

'Οραση Υπολογιστών: Τεχνική Αναφορά 2ης Εργασίας

Παπαδόπουλος Αριστείδης Α.Μ.:57576

• Εντοπισμός γωνιών:

Οι θέσεις των γωνιών προκύπτουν από τους SIFT & SURF, και όχι από τον ανιχνευτή του Harris (Harris Corner Detection) διότι τα αντικείμενα των εικόνων βρίσκονται σε διαφορετική κλίμακα. Επομένως χρειάζεται να χρησιμοποιηθεί ένας εντοπιστής που δεν επηρεάζεται από την αλλαγή κλίμακας. Έτσι, θα χρησιμοποιήσω τον εντοπιστή - περιγραφέα SIFT (Scale Invariant Feature Transform) και μια γρηγορότερη εκδοχή του, τον εντοπιστή - περιγραφέα SURF (Speeded Up Robust Features). Ως γωνία ορίζονται σημεία που έχουν μεταβολή στην φωτεινότητα και στις δύο κατευθύνσεις. Έτσι, ο εντοπιστής SIFT χρησιμοποιεί τα μέγιστα ή τα ελάχιστα που προκύπτουν από την διαφορά 2 Γκαουσσιανών κατανομών, με διαφορετική απόκλιση σ., που μεγιστοποιούν την εξίσωση αυτής της διαφοράς. Στην συνέχεια, για κάθε σημείο ενδιαφέροντος, προκύπτει ένας περιγραφέας, ένα διάνυσμα που περιέχει ορισμένα χαρακτηριστικά για το κάθε σημείο. Ο περιγραφέας προκύπτει με την εξής διαδικασία:

1. Σε κάθε σημείο ενδιαφέροντος τοποθετείται ένα παράθυρο διαστάσεων 16x16.
2. Κάθε παράθυρο χωρίζεται σε 4 μικρότερα κελιά, όπου κάθε κελί περιέχει εικονοστοιχεία που τοποθετούνται σε παράθυρα 4x4.
3. Για κάθε εικονοστοιχείο υπολογίζεται η κατεύθυνση της βαθμίδας.
4. Με χρήση κατωφλίου αφαιρούνται οι μεταβολές που είναι μικρότερες του ορίου.
5. Έτσι δημιουργείται ένα ιστόγραμμα 32 θέσεων για κάθε υποκελί, που είναι 8 διαστάσεων (λόγω των 8 διαφορετικών κατευθύνσεων της βαθμίδας) και τελικά δημιουργείται ένας περιγραφέας 128 θέσεων (8 διαστάσεις * 16 υποκελιά). Το τελικό αποτέλεσμα είναι ο περιγραφέας SIFT.

Ο περιγραφέας SURF είναι μία εύρωστη και γρήγορη εκδοχή του SIFT, όπου ακολουθείται η ίδια διαδικασία, με την διαφορά ότι ο περιγραφέας που προκύπτει είναι 64 θέσεων (4 διαστάσεις * 16 υποκελιά). Οι 4 διαστάσεις αποτελούν στοιχεία αθροίσματος, που είναι ανεξάρτητα της κάθε εικόνας/υποκελιού. Ακολουθούν τα αποτελέσματα στην εύρεση των σημείων ενδιαφέροντος (keypoints ή features) για κάθε εικόνα, πρώτα για τον περιγραφέα SIFT και μετά για τον SURF. Παρατηρούμε διαφορές στα αποτελέσματα της κάθε μεθοδολογίας, γιατί έχουν διαφορετική παράμετρο στον αριθμό των keypoints. Συγκεκριμένα, αν χρησιμοποιούσα στον εντοπιστή SURF τον ίδιο αριθμό με αυτόν του εντοπιστή SIFT, το αποτέλεσμα από τον εντοπιστή SURF ήταν κάτι εντελώς διαφορετικό από το επιθυμητό και πολύ πιο αργό από την αναμενόμενη ταχύτητα που περίμενα, επομένως δοκίμασα αρκετές τιμές, μέχρι να καταλήξω σε ένα ικανοποιητικό αποτέλεσμα για την συνέχεια της υλοποίησης (στο matching & cross-checking).



Εικ.1 Τα σημεία ενδιαφέροντος για την εικόνα hotel-00 με τον εντοπιστή SIFT.



Εικ.2 Τα σημεία ενδιαφέροντος για την εικόνα hotel-00 με τον εντοπιστή SURF.



Εικ.3 Τα σημεία ενδιαφέροντος για την εικόνα hotel-01 με τον εντοπιστή SIFT.



Εικ.4 Τα σημεία ενδιαφέροντος για την εικόνα hotel-01 με τον εντοπιστή SURF.



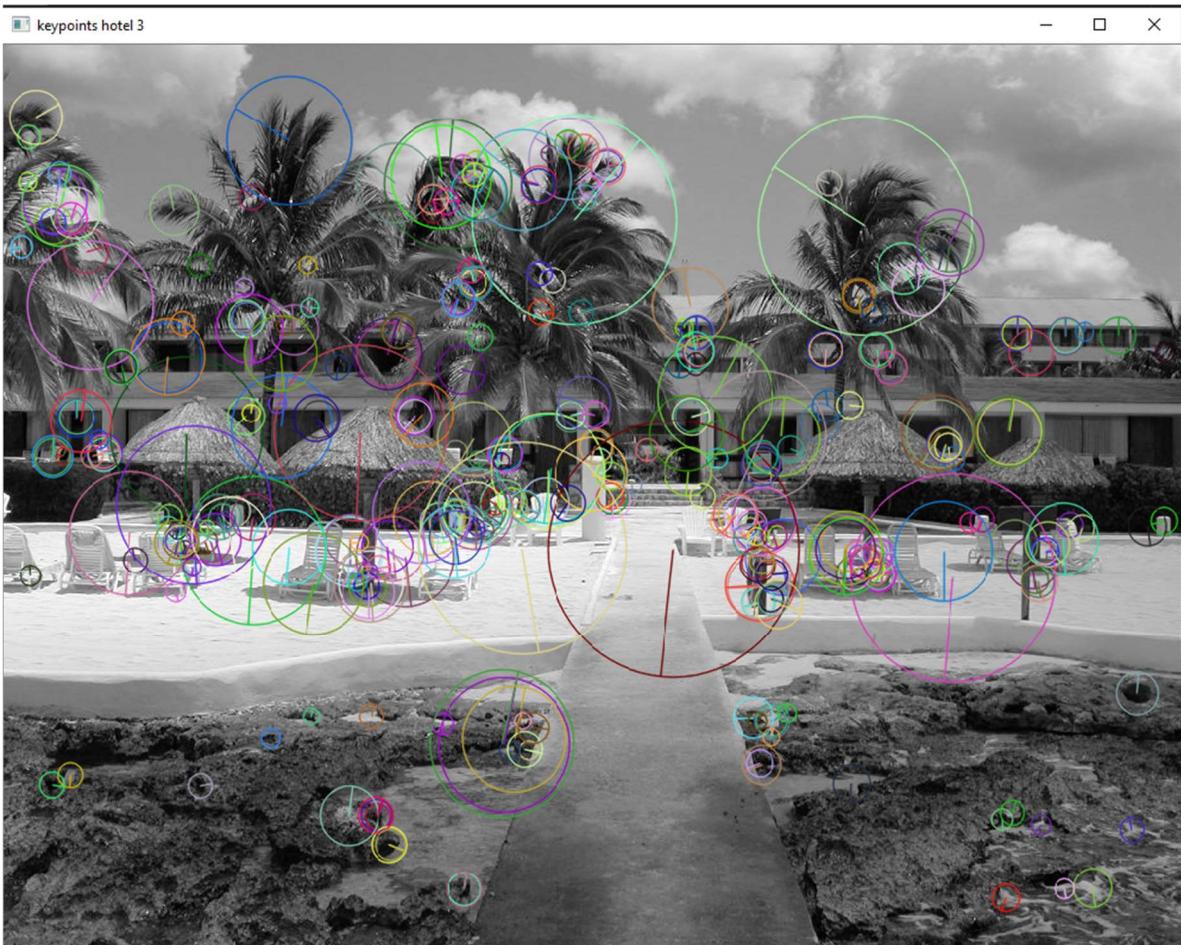
Εικ.5 Τα σημεία ενδιαφέροντος για την εικόνα hotel-02 με τον εντοπιστή SIFT.



Εικ.6 Τα σημεία ενδιαφέροντος για την εικόνα hotel-02 με τον εντοπιστή SURF.



Εικ.7 Τα σημεία ενδιαφέροντος για την εικόνα hotel-03 με τον εντοπιστή SIFT.



Εικ.8 Τα σημεία ενδιαφέροντος για την εικόνα hotel-03 με τον εντοπιστή SURF.



Εικ.9 Τα σημεία ενδιαφέροντος για το πανόραμα μεταξύ των εικόνων hotel01 – hotel00 με τον εντοπιστή SIFT.



Εικ.10 Τα σημεία ενδιαφέροντος για το πανόραμα μεταξύ των εικόνων hotel01 – hotel00 με τον εντοπιστή SURF.



Εικ.11 Τα σημεία ενδιαφέροντος για το πανόραμα μεταξύ των εικόνων hotel02,hotel01,hotel00 με τον εντοπιστή SIFT.



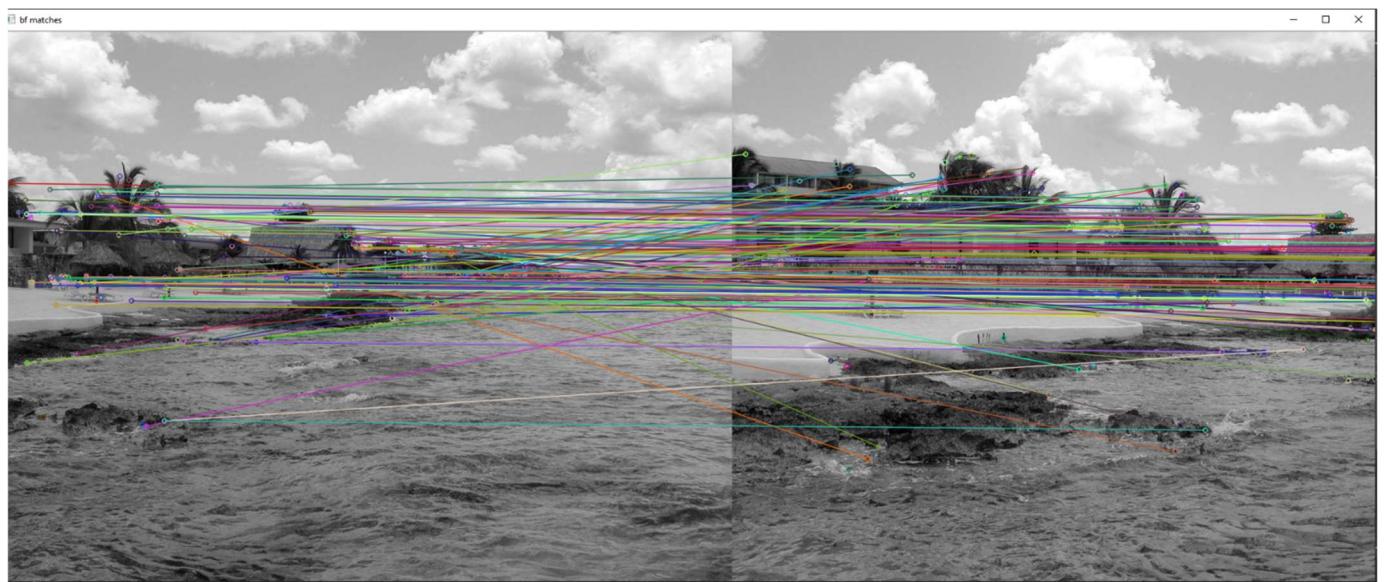
Εικ.12 Τα σημεία ενδιαφέροντος για το πανόραμα μεταξύ των εικόνων hotel02,hotel01,hotel00 με τον εντοπιστή SURF.

- **Συνταιριασμα και διασταύρωση (matching & cross-checking)**

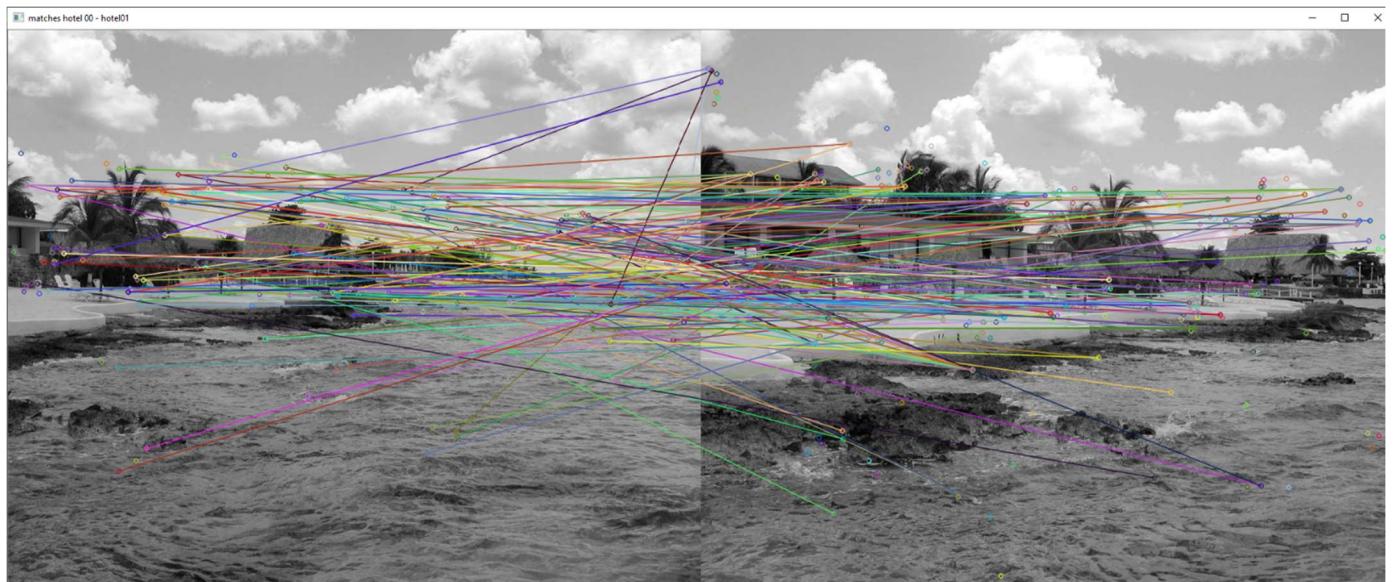
```
def matchcross(d, d1):
    matches = []
    for i in range(d.shape[0]):
        vector = d[i, :]
        euclid_dis = np.linalg.norm(d1 - vector, axis=1)
        idx = np.argmin(euclid_dis)
        min_dist = euclid_dis[idx]
        for j in range(d1.shape[0]):
            vector1 = d1[j, :]
            euclid_dis1 = np.linalg.norm(d - vector1, axis=1)
            idx1 = np.argmin(euclid_dis1)
            if idx == idx1:
                matches.append(cv2.DMatch(i, idx1, min_dist))
    return matches
```

Εικ.13 Η υλοποίηση της συνάρτησης που υλοποιεί τα συνταιριάσματα και το cross-checking.

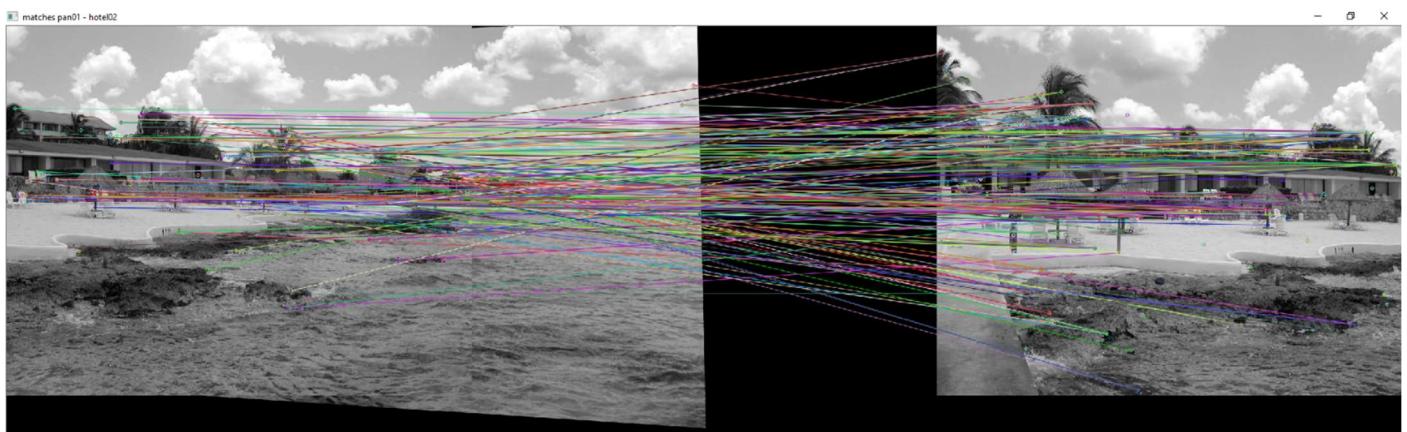
Το συνταιριασμα αλλά και η διασταύρωση των κοινών keypoints βασίζεται στην μεθοδολογία που κάναμε στο εργαστήριο στην συνάρτηση match1. Αρχικά απομονώνεται ο περιγραφέας κάθε keypoint της 1^{ης} εικόνας σε μία μεταβλητή, και στην συνέχεια υπολογίζεται η ευκλείδεια απόσταση (με την χρήση της συνάρτησης np.linalg.norm) του περιγραφέα που απομονώθηκε με κάθε περιγραφέα από την 2^η εικόνα με την οποία ψάχνουμε τα κοινά σημεία ενδιαφέροντος. Έπειτα, βρίσκουμε την θέση της ελάχιστης ευκλείδειας απόστασης και υπολογίζεται πόσο είναι αυτή. Ακολουθείται η ίδια διαδικασία για την 2^η με την 1^η εικόνα, και στην συνέχεια γίνεται έλεγχος αν η θέση της ελάχιστης απόστασης της 1^{ης} διαδικασίας (1^η με 2^η) είναι ίδια με την 2^η διαδικασία (2^η με 1^η). Αν ισχύει ο άνωθεν έλεγχος, τότε το ορίζεται ως συνταιριασμα (match) και αποθηκεύεται σε μια λίστα. Έτσι βρίσκονται τα κοινά σημεία ενδιαφέροντος μεταξύ 2 εικόνων, για να ακολουθήσει η διαδικασία της ομογραφίας και της τελικής "επικόλλησης" της μίας εικόνας πάνω στην άλλη. Σε σύγκριση με την ίδια διαδικασία που παρέχει ο brute force matcher, τα αποτελέσματα είναι σχεδόν πανομοιότυπα, όπως διαπιστώνεται από τις σχετικές εικόνες για τις εικόνες hotel00 και hotel01.



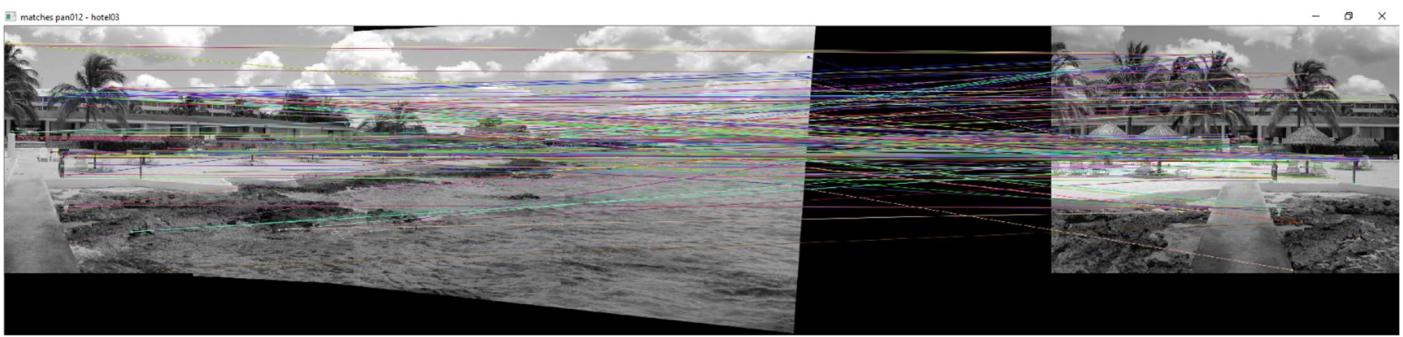
Εικ.14 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες hotel-00(αριστερά) και hotel-01(δεξιά) με τον περιγραφέα SIFT και την χρήση BFMatcher.



Εικ.15 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες hotel-00(αριστερά) και hotel-01(δεξιά) με τον περιγραφέα SIFT και την χρήση της "χειροκίνητης συνάρτησης".

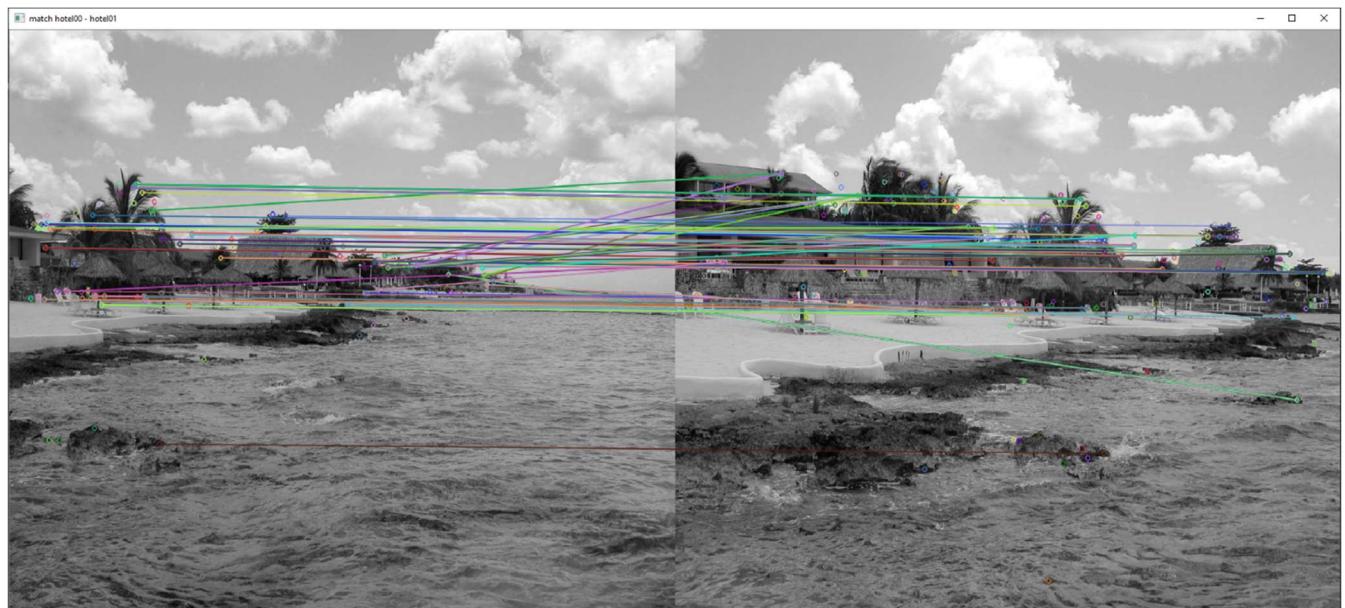


Εικ.16 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες pan01(αριστερά) και hotel-02(δεξιά) με τον περιγραφέα SIFT .

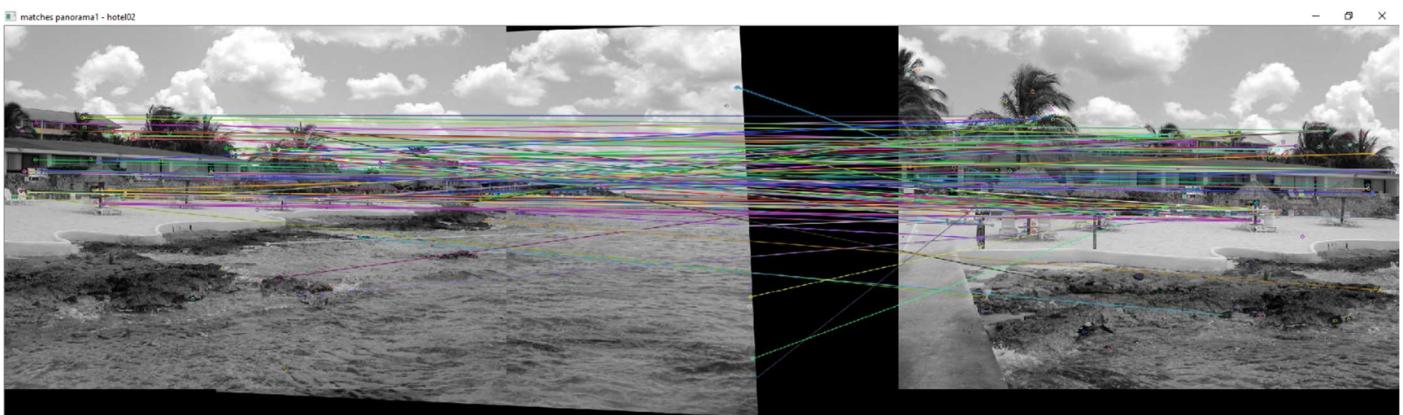


Εικ.17 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες pan012 (αριστερά) και hotel-03 (δεξιά) με τον περιγραφέα SIFT.

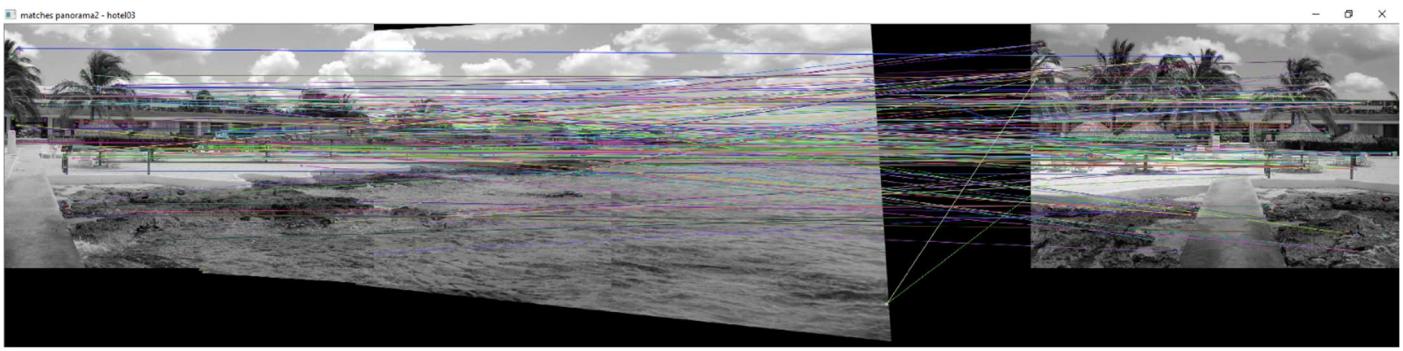
Για να δημιουργηθεί το πανόραμα, αρχικά συνένωσα τις εικόνες hotel00 και hotel01, και στην συνέχεια βρήκα περιγραφείς για το αποτέλεσμα τους και ακολούθησα την παραπάνω μεθοδολογία για αυτούς τους περιγραφείς με τους περιγραφείς της εικόνας hotel02. Τέλος, βρήκα τους περιγραφείς του αποτελέσματος της προηγούμενης διαδικασίας και ακολουθώντας την ίδια μεθοδολογία με πριν, χρησιμοποιώντας στο τελευταίο βήμα και τους περιγραφείς της εικόνας hotel03 για να προκύψει το τελικό πανόραμα. Ακολουθούν τα ταιριάσματα με τον εντοπιστή SURF.



Εικ.18 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες hotel-00(αριστερά) και hotel-01(δεξιά) με τον περιγραφέα SURF.



Εικ.19 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες panorama1 (αριστερά) και hotel-02(δεξιά) με τον περιγραφέα SURF.



Εικ.20 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες panorama2(αριστερά) και hotel-03(δεξιά) με τον περιγραφέα SURF.

Υπάρχουν ορισμένα ταιριάσματα, τα οποία είναι άστοχα, ταιριάσματα που δεν υπάρχουν καθόλου ή ταιριάσματα που θα έπρεπε να υπάρχουν αλλά δεν εντοπίζονται, λόγω του τρόπου υλοποίησης της συνάρτησης που πραγματοποιεί τα ταιριάσματα. Όμως, τα ταιριάσματα αυτά, θα απορριφθούν στην συνέχεια κατά την δημιουργία του πίνακα ομογραφίας με την μέθοδο του RANSAC, όπως θα αναλυθεί παρακάτω.

• Εύρεση του πίνακα ομογραφίας και δημιουργία πανοράματος

```

def points(k, k1, match):
    pt1 = []
    pt2 = []
    for x in match:
        pt1.append(k[x.queryIdx].pt)
        pt2.append(k1[x.trainIdx].pt)
    pt1 = np.array(pt1)
    pt2 = np.array(pt2)
    return pt1, pt2

def creating_pan(pt1, pt2, img1, img2):
    M, _ = cv2.findHomography(pt1, pt2, cv2.RANSAC)
    pan_out = cv2.warpPerspective(img1, M, (img1.shape[1]+img2.shape[1], img1.shape[0] + img2.shape[0]))
    pan_out[0:img2.shape[0], 0:img2.shape[1]] = img2
    return pan_out

```

Εικ.21 Η υλοποίηση της εύρεσης των συντεταγμένων των ταιριασμάτων, του πίνακα ομογραφίας και της δημιουργίας του πανοράματος.

Για να δημιουργηθεί ο πίνακας ομογραφίας, που θα μας δείξει πώς πρέπει να μετασχηματιστεί η μία εικόνα για να ταιριάξει με την άλλη, πρέπει να γίνει αντιστοιχίση των συνταιριασμάτων, με τις συντεταγμένες τους στην κάθε εικόνα. Η διαδικασία αυτή πραγματοποιείται με την συνάρτηση points, όπου έχει ως ορίσματα τα keypoints της μίας εικόνας, τα keypoints της δεύτερης εικόνας, και την λίστα με τα συνταιριάσματά τους. Για κάθε συνταιριασμό εντός της λίστας, τοποθετούνται σε μία λίστα οι συντεταγμένες που μας υποδεικνύει το συνταιριασμό για τις 2 εικόνες. Στην συνέχεια, δημιουργείται ο πίνακας ομογραφίας. Ο πίνακας ομογραφίας θα μπορέσει να μετασχηματίσει κάθε σημείο της μίας εικόνας με τέτοιο τρόπο, ώστε να ταιριάξει με το ομόλογό του στην δεύτερη εικόνα. Επιπλέον, γίνεται απόρριψη των ταιριασμάτων που ανήκουν στις κατηγορίες που αναφέρθηκαν στην προηγούμενη ενότητα, με την μέθοδο RANSAC. Το ακρωνύμιο RANSAC προκύπτει από τα αρχικά των Random Sample Consensus και ακολουθείται η εξής μεθοδολογία:

1. Επιλέγεται ένα ταιριασμό, και ακολουθείται ο προσανατολισμός του, και υπολογίζονται πόσα ταιριάσματα έχουν τον ίδιο προσανατολισμό, και ποια στοιχεία έχουν την μεγαλύτερη πιθανότητα να είναι σωστές οι αντιστοιχίσεις (inliers).

2. Κρατούνται όσα στοιχεία έχουν την μεγαλύτερη ποικιλία αντιστοιχίσεων ανά προσανατολισμό,ενώ τα υπόλοιπα παραμερίζονται(outliers).
3. Αυτό επαναλαμβάνεται μέχρι να σταματήσουν οι αλλαγές των στοιχείων.

Στην συνέχεια, περιστρέφεται σύμφωνα με τον πίνακα ομογραφίας η μία εικόνα, και επικολλάται η δεύτερη εικόνα, επικαλύπτοντας τα κοινά τους σημεία. Η διαδικασία που ακολουθησα για να παράγω το τελικό πανόραμα είναι η ακόλουθη:

1. Ξεκινάω με τις εικόνες hotel00, hotel01 και βρίσκω το πανόραμά τους(pan01).
2. Υπολογίζω τα keypoints και τους περιγραφείς του αποτελέσματος, και βρίσκω το πανόραμα για το πανόραμα pan01 και την εικόνα hotel02(προκύπτει το pan012).
3. Επαναλαμβάνω το βήμα 2, και βρίσκω το τελικό πανόραμα μεταξύ των pan012 και hotel03.

Δεν ακολουθησα την αντίστροφη διαδικασία, δηλαδή πανόραμα για τις hotel03, hotel02 κ.ο.κ. διότι στοιχεία της εικόνας hotel03 μετατοπίζονται με τέτοιο τρόπο, όπου καταλήγουν στα αρνητικά(εκτός του περιθωρίου της εικόνας) και δεν γίνεται να προβληθούν. Ακολουθούν τα πανοράματα για κάθε εντοπιστή SIFT, SURF.



Εικ.22 Το πανόραμα μεταξύ των εικόνων hotel00(στα δεξιά) και hotel01(στα αριστερά).



Εικ.23 Το πανόραμα μεταξύ των εικόνων hotel02(στα αριστερά) και του προηγούμενου πανοράματος(στα δεξιά).



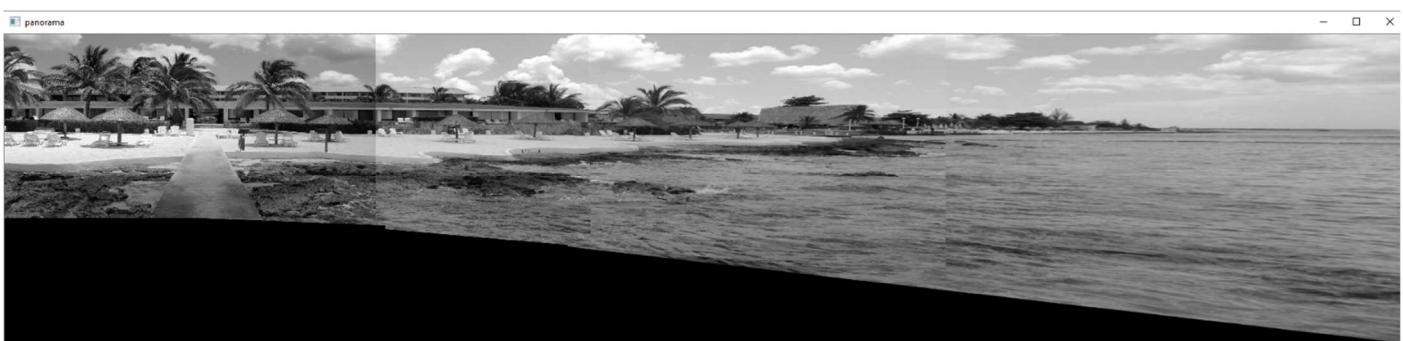
Εικ.24 Το τελικό πανόραμα με την μέθοδο SIFT.



Εικ.25 Το πανόραμα μεταξύ των εικόνων hotel00(στα δεξιά) και hotel01(στα αριστερά) με SURF.



Εικ.25 Το πανόραμα μεταξύ των εικόνων hotel02(στα αριστερά) και του προηγούμενου πανοράματος (στα δεξιά) με SURF.



Εικ.26 Το τελικό πανόραμα με την μέθοδο SURF.

Τα αποτελέσματα και των 2 τρόπων, είναι πανομοιότυπα. Μεταξύ στο πανόραμα των εικόνων hotel00, hotel01 παρατηρείται μια ασυνέχεια κατά μήκος του κατακόρυφου άξονα στο σημείο που επικαλύπτονται οι 2 εικόνες (όπως και μεταξύ των υπολοίπων πανοραμάτων). Αυτό οφείλεται στην

διαδικασία που δημιουργήθηκε το πανόραμα, επειδή η μία εικόνα επικάθεται πάνω σε αυτήν που μετασχηματίστηκε, καθώς και σε διαφορετικό περιεχόμενο κατά την διαδικασία λήψεων των φωτογραφιών (π.χ. στην αποτύπωση των κυμάτων).



Εικ.27 Παράδειγμα ασυνέχειας κατά την δημιουργία πανοράματος (εικόνες hotel01-hotel00).

- **Σύγκριση των αποτελεσμάτων με αυτά του Image Composite Editor(ICE)**



Εικ.28 Αρχικό πανόραμα με την χρήση του προγράμματος Image Composite Editor.

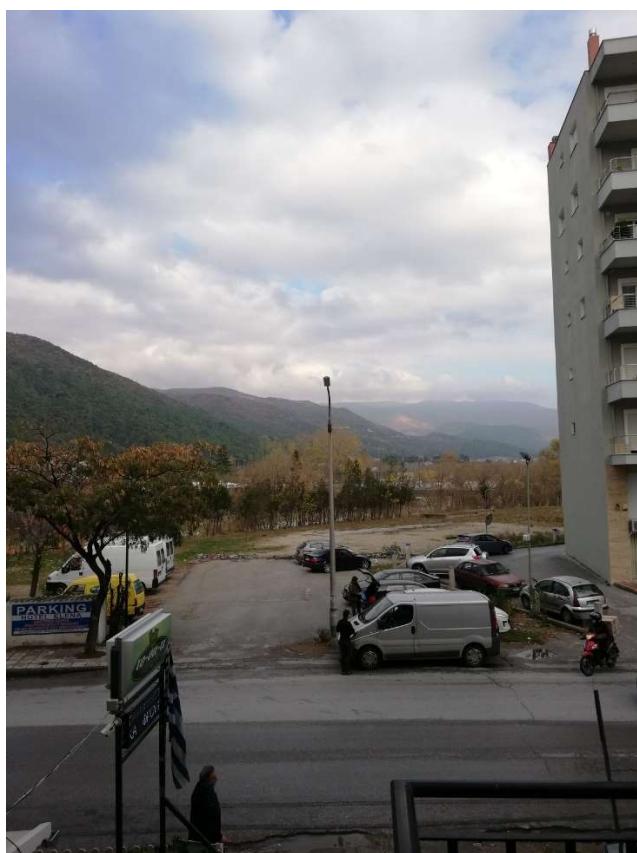
Αρχικά, μέσω του ICE λαμβάνεται το παραπάνω αποτέλεσμα. Οι διαφορές που εντοπίζονται οφείλονται στο γεγονός ότι το ICE επιλύει το πρόβλημα της παρούσας υλοποιησης, δηλαδή δημιουργεί ένα πανόραμα με κέντρο τις εικόνες hotel02-hotel03, και μετασχηματίζει τις υπόλοιπες, χωρίς να ξεφεύγει εκτός περιθωρίου. Αντίθετα, η υλοποιηση μου χρησιμοποιεί ως κέντρο την εκάστοτε εικόνα που απεικονίζει το "αριστερότερο" τοπίο. Ωστόσο, αν πειραχτούν οι ρυθμίσεις του ICE με τέτοιο τρόπο, ώστε στο κέντρο να βρεθεί η εικόνα hotel03, το αποτέλεσμα είναι πολύ κοντά σε αυτό της υλοποιησής μου.



Εικ.29 Πανόραμα που προκύπτει "πειράζοντας" τις ρυθμίσεις του ICE.

- **Επανάληψη διαδικασίας με δικές μου εικόνες**

Οι εικόνες που έβγαλα αρχικά ήταν διαστάσεων 3456x4608, οπότε χρειάστηκε να μειώσω τις διαστάσεις τους σε 1200x1600, γιατί σε διαφορετική περίπτωση θα χρειαζόταν πολύ περισσότερος χρόνος για να προκύψει το επιθυμητό αποτέλεσμα. Οι εικόνες είναι η εξής:

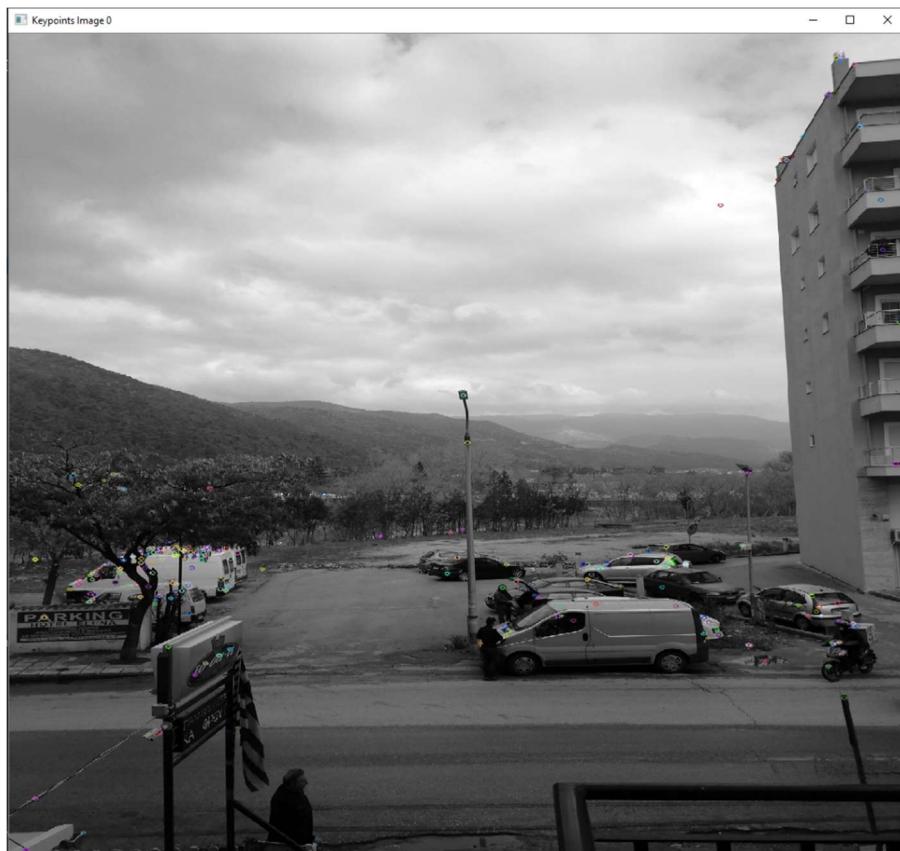


Εικ.30,31 Αρχικές εικόνες που χρησιμοποιήσα, με όνομα view0(αριστερά) και view1(δεξιά).

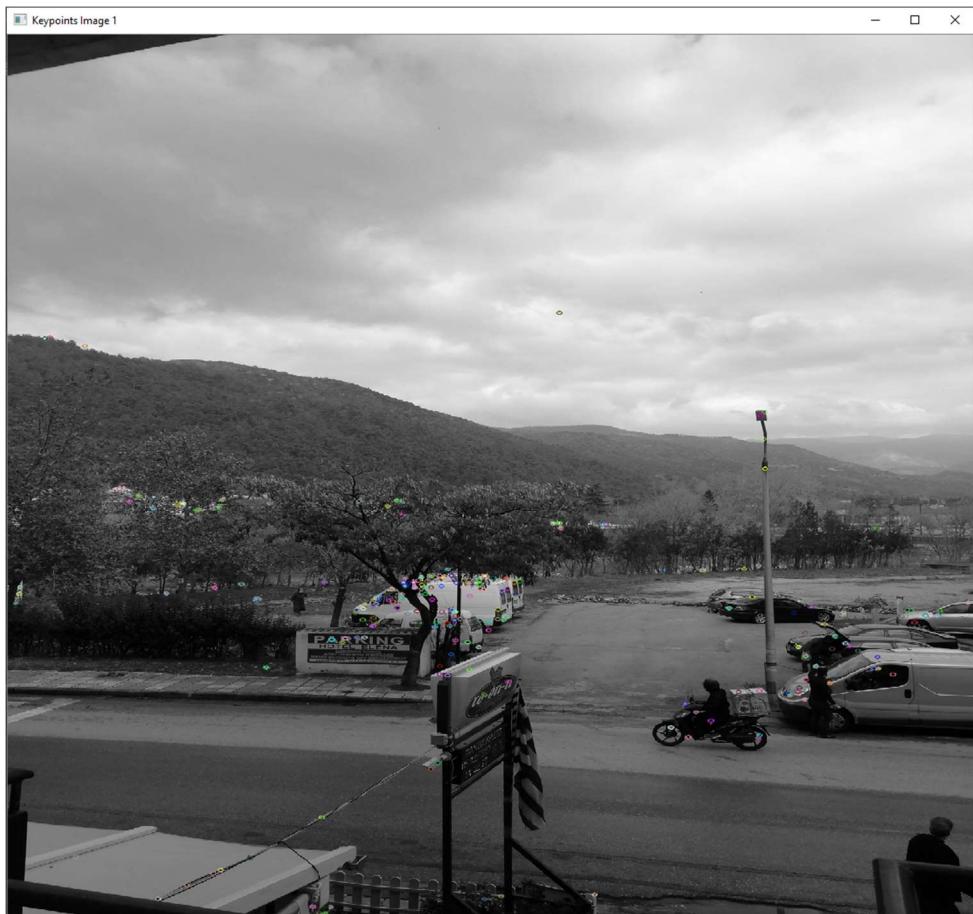


Εικ.32,33 Αρχικές εικόνες που χρησιμοποιήσα, με όνομα view2(αριστερά) και view3(δεξιά).

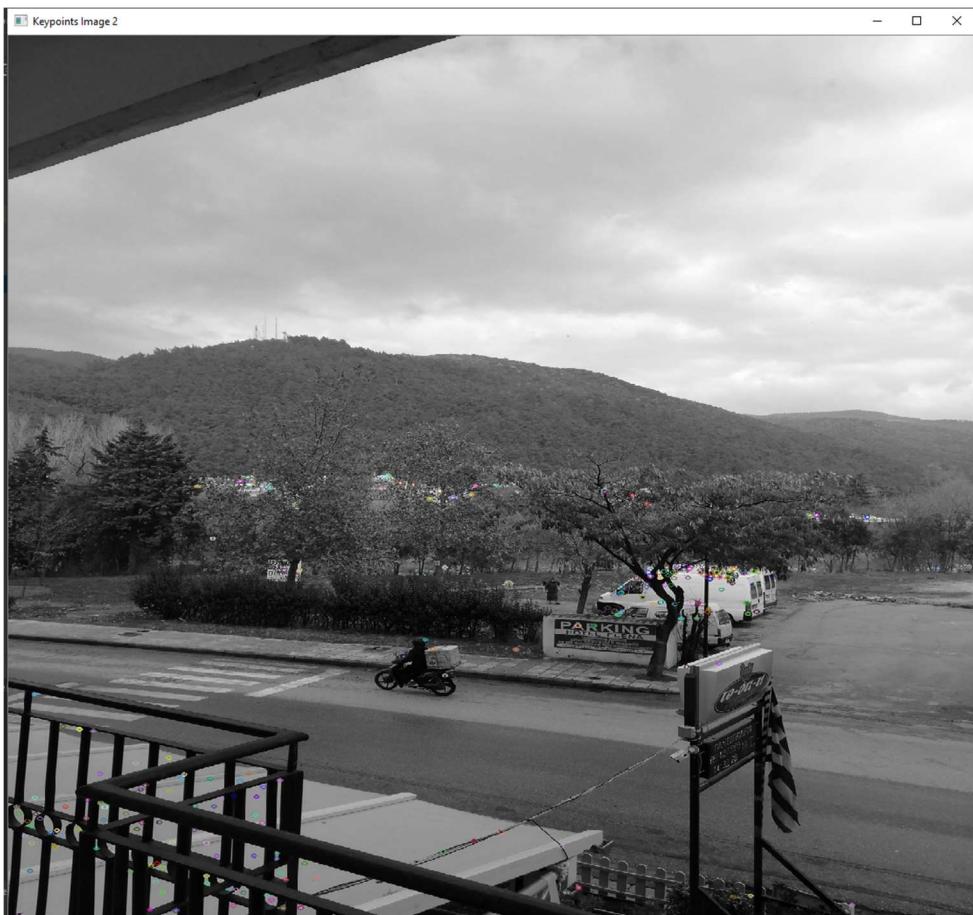
Ξεκινώντας την ίδια διαδικασία για τις προηγούμενες εικόνες, αρχικά θα βρεθούν τα κοινά σημεία ενδιαφέροντος, όπως γίνεται αντιληπτό από τις ακόλουθες εικόνες, ενώ όποια ταιριάσματα είναι λάθος, απομακρύνονται με την διαδικασία του RANSAC που αναλύθηκε προηγουμένως. Ακολουθούν τα αποτελέσματα από τον εντοπισμό των σημείων ενδιαφέροντος, τα συνταιριάσματα, τα επιμέρους πανοράματα καθώς και το τελικό πανόραμα.



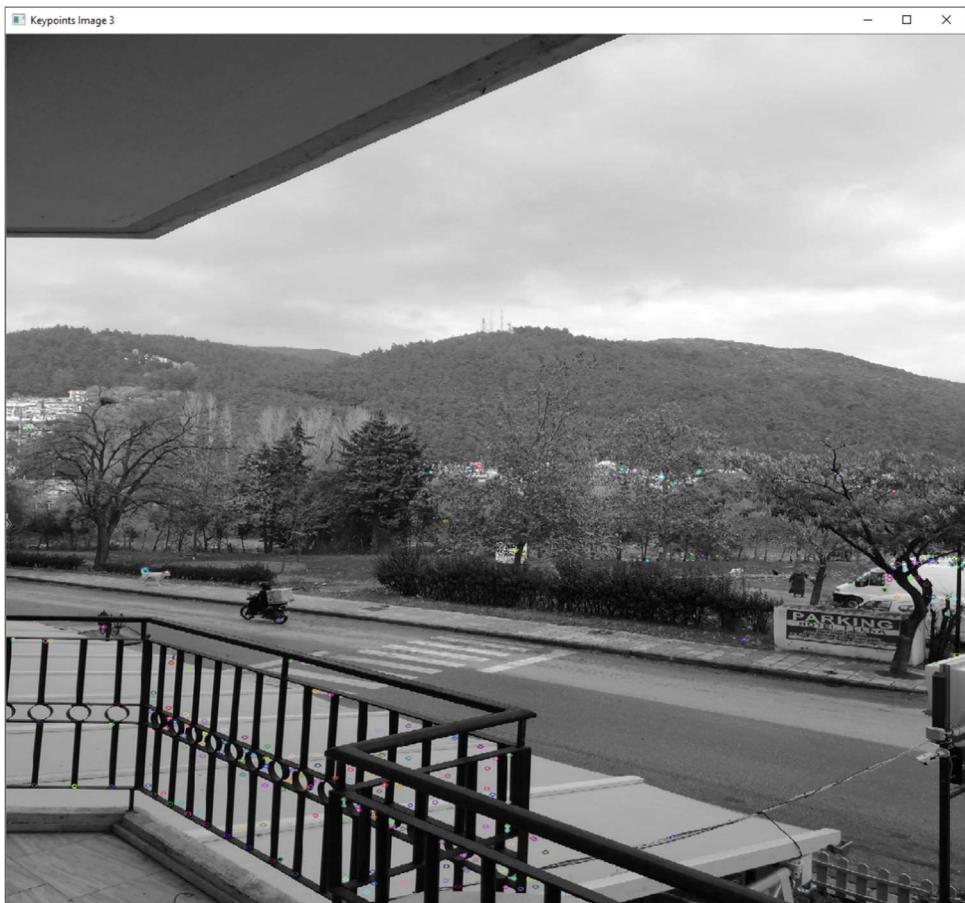
Εικ.34 Τα σημεία ενδιαφέροντος με τον εντοπιστή SIFT, για την εικόνα view0.



Εικ.35 Τα σημεία ενδιαφέροντος με τον εντοπιστή SIFT, για την εικόνα view1.



Εικ.36 Τα σημεία ενδιαφέροντος με τον εντοπιστή SIFT, για την εικόνα view2.



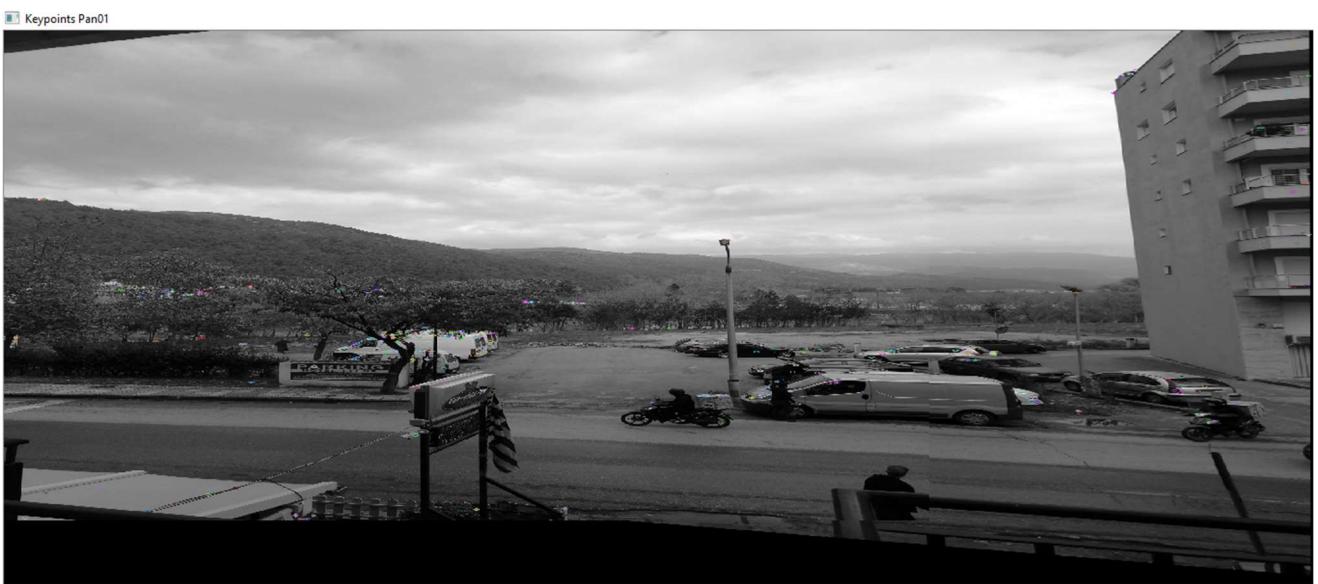
Εικ.37 Τα σημεία ενδιαφέροντος με τον εντοπιστή SIFT, για την εικόνα view3.



Εικ.38 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες view0(αριστερά) και view1(δεξιά) με τον περιγραφέα SIFT.



Εικ.39 Το πανόραμα που προκύπτει μεταξύ των εικόνων view1 και view0 με τον περιγραφέα SIFT.



Εικ.40 Τα σημεία ενδιαφέροντος για το προηγούμενο πανόραμα.



Εικ.41 Τα κοινά σημεία ενδιαφέροντος για το προηγούμενο πανόραμα(αριστερά) και view2(δεξιά) με τον περιγραφέα SIFT.



Εικ.42 Το πανόραμα που προκύπτει για τα προηγούμενα κοινά σημεία ενδιαφέροντος.



Εικ.43 Τα σημεία ενδιαφέροντος για το προηγούμενο πανόραμα.



Εικ.44 Τα κοινά σημεία ενδιαφέροντος για το προηγούμενο πανόραμα(αριστερά) και view3(δεξιά) με τον περιγραφέα SIFT.



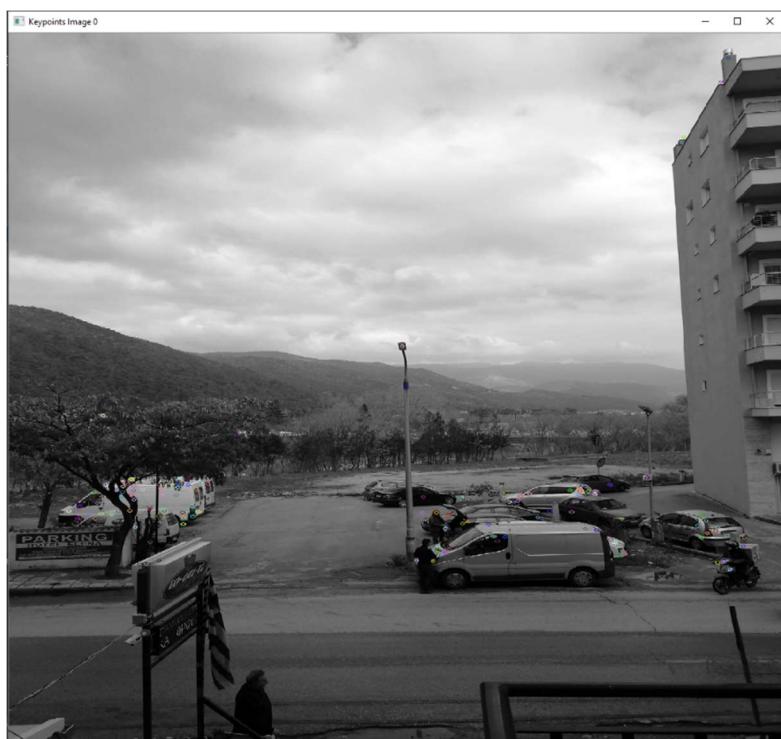
Εικ.45 Το τελικό πανόραμα για τις εικόνες που χρησιμοποίησα με τον περιγραφέα SIFT.

Το τελικό πανόραμα που προκύπτει, αν και επιτυχές, έχει κάποιες αστοχίες, που οφείλονται εν μέρη στην επιλεγμένη υλοποίηση, αλλά και στα αντικείμενα των φωτογραφιών. Αρχικά, λόγω της απεικόνισης του δρόμου σε ώρες κίνησης, εντοπίζεται το μηχανάκι σε 3 θέσεις ταυτόχρονα, γεγονός που οφείλεται στην κίνηση που κάνει. Επιπλέον, στα σημεία σύνδεσης των εικόνων, εντοπίζονται κάποιες ασυνέχειες (για παράδειγμα στα κάγκελα στο δεξί μέρος του πανοράματος), που οφείλονται στην υλοποίηση, ενώ τέλος, φαίνεται να κόβεται το άνω μέρος της δεξιάς εικόνας (view0). Αντίθετα, το πανόραμα που προκύπτει από τον Image Composite Editor (είναι με χρώμα, διότι επέλεξα να "διαβάσω" τις εικόνες με grayscale στο PyCharm) φαίνεται να διορθώνει όλα τα παραπάνω προβλήματα.



Εικ.46 Πανόραμα που προκύπτει "πειράζοντας" τις ρυθμίσεις του ICE (όπως αναφέρθηκε σε προηγούμενη ενότητα).

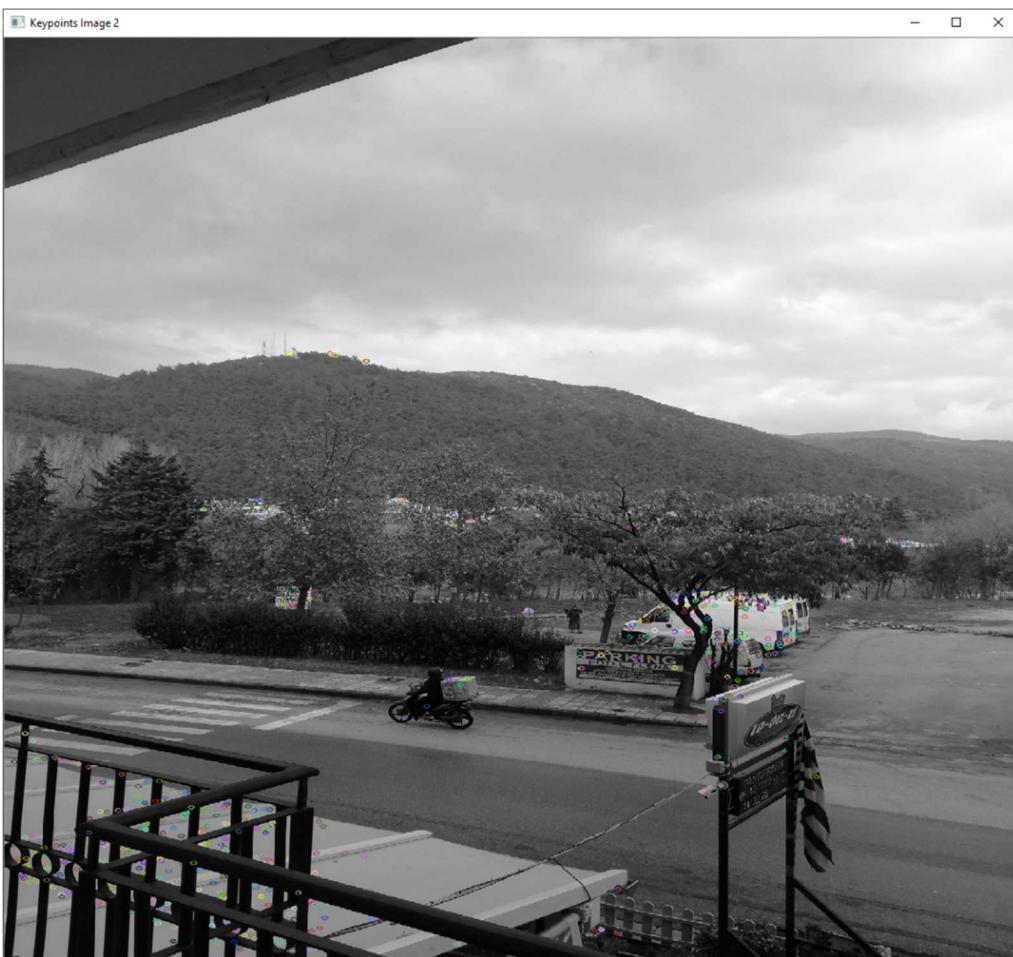
Ακολουθούν τα αποτελέσματα με την χρήση περιγραφέα SURF, όπου τα συμπεράσματα, αλλά και τα προβλήματα παραμένουν τα ίδια με την χρήση του περιγραφέα SIFT.



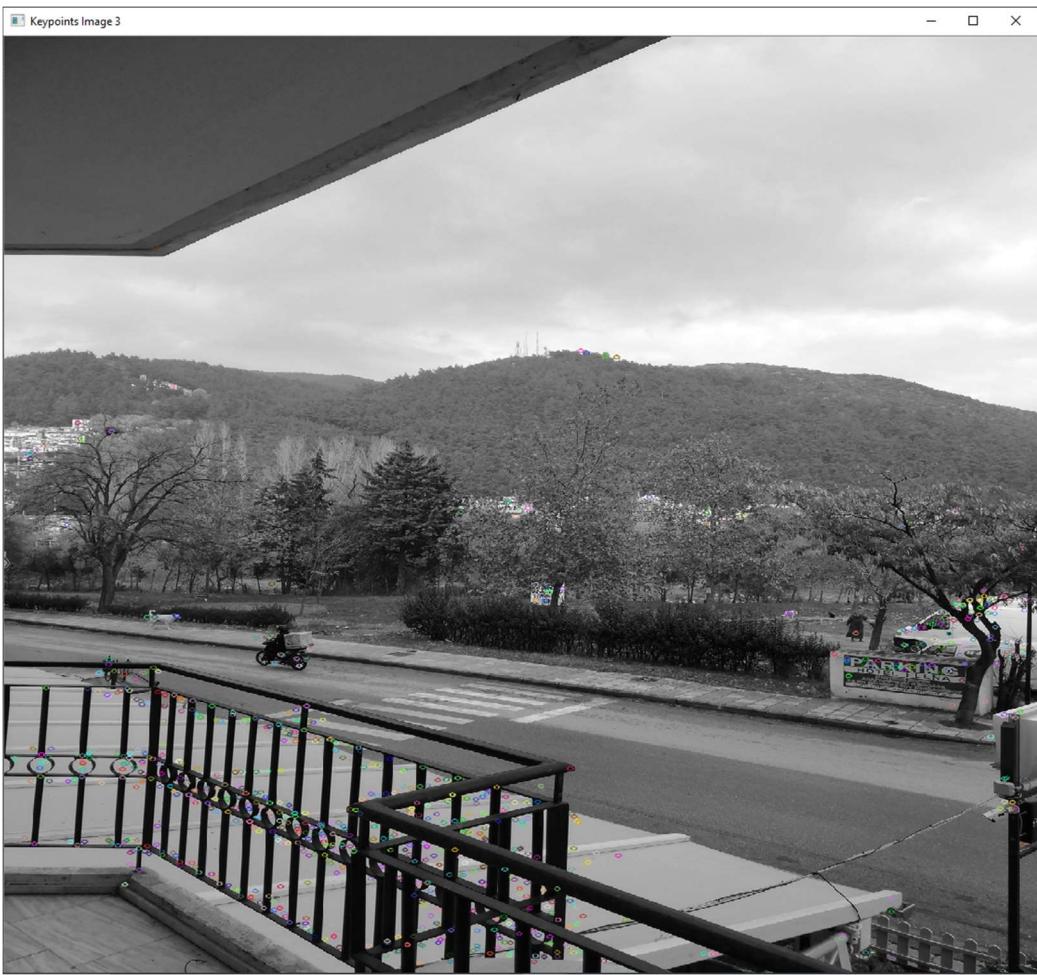
Εικ.47 Τα σημεία ενδιαφέροντος με τον εντοπιστή SURF, για την εικόνα view0.



Εικ.48 Τα σημεία ενδιαφέροντος με τον εντοπιστή SURF, για την εικόνα view1.



Εικ.49 Τα σημεία ενδιαφέροντος με τον εντοπιστή SURF, για την εικόνα view2.



Εικ.50 Τα σημεία ενδιαφέροντος με τον εντοπιστή SURF, για την εικόνα view3.



Εικ.51 Τα κοινά σημεία ενδιαφέροντος για τις εικόνες view0(αριστερά) και view1(δεξιά) με τον περιγραφέα SURF.

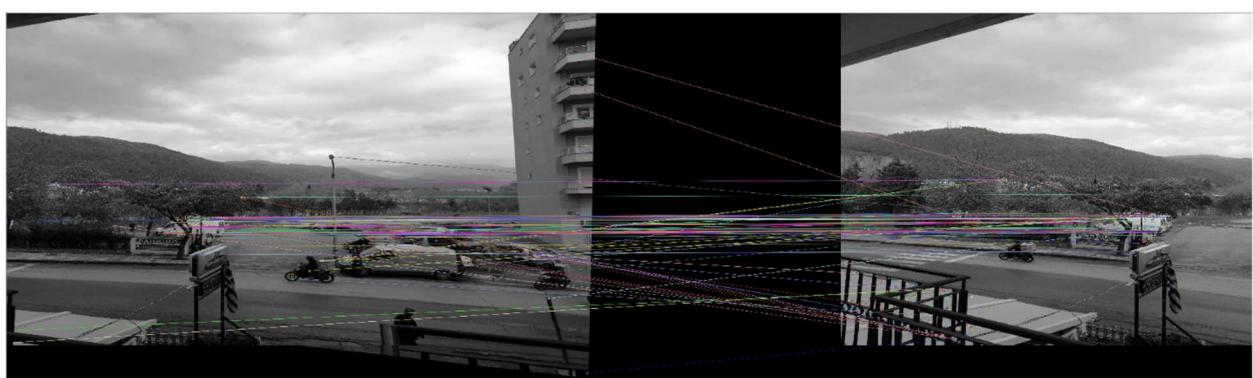
pan01



Εικ.52 Το πανόραμα που προκύπτει για τις εικόνες view1-view2.



Εικ.53 Τα σημεία ενδιαφέροντος για το προηγούμενο πανόραμα.



Εικ.54 Τα κοινά σημεία ενδιαφέροντος για το προηγούμενο πανόραμα(αριστερά) και view2(δεξιά) με τον περιγραφέα SURF.



Εικ.55 Το πανόραμα που προκύπτει για τις εικόνες view2,view1,view0.



Εικ.56 Τα σημεία ενδιαφέροντος για το προηγούμενο πανόραμα.



Εικ.57 Τα κοινά σημεία ενδιαφέροντος για το προηγούμενο πανόραμα(αριστερά) και view3(δεξιά) με τον περιγραφέα SURF.



Εικ.58 Το τελικό πανόραμα με την χρήση του περιγραφέα SURF.

Εν κατακλείδι,και τα δύο πανοράματα παρουσιάζουν το ίδιο αποτέλεσμα,με τις ίδιες αστοχίες και με τις ίδιες επιτυχίες.Οι αστοχίες θα μπορούσαν να εξαλειφθούν αν ήταν διαφορετική η υλοποίηση ή με την χρήση διαφορετικών εικόνων(για παράδειγμα κάποιο τοπίο).Ωστόσο,θεωρώ πως το αποτέλεσμα είναι ικανοποιητικό,αν ληφθεί υπόψη το αντικείμενο των επιλεγμένων φωτογραφιών,καθώς στα σημεία που δεν υπάρχει κίνηση(όπως ο ουρανός ή τα βουνά),υπάρχει μια επιτυχής προσπάθεια στην δημιουργία πανοράματος.

• **Βιβλιογραφία**

- Richard Szeliski: Computer Vision Algorithms and Applications
- NumPy Documentation : <https://numpy.org/doc/stable/contents.html>
- OpenCV Documentation : <https://docs.opencv.org/3.4/index.html>
- David G. Lowe : Distinctive Image Features from Scale-Invariant Keypoints -
https://www.robots.ox.ac.uk/~vgg/research/affine/det_eval_files/lowe_ijcv2004.pdf