Small Satellite Research Laboratory

Franklin College of Arts and Sciences

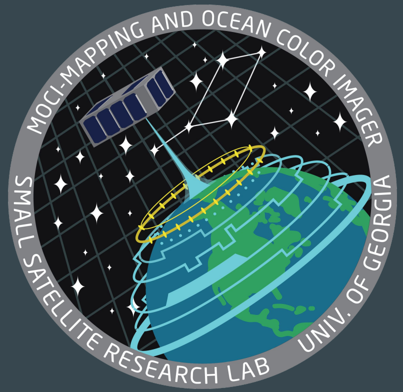
UNIVERSITY OF GEORGIA

Batch Analytical Comparisons of on Orbit Multi-View Stereo, Surface Reconstructions, Rasterization, and Digital Surface Models

Caleb Adams, Nicholas Neel, David Cotten

What is the UGA SSRL?

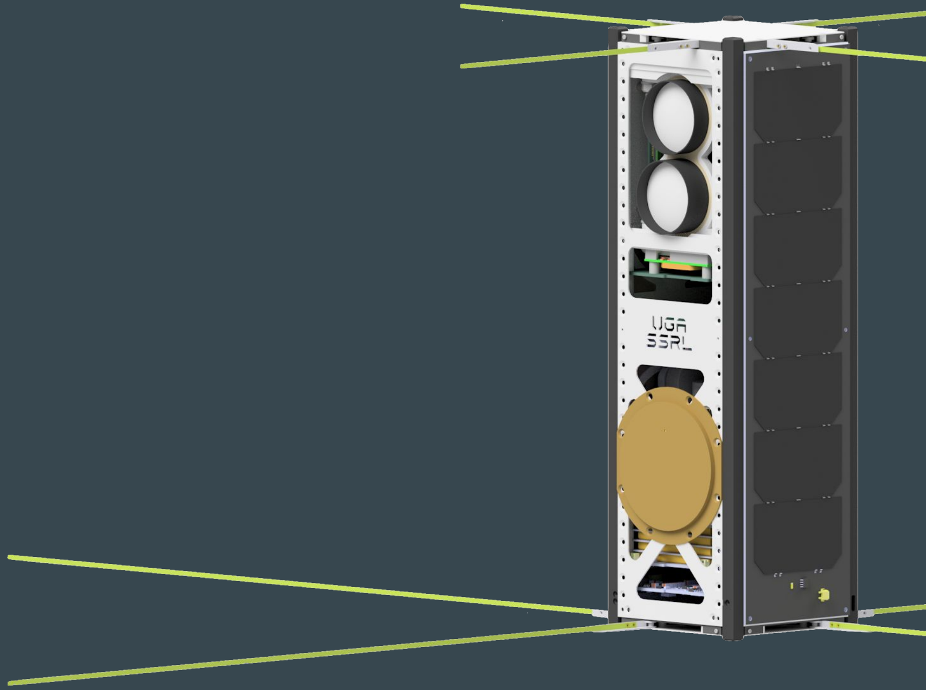
- Student Run
- Student Founded
- Faculty Supported
- NASA and UNP/AFRL funded
- 2 Cube Satellite Missions
- 54 Student Researchers



Small Satellite Research Laboratory
Franklin College of Arts and Sciences
UNIVERSITY OF GEORGIA

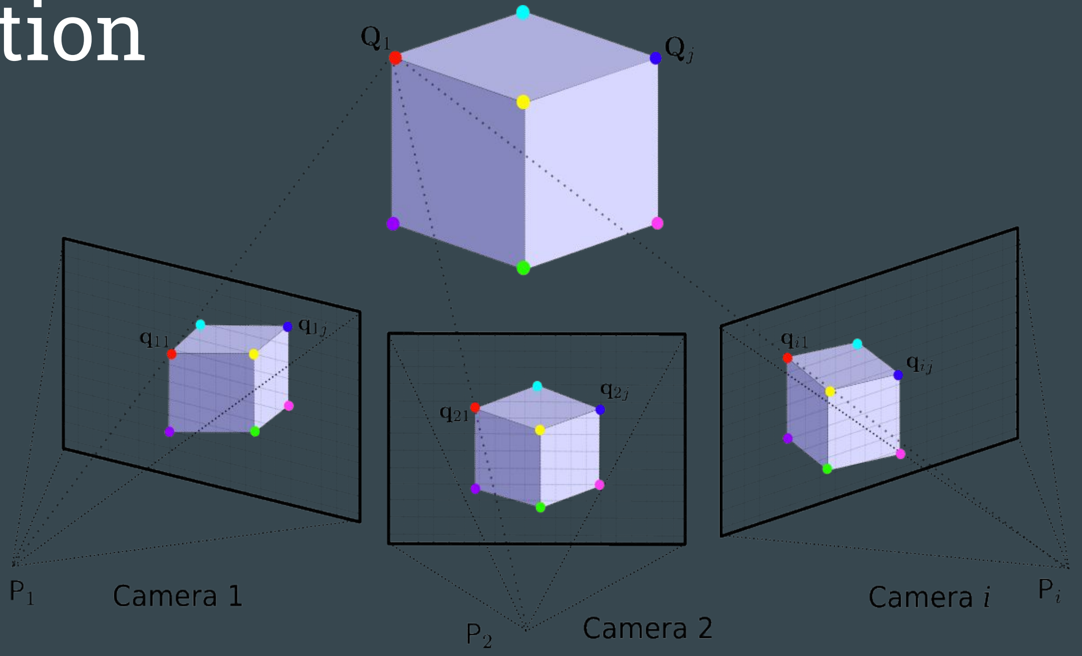
The Mapping and Ocean Color Imager Satellite

- Will produce near real time Digital Surface Models and Digital Elevation Models
- Utilizes & upgrades existing Structure from Motion techniques
    - Typically terrestrial algorithms
    - Typically utilized with UAVs



Structure from Motion

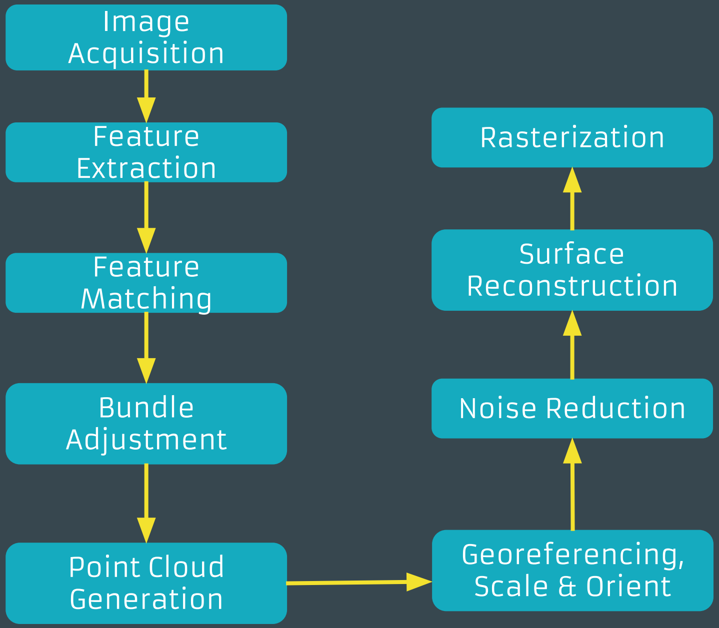
- Input is a set of 2D images, output is a 3D structure
- Generates a point cloud from multiple images from multiple angles
- Sort of like saying “cloud computing” or “Big Data”, it’s really just a buzz word with lots of complicated parts



Courtesy of Julien Michot

Structure from Motion … Our Workflow

- Based roughly off of the RIT workflow
- Each step takes the output of the previous as an input
- Each step is a program

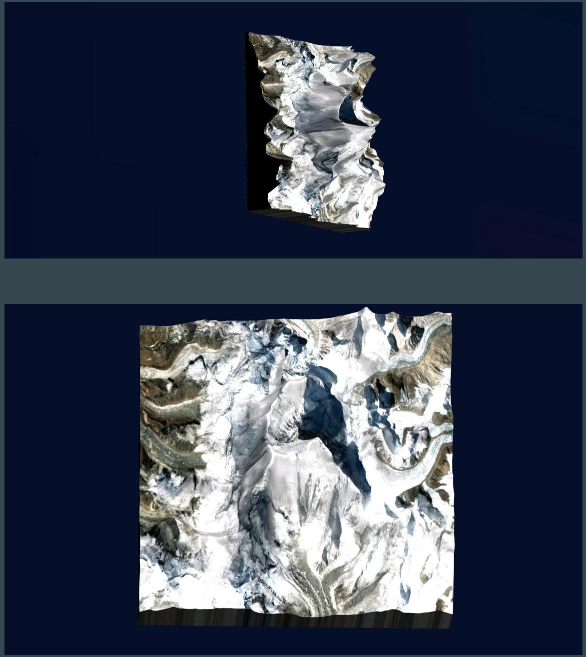


Simulating Realistic Data Acquisition

- Custom program with Blender as rendering engine
- Utilizes GDAL in post processing
- Build a .json config file with test parameters
- Script can be automated to test multiple SfM solutions
- Procedural terrain generation for varying large test sets

```
{
    "LandscapeType": "FILE",
    "objTexture": {
        "Path": "stock/everest.jpg",
        "TYPE": "PROJECTED"
    },
    "LandscapeLatitude": 0.0,
    "SimulationScale": 200.0,
    "LandscapePath": "stock/everest.ply",
    "ImageFormat": [
        "JPEG"
    ],
    "pathImages": "simulations/test/34/iteration-0/images",
    "PhotographCount": 30,
    "SensorFOV": 4.6,
    "SensorCropPercent": 100,
    "OrbitalInclination": 0.0,
    "LandscapeLongitude": 0.0,
    "SensorResX": 4096,
    "PhotographRange": 0.4363323129985824,
    "OrbitalAltitude": 400.0,
    "SensorResY": 3120,
    "SensorFocalLength": 75.0,
    "LandscapeZ": 8.8,
    "LandscapeY": 18.0,
    "LandscapeX": 18.0,
    "OrbitalOrientation": 0.0,
    "RenderOnly": false,
    "LandscapeScale": 1.0
}
```

example .json input data



example visual output of mount everest from the simulation

Simulating SfM

- Takes in the .json as an arg
- Runs through workflow
  - workflow can be stopped at discrete steps
- Outputs:
  - DEM/DSM
  - Feature Set
  - Dense/Sparse Point Cloud
  - GeoTiff (Raster)

```
./main.py config.json
```

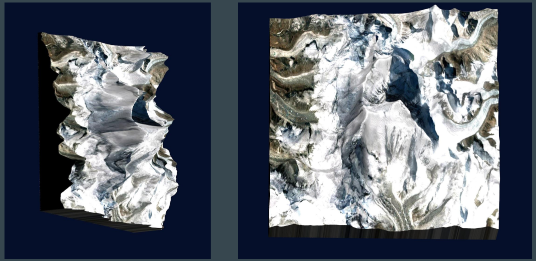
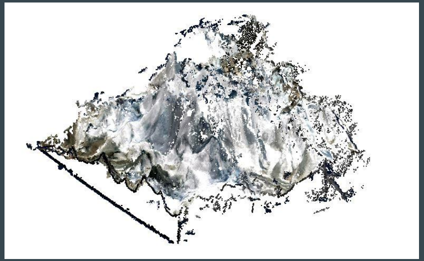
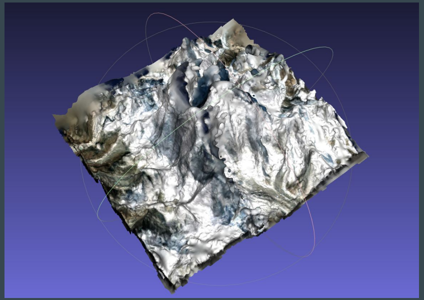


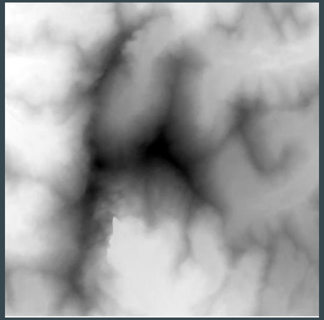
image acquisition



point cloud generation



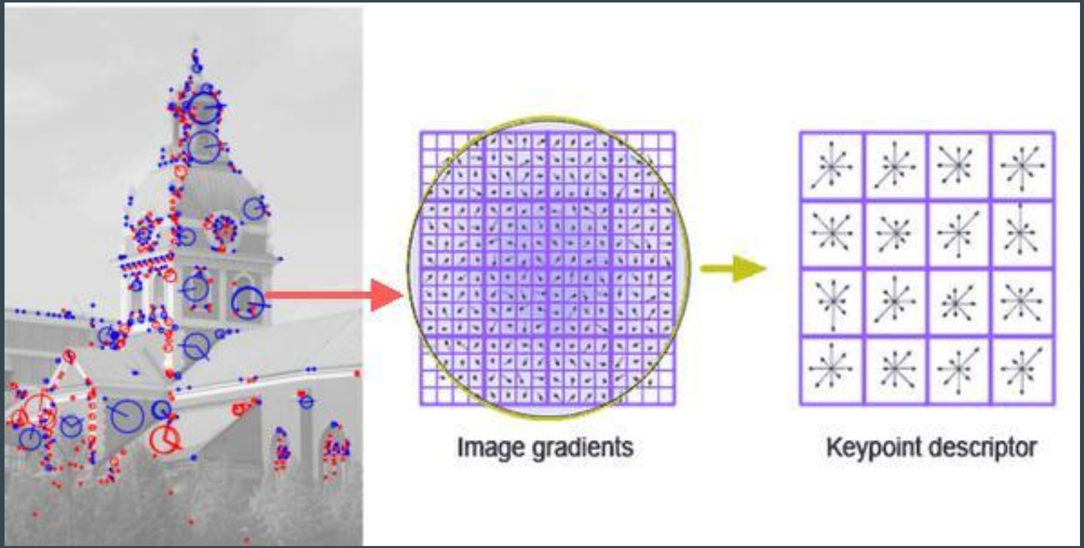
surface reconstruction



GeoTiff Generation (Rasterization)

Simulating SfM - Feature Extraction

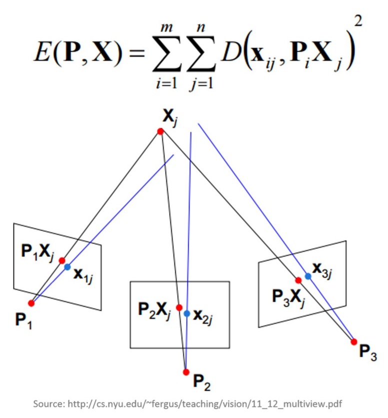
- Scale Invariant Feature Transform (SIFT) Algorithm
  - Implemented on an FPGA
  - Also CUDA implementation
- Features then need to be matched (we're glossing over that step here, may be added in the future)



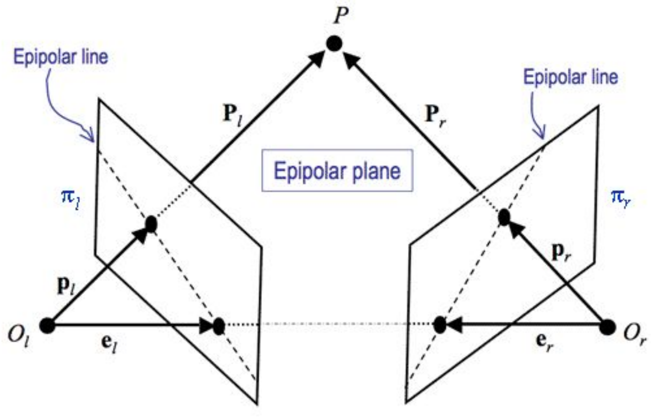
http://www.codeproject.com/KB/recipes/619039/SIFT.JPG

Simulating SfM - Bundle Adjustment

- Addition of the Epipolar Constraint may remove Bundle Adjustment
  - Camera position is known
  - Center of rotation is known
- Given the set of image coordinates x_j^i find the set of camera matrices, P^i, and the points X_j such that $P^iX_j = x_j^i$ This is known as project reconstruction
- With known P^i, the epipolar constraint could be used to make a sparse point cloud generation faster than a pure bundle adjustment approach



bundle adjustment



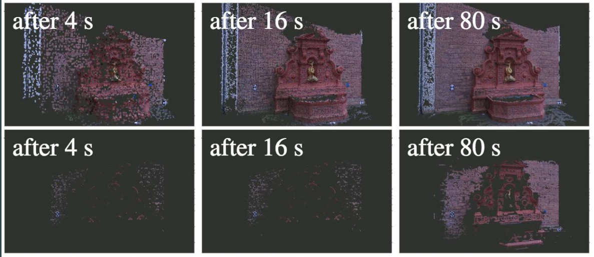
Epipolar constrained reconstruction
https://www.cs.auckland.ac.nz

Small Satellite Research Laboratory
Franklin College of Arts and Sciences
UNIVERSITY OF GEORGIA

Simulating SfM - Point Cloud Reconstruction

- Patch-based Multi-View Stereo (pmvs)
- Hierarchical Progressive Multi-View Stereo (hpmvs)
- Runs a dense reconstruction from the sparse reconstruction
  - after a sparse bundle adjustment
  - after an epipolar constrained reconstruction

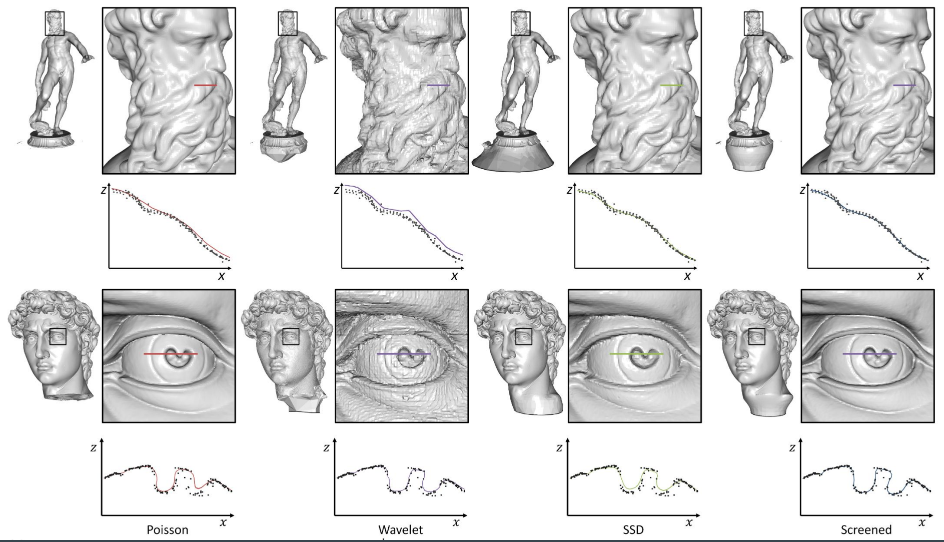
pmvs - https://www.di.ens.fr/pmvs/pmvs-1/images/overflow.jpg



hpmvs - https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Locher_Progressive_Prioritized_Multi-View_CVPR_2016_paper.pdf

Simulating SfM - Surface Reconstruction

- Given a set of oriented points, build a 3D model from those points to approximate the original model
- Screened Poisson Surface Reconstruction
- Texturing post-reconstruction



http://www.cs.jhu.edu/~misha/MyPapers/ToG13.pdf

Initial results ...

- Let us find what to improve
- Improvement from 18m GSD would be needed for finer resolution.
- With a single onboard GPU, compute time was estimated to be 3.3 hours worst case and 0.9 hours best case
- SIFT, tie point generation, and feature detection in general are target areas for improvement/optimization

Table 2: Data sizes for the successful tests in Table 1

Test	Image Count	Image Size	Sparse Cloud	Sparse Size	Dense Cloud	Dense Size	Mesh	Mesh Size	Texture Size	Total Size
Phobos	2	1.97MB	515	67KB	1,051	8.85MB	Fail	N/A	N/A	10.9MB
Pluto Test 1	16	1.63MB	11,953	3.1MB	35,153	47.6MB	41,273	1.6MB	287KB	53.93MB
Pluto Test 2	15	1.6 MB	11,877	3.1MB	39,804	16.1MB	42,866	3.7MB	5MB	29.5MB
ISS	30	50.1MB	11,729	6.9MB	301,249	279MB	146,138	12.6MB	24.5MB	337.1MB
Test 11	100	4.14MB	30	14KB	41	277KB	Fail	N/A	N/A	4.2MB
Test 12	40	1.76MB	20	32KB	Fail	N/A	N/A	N/A	N/A	1.8MB
Test 13	100	10MB	514	542KB	79,674	39MB	66,400	5.7MB	5.2MB	60.5MB
Test 14	40	3MB	547	158KB	50,438	24.1MB	21,379	1.8MB	N/A	29.1MB
Test 15	200	11.1MB	39	150KB	Fail	N/A	N/A	N/A	N/A	11.3MB
Test 16	100	5.94MB	35	102KB	Fail	N/A	N/A	N/A	N/A	6.1MB
Test 17	80	5.33MB	241	216KB	11,203	11.5MB	Fail	N/A	N/A	17.1MB
Test 18	100	10.1MB	672	363KB	87,156	58.8MB	42,389	3.7MB	789KB	73.8MB
Test 19	80	7.9MB	670	334KB	113,508	57.9MB	49,484	1.8MB	N/A	67.9MB
Test 20	60	5.39MB	1,208	450KB	107,852	47.8MB	45,312	3.9MB	667KB	58.3MB
Test 21	40	3.06MB	329	108KB	20,204	15.3MB	16,717	612KB	7KB	19.1MB
Test 22	32	2.38MB	34	60KB	4,956	3.86MB	Fail	N/A	N/A	6.3MB

(4612/325) * (60) * (14) = ~11,920.25 seconds, or 198 minutes or 3.3 hours

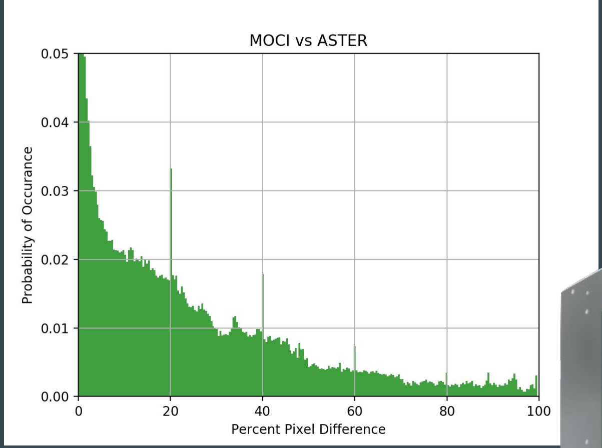
(4612/325) * (60) * (4) = ~3,405.78 seconds or 57 minutes or 0.9 hours

SSRL generated test data

*Uses eSOM TK1 and not Jetson TX1 or TX2, TX models operate around 6-14 minutes

Current Results

- After 500+ tests…
- Testing with a custom 6.4 m GSD camera
- Comparing DSM/DEM's with ASTER data (15m GSD data)
- Allows for better reconstructions!
- Simulations defined, finalized, confirmed our hardware!
  - Custom 6.4m GSD camera
  - Integrated FPGA (Opal Kelly XEM7310)
  - Integrated GPU (TX2)

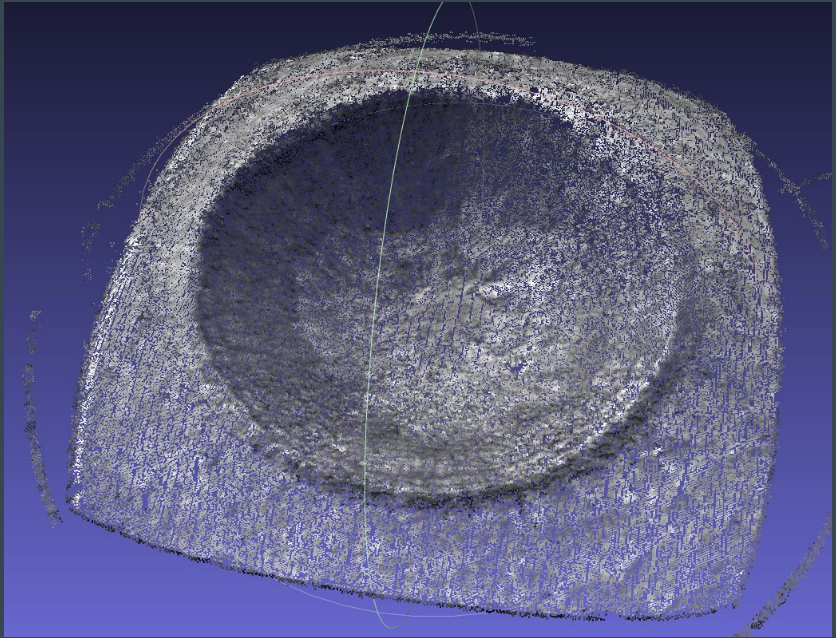


MOCI's accuracy relative to ASTER data

The MOCI 1U bionicle payload

Optimization & Future Plans

- Now that we have proven feasibility…
  - Test with other planets!
- Use constrained geometry advantageously
- AI & neural nets to get better workflows
- Inserting custom programs into our workflow



Dense point cloud of Lenne crater on the moon, using MOCI & simulations

Questions?

smallsat.uga.edu
University of Georgia