Open Source Software for Deriving Meshes from Point Clouds with Error Propagation

Funded by Phase 2 SBIR NGA183-002
NGA PMs: Chris Clasen, Yunting Su

P3DL
August 10th, 2021

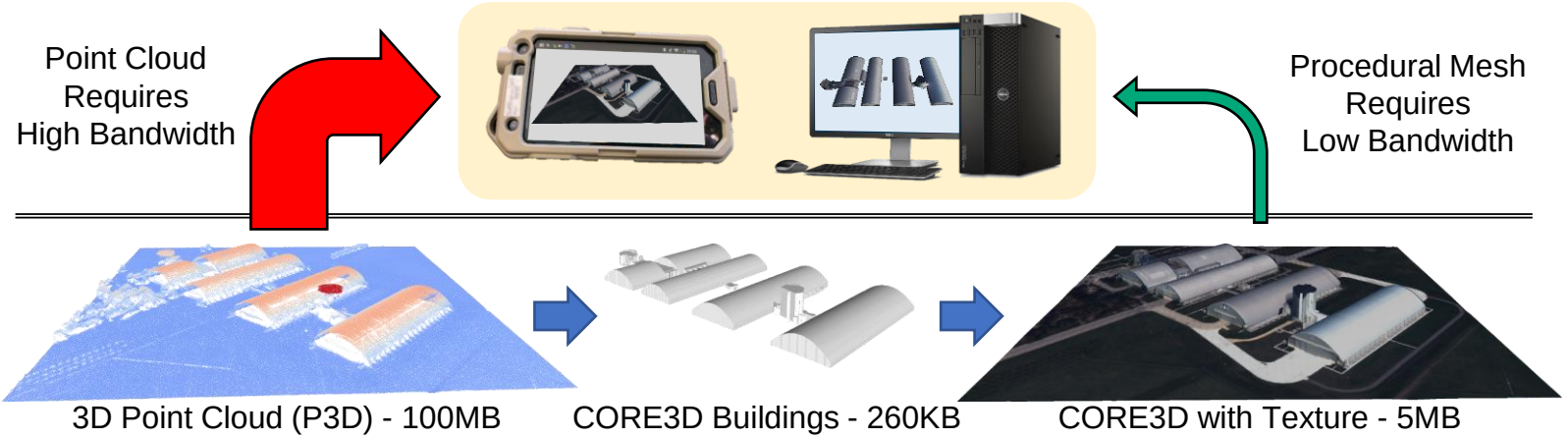
Dr. Matt Leotta

Supported by: Dan Lipsa, Jacob Nesbitt, Chet Nieter, Eugene Borovikov

Problem Summary

- Need to support 3D geospatial data transmission for web visualization over low bandwidth connections
- Point clouds require high bandwidth, textured mesh models are more efficient
- How do we transform point clouds into geospatial meshes? ...and propagate error/uncertainty
- Which standards are best for storage, transmission, and visualization of these meshes?

New addition



Phase I Conclusions

Phase I compared OGC standards for 3D web transmission

- Study focused largely on comparing CDB to 3D Tiles
- Presented detail results at P3DL last year
- Conclusions
  - **3D Tiles** is more efficient than CDB given best practices
    - CDB is not optimized for web transmission
    - Either CDB or 3D Tiles can be significantly slower if not tuned for optimal tile size, minimizing server requests, etc.
  - OGC standards are open, Cesium's viewer is open source but…
  - There is a **lack of non-proprietary tools for encoding 3D Tiles**
- Goal: Provide an end-to-end open source solution

OGC® Making location count.

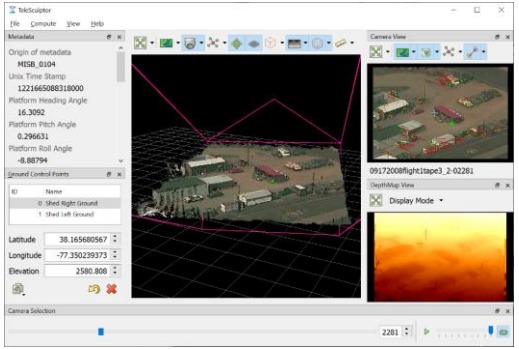
CDB

VS

CESIUM 3DTiles

Relevant Kitware Open Source Tools to Leverage

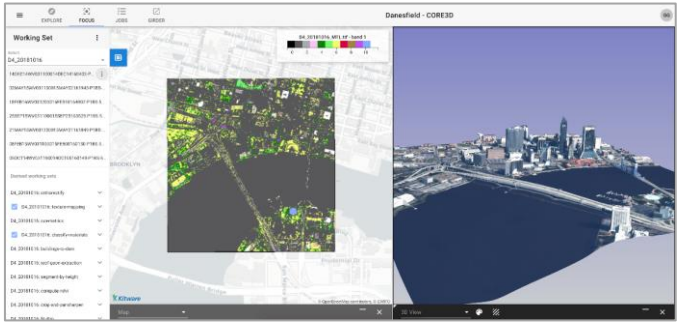
TeleSculptor / KWIVER



https://telesculptor.org/

3D from FMV
Desktop app and toolkit

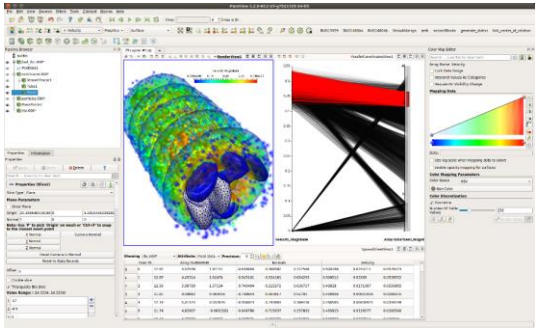
Danesfield App / Danesfield



https://github.com/kitware/danesfield

Semantic 3D from satellite imagery
Web app and toolkit

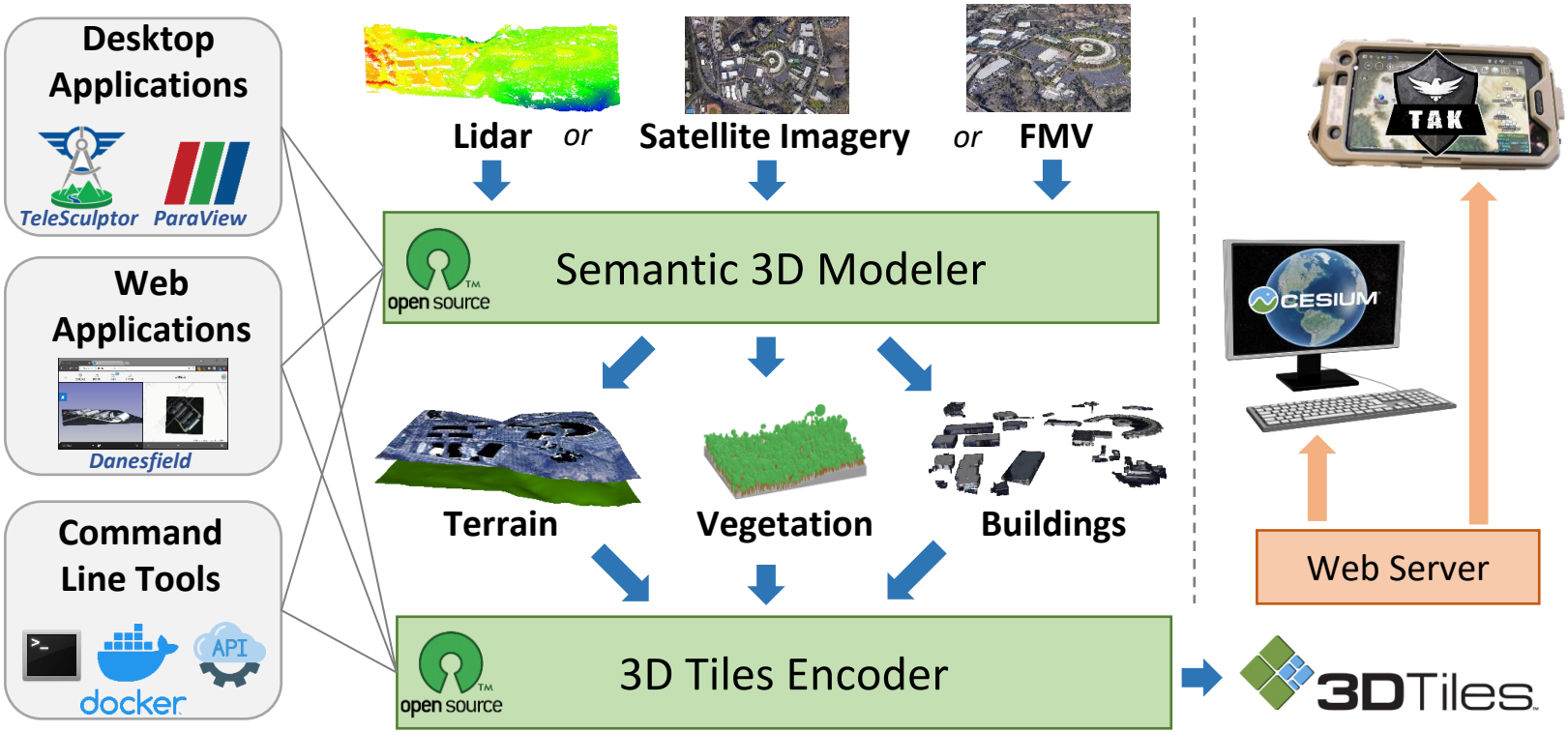
ParaView / VTK



https://www.paraview.org/

3D analysis and visualization
Desktop and web apps and toolkit

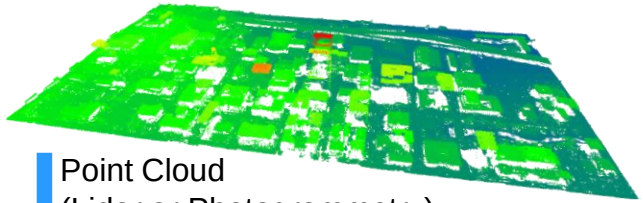
Phase II Approach



Semantic 3D Modeling

- Build on work started on IARPA CORE3D,
- Implemented in Danesfield
  https://github.com/Kitware/Danesfield/
- Primary steps
  - Semantic segmentation into layers, e.g.
    - Ground terrain
    - Buildings
    - Vegetation
  - Converting point clouds to simplified meshes
  - Constructing texture maps
- Need to adapt this work from satellite (P3D) to Lidar, FMV, other point clouds

Imagery (Satellite or Airborne)



Point Cloud
(Lidar or Photogrammetry)

Semantic 3D Meshes

Textured 3D Meshes

Progress to Report

- Comparison of alternate of point cloud sources
  - VisSat (Cornell) open source alternative to P3D (Raytheon)
  - Direct point cloud processing (no images)
- Open source 3D Tiles conversion
- Preliminary work on GPM error propagation from points to meshes

Danesfield with Other Point Cloud Sources

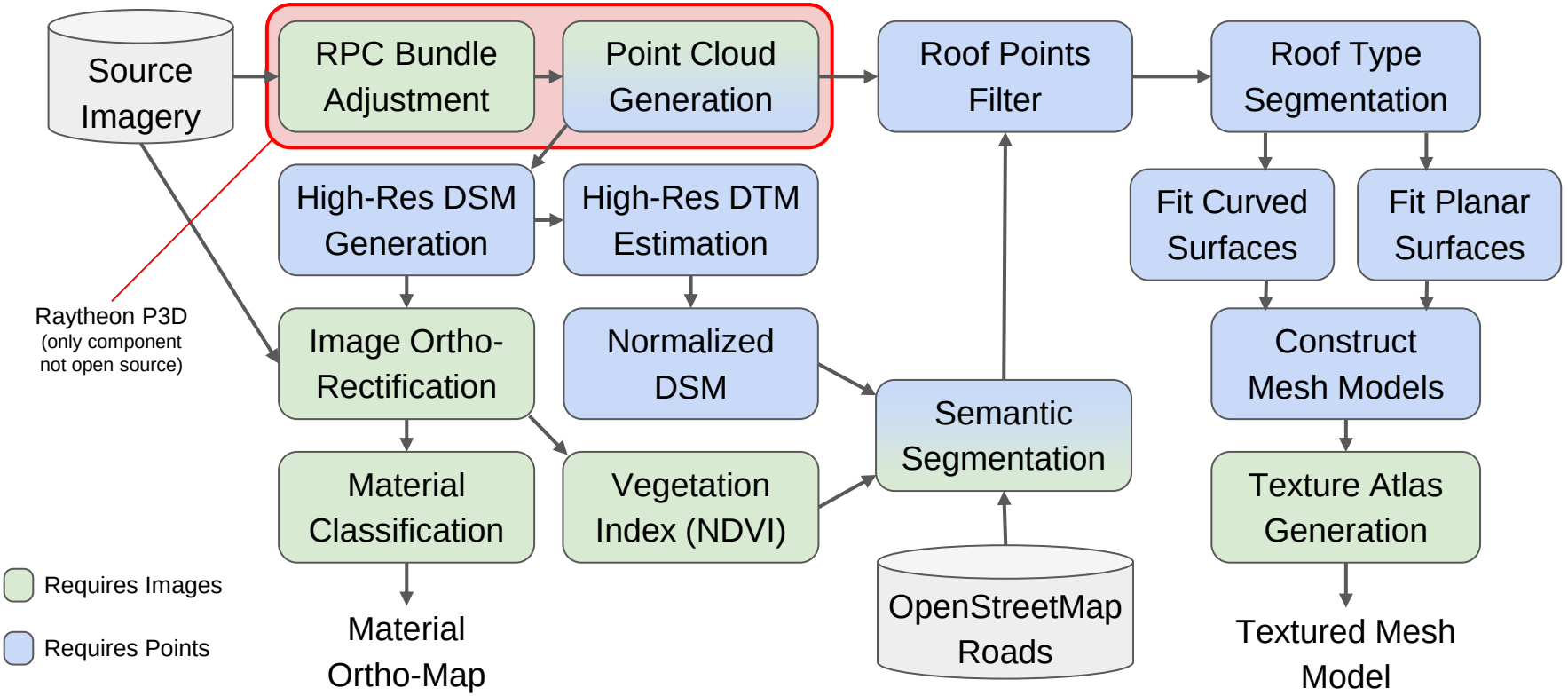
Danesfield, as developed for CORE3D

- Converts satellite images to texture meshes
- Uses Raytheon P3D to convert satellite images to point clouds
- Is fully open source except for P3D

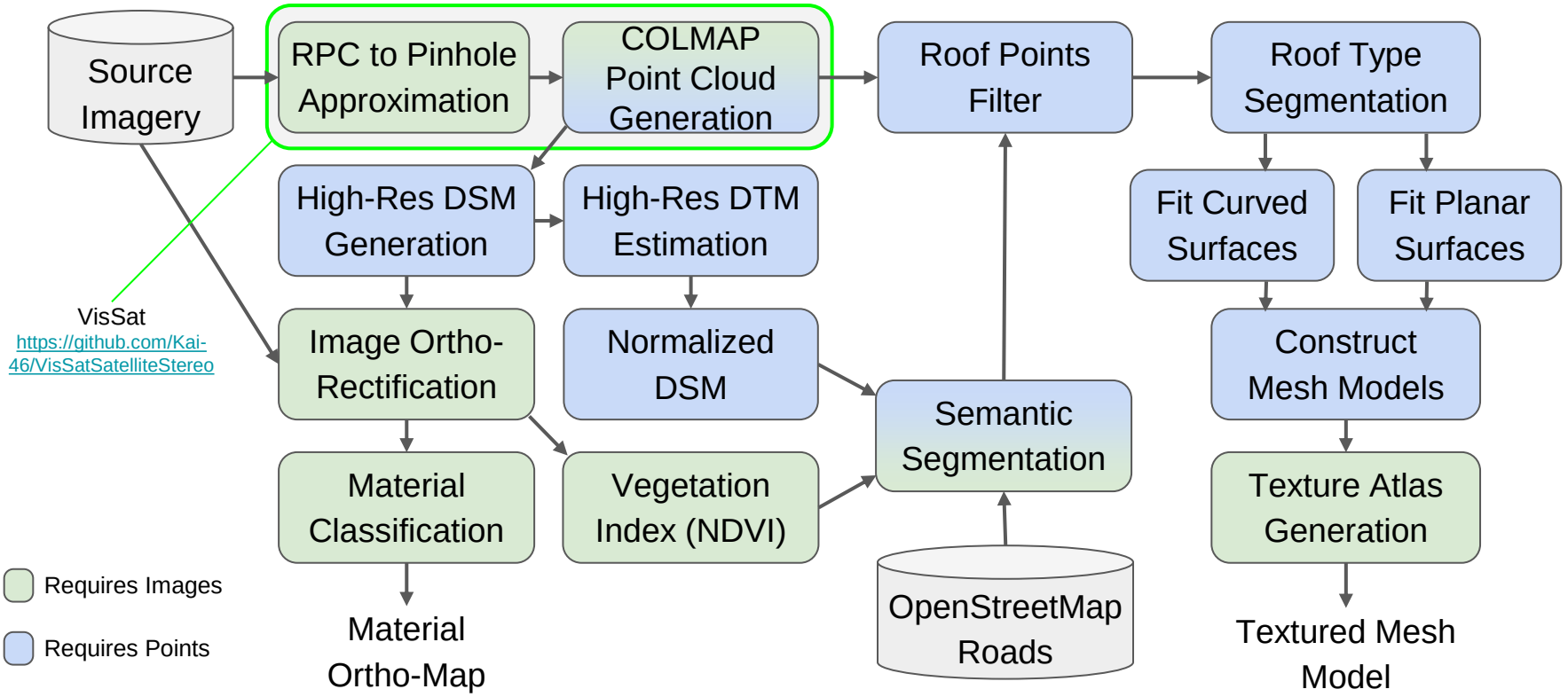
Goals:

- Fully open source alternative for P3D in the Danesfield pipeline
- Option to process existing point clouds without images (e.g. Lidar)

Original Danesfield System Pipeline (CORE3D)

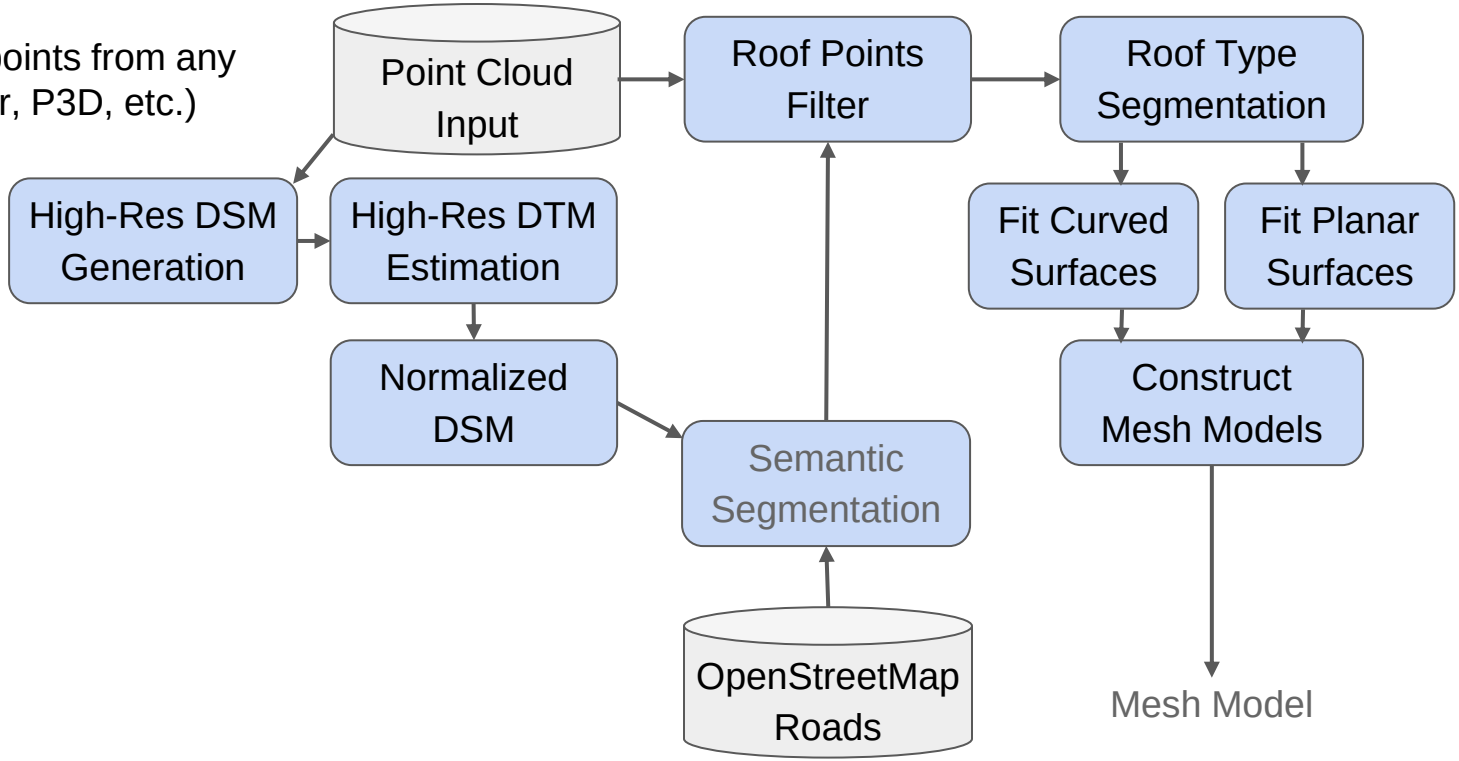


Open Source Danesfield System Pipeline



Danesfield Starting with Point Clouds

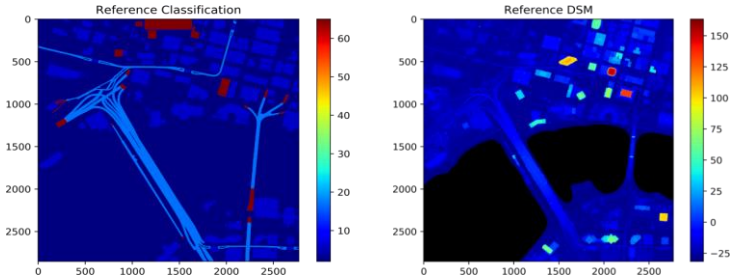
Geospatial points from any source (Lidar, P3D, etc.)



Comparison

Evaluated on Jacksonville, FL data from CORE3D dataset using CORE3D metrics

- P3D has better correctness (precision)
- VisSat hs better completeness (recall)
- Similar F-scores



CORE3D Ground Truth Data
https://spacenet.ai/core3d/

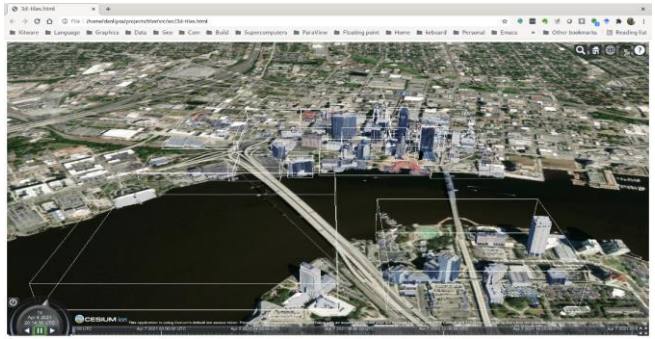
	Using Images	Without Images
P3D	P 0.91 R 0.75 F 0.82	P **0.92** R 0.73 F 0.82
VisSat	P 0.82 R **0.87** F **0.85**	P 0.79 R **0.87** F 0.83

CORE3D Metrics Comparison

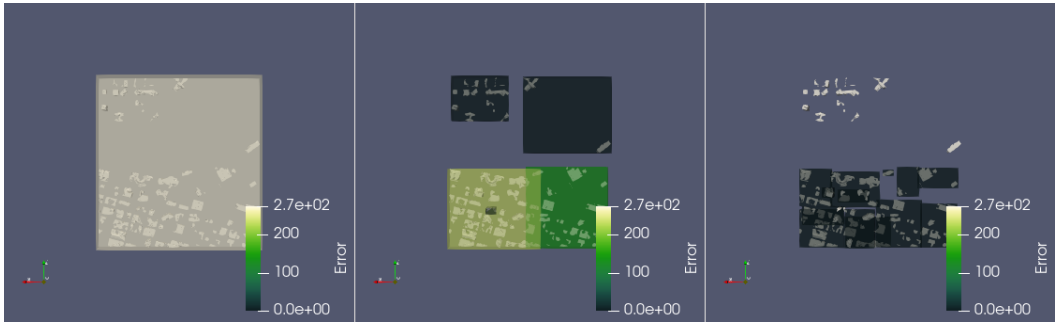
Metric	P3D	P3D no images	VisSat	VisSat no images
2D Correctness	**0.90**	**0.91**	0.72	0.65
2D Completeness	0.71	0.70	**0.87**	**0.87**
2D F-Score	**0.79**	**0.79**	**0.79**	0.74
3D Correctness	**0.91**	**0.92**	0.82	0.79
3D Completeness	0.75	0.73	**0.87**	**0.87**
3D F-Score	0.82	0.82	**0.85**	0.83
Geolocation Error (m)	**2.24**	2.40	2.71	2.75
Z-RMSE (m)	0.60	0.63	0.59	**0.54**
H-RMSE (m)	2.06	2.00	**1.80**	**1.80**

3D Tiles Conversion

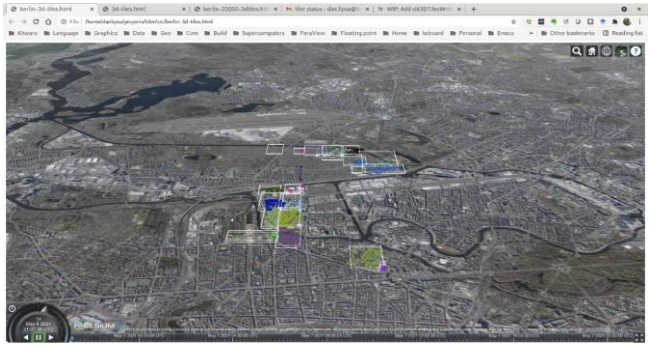
- Implemented initial 3D Tiles writer in VTK
  - Builds octree using building centers
  - External scripts for gltf -> glb -> b3dm conversion
  - Additional work needed on tile optimization
- Integrated 3D Tiles conversion in Danesfield



Jacksonville: 20 buildings per tile



Jacksonville: Each octree level, colored by geometric error

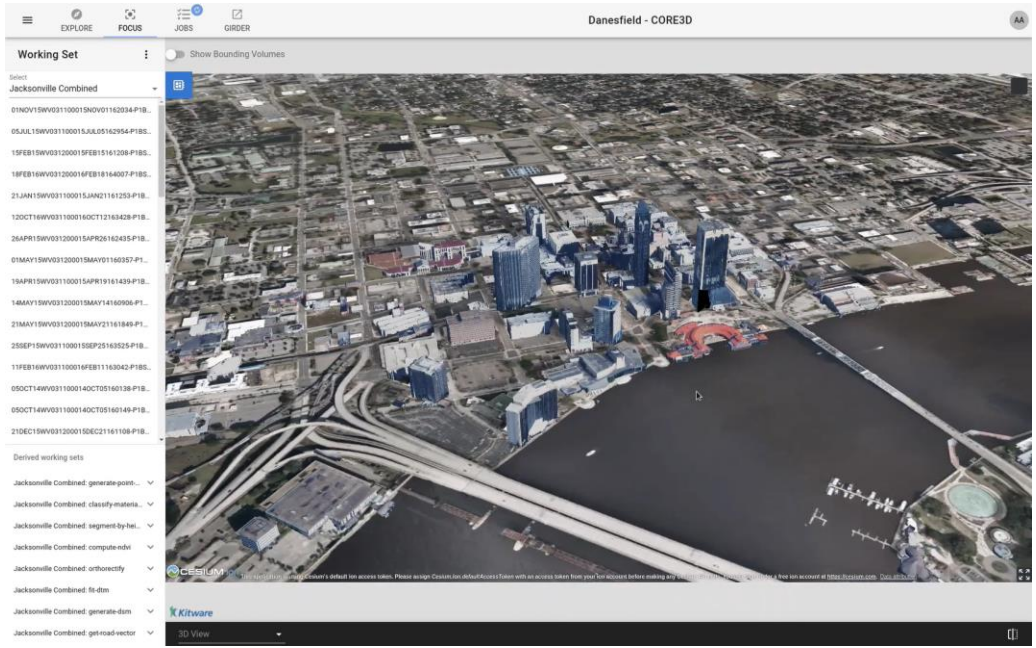


Berlin: 800 buildings

Danesfield Web App with 3D Tiles

Integrated 3D Tiles conversion into Danesfield backend and Cesium.JS viewer into Danesfield web app

Danesfield Map Output (DSM)



3D Tiles in Danesfield with Cesium.JS

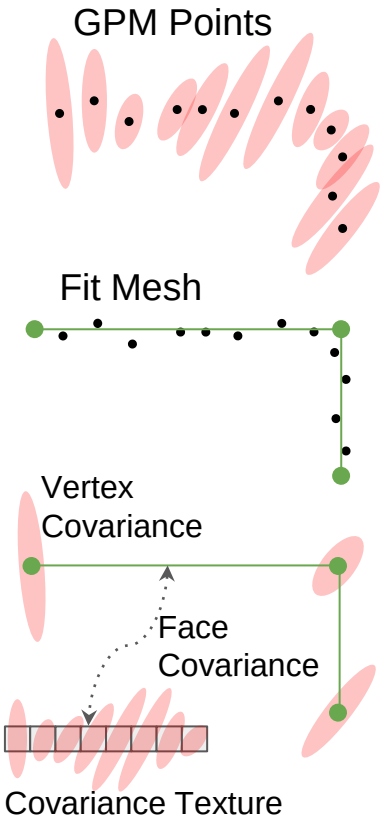
GPM for Mesh: Initial Progress

Goals:

- Adapt the Ground-Space Generic Point-cloud Model (GS GPM) to meshes
- Propagate GS GPM covariance from points to meshes during conversion

Approach:

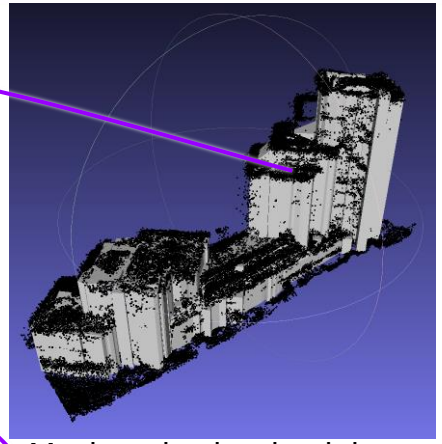
- Meshes are sparse point clouds (vertices) plus faces
- **Represent uncertainty on vertices as in GPM**
- Uncertainty may vary across a mesh face
  - Many surface points map to a single mesh face
  - Analogous problem: encoding color across a mesh face
- **Encode face relative uncertainty in a texture map**
  - Similar to GPM for raster
- Project point uncertainty into onto nearby faces



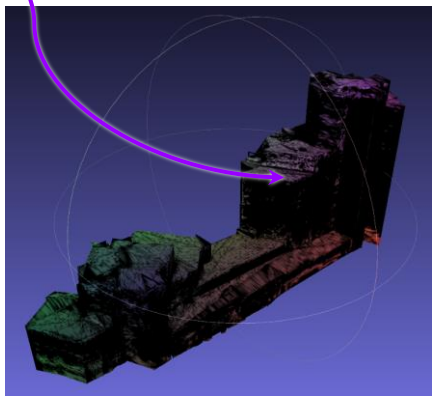
Progress on mapping point clouds

- Implemented functions for
  - Mapping points to nearest location on mesh surface
  - Mapping mesh surface points into texture space
  - Storing point cloud attributes mapped into texture space
- Code contributed to KWIVER, accessible in both TeleSculptor and Danesfield

Texture map
point cloud data
Point (X,Y,Z)
colored as (R,G,B)



Mesh and point cloud data



Mesh with new texture map

Ongoing and Future Work

- Optimize 3D Tiles conversion
  - Mesh compression, additive refinement, texture atlases
  - Code improvements and cleanup
- Support FMV as input source
  - Integrate TeleSculptor algorithms into Danesfield
- Add 3D Tiles output to desktop applications (TeleSculptor and ParaView)
- GPM to mesh error propagation
  - Complete mapping of point cloud attributes to mesh textures
  - Incorporate mesh fitting error into propagation
  - Add covariance estimation to TeleSculptor algorithms

Questions?

Who needs a laser when you've got a video camera? #PassiveSensorShark

Jewell, O.J. et al. "Cryptic habitat use of white sharks in kelp forest revealed by animal-borne video". *Biology letters*, 15(4), 2019.