Technische Universität München
Fakultät für Informatik
Prof. Dr. N. Navab

WiSe 2011/2012
Exercise Sheet 7
January 19, 2012

Exercises in 3D Computer Vision II

Exercise 1 Factorization

In this exercise, we will recapitulate the basic properties of the factorization algorithm for the structure from motion problem. Let $\mathbf{X}_j$, $j = \{1,\ldots,n\}$, denote a set of 3D points, let $\mathbf{P}^i$, $i = \{1,\ldots,m\}$, represent camera matrices and let $\mathbf{x}^i_j$ be the projection of the j-th 3D point into the i-th camera view.

a) What is the objective of the factorization algorithm?

b) Recapitulate the basic steps of the *affine* factorization algorithm.

c) Recall the properties of an affine camera. Why is an affine camera model assumed in the algorithm above?

d) Explain why all 3D points have to be visible in all images.

e) Write down the error that is being minimized in the affine factorization algorithm.

f) Recapitulate the basic steps of the *projective* factorization algorithm.

g) Compare the assumptions made on the input 2D data and the properties of the computed 3D reconstruction between the affine and projective factorization algorithms.

h) Explain why the image data points need to be normalized for the projective factorization algorithm and how to perform this normalization.

i) Write down the error that is minimized in the projective factorization algorithm.

j) Explain the necessity and the process of normalizing the projective depths λ^i_j. *Hint:* Start by interpreting the error that is minimized and think of how the error equation simplifies when the error becomes minimal. Then, observe what properties you want the projective depths to fulfill.

Exercise 2 (H) Structure From Motion

In this exercise, you are asked to implement the factorization algorithm in MATLAB. You can find two data sets on the lecture website: pictures (`hotel_pictures.zip`) and 2D feature locations (`hotel_points.dlm`). The pictures show an artificial house model from 100 perspectives and there are 133 features, visible in all of the pictures.

a) Read in the data points and display 10 of the provided images, overlaid with the corresponding feature points (marked with green crosses).

b) Write a MATLAB function `factor_affine.m` that takes as an input a measurement matrix $\mathbf{M}$, i.e. a matrix containing all the 2D feature points, concatenated for all the views. The function should return three estimated quantities: the translations for all views, the structure matrix $\hat{\mathbf{S}}$ and the projection matrix $\hat{\mathbf{P}}$.

c) Apply your affine factorization function to the provided 2D feature points and display the reconstructed 3D points (use MATLAB command `plot3`). In addition, display the estimated camera locations in your reconstructed 3D scene. *Hint:* You can assume a reasonable overall scaling factor for your reconstructed 3D scene for displaying the camera locations.