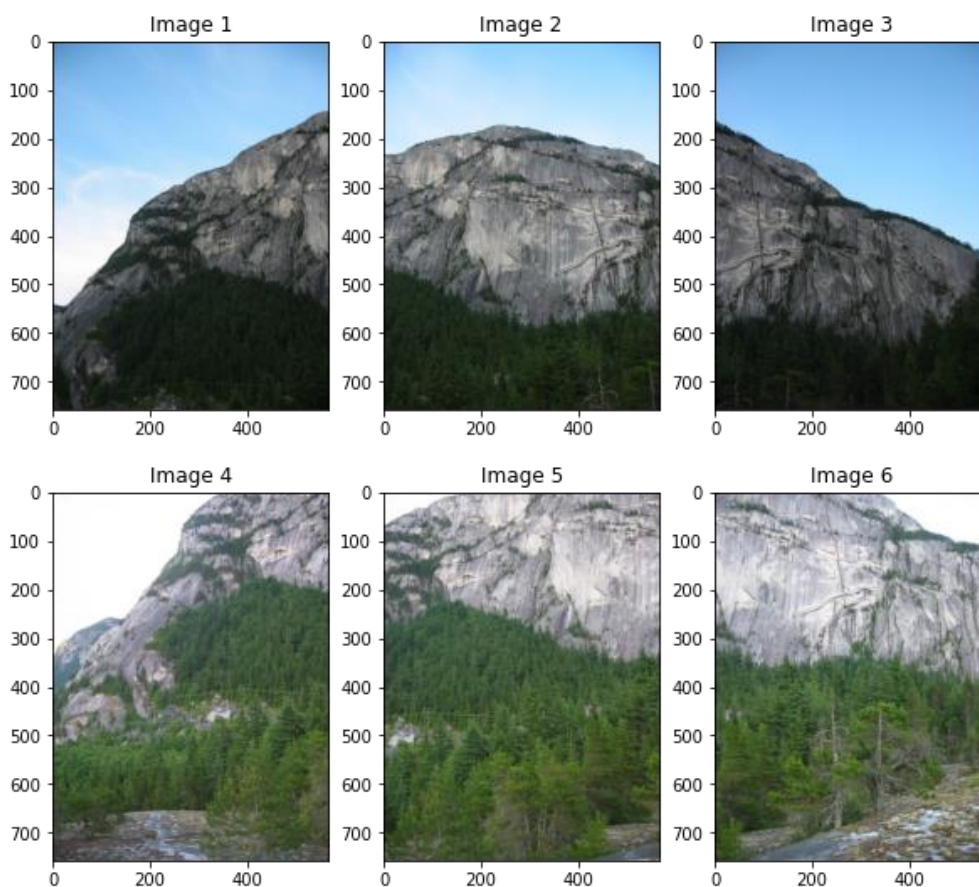
Σκοπός του μέρους αυτού είναι η δημιουργία μιας εφαρμογής για την συνένωση 6 διαδοχικών εικόνων που έχουν ληφθεί από μια κάμερα σε σταθερό σημείο, η οποία έχει περιστραφεί 6 φορές. Η δημιουργία της εφαρμογής βασίζεται στον υπολογισμό ενός πίνακα ομογραφίας Η μεταξύ διαδοχικών εικόνων, για την δημιουργία ενός κοινού πλαισίου αναφοράς. Τελικό αποτέλεσμα της διαδικασίας είναι μια εικόνα πανοράματος, με στόχο την τεχνητή αύξηση του πεδίου ορατότητας (field of view).

Περιγραφή του αλγορίθμου συνένωσης εικόνων

Η βασική ρουτίνα που χρησιμοποιήθηκε για τον σχηματισμό του ζητούμενου πανοράματος είναι η `image_stitching(I1,I2)` η οποία θα συνενώνει την εικόνα I1 στο πλαίσιο αναφορά I2. Παρακάτω ακολουθεί η ανάλυση της ρουτίνας για τις 2 πρώτες εικόνες του φακέλου Part 3, ενώ στην συνέχεια θα αποδοθούν περιληπτικά τα αποτελέσματά της και για τις υπόλοιπες.

Βήμα 0: Διάβασμα Εικόνων

Οι 6 εικόνες που δίνονται στον φάκελο Part3 του συμπληρωματικού υλικού της άσκησης αποτελούν διαδοχικά τμήματα ενός βουνού και απεικονίζονται παρακάτω (στον χώρο RGB)

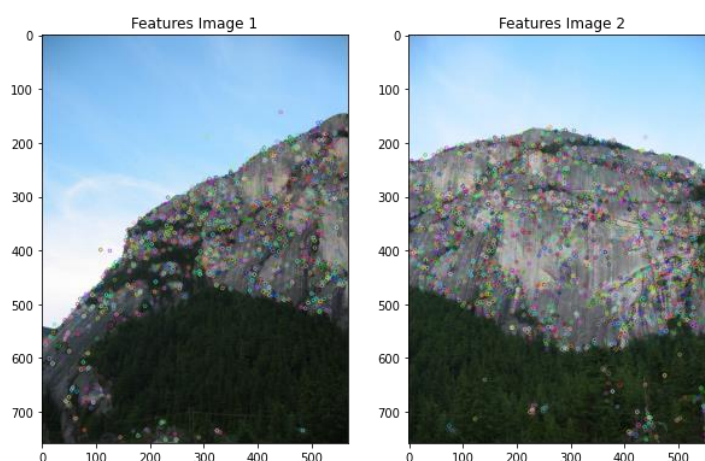


Βήμα 1: Εντοπισμός χαρακτηριστικών ενδιαφέροντος και εξαγωγή περιγραφητών

Χρησιμοποιούμε τις δοθείσες ρουτίνες σε python για την υλοποίηση του βήματος 1:

- SIFT_create: Για την εξαγωγή των χαρακτηριστικών ενδιαφέροντος και των περιγραφητών
- detectAndCompute: Για τον υπολογισμό των σημείων ενδιαφέροντος
- drawKeypoints: Για την απεικόνιση των σημείων ενδιαφέροντος πάνω στις αρχικές εικόνες

Έτσι, λαμβάνουμε τις εικόνες I1,I2 με σημειωμένα τα εν λόγω σημεία

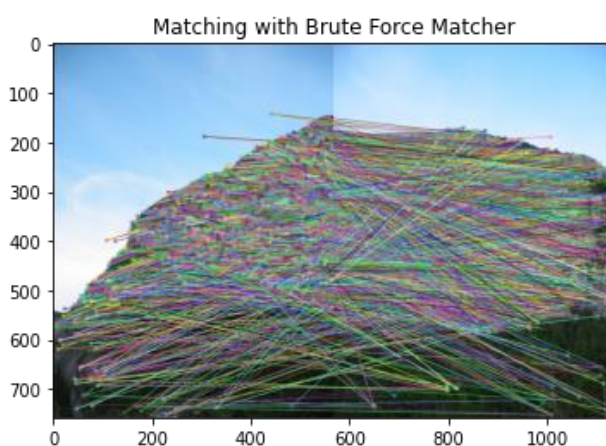


Βήμα 2: Διαδικασία ταιριάσματος χαρακτηριστικών μεταξύ δύο εικόνων

Στο 2ο βήμα, καλούμαστε να αντιστοιχίσουμε κάθε χαρακτηριστικό της εικόνας A με τα 2 κοντινότερα χαρακτηριστικά της εικόνας B, μέσω υπολογισμού της απόστασης των αντίστοιχων περιγραφητών.

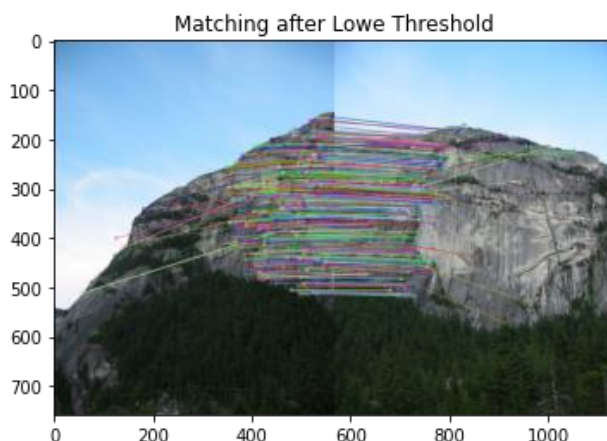
Μεταξύ των 2 δοθέντων επιλογών, επιλέχθηκε η εύρεση των πραγματικών κοντινότερων γειτόνων, μέσω των ρουτίνων BFMatcher και knnMatch, και όχι των προσεγγιστικών, διότι παρατηρήσαμε ότι δεν επιβάρυνε σημαντικά την συνολική ρουτίνα image_stitching(I1,I2)

Έτσι, λάβαμε τα εξής αποτελέσματα:



Βήμα 3: Διατήρηση μόνο των πιο έγκυρων αντιστοιχίσεων μέσω εφαρμογής του κριτηρίου Lowe

Από τα αποτελέσματα του βήματος 2, παρατηρούμε ότι έχουμε ένα μεγάλο πλήθος αντιστοιχίσεων χαρακτηριστικών ανάμεσα στις 2 εικόνες, αρκετές εκ των οποίων είναι λάθος. Σε αυτό το σημείο, εφαρμόζουμε ένα κατώφλι απόστασης ίσο με 0.8 και κρατάμε τις αντιστοιχίσεις των οποίων ο λόγος είναι μικρότερος από αυτό. Έτσι, λαμβάνουμε το εξής αποτέλεσμα



Ενώ παρατηρείται και αισθητή μείωση του πλήθους αντιστοιχίσεων σε σχέση με το βήμα 2, ενώ παράλληλα γίνονται πιο ευδιάκριτες οι περιοχές των 2 εικόνων μεταξύ των οποίων θα πραγματοποιηθεί το stitching (εν προκειμένω η πάνω δεξιά περιοχή της I1 με την πάνω αριστερά περιοχή της I2, κάτι που επαληθεύεται και οπτικά)

Features before Lowe Threshold: 1363
Features after Lowe Threshold: 447

Βήμα 4: Υπολογισμός ομογραφίας με RANSAC

Χρησιμοποιούμε τις 447 αντιστοιχίσεις μεταξύ των 2 εικόνων για τον υπολογισμό του πίνακα ομογραφίας. Συγκεκριμένα, έχουμε ορίσει την 1η εικόνα ως `query_img` και την 2η ως `train_img`, γεγονός που μας οδηγεί στον υπολογισμό της ομογραφίας HBA για την μετάβαση της εικόνας A στο πλαίσιο αναφοράς της εικόνας B.

Για τις εικόνες I1 και I2 του part3 λαμβάνουμε τον κάτωθι πίνακα ομογραφίας:

```
The Homography matrix is: [[ 1.23547114e+00 -2.16987563e-01 -3.50359938e+02]
[ 3.78386367e-01 1.16247234e+00 -1.64445985e+02]
[ 4.51256004e-04 -3.98707431e-05 1.00000000e+00]]
```

Βήματα 5 και 6: Υπολογισμός μετασχηματισμένης εικόνας και συνένωση των εικόνων

Επιλέχθηκε η από κοινού υλοποίηση των βημάτων 5 και 6 σε μία ενιαία υπορουτίνα `warpImages(img1,img2,H)` όπου η `img1` είναι η σταθερή εικόνα και η `img2` είναι η εικόνα που καλείται να μετασχηματιστεί, ενώ ο πίνακας `H` είναι ο πίνακας ομογραφίας του βήματος 4. Στην περίπτωση μας, καλούμε την `warpImages(I2,I1,HBA)`.

Αρχικά, υπολογίζουμε τα 4 ακριανά σημεία της εικόνας `I1` τα οποία και τα μετασχηματίζουμε στο πλαίσιο αναφοράς της εικόνας `I2` μέσω της εντολής `perspectiveTransformMan()`. Συγκεκριμένα, η εν λόγω εντολή λαμβάνει ως εισόδους τα ακριανά σημεία της `I1` καθώς και τον πίνακα ομογραφίας του βήματος 4- και αφού προστεθεί ανά συντεταγμένη μία 3η γραμμή μηδενικών για την μετατροπή τους σε ομογενείς- υλοποιεί το γινόμενο $\text{coords_transformed} = H * \text{coords}$ (για κάθε σημείο εισόδου ξεχωριστά). Τέλος, πραγματοποιείται διαίρεση με την ομογενή συντεταγμένη για να λάβουμε το ζητούμενο ζεύγος μετασχηματισμένων συντεταγμένων στην μορφή `[x,y]`

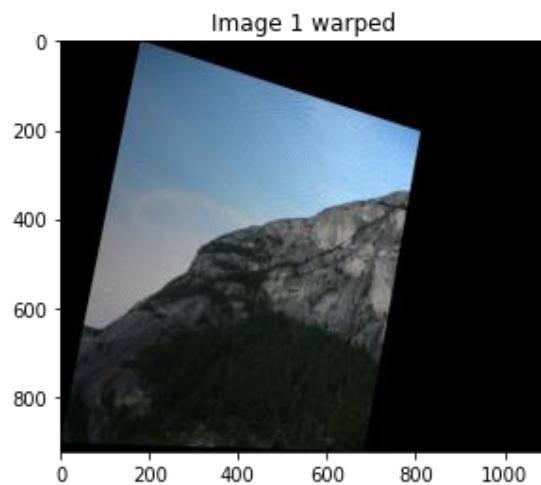
Εν συνεχεία, εισάγουμε σε κοινή λίστα τα παραπάνω σημεία με τα ακριανά σημεία της εικόνας `I2` και υπολογίζουμε την `minimum` τιμή ανά συντεταγμένη. Οι τιμές αυτές αντιστοιχούν στις άνω αριστερά συντεταγμένες της νέας εικόνας που καλούμαστε να σχηματίσουμε. Για τις εικόνες `I1 I2`, έχουμε:

Top Left Coords: [-531, -164]

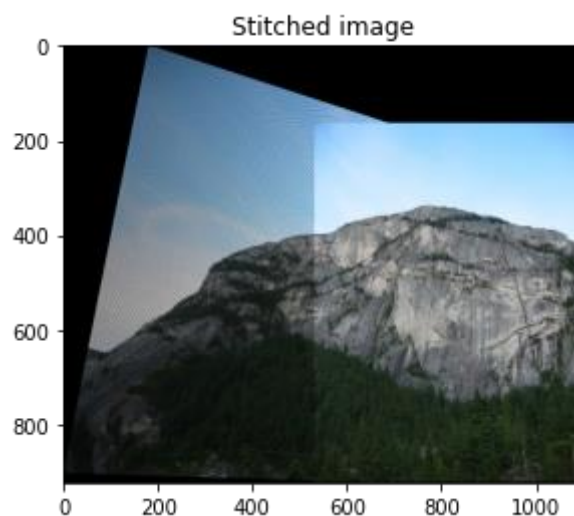
Όσον αφορά την εύρεση της `img1_warped`, αρχικά ορίζουμε την συνολική εικόνα εξόδου με διαστάσεις `[ymax-ymin, xmax-xmin]`, μιας και η μοντελοποίηση του προβλήματος πανοράματος αντιστοιχεί τις συντεταγμένες `y` στις γραμμές της εκάστοτε εικόνας, και τις συντεταγμένες `x` στις στήλες.

Στην συνέχεια, χρησιμοποιούμε την ρουτίνα `perspectiveTransformMan()` για κάθε συντεταγμένη της εικόνας `I1`, ώστε να βρούμε που αντιστοιχεί στο πλαίσιο αναφοράς της εικόνας `I2`. Οι συντεταγμένες στην τελική εικόνα προκύπτουν αν προσθέσουμε σε αυτές που βρήκαμε από την τελευταία κλήση της ρουτίνας `perspectiveTransformMan()` την απόλυτη τιμή των Top Left Coords που υπολογίσαμε παραπάνω.

Λαμβάνουμε τελικά:



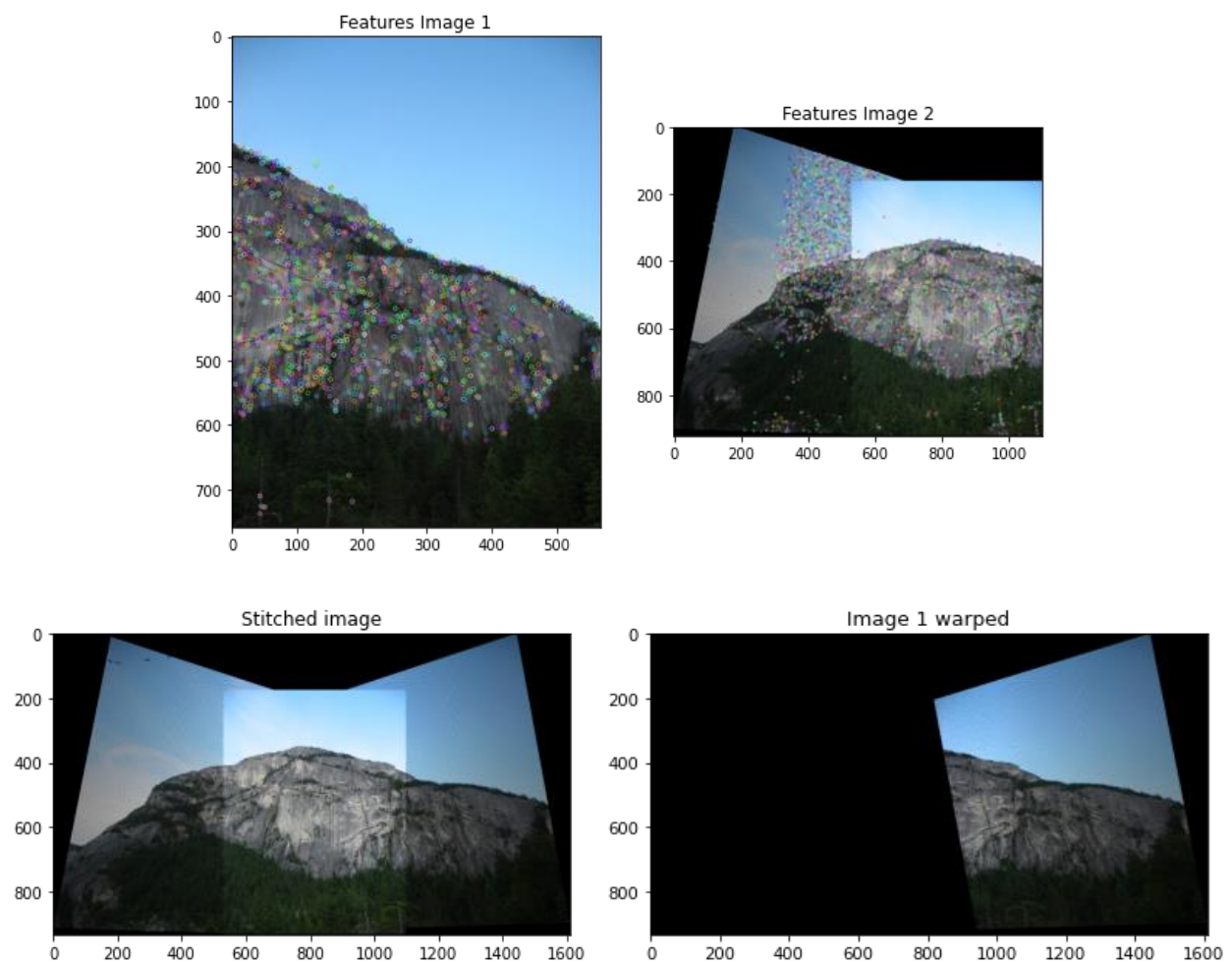
Για να πάρουμε την `stitched_image` προσθέτουμε τα pixels της `I2` στην `I1_warped`, πάλι προσθέτοντας στις συντεταγμένες της `I2` την απόλυτη τιμή των Top Left Coords, ενώ για να αποφύγουμε την ύπαρξη τυχόν μαύρων pixels τα οποία θα επικαλύψουν την ήδη σχηματισμένη `img1_warped`, κρατάμε μόνο τα μη μηδενικά pixels της `I2`. Η παραπάνω διαδικασία οδηγεί στο ζητούμενο stitching:



Επαναληπτική κλήση της ρουτίνας `image_stitching(I1,I2)` για την δημιουργία του πανοράματος

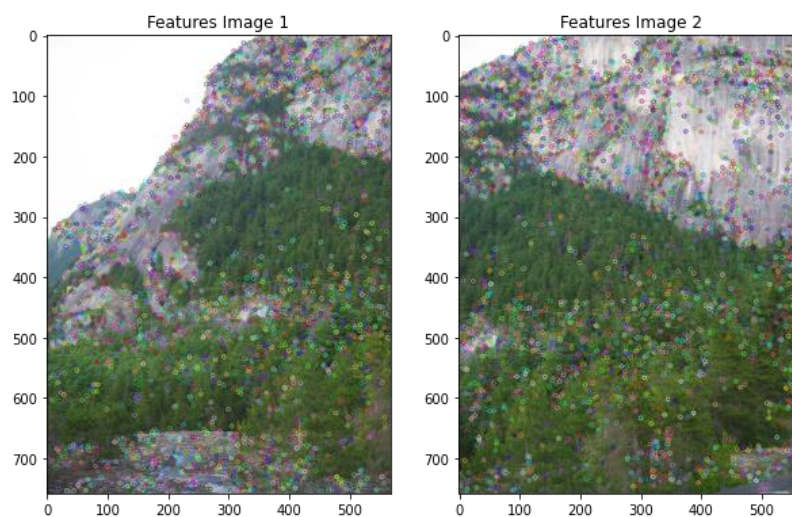
Ακολουθεί η περιληπτική παρουσίαση των αποτελεσμάτων της ρουτίνας μας και για τις υπόλοιπες εικόνες

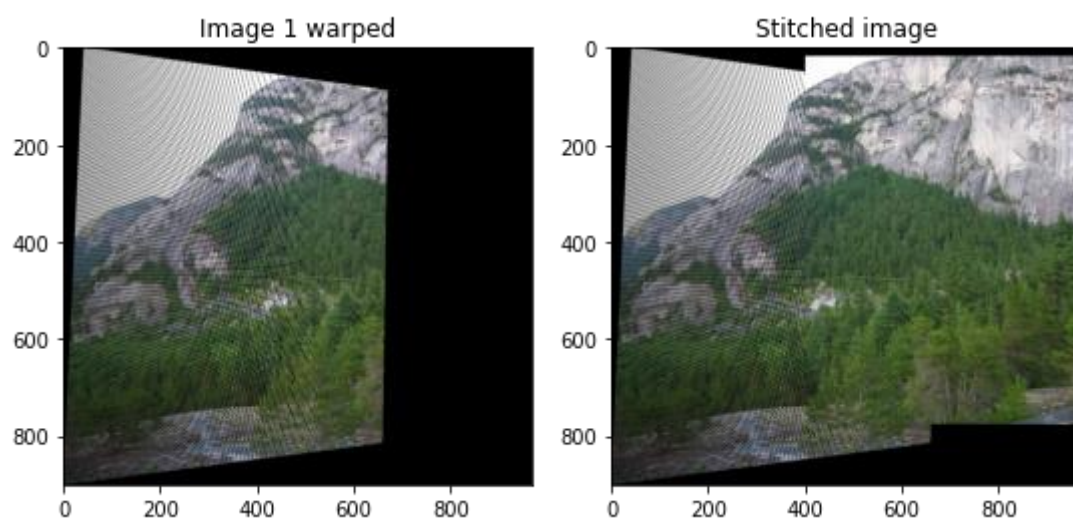
`image_stitching(I3,I12_stitched)`: Για την ένταξη της εικόνας `I3` στο πλαίσιο αναφορά της εικόνας `I12_stitched`



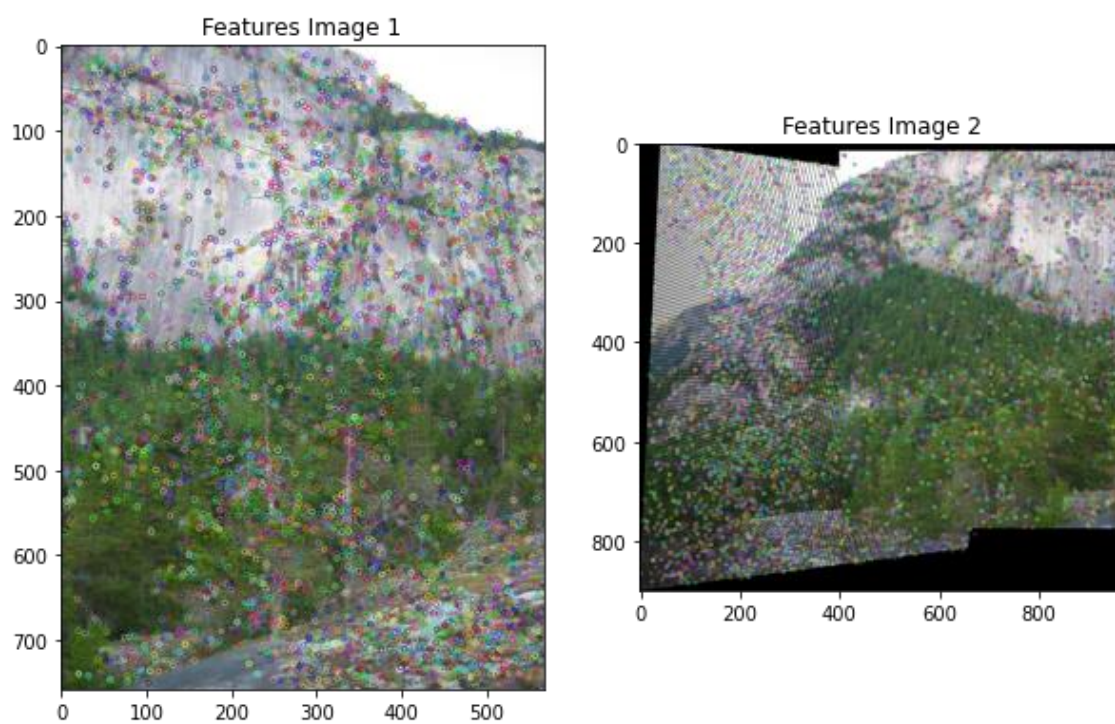
Στην συνέχεια, εργαζόμαστε με όμοιο τρόπο για τις εικόνες που αφορούν το κάτω μέρος του βουνού.

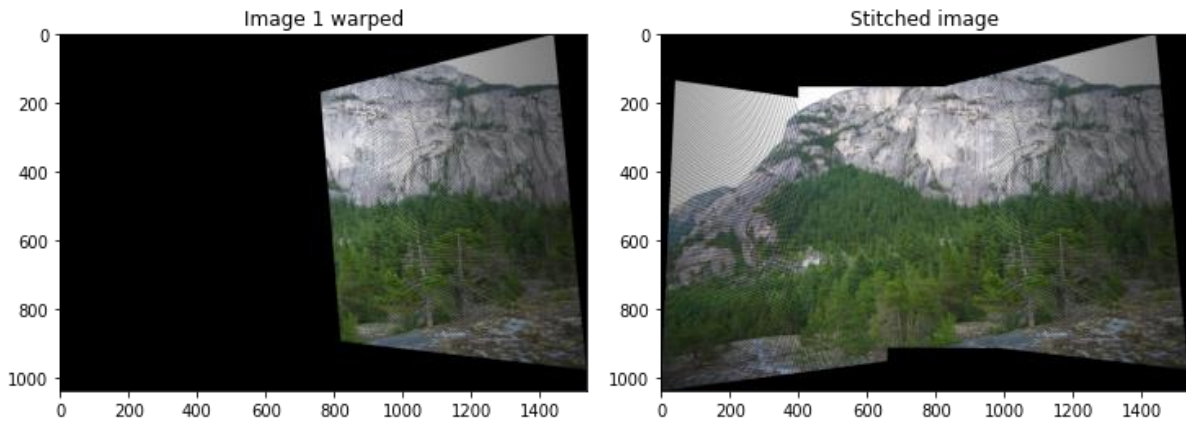
`image_stitching(I4,I5)`: Για την ένταξη της εικόνας I4 στο πλαίσιο αναφορά της εικόνας I5



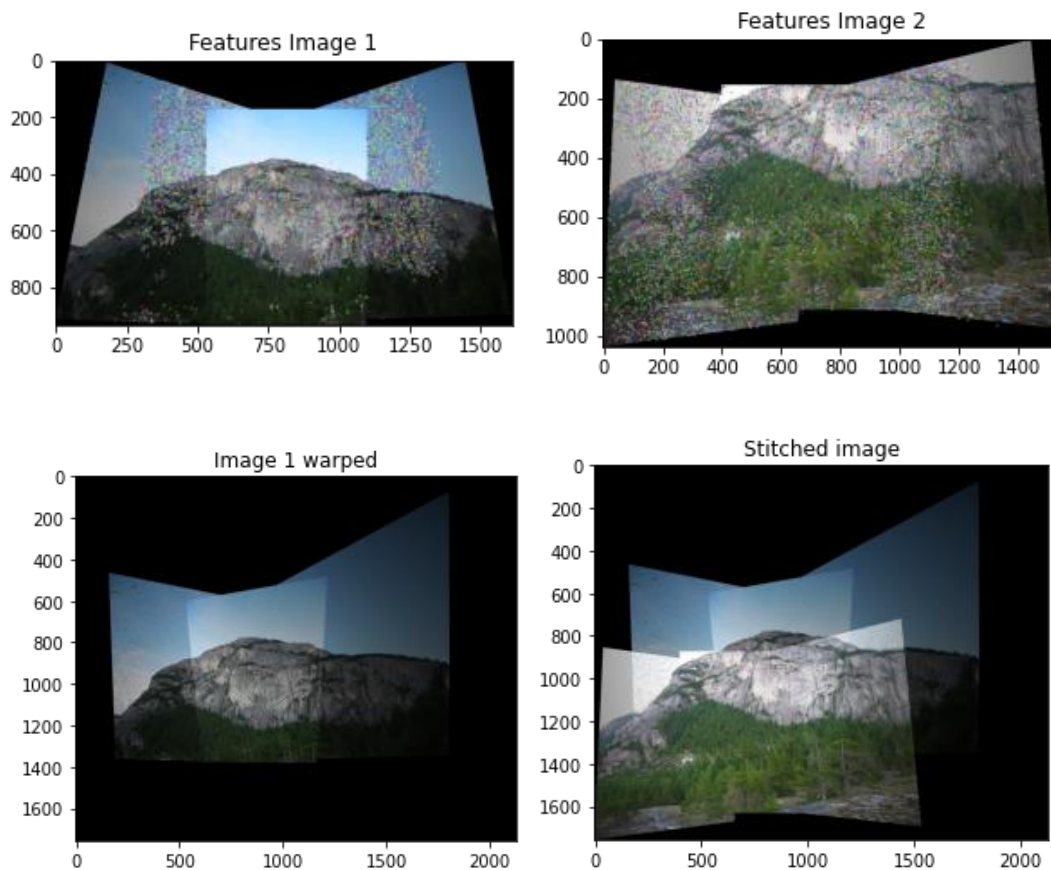


`image_stitching(I6,I45_stitched)`: Για την ένταξη της εικόνας I6 στο πλαίσιο αναφορά της άνωθι εικόνας I45_stitched.





`image_stitching(I123_stitched,I456_stitched)`: Για την ένταξη της εικόνας I123 στο πλαίσιο αναφοράς της άνωθι εικόνας I456 και τη εξαγωγή του ζητούμενου πανοράματος



Παρατηρούμε επειδή κατά την υλοποίηση των βημάτων 5 και 6 έγινε η άμεση μετάβαση από τις συντεταγμένες της εικόνας I1 στο πλαίσιο αναφοράς της εικόνας I2 - αντί του `inverse warping` - οδήγησε σε σχετική μείωση της ποιότητας του τελικού πανοράματος. Αυτό σχετίζεται με την ύπαρξη μηδενικών σε ορισμένα pixels της I1 που με την σειρά του οφείλεται στην μετατροπή των δεκαδικών συντεταγμένων σε ακέραιες μέσω της συνάρτησης `np.round()`. Το παραπάνω φαινόμενο εντείνεται προφανώς με την αύξηση των κλήσεων της ρουτίνας μας.