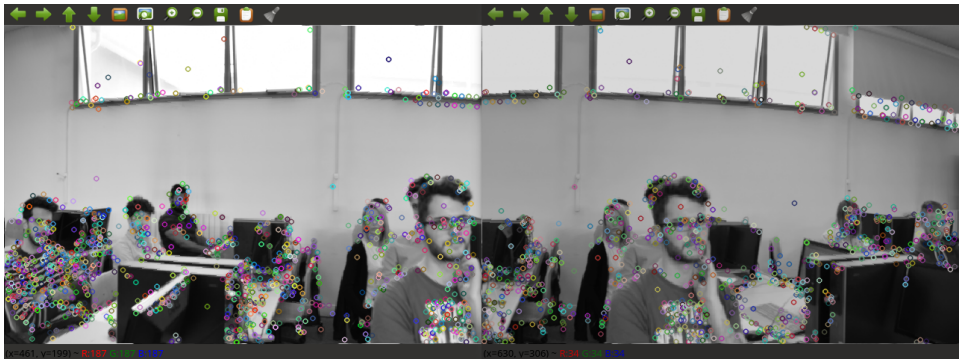
Laboratory report (Features and Panoramic Image) - Luca Dolci 1234008
The program provides the class `PanoramicImage` which compute the panoramic image by merging together ome input images. The public interface is composed by the `loadImages()` method, which harvest the images from a given directory, and by the `elaborate()` method which performs the computation in steps:

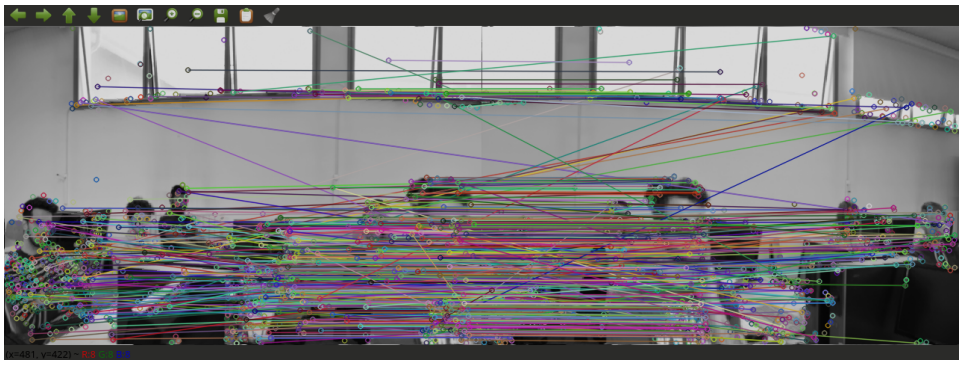
1. Project every single image in a cylindrical surface. The FOV parameter is differente for each dataset provided.

2. Extract the feature from each different image. This is done using SIFT features (with l_2 norm). In this step the descriptors for each keypoints are also computed. The methods `detect()` and `compute()` of the class `SIFT` are used for the keypoints and the descriptors, respectively. The results after this step is the seguent:



   In this two images 1266 and 992 feaures are detected for the first and second image.

3. Features matching. Performed using the brute force method (mehtod `match()` of the class BGMatcher, which requires two set of descriptors computed in previous step), the result is set of istances of the DMatch class. Each istance provides the distance of the two keypoints and the index of the matched keypoints, one for the left image and one for the right image. The result of this step on the first two images is the seguent:



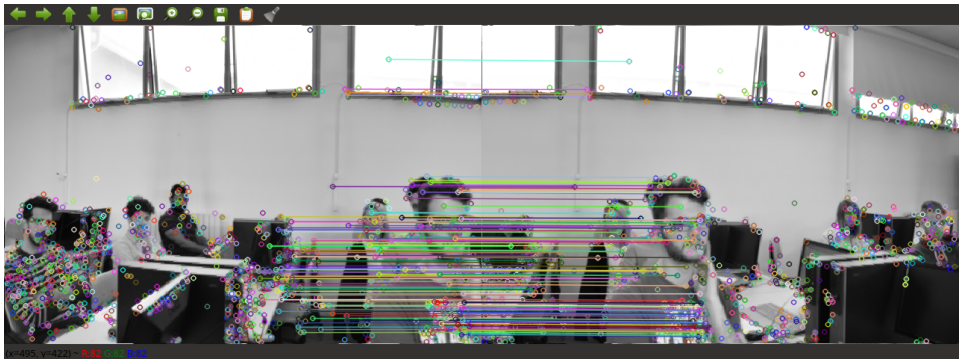
   457 matches are found.

4. Matches refinement. This step has two substep:

   (a) Discard matches based on distance. The matches for each pair of images are discarded if their distance is less or equal than the minmum times the ratio, a crucial parameter that the user can set at the beginning of the computation. This can change the alignment (a future step). A low value can cause a poor alignment because lots of matches will be discarded and the alignment will be based on few matches, on the other hands, with a too high value some fake matches can remain, which can cause disaligment.

   (b) Discard matches based on homography. The matches for each pair of images are discarded if they are labeled as 'outilers' by the RANSAC algorithm. This actually helps a lot, discarding matches with different 'slope' (see figure below) from the rest of the matches.

   The result, with ratio equal to 10, is the seguent:



   The remaining matches are 164. With ratio equals to two ony 9 matches remains.

5. Alignment. For each image and for each matches set (one on the left and one on the right) the mean position of the keypoints are computed. This represent the translation on the x and y axis, which are indipendet:

   - on the x axis: each image is cropped such that the left margin match the x coordinate of the mean position of keypoints matched with the left image, the same on the right. If this coordinates overlaps, the image is discarded.
   - on the y axis: each image is translated with respect to the posion on the first image, computing the local translation with the left image (difference between the y coordinate in the positions of the mathced keypoints) and performing the translation using an affine transformation and a transformation matrix.

   Finally the whole image can be cropped in order to avoid black strips. The final result is the seguent: