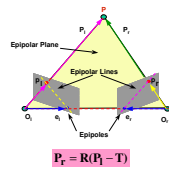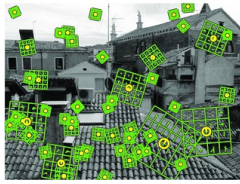# INF5300 – Lab: Geometry from image sequences, repetition

This lab will use MATLAB to implement and explore geometry estimation from feature matching in images.

Geometry from image sequences

Install VLFeat for Matlab: http://www.vlfeat.org/install-matlab.html

Follow the simple tutorial for matching features in two images of your own choice:
http://www.vlfeat.org/applications/sift-mosaic-code.html

For reference: http://www.vlfeat.org/overview/sift.html

Generate a rudimentary depth map from the matched features following this approach:

- Locate matched features in the two camera views
- Find the homography between images to define the transform that aligns the features and use this to locate the camera relative
- Calculate depth in one of the images for each of the features by triangulation (Szelizki, Chap 7.1)
- Interpolate between these samples to make an approximate depth map

The above may not give good results. Since we assume that we do not have the camera internal matrices, exact 3D information cannot be found. One approach would be the so called "fundamental matrix" defining a transform between views, and used to give you a relative estimate of depth. Follow the code from
http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/Robust/example/index.html and combine with the SIFT features to calculate the fundamental matrix in your views (Szeliski, Chap 7.2). This matrix can be used to estimate depths in one of the views by setting that views camera matrix to $P = [I \mid 0]$ and triangulate.

http://vision.middlebury.edu/mview/data/ has some multiviews that may be used for depth reconstruction. Or you could use images from your webcamera considering that you should shift mostly translational, and not too far.

Some useful links:
Extensions to Image Processing Toolbox: http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html
Useful image processing functions: http://www.csse.uwa.edu.au/~pk/research/matlabfns/
A set of Matlab functions for 3D computer vision: http://www.daesik80.com/matlabfns/matlabfns.htm
Matlab code for multiview geometry (poorly documented): http://www.robots.ox.ac.uk/~vgg/hzbook/code/