

**ΕΛΛΗΝΙΚΟ ΜΕΣΟΓΕΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**

ΔΠΜΣ ΠΡΟΗΓΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΓΩΓΗΣ ΑΥΤΟΜΑΤΙΣΜΟΥ ΚΑΙ ΡΟΜΠΟΤΙΚΗΣ



ΕΡΓΑΣΙΑ 2η ΜΗΧΑΝΙΚΗ ΟΡΑΣΗ



**ΣΠΟΥΔΑΣΤΗΣ:
ΔΡΑΚΟΣ ΙΩΑΝΝΗΣ (ΑΜ:ΜΗ117)**

ΗΡΑΚΛΕΙΟ 2020

Exercise 1 - Blob Detection [3/10]

Develop a simple 'blob' detector. The detector should rely on the LoG filter.

1. Implement a function to detect blobs of a given radius. [2]

2. Add an extra parameter to the detector to select either: i) dark blobs, ii) light blobs, or iii) both. [0.5]

3. Extend the detector so instead of a single radius it accepts a range (min, max). [0.5]

Test the detector firstly in the images 'black_dots.jpg', 'white_dots.jpg' and then also in 'coins.tiff' and 'circles_{}.jpg'. Present the results by showing the detected blobs as circles in the initial images.

Περιγραφή:

Σε αυτήν την άσκηση μας ζητήθηκε να δημιουργήσουμε έναν αλγόριθμο ο οποίος να ανιχνεύει blobs συγκεκριμένης ακτίνας σε ζητούμενες εικόνες. Περιοχές δηλαδή οι οποίες περιέχουν pixels με συγκεκριμένες τιμές φωτεινότητας. Ο αλγόριθμός μας πρέπει να είναι σε θέση να ανιχνεύει σκούρα blobs, ανοιχτόχρωμα blobs ή και τα δύο μαζί. Επίσης να μπορεί να εκτελέσει τα παραπάνω και για μεταβλητή ακτίνα. Για να το καταφέρουμε αυτό αρχικά δημιουργήσαμε ένα μενού το οποίο ζητά από τον χρήστη να επιλέξει την φωτογραφία. Στην συνέχεια το αν θέλει να ανιχνεύσει ανοιχτές, σκούρες ή και τις δύο τιμές φωτεινότητας. Όπως επίσης δίνει και την επιλογή ανίχνευσης για μεταβλητές τιμές ακτίνας. Τέλος ζητάει την επιθυμητή ή το εύρος των επιθυμητών ακτίνων. Για την ανίχνευση των blobs αρχικά μετατρέπουμε την εικόνα μας σε grayscale. Εφαρμόζουμε ένα LoG φίλτρο. Χρησιμοποιώντας το συγκεκριμένο φίλτρο ουσιαστικά δημιουργούμε τις περιοχές που θέλουμε να δουλέψουμε αφού το φίλτρο μας επιστρέφει τα κέντρα των blobs. Για να βρούμε τα σημεία αυτά χρησιμοποιούμε την `local_maxima_3d` που μας επιστρέφει τα σημεία των κέντρων και τις ακτίνες των blobs. Στην συνέχεια δημιουργήσαμε τις `find blobs`. Μια για κάθε περίπτωση ανοιχτά, σκούρα και τα δύο. Αφού εφαρμόζουμε ένα `threshold` (όπου εδώ το έχουμε ορίσει 0,7), ορίζοντας την ένταση της φωτεινότητας, κρατάμε τις τιμές που θέλουμε.

Για σκούρα έχω:

```
img_blobs = img_log < 0.7 * max_val
```

Για ανοιχτόχρωμα έχω:

```
img_blobs = np.abs(img_log) > 0.7*np.max(np.abs(img_log))
```

Και για τα δύο:

```
img_blobs = ((img_log < 0.7*max_val) | (np.abs(img_log) > 0.7*max_val))
```

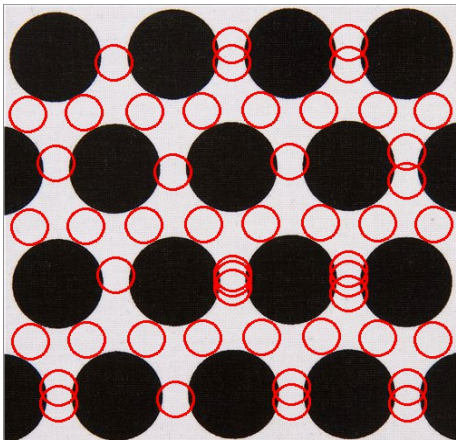
Εδώ αξίζει να σημειωθεί ότι για την αποτύπωση των ανοιχτόχρωμων blobs αντιμετωπίσα μια δυσκολία καθώς δεν μου τα εμφάνιζε σωστά. Αυτό λύθηκε όταν τελείως διαισθητικά για αυτή την περίπτωση εφάρμοσα το αρνητικό φίλτρο (-LoG) όπου φαίνεται να δουλεύει όπως θα δείτε και παρακάτω.

Όσον αφορά την περίπτωση των μεταβλητών ακτίνων δεν υλοποίησα την `non_maxima_3d`, άρα μου εμφανίζει μεν blobs μεταβλητών ακτίνων αλλά τα εμφανίζει όλα.

Αποτελέσματα εικόνων:
Για την εικόνα black dots.

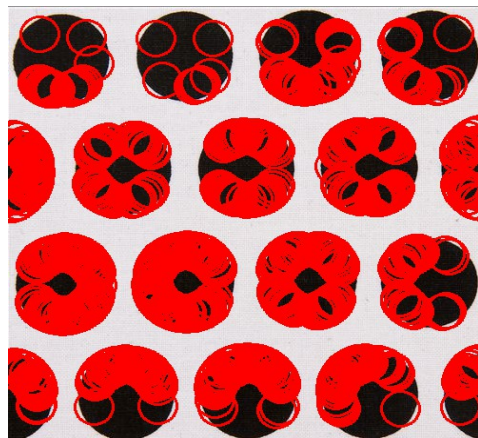
R=20

Detecting=Light Blobs



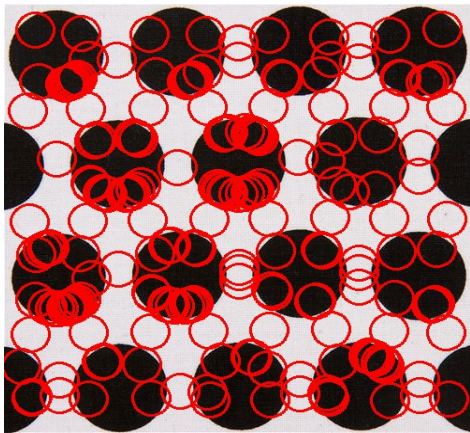
R=20

Detecting=Dark Blobs



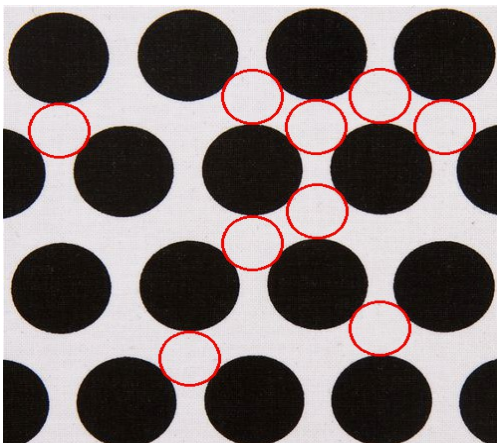
R=20

Detecting=both



R=30

Detecting=Light Blobs



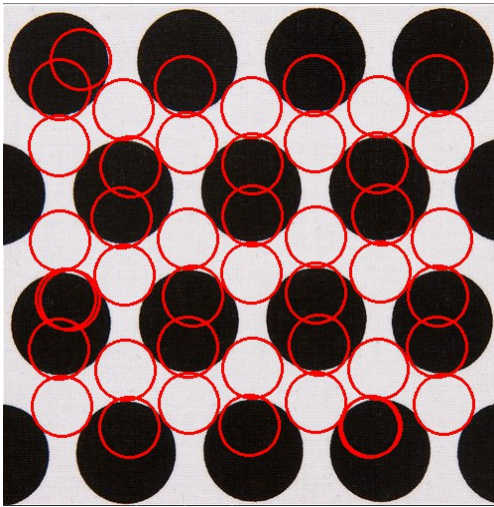
R=30

Detecting=Dark Blobs



R=30

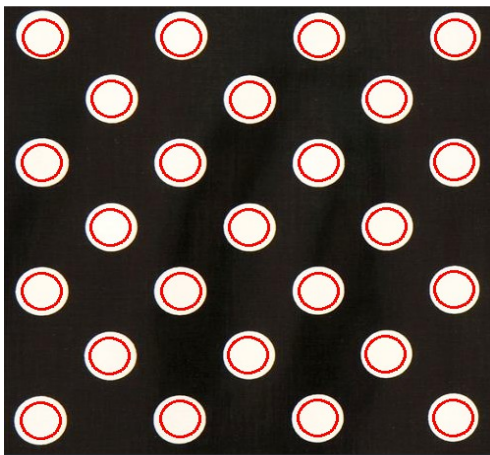
Detecting=both



Για την εικόνα white dots.

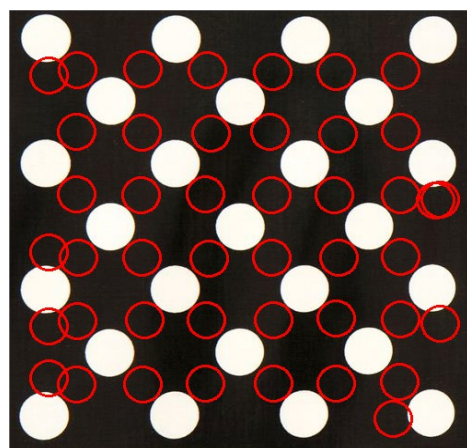
R=20

Detecting=Light Blobs



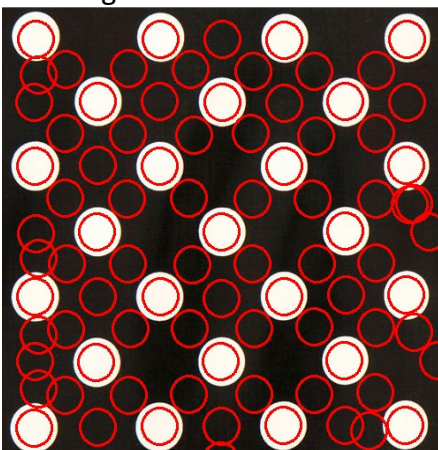
R=20

Detecting=Dark Blobs

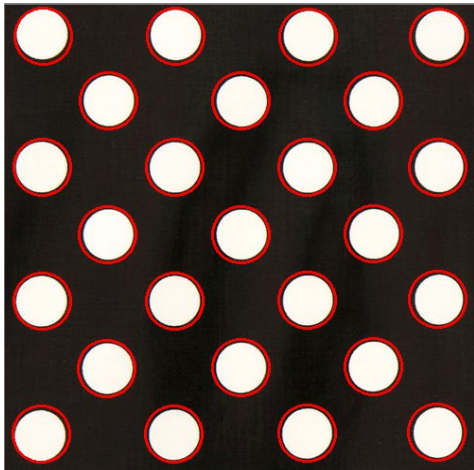


R=20

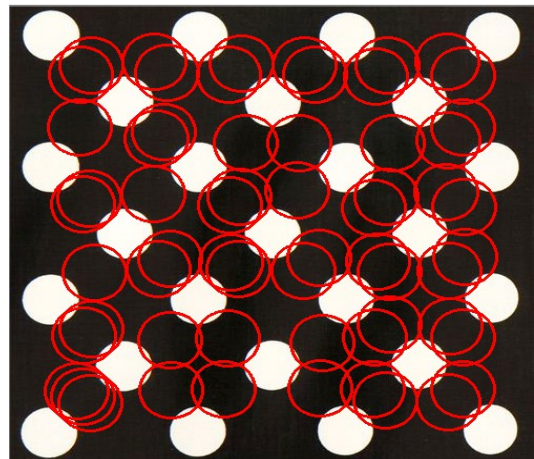
Detecting=both



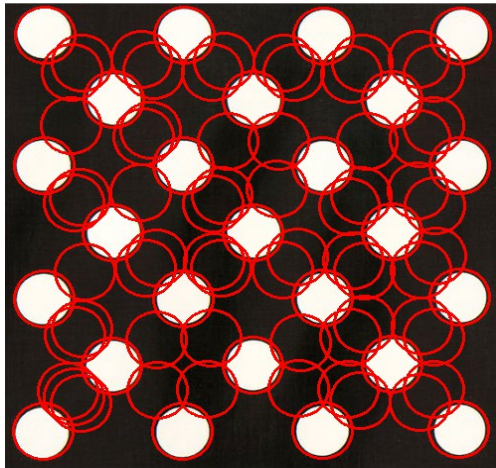
R=30
Detecting=Light Blobs



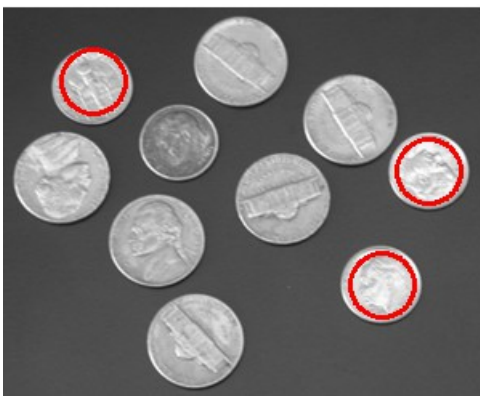
R=30
Detecting=Dark Blobs



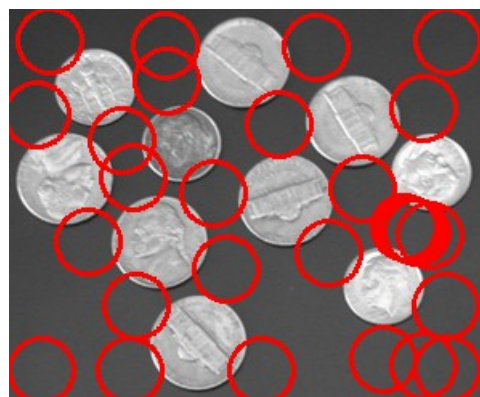
R=30
Detecting=both



Για την εικόνα coins.
R=20
Detecting=Light Blobs

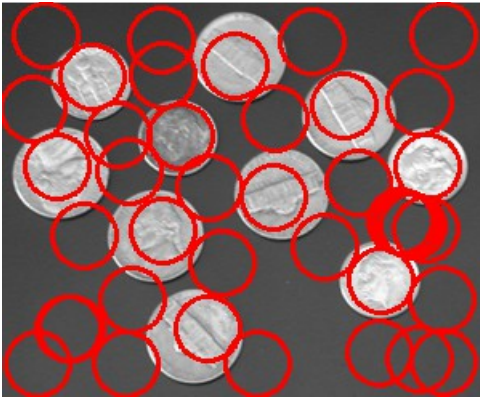


R=20
Detecting=Dark Blobs



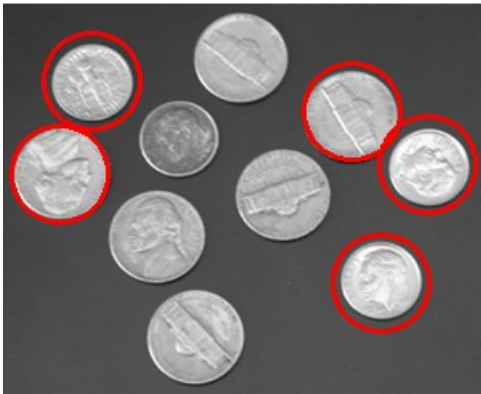
R=20

Detecting=both



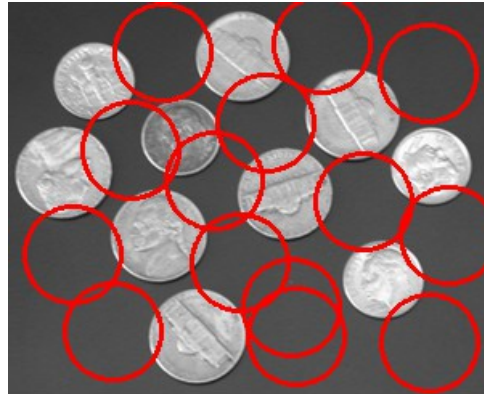
R=30

Detecting=Light Blobs



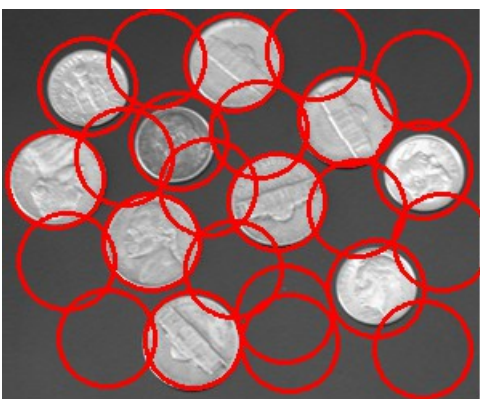
R=30

Detecting=Dark Blobs



R=30

Detecting=both



Για την εικόνα circles_02

R=20

Detecting=Light Blobs



R=20

Detecting=Dark Blobs



R=20

Detecting=both



R=30

Detecting=Light Blobs



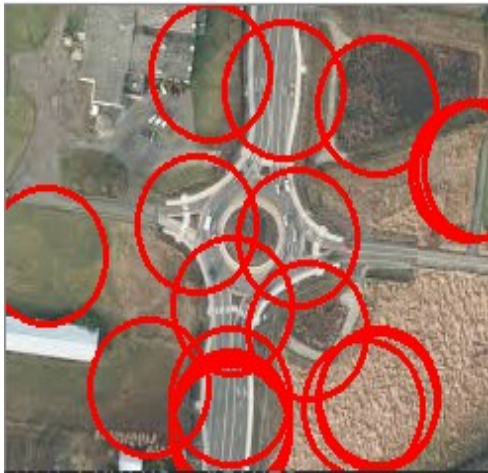
R=30

Detecting=Dark Blobs



R=30

Detecting=both



Και μερικές εικόνες για μεταβλητες τιμές ακτίνων:

Εδώ έχουμε ακτίνα 10-50 και εξετάζουμε light blobs



Εδώ έχουμε ακτίνα 10-50 και εξετάζουμε dark blobs



Εδώ έχουμε ακτίνα 10-30 και εξετάζουμε light blobs



Εδώ έχουμε ακτίνα 10-30 και εξετάζουμε dark blobs



Παρατηρούμε ότι ο αλγοριθμός μας δεν λειτουργεί όπως θα έπρεπε αλλά εμφανίζει blobs μεταβλητών ακτίνων.

Σημείωση: Σε αυτό το σημείο θα ήθελα να σας ενημερώσω πως ενώ χρησιμοποιώ το colab για κάποιο λόγο αντιμετώπισα πρόβλημα στο μενού επιλογών. Κολλούσε και δεν μπόρεσα να βρώ τι φταίει. Κατεβάζοντας το Anaconda και χρησιμοποιώντας το spyder λειτουργούσε κανονικά. Έτσι σας παραθέτω και τα δύο αρχεία εφόσον αρχικά την άσκηση την υλοποίησα στο colab και συνέχισα σε αυτό στην δεύτερη άσκηση.

Exercise 2 - Feature Based Tracking and RANSAC [7/10]

The aim is to estimate the pose of an object in an image sequence. The template appearance of the object is extracted from the rst frame. The pose of the object in each frame is dened by the coordinates of the points on its contour (i.e. the four corners). Use the 'Melita' dataset to test your algorithm.

Steps:

1. Set the template in the rst frame, extract and store features from it.
2. For each frame:
 - Extract features and match them with the template features.
 - Using the feature correspondences calculate the transformation between the template and the object. Implement both the ane and the homography transformation to compare their performance on the task.
 - To deal with outliers use the RANSAC algorithm.
 - Using the estimated transformation transform the contour points of the template and draw the contour of the object in the current frame.

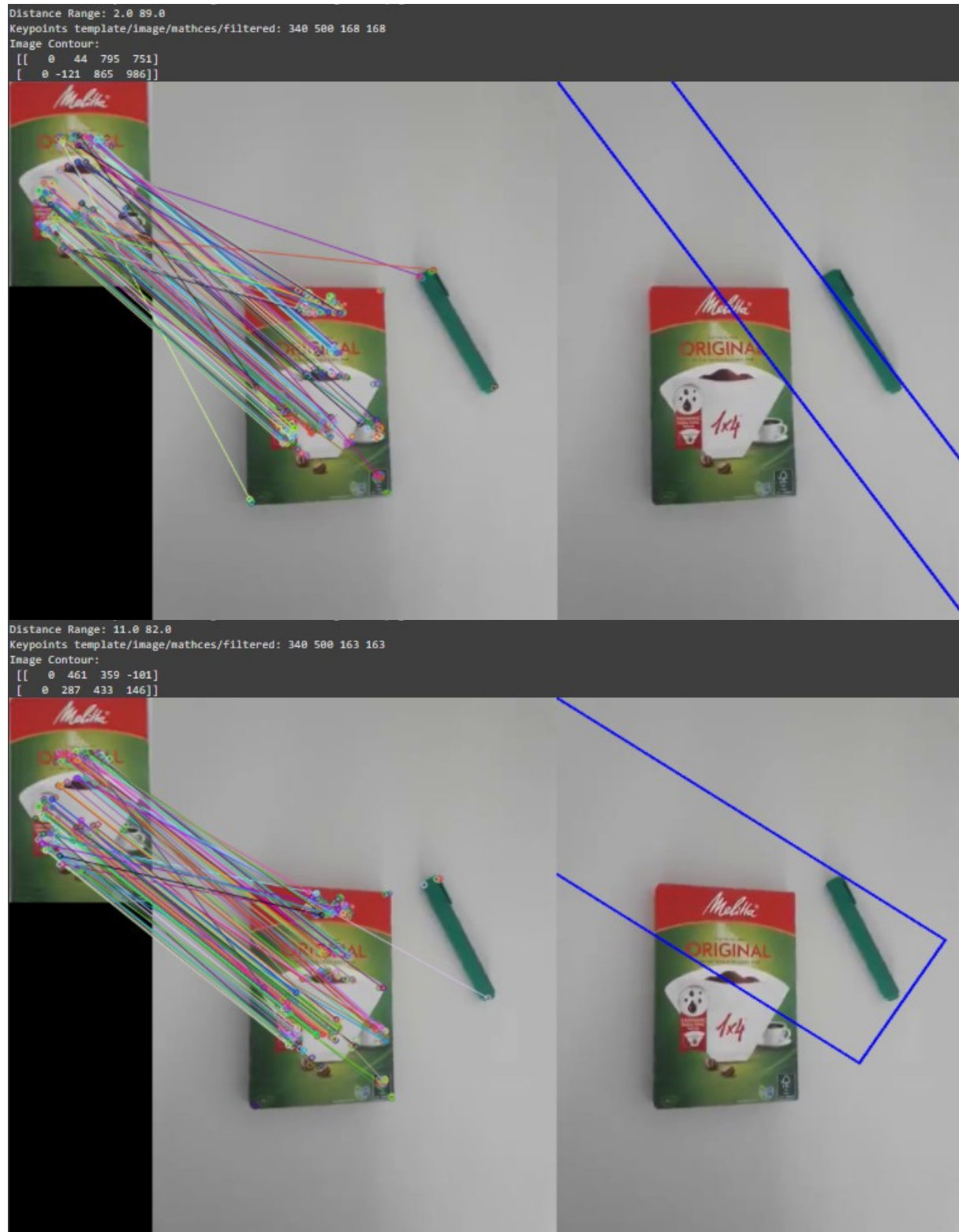
Περιγραφή:

Στην συγκεκριμένη άσκηση κληθήκαμε ουσιαστικά να δημιουργήσουμε έναν αλγόριθμο ο οποίος να μπορεί να αναγνωρίσει μια εικόνα σε διαφορετικά frames. Για να γίνει αυτό αρχικά δημιουργείται ένα template της εικόνας.



Στην συνέχεια θα έπρεπε να γίνει μετασχηματισμός Homography και Affine αρχικά χωρίς την χρήση Ransac και στην συνέχεια με την χρήση του έτσι ώστε να συγκρίνουμε τις εικόνες. Εδώ όπως θα παρατηρήσετε στον κώδικα και στις φωτογραφίες που θα σας δείξω παρακάτω ο μετασχηματισμός Homography δουλεύει λάθος. Ο μετασχηματισμός affine λειτουργεί προσεγγιστικά σωστά σε κάποιες περιπτώσεις αλλά όχι όπως πρέπει. Έτσι δεν μπορώ να προχωρήσω σε σύγκριση.

Μερικές εικόνες χρησιμοποιώντας Affine με Ransac.



Distance Range: 10.0 85.0
Keypoints template/image/mathces/filtered: 340 500 164 164
Image Contour:
[[0 379 416 37]
[0 223 484 261]]



Keypoints template/image/mathces/filtered: 340 500 150 150
Image Contour:
[[0 -527 989 1518]
[1 -676 1053 1730]]



Τα αποτελέσματα που πήρα χωρίς Ransac δεν είχαν κάποια διαφορά. Δυστυχώς χωρίς να ξέρω τον λόγο.Οπότε δεν έκρινα σκόπιμο να σας παρουσιάσω κάποια αποτελέσματα. Όπως επίσης και για την Homography η οποία δεν λειτουργεί καθόλου.

