Computer Vision, Assignment 4
Model Fitting

1 Instructions

In this assignment you will study model fitting. In particular you will use random sampling consensus RANSAC to robustly fit various models such as planes, homographies and essential matrices. The data for the assignment is available on Canvas.

The assignment is due Thursday Dec 3, 2020. Make sure you answer all questions and provide complete solutions to the exercises. Collect all the solutions and plots in one easily readable pdf-file. Write your name, the name of your collaborator (or that you have completed the exercises on your own) and the assignment number on the first page of the report. After each exercise there is a gray box with instructions on what should be included in the report. In addition, all the code should be submitted as m-files. Make sure that your matlab scripts are well commented and can be executed directly (that is, without loading any data, setting parameters etc. Such things should be done in the script).

You will have time to work with the assignments during the computer laboratory sessions / exercise sessions. These sessions are intended to provide an opportunity for asking questions on things you have had problems with or just to work with the assignment. More specifically, during the laboratory sessions you should concentrate on the exercises marked "Computer Exercise". The rest of the exercises are intended to provide hints and prepare you for the computer exercises. You are expected to have solved these before you go to the laboratory sessions.

The report should be written individually, however you are encouraged to work together in pairs (in the lab session you might have to work in pairs). Note that it is NOT allowed to do the assignments in larger groups than two persons. Keep in mind that everyone is responsible for their own report and should be able to explain all the solutions.

For a passing grade (3) in the course, all exercises except for the ones marked as OPTIONAL need to be completed and submitted before the due date. For higher grades (4 or 5), sufficiently many of the optional exercises should be correctly completed and submitted before the due date.

2 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 20% outliers, how many sample sets do we need to draw to achieve a success rate of 99%?

For the report: Submit the answers.

Exercise 2. (OPTIONAL, 1.3 points.) Suppose that (x_i, y_i, z_i), $i = 1, ..., m$ are 3D points to which we want to fit a plane (a, b, c, d). In this exercise you will derive the formula for the solution to the total least squares problem of plane fitting. We then want to solve

$$\min_{a,b,c} \quad \sum_{i=1}^{m}(ax_i + by_i + cz_i + d)^2 \tag{1}$$

$$\text{such that} \quad a^2 + b^2 + c^2 = 1, \tag{2}$$

that is, minimize the sum of squared distances from the plane to the points (see lecture notes).

Show by taking the derivative with respect to d that given a, b, and c, the optimal d must fulfill

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

where

$$(\bar{x}, \bar{y}, \bar{z}) = \frac{1}{m}\sum_{i=1}^{m}(x_i, y_i, z_i). \tag{4}$$

Substituting into (1) we get

$$\min_{a,b,c} \quad \sum_{i=1}^{m}(a\tilde{x}_i + b\tilde{y}_i + c\tilde{z}_i)^2 \tag{5}$$

$$\text{such that} \quad 1 - (a^2 + b^2 + c^2) = 0, \tag{6}$$

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

$$\min_t \quad f(t) \tag{7}$$

$$\text{such that} \quad g(t) = 0. \tag{8}$$

According to the method of Lagrange multipliers (see e.g. Persson-Böiers, "Analys i flera variabler"), the solution of such a system has to fulfill

$$\nabla f(t) + \lambda \nabla g(t) = 0. \tag{9}$$

Show that the solution (a, b, c) of (5)-(6) must be an eigenvector of the matrix

$$\sum_{i=1}^{m}\begin{pmatrix} \tilde{x}_i^2 & \tilde{x}_i\tilde{y}_i & \tilde{x}_i\tilde{z}_i \\ \tilde{y}_i\tilde{x}_i & \tilde{y}_i^2 & \tilde{y}_i\tilde{z}_i \\ \tilde{z}_i\tilde{x}_i & \tilde{z}_i\tilde{y}_i & \tilde{z}_i^2 \end{pmatrix}, \tag{10}$$

corresponding to the smallest eigenvalue.

For the report: Complete derivation.

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.

Figure 1: The images house1.jpg and house2.jpg and the 3D points.

Solve the total least squares problem with all the points. Compute the RMS distance between the estimated plane and all 3D-points

$$e_{RMS} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\frac{(ax_i + by_i + cz_i + d)^2}{a^2 + b^2 + c^2}}. \tag{11}$$

Now instead use RANSAC to robustly fit a plane to the 3D points X. A suggested inlier threshold is 0.1. Again, compute the RMS distance between the new plane estimate and all 3D points. Also plot a histogram of the absolute distances as before. Which is the better plane estimate, and why?

Project the filtered (inlier) 3D points into both cameras, and plot in the corresponding images. Where are these located, i.e. what plane seems to have been detected?

At this point (if you want) you can try to make a final fit to all inliers by solving the total least squares problem again, but with only the inliers. This might further improve results, provided the selected inlier set is good.

In Assignment 1, in Exercise 6 (optional), it was verified that for two normalized cameras $\tilde{P}_1 = \begin{bmatrix} I & 0 \end{bmatrix}$, $\tilde{P}_2 = \begin{bmatrix} R & t \end{bmatrix}$ and for 3D points on a plane $\Pi = (\pi^T, 1)^T$, projected into both cameras, the homography $H = (R - t\pi^T)$ maps the projected image points in $\tilde{P}_1$ to the corresponding image points in $\tilde{P}_2$. Now, the cameras P_1 and P_2 provided in this exercise are conveniently enough exactly on the form $P_1 = K\tilde{P}_1$, $P_2 = K\tilde{P}_2$, with normalized cameras on the form given above. Compute the homography H corresponding to the estimated plane. Plot the provided image points x in image 1. Transform the points using the homography and plot them in image 2. Which points seem to be transformed correctly, and why?

```
Useful matlab commands:

meanX = mean(X,2); %Computes the mean 3D point
Xtilde = (X - repmat(meanX,[1 size(X,2)]));
%Subtracts the mean from the 3D points
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));
%Makes sure that the plane normal has a unit length norm

inliers =  abs(plane'*X) <= 0.1;
%Finds the the indices for which the distance to the plane is less than 0.1.
%Note: Works only if the 4th coordinate of all the points in X are 1,
       and the plane has unit normal.

RMS = sqrt(sum((plane'*X).^2)/size(X,2)); %Computes the RMS error
```

For the report: Submit the m-file as well as RMS values and histogram plots (both when using all points and for RANSAC). Also submit the plots of points x in image 1, and the transformed points in image 2. Finally, provide answers to all questions.

3 Robust Homography Estimation and Stitching

Exercise 3. Show that if the two cameras $P_1 = [A_1\ t_1]$ and $P_2 = [A_2\ t_2]$ have the same camera center then there is a homography H that transforms the the first image in to the second one. (You can assume that A_1 and A_2 are invertible.)

For the report: Complete solution.

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?

For the report: Answers are enough.

Computer Exercise 2. In this exercise you will use RANSAC to estimate homographies for creating panoramas.

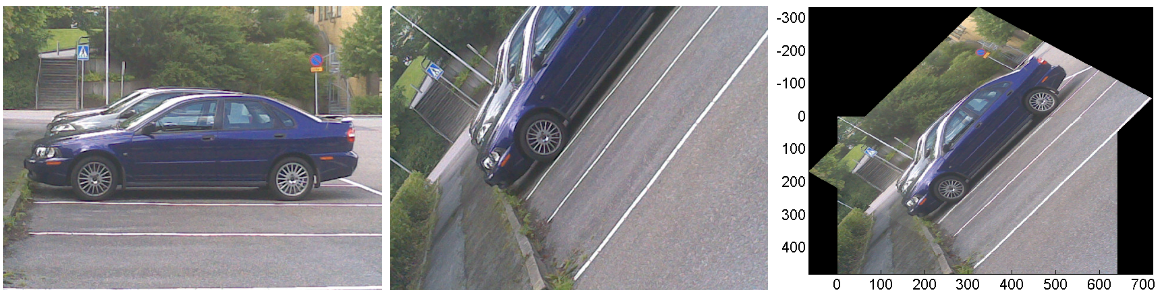

Figure 2: Image a.jpg, b.jpg and a panorama.

You can use the two images `a.jpg` and `b.jpg` (see Figure 2), but feel free to use other images if you want to (as long as they are taken from the same position, i.e. having same camera center). You will need to use VLfeat as in Assignment 2 to generate potential matches, and then determine inliers using RANSAC.

Begin by loading the two images in Matlab and display them. The images should be partly overlapping. The goal is to place them on top of each other as in Figure 2. Use VLFeat to compute SIFT features for both images and match them.

```
Useful matlab commands:

[fA dA] = vl_sift( single(rgb2gray(A)) );
[fB dB] = vl_sift( single(rgb2gray(B)) );

matches = vl_ubcmatch(dA,dB);

xA = fA(1:2,matches(1,:));
xB = fB(1:2,matches(2,:));
```

How many SIFT features did you find for the two images, respectively? How many matches did you find?

Now you should 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, and normalization does not make any difference.) A reasonable threshold for inliers is 5 pixels.

How many inliers did you find?

Next transform the images to a common coordinate system using the estimated homography.

```
Useful matlab commands:

tform = maketform('projective',bestH');
%Creates a transfomation that matlab can use for images
%Note: imtransform uses the transposed homography
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));
%Creates a larger version of B

newAB = newB;
newAB(newB < newA) = newA(newB < newA);
%Writes both images in the new image. %(A somewhat hacky solution is needed
%since pixels outside the valid image area are not allways zero...)
```

For the report: The m-file, a plot of the panorama (and the two images if you don't use a.jpg and b.jpg), and answers to all the questions.

Exercise 5. (OPTIONAL, 0.7 points.) Consider the following two polynomial equations in x and y:

$$x^2 + 2y^2 - 6 = 0$$

$$xy - 2 = 0.$$

There are four solutions to the above equations and your task is to compute them with the Action Matrix Method, see Lecture 7. Use monomial basis $\mathbf{m} = [1, x, y, y^2]^T$ and compute the action matrix M for multiplication with x. Hence, you need to derive M that fulfills

$$x\mathbf{m} = M^T\mathbf{m}.$$

What are the (right) eigenvectors to M^T? Scale them so that the first entry is 1 (since the first monomial in $\mathbf{m}$ is 1). From the eigenvectors, determine all solutions (all four solutions should be possible to find).

For the report: Complete solution, including action matrix M^T, eigenvectors (scaled), and the inferred solutions to the polynomial equations.