M4 Project
Week 2: 3D recovery of urban scenes

Group members:

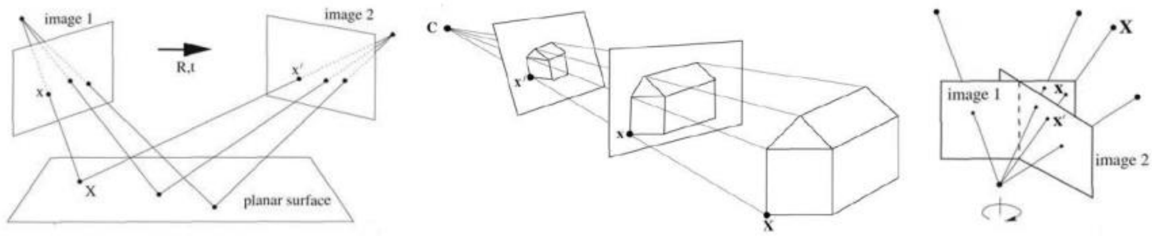
José Manuel López, Alex Martin, Marcos V. Conde, Oriol Catalan Yarza

Group ID: 4

1. Introduction

In this second session of the project, we aim to understand homographies and how they relate to images. Also, we review different methods to obtain homographies and their applications:

- Algebraic Method: Robust Normalized DLT.
- Geometric Method: Gold-Standard.
- Camera Calibration from planar patterns.
- Image mosaics.
- Logo Replacement.



2. Robust DLT algorithm

To perform the estimation of the homography between two images, it can be used the direct linear transform algorithm. Taking into account that two images have points that appear at both, we can extract the homography using the coordinates of these points at each of the images. The points from an image to another image' can be transformed with the relationship:

$$x' = H \cdot x$$

If we express the homography H as a column of transposed vectors, we can express the equation as:

$$\begin{pmatrix} x_i' \\ y_i' \\ w_i' \end{pmatrix} = \begin{pmatrix} \mathbf{h}_1^T \mathbf{x}_i \\ \mathbf{h}_2^T \mathbf{x}_i \\ \mathbf{h}_3^T \mathbf{x}_i \end{pmatrix}$$

From this relationship and transforming from homogeneous coordinates to projective coordinates each side, we can obtain the following set of equations:

$$x_i' \mathbf{h}_3^T \mathbf{x}_i - w_i' \mathbf{h}_1^T \mathbf{x}_i = 0$$

$$y_i' \mathbf{h}_3^T \mathbf{x}_i - w_i' \mathbf{h}_2^T \mathbf{x}_i = 0$$

which can be disposed in a multiplication of a matrix (2x9) with another matrix (9x1):

$$\begin{pmatrix} \mathbf{0}^T & -w_i' \mathbf{x}_i^T & y_i' \mathbf{x}_i^T \\ w_i' \mathbf{x}_i^T & \mathbf{0}^T & -x_i' \mathbf{x}_i^T \end{pmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix} = \mathbf{0}$$

So then by solving this equation with the SVD the values of the homography that relates both images can be obtained as the last column of the V matrix.

To have a better performance of the DLT we can add a RANSAC algorithm to estimate the homography with only points that are considered inliers and avoid noise. This is done by selecting 4 random keypoints from the set of detected ones and applying the homographies to the points x and x' to calculate the distance between the original keypoint and the transformed one from the other image with a distance function, which in our case was:

$$d^2 = \| [H^{-1} \cdot x'_i] - [x_i] \|^2 + \| [H \cdot x_i] - [x'_i] \|^2$$

If this value was below a threshold, the point was classified as an inlier, and was added to the list of points that were used when the RANSAC algorithm finished to re-estimate the H matrix.

3. Gold Standard

With the Gold Standard method, we just refine the previously calculated homography by minimizing the geometric error. This error is calculated with the euclidean distance between the detected keypoints and the transformed keypoints with the previous homography. Then refined homography is found as the solution of minimizing the geometric error with the Levenberg-Marquardt algorithm.

4. Results and comments

1) Homography estimation with the DLT algorithm

The different functions that had to be completed were done following the previously explained DLT algorithm with RANSAC and then the method was applied to perform an image mosaic of 4 different sets of 3 photos.

Comment the results in every of the four cases: hypothesize why it works or does not work

The sets of photos can be compared in two pairs, the castle and llanes images and the two aerial sets.

For the first pair, we can observe that the llanes mosaic (Fig. 1) represents successfully the panoramic view of the 3 images, and we can observe that there aren't any strange artifacts due to a non-exact estimated homography. On the other hand, for the castle mosaic (Fig. 2) we can appreciate artifacts in the middle of the panorama where the images are being united.

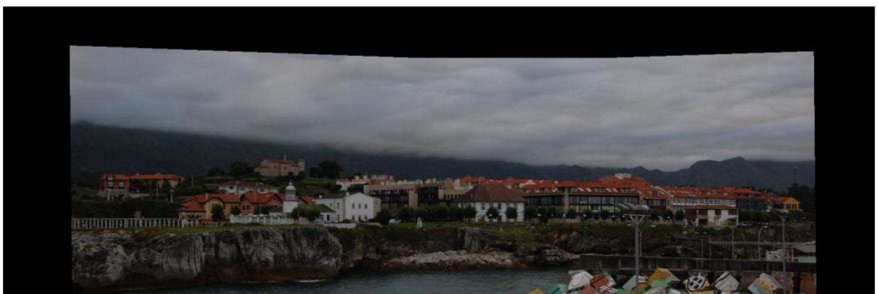


Figure 1: image mosaic with the llanes images

Figure 2: image mosaic with the castles images

With the second pair of sets we can also observe the same for the aerial set 13 (Fig. 3) the results are what we seek, but for aerial set 22 (Fig. 3) the results are even worse than in the castle images.

Figure 3: image mosaic with the aerial set 13

Figure 4: image mosaic with the aerial set 22

The reason for this behavior may be due to different factors. First, we can observe that the images used to obtain the different mosaics are more similar in the two successful sets. This allows the keypoint detector to provide more keypoints and therefore the probability of using more inliers to the homography calculation increases. Another factor can be that the camera position of the llanes and aerial 22 images is translated and few rotated in the vertical axis (yaw), less than in the case of the castle and the aerial 13 images.

If we observe the detected keypoints and the inliers used to calculate the homographies we can see how the quantity in the case of the non-distorted mosaics is greater than in the other two cases.

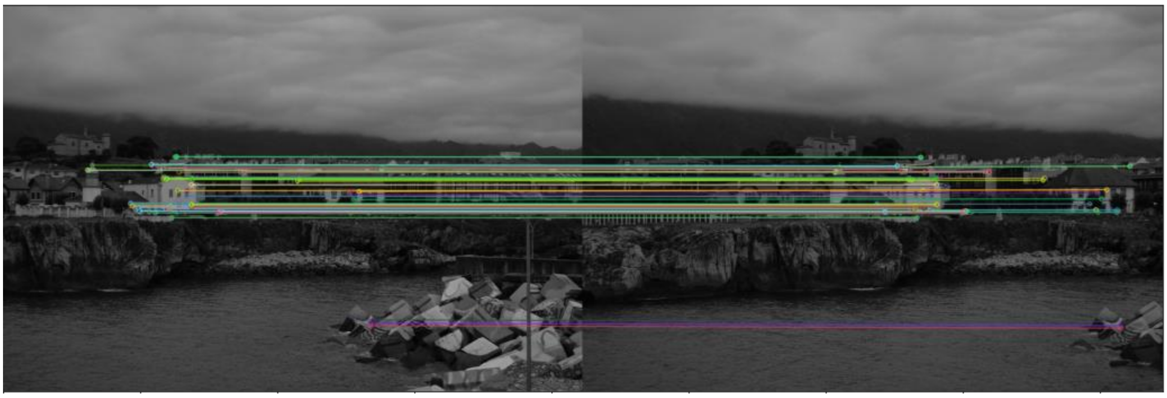


Figure 5: image 2 and 3 of the llanes set with the inliers used to compute the homography.

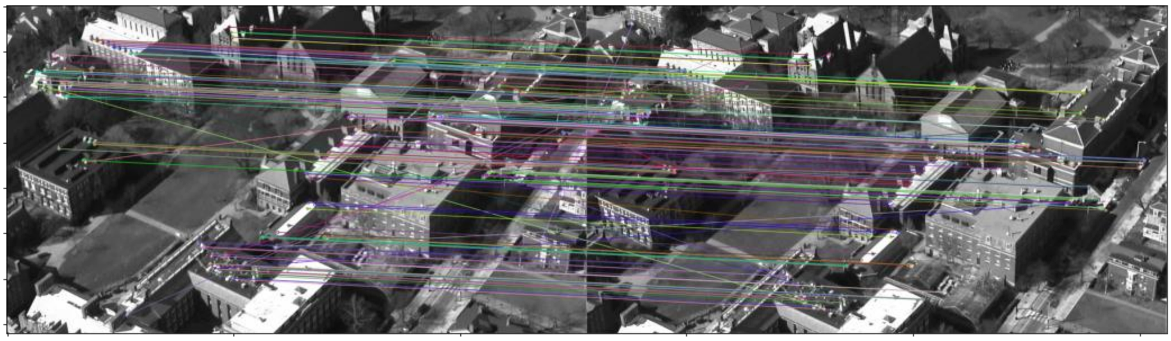


Figure 6: image 2 and 3 of the aerial 22 set with the inliers used to compute the homography

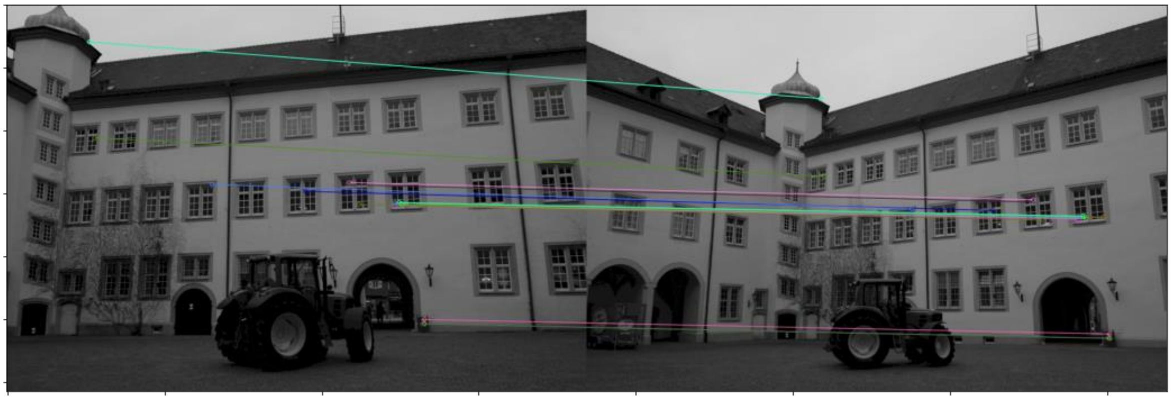


Figure 7: image 2 and 3 of the castle set with the inliers used to compute the homography

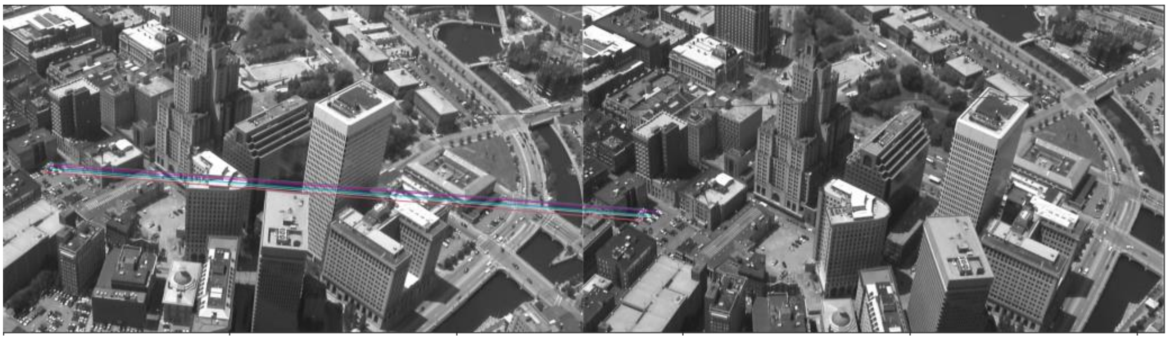


Figure 7: image 2 and 3 of the aerial 13 set with the inliers used to compute the homography

2) Refinement of the estimated homography with the Gold Standard algorithm

Unfortunately we were unable to properly code an implementation of the Gold Standard Algorithm in time due to some issues with the formatting of the input data for the Levenberg-Marquardt algorithm as well as also having issues retrieving the result data which contains the refined keypoints and homography estimations.

On our first implementation, we were able to feed the data to the algorithm, however we appreciated that the result data retrieved was not correct and thus had to find a new way to make the algorithm work. This new implementation was to be a working one, but we encountered some issues when processing the input data with a mismatched number of variables and points.

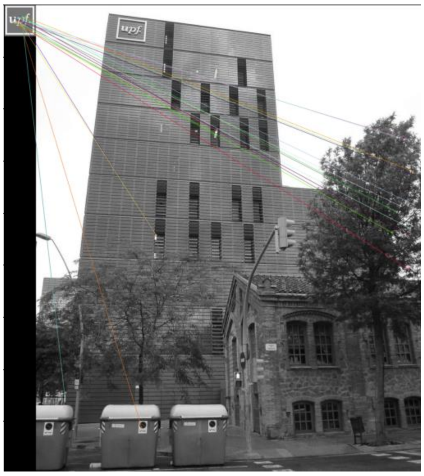
OPTIONAL TASKS

Application: Logo detection

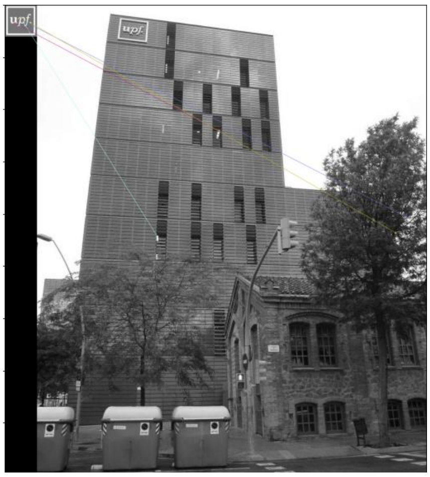
Detect the UPF logo in the two UPF images using the DLT algorithm. Interpret and comment the results

In the same way as in the mosaic images, we first find the keypoints between the images to then apply the RANSAC method and get only the inliers to compute the homography.

For the building image, we weren't unable to find a set of keypoints that were good enough to then apply the RANSAC algorithm and obtain a set of all inliers. Many of the keypoints associated with the logo image weren't part of the logo in the building, as you can see in the following image, and as well the keypoints returned by the RANSAC method weren't inliers.



Keypoints detected



Keypoints returned by the RANSAC method

The stand logo had a better behavior and the Robust DLT accomplished to find a set of keypoints that were inliers.

Keypoints detected

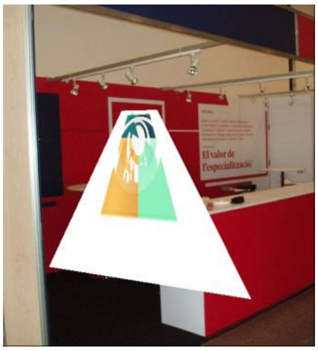
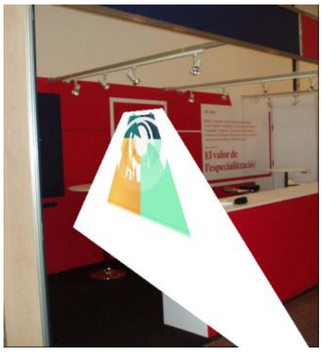
Inliers returned by the RANSAC method

To perform the logo detection, we increased the threshold in the ***keypoints_and_matches_2img()*** with respect to the used in the image mosaics, to have more keypoints considered as good matches function which permitted the RANSAC to have more points to find a set of inliers. In the case of the building photo, this didn't change the results, may be due to the fact the logo in the image is very little and the keypoints detects wrongly all the keypoints from the logo which in the case of the stand as it is bigger in the image this doesn't happen.

Application: Logo replacement

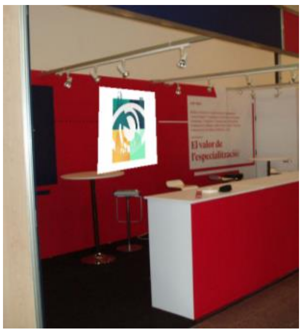
Replace the UPF logo by the master logo in one of the previous images using the DLT algorithm.

With the homography calculated with the inliers in the stand photo, we tried to replace the UPF logo with the master's one, but the results weren't satisfactory. Trying to adjust the used inlier to estimate the homography we tried to change the threshold used at the RANSAC function to see if this affected the final result, but this wasn't the case and these are the images that we obtained.

The results var each the time the homography is estimated as the inliers change, and the threshold value of the RANSAC method doesn't impact much to the final result.

As we were surprised that the logo replacement failed as the DLT algorithm worked before we tried to compute the homography with the OpenCV function and then use it to replace the logo. The obtained result was what we expected.



Although the parameters of the RANSAC method used were the same, the OpenCV implementation uses the Levenberg-Marquardt method to reduce the re-projection error, which permits the logo to be replaced successfully.

5. Conclusions

The DLT algorithm has been proven to be an efficient method to perform the homography estimation with enough points correspondences to make the calculations and perform an image mosaic or logo detection. When the point correspondences are scarce, the obtained results are not satisfactory as the noise present in the points correspondences affect too much to the estimation.