V.P. - Planar homographies and images mosaicing

Yannick PORTO

***Abstract*—The goal of this practical session is to estimate planar homographies between different views of a scene and to automatically stitch the images into a mosaic or panorama.**

I. Introduction

A panorama is obtained by stitching multiple images of a common scene taking from different viewpoint. To do so, the images need to be transformed according to a reference image. This transformation is performed thanks to a homographie computed with matched features between images and the reference one. We will explain here how to compute this homographie and show how is built the mosaic.

II. The set of images

We start with a simple set of images representing the Keble College in Oxford. The reference frame is the center one, because here the edges of the building are horizontals and all the other images can be stitch to this one.

Fig. 1. Left image

Fig. 2. Middle image

Fig. 3. Right image

III. Detecting and Matching local features

Some local features have to be extracted for comparing the images themselves. Instead of comparing all pixels value of the images, keypoints are a faster way to compute some matching points.
Different keypoints detector can be used, for instance Harris, SURF and FAST are correct for our application.

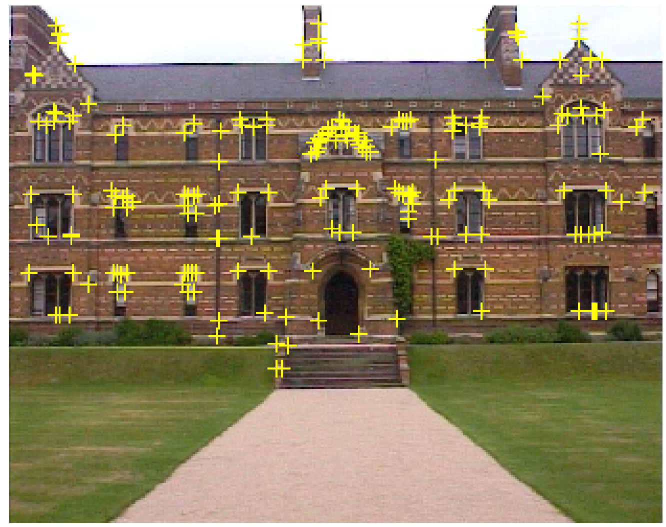

Fig. 4. Middle image with 200 detected Harris corners

Once these keypoints are detected, we need to extract the features from them.

One type of features, for harris corners, can be $21 * 21$ grayscale patches extracted from the image blurred by a Gaussian. So for every keypoints, a feature vector of $21 * 21$ is given.

The matching procedure consist in comparing the sum difference of pixels values (the euclidean distance) between each patch and see which pairs have the smallest one. This process is more robust if we apply a D.Lowe's ratio test so that less incorrect matches are kept. We therefore discard the points which have a ratio $\frac{1-NN}{2-NN}$ above 0.8.

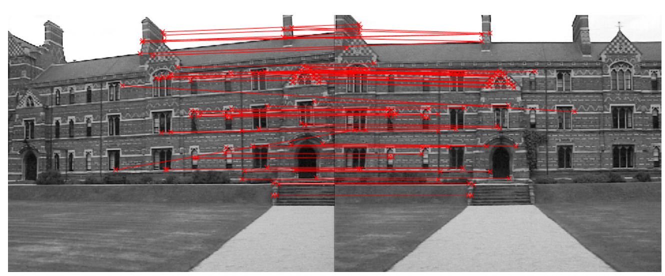

Fig. 5. Matching features between the left image and the reference (visualization to see if matches are correct

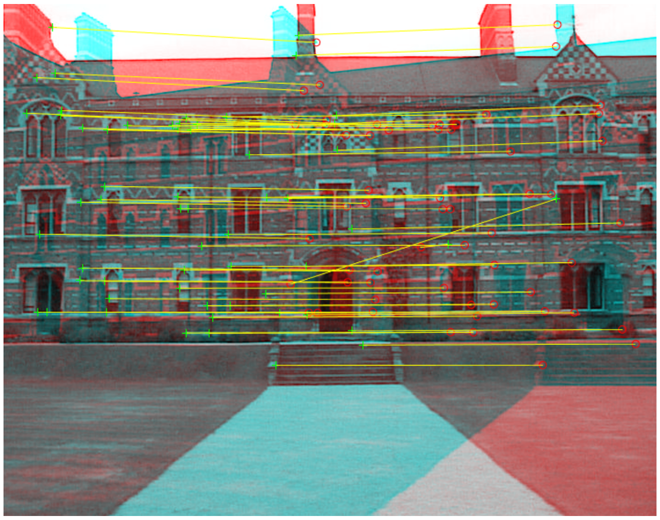

Fig. 6. Matching features between the left image and the reference (other visualization to see how the image is displaced)

IV. RANSAC HOMOGRAPHIE ESTIMATION

When the list of matches is obtained, the transformation from the first keypoints to the reference ones have to be found.

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{pmatrix} \tag{1}$$

As we need a homography, only four points can be used in both images. Following the RANSAC method, four random points and their correspondences can be picked into the list of matches and inserted in the equation. Equations 2, 3, 4 show how to compute the unknowns of the homography matrix with these points.

$$x = \frac{x_i}{z_i} = \frac{h_{11} * X_i + h_{12} * Y_i + h_{13}}{h_{31} * X_i + h_{32} * Y_i + 1} \tag{2}$$

$$\Leftrightarrow \begin{cases} h_{31} * x_i * X_i + h_{32} * x_i * Y_i + x_i - h_{11} * X_i - \\ \quad h_{12} * Y_i - h_{13} = 0 \\ h_{31} * y_i * X_i + h_{32} * y_i * Y_i + y_i - h_{21} * X_i - \\ \quad h_{22} * Y_i - h_{23} = 0 \end{cases} \tag{3}$$

$$\begin{pmatrix} & & & & & \cdots & & & \\ -X_i & -Y_i & -1 & 0 & 0 & 0 & x_i{*}X_i & x_i{*}Y_i & x_i \\ 0 & 0 & 0 & -X_i & -Y_i & -1 & y_i{*}Xi & y_i{*}Y_i & y_i \\ & & & & & \cdots & & & \end{pmatrix} * \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ 1 \end{bmatrix} = 0 \tag{4}$$

The least square solutions of the last equation gives the unknowns of the homography matrix.

Since the homography is found, the transformation can be applied on all the keypoints. These transformed points are compared with their previous matches so that we can check how far they are from them. By setting a maximum deviation, we label the transformed keypoints with a distance under this threshold as inliers.

The homography is then computed again as long as a sufficient number of inliers or a certain number of iterations is not reached. We found in our experiment that a maximum deviation of 2 pixels, a threshold of 50 inliers and a maximum of 100 iterations lead to the best results with the previous configurations.

V. WARPING THE IMAGES AND BUILDING THE MOSAIC

A homography is computed between all the images to stitch. Here two homographies have been computed : one between the left image and the middle (reference) image, and one between the right image and the middle image.

The size of the panorama is set to the maximum size it can have, so that no pixel can be found outside the frame. Therefore, we warp the all the images one by one into a frame of this size.

Fig. 7. Warped left image into the final frame

As the multiple pieces will overlap, we have to choose a criterion to select the pixels to keep. We take simply here the maximum value between two overlapping pixels. You can see the result in figure 8.

Fig. 8. Obtained mosaic

VI. CONCLUSION

In the RANSAC method, the parameters are really important to get a correct result at the end. Tuning the maximum deviation for labeling inliers involved a lot in the homography precision and the maximum number of inliers and iterations allows to play with the speed of the process. The D.Lowe threshold ratio is as well a big criterion of success. Changing this threshold leads some time to terrible results.

Building and writing all the steps of the mosaic construction is really helpful to understand some notions of computer vision. It allows us to see also what are the benefit of some detectors and descriptors compared to some others. We tried to built the mosaic with other kind of keypoints such as SURF and FAST under the matlab built-in functions which are really simple tool for building mosaics. So we designed a gui where you can enter images, select the type of detector you want and built the mosaic from these parameters. Here is a screenshot of what the gui looks like.

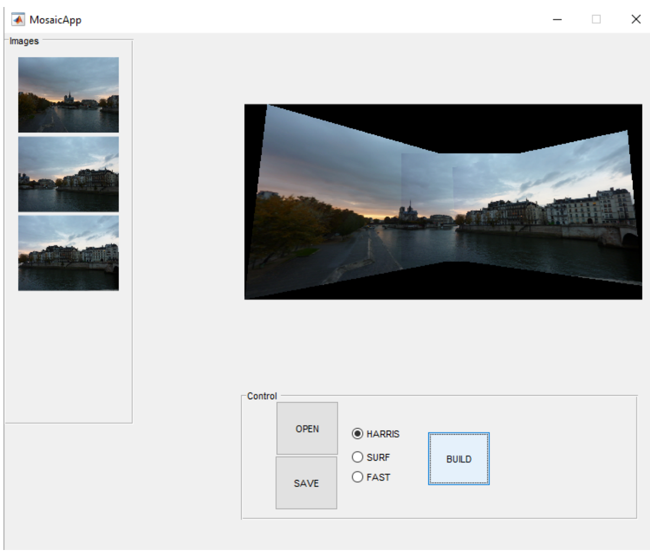

Fig. 9. GUI with panorama built from images of Paris