Open Source Tools for 3D Forensic Reconstructions - Part 4

Eugene Liscio, P. Eng.

January 2012

In the past several years, there have been some interesting developments and availability of open source software that allows people to create point clouds and models based on using nothing more than their digital cameras and free software available online. The development has been rather rapid and the options of using different packages has been growing with more research and individual programmers donating their time to provide solutions to the creation of 3D point clouds and meshes that can be directly imported into CAD or modeling packages. Not many people in the Forensics area have considered what open source can do for them nor might they be aware of the capabilities of using open source tools.

This article is the fourth instalment in a series of 4 articles that will present several open source solutions to creating point clouds and 3D models for use in Forensic applications and will look at some of the pros and cons of each.

In my three previous articles I provided information on different software that can create point clouds or meshes by capturing photos of an object or scene. In this last instalment, I will discuss another open source option that is perhaps the most popular among 3D enthusiasts. MeshLab has come a long way from the days of being a small lab project and today, there are many researchers and 3D enthusiasts who use it for point cloud manipulation, meshing and analysis.

MeshLab

Perhaps the most popular 3D point cloud and meshing software available as open source, MeshLab has really come a long way with respect to its features. MeshLab began in Italy at the University of Pisa in late 2005. It is an open-source general-purpose system aimed at the processing of the typical not-so-small unstructured 3D models that arise in the 3D scanning pipeline. MeshLab is oriented to the management and processing of unstructured large meshes and provides a set of tools for editing, cleaning, healing, inspecting, rendering and converting these kinds of meshes.[1]

MeshLab is available for Windows, MacOSX, and Linux. So, in order to get started, simply download the latest version of MeshLab from here. You can also download some sample files from the Stanford 3D Scanning Repository.

Words of caution to those who do not like to read instructions or prefer to jump right into the software and learn on the go. I would highly recommend that you look at some tutorials on YouTube or on the MeshLab website and save yourself a lot of heartache. MeshLab is quite powerful, but it takes some time to get to know many of the features and where to find many of the specific functions. In addition, there is a specific working procedure and method that you need to understand before jumping in head first. Sometimes it takes some tweaking to get the settings just right and other times, the software can be buggy. However, you can often be surprised by what you can produce with MeshLab.

[1] http://en.wikipedia.org/wiki/MeshLab

When you first load up MeshLab you are presented with an interface that is rather clean and simple. In order to import a file, you need to either import an existing MeshLab project or use the "Import Mesh" command. This gives you a very large list of possible formats to import, however, the default format for MeshLab is the PLY file format.

PLY is a computer file format known as the Polygon File Format or the Stanford Triangle Format. The format was principally designed to store three dimensional data from 3D scanners. It supports a relatively simple description of a single object as a list of nominally flat polygons. A variety of properties can be stored including: color and transparency, surface normals, texture coordinates and data confidence values. The format permits one to have different properties for the front and back of a polygon.[2] There are two versions of the PLY file, one in ASCII, the other in binary format.

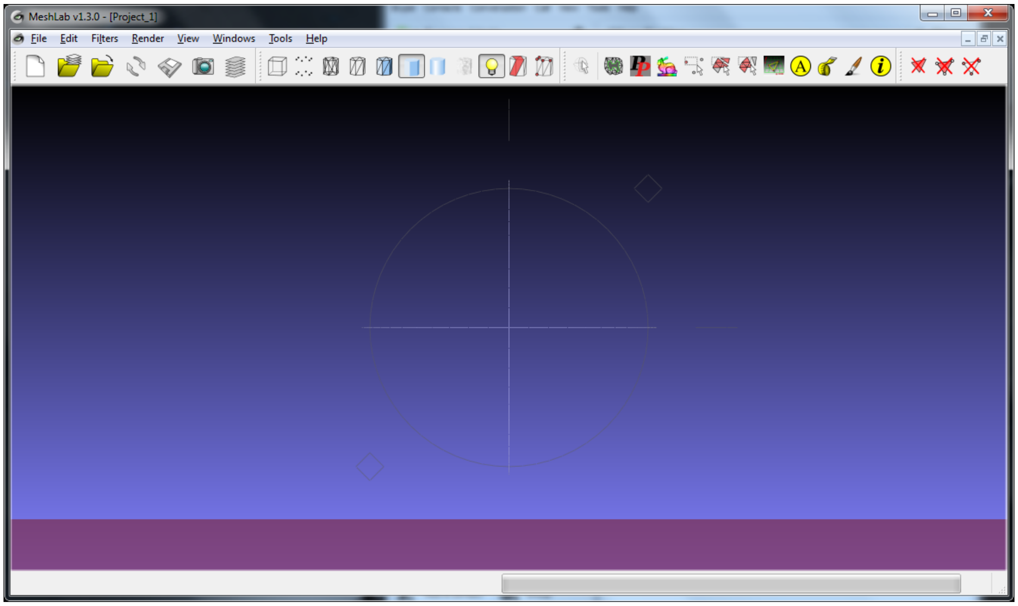
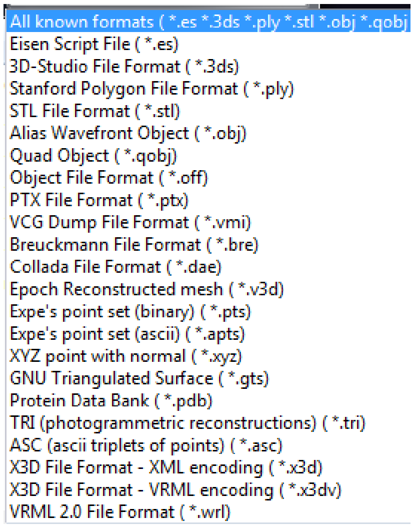


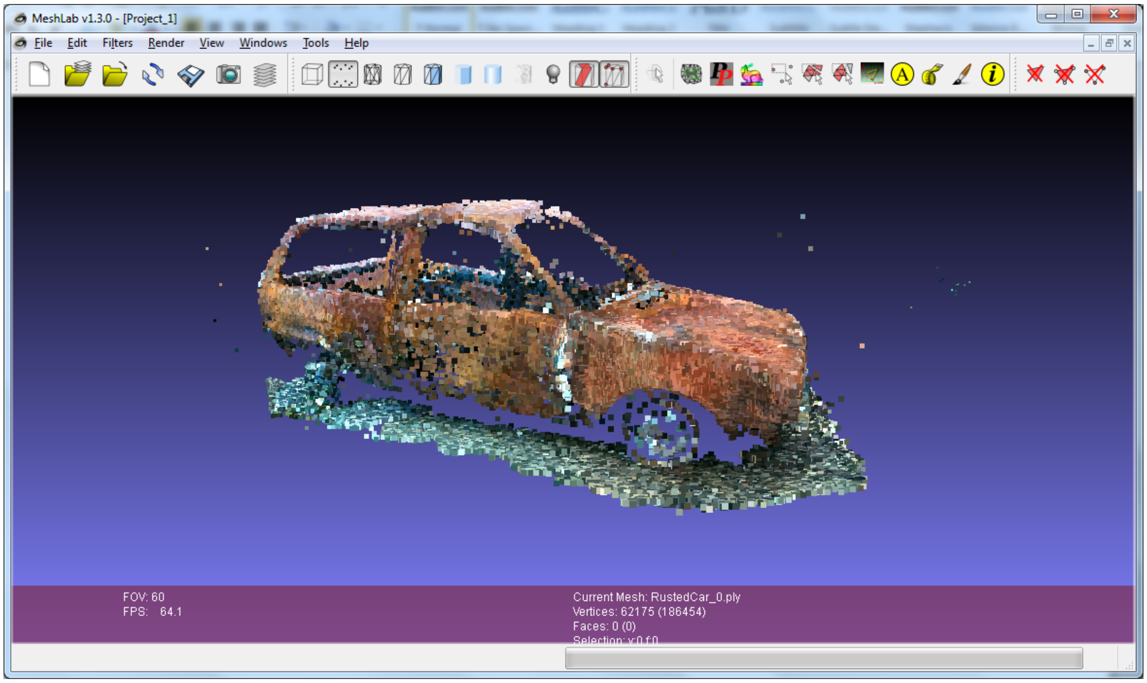
Figure 1. MeshLab Interface



All known formats (*.es *.3ds *.ply *.stl *.obj *.qobj
Eisen Script File (*.es)
3D-Studio File Format (*.3ds)
Stanford Polygon File Format (*.ply)
STL File Format (*.stl)
Alias Wavefront Object (*.obj)
Quad Object (*.qobj)
Object File Format (*.off)
PTX File Format (*.ptx)
VCG Dump File Format (*.vmi)
Breuckmann File Format (*.bre)
Collada File Format (*.dae)
Epoch Reconstructed mesh (*.v3d)
Expe's point set (binary) (*.pts)
Expe's point set (ascii) (*.apts)
XYZ point with normal (*.xyz)
GNU Triangulated Surface (*.gts)
Protein Data Bank (*.pdb)
TRI (photogrammetric reconstructions) (*.tri)
ASC (ascii triplets of points) (*.asc)
X3D File Format - XML encoding (*.x3d)
X3D File Format - VRML encoding (*.x3dv)
VRML 2.0 File Format (*.wrl)

Figure 2. Possible Import Formats for MeshLab

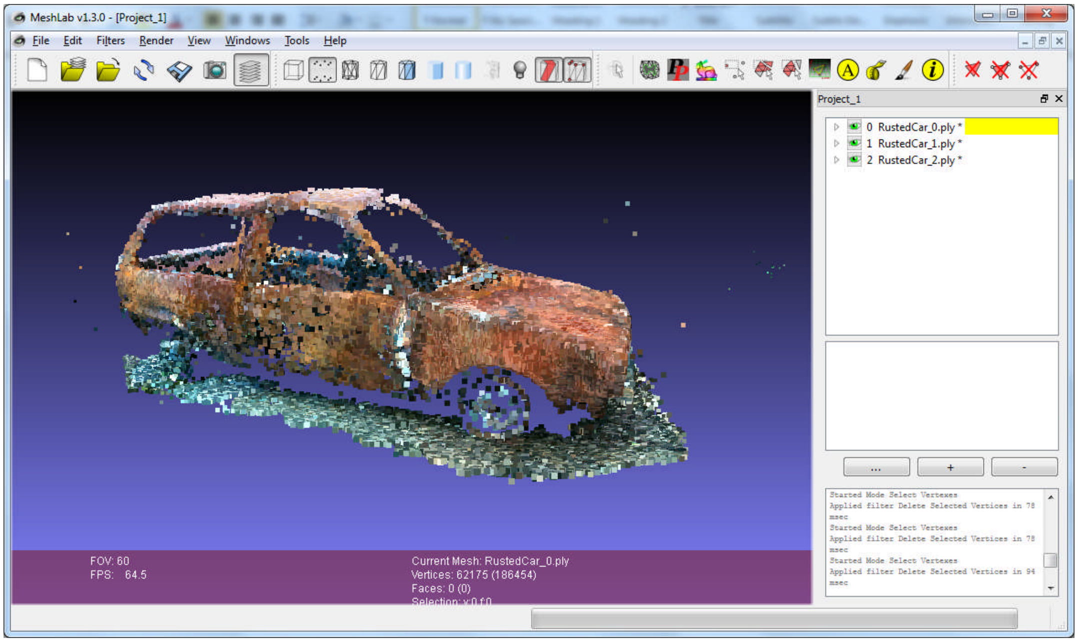
[2] http://en.wikipedia.org/wiki/PLY_(file_format)


MeshLab v1.3.0 - [Project_1]
File Edit Filters Render View Windows Tools Help

```
FOV: 60
FPS:   64.1
Current Mesh: RustedCar_0.ply
Vertices: 62175 (186454)
Faces: 0 (0)
Selection: v:0 f:0
```

Figure 3. Imported PLY file (points only)

With respect to editing the point cloud, there are some simple selection tools that allow one to highlight and delete points as necessary using the select vertices icon and the delete vertices icon. There is also a layers feature that makes selection and visibility of individual meshes possible. This is available from the layers icon and brings up a small window in on the right side of the 3D Viewer window as shown below:


MeshLab v1.3.0 - [Project_1]
File Edit Filters Render View Windows Tools Help
Project_1
...
+
-

```
0  RustedCar_0.ply *
1  RustedCar_1.ply *
2  RustedCar_2.ply *
```

```
Started Mode Select Vertexes
Applied filter Delete Selected Vertices in 78
msec
Started Mode Select Vertexes
Applied filter Delete Selected Vertices in 78
msec
Started Mode Select Vertexes
Applied filter Delete Selected Vertices in 94
msec
```

```
FOV: 60
FPS:   64.5
Current Mesh: RustedCar_0.ply
Vertices: 62175 (186454)
Faces: 0 (0)
Selection: v:0 f:0
```

Figure 4. Layers window active with all layers turned on.

From this window you can add or delete layers, plus there is a small contextual window at the bottom that gives you specific information about each of the processes that are run on any point cloud or mesh.

Meshing

The two most popular meshing options in MeshLab are the Poisson and Ball Pivoting methods of surface Poisson reconstruction from a point cloud. The Poisson reconstruction is often used when a water tight mesh is needed. Using this method for point clouds that are not completely closed can often give a lot of artifacts on the edges where the algorithm tries to wrap around the object to create a fully closed mesh.

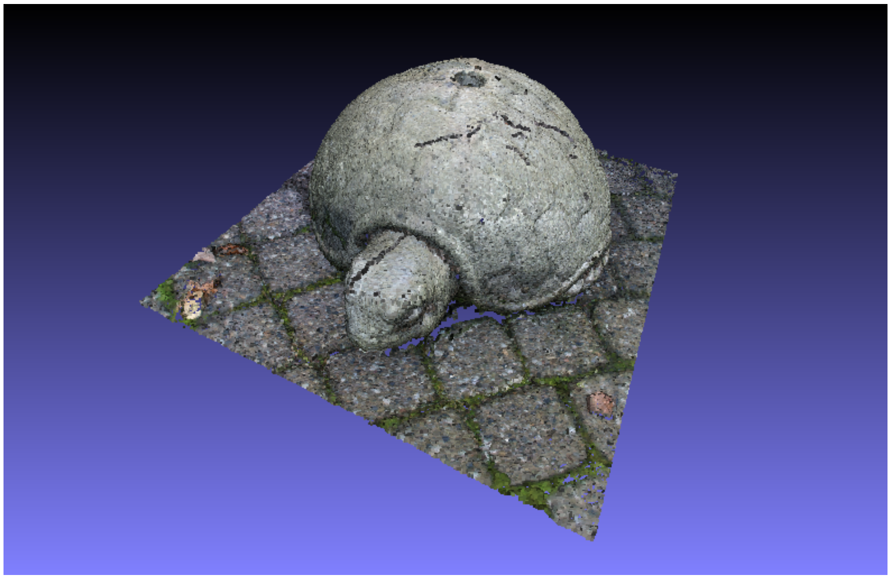


Figure 5. Point cloud of a stone turtle prior to Poisson meshing.

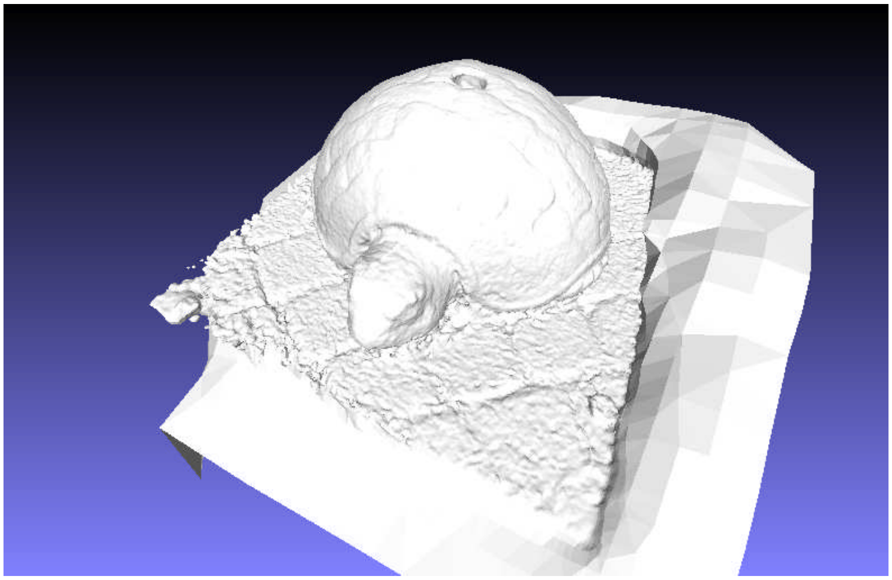


Figure 6. Meshed point cloud using Poisson reconstruction in MeshLab.

The Ball Pivoting Algorithm (Bernardini et al. 1999) starts with a seed triangle, the BPA algorithm pivots a ball around an edge (i.e. it revolves around the edge while keeping in contact with the edge endpoints) until it touches another point, forming another triangle. The process continues until all reachable edges have been tried.

There is a good tutorial on meshing and preparing a point cloud for surface reconstruction here.

Importing Photosynth Point Clouds

Another interesting option in MeshLab is the ability to import a Photosynth point cloud directly into the 3D Viewer without having to use a third party application. Directly inside the Filters menu and under Create New Mesh Layer, you will find an option for Import Photosynth Data.

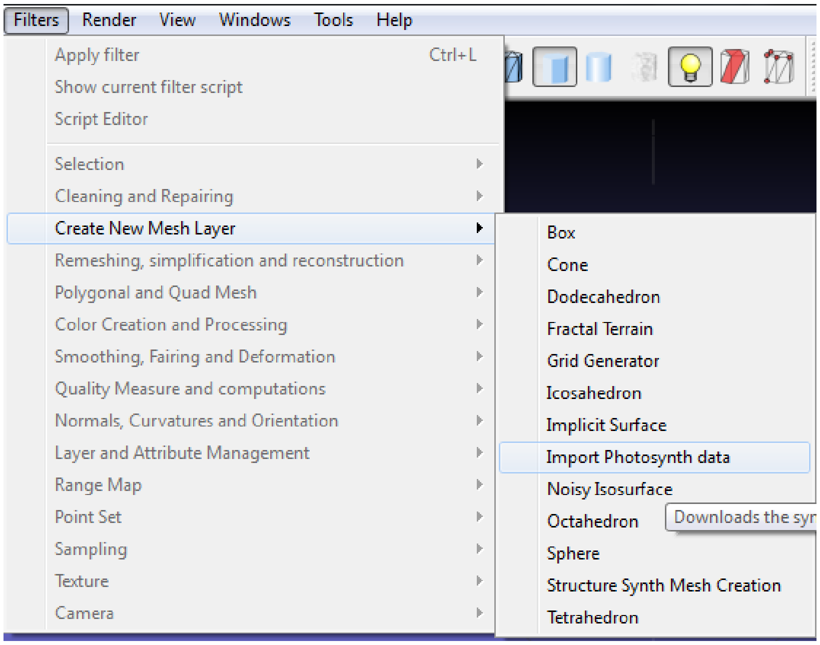
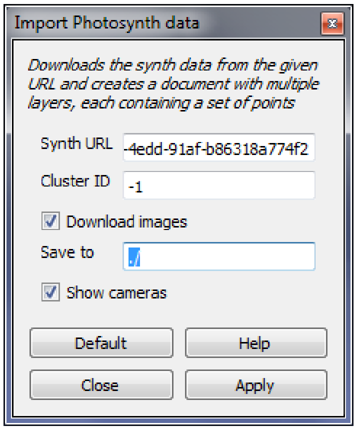


Figure 7. Import Photosynth Data under the Create New Mesh Layer sub-menu item.

This brings up a very simple menu where you can enter the URL of the created Photosynth and there is an option to download the images plus calculated camera positions. A sample URL will look like the following:

http://photosynth.net/view.aspx?cid=49c5c1e9-1d19-4edd-91af-b86318a774f2


Import Photosynth data

Downloads the synth data from the given URL and creates a document with multiple layers, each containing a set of points

Synth URL -4edd-91af-b86318a774f2

Cluster ID -1

☑ Download images

Save to ./

☑ Show cameras

Figure 8. Import Photosynth Data window.

It's important to remember that in the "Save to" line, ensure that the directory you are saving to already exists. When completed filling out the fields, click "Apply" and the points should be imported directly from the Photosynth server. Note that you must select the ⁙ icon so you can see the vertices/points. At this time, only points are imported and not the RGB color information. For better viewing, it also helps to turn off the default lighting (click the 💡 icon).

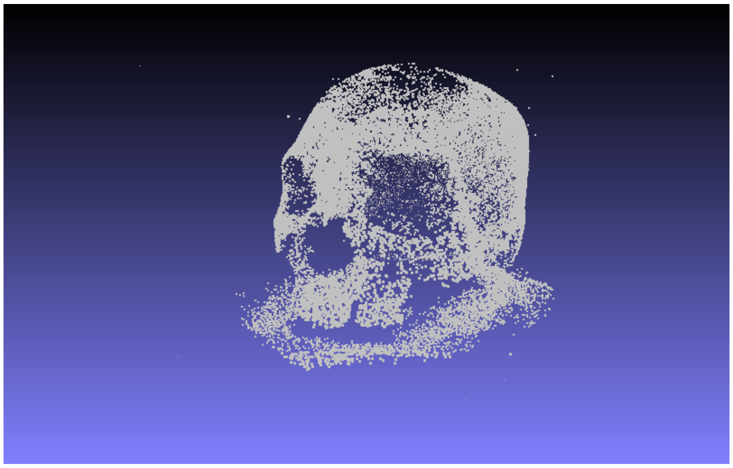
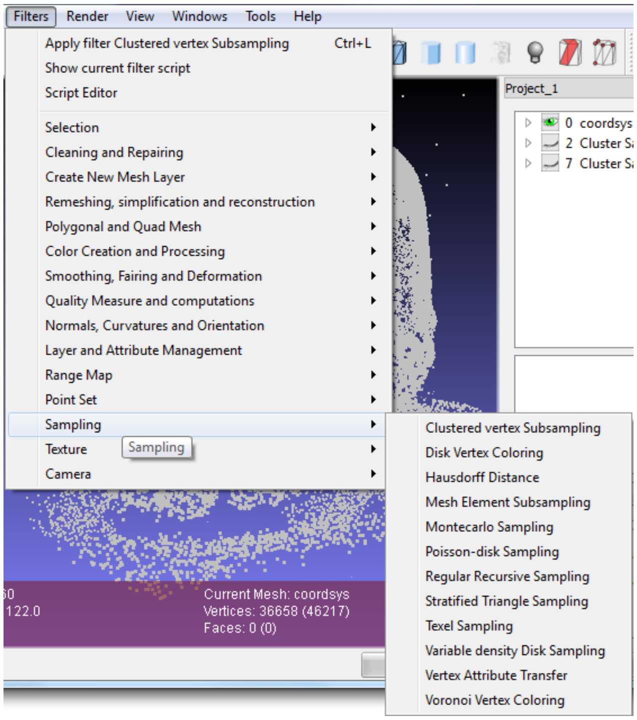


Figure 9. Imported Photosynth into MeshLab.

Sampling

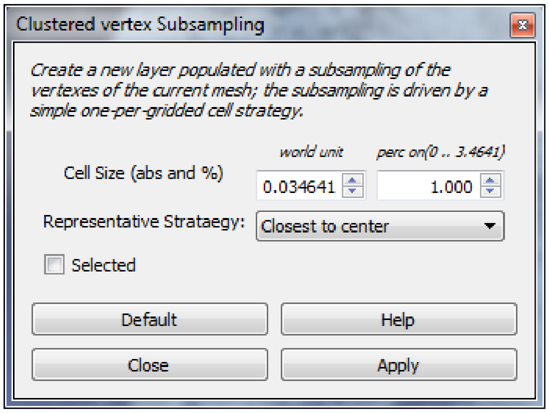
Although MeshLab can handle millions of points, many of the processes take a long time and you will find it much easier to simplify or decimate the point cloud by using the tools in the Sampling menu. As an example by using the previously imported point cloud of the skull (above) from Photosynth, there are currently over 36,000 points. Although this is a relatively small point cloud, MeshLab reportedly has a maximum capacity of handling point clouds in the range of 31 million points or thereabouts.


Filters
Render
View
Windows
Tools
Help
Apply filter Clustered vertex Subsampling
Ctrl+L
Show current filter script
Script Editor
Selection
Cleaning and Repairing
Create New Mesh Layer
Remeshing, simplification and reconstruction
Polygonal and Quad Mesh
Color Creation and Processing
Smoothing, Fairing and Deformation
Quality Measure and computations
Normals, Curvatures and Orientation
Layer and Attribute Management
Range Map
Point Set
Sampling
Texture
Sampling
Camera
Clustered vertex Subsampling
Disk Vertex Coloring
Hausdorff Distance
Mesh Element Subsampling
Montecarlo Sampling
Poisson-disk Sampling
Regular Recursive Sampling
Stratified Triangle Sampling
Texel Sampling
Variable density Disk Sampling
Vertex Attribute Transfer
Voronoi Vertex Coloring
Project_1
0 coordsys
2 Cluster S
7 Cluster S
122.0

Current Mesh: coordsys
Vertices: 36658 (46217)
Faces: 0 (0)

Figure 10. Sampling menu

In the top of the Sampling sub-menu is a feature called “Clustered Vertex Subsampling”. This is a decimation algorithm that will reduce the size of the point cloud. Many of the other options in the Sampling sub-menu are for meshes only and will not work on point clouds.



Create a new layer populated with a subsampling of the vertexes of the current mesh; the subsampling is driven by a simple one-per-gridded cell strategy.

	world unit	*perc on(0 .. 3.4641)*
Cell Size (abs and %)	0.034641	1.000

Representative Strataegy: Closest to center

□ Selected

Figure 11. Custered Vertex Subsampling window.

Using the default values will result in Figure 10. The larger the values of the “Cell Size”, the greater the amount of decimation. Often experimentation is required to get the desired amount of reduction in points.

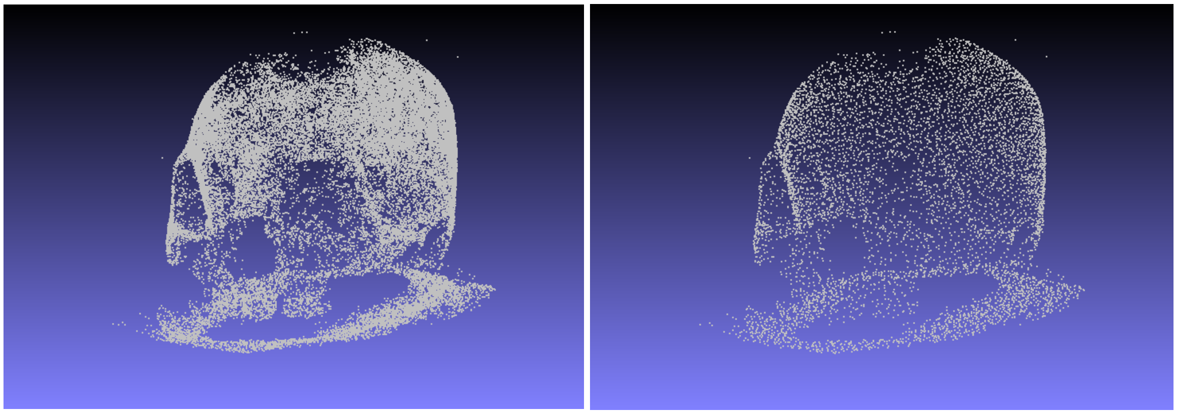
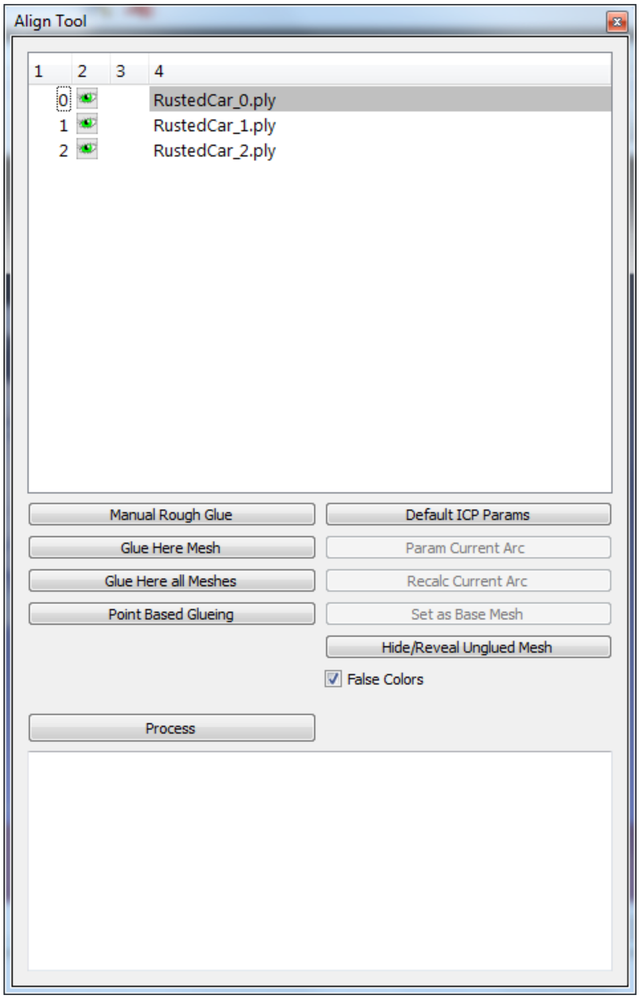


Figure 12. Before and after sampling of point cloud. Decimation from approximately 36,000 points to 7,000 points.

Registration

Registration is possible through the Align icon A which brings up a contextual menu as follows:



1	2	3	4
0			RustedCar_0.ply
1			RustedCar_1.ply
2			RustedCar_2.ply

☑ False Colors

Figure 13. Align menu for registration of scans.

There are 3 key concepts to registration.

ICP: The basic algorithm that automatically precisely aligns a moving mesh M with a fixed one F. The main idea is that we choose a set of (well distributed) points over M and we search on F for the corresponding nearest points. These pairs are used to find the best rigid transformation that brings the points of M onto the corresponding points of F. ICP has a lot of tunable parameters.

Global Alignment: This is also known as multiview registration. A final step that evenly distributes the alignment error among all the alignments in order to avoid the biased accumulation of error.

Absolute Scale: The parameters of the alignment tool are in absolute units, and the defaults are acceptable for a standard scanner outputting meshes in millimeter units. So for example the target error (i.e. the error that the ICP try to achieve) is 0.05 mm something that can be achieved with a good scanner. If your range maps are in a different unit (microns, kilometers,...etc.) you must adjust the default alignment parameters or the alignment process will fail.

Overall, there are many other useful features in MeshLab and it’s a matter of testing and tweaking. Unfortunately, the documentation isn’t always the greatest, but there are some YouTube videos and Blogs around that provide some decent information. There is also a user’s forum where you can post issues or ask questions about specific features, however, nothing beats just trying the software out with your own sample data sets.

Eugene Liscio, P. Eng, is the owner of AI^2-3D. A company that specializes in 3D Forensic Measurement, Analysis and Visualizations. AI2-3D, Toronto, Ontario, 416-704-2695, information@ai2-3d.com, www.ai2-3d.com.