# So, You Want to 3D Print A Landscape?

**Dylan J. McKevitt[1a], Victoria Couser[1b], Emily Jackson[1c]**

[1]Cedarville University Department of Science and Mathematics, Cedarville, OH

[a] dylanmckevitt@cedarville.edu, [b] vcouser@cedarville.edu, [c] emilyrjackson@cedarville.edu

Printed October, 2015 for Geological Society of America Annual Meeting in Baltimore, MD. Paper No. 289-5, Booth 61.

1-4 NOVEMBER GSA 2015 Baltimore, Maryland, USA

## Abstract

With a variety of both free and costly software available today, it is possible for just about anyone to create a detailed digital 3D landscape model for rotational viewing on a computer or even for 3D printing. The data used to create such a model can be obtained from a variety of sources and file types, but it is becoming more and more common to digitally reconstruct a landscape surface from quality, geotagged aerial imagery taken by Unmanned Aerial Vehicles (a.k.a. drones or Unmanned Aerial Systems). However, freely available LiDAR, contour, or other elevation data (in the form of LAS, DWG, or DEM files) from the internet can also be used. The process of getting to the common 3D file type (STL or OBJ) may or may not involve expensive CAD or GIS software. And, the starting point need not be photographic images (like JPG); in fact, the process can be significantly shortened and simplified if an existing CAD file (such as a DWG) is available for the site. Whether this truncated process is generally viable is a part of this study. The desired end product was a 3D printed model; the question addressed became – How do 3D models created from various data types (such as LiDAR, DEM, UAV aerial imagery, and DWG of topographic contours), and the workflows that yielded them, compare in quality and detail, cost, and overall simplicity? Two very different terrains (study areas) were examined. Multiple data manipulation workflows for creating final 3D-printable STL files were applied to UAV-derived photos and internet-derived elevation data for each of the study areas. The "best" and most viable workflow was found to depend on a few major factors; site specifics including desired resolution and available data, processing power of hardware / software, and cost.

## Goals

Survey the available software and workflows for generating 3D-printable models from various data types.

Determine the "best" workflow(s) based on simplicity, cost, starting data type, and quality of the final 3D model. *In particular, compare results from UAV imagery vs data from online sources.*

Compile a guidebook for 3D-printing landscapes.

## Procedure

Two locations, two methods of data collection, and numerous possible workflows to get a final, 3D-printable STL file (see diagrams to the right).

### File Types:

*STL*: Stereo Lithographic data format used by various CAD systems and stereo lithographic printing machines; describes only surface geometry.
*OBJ*: Wavefront Technologies' format for representing 3D geometry; stores the position of each vertex, the UV position of each texture coordinate vertex, vertex normals, and the faces that make each polygon defined as a list of vertices, and texture vertices.
*PLY*: Polygon File Format / Stanford Triangle Format; stores a list of nominally flat polygons with possible properties including color and transparency, surface normals, texture coordinates and data confidence values.
*DWG*: AutoCAD and Open Design Alliance applications, Autodesk Inventor Drawing file.
*DEM*: Standard geospatial file format developed by the United States Geological Survey for storing a raster-based digital elevation model.
*JPEG* or *PNG*: Digital raster image formats. Joint Photographic Experts Group format supports lossy compression; Portable Network Graphics supports lossless data compression.
*LAS*: LASer format developed for LiDAR data (**Li**ght **D**etection **A**nd **R**anging is an optical remote-sensing technique that uses laser light to densely sample the surface of the earth, producing highly accurate x,y,z measurements).
*TIN*: Triangulated irregular network; vector-based GIS format used to represent a surface.

### Materials and Programs:

**DJI Phantom II Vision+:** Drone with accompanying camera, software, and smartphone. Cost (minus smartphone): about $1000.

**MakerBot Replicator 2:** 3D printer with accompanying software. Cost: $2500.

**MeshLab**: Free software for the processing and editing of unstructured 3D triangular meshes. Visual Computing Lab – ISTI – CNR, http://meshlab.sourceforge.net/

**Pix4D Mapper Pro:** Image processing software for mapping and modeling georeferenced 2D mosaics and 3D models from aerial imagery. Cost: $350/month, $3500/yr, $8700/one-time charge.

**SITEOPS Topo Lite:** Site engineering software; a free online service that e-mails you a DWG of contour lines (1 ft interval) for any area in the U.S.A. you select.

**ArcGIS / ArcMap**: By ESRI; powerful, versatile GIS software. Cost: Starting at a single license for $1500.

**VisualSFM:** Free software (by Changchang Wu) for 3D reconstruction using structure from motion. Combined with Yasutaka Furukawa's PMVS/CMVS tool chain, it takes many JPEG images as input and outputs a PLY file.

**Blender:** Free and powerful 3D rendering and modeling software.

## Results and Considerations

Although a wide variety of programs (ranging from free to costly) and possible workflows exist for the generation of a 3D-printable landscape model, the best route is site-specific. The starting data type and processing power (computer RAM and graphics card) are significant limiting factors. Although costly programs can be simpler to use, it may still be necessary or desirable to use them in conjunction with free software. Indeed, expensive programs are not always necessary; acceptable results can be obtained by utilizing a variety of free software and internet data sources. This research only utilized a sampling of available software—a quick internet search will reveal diverse other workflows and useful programs.

### Considerations include:

**1) Site Specifics, Desired Resolution, and Available Data:** for remote locations or if high-resolution is desired, UAV imagery is the best option; otherwise, freely-available contour, DEM, LiDAR data, etc. may be acceptable.
**2) Processing Power:** image processing by programs like Pix4D and VisualSFM require hefty hardware. Subsampling point clouds or simplifying meshes to reduce file size is a partial solution.
**3) Cost:** how much are you willing to pay?

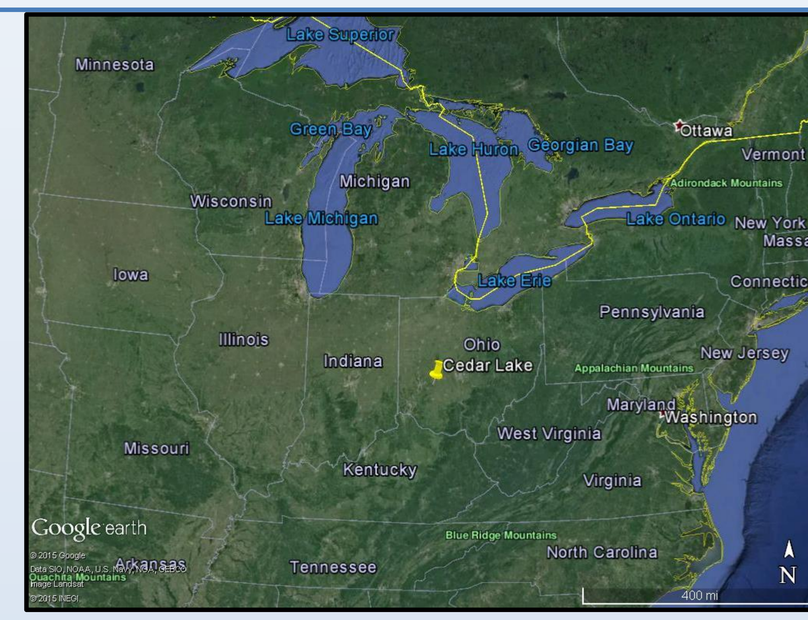
### Bonus!

www.opentopography.org, of the San Diego Supercomputer Center, provides free, high-resolution LiDAR data for select locations around the world. Depending on the location, you can download point cloud, DEM, and/or TIN data. A GeoTiff TIN obtained this way can be opened in *MicroDEM* (a free microcomputer mapping program written by Professor Peter Guth of the Oceanography Department, U.S. Naval Academy) and exported as an OBJ file for further refining in Blender / MeshLab using File → Save DEM → Caveat emptor → 3D (OBJ).

*Want to add text or symbols (like north arrows or scale bars) to models?* If generating a DEM image in ArcMap for use in Blender, try overlaying thick text or symbols and shading them slightly different from the surrounding DEM to either emboss or engrave them. *Blender* can also be used to write text, extrude it, and Boolean it to emboss or engrave onto a model. *Autodesk Meshmixer* (free software for manipulating and editing meshes) is useful for unioning or subtracting meshes from each other; import your 3D landscape along with a mesh of text or a symbol and combine them.

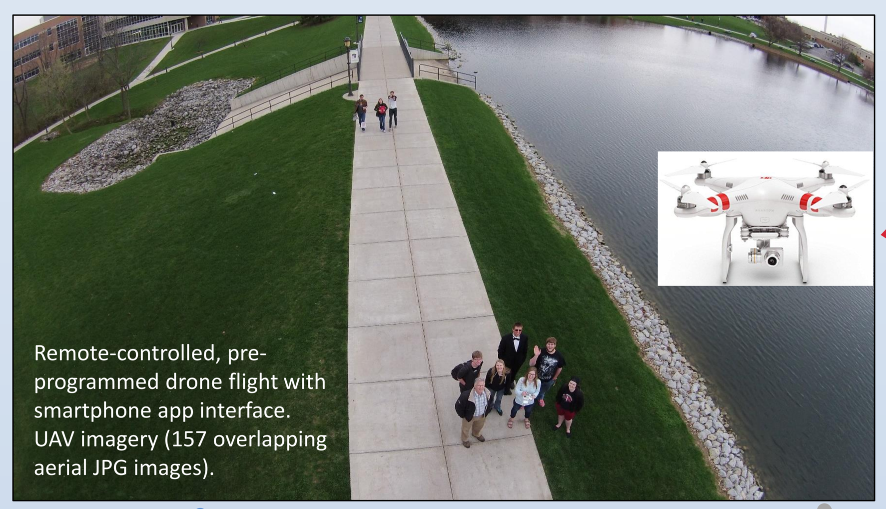
*Want to embed your 3D model into a PDF?* A costly option (unless you just use the free trial) is *Tetra4D 3D PDF Converter*. A free option (untested by the authors) involves using *LaTeX* (with the media9 package – not the outdated media15 package) to embed a U3D file-type model. Details at http://rainnic.altervista.org/content/embed-3ds-pdf-latex-u3d.

*Don't have access to a 3D printer?* You can upload your model and get a price quote for printing in 40+ materials at http://www.shapeways.com; if one appeals to you, they'll print your model and mail it!

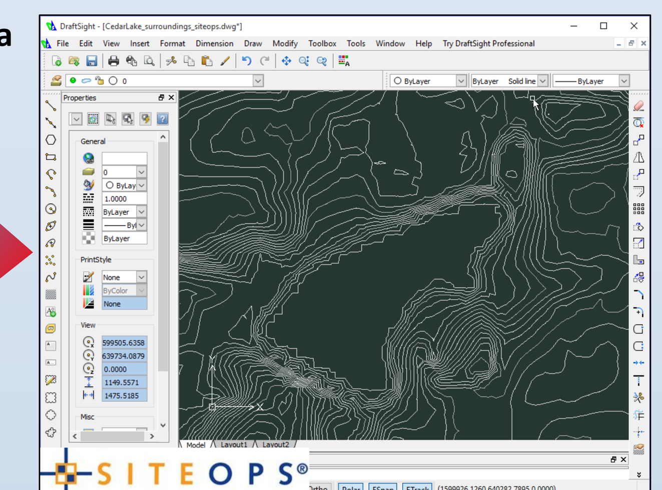




**Two Methods of Data Collection: UAV and Internet**



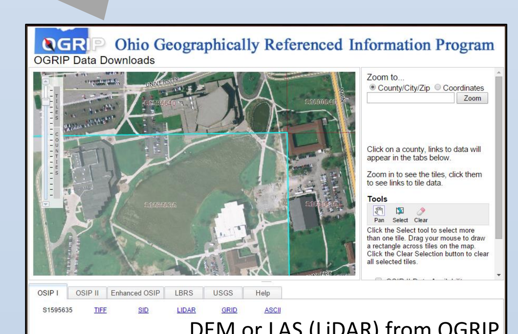
Remote-controlled, pre-programmed drone flight with smartphone app interface. UAV imagery (157 overlapping aerial JPG images).



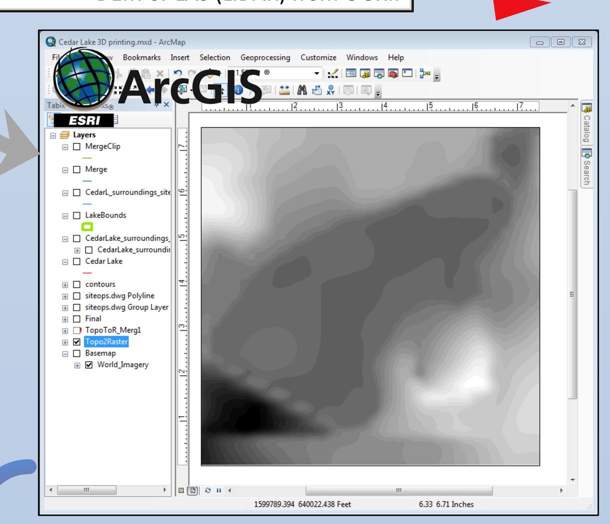
DWG of 1ft contours

Image processing (generate PLY point cloud, orthomosaic, and potentially STL).

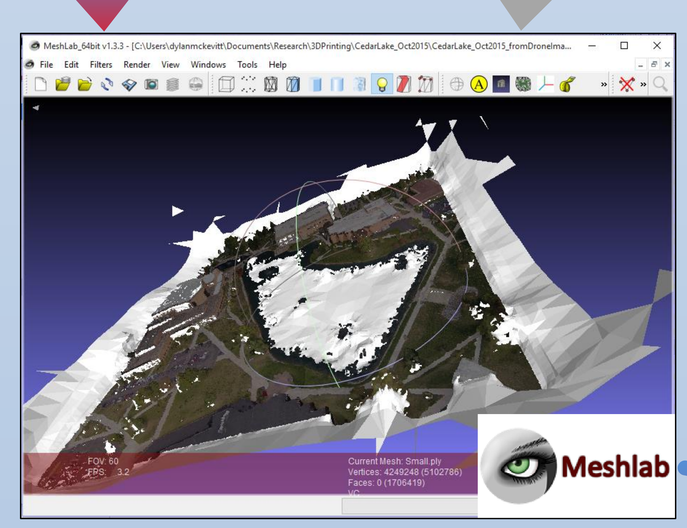
Process images and output PLY point cloud.



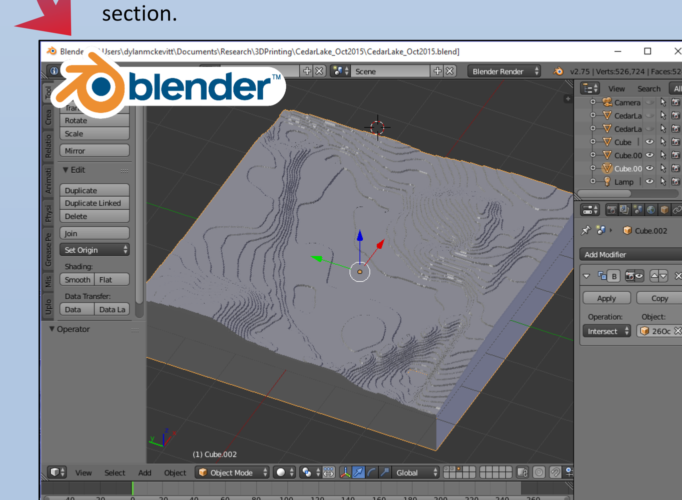
DEM or LAS (LiDAR) from OGRIP



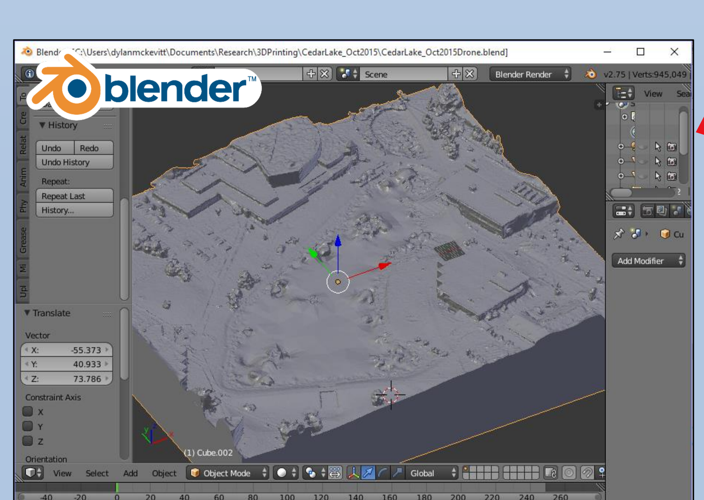
Convert DWG or LiDAR into raster DEM (and possibly TIN) using ArcMap. Possibly import DEM and "clip out" desired section.



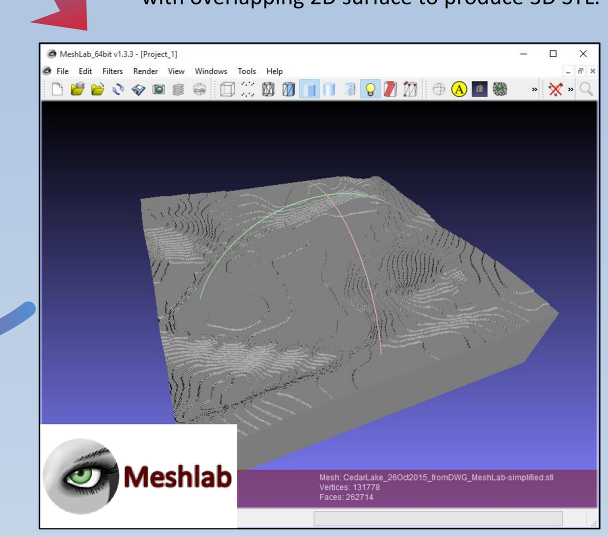
Cleaning of point cloud (PLY or other format). Possibly subsampling to reduce file size. Compute normals and surface reconstruction. Output STL. Possibly quadric edge collapse decimation to reduce file size.



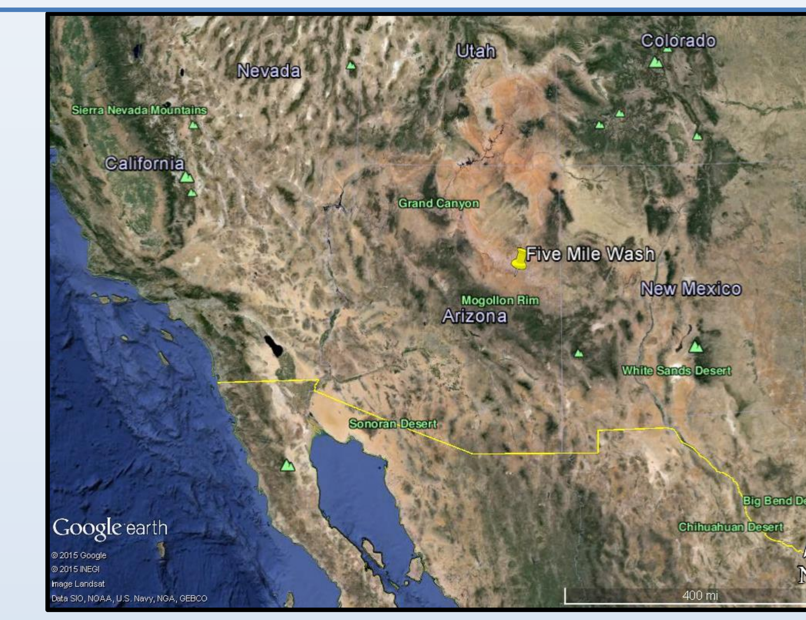
Import image as plane; subdivide surface and displace modifiers; insert cube and Boolean modifier with overlapping 2D surface to produce 3D STL.



Import thin, 2D STL surface mesh. Insert cube and Boolean with overlapping 2D surface to produce 3D STL or OBJ.



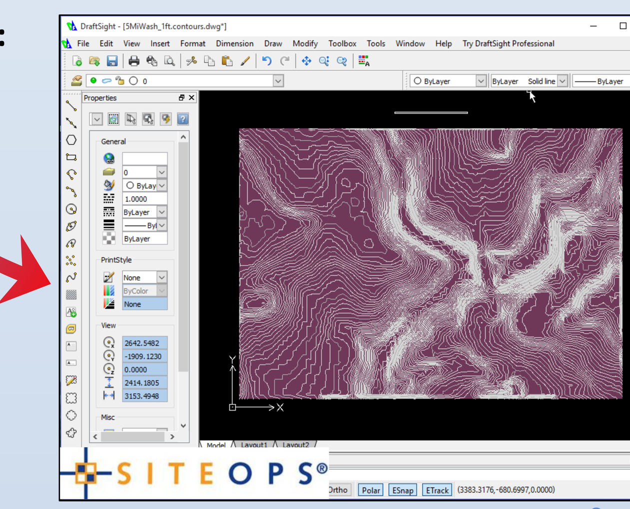
Import STL and simplify mesh using quadric edge collapse decimation to reduce file size.



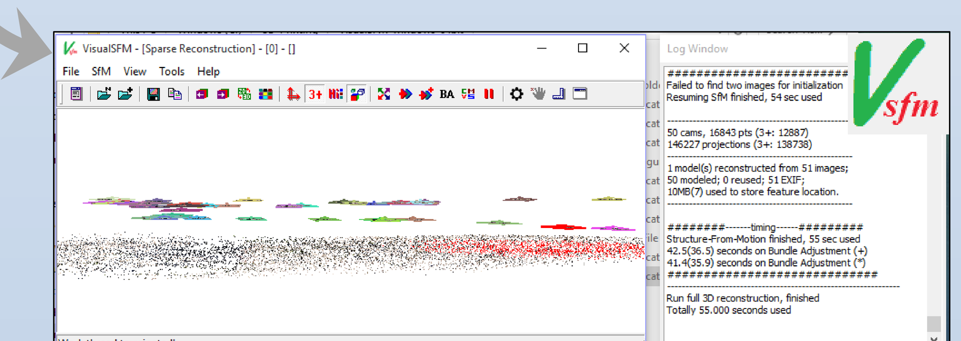


**Two Methods of Data Collection: UAV and Internet**

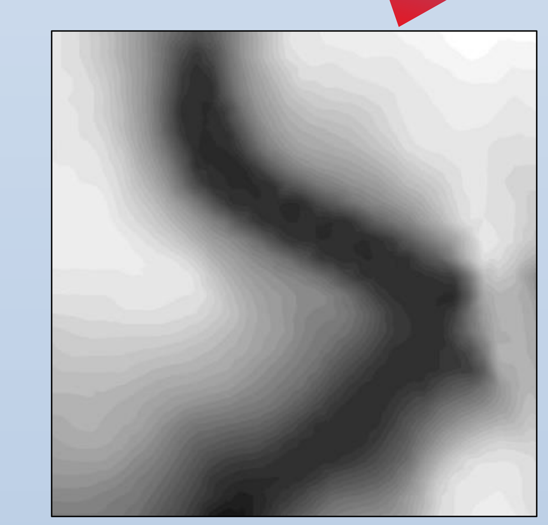
Remote-controlled, pre-programmed drone flight with smartphone interface and DJI app.
UAV imagery (51 overlapping aerial JPG images).



DWG of 1ft contours



Process images and output PLY point cloud.



Convert DWG into raster DEM (and possibly TIN) using ArcMap.

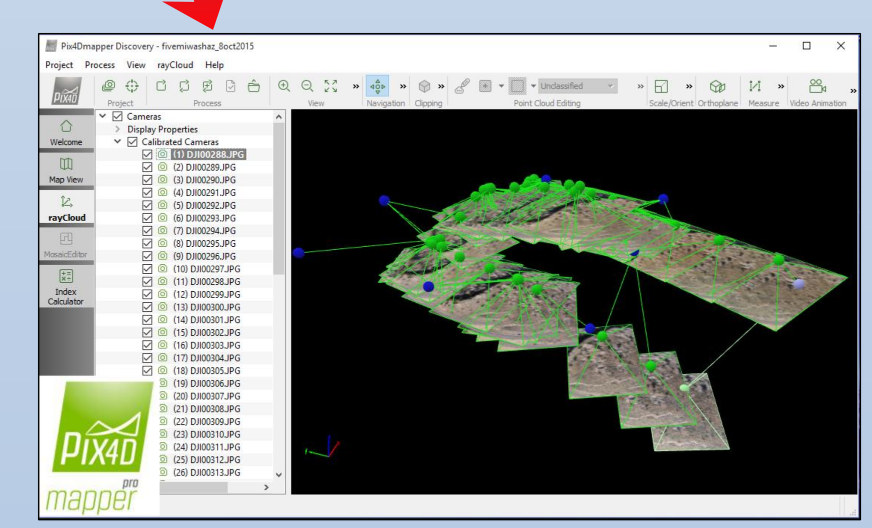
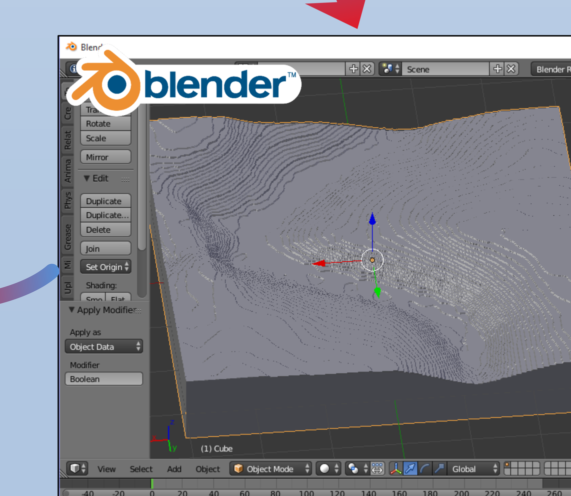
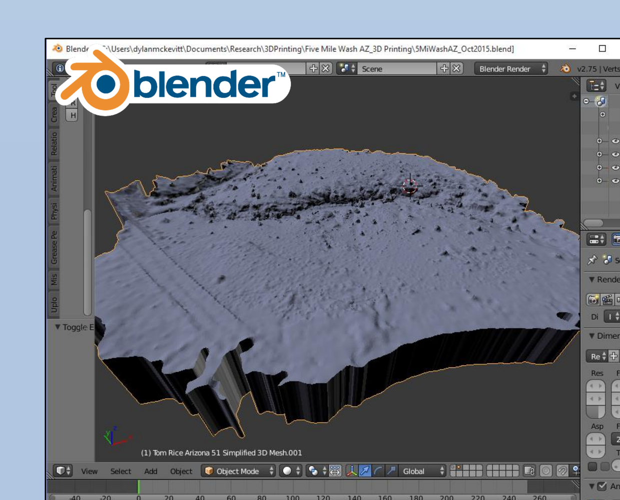


Image processing (generate PLY point cloud, orthomosaic, and potentially STL).



Import image as plane; subdivide surface and displace modifiers; insert cube and Boolean modifier with overlapping 2D surface to produce 3D STL.

Cleaning of point cloud (PLY or other format). Possibly subsampling to reduce file size. Compute normals and surface reconstruction. Output STL. Possibly quadric edge collapse decimation to reduce file size.



Uneven edges: extrude 2D STL surface and flatten base by scaling or cutting along a plane.
Straight / square edges: insert cube / etc. and Boolean with overlapping 2D surface to produce 3D STL or OBJ.

Import and print STL or OBJ.

Three models produced; the first (far left) was an attempt to print only the 2D STL surface.


## Acknowledgements

We would like to thank **Professor T. Rice** for guiding and assisting us throughout this research; the **Cedarville University Engineering Department** for the use of their MakerBot 3D printer; **Sarah Maithel** for the opportunity to accompany her on field work in Arizona; and **Matt Quimby** of Remote Intelligence for assistance in processing with Pix4D.