University of Louisville **Dr. Aly A. Farag**
Electrical and Computer Engineering **Spring 2008**

ECE 619/645 – Computer Vision

Lab # 2 Camera Calibration

(Issued 2/14 – Due 2/28)

Camera calibration is the process of estimating two sets of parameters: the intrinsic parameters and the extrinsic parameters. Notations will be according to our textbook [1]. A good tutorial is in reference [2]. The technique to be implemented for camera calibration bears a great similarity to the Tsai approach [3].

The intrinsic parameters are the parameters necessary to link the pixel coordinates of an image point with the corresponding coordinates in the camera reference frame. These parameters are the entries of the intrinsic parameters matrix *K* where *K* is 3x3 matrix defined as:

$$K = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_0 \\ 0 & \beta/\sin\theta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where θ is the angle between u-axis and v-axis of the image plan (usually $\theta = \pi/2$) and (u_0 , v_0) are the coordinates of the image center(principle point) while α and β are the focal length in pixels in u and v directions respectively.

The extrinsic parameters define the location and orientation of the camera reference frame with respect to a known world reference frame. The extrinsic parameters thus defined as any set of geometric parameters that identify uniquely the transformation between the unknown camera reference frame and a known reference frame, named the world reference frame.

A typical choice for describing the transformation between camera and world frame is to use

- A 3-D translation vector, $t = [\, t_x\ t_y\ t_z \,]^T$, is describing the relative positions of the origins of the two reference frames, and
- A 3x3 rotation matrix, R, is an orthogonal matrix ($R^TR = RR^T = \mathbf{I}$) that brings the corresponding axes of the two frames onto each other. We can express the 3-D rotation as the result of three consecutive rotations around the coordinate axes by angles θ_x, θ_y *and* θ_z . The angles are then the three free parameters of R. The rotation matrix R can be expressed in terms of θ_x, θ_y *and* θ_z as:

$$ {}^{C}_{W}R = \begin{pmatrix} \cos\theta_y \cos\theta_z & \cos\theta_z \sin\theta_x \sin\theta_y - \cos\theta_x \sin\theta_z & \sin\theta_x \sin\theta_z + \cos\theta_x \cos\theta_z \sin\theta_y \\ \cos\theta_y \sin\theta_z & \sin\theta_x \sin\theta_y \sin\theta_z + \cos\theta_x \cos\theta_z & \cos\theta_x \sin\theta_y \sin\theta_z - \cos\theta_z \sin\theta_x \\ -\sin\theta_y & \cos\theta_y \sin\theta_x & \cos\theta_x \cos\theta_y \end{pmatrix} \quad (2) $$

The extrinsic parameters matrix, D, can be expressed in terms of t and ${}^{C}_{W}R$ as:

$$ D = \begin{bmatrix} {}^{C}_{W}R & t \\ 0_3^T & 1 \end{bmatrix} \quad (3) $$

where D is 4x4 matrix and 0_3=[0 0 0]^T.

The projection matrix is composition of the two sets of parameters. Hence, the projection matrix is the transformation between the 3-D world frame and the 2-D image frame:

$$ \begin{bmatrix} p \\ 1 \end{bmatrix} = \frac{1}{z_c} M \begin{bmatrix} {}^{w}P \\ 1 \end{bmatrix} \quad (4) $$

Where p is a vector that represents the point (*u,v*) on the image frame and ${}^{w}P$ is a vector that represents the point (*x,y,z*) in the 3-D space.

The projection matrix has the general form:

$$ M = AM_{proj} D \quad (5) $$

Where M_{proj} depends on the type of projection as follows:

- For perspective projection $M_{proj} = M_p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ (6)
- For weak perspective projection $M_{proj} = M_{wp} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & Z_c \end{bmatrix}$ (7)
- For para-perspective projection $M_{proj} = M_{pp} = \begin{bmatrix} 1 & 0 & -X_c / Z_c & X_c \\ 0 & 1 & -Y_c / Z_c & Y_c \\ 0 & 0 & 0 & Z_c \end{bmatrix}$ (8)

Many of the mechanics of camera calibration can be derived as follows (see references for details): Assume that the projection matrix M, defined up to an arbitrary scale factor, is:

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \tag{9}$$

Given any *N* 3-D points and their corresponding 2-D points, the relationship between the 2D and 3D perspective projections is

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z_c} \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \tag{10}$$

where the i*th* values (among N realizations) can be easily written as:

$$u_i = \frac{m_{11}X_i + m_{12}Y_i + m_{13}Z_i + m_{14}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}} \tag{11}$$

$$v_i = \frac{m_{21}X_i + m_{22}Y_i + m_{23}Z_i + m_{24}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}}. \tag{12}$$

The N values of u_i and v_i can be arranged in 2N linear equations in m's in the form

$$Pm = 0, \tag{13}$$

Where,

$$P = \begin{bmatrix} x_1 & y_1 & z_1 & 1 & 0 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 & -u_1z_1 & -u_1 \\ 0 & 0 & 0 & 0 & x_1 & y_1 & z_1 & 1 & -v_1x_1 & -v_1y_1 & -v_1z_1 & -v_1 \\ x_2 & y_2 & z_2 & 1 & 0 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 & -u_2z_2 & -u_2 \\ 0 & 0 & 0 & 0 & x_2 & y_2 & z_2 & 1 & -v_2x_2 & -v_2y_2 & -v_2z_2 & -v_2 \\ . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . \\ x_N & y_N & z_N & 1 & 0 & 0 & 0 & 0 & -u_Nx_N & -u_Ny_N & -u_Nz_N & -u_N \\ 0 & 0 & 0 & 0 & x_N & y_N & z_N & 1 & -v_Nx_N & -v_Ny_N & -v_Nz_N & -v_N \end{bmatrix} \tag{14}$$

and

$$m = [m_{11}, m_{12}, m_{13}, \ldots\ldots\ldots, m_{33}, m_{34}]^{\mathrm{T}} \tag{15}$$

As discussed in class (see [1] [2]) then the unknown m's can be recovered by the decomposition of P as follows:

$$P = USV^T \tag{16}$$

The solution is the eignvector V corresponds to the smallest eignvalue in the main diagonal of S. Matlab command: [U,S,V] = svd(P) decomposes P into U, S, and V matrices. Now assume that the estimated perspective projection matrix M as above, then

- Let *M* = (*A b*), with a_1^T, a_2^T, a_3^T are the rows of *A,* and b is the 4th column of M, then

$$\rho(A b) = K(R\ \mathrm{t}), \text{ and } R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{17}$$

Hence,

$$\rho\begin{pmatrix} a_1^T \\ a_2^T \\ a_3^T \end{pmatrix} = \begin{pmatrix} \alpha r_1^T - \alpha \cot\theta\, r_2^T + u_0 r_3^T \\ \frac{\beta}{\sin\theta} r_2^T + v_0 r_3^T \\ r_3^T \end{pmatrix} \tag{18}$$

$$\rho = \frac{\varepsilon}{|a_3|}, where\ \varepsilon = \pm 1 \tag{19}$$

$$r_3 = \rho a_3 \tag{20}$$

- Let

$$a_1 = [m_{11}\ m_{12}\ m_{13}]^{\mathrm{T}}$$

$$a_2 = [m_{21}\ m_{22}\ m_{23}]^{\mathrm{T}}$$

$$a_3 = [m_{31}\ m_{32}\ m_{33}]^{\mathrm{T}}$$

$$b = [m_{14}\ m_{24}\ m_{34}]^{\mathrm{T}} \ ,$$

Therefore,

- $u_0 = \rho^2 (a_1 \cdot a_3)$ and $v_0 = \rho^2 (a_2 \cdot a_3)$

$$\cos\theta = -\frac{(a_1 \times a_3)\cdot(a_2 \times a_3)}{|a_1 \times a_3||a_2 \times a_3|} \tag{21}$$

- Then α and β can be recovered as:

$$\alpha = \rho^2 |a_1 \times a_3| \sin\theta \text{ and } \beta = \rho^2 |a_2 \times a_3| \sin\theta \tag{22}$$

- The remaining parameters can be computed as;

$$r_1 = \frac{a_2 \times a_3}{|a_2 \times a_3|} \tag{23}$$

$$r_2 = r_3 \times r_1 \tag{24}$$

- $\theta_y = \sin^{-1} r_{13}$, $\theta_x = \cos^{-1}(r_{33} / \cos\theta_y)$, and $\theta_z = \cos^{-1}(r_{11} / \cos\theta_y)$. (25)
- $t = \rho K^{-1} b$ (26)

Required Tasks

In this laboratory, we will estimate the projection matrix (i.e., the camera parameters) on synthetic and real data.

Part 1: Simulation with Synthetic Data

This part gives an idea about how to investigate the performance of camera calibration techniques. Due to absence of ground truth in case of real data, simulations with synthetic data are carried out. This series of experiments consists first of choosing specific external and internal camera parameters, which are referred to later as *ground truth*. After applying the calibration techniques, the obtained camera parameters are compared to the ground truth values. It is required to do the following:

1. For the given camera calibration parameters, obtain the perspective projection matrix $M_{\text{groundtruth}}$.

Parameter	t_x*(mm)*	t_y	t_z	θ_x*(rad)*	θ_y	θ_z	α *(pixel)*	β	u_0	v_0	θ*(rad)*
Ground Truth	-27	-28	701	0.09	0.8	-0.03	556	549	172	121	$\pi/2$

2. Generate a set of 3D points (x_w, y_w, z_w), where the number of points should be greater than six points, you can measure those points physically on a calibration pattern or you can use the 3-D points in file "points.3d" on the course website.
3. Generate the corresponding projected 2-D points using Eq(4).
4. To simulate errors in 2-D points, add white Gaussian noise with different signal-to-noise ration to the obtained 2-D points with signal-to-noise ratio ranging from 10 to 200 dB.
5. Use the linear calibration approach (use singular value decomposition (SVD) method, see above) to estimate the projection matrix *M*.
6. Compare between the obtained projection matrix *M* and the one you computed in step (1), i.e. $M_{groundtruth}$. Comment on your result.
7. Decompose the obtained projection matrix into its corresponding camera parameters, (see above).
8. Compare between the obtained camera parameters and the corresponding ground truth parameters. Comment on your result.
9. Implement *Gauss-Newton, Levenberg* and *Levenberg-Marquardt* nonlinear optimization algorithms using:
    a. Random initialization of camera parameters.

b. The output of the linear approach as an initial solution.

c. Comment on the performance of each nonlinear algorithm in terms of:

i. Sensitivity to the initial solution.

ii. Convergence to the optimal solution (speed and accuracy) in both cases (a) and (b).

iii. Sensitivity to the noise level in the 2D data.

10. Compare the results of linear and nonlinear approaches compared to the ground truth parameters, comment on your results.

11. Repeat (1-10) using different sets of 3D points, generalize your conclusion to answer the following questions:

a. Which is better, linear or nonlinear approaches, in terms of accuracy, speed and simplicity?.

b. Which is better, *Gauss-Newton, Levenberg* or *Levenberg-Marquardt?*

c. Which is better, random initialization or educated guess initialization, i.e. using the output of the linear approach as an initial solution to the nonlinear approaches?

d. Which is robust in case of noisy data?

12. Extra Credit:

a. Add to your report and appendix talking about different projection types.

b. Implement your camera calibration using different types of projections and comment on your results.

13. Extra Credit:

a. Add to your report an appendix talking about the physical meaning of *radial distortion*.

b. Derive the nonlinear equations of camera calibration when *radial distortion* is taken into consideration.

c. Modify your programs to handle the radial distortion case.

d. Provide your results and comments in case of radial distortion.

Part 2: Using Real Data

The idea of calibration is to match known 3D points to the corresponding 2D image points, writing down the projection equations and simultaneously solving them to extract the parameters. This part considers the following case: Using the camera calibration pattern and a set of known 3D points that are matched to the corresponding image points. The points have to be non-coplanar if we are to determine the full projection matrix. For more details see reference [3]. The accuracy of calibration depends on the accuracy of the measurements of the calibration pattern. To accomplish this part of the experiment do the following:

1. Design your calibration pattern, commenting on the reasons of choosing such pattern.
2. Define your world coordinate system.

3. Measure the 3D points on your calibration pattern (e.g., the 3D values of corner points). Use the metric system (mm, cm or meters). Calibration patterns are available at the CVIP lab where you can make these measurements or you can use your own.
4. Use your own camera (laptop camera/ webcam) to capture an image for your calibration pattern. Make sure that the two faces of the pattern appear in the image.
5. Find the 2D coordinates of the corner points in the captured image corresponding to the 3D points in step (3).

Extra credit:

Use image processing techniques to automatically detect 2D points on images of calibration pattern with their 3D world points. Goal is to minimize user intervention, you can make use of edge detection, linear Hough transform, corner detectors … etc. You may use the algorithm in page 84 in the reference [4] below.

6. Match 3D points in step (3) and 2D points in step (5) to find camera parameters as you did in part A.
7. Refine all the camera parameters by applying the nonlinear optimization approaches
8. Include radial-distortion correction parameters in your calibration procedure.
9. Finally demonstrate the accuracy of the estimated calibration parameters by projecting the corner points in the calibration pattern into the image plane. By visually comparing the locations of the projected corner points and their actual locations, one can ascertain the accuracy of a calibration procedure.

References

1. David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach*, Prentice Hall, New Jersey, 2003.
2. Shireen Y. Elhabian and Amal A. Farag, *On Image Formation*, CVIP Lab TR-02-08, University of Louisville, February 2008.
3. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision, Roger Y. Tsai, *IEEE International Conference on Computer Vision and Pattern Recognition*, 1986, pp 365-374.
4. M. Trucco, and A. Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall 1998.

Acknowledgements: *This Laboratory evolved over various iterations of offering of ECE 619(ECE645). Special thanks are due to Dr. Moumen Ahmed and Dr. Ahmed Eid who helped in putting this document together during 2002-2005. Their PhD work is readily available on the CVIP Lab website www.cvip.uofl.edu. The Laboratory was also implemented on the CardEye vision system by Abdelrehim Ahmed and Asem Ali and was maintained throughout 2005-2007. During Spring 08, Shireen Elhabian and Amal Farag updated the laboratory documentation and provided a clear procedure for calibration. Their technical report [2] is posted on the course website and on blackboard.*