Assignment A11: Range Data

CS 5320/6320
Spring 2016

Assigned: 23 March 2016

Due: 27 April 2016

For this problem, handin a lab report A11.pdf (include name, date, assignment and class number in pdf) which develops and studies range data analysis. Given the synthetic range data image (ht_im.mat in the class data/A11 directory) and its corresponding ground truth data defining the topo class at each point, study the impact of Gaussian noise in the range value on the topo classification.

CS6320 Apply the topo function to the actual range image found in scene1.mat (use scene1.depthImage) in the class data/A11 directory.

You should handin the report A11.pdf as well as the source code developed in the study. The code should conform to the style requested in the class materials.

In addition, please turn in all files before the start of class time (1:25pm) on April 27, 2016. Write a lab report in the format (please do not deviate from this format!) described in the course materials.

```
function points_im = CS5320_range2pts(range_im)
% CS5320_range2pts - convert range image to full x,y,z points image
% On input:
%     range_im (mxnxd array): range image (d is either 1 or 3)
% On output:
%     points_im (mxnx3 array): x,y,z points in image format
% Call:
%     ht_im_pts = CS5320_range2pts(ht_im);
% Author:
%     <Your Name>
```

```
%     UU
%     Spring 2016
%

function normals = CS5320_normals(pts_im,k)
% CS5320_normals - compute normals on range data
% On input:
%     im (mxnx3 array): range data image
%     k (int): uses 2k+1 by 2k+1 neighborhood for normal computation
% On output:
%     normals (mxnx3 array): normals at each point
% Call:
%     ht_im_normals = CS5320_normals(ht_im_pts,2);
% Author:
%     <Your Name>
%     UU
%     Spring 2016
%

function planes = CS5320_planes(points,normals,k)
% CS5320_planes - find best plane fit at each pixel
% On input:
%     points (mxnx3 array): x,y,z points
%     normals (mxnx3 array): surface normal at each point
%     k (int): uses 2k+1 by 2k+1 window to fit plane
% On output:
%     planes (mxnm5 array): plane parameters and error of fit
%       channels 1-4: a,b,c,d plane parameters
%       channel 5: mean error of fit in window
% Call:
%     im_pl = CS5320_planes(im_xyz,im_nor,5);
% Author:
%     <Your Name>
%     UU
%     Spring 2016
%

function v = CS5320_extract_data(im,r1,c1,r2,c2,channels)
% CS5320_extract_data - pulls segment of data from image
% On input:
%     im (mxnxp array): p-dimensional array
```

```
%     r1 (int): row value of first pixel
%     c1 (int): col value of first pixel
%     r2 (int): row value of second pixel
%     c2 (int): col value of second pixel
%     channels (1xk vector): channel indexes to extract
% On output:
%     v (qxs array): extracted data from channels on the segment
%       from [r1,c1] to [r2,c2]
%       q is the number of pixels from [r1,c1] to [r2,c2]
%       s is the number of channels extracted
% Call:
%     ht_im_pts = CS5320_range2pts(ht_im);
%     v = CS5320_extract_data(ht_im_pts,51,21,51,27,[1:3]);
% Author:
%     <Your Name>
%     UU
%     Spring 2016
%

function topo = CS5320_topo(points,normals,planes,k)
% CS5320_topo - determine topo classes of range image
% On input:
%     points (mxnx3 array): x, y, and z channel range image
%     normals (mxnm3 array): normals at each point
%     planes (mxnx5 array): plane info at each point (a,b,c,d,err)
%     k (int): uses 2k+1 by 2k+1 window
% On output:
%     topo (mxnx7 array): topo class likelihoods (in range [0,1])
%       channel 1: FLAT
%       channel 2: PEAK
%       channel 3: PIT
%       channel 4: RIDGE
%       channel 5: RAVINE
%       channel 6: HILLSIDE
%       channel 7: JUMP_EDGE
% Call:
%     topo = CS5320_topo(ht_im_pts,ht_im_normals,ht_im_planes,2);
% Author:
%     <Your Name>
%     UU
%     Spring 2016
```

%