

EE 576 - Project 2

The project aims to familiarize you with transformations between images. For this, consider the image sequence given. Consider the following transformation (homography map) between a pair of consecutive images:

$$x' = Hx = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix} x$$

where x refers to the homogeneous representation of a pixel from the first image and x' refers to the homogeneous coordinates of the corresponding pixel in the second image.

1. Using your code from the previous project, given a number n in the sequence, display the n^{th} and $n + 1^{th}$ images side by side in the first and second cells of your 2×2 grid of images. Then for each $i = 1, \dots, N \gg 3$ from the first image, designate the corresponding point from the other using the mouse, clicking on the point in the first image and then clicking on the corresponding point in the second image. Store these points as two vectors of 2D points (`vector<cv::Point2D>`).

2. First, consider the method as discussed in class. Let $h = [h_{11} \ h_{12} \ h_{13} \ h_{21} \ h_{22} \ h_{23}]$ and define $Ah = a$ as the equation to be solved to find the h vector using the pairs of corresponding matched points. What is the minimum number N_1 of corresponding points? Using the corresponding points as found in part (1), setup the $Ah = a$ equation and solve for h using pseudo-inverse method of OpenCV. Finally, define and print the H matrix. Now use the H matrix to determine for each point in part (1), the corresponding point in the second image. Show each matched pair by putting a small circle of random color (same color in the first and second image) on the respective images. Compute the average error $e = \frac{1}{N} \sum_{i=1}^N \|x'_i - Hx_i\|$.

3. Now use OpenCV's `findHomography` and `drawMatches` and to write a C/C++ method named `findHomographyMap` as follows:

1. Input: A set of N_1 corresponding points x and x'
2. Function: Computing the homography matrix using OpenCV api (i.e. `findHomography`). Pls refer to https://docs.opencv.org/4.x/d9/dab/tutorial_homography.html
3. Now apply the OpenCV computed transform H on the points selected in the first image and show the corresponding matched points in the second image using `drawMatches` method. Again compute the average error e .

4. Write a max 2 page report in using Latex in which you show your results from parts (2) and (3), comment on how the results are and how they compare. Please use the report template (latex) provided. Pls also submit your latex files, figures and pdf file.

Recall that your code should be as **modular** as possible. This means that

- All the parameter settings and method-class definitions should be in a header file (*.h or *.hpp files)
- The code should not have any numbers directly - rather you should use parameter values as defined in the header file.
- The methods-functions need to be defined in a separate *.cpp file
- The main method should be minimal as possible
- The code should not contain any references to your local directory setup.

- Please be sure to read the Project Grading Policy in the course syllabus.
- Please also make sure that your hand-in complies fully with the instructions as specified therein.
- If you are familiar with OOP, you may use C++ and generate the appropriate classes as required.
- Pls upload **source codes** in a rar/zip file named as follows **HwX_LastNameFirstNameInitial**. For example a student named Ali Kayhan would hand in his first homework with the following name: **Hw1_KayhanA.zip**. Pls do not include any executable files. Also pls do not include anydata files unless otherwise asked.