HW 3: Advanced Topics in Computer Vision (580.464)

Instructor: René Vidal, Phone: 410-516-7306, E-mail: rvidal@cis.jhu.edu

Due 04/06/05 beginning of the class

1. **(20 points) Experimental Evaluation of the 8-point Algorithm from Two Perspective Views.** In this problem, you are asked to test the performance of the 8-point algorithm for different levels of noise, baseline, depth variation, and field of view.

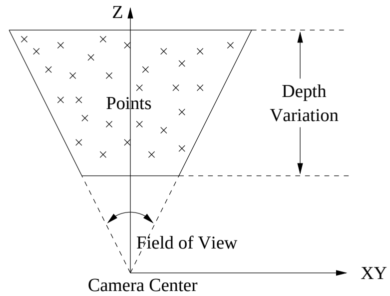

Figure 1: Truncated pyramid used to generate the structure.

   (a) **(5 points)** Use your function `points` to generate $P = 20$ points with a FOV of $60°$ and a depth variation from $Z_{\text{min}} = 1$ to $Z_{\text{max}} = 20$ units of focal length. Use your function `project` to project those 3D points onto two perspective images related by motion (R, T), where R is a randomly chosen rotation of $5°$ and T is a randomly chosen translation with norm $\|T\| = \tau Z_{\text{min}}$, where $\tau = 1$. Add zero-mean Gaussian noise with standard deviation $\sigma = 0, 0.5, 1, 1.5, 2$ pixels to the image data. When adding noise, use a calibration matrix of `K = [250 0 250; 0 250 250; 0 0 1]`. For each noise level, run 100 trials of your function `twoviewSFM` (i.e. 100 randomly chosen 3D points and motions) and compute the estimated rotation $\tilde{R}$ and translation $\tilde{T}$. For each trial compute the error in rotation and translation as

$$\text{Rot. error} = \text{acos}\Big(\frac{\text{trace}(R\tilde{R}^T) - 1}{2}\Big) \qquad \text{(degrees).} \tag{1}$$

$$\text{Trans. error} = \text{acos}\Big(\frac{T^T\tilde{T}}{\|T\|\|\tilde{T}\|}\Big) \qquad \text{(degrees).} \tag{2}$$

   Plot of the mean error (over the trials) for both rotation and translation as a function of noise. How does the error behave as a function of the noise level?

   (b) **(5 points)** For $\sigma = 1$, plot the mean errors as a function of the baseline $\tau = 0.1, 0.4, 0.7, 1.0, 1.3$. What is the effect of τ in the performance of the algorithm? Explain.

   (c) **(5 points)** For $\sigma = 1$, and $\tau = 1$ plot the mean errors as a function of the depth variation $Z_{\text{max}} = 1.1, 5, 10, 15, 20, 25$. What is the effect of Z_{max} in the performance of the algorithm? Explain.

   (d) **(5 points)** For $\sigma = 1$, $\tau = 1$ and $Z_{\text{max}} = 20$ plot the mean errors as a function of field of view $FOV = 20, 40, 60, 80, 100°$. What is the effect of FOV in the performance of the algorithm? Explain.

2. **(25 points) Implementation of the 7-point Algorithm from Three Perspective Views**

   (a) **(4 points)** Implement a function `T = trifocaltensor(x)` that computes the trifocal tensor `T` $\in \mathbb{R}^{3\times3\times3}$ from a set of point correspondences `x` $\in \mathbb{R}^{3\times P\times F}$, where P is the number of point correspondences and $F = 3$ is the number of views.

   (b) **(4 points)** Implement a function `[e2,e3] = T2Epipoles(T)` that computes the right epipoles in the second and third views from the trifocal tensor `T`.

   (c) **(8 points)** Implement a function `T = trifocaltensor(x,e2,e3)` that computes the trifocal tensor `T` $\in \mathbb{R}^{3\times3\times3}$ with given epipoles `e2,e3` from a set of point correspondences `x` $\in \mathbb{R}^{3\times P\times F}$. You should also submit your derivation of the algorithm. Notice that the function `trifocaltensor` is being overloaded, hence only one function should be written for parts a) and c).

   (d) **(3 points)** Implement a function `[R2,R3] = T2Rotations(T,e2,e3)` that computes the rotation matrices `R2,R3` in the second and third views from the trifocal tensor `T` and the epipoles `e2,e3`.

   (e) **(3 points)** Implement a function `[F2,F3] = T2Fundamental(T,e2,e3)` that computes the fundamental matrices `F2,F3` in the second and third views from the trifocal tensor `T` and the epipoles `e2,e3`.

   (f) **(3 points)** Generate noiseless data in three views and test the above functions. Repeat with noisy data. Which version of the `trifocaltensor` function works better in terms of the rotation and translation errors defined in Problem 1b)?

3. **(20 points) Motion Estimation from Multiple Views of Multiple Line Features.** Following the development of Section 8.3.3, derive a multiple view factorization algorithm for the line case using the rank condition on M_l. The algorithm is in spirit similar to algorithm 8.1. for point features, with the main difference being initialization. In particular, answer the following questions

   (a) **(3 points)** Given M_l describe how to compute the distance and the direction of each line l^j assuming known motions R_i, T_i and hence known M_l.

   (b) **(3 points)** Given the known 3-D line parameters show how to estimate $[R_i, T_i]$ for $i = 1, \dots n$.

   (c) **(3 points)** Integrate the above steps into an overall factorization based algorithm.

   (d) **(3 points)** How many lines in general position are needed? Why a minimum of three line features is needed?

   (e) **(8 points)** Show how to initialize the algorithm from three views as follows.

      i. Consider the trilinear constraint in (8.52)

      $$(l_2^T R_2 T_3^T l_3 - l_3^T R_3 T_2^T l_2)\widehat{l_1} = 0$$

      and show that it can be re-written as

      $$\begin{pmatrix} l_2^T G_1 l_3 & l_2^T G_2 l_3 & l_2^T G_3 l_3 \end{pmatrix} \widehat{l_1} = 0 \tag{3}$$

      where $G_1 = \mathbf{r}_2^1 T_3^T - T_2 \mathbf{r}_3^{1T} \in \mathbb{R}^{3\times3}$, $G_2 = \mathbf{r}_2^2 T_3^T - T_2 \mathbf{r}_3^{2T} \in \mathbb{R}^{3\times3}$ and $G_3 = \mathbf{r}_2^3 T_3^T - T_2 \mathbf{r}_3^{3T} \in \mathbb{R}^{3\times3}$, with $R_2 = [\mathbf{r}_2^1\ \mathbf{r}_2^2\ \mathbf{r}_2^3]$ and $R_3 = [\mathbf{r}_3^1\ \mathbf{r}_3^2\ \mathbf{r}_3^3]$.

      ii. Show that one can solve linearly for the 27 unknowns in G_1, G_2 and G_3 up to a scale factor from 13 line features using (3).

      iii. Show that $G_1^T \widehat{T_2} \mathbf{r}_2^1 = G_2^T \widehat{T_2} \mathbf{r}_2^2 = G_3^T \widehat{T_2} \mathbf{r}_2^3 = 0$. Thus, assuming that G_1, G_2 and G_3 are rank 2 matrices, the following matrix is known (with each column up to a scale factor) $H_2 = [\widehat{T_2}\mathbf{r}_2^1\ \widehat{T_2}\mathbf{r}_2^2\ \widehat{T_2}\mathbf{r}_2^3]$.

      iv. Show that the range of H_2 is the same as the range of the essential matrix $E_2 = \widehat{T_2} R_2$ and that one can obtain E_2 from the SVD of H_2. Outline a similar procedure to obtain $E_3 = \widehat{T_3} R_3$ from the right null space of G_1, G_2 and G_3.

      v. Show that one can now recover (R_2, T_2) and (R_3, T_3) using the last step in the 8-point algorithm.

4. **(15 points) Optimal Reconstruction from Three Perspective Views.** Derive an expression for the first order approximation to the reprojection error in the case of point correspondences in three perspective views.

5. **(10 points) Calibration with Partial Knowledge of the Structure and K.** Consider a camera with the calibration matrix

$$K = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where the only unknown parameter is the focal length f. Assume that the image of the center of projection is known, the skew is zero and the aspect ratio is 1. Suppose you have a single view of the rectangular planar structure, whose four end points in the world coordinate frame have following coordinates $\mathbf{X}_1 = [0, 0, 0, 1]^T$, $\mathbf{X}_2 = [\alpha b, 0, 0, 1]^T$, $\mathbf{X}_2 = [0, b, 0, 1]^T$, $\mathbf{X}_4 = [\alpha b, b, 0, 1]^T$, where one of the dimension of the plane b as well as the ratio α between the two sides of the rectangle are unknown.

   (a) Write down the projection equation for this special case relating the 3-D coordinates of the planar points to their image projections.

   (b) Show that the image coordinates $\mathbf{x}$ and the 3-D coordinates of points on the world plane are in fact related by a 3×3 homography matrix of the following form

   $$\lambda \mathbf{x} = H[X, Y, 1]^T.$$

   Write down the explicit form of H in terms of camera pose R, T and the intrinsic parameters of the camera H.

   (c) Assuming the known structure (up to scales α, b) describe an algorithm for recovering the unknown homography H.

   (d) Given H describe steps which would enable you to factor it and recover the unknown focal length f and rotation R. Also recover translation T and ratio α up to a universal scale factor.