# VMMC Final project

**Zhanbota Bissaliyeva**

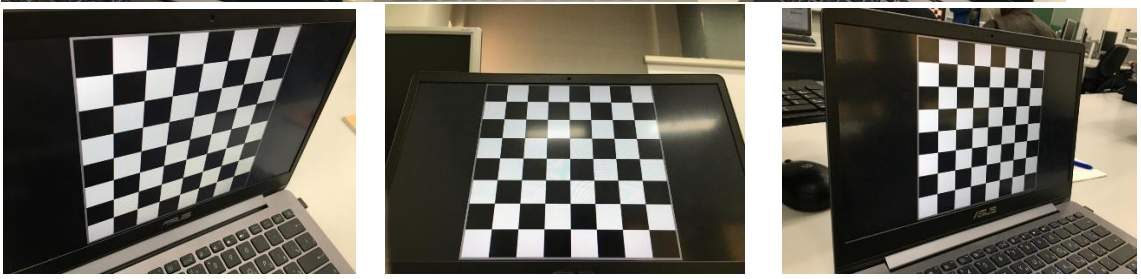
The final aim of the proposed project is to obtain a 3D reconstruction of a 3D scene/object. For this purpose, codes implemented during each unit were used to select and calibrate a camera, to select an adequate number of views from the chosen scene, to extract and match feature points between views, to compute the fundamental matrix between views, to obtain a 3D points cloud reconstruction, and finally to represent object geometric elements over this points cloud.

## Section 1: Intrinsic parameters of a camera

Using code developed during first unit we calibrate a camera, which we use throughout entire project, using two checkerboards. Necessary data:

- Size, in millimeters, of the checkerboards in my screen.
- The set of images of the screen checkerboards used for calibration.

  **Bigger checkerboard(1080p):** 160 mm, each cell 20mm



- The resolution of these captured images (in pixels).
  - **For both:** 1024x768 pixel
- The obtained matrix of internal parameters, **A** .

$$A_{1080*1080} = \begin{bmatrix} 888.891 & 2.5981 & 503.4565 \\ 0 & 882.1493 & 383.7843 \\ 0 & 0 & 1 \end{bmatrix}$$

- Analysis of additional aspects:

To make the analysis clear here we need to interpret obtained A matrix.

$$A = \begin{bmatrix} f * k_u & -f * k_u * cos\theta & u_0 & 0 \\ 0 & f * k_v / sin\theta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{ or } A = \begin{bmatrix} f_u & s & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where:

f – focal length (in pixels)

$k_u, k_v$ – size of pixel

$u_0,\ v_0$ – principal point coordinates, projection of the camera center in the image plane

$\theta$ – angle between the image axes

s – pixel skew coefficient

- Are the pixels of your camera square?

  For pixels to be square $f_u$ should be equal to $f_v$. As we can see from A matrix pixels of my camera are not square, moreover it is not even rectangular, since for pixel to be rectangular s should be equal 0.

- Which is the degree of coincidence between the principal point and the center of the image plane?

  The principal point coordinates $u_0, v_0$ are equal 503.46 and 383.78 respectively. Considering that resolution of the image is 1024x768, the center of the image is (512,384), the coincidence(C) between the principal point and center of image plane can be calculated as follows:

$$C = \frac{98.33 + 99.94}{2} = 99.14\%$$

- Are the axes of the image plane orthogonal?

  Considering information drawn from matrix A we can calculate angle as follows:

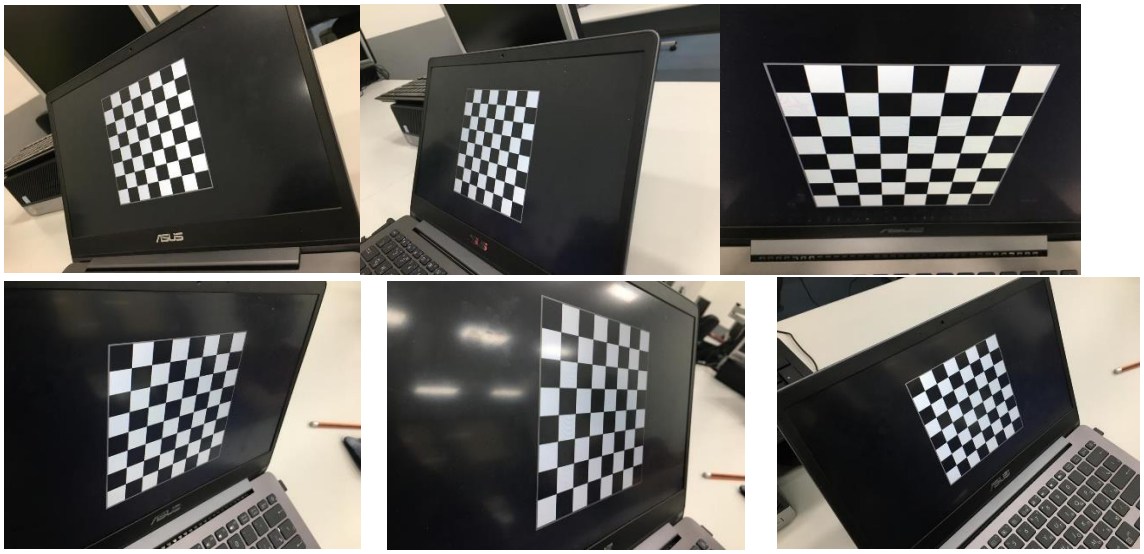
$$\theta = \text{acos}\left(-\frac{f * k_u * cos\theta}{f * k_u}\right) = 89.83°$$

  Considering obtained angle which is almost $\frac{\pi}{2}$, we can say that axes of the image plane are orthogonal.

2. Repeat the process for smaller checkerboard:

- Size, in millimeters, of the checkerboards in my screen.
- The set of images of the screen checkerboards used for calibration.

**Smaller checkerboard(720p):** 110 mm, each cell 13.75mm



- The resolution of these captured images (in pixels).

  **For both:** 1024x768 pixel

- The obtained matrix of internal parameters, **A** .

$$A_{720*720} = \begin{bmatrix} 915.6476 & -18.9467 & 531.4779 \\ 0 & 885.7552 & 417.0804 \\ 0 & 0 & 1 \end{bmatrix}$$

- Analysis of additional aspects:
- Are the pixels of your camera square?

Same assumption as for the bigger checkerboard can be made.

- Which is the degree of coincidence between the principal point and the center of the image plane?

  The principal point coordinates $u_0, v_0$ are equal 531.48 and 417.08 respectively. Considering that resolution of the image is 1024x768, the center of the image is (512,384), the coincidence(C) between the principal point and center of image plane can be calculated as follows:

$$C = \frac{96.2 + 91.39}{2} = 93.8\%$$

- Are the axes of the image plane orthogonal?

  Considering information drawn from matrix A the angle is $\theta = 88{,}81°$

- Comments:

  Theoretically, the two A matrices we obtained should have been the same because they contain internal parameters of the same camera. However, as we can see, practically, they are different which possibly because the process of taking photos on the phone brings noise considering particularly shaking hands or touching screen shake.

# Section 2: Local matches between several views of an object.

In this section, we were asked to use the calibrated camera from Section 1 to capture several views of a scene. Then, for pairs of views, we needed to detect, describe and match feature points using several of the methods explained in class. And according to obtained results select a detector-descriptor couple and a pair of views according to qualitative and quantitative indicators.

1. **Object/Scene capture.**

   In suggestions we were told to use 3D objects with edges and sharp contours to obtain better matching, vary angles of capturing views and distance of camera. The scene captured is considered as having a reasonable distance and angle change. Although, the scene is quite simple the painting on the right turned out to be quite a challenge for section3.

   In order to provide more information three pairs of images were chosen. The difference is in the angle between images in a pair. Thus, first pair has a smallest angle, they were taken consecutively, second pair is with a gap of 1 image and the last pair with the biggest angle and a gap of 2 images.

1st pair 2nd pair 3rd pair

2. **Qualitative and quantitative evaluation.**

   After selection of pairs of views we need to extract and describe feature points for each view of the pair, and match points between the views. Following detector + descriptor combinations were tested: DoH + SIFT, SURF + SURF, KAZE + KAZE, SIFT + DSP-SIFT.

   In order to select the best pair of views and detector-descriptor combination for every selected pair of views and every detector + descriptor combination fundamental matrix was estimated. Moreover, for all pairs the quality of the estimated fundamental matrix was qualitatively evaluated using vgg_gui_F.m function and number of inliers correspondences.

*Table 1. Fundamental matrices for 4 combinations, pair1*

| SIFT + DSP-SIFT | DoH + SIFT |
|---|---|
| $\begin{bmatrix} -0.0 & 0.0 & -0.0047 \\ -0.0 & 0.0 & -0.0199 \\ 0.0043 & 0.0205 & 0.9996 \end{bmatrix}$ | $\begin{bmatrix} 0.0 & -0.0 & 0.0077 \\ 0.0 & -0.0 & -0.0124 \\ -0.0111 & 0.0125 & 0.9998 \end{bmatrix}$ |
| **KAZE + KAZE** | **SURF + SURF** |
| $\begin{bmatrix} 0.0 & -0.0 & -0.0024 \\ 0.0 & 0.0 & -0.0192 \\ 0.0016 & 0.0196 & 0.9996 \end{bmatrix}$ | $\begin{bmatrix} -0.0 & -0.0 & -0.0048 \\ -0.0 & 0.0 & -0.0201 \\ 0.0044 & 0.0205 & 0.9996 \end{bmatrix}$ |

*Table 2. Fundamental matrices for 4 combinations, pair2*

| SIFT + DSP-SIFT | DoH + SIFT |
|---|---|
| $\begin{bmatrix} 0.0 & -0.0 & 0.0192 \\ 0.0 & -0.0 & -0.0543 \\ -0.0252 & 0.056 & 0.9965 \end{bmatrix}$ | $\begin{bmatrix} 0.0 & -0.0 & 0.0148 \\ 0.0 & -0.0 & -0.0488 \\ -0.0198 & 0.0499 & 0.9973 \end{bmatrix}$ |
| **KAZE + KAZE** | **SURF + SURF** |
| $\begin{bmatrix} 0.0 & -0.0 & 0.0432 \\ 0.0 & 0.0 & -0.0642 \\ -0.0546 & 0.0662 & 0.9933 \end{bmatrix}$ | $\begin{bmatrix} 0.0 & -0.0 & 0.0155 \\ 0.0 & -0.0 & -0.0511 \\ -0.0208 & 0.0529 & 0.997 \end{bmatrix}$ |

*Table 3. Fundamental matrices for 4 combinations, pair3*

| SIFT + DSP-SIFT | DoH + SIFT |
|---|---|
| $\begin{bmatrix} 0.0 & 0.0 & 0.0032 \\ 0.0 & 0.0 & 0.0534 \\ -0.0 & -0.0672 & 0.9963 \end{bmatrix}$ | Not enough matching points |
| **KAZE + KAZE** | **SURF + SURF** |
| $\begin{bmatrix} -0.0 & 0.0 & -0.0 \\ 0.0 & 0.0 & 0.0327 \\ 0.0037 & -0.0431 & 0.9985 \end{bmatrix}$ | $\begin{bmatrix} -0.0 & 0.0 & -0.0058 \\ -0.0 & 0.0 & 0.0422 \\ 0.0121 & -0.0538 & 0.9976 \end{bmatrix}$ |

*Table 4. Quantitative evaluation using number of inliers*

| | № of the pair | Detector + descriptor combination | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | DoH+SIFT | | | SURF + SURF | | | KAZE + KAZE | | | SIFT + DSP-SIFT | | |
| Detected points, useful points, percentage | 1 | 265 | 133 | 50.19 | 727 | 364 | 50.07 | 3192 | 1596 | 50 | 1327 | 664 | 50.04 |
| | 2 | 215 | 108 | 50.23 | 566 | 283 | 50 | 2435 | 1218 | 50.02 | 971 | 486 | 50.05 |
| | 3 | - | - | - | 312 | 156 | 50 | 549 | 275 | 50.09 | 58 | 29 | 50 |

According to the visual analysis and information we see in the table 4 following conclusions were made:

1. With increasing difference between views, performance of the detector + descriptor combination was getting worse. Thus, for the first pair of images with slight rotation and distance change we have much more inliers than for other two for every detector + descriptor combination.
2. Analyzing feature points distribution we can easily tell the difference between the combinations.
   - Worst performance was demonstrated by combination of DoH detector and SIFT descriptor. Considering number of inliers we see that this combination didn't allow us to find enough matching points for 3rd pair of views, while for first two pairs inliers were found in relatively small amount.
   - SURF + SURF combination provides more detections and thus matching points in comparison with previous one. However, it is not enough to outperform other combinations.
   - KAZE + KAZE created the biggest amount of feature points and thus have more matching points. However, it is computationally expensive and results in false matchings, which can badly influence our following 3D reconstruction. Reason for false matching for this particular scene were mostly color similarities.

- The last but not least is SIFT + DSP-SIFT combination, which shows the best performance according to all the qualities, we consider. It has less detections and matchings comparing to KAZE combination, however it doesn't have any false matches.

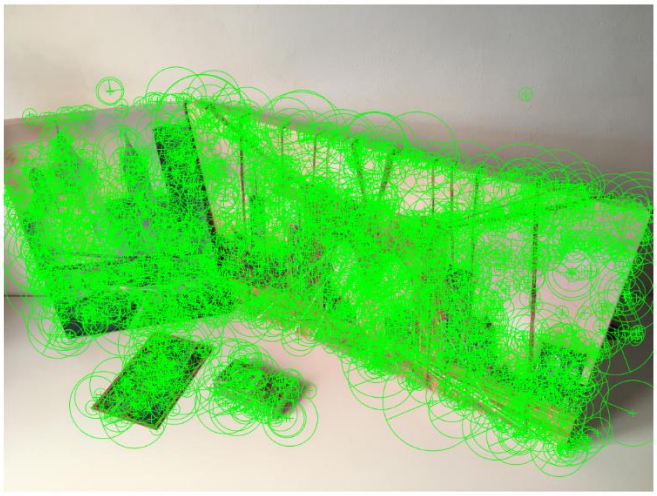
3. The qualitative evaluation also includes operation with vgg_gui_F.m function. For that, we can click and drag to move the point on one image and corresponding epipolar line on the other image. If it is crossing the same point then we evaluate the detector to work properly. We tried to choose points those regions where matching points were not obtained. However fundamental matrix performs quite well.

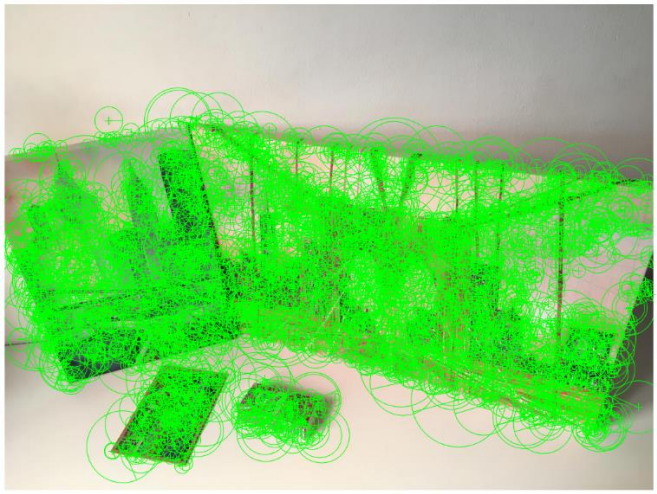
3. **Selection.**

According to the results obtained in the previous stage we chose as the best 1st pair and SURF and DSP-SIFT combination. Although quantitatively it is not the best, for the sake of the third section task we chose this combination. Below are provided

- The estimated Fundamental matrix.
- The pair of images with the correspondences overlaid on them.
- The warped images.
- The image with matching points
- Screen captures of the vgg_gui_F.m GUI

$$F = \begin{bmatrix} -0.0 & 0.0 & -0.0047 \\ -0.0 & 0.0 & -0.0199 \\ 0.0043 & 0.0205 & 0.9996 \end{bmatrix}$$











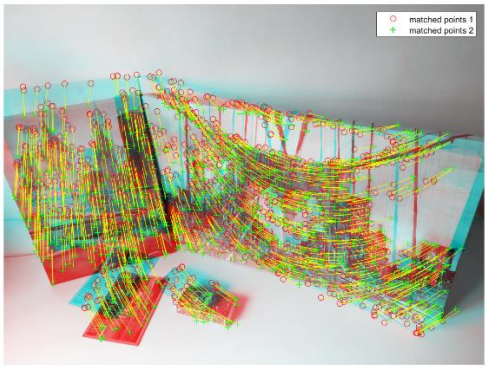




Image 1

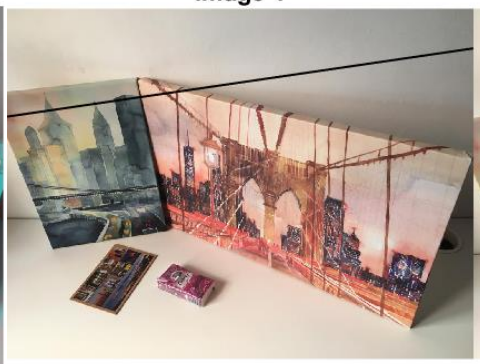


Image 2

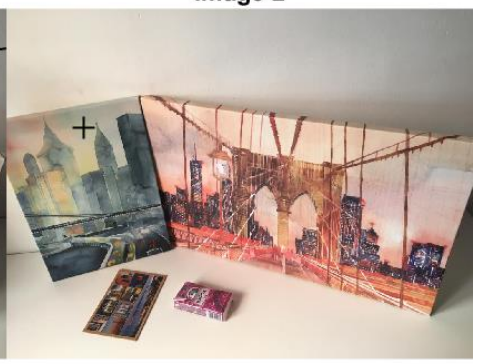


Image 1

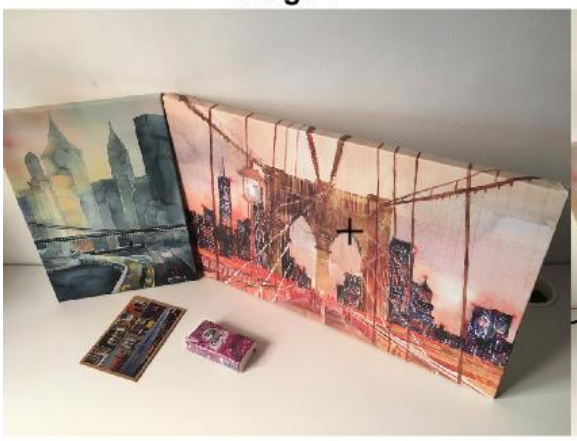


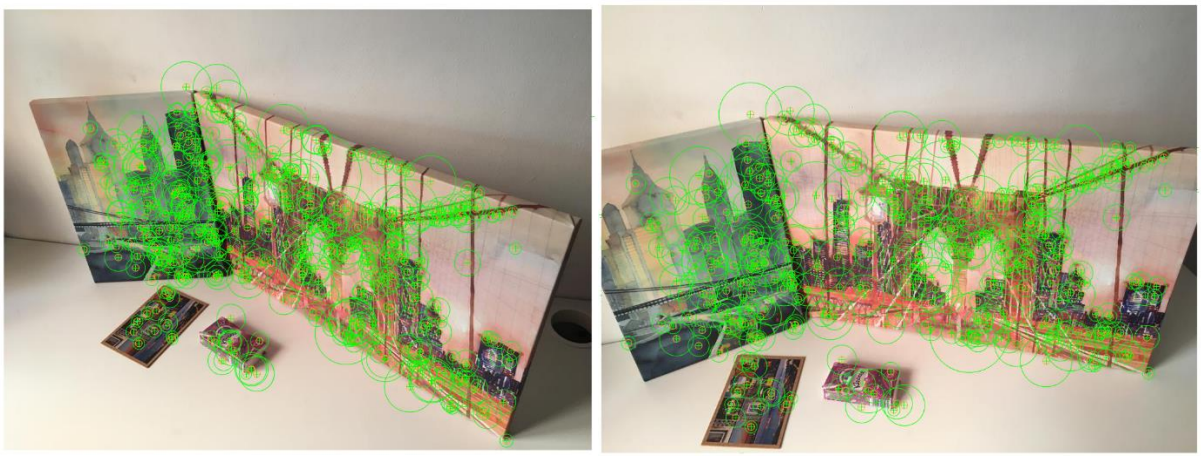
Image 2

# Section 3: 3D reconstruction and calibration

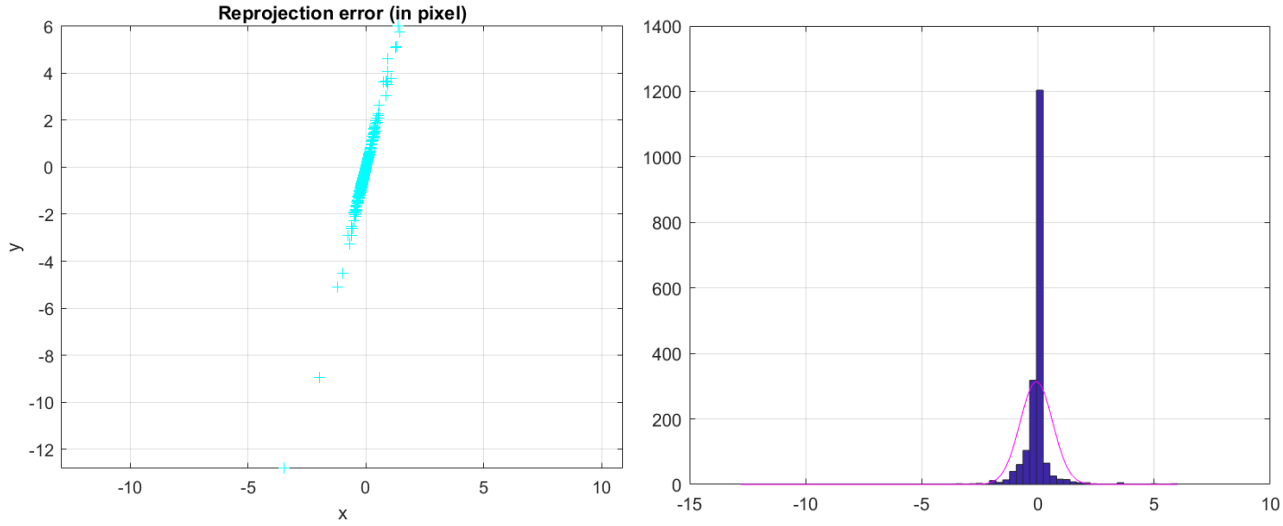
In this section, we are using the intrinsic parameters of the camera, obtained in Section 1 and the feature point matches between images, obtained in Section 2, to obtain a 3D reconstruction of the scene.

1. The images that have been used for the N-view point matching, indicating the detected interest points in each of them. For 3D points cloud reconstruction initially 8 images were used. However, because of the big angle between some of the images the size of dataset was reduced to 4.

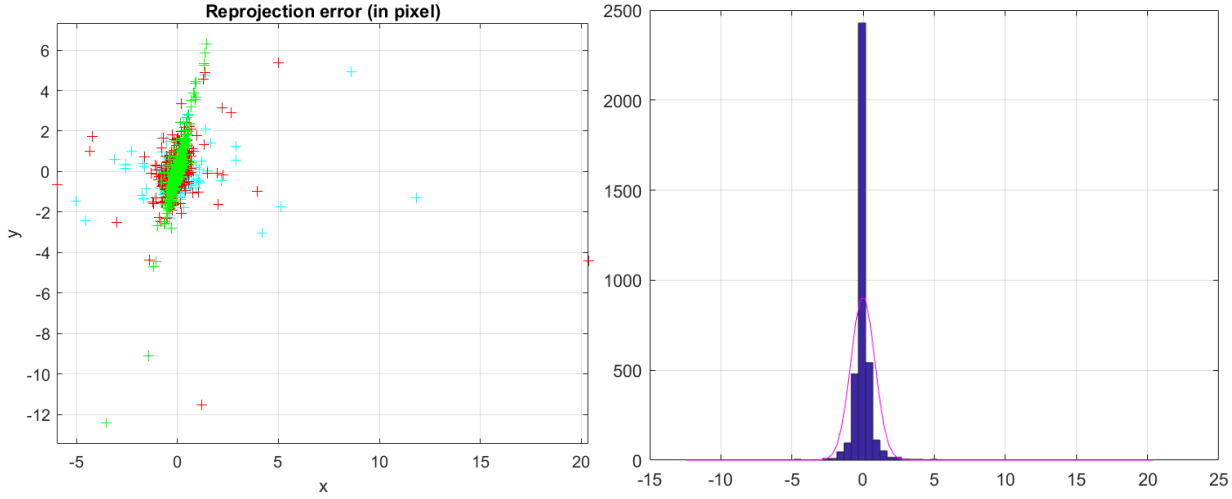
2. Next, we need to compute the Fundamental matrix and an initial projective reconstruction from 2 of the cameras.
    a. Images that have been used for the estimation of the Fundamental matrix. For initial projective reconstruction were used first and the last images of the sequence.



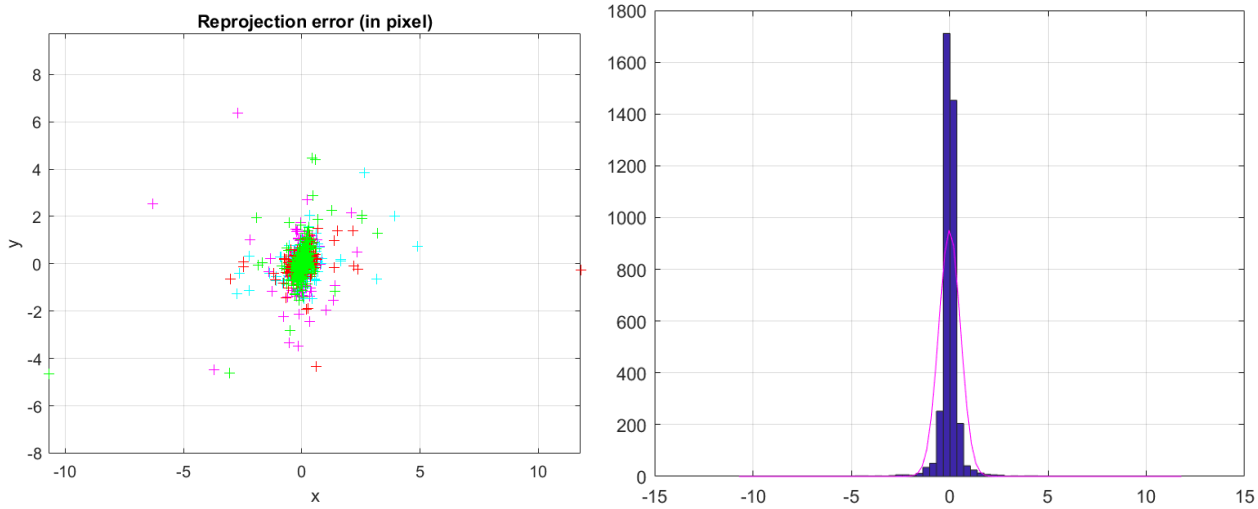
b. The mean re-projection error and the reprojection error histogram.




3. Then we should improve this initial reconstruction by means of a Projective Bundle Adjustment, using a higher number (all) of images.
   a. The mean re-projection error and the reprojection error histogram at two points:
      (i) After the resectioning step




      (ii) After the Projective Bundle Adjustment step.




   b. Comments on the justification of the different re-projection error values in 2.b and the two steps of 3.a.

First, we are using 8 points algorithm for reconstruction from only two images (in my case first and last images), the reprojection error is calculated from the 2D coordinates on image planes and the reprojected 2D coordinates. Next step is to use the resectioning, which is an initialization to obtain the projection matrices from higher number of cameras (in our case images) and homogeneous coordinates of 3D points. Thus, reprojection error we recalculate is higher. However, during bundle adjustment step we update projection matrices of all the cameras and thus obtain lower error.

4. Now we need to obtain a Euclidean reconstruction of the scene.
    a. Illustrative results of a 3D point cloud reconstruction. The only proper reconstruction out of all 4 solutions was illustrated in solution2. As we can see, two big paintings are reconstructed perfectly fine; however, reconstruction of smaller objects is quite poor. Solution1 showed the same results, but with cameras being behind the scene.