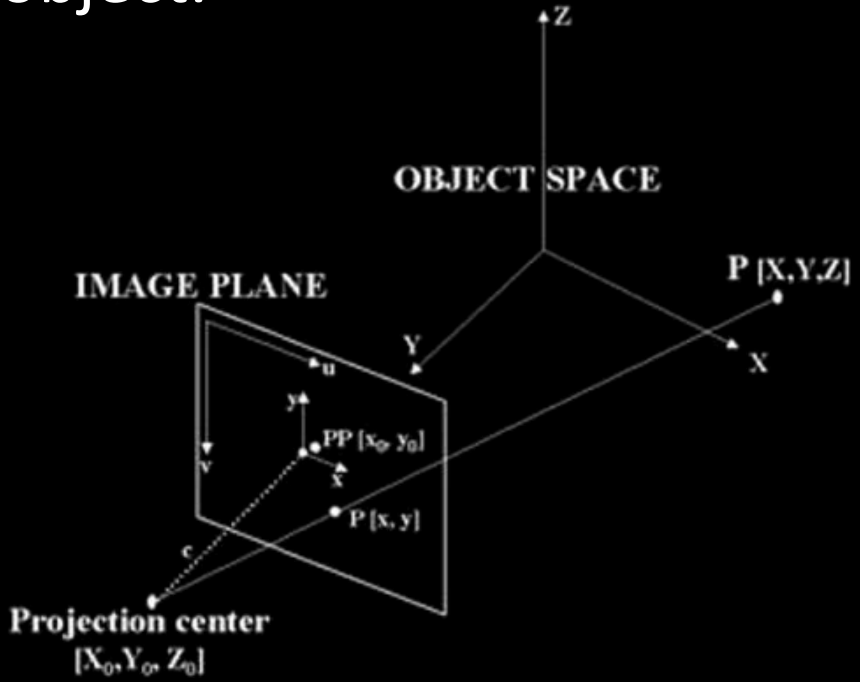
COPIS

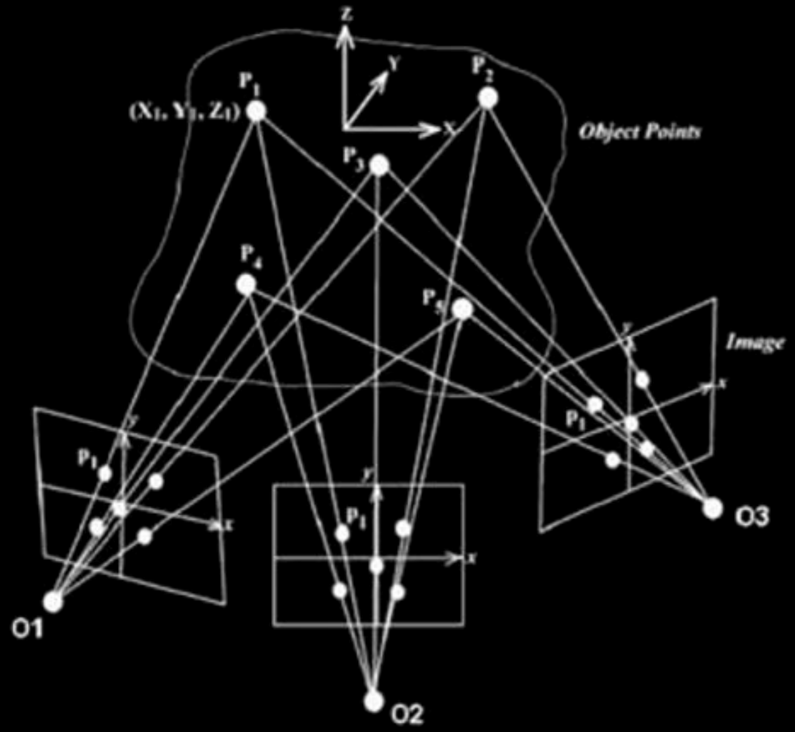
Computer Operated Photogrammetric Imaging System

Nelson Rios
Henry L. Bart

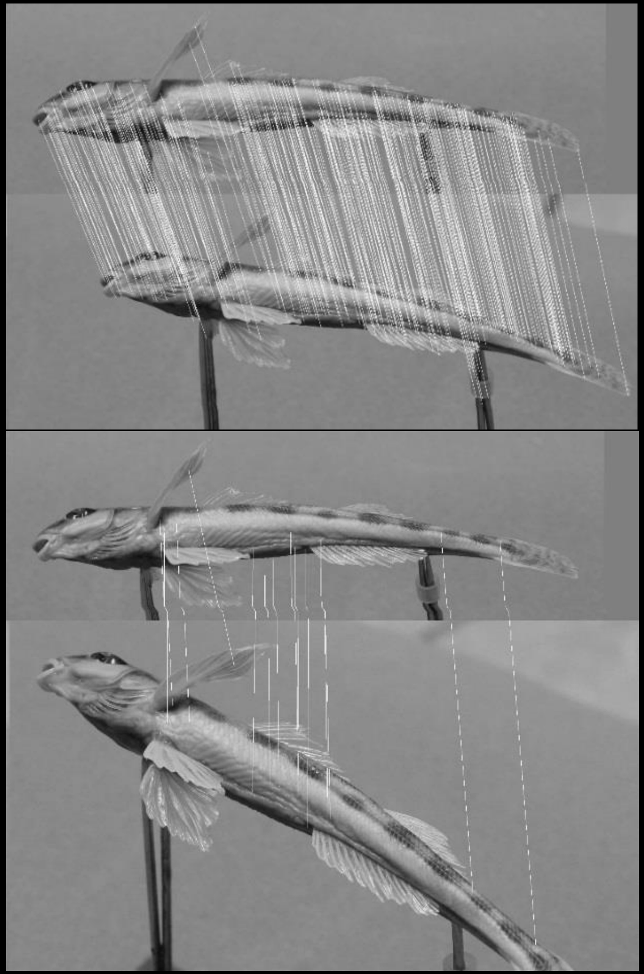
Photogrammetry

- Photogrammetry is a method of recovering the three dimensional position of points on an object's surface from 2 or more photographs of the object.





Features identified within an image using Scale Invariant Feature Transformation (SIFT). Scale and orientation indicated by arrows.



Matching keypoints between two pairs of images.

Results of SIFT feature detection, pairwise keypoint matching and Bundle Adjustment using Visual SfM ver 5.26.

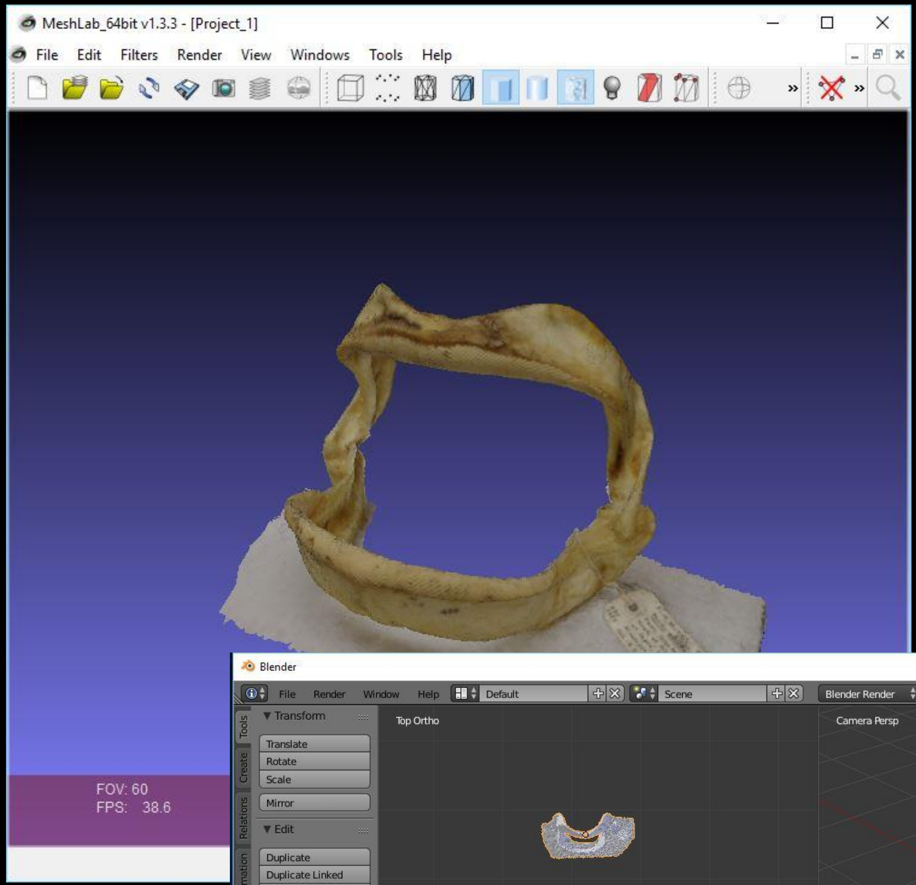
Dense Reconstruction

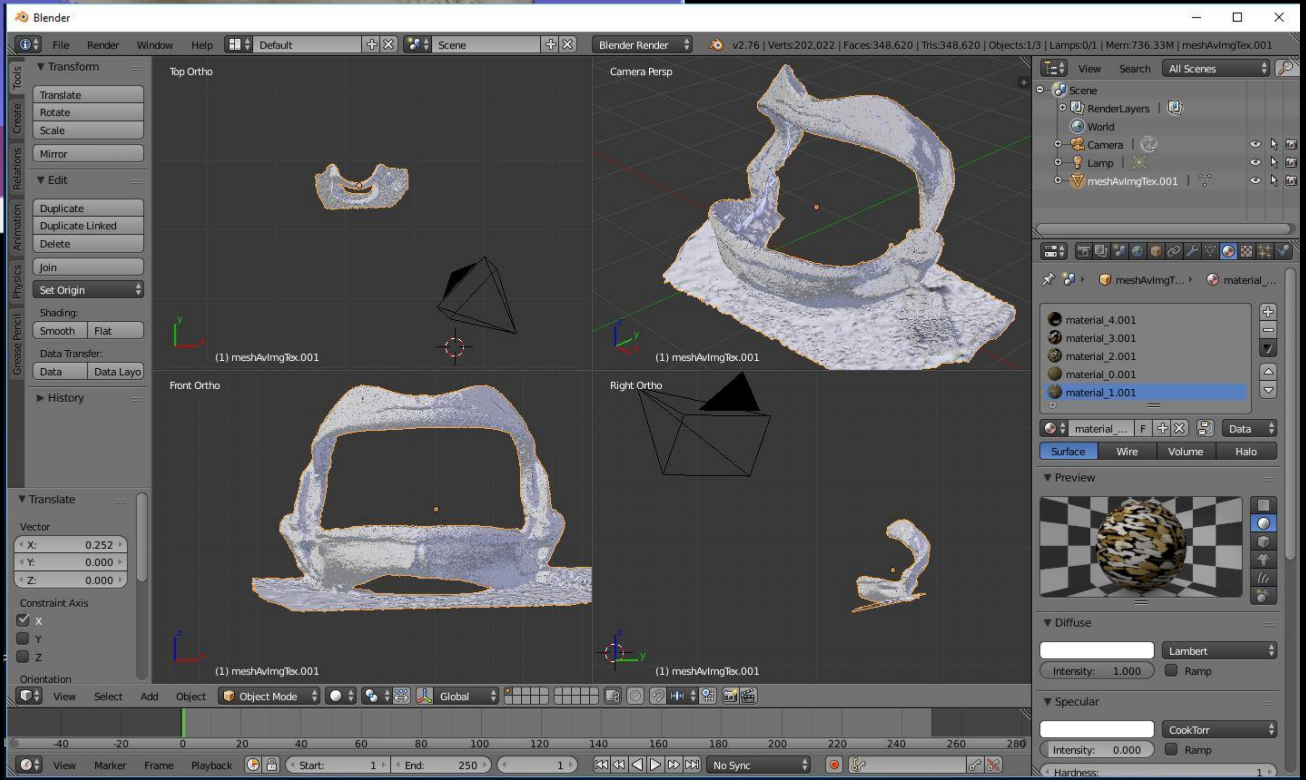
```
[prematching:minangle]=[3.0]
[grow]=UNDEF
[grow:minnumofconsistentcams]=[6]
[filter]=UNDEF
[filter:minnumofconsistentcams]=[2]
[hallucinationsfiltering]=UNDEF
[hallucinationsfiltering:useskyprior]=[FALSE]
[hallucinationsfiltering:doleavelargestfullsegmentonly]=[FALSE]
[hallucinationsfiltering:doremovehugetriangles]=[TRUE]
[largescale]=UNDEF
[largescale:dogenerateandreconstructspacemaxpts]=[TRUE]
[largescale:dogeneratespace]=[TRUE]
[largescale:planmaxpts]=[3000000]
[generatevideoframes]=UNDEF
[generatevideoframes:nintermed]=[20]
[dem]=UNDEF
[dem:demw]=[8000]
[dem:demh]=[8000]
Preparing data
Processing input images
Loading input data
Resizing input images
Creating temp data
PSSGM autoScaleStep 2 4
nImgsInGPUAtTime 2, scales 2, gammaC 5.500000, gammaP 8.000000, subPixel 1, varianceWSH 4
Device memory - used: 694.415649, free: 3401.584473, total: 4096.000000
Device memory - used: 866.665649, free: 3229.334473, total: 4096.000000
PSSGM rc 0 of 64 elapsed time 1 minutes 28 seconds 3 miliseconds
PSSGM rc 1 of 64 elapsed time 1 minutes 20 seconds 791 miliseconds
```

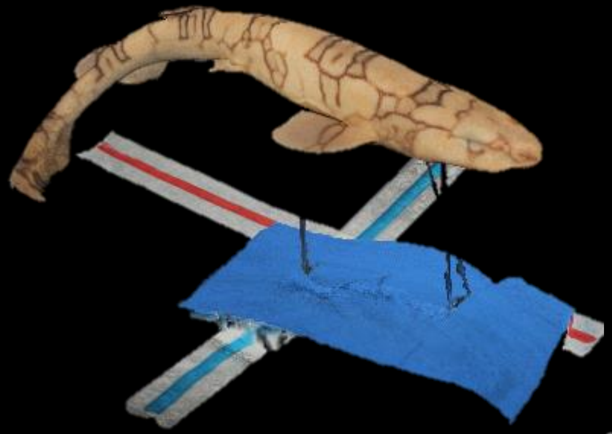
CMPMVS - Multi-View Reconstruction Software

Authors: Michal Jancosek & Tomas Pajdla
Software written by: Michal Jancosek
Latest version: 0.6.0
Release date: September 28, 2012
Reference to cite:
[1] *M. Jancosek, T. Pajdla. Multi-View Reconstruction Preserving Weakly-Supported Surfaces*, CVPR 2011 - IEEE Conference on Computer Vision and Pattern Recognition 2011 (pdf).

Images & camera positions -> dense point cloud -> mesh -> textured mesh


MeshLab_64bit v1.3.3 - [Project_1]
File Edit Filters Render View Windows Tools Help
FOV: 60
FPS: 38.6


Blender
File Render Window Help
Default
Scene
Blender Render
v2.76 | Verts:202,022 | Faces:348,620 | Tris:348,620 | Objects:1/3 | Lamps:0/1 | Mem:736.33M | meshAvImgTex.001
Transform
Translate
Rotate
Scale
Mirror
Edit
Duplicate
Duplicate Linked
Delete
Join
Set Origin
Shading:
Smooth
Flat
Data Transfer:
Data
Data Layo
History
Top Ortho
Camera Persp
Front Ortho
Right Ortho
(1) meshAvImgTex.001
Translate
Vector
X: 0.252
Y: 0.000
Z: 0.000
Constraint Axis
X
Y
Z
Orientation
View Search All Scenes
Scene
RenderLayers
World
Camera
Lamp
meshAvImgTex.001
material_4.001
material_3.001
material_2.001
material_0.001
material_1.001
Surface Wire Volume Halo
Preview
Diffuse
Lambert
Intensity: 1.000
Ramp
Specular
CookTorr
Intensity: 0.000
Hardness: 1
View Select Add Object Object Mode Global
View Marker Frame Playback Start: 1 End: 250 1 No Sync




Alternatives…

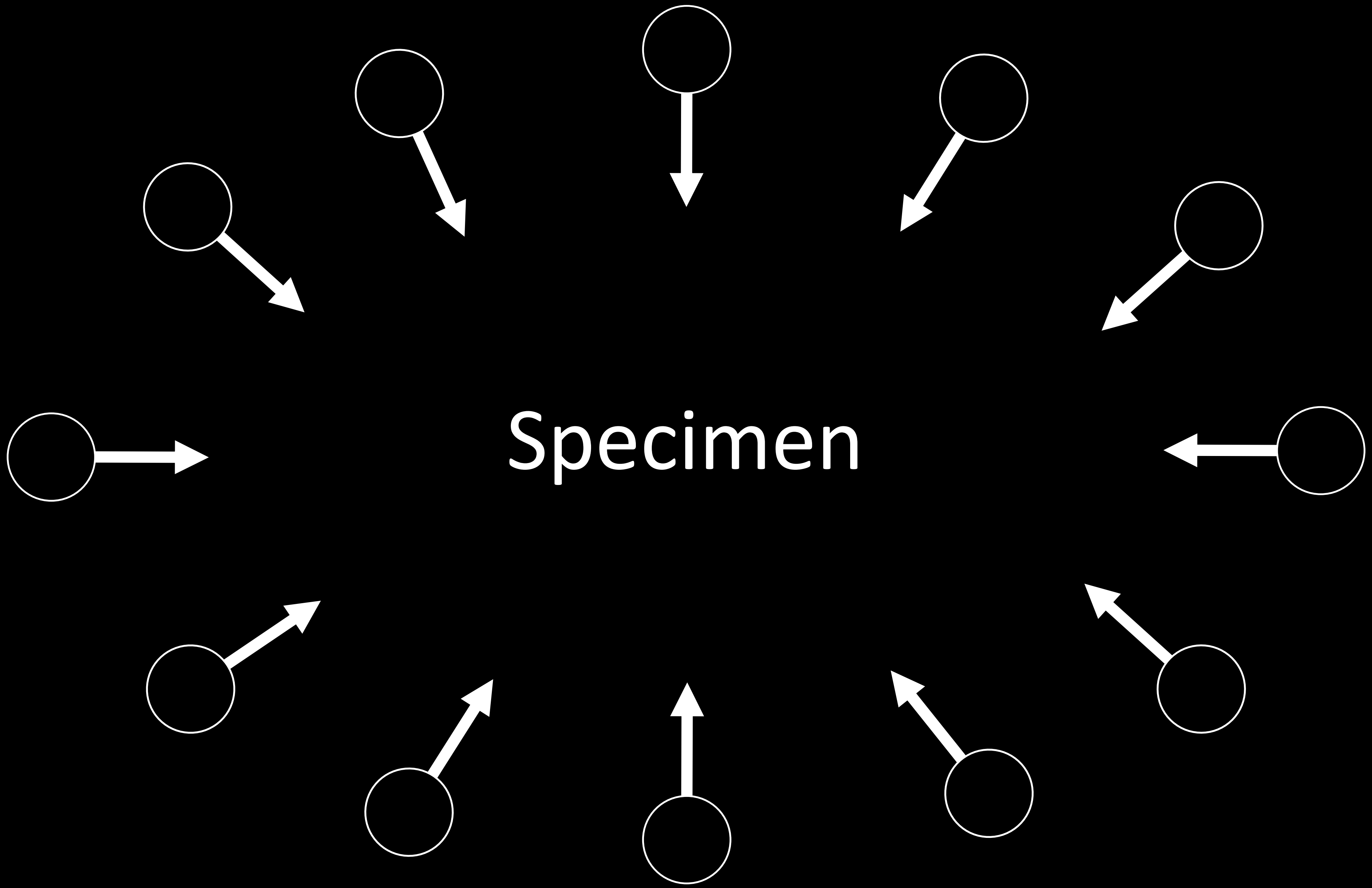
- AGIS Photoscan
- 3DSom
- Capturing Reality
- Python Photogrammetry Toolkit
- Theia Vision Library

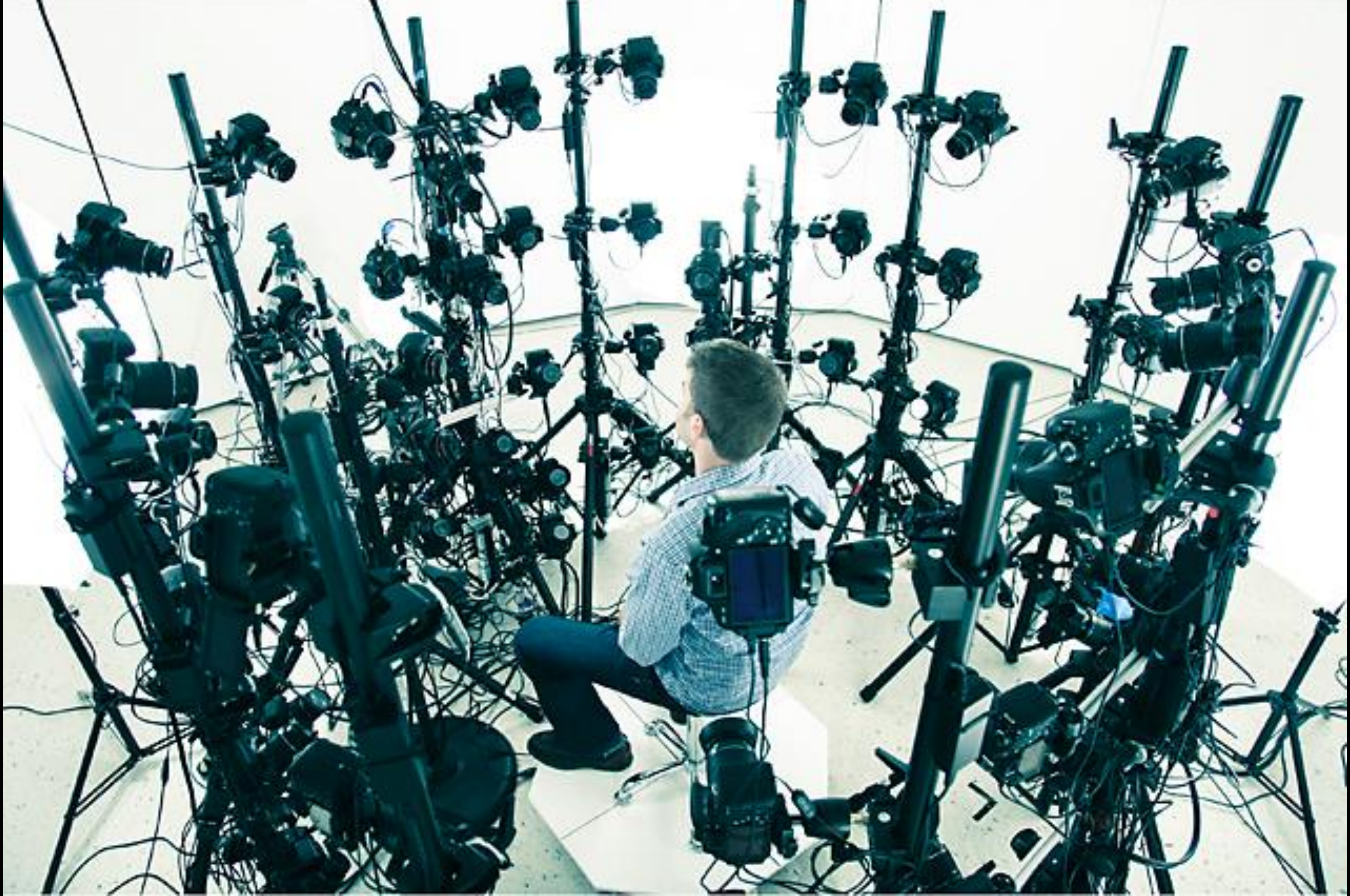
Photogrammetry

- Pros
  - Accuracy
  - Resolution
  - Color
  - Texture
  - Images can be re-processed as algorithms improve.
- Cons
  - Specimen Staging
  - Camera & Lighting Setup
  - Computationally Intensive
  - Post Processing

Best Practices

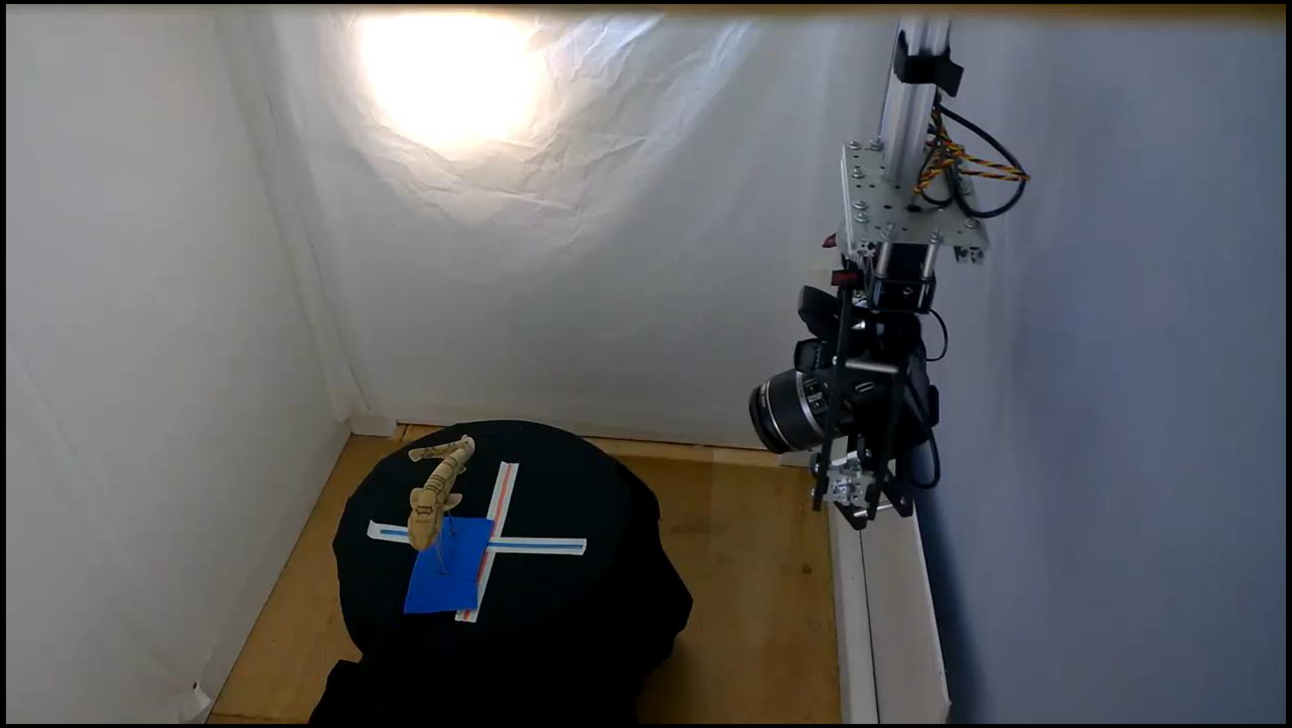
- Use high resolution photos
- Avoid blurred, out of focus photos of specimen
- Aperture should be high enough to result in sufficient focal depth
- Avoid specimens with un-textured, shiny, mirrored and/or transparent surfaces
- Avoid moving objects within the scene to be reconstructed
- Avoid any movement of the specimen


Specimen



Automating Image Acquisition for Photogrammetry

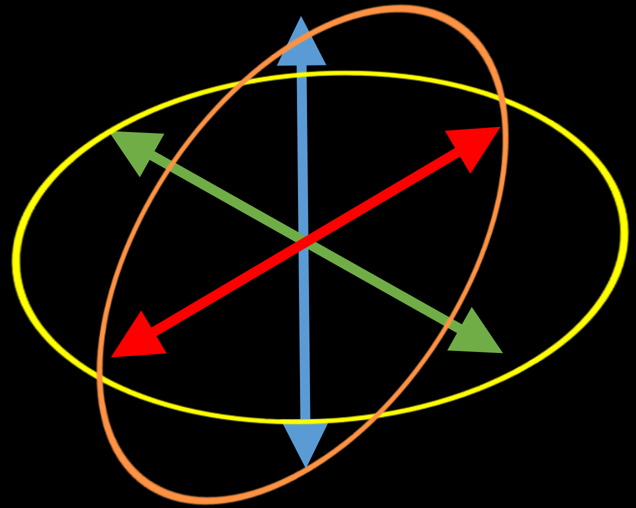
- Design Principles:
  - Fast, high-resolution image capture
    - target of 5 minutes per specimen @ 200 photos
  - Accurate positioning of cameras
  - Scalable
  - Cost effective



Early Prototype/Proof of Concept

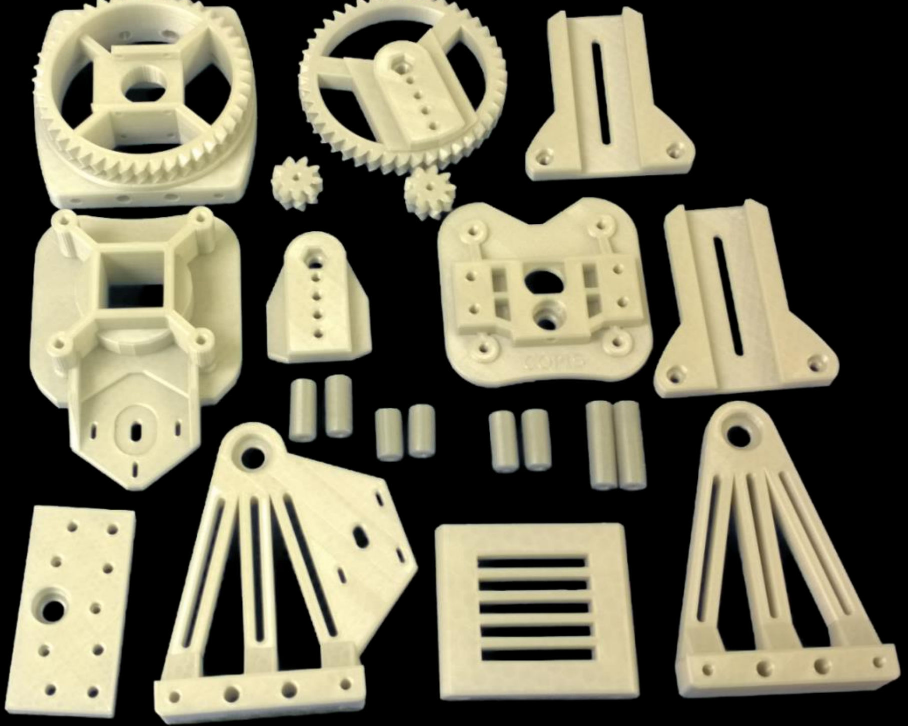
Current Status

5 Axis



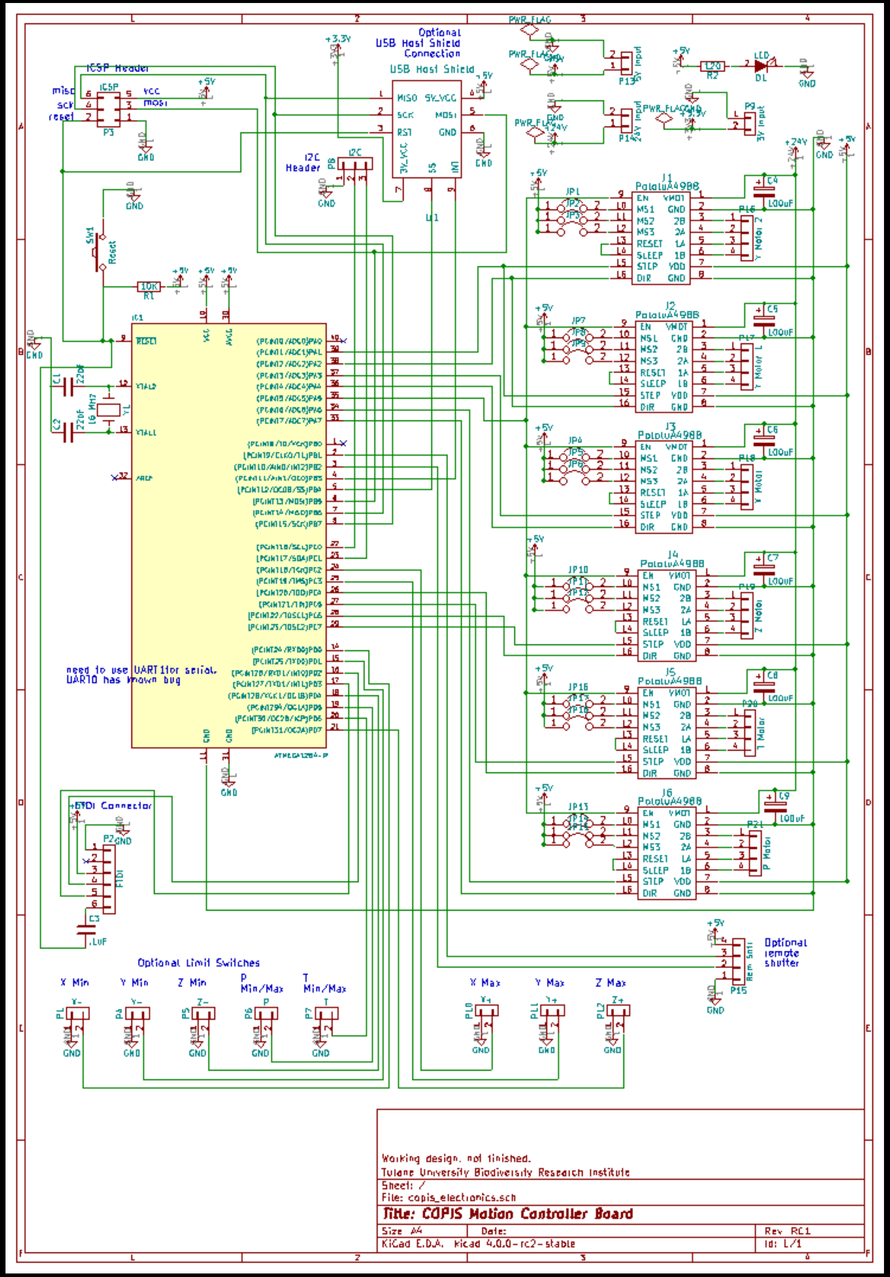
- Each axis driven by NEMA 17, 76.4 oz-in 1.8 degree/step stepper motors
- X & Y:
  - belt and pinion (.39um positional resolution at 1/16 stepping)
  - GT2 timing belt
  - 18 tooth pully
  - V-wheels on v-slotted aluminum extrusion
  - 1/8in Aluminum for linear carriage plates
  - paired motors on y-axis
- Z:
  - M8x1.25 threaded rod (6.25um positional resolution at full stepping)
  - V-wheels on v-slotted aluminum extrusion
  - 3D printed top and bottom plates
  - 1/8in Aluminum for linear carriage plates
- Pan & Tilt:
  - 3d printed Pan/Tilt
  - Modified from a design by Josh Sheldon (http://dcc.umd.edu/portfolio/jsheldon/)
  - 9:44 or 9:110 gear reductions (.023 or .0092 degree positional resolution at 1/16 stepping)
  - 1 inch hollow shaft for cabling to pass through

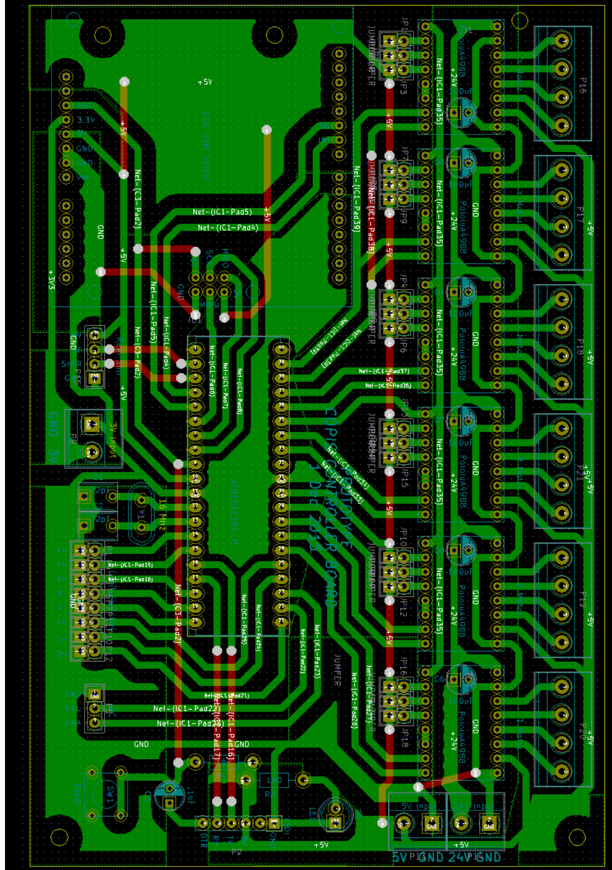
Pan and Tilt Camera Mount

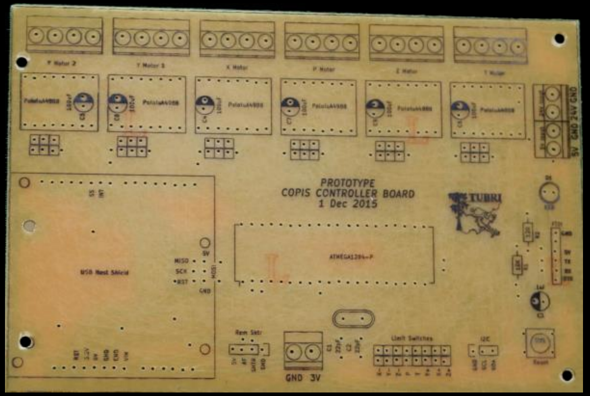


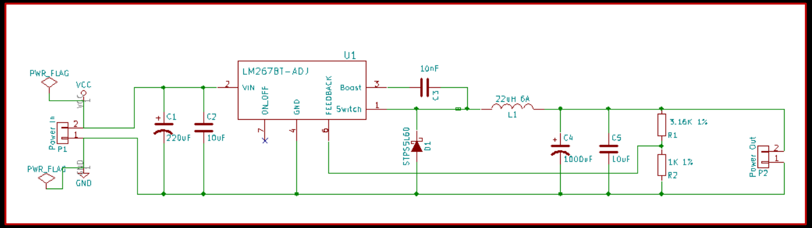
Electronics

- Designed for ease of production
  - DIY etching, milling or professional fabrication.
- ATMEGA 1284-P Microcontroller
- Allegro A4988 Motor Driver
  - Customizable Microstepping
    - Full, ½, ¼, 1/8 , 1/16 via jumpers
- Max 3421E USB Peripheral/Host Controller



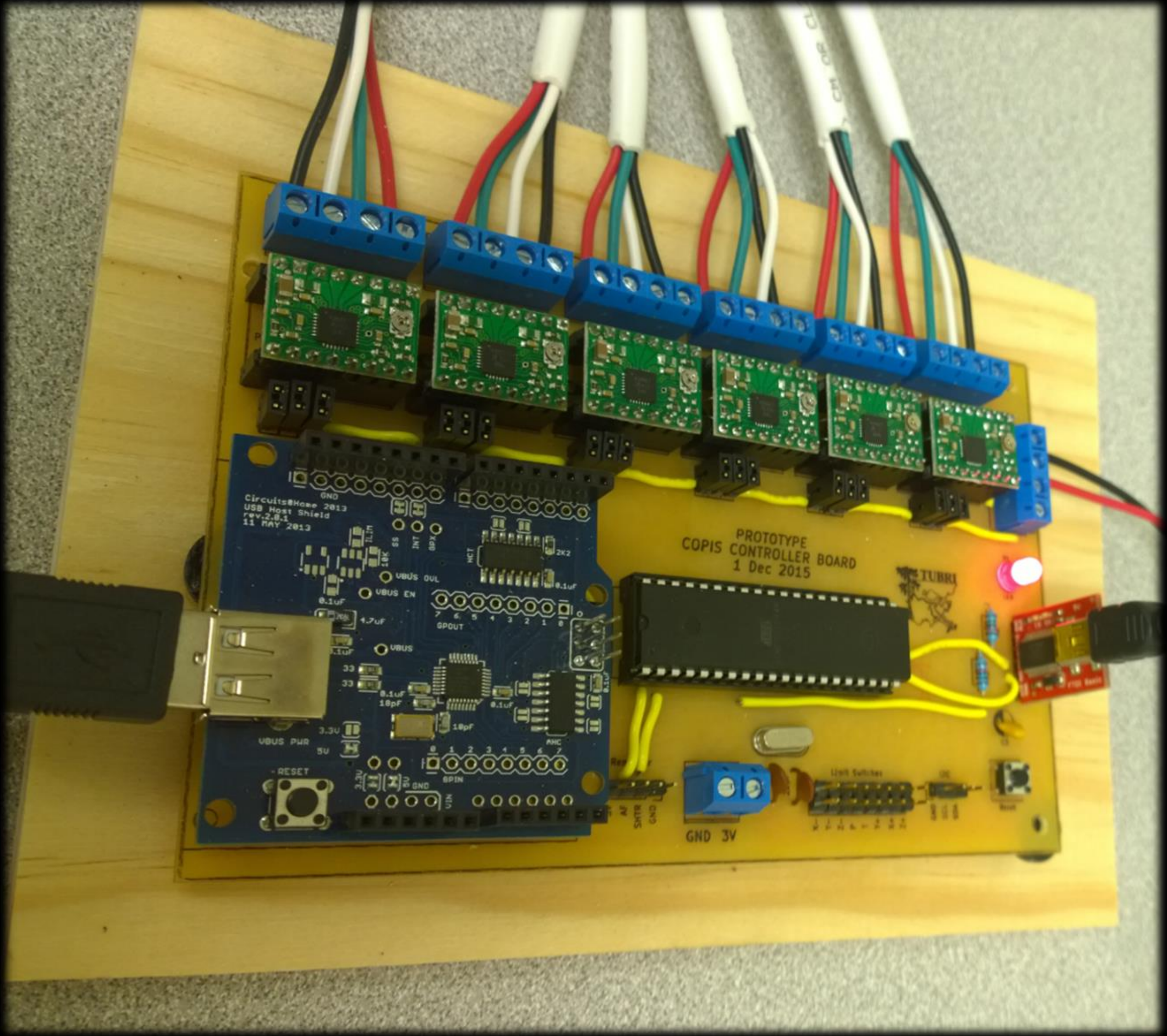




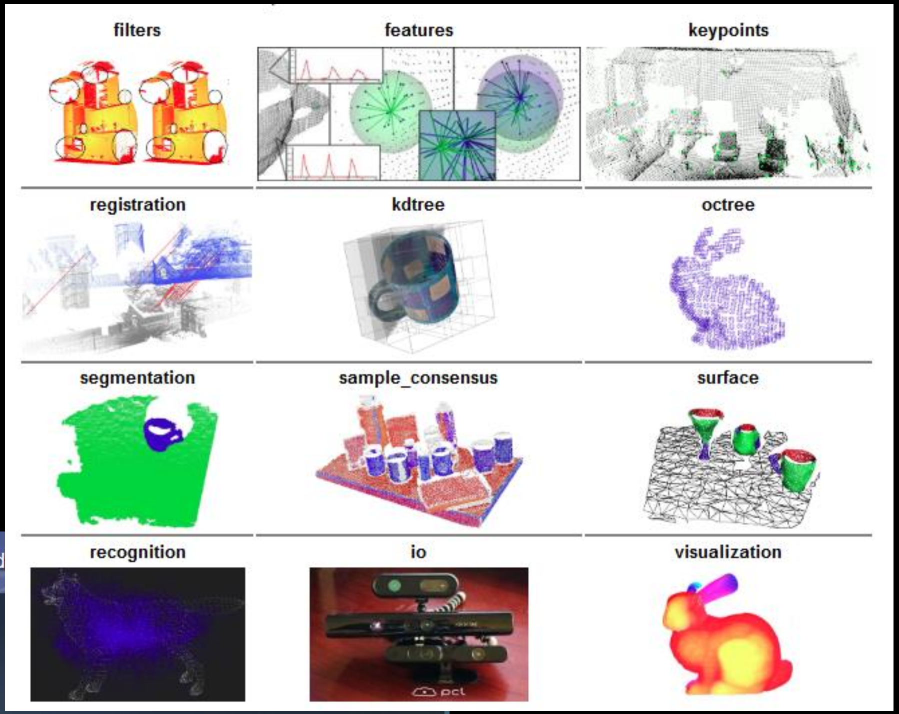


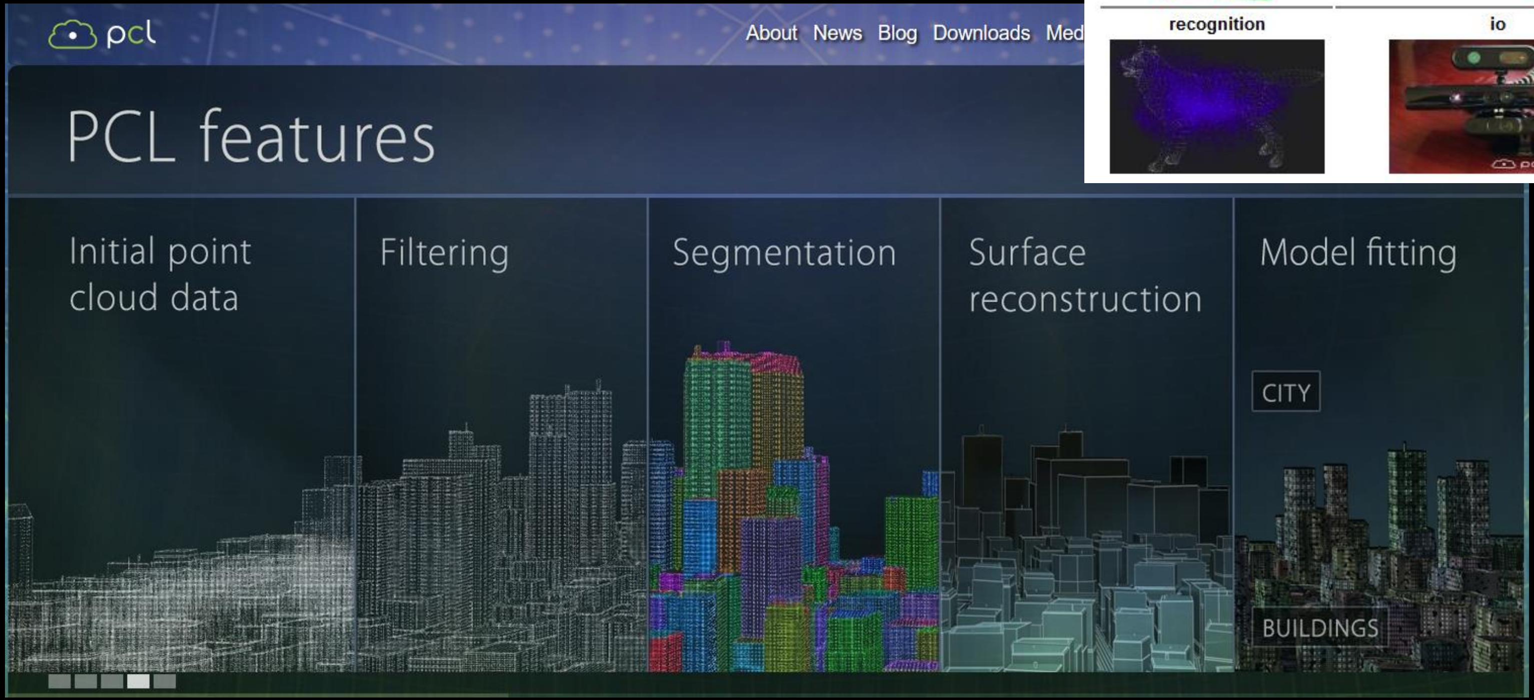
Mainboard Firmware

- Written in C++
- Camera Control via:
    - PTP – Picture Transfer Protocol
    - Remote Shutter & Autofocus
    - PC using manufacturer API’s
- X, Y, Z, Pan & Tilt min/max limit switches.
- Multi-board communication via I2C serial bus
- Motion control commands based on a variation of G-Code with extensions for Pan/Tilt and camera control.



Utilizing Point Clouds





copis.tubri.org

Funding provided by:

INSTITUTE of Museum and Library SERVICES