
PANORAMA CONSTRUCTION

Vincent Matthys

1 Objective

This program allows the user to reconstruct a scene from two images, when the two images comes from a static scene with 2 different views, which are overlapping over a certain area, in the following particular conditions :

- the observed scene is flat
- the observed scene is not flat, but the two views differs only from a rotation around the optical center and change of internal parameters of camera.

The user is asked to select sequentially matching points of both images. Then the program builds the panorama, and saves it in the folder of current program execution. The program ends with any user click after the construction of the panorama.

2 Material & Method

A single C++ file, named *Panorama.cpp* loads and displays two images, which can be passed in arguments to the program named *Panorama*. The coordinates of matching points ((x_i, y_i) matching for (x'_i, y'_i)) are used to reconstruct the homography between the two images, solving the following linear system :

$$\begin{bmatrix} \vdots & \vdots \\ x_i & y_i & 1 & 0 & 0 & 0 & -x'_i x_i & -x'_i y_i \\ 0 & 0 & 0 & x_i & y_i & 1 & -y'_i x_i & -y'_i y_i \\ \vdots & \vdots \end{bmatrix} \cdot \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} \vdots \\ x'_i \\ y'_i \\ \vdots \end{bmatrix}$$

where the h_{ij} are the homography values, for h_{33} arbitrary set to 1, without loss of generality.

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

To successfully finds H, the system requires at least 4 correspondences, each one giving two equations. If there is more than 4 correspondences, the program solves the system using least

squares programming. A sanity check is introduced and allows user the homography found. Every line should be $(0, 0, 0)$ vector. In reality, it is only close to the null vector, because, in particular, of errors during selection of points and the intrinsic noise of the camera.

To build the panorama, the program uses the pull method, which requires to inverse the homography, and queries the preimage by H , which is the image by H , of each pixel in the final image. If it lies in the first image, then the pixel is pulled in the final image. In the overlapping area, the program simply averages the value of the two images.

3 Results

We present here 3 use cases obtained with the default images :

1. first view, left of the scene. Size : 755 x 499
2. second view, right of the scene. Size : 755 x 499

a Wrong use case

a.1 Not enough correspondences

When the user doesn't give enough correspondences, the sanity check doesn't display, since the linear system can't be solved. Moreover, the homography is given, by default, as the identity matrix. Hence the panorama constructed in figure 1 is only the average of both image, as if they were taken with the same view.

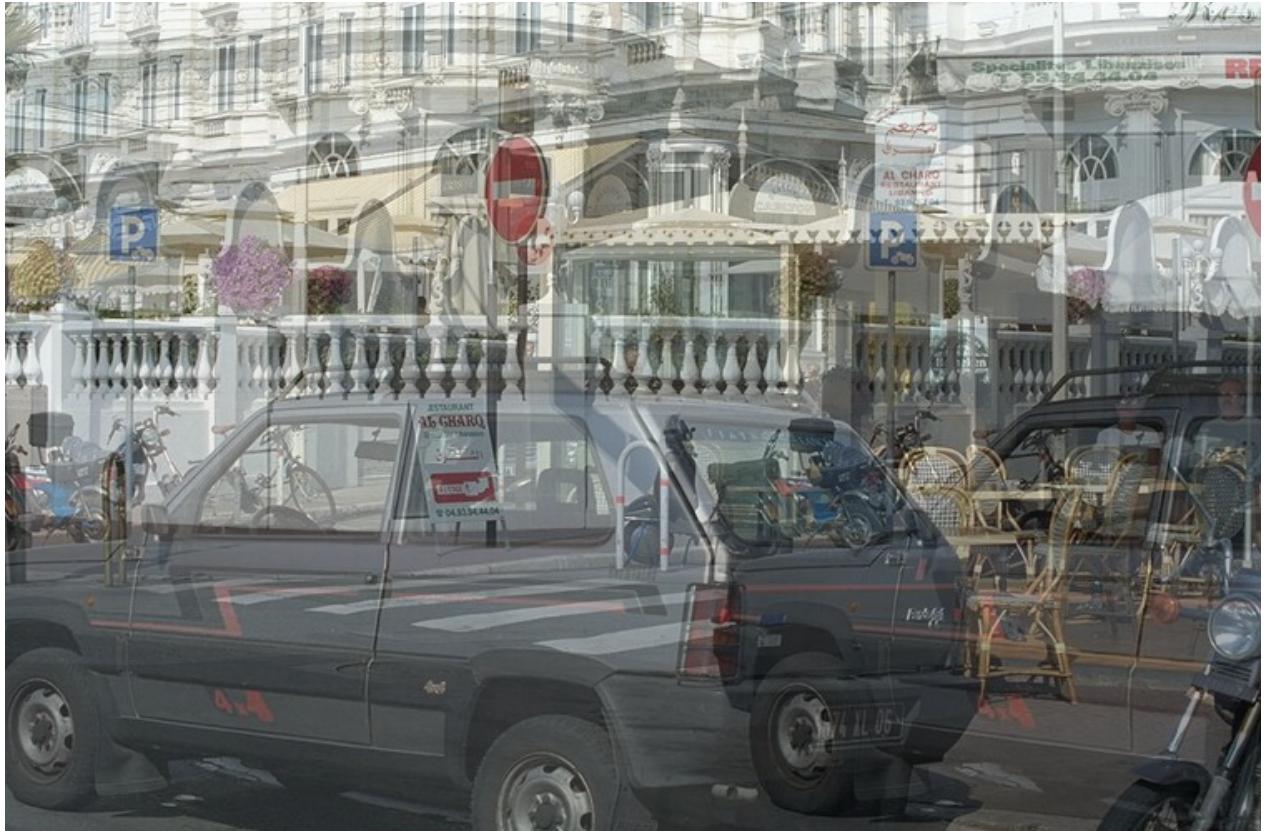


FIGURE 1 – Panorama construction with not enough correspondences. Size : 755 x 499

a.2 Wrong correspondences

When the user doesn't take proper correspondances, the homography found relies on wrong correspondances, hence a obvious wrong panorama construction, with geometric deformations. In the case presented in figure 2, we can see those deformations and the homography found is the following :

$$H = \begin{bmatrix} 0.792 & -0.281 & -279 \\ -0.040 & 0.250 & 100 \\ 0.00 & 0.00 & 1 \end{bmatrix}$$



FIGURE 2 – Panorama construction with obvious wrong correspondences. A frame has been added to see the image size of the panorama : 5270 x 3565

b With 4 correspondences

In figure 3, a minimum number of 4 correspondences allows to find a proper homography between the two views. The overlapping area still remains a little blurred, and the panorama construction highlights is not rectangular, which shows that the homography found, given below is not perfect. Moreover, we can notice some aliasing on vertical lines on the pulled image.

$$H = \begin{bmatrix} 1.02 & 7.36e-5 & -468 \\ 0.0120 & 1.02 & -8 \\ 0.00 & 0.00 & 1 \end{bmatrix}$$



FIGURE 3 – Top left : first view of the scene. Top right : second view of the scene. Top : circles highlight the correspondances. Bottom : panorama construction with the given correspondances. Size : 1254 x 524

c With more than 4 correspondances

In figure 4, 10 correspondences allow to find an almost perfect homography between the two views. The overlapping area is nearly not blurred, and the panorama format is almost rectangular, which corresponds to a rotation of the view between the optical center.

d With 4 correspondances



FIGURE 4 – Top left : first view of the scene. Top right : second view of the scene. Top : circles highlight the correspondances. Bottom : panorama construction with the given correspondances. Size : 1218 x 503