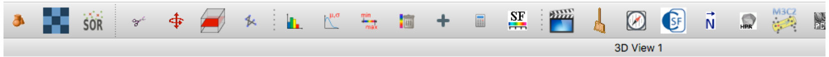
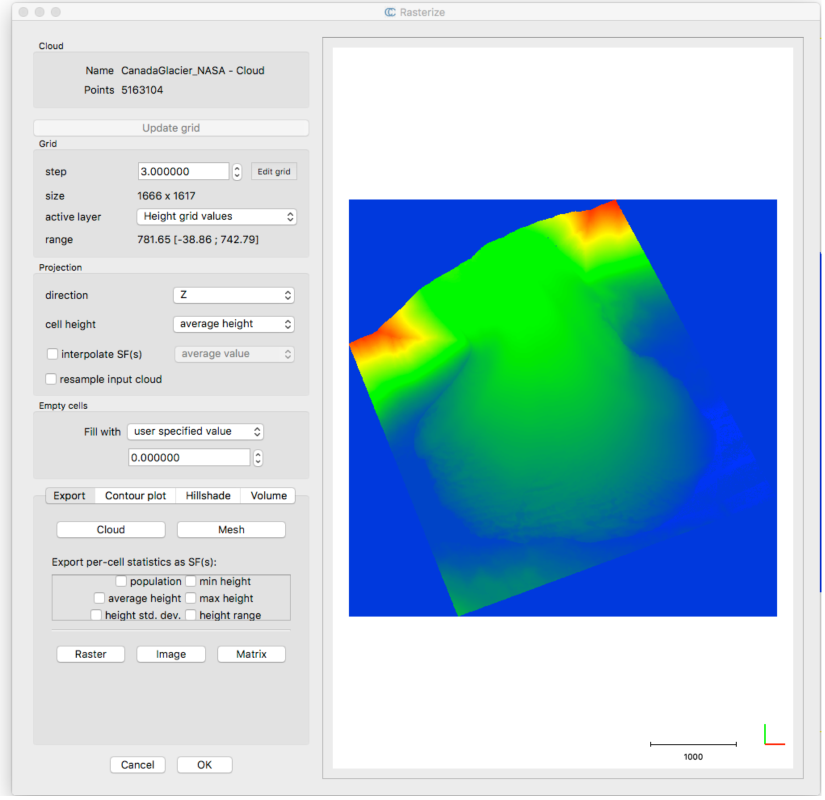
PIVlab for Glacier Motion Tutorial

1. Read Data into CloudCompare for both the NASA and NCALM collections of the Canada Glacier.
   a. Make sure to apply the same relative shift for each of the datasets, so they remain in the same coordinate system.
   b. Export coordinates (Z) to scalar field (Edit -> ScalarFields -> Export Coords to (SF))
   c. Look at point density for each of the datasets. (Tools -> Other -> Density). Use a 5 m radius and surface density. For NASA data, adjust the display and saturation to 0 to 1.5. (Density is between 0.25 and 0.75 pts/m^2)
   d. For NASA data, look at surface roughness, (Tools -> Other -> Roughness) initially using a radius of 5, and then a radius of 10 (Roughness estimation is very simple: for each point, the 'roughness' value is equal to the distance between this point and the best fitting plane computed on its nearest neighbors.) – Note the data noise at the 25 to 50 cm level.
   e. Load in the NCALM data and look at precomputed density and roughness (density is ~ 1 to 5 pts/m2 – discuss why the differences). Note the differences in roughness compared to the NASA data.
2. Create DEM rasters from the lidar point clouds using CloudCompare.

   With point cloud highlighted, select "Checkerboard" icon to pull up the following dialog box.




Rasterize

Cloud
Name CanadaGlacier_NASA - Cloud
Points 5163104

Update grid

Grid
step 3.000000
size 1666 x 1617
active layer Height grid values
range 781.65 [-38.86 ; 742.79]

Projection
direction Z
cell height average height
□ interpolate SF(s) average value
□ resample input cloud

Empty cells
Fill with user specified value
0.000000

Export per-cell statistics as SF(s):
□ population □ min height
□ average height □ max height
□ height std. dev. □ height range

    a. Resolution selection: Choose a resolution approximately equal to the point spacing. You can go smaller and interpolate empty cells and get comparable results. If you go larger, you will eventually blur your features and reduce the ability to see the movement.
    b. Raster Extents: You must use the exact same spatial extents for each raster; otherwise, you will be tracking the difference in the raster extents rather than movement within the raster. (Show the Edit Grid Box)
    c. Once parameters set, select Update Grid, and then export using the Raster button at the bottom of the dialog.

3. Convert rasters to standard 8-bit and 16-bit grayscale format.
    a. PIVlab is built to work on 8-bit grayscale images.
    b. Open PIVLab by typing 'PIVlab_GUI' at command prompt in Matlab.
    c. We then need to read in double precision CloudCompare rasters, and convert them to 8bit (and 16bit).
    d. MATLAB code:

```
% read rasters
nasaRaster = 'nasa3m.tif';
ncalmRaster = 'ncalm3m.tif';
nasaDbl = imread(nasaRaster);
ncalmDbl = imread(ncalmRaster);

% export 8-bit and 16-bit grayscale rasters
nasa3m_8bit = uint8(mat2gray(nasaDbl)*256);
imwrite(nasa3m_8bit, 'nasa3m_8bit.tif');
ncalm3m_8bit = uint8(mat2gray(ncalmDbl)*256);
imwrite(ncalm3m_8bit, 'ncalm3m_8bit.tif');
nasa3m_16bit = uint16(mat2gray(nasaDbl)*65535); imwrite(nasa3m_16bit,
'D:\Preston_PIV_PTV\nasa3m_16bit.tif');
ncalm3m_16bit = uint16(mat2gray(ncalmDbl)*65535); imwrite(ncalm3m_16bit,
'D:\Preston_PIV_PTV\ncalm3m_16bit.tif');
```

4. Get familiar with PIVlab using the 8-bit raster images.
    a. Get familiar with the basic functionality.
        i. Load and import: nasa3m_8bit.tif and ncalm3m_8bit.tif
        ii. ROI: Select an ROI on the lower portion (not the neck) of the glacier
            1. Only areas within the mask will be processed.
        iii. Mask: Demonstrate creating a mask.
            1. Masked portions will not be processed, which speeds things up.
            2. Remove the mask for now.
        iv. Pre-processing Filters: Demonstrate the impact of the different filters.
            1. Note the high-pass filter contour artifacts. Discuss.
            2. Remove all filters for now.
        v. PIV Processing Options:
            1. DCC is direct cross-correlation. It works a bit better than the FFT method in many cases, but is slower.
                a. Interrogation Area: Larger areas produce stronger the solutions, but large areas also act as a low-pass filter on the movement detection.
                b. Step Size: Controls the spacing of the results.

c. Note: Loss of correlation occurs if the window size is too small

2. FFT computes the correlation in the frequency domain. It is faster than DCC. It requires multiples passes, with each pass using a smaller window size to refine the results from the prior pass.

a. Interrogation Area & Step Size: Similar to DCC definitions. The step size of the final pass controls the spacing of the results.

b. Note: As with DCC, if your first pass window size is too small, you will lose correlation.

vi. Post Processing Vector Validation:

1. Select velocity limits: Select velocity limits based on knowledge of glacier movement direction (down and right are positive directions in image space)

2. Interpolate missing data: Looks nice, but use sparingly.

3. Manually reject vector: Sselect the base, not the tip, of the vector.

4. Stdev & Local median filters: Experiment with different numbers to remove outliers.

5. Try to get decent results with the 8-bit rasters inside the ROI and quantify the displacement.

a. Re-run the above steps on the same ROI after applying the CLAHE (Contrast Limited Adaptive Histogram Equalization) and/or the high-pass filter and Re-run with different pre-processing filters

i. Results are better, but probably not great

ii. Note how the contour artifacts do not produce good solutions.

b. Calibrate Distance and Time. We've been purely qualitative thus far. Attach some real numbers to the velocities.

i. Distance: Use "Select reference distance" and click two points. You can compute the "real" distance from the provided pixel distance. Note it is in millimeters.

ii. Time Step: Convert 13 years to milliseconds: 3600*24*365*13*1000 = 4.1 x 10^{11}. For the scientists in the room, that's the number 41 followed by 10 zeros ;-)

c. Look at some Statistics.

i. Plot a histogram of the vector magnitudes. They should be centered around 0.5-0.6 x 10^{-7} m/s, which is about 1.6-1.9 m/year.

6. Attempt to capture the entire glacier motion with the 8-bit data

a. Will not work due to 8-bit contour artifacts.

b. How can we use higher-resolution information (16-bit)?

7. PIVlab with 16-bit images

a. Show some of the oddities that occur when applying ROIs and filters to 16-bit images in PIVlab.

b. Solution is to apply a high-pass filter on the 16-bit rasters and then convert the result to an 8-bit raster prior to importing to PIVlab.

i. Note how a high-pass filter accents roughness while normalizing the elevation trend.

```
% read 16-bit rasters
nasaRaster = 'D:\Preston_PIV_PTV\nasa3m_16bit.tif';
ncalmRaster = 'D:\Preston_PIV_PTV\ncalm3m_16bit.tif';
nasa3m = imread(nasaRaster);
ncalm3m = imread(ncalmRaster);
```

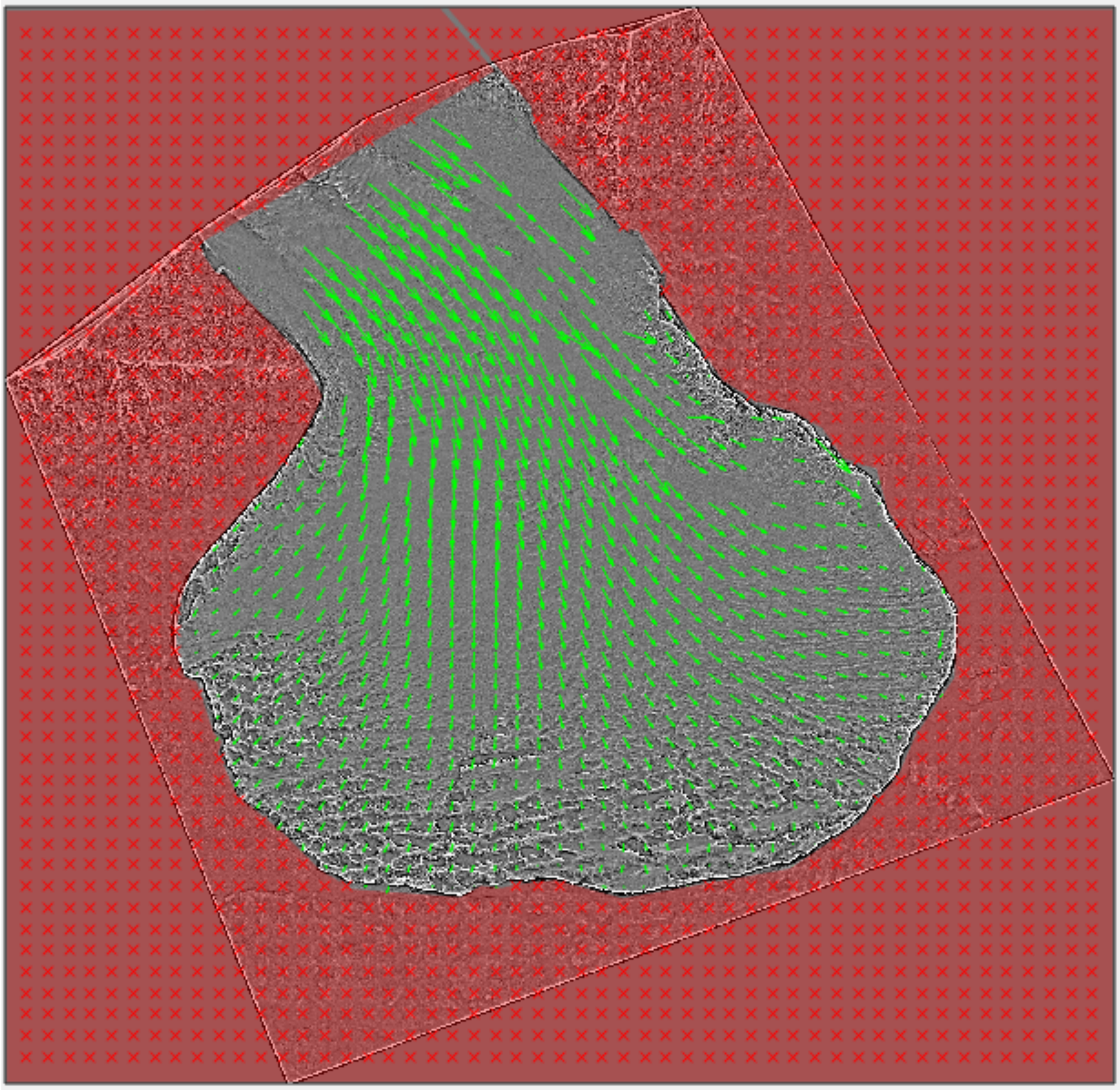
```
% create 8-bit highpass filtered images of the  original 16-bit images
h = fspecial('gaussian',5,5);

nasa3m_hp = double(nasa3m)-(imfilter(double(nasa3m), h, 'replicate'));
low = prctile(nasa3m_hp(:), 1);
high = prctile(nasa3m_hp(:), 99);
nasa3m_hp = uint8(mat2gray(nasa3m_hp, [low high])*255);
imwrite(nasa3m_hp, 'D:\Preston_PIV_PTV\nasa3m_hp.tif');

ncalm3m_hp = double(ncalm3m)-(imfilter(double(ncalm3m), h, 'replicate'));
low = prctile(ncalm3m_hp(:), 1);
high = prctile(ncalm3m_hp(:), 99);
ncalm3m_hp = uint8(mat2gray(ncalm3m_hp, [low high])*255);
imwrite(ncalm3m_hp, 'D:\Preston_PIV_PTV\ncalm3m_hp.tif');
```

c. Run PIVlab on the down-converted, high-passed 16-bit rasters.

  i. Glacier ROI works better.

  ii. Also produces acceptable results for the entire glacier.

  iii. If you have time, create a mask before processing to isolate the glacier movement:



8. Additional Functionality
    a. Plot
    b. Extraction via File>Save

Key Points

- Rasters must cover exact same spatial extents
- Raster size should be equal to the nominal maximum distance between points
- PIVlab is baked for 8-bit grayscale. If you have a large elevation change, this is not good.
- High-pass filtering emphasizes roughness features while normalizing elevation. This is good. But high-passing an 8-bit image with large elevation changes produces contour artifacts. This is not good.
- Best of both worlds is to apply a high-pass filter to a 16-bit (or double precision) raster and save the result as an 8-bit image. Then run through PIVlab.