ENGN4528/6528 Computer Vision 2019
Computer-Lab-3 (CLab-3)

Objectives:

This is CLab-3 for ENGN4528/6528. The goal of this lab is to help you familiar with, and practice:

- DLT method for two-view homography estimation.
- Basic multi-view camera geometry, camera calibration. They are the basic building blocks for 3D visual reconstruction systems.

Note, however, in all the lab task descriptions given below, we by default use Matlab as the preferred language. You are however free to choose Python if you are more comfortable with Python. If you have not used Matlab or Python before, this lab is an opportunity to get you to quickly familiar with basic language usages of Matlab/Python for image processing and computer vision.

Special Notes:

1. Each computer lab has three weeks: session-A and session-B. Tutors/Lab instructor will provide basic supervision to both sessions.
2. Your Lab will be marked based on the overall quality of your Lab Report. The report is to be uploaded to Wattle site before the due time, which is usually on the Friday evening of Week-3 session of your lab.
3. It is normal if you cannot finish all the tasks within the two 2-hour sessions — these tasks are designed so that you will have to spend about 9 hours to finish all the tasks including finishing your Lab report. This suggests that, before attending the third lab session (in Week-2 of each CLab), you must make sure that you have almost complete 80.

Academic Integrity

You are expected to comply with the University Policy on Academic Integrity and Plagiarism. You are allowed to talk with / work with other students on lab and project assignments. You can share ideas but not code, you should submit your own work. Your course instructors reserve the right to determine an appropriate penalty based on the violation of academic dishonesty that occurs. Violations of the university policy can result in severe penalties.

CLab-3 Tasks

1 Two-View DLT based homography estimation. (4 marks)

(a) Left.

(b) Right.

A transformation from the projective space $\mathbb{P}^3$ to itself is called homography. A homography is represented by a 3x3 matrix with 8 degree of freedom (scale, as usual, does not matter)

$$\begin{bmatrix} x^C w \\ y^C w \\ w \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x^R \\ y^R \\ 1 \end{bmatrix} \tag{1}$$

The goal of this task is to use the DLT algorithm to estimate a 3x3 homography matrix. Pick any 6 corresponding coplanar points in the images ”left.jpg” and ”right.jpg” and get their image coordinates. In doing this step you may find it useful to check the Matlab function. *ginput*.

Calculate the 3x3 homography matrix between the two images based on your selected 6 pairs of corresponding points, using the DLT algorithm. You are required to implement your function in the following syntax.

H = DLT(u2Trans, v2Trans, uBase, vBase)

```
% INPUTS: u2Trans, v2Trans - vectors with coordinates u and v of the
% transformed image point (p') uBase, vBase - vectors with coordinates u
% and v of the original base image point p
% OUTPUT H: a 3x3 Homography matrix
% your name, date
```

In doing this lab task, should include the followings in your lab report:

1. List your source code for homography estimation and display the two images and the location of six pairs of selected points.
2. List the 3x3 camera homography matrix H that you have calculated.
3. Wrap the left image according to the calculated homography.Study the factors that affect the rectified results, e.g., the distance between selected points.

2 3D-2D Camera Calibration(Camera Resectioning). (6 marks)

(Acknowledgement: This task material is courtesy of Professor. Du Huynh of UWA).

Camera calibration involves finding the geometric relationship between 3D world coordinates and their 2D projected positions in the image. Four images, stereo2012a.jpg, stereo2012b.jpg, stereo2012c.jpg, and stereo2012d.jpg, are given for this CLab. These images are different views of a calibration target and some objects. For example, the diagram below is stereo2012a.jpg with some text superimposed onto it:

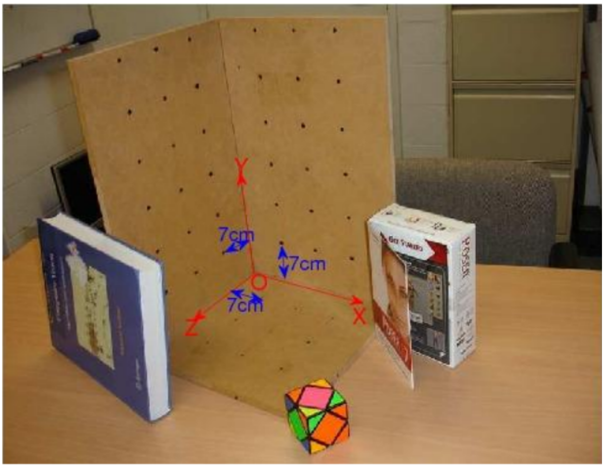


Figure 1: (Do not directly use the above image for your camera calibration work as it has been scaled for illustration. Use the original (unlabelled) image files provided.)

On the calibration target there are 3 mutually orthogonal faces. The points marked on each face form a regular grid. They are all 7cm apart.

Write a Matlab function to the following specification

function C = calibrate(im, XYZ, uv)

```
% CALIBRATE
%
% Function to perform camera calibration
%
% Usage: C = calibrate(im, XYZ, uv)
%
% Where:
% im - is the image of the calibration target.
% XYZ - is a N × 3 array of XYZ coordinates of the calibration target points.
% uv - is a N × 2 array of the image coordinates of the calibration target points.
% C - is the 3 × 4 camera calibration matrix.
```

```
% The variable N should be an integer greater than or equal to 6.
%
% This function plots the uv coordinates onto the image of
% the calibration target. It also projects the XYZ coordinates
% back into image coordinates using the calibration matrix
% and plots these points too as a visual check on the accuracy of
% the calibration process.
% Lines from the origin to the vanishing points in the X, Y and
% Z directions are overlaid on the image.
% The mean squared error between the positions of the uv coordinates and the
% projected XYZ coordinates is also reported.
```

From the 4 supplied images (stereo2012a.jpg, stereo2012b.jpg, stereo2012c.jpg, and stereo2012d.jpg), choose any image to work with. Use the suggested right-hand coordinate system shown in the diagram above and choose a sufficient number of calibration points on the calibration target.

Store the XYZ coordinates in a file so that you can load the data into Matlab (using the **load** function) and use them again and again. Note that each image can be calibrated independently. So you can choose different calibration points to calibrate each image. Neither do the numbers of calibration points need to be the same for your chosen images.

The **uv** coordinates can be obtained using the MATLAB function **ginput**. If one invokes **ginput** as follows:

```
uv = ginput(12) % e.g., to digitise 12 points in the image
```

and digitises a series of points by clicking with the left mouse button, then **uv** will be a matrix containing the column and row coordinates of the points that you digitised.

After the above operation, the variable **uv** should be a 12×2 matrix, each row of which should contain the coordinates of one image point.

Note: You need to ensure that, for each image, the numbers of 3D and 2D calibration points are the same. Thus, if your **uv** variable is a 12×2 matrix, then your XYZ variable should be a 12×3 matrix. Also, the points should appear in the same order in the two matrices.

Use the DLT algorithm to solve the unknown camera calibration matrix of size 3×4. (Refer to textbook, page 284, Section 6.2, or online resource to set up the DLT linear matrix equation that needs to be solved.)

You will end up setting up an equation of the form:

$$Aq = b$$

where:
A is a $2N \times 11$ matrix of constraint coefficients (in this case, $N = 12$).
q is an 11 element column vector of calibration matrix coefficients to be solved, and
b is a $2N$ column vector of the image coordinates of the target points.

This is the vector that is composed of the coordinates stored in the **uv** matrix. This over-constrained set of equations can be solved (in a least squares sense) in Matlab using the expression

$$q = A\backslash b$$

In this case where the matrix equation is over-constrained Matlab will perform a QR decomposition on the matrix A in order to solve the least squares solution. This technique overcomes many of the numerical problems that arise if one attempts to solve least squares problems via classical equations. (do a help on **mldivide**).

Hints:

1. In writing your code you will want to 'reshape' a 2D vector into a 1D vector. You can use the Matlab function reshape to reshape a matrix to arbitrary dimensions. You can also use the colon operator as follows:

$$a = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

$$b = a(:)$$

   Note however that Matlab extracts data from the source matrix by column(its Fortran heritage).

2. You can save your calibration matrices using Matlab's **save** command. For example the command

   save mydata im1 im2 calibPts uv1 uv2 C1 C2 % (do not use commas between variable names)

   will save your variables im1 im2, calibPts, uv1, uv2, C1,and C2 in a file called mydata.mat. At a later date you can reload this data with the command:

   load mydata

   The variables im1 im2, calibPts, uv1, uv2, C1,and C2 will then be restored in your workspace.

Report Requirement and Marking Criteria: the same as all previous Lab Reports, the overall professionalism of your lab report will affect your final marks.

For this Task, you must complete the followings in your Lab-Report:

1. List your calibrate.m code in your PDF file.

2. List which image you have chosen for your experiment, and display the image in your PDF file.

3. List the 3×4 camera calibration matrix C that you have calculated.

4. Decompose the $\mathbf{C}$ matrix into $\mathbf{K}, \mathbf{R}, \mathbf{t}$, such that $\mathbf{C} = \mathbf{K}[\mathbf{R}|\mathbf{t}]$, by using the following provided matlab code (vgg_KR_from_P.m). List the results in your PDF file.

5. Answer the following questions:

   (a) what is the focal length of the camera (and in what unit of measurement)?

   (b) what is the pitch angle of the camera with respect to the X-Z plane in the world coordinate system? (Assuming the X-Z plan is the ground plane, then the pitch angle is the elevation angle of the camera's optical axis above the ground-plane horizon.)

   (Note: please only upload a single PDF file as your Lab Report. No separate Matlab file, or zip file, or image files are allowed. For more detail. Please refer to the following Template and General Instructions for Lab Report on following pages.)

Lab Report Requirement

Clab-* Report

ENGN4528/6528

name
UID
Master/Bachelor

dd/mm/yyyy

Lab Report Requirement

1 Files

Upload a single PDF file by the due date. You must use the following file name: CLab-1/2/3-Report-Uxxxxxx.pdf, replacing Uxxxxx with your uni-ID.

Your pdf file must contain the following contents:

(a) The report generally contains sample results from all the Lab Tasks, along with necessary comments, descriptions, observations, results, analysis and implementations.

(b) Please include your code in the Appendix.

2 Lab Report

Kindly document different question under respective headings provided with the assignment. For example:

Task-1: The Question

1. Your first question under this theme

Documentation, observations, results, analysis etc.

2. Your second question under this theme

Documentation, observations, results, analysis etc.

3. Your third question under this theme

Documentation, observations, results, analysis etc.

2.1 General Formatting Instruction

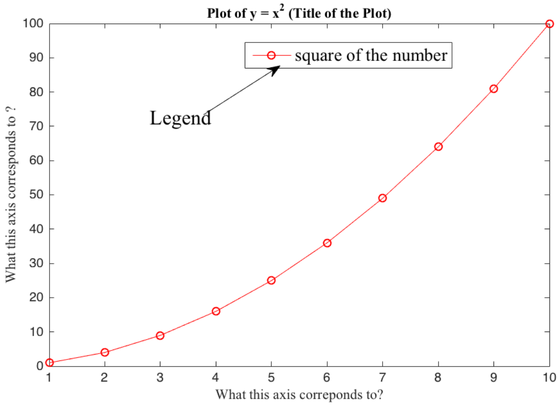
- Kindly use the same font single-spaced type for the entire document as much as possible, you may use the bold and italic version of the same font to highlight the important points. Few recommended fonts are Times New Roman, Times, which are quite widely used font to document projects and research papers. Too big or too small font sizes are not encouraged.

- Kindly, use appropriate font size for sections heading and its contents accordingly. For example, 14-15 point Times, boldface type for heading and 11-12 point single-spaced type for the content is one of the widely used font sizes for documenting research papers.

- Please number all your sections and subsections of the tasks as provided in the assignment.
- Please show the images mentioned in each task to make your answer more clear. **Brief explanations on how you solve the problem are expected**.
- Please give your own answer following the question guidelines.
- Handwriting draft is not permitted.

2.2 Table, Figures and Plots

This is one of the important aspects of evaluating your report. The figures and the caption of the tables must be appropriately addressed. The figure should have an appropriate title if required. All the legends in the figure should be properly highlighted. The caption of the figure should explain your observation and understanding which may comprise of quantitative or qualitative evaluation to endorse your observation. Some of the widely used font to caption your figure, table and callouts are 10-11 point Roman type, 10-11 point Helvetica non-boldface type. Kindly, adjust the size of the figure in the document appropriately such that it is clearly visible and perfectly eligible to illustrate your observation. We encourage you to look into the below example for reference. Note: You cannot insist we can zoom in or out to see tiny details on the graphs, plots, photographs, illustration, etc. Also, make sure the figures you include in your document is not a copyright image.



Caption: Variation in the y-axis corresponding to the values in the x-axis and What does this mean, your observations?

For the tables, graphs and others as well, kindly document the purpose of the statistical illustration which should include titles and proper labelling of the data and statistics.

Please follow the requirements to write your own Lab report.

=========== END of C-Lab-3 =========