Homework 5 – Camera Calibration -- Due Tue. 23 Nov.

We need to determine the internal camera parameters for a digital camera in order that it can be used later for photogrammetric applications. I have provided six photographs of a target rich scene. You need to examine the photos, select at least 30 well defined points (be careful about mis-identifying due to repetitive patterns, etc.) and measure all of the points occurring on each photo. Note: due to different coverage not every point will be visible on every photo. Each point must fall on at least two photos. Try to select points so that they cover the field of view, that is you want points in the center as well as in the corners (if possible). Try to cover the depth range also, where possible.

You may measure the points in photoshop, and write down point ID, x, y for each point or you may use a matlab script that I provide (imeas.m) to read points and record results in a file. All points for a photo should go into one file, the file name should be ph5194.txt, etc., incorporating the photo ID into the filename.

When you are done collecting all points from all six photos, you should run the bundle block adjustment, first with the fixed (approximate) parameters to make sure that you do not have blunders. Run the fixed solution by copying cam_fix.dat to cam.dat. If your initial solution does not converge, you might want to start with just a few points (5 or 6), get that working, and add a few points at a time to help locate any bad measurements or mis-dentifications. Measurement debug is really the hard part here.

Shown here is the camera that we are calibrating, a Canon XT, with prime (fixed focal length – i.e. non-zoom) lens, nominal focal length = 28mm. Note that the focus ring is set at infinity and the auto-focus is turned off. We want to have the lens in a fixed and repeatable configuration.

The sensor (CMOS) size is 3456 x 2304 pixels, physical dimensions 22.2 mm x 14.8 mm, this is from documentation that came with the camera, if your camera does not provide these physical dimensions, then you need to be a detective and snoop around to fine it. The dimensions only need to be approximate. Using that relationship between pixels and length, we can convert the nominal focal length to pixels: f = 4359 pixels. You can work in either pixels or millimeters, you just have to be consistent. The given files are set up to use pixels.

For the calibration photos we want convergent images with some “kappa diversity”. I visited two floors of a parking garage and took photos of the MSEE building at Purdue. Since the calibration axes are fixed to the camera, if you rotate an image for convenient viewing and measuring, you need to get the x,y back into the landscape orientation before adjustment and calibration (I provide rotpho.m to help with this). Shown here are thumbnails of the photos.

5198

5196

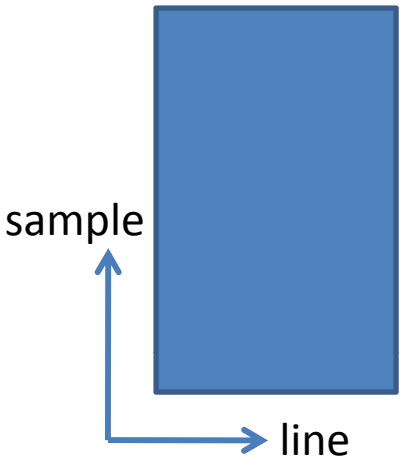
5194

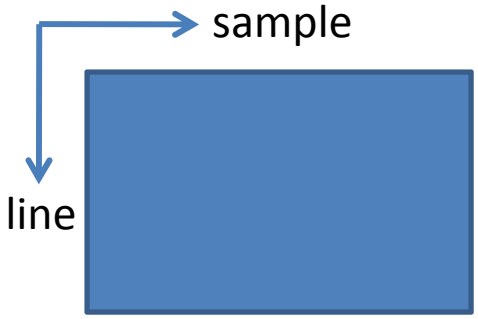
5203

5201

5199

When the camera was rotated for kappa diversity, the “line” runs from left to right, and the “sample” runs from bottom to top, is it clear?





pba_sc2: photogrammetric block adjustment with self calibration
file guide

pba_sc2.m	main matlab script
collin.m	matlab function to evaluate collinearity
gencof.m	matlab function to make numerical partials
int_leq2.m	matlab function to compute n-ray intersection
gndx.m	matlab function to get point index from id
---	---
rotpho.m	matlab script to rotate image coordinates from portrait back to landscape mode
imeas.m	matlab script to enable multiple point measurement and recording in file
---	---
cam.dat	internal camera parameters & sigmas
phofiles.dat	list of files of image measurements
sig.dat	sigmas for point meas & pass points
delta.dat	deltas for numerical derivatives
pho.dat	orientation & exposure station for each photo in the block & sigmas
phxxxx.dat	individual image measurement files you will create these, ph5194.dat, ph5196.dat,ph5198.dat,ph5199.dat, ph5201.dat,ph5203.dat
---	---
cp_save.dat	can be use for control points (we do not use)
cam_free.dat	version of cam.dat to estimate internal camera parameters
cam_fix.dat	version of cam.dat to fix internal camera parameters
pho_constr.dat	version of the pho.dat with constraints at three photo stations
pho_unconstr.dat	version of the pho.dat with the exterior orientation parameters free
ph50.dat	example for format of an image measurement file

Annotated version of cam.dat -- specify fix or free by the magnitude of a priori weights

0.0	0.001	x0 and sigma (pixels)
0.0	0.001	y0 and sigma (pixels)
4359.00	0.001	f and sigma (pixels)
0.00000	0.0001	k1 and sigma
0.00000	0.0001	k2 and sigma
0.00000	0.0001	p1 and sigma
0.00000	0.0001	p2 and sigma
2		photo coord option, 2=photoshop x,y
3456		image width (pixels)
2304		image height (pixels)

The first seven entries here x0, y0, f, K1, K2, P1, P2 are the camera calibration parameters that we are looking for. In later applications we will hold these fixed. K's are radial lens distortion, P's are decentering distortion.

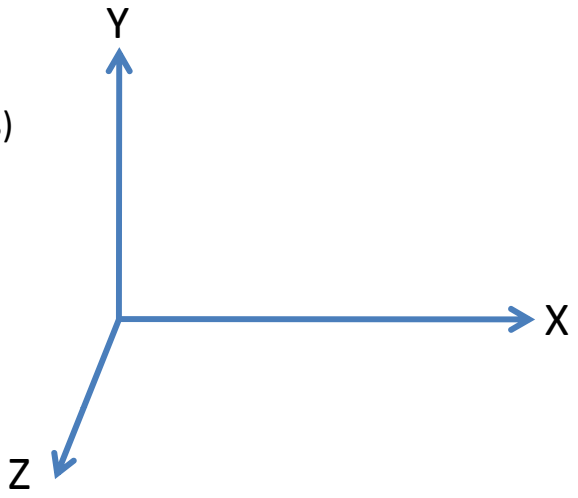
Annotated version of phofiles.dat -- this tells where the image measurements are found

ph5194.txt	1st image measurement file
ph5196.txt	2nd image measurement file
ph5198.txt	3rd image measurement file
ph5199.txt	4th image measurement file
ph5201.txt	5th image measurement file
ph5203.txt	6th image measurement file

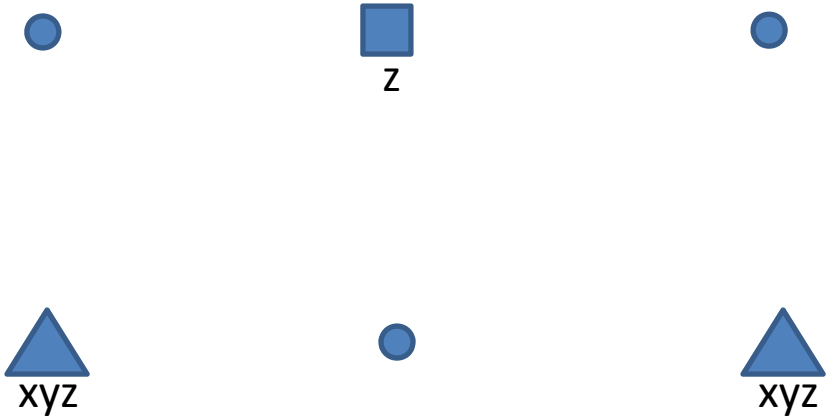
```
ph5194
-0.271767     0.342255     1.669413
1.00000e+008 1.00000e+008 1.00000e+008
67.394       14.684       152.575
1.00000e+008 1.00000e+008 1.00000e+008
ph5196
-0.300701    -0.099007     1.565673
1.00e+008  1.00e+008  1.00e+008
49.183       14.689       152.414
1.00e+008  1.00e+008  1.00e-003
ph5198
-0.290296    -0.371191     1.500170
1.00000e+008 1.00000e+008 1.00000e+008
29.262       14.510       151.816
1.00000e+008 1.00000e+008 1.00000e+008
ph5199
-0.107810     0.341194     0.013918
1.00e+008 1.00e+008 1.00e+008
67.190       5.961        152.431
1.00e-003  1.0e-003  1.00e-003
ph5201
-0.062686    -0.060222    -0.007330
1.00e+008  1.00e+008 1.0e+008
49.153       5.997        152.173
1.00e+008  1.00e+008 1.0e+008
ph5203
-0.038942    -0.581643    -0.007424
1.00e+008  1.00e+008  1.00e+008
29.475       6.144        152.138
1.00e-003  1.00e-003  1.00e-003
```

photo ID
omega, phi, kappa (radians)
prior sigmas
XL,YL,ZL (object units)
prior sigmas
photo ID
omega, phi, kappa (radians)
prior sigmas
XL,YL,ZL (object units)
prior sigmas
etc.
etc.
etc.
etc.
etc.

Annotated version of pho.dat



Note the pattern of fixed and free photo stations -- it is a set of minimal constraints



1 350 1837	point ID x y
2 1232 1857	point ID x y
3 1865 1865	point ID x y
4 2496 1865	point ID x y
5 3096 1860	point ID x y
6 363 1579	point ID x y
7 1237 1595	etc.
8 1868 1604	etc.
9 2496 1607	etc.
10 3091 1609	etc.
11 375 1325	etc.
12 1246 1335	
13 1875 1340	
14 2491 1348	
15 3080 1355	
16 390 1073	
17 1254 1078	
18 1873 1083	
19 2485 1093	
20 3071 1105	
21 414 755	
22 1271 782	
23 1879 790	
24 2485 803	
25 3059 820	

Annotated version of a photo measurement file phxxxx.dat

STEPS

•Copy all files into one folder
•Select points
•Measure each photo for all selected points that fall on that photo
•Run pba_sc2 script using fixed internal camera parameters to check measurement quality (your residuals should not be larger than approx. 10 pixels)
•Run pba_sc2 script using free internal camera parameters, note the estimated values for internal parameters at the end of the listing
•Capture final output using “diary” command
•Submit final listing, and all photo measurement files, and some graphic showing point selection

If you have trouble with the first adjustment step – suggest running with few points, then add others a few at a time. You will need to be conversant with notepad or other text editor to do this.

You are welcome to do this with your own camera, but getting approximate coordinates and orientations for the photo stations is a challenge – I am working on some algorithms and software to get those for you, but it is not ready to go. It will be based on multiple runs of the the linear, 8-point algorithm for relative orientation, between successive pairs of photos, until we have all photos connected. Also getting physical sensor size may be a challenge.