ECE661: Homework 3

Fall 2018
Deadline : September 13, 2018 , 1:30 pm

Turn in typed solutions via Blackboard. Additional instructions can be found at [I]

1 Introduction

The goal of this homework is to eliminate the projective and the affine distortions in the two images in Figure 1. You basically have the following three methods at your disposal:

1. Using point-to-point correspondences (in exactly the same manner as you did in the previous homework) to find a homography between two images, assuming that one represents the original scene and the other its photograph with projective and affine distortion and then reversing the homography to eliminate the distortion in the latter image.

2. You use what is known as the 2-Step method in which you first remove the projective distortion using the Vanishing Line method discussed in Lecture 4. Subsequently, you remove the affine distortion by using the $Cos\theta$ expression with θ equal to 90°. Note that you *must* first remove the projective distortion before you can remove the affine distortion with the $Cos\theta$ based method.

3. You use what is known as the 1-Step method that gets rid of both the projective and the affine distortion in one go.

Obviously, the first method, the one based on point-to-point correspondences, is most straightforward. However, it often requires a large number of correspondences to give a numerically stable solution for estimating the homography.

The main idea in the second and the third methods listed above is to use some "high-level" knowledge regarding the geometry of imaging for doing the same job but with fewer correspondences.

Regarding the implementation of the Vanishing Line (VL) method for removing the projective distortion, note that you will have to estimate the VL in the image plane. For that you'll need to click on the pixels that fall on lines that are parallel in the original scene. Taking the cross-product of two such pixels on any one line in the image will give you the homogeneous representation of that line. Taking the cross-product of 3-vectors for two different lines (which are parallel in the original scene) will give you the homogeneous representation for the Vanishing Point for those two lines. And then taking the cross-product of two such vanishing points for two different pairs of parallel lines will give you the VL you need for getting rid of the projective distortion.

Regarding the 1-Step method that gets rid of both the projective and the affine distortions in one go, as discussed in Lecture 5, this requires estimating $C_\infty^{*'}$, which is the image of the Dual Degenerate Conic C_∞^* in your image plane. As you know, the latter is given by

$$C_\infty^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

We now represent its projection in the image plane by

$$C_\infty^{*'} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

Assuming that the lines l' and m' are the images of two perpendicular lines l and m in the original scene, we estimate the parameters a, b, c, d, e, f from a set of simultaneous equations that look like:

$$l'^T C_\infty^{*'} m' = 0$$

You will need at least five such equations.

Once you have estimated the image $C_\infty^{*'}$ of the Dual Degenerate Conic C_∞^*, it can be shown that the homography that gets rid of both the projective and the affine distortion is a solution of the following SVD decomposition of $C_\infty^{*'}$:

$$C_\infty^{*'} = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} U^T$$

What this basically says is that when you carry out an SVD of $C_\infty^{*'}$, the matrix in the middle will be of rank 2 and the 3x3 matrix U of eigenvectors will be the solution you are looking for. If you apply U to the image, that should get rid of both the projective and the affine distortion. This is referred to as the 1-step approach to the elimination of both distortions.

For further information on the 1-step approach, see pages 42, 55 and 56 of the text.

(a) Building

(b) Portraits

2 Tasks

Using Point-to-Point Correspondences: You have used point-point correspondences in the previous homework to transform images. In this homework you will use a set of corresponding points to remove distortion from images.

What you need for this are the height and width of a planar object in the scene that is captured by the image. You can find the height and width of the planar objects found in the images shown in Figure 1 at [I].

With the given height and width, your points in the undistorted image should be (0,0), (0, width), (height, 0), and (height, width). Their corresponding points in the distorted input image can be manually measured using a software such as GIMP.

After you have found the correspondences you simply need to apply the homography to the input image to remove distortion.

2-Step Method: Use the 2-step method as described above to first remove the projective distortion and then the affine distortion.

You will get extra credit if you show the calculation of different Vanishing Lines in your image using different sets of parallel lines. In order to create a scene with multiple sets of parallel lines going in different directions, you may have to create one yourself either with a graphics program or by physically drawing lines on a sheet of paper. See if you actually get distinctly different vanishing lines or if they all turn out to be the same VL (except for minor discrepancies caused by sampling errors and numerical precision issues in the computer).

1-Step Method: As stated in the Introduction, in this method estimate the camera image $C_\infty^{*'}$ of the Dual Degenerate Conic C_∞^*. From the parameters of the conic images, you estimate the homography that gets rid of both the projective and the affine distortion in one go.

Show results on your own images: Using at least 2 of your own images repeat above tasks. The images you capture should have significant projective and affine distortion.

2.1 Notes

1. You can find the provided images on the course website [I].
2. You can use any image editor such as GIMP to determine the pixel coordinate values in the image.
3. You can use the OpenCV C++/Python libraries to handle low-level image and matrix operations. However you cannot use the built-in opencv functions such as findHomography or warpPerspective.

2.2 Submission

1. Turn in a typed pdf of your report via Blackboard.
2. Your pdf must include a description of
   - The logic that you used to solve the given tasks.
   - The steps that you used for each of the tasks with relevant equations
   - The input and output images for each task. **Illustrate clearly the parallel / orthogonal lines that you chose in the input image.**
   - Your source code.
   - Your observations on the relative performance of the two methods.
3. You are permitted to look at sample solutions from previous years to get an understanding of how to solve the problems. **Your final report must be your own.**

References

[I] `https://engineering.purdue.edu/RVL/ECE661_2018`