Machine Learning for Bathymetric Processing

By Andy Hoggarth
Sonar Noise

Types of noise:

• Cavitation/bubble sweep
• Loss of bottom on outer beams
• Loss of bottom lock
• False returns from the water column
Data Cleaning

Time Consuming and labour-intensive

• Current methods:
  • Manual (dot-killing)
  • Simple filters
  • Statistical filters
  • Terrain model binning and filtering
Processing Challenges

Regardless of method or filter type
• Complex features
• And steep slopes

are problematic....
Machine Learning for Bathy

• Machine Learning is Good at:
  • Pattern recognition
  • Prediction

• Requires an extensive library of training data (feature maps)
  • Different data examples
  • Training the algorithm is intensive and requires heavy duty hardware
    o Using the algorithm is not as intensive
    o Cloud based GPU processing is efficient
    o Local PC or Server processing is possible if GPU requirements can be met

• NVIDIA Titan RTX GPU for training the algorithm
• AWS cloud GPU service for day-to-day processing
Convolutional Neural Networks (CNN)

- CNN is very popular and many libraries are available

- Little published work on 3D point clouds
  - Teledyne CARIS is pioneering a 3DCNN for classifying Sonar Noise and more...
  - Could it help MAP THE GAPS?
Sonar Noise Classification Workflow in HIPS

1. Divide dataset into tiles
2. Create a voxel grid for each tile
3. Classify the voxel grid
4. Map the result back to the points
Step 1.

Data is analyzed and divided into tiles

- Based on number of points
  - Low density = big tiles

- and an optional finest vertical resolution
Step 2

A voxel grid is created of all points in each tile

• Get the tile points
• Create voxel grid
Step 3.

Classify the voxel grid

• Send voxels to AI (cloud or local)
  o 100 times smaller than full density point cloud
• Receive classification from AI
Step 4.

Map the result back to the points

- Adjust confidence threshold if required
- Apply as filter to dataset
- Points falling inside Rejected voxels are Rejected (like CUBE)
New Regular Gridded Surface

Input
Source: All Track Lines

Options
Resolution: Automatic
Gridding Method: ShOest Depth true Position
Filter Data:
Minimum: 0.000000
Maximum: 0.000000

Output
Extent: 417673.18, 5579130.29

Fixed extent
Output Coordinate Reference System: WGS 84 / UTM zone 30N [WGS84] EPSG:32630
Output File: C:/My Testing Data/DepFiles/SINC/bin/Classified and Filtered 50%.csar

OK  Cancel  Help
Performance / Accuracy

- > 97% “real” points retained
- ~95% noisy points classified
Reducing Processing Times

Shallow Survey 2015 Demo Dataset

- Collected using Teledyne Seabat T20P
- Approx 175M soundings
- 44 line km of survey
- Roughly 9 ½ hours online survey time
Time Trials

• Time trials (N=1, HH:MM)

<table>
<thead>
<tr>
<th>Workflow</th>
<th>Automatic</th>
<th>Manual</th>
<th>Total</th>
<th>Acquisition/Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual</td>
<td>00:00</td>
<td>05:52</td>
<td>05:52</td>
<td>1h/36m</td>
</tr>
<tr>
<td>Terrain Model Filtering</td>
<td>00:29</td>
<td>02:30</td>
<td>02:59</td>
<td>1h/20m</td>
</tr>
<tr>
<td>AI Noise Classifier</td>
<td>00:20</td>
<td>00:35</td>
<td>00:55</td>
<td>1h/6m</td>
</tr>
</tbody>
</table>
Other CARIS AI Initiatives

- Processing SDB (TCarta)
- Intruder Detection (Blueview)
- Topo Bathy Lidar (Optech)
- Target Detection (Gavia)
Mapping the Gaps

• Option to apply AI when contributing bathy data (e.g. DCDB website)
• Run data through Cloud based AI algorithm for noise removal
  o Providing data processing consistency
• Streamline data integration into regional and / or GEBCO grid
• Potential to use AI to make legacy data consistent
• Develop a “community” to further train the AI algorithm
  o Incentivize
Thank You
Questions?

@hydrohogg