**Lucas Versini**

**Practical session n°1**
**October 20, 2024**

# 3D Computer Vision
**Master MVA**

**Practical session n°1**

## 1 Objective

The objective is to combine two images of the same scene into a single panoramic image.
In order to do this, the user needs to click on at least 4 points $\mathbf{x}_i$ in the first image, and on the corresponding points $\mathbf{x'}_i$ in the second one.
This allows us to compute a homography $H$ which maps the first image to the second one, and to "stitch" the images together, see Figure 1 for an example.

## 2 Technical explanations

We would like to find a homography $H$ such that for each $i \in \{1, \dots, N\}$, $H\mathbf{x}_i \approx \lambda_i \mathbf{x'}_i (*)$.

We construct (as detailed in the first lecture) a matrix $A$ with $2n$ rows (where $n \geq 4$ is the number of corresponding points) and 9 columns based on the points coordinates. The coefficients of $H$ are stacked into a 9-dimensional vector $h$.
Then $(*)$ becomes $Ah = 0$. To solve it, we can fix $h_9 = H_{3,3} = 1$, as $H$ is determined up to a scale factor. Thien the equation becomes $A_{:,1:8}h_{1:8} = -A_{:,9}$.

In most cases, the equation does not have a solution (except when $n = 4$), so we instead solve $\min_h \|A_{:,1:8}h_{1:8} + A_{:,9}\|_2^2$.
After solving for $h$, $H$ is recovered.

Then, we start with the second image, expand it (by adding white all around) if necessary so that the image by $H$ of the first image fits in the frame.
Then, the color of a pixel is the average of the color of this pixel in the second image, and the color of the pixel in the first image (obtained using $H^{-1}$). So here, we pull pixels from the first image.
Note that in some parts of the image, only the color from one of the images will be used.

## 3 Results

### 3.1 Basic results

The results obtained for the images given in the exercise can be seen in Figure 1.
The result is satisfying, although some inconsistencies in color blending are noticeable in the center of the panorama, where colors from the two images are averaged.
A slight blur is also present (see for instance the Parking sign), which also results from the averaging process.

### 3.2 Impact of the number of points

We can also look at the result when using more than 4 pairs of points in Figure 2.

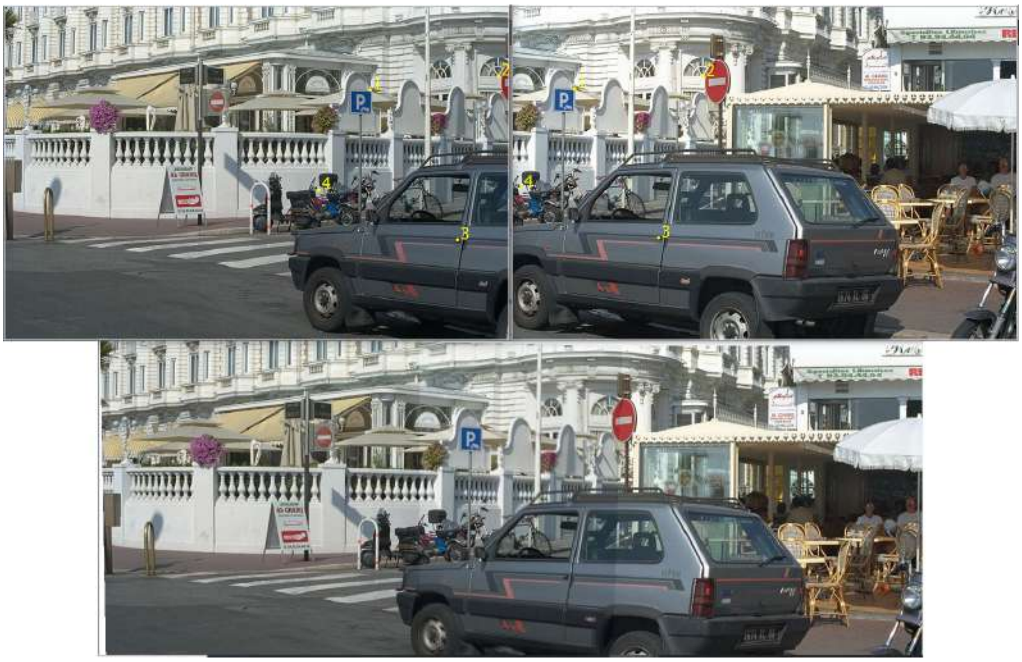

Figure 1: Panorama for the given images with 4 pairs of points

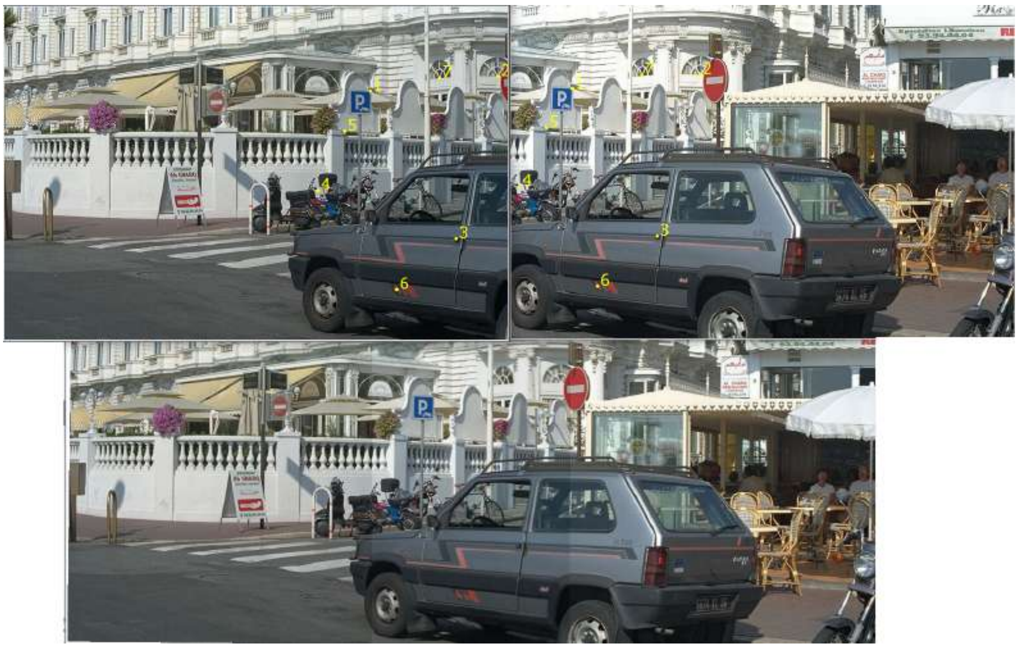

Figure 2: Panorama for the given images with 7 pairs of points

We see in this case that the homography seems to have distorded less than before the first image. It seems that using more points increases the robustness of the computed homography. This can be understood by the fact that for each pair of points, a small error may be committed by the user (who does not click exactly on the right pixel). Using more points is then preferable, as it allows to limitate the effect of each error.
Of course, using more points implies that the system to solve is more complex. However in this case (that is, with a small number of points) the difference in computation time is unnoticeable.

# 4 Limitations

## 4.1 Existence of a homography

This technique relies on the assumption that a homography exists between the two images, which is true when the images are captured from the same position with only a rotation between them.
However, when the transformation includes both rotation and translation, the panorama technique fails, resulting in a distorted output, like in Figure 3, where the result is obviously not what was expected.

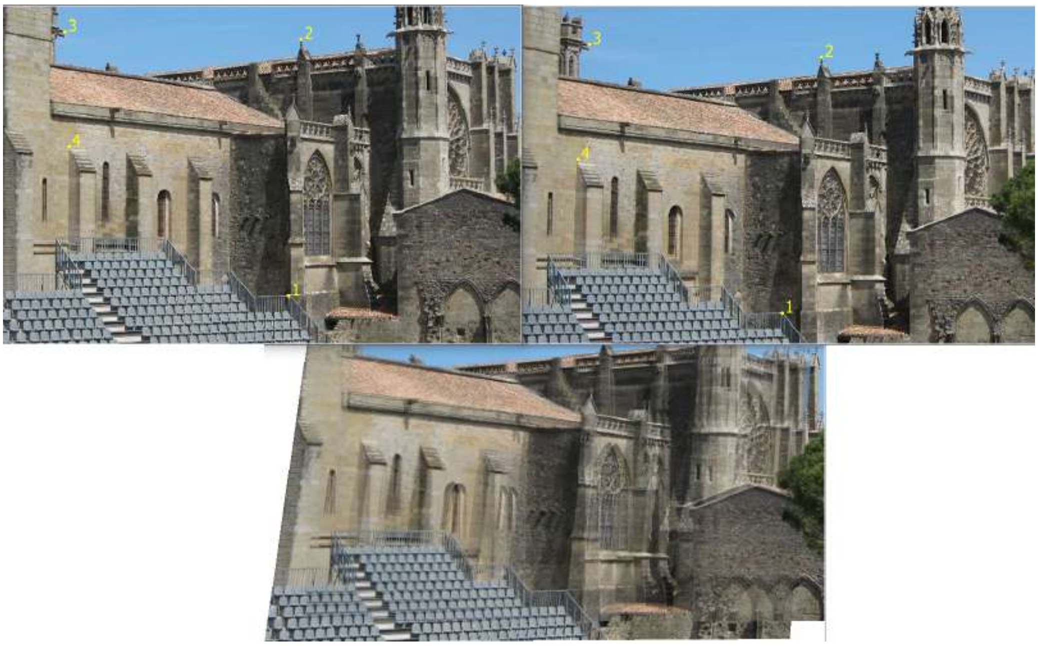

Figure 3: Panorama for for non-homographically related images

## 4.2 Distribution of the points

If the points used to compute the homography are clustered in a small region of the image, the homography may be accurate in that area but fail to generalize well to the rest of the image. Ideally, points should be spread across different regions of the images to better constrain the transformation.
This is illustrated in Figure 4, where we used four pairs of keypoints corresponding to the four corners of the parking sign. In the resulting image, this sign is perfectly clear, but most of the image is not.

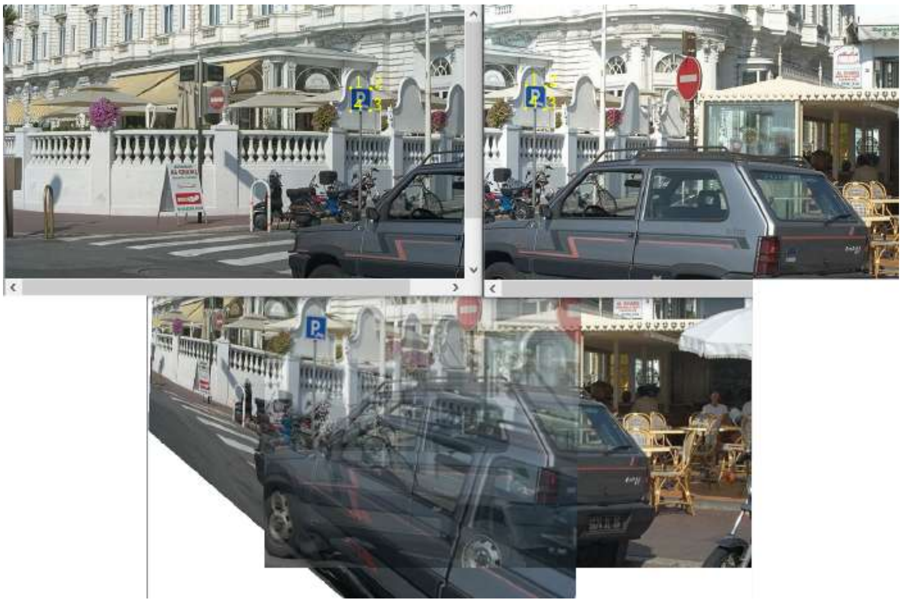

Figure 4: Panorama for clustered points