MVG Assignment 4
Model Fitting

Instructions

- Due date: June 1 2021.
- Submission link: `https://www.dropbox.com/request/64CREJwn1yYcbC6VJd8N`
- You should submit a zip file containing: A pdf file with all complete solutions and the code with a main script which calls the required scripts/functions for each computer exercise.
- The report should be written individually, however you are encouraged to work together.
- We highly recommend typing your solutions in Latex.
- Notice that in the end of this exercise you are given a list of useful matlab commands for all the computer exercises.

Plane Fitting

Exercise 1

In RANSAC, the size of the sample set depends on the degrees of freedom of the model. If we want to fit a 3D plane to a set of points **how many** degrees of freedom does the model have?
If the point set contains 10% outliers, **how many** sample sets do we need to draw to achieve a success rate of 98%?

Exercise 2

In this exercise you will derive the formula for the solution of the total least squares problem.

a.

Suppose that $(x_i, y_i, z_i), i = 1, \dots, m$ are 3D points that we want to fit to a plane (a, b, c, d). We want to minimize the sum of squared distances from the plane to the points by solving:

$$\min \sum_{i=1}^{m} (ax_i + by_i + cz_i + d)^2 \tag{1}$$

$$\text{s.t. } a^2 + b^2 + c^2 = 1 \tag{2}$$

Given a, b, and c, **show** that the optimal d must fulfill:

$$d = -(a\overline{x} + b\overline{y} + c\overline{z}) \tag{3}$$

Where $(\overline{x}, \overline{y}, \overline{z}) = \frac{1}{m} \sum_{i=1}^{m} (x_i, y_i, z_i)$

b.

Substituting the optimal d we get the following problem:

$$\min \sum_{i=1}^{m} (a\tilde{x}_i + b\tilde{y}_i + c\tilde{z}_i)^2 \tag{4}$$

$$\text{s.t. } 1 - (a^2 + b^2 + c^2) = 0 \tag{5}$$

(a) house1.jpg

(b) house2.jpg

Figure 1

Where $(\tilde{x}, \tilde{y}, \tilde{z}) = (x_i - \overline{x}, y_i - \overline{y}, z_i - \overline{z})$.
This is a constrained optimization problem of the type

$$\min f(t) \tag{6}$$

$$\text{s.t. } g(t) = 0 \tag{7}$$

Using Lagrange multipliers, the solution to such a problem fulfills

$$\nabla f(t) + \lambda \nabla g(t) = 0 \tag{8}$$

Show that the solution to the problem in equation 4 must be an eigenvector corresponding to the smallest eigenvalue of the matrix

$$\sum_{i=1}^{m} \begin{bmatrix} \tilde{x}_i^2 & \tilde{x}_i\tilde{y}_i & \tilde{x}_i\tilde{z}_i \\ \tilde{y}_i\tilde{x}_i & \tilde{y}_i^2 & \tilde{y}_i\tilde{z}_i \\ \tilde{z}_i\tilde{x}_i & \tilde{z}_i\tilde{y}_i & \tilde{z}_i^2 \end{bmatrix} \tag{9}$$

Computer Exercise 1

Figure 1 shows two images of a house and a set of 3D points from the walls of the house. The goal of this exercise is to estimate the location of the wall with the most 3D points.
The file `compEx1data.mat` contains cameras P, inner parameters K for both cameras, scene points X and some extra points x from image 1.

a.

Solve the total least squares problem with all the points and find the plane which contains the wall.
Compute the RMS distance between the 3D-points and the plane

$$e_{RMS} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\frac{(ax_i + by_i + cz_i + d)^2}{a^2 + b^2 + c^2}} \tag{10}$$

b.

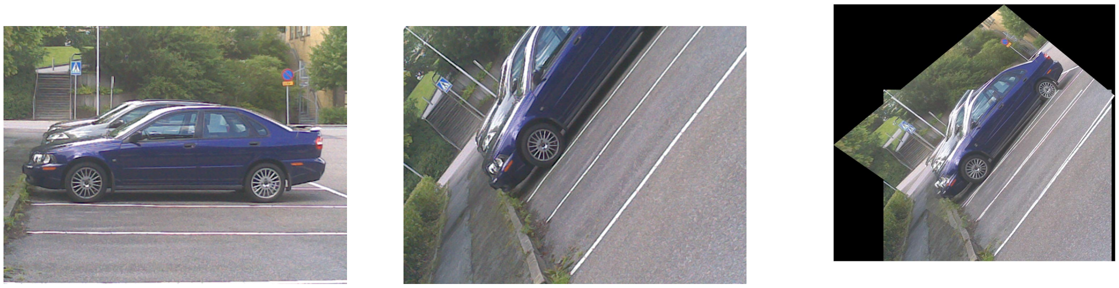
Use RANSAC to robustly fit a plane to the 3D points X.
If a 3D point is an inlier when its distance to the plane is less than 0.1, **how many** inliers do you get?
Compute the RMS distance between the plane obtained with RANSAC to the 3D points. Is there any improvement?
Plot the absolute distances between the plane and the points in a histogram with 100 bins.

c.

Solve the total least squares problem with only the inliers.
Compute the RMS distance between the 3D-points and the new plane.
Plot the absolute distances between the plane and the points in a histogram with 100 bins. **Which** estimation produced the best result?
Plot the projection of the inliers into the images. Where are they located?

d.

Using the method in Assignment 1 Exercise 6 (Reminder: $H = R - t\pi^T$ where $P1 = \begin{bmatrix} I & 0 \end{bmatrix}$,$P2 = \begin{bmatrix} R & t \end{bmatrix}$ and $\Pi = (\pi^T, 1)^T$ is the plane) **compute** a homography from camera 1 to camera 2. (Don't forget that the formula only works for normalized cameras.)
Plot the points x in image 1. **Transform** the points using the homography and **plot** them in image 2. **Which** ones seem to be correct, and why?
For the report: The figures, the RMS errors with all the points, with RANSAC, and with all the inliers. and answers.



(a) a.jpg (b) b.jpg (c) Goal image

Figure 2

Robust Homography Estimation and Stitching

Exercise 3

Show that if the two cameras $P_1 = \begin{bmatrix} A_1 & t_1 \end{bmatrix}$ and $P_1 = \begin{bmatrix} A_2 & t_2 \end{bmatrix}$ have the same camera center then there is a homography H that transforms the first image into the second one. (You can assume that $A1$ and $A2$ are invertible.)

Exercise 4

Suppose that we want to find a homography that transforms one 2D point set into another.
How many degrees of freedom does a homography have?
What is the minimal number of point correspondences that you need to determine the homgraphy?
If the number of incorrect correspondences is 10% **how many** iterations of RANSAC do you need to find an outlier free sample set with 98% probability?

Computer Exercise 2

In this exercise you will use RANSAC to estimate homographies for creating panoramas.
You are given two images, and matching points between them, `xA` and `xB`. The goal is to place the two images on top of each other, as in figure 2.
Find a homography describing the transformation between the two images. Because not all matches are correct, you need to use RANSAC to find a set of good correspondences (inliers).
To estimate the homography use DLT with a minimal number of points needed to estimate the homography. (Note that in this case the least squares system will have an exact solution, so normalization does not make any difference.) A reasonable threshold for inliers is 5 pixels.
How many inliers did you find?

Transform the images to a common coordinate system using the estimated homography. For the report: The transformed image and short answers.

Usefull Matlab commands

1

- M = Xtilde(1:3,:)*Xtilde(1:3 ,:)’; % Computes the matrix from Exercise 2
- [V,D] = eig(M); % Computes eigenvalues and eigenvectors of M
- plane = null(X(:, randind)’); % Computes a plane from a sample set.
- plane = plane ./ norm(plane(1:3)); % normalize the plane to have length 1
- inliers = abs(plane’*X) <= 0.1; % If the fourth coordinate of the points in X is 1, finds the points with distance less than 0.1 to the plane.
- RMS = sqrt(sum((plane’*X).^2)/ size(X ,2));

2

- tform = maketform(’projective’,bestH’); % Creates a transfomation that matlab can use for images. Notice that H is transposed.
- transfbounds = findbounds(tform ,[1, 1; size(A,2), size(A,1)]); % Finds the bounds of the transformed image
- xdata = [min([transfbounds(: ,1); 1]) max([transfbounds(: ,1); size(B ,2)])];
- ydata = [min([transfbounds(: ,2); 1]) max([transfbounds(: ,2); size(B ,1)])];

% Computes bounds of a new image such that both the old ones will fit.

- [newA] = imtransform(A,tform ,’xdata’,xdata ,’ydata’,ydata); % Transform the image using bestH
- tform2 = maketform (’projective’,eye(3));
- [newB] = imtransform(B,tform2 ,’xdata’,xdata ,’ydata’,ydata ,’size’,size(newA));

% Put B inside bigger boundaries for both images

- newAB = newB;
- newAB(newB < newA) = newA(newB < newA);

% Writes both images in the new image.