Visió Tridimensional

Pràctica 5: 3D Reconstruction

Goals and Requirements

The goal of the assignment is to learn the following concepts.

- The Essential matrix and how to compute it from the Fundamental matrix and the camera calibration.
- How to recover the camera motion from the essential matrix.
- How to triangulate point matches to reconstruct their 3D position.

We will assume that we already know the following concepts:

- What is the Fundamental matrix and how to compute it from SIFT correspondences.
- What is the camera projection matrix and how it is decomposed in internal and external parameter matrices.
- The algebraic relation between the Fundamental matrix and the camera matrices.

Introduction

In this session, we are going to compute the 3D position of a pair of cameras and a set of keypoints. For this, we will first compute correspondences between the two images. Then, we will robustly compute the Fundamental matrix using the code we did in the last session. From the Fundamental matrix and the camera calibration matrix that we did compute in the third session, we will get the Essential matrix, which encodes the motion between the two cameras. We will then compute the motion between the cameras and, finally, triangulate the matched keypoints to obtain their 3D position.

Directions

Triangulation

Before recovering the camera motion, we will need to first implement a function to triangulate matches. This is a function that takes as input a match and the projection matrices of two cameras, and computes the 3D position of a match.

We want then to find the 3D point X such that its projection onto the images are $x = PX$ and $x' = P'X$. These are homogeneous equations that yield to the following system of equations.

$$\begin{aligned}
x_1(p_3X) &= p_1X && (1)\\
x_2(p_3X) &= p_2X && (2)\\
x'_1(p'_3X) &= p'_1X && (3)\\
x'_2(p'_3X) &= p'_2X && (4)
\end{aligned}$$

where p_i refers to row i of matrix P.

Question 1 *Write the steps to show that the system of equations is equivalent to the homogeneous equations* $x = PX$ *and* $x' = P'X$.

Question 2 *Write the equations in the matrix form* $AX = 0$.

Question 3 *Write the code to solve the system in the function* `triangulate.m`. *Use the test code provided in* `practica5.m` *to validate that* `triangulate` *is working properly.*

Load the images and match keypoints

Now with the `triangulate` function implemented we are ready to start. As usual, we will start by loading the images and matching SIFT keypoints between them. Use the images of your face that we captured with the multi-camera system. Choose two consecutive cameras to work with (for example camera 2 and 3).

Compute Fundamental Matrix

Use the code from the last practical session to compute the Fundamental matrix from the matched keypoints. You will need to first copy the functions `fundamental_matrix` and `ransac_fundamental_matrix` that you completed during the last session onto the working directory.

Compute Essential Matrix and Camera Positions

As seen in the theory class (see the theory `Visio3DT5.pdf`, page 10), the motion between the cameras can be computed from the Essential matrix. To compute the Essential matrix, `E`, we will need the Fundamental matrix, `F`, that we have just computed and the camera calibration matrix, `K`, that we computed during the third session.

Question 4 *Compute the Essential matrix from the Fundamental matrix and the camera calibration matrix.*

We will assume that the first camera is at the origin, with no rotation, and that the second camera has a rotation and translation with respect to the first.

Question 5 *Write the camera projection matrix* `P` *for the first camera.*

The rotation and translation of the second camera can be computed from the SVD decomposition of `E`. There are 4 possible solutions.

Question 6 *Complete the code to compute the 4 candidate solutions for the second camera projection matrix.*

Question 7 *How can we choose the right solution from the 4 candidates? The code to do it is provided, explain how it works.*

Once the proper solution is chosen, we can triangulate all the matches to get a sparse point cloud.

Question 8 *Complete the code to triangulate all matches.*

Use the provided code to plot the reconstructed points. If everything went fine, you should recognize the sparse points corresponding to the different objects at their corresponding 3D positions.

Finally, we are going to compute the reprojection error of the reconstructed points. This is the distance between the detected keypoints, `x1` and `x2`, and the projection of the reconstructed points.

Question 9 *Complete the code to compute the reprojection error of each match. Plot the histogram of the errors.*

Question 10 *Change the SIFT threshold to 0.05 first and then to 0.001. Repeat the reconstruction for these two new set of 2D points and comment the results obtained.*

Question 11 *Repeat all the process with your own images taken with your camera. Try reconstructions for different pairs of images and comment the results obtained.*