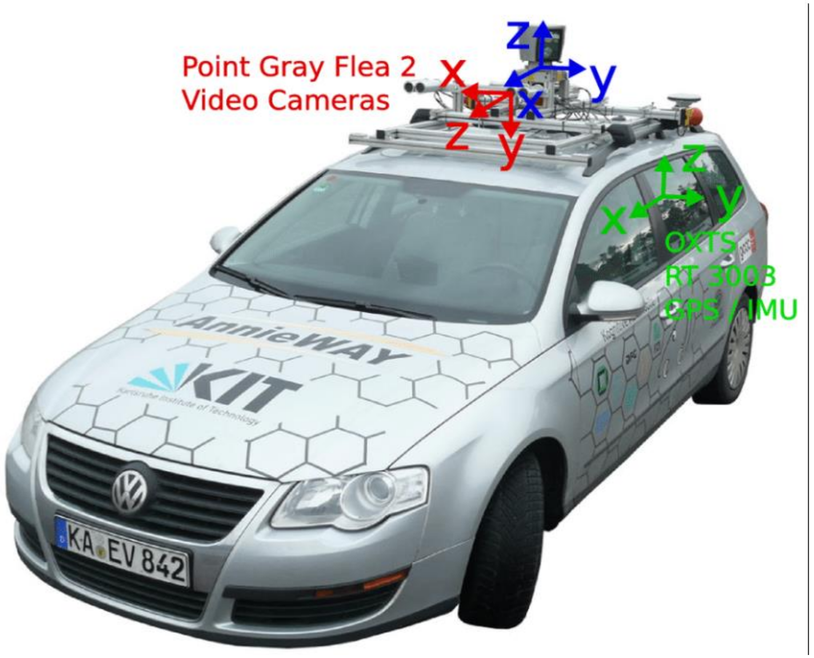
Assignment 1
CS 9645

Noise Robustness of PnP Routine for Cross-Calibration

Mohammad Karami

Instructor: Dr. Steven Beauchemin

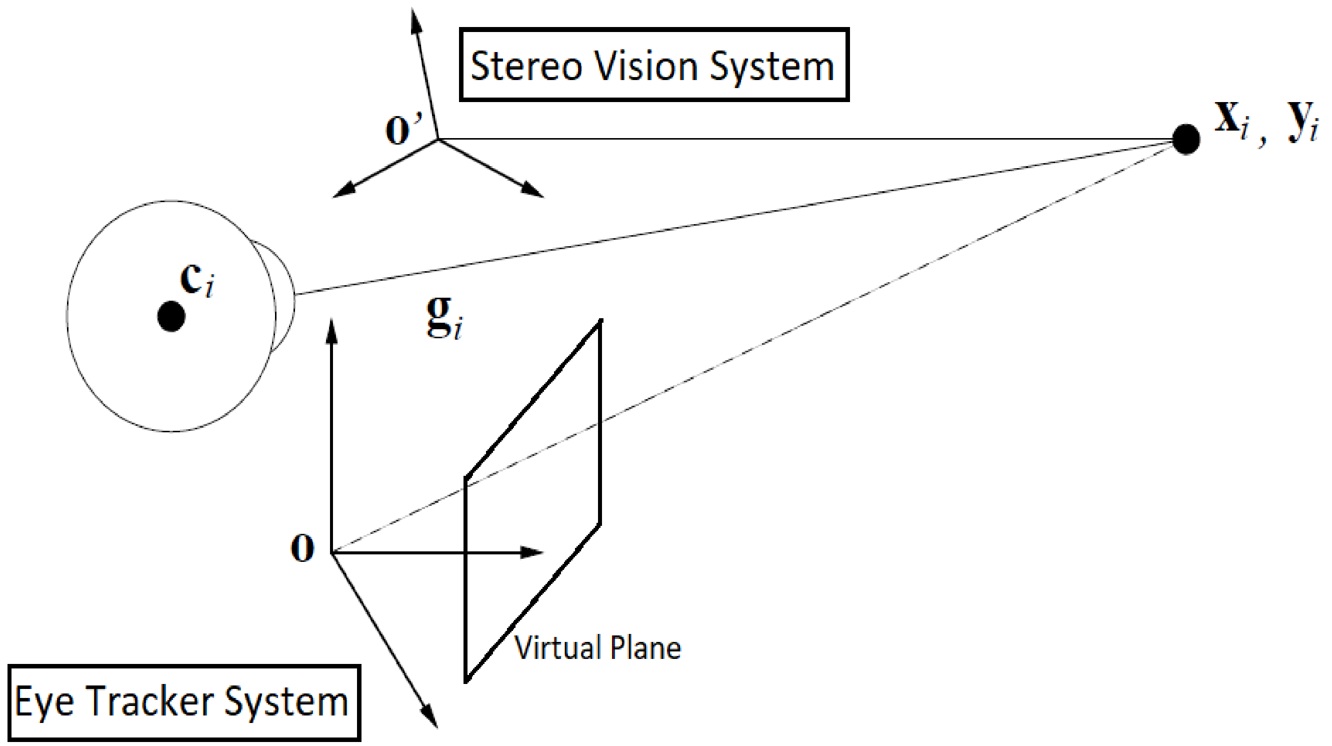


Western Science

Problem

- Find the Rotation and Translation matrix between two coordinate systems of stereo vision system and eye tracker system
- Use PnP() routine
- Add Gaussian noise and see robustness of PnP() with respect to the noise
- Six set of synthetic data of gaze vector and world coordinate points have been provided

World coordinate points	Gaze vector
(0,3,50)	(0,3,30)
(2,-5,47)	(-3,-5,28)
(-1,7,60)	(10,7,31)
(5,-1,40)	(-10,1,25)
(0,2,45)	(-5,2,30)
(3,-4,44)	(-6,-4,27)



Courtesy of Taha 2013

Solution

1. K = 1

2. Add a Gaussian noise with *mean=0* and *StandardDeviation=0.01*k*m* to each element of gaze vector (m is the magnitude of gaze vector).

3. Find Cross-Section of gaze vector with virtual plane located at *Z=1* in the eye tracker coordinate system.

4. Use the cross-section point from step2 and 3D noiseless points to obtain Rotation and Translation matrix from EPnP

5. Obtain the unknown scalar factor, ω_i, from the following equation:

$$\omega_i \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = M \left(R \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} + t \right)$$

Solution

6. Obtain the reprojection 3D points from the following equation:

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = R^{-1}M^{-1}\omega \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} - R^{-1}t$$

7. Obtain average norm difference between reprojection 3D points and noiseless 3D points, referred to as error

8. If K < 20, k++ and go to step 1

9. Plot error for the 20 iterations

10. Follow the same procedure, but use PnPRansac and compare the results with EPnP. (PnPRansac is supposed to minimize the reprojection error).

Results

Noiseless extrinsic camera matrix from EPnP is (done in C++):

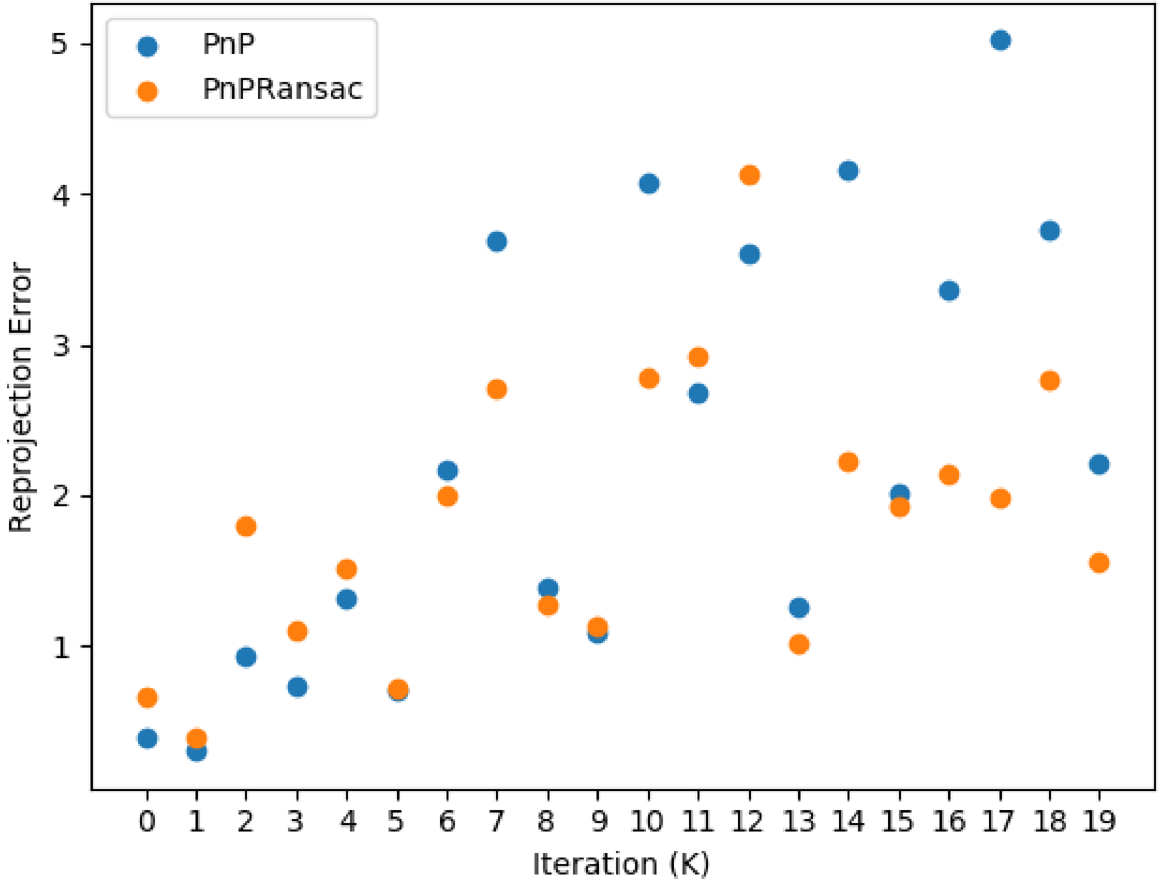
- Its comparison with PnPRansac is also provided in the Appendix Slide
- Note that OpenCV PnP provides different result in C++ and Python.
  Its Comparison is also provided in Appendix Slide

$$\left[\begin{array}{ccc:c} 0.07 & -0.02 & 0.99 & -49.49 \\ 0.25 & 0.96 & 0.004 & -0.065 \\ -0.96 & 0.25 & 0.074 & 24.29 \end{array}\right]$$

Results

Comparison between PnP and PnPRansac with respect to the increase in Gaussian random noise to the gaze vector:

- As expected, by increasing noise, the reprojection error is increasing as well.
- For low-noise (k<10), there is not much difference between EPnP and PnPRansac!
- For high-noise (K>10), PnPRansac acts better than EPnP!



How To Run the C++ Code

1. Run the make file: $make

2. Run the executable file: ./calib.out

3. A Text file named “Error.txt” will be generated. This file includes the reprojection error from PnP and PnPRansac

4. Run the Python file “Plot_error.py” to read the text file and generate the plot.
   The python file uses Matplotlib for visualization.

Courtesy of Taha 2013

Appendix

Comparison between EPnP and PnPRansac in OpenCV of C++ and Python (noiseless case):
Only C++ code is submitted!

	EPnP	PnPRansac
C++	$\begin{bmatrix} 0.07 & -0.02 & 0.99 & \vdots & -49.49 \\ 0.25 & 0.96 & 0.004 & \vdots & -0.065 \\ -0.96 & 0.25 & 0.07 & \vdots & 24.29 \end{bmatrix}$	$\begin{bmatrix} 0.31 & -0.06 & 0.95 & \vdots & -46.69 \\ 0.22 & 0.97 & -0.02 & \vdots & 0.86 \\ -0.92 & 0.22 & 0.31 & \vdots & 9.2 \end{bmatrix}$
Python	$\begin{bmatrix} 0.29 & -0.01 & 0.96 & \vdots & -47.38 \\ 0.17 & 0.98 & -0.04 & \vdots & 2.15 \\ -0.94 & 0.18 & 0.29 & \vdots & 11.35 \end{bmatrix}$	$\begin{bmatrix} 0.07 & -0.02 & 0.99 & \vdots & -49.49 \\ 0.25 & 0.96 & 0.003 & \vdots & -0.05 \\ -0.96 & 0.25 & 0.08 & \vdots & 24.3 \end{bmatrix}$

Courtesy of Taha 2013