

Computer VISION

Report TP1 : Panorama

Shubhamkumar Patel

The goal of this implementation is to reconstruct a panorama from two images that were captured in different conditions and perspectives. The way to do this is to first identify key points that are similar in both images. In fact we need at least 4 points from each image in order to recover all the information needed to apply transformation and reconstruct a panorama image.

We will prompt the user to identify and select 4 similar key points from the two images which will give us 8 coordinates for the 8 pixels selected. Here is an example of this process of key points identification and selection.



Figures with key points identified in red in the left image and in green in the right one.

We now have to calculate the Homography Matrix H . To do so we will take all 14 selected points and use the algorithm to calculate A_i until we reach the optimal matrix H . Once we have the matrix we will use the method by pulling pixels from the original image by interpolation to apply the homography to our image.

By doing so we get the final image of the Panorama as follows :



The method by pulling pixels from the original image worked better because there are no empty pixels represented as dark or bright lines in the zones where the two images overlap. The method with pushing pixels to the transformed image had these issues so I chose to stick with the method with pulling pixels by interpolation.

Conclusion

We succeeded in getting the homographic transformation between the two images that allowed us to reconstruct a panoramic image from the two input images. We still need to think about the transition zone that appears on the scene and we can imagine multiple scenarios for solving it. One way to deal with the overlapping zone is by pulling the pixels from the original image and using an average of pixels from both images in order to get a better result but that would still differentiate the overlapping zone from the rest. There is also the problem of artifacts that may appear in the end result if we didn't choose enough points in the first step.