Computational Geometry and Archeological Refits

Nzingha Joseph

Introduction

Summer research at **Carleton College** with **Prof. Rob Thompson** (and fellow student **Evelene Zhang)**

Prototyping automatic bone refitting based on existing mathematical ideas

Refitting (and why it matters)

Refitting is the **reassembly of bone/stone fragments**.

Allows for a better understanding of early human life.

Impractical due to funding and time

Automatic refitting

Make 3D scans of bone fragments

Input 3D scans of bones

Analyze the geometry of bones

Find best alignment of the bones

Apply motions to achieve this alignment

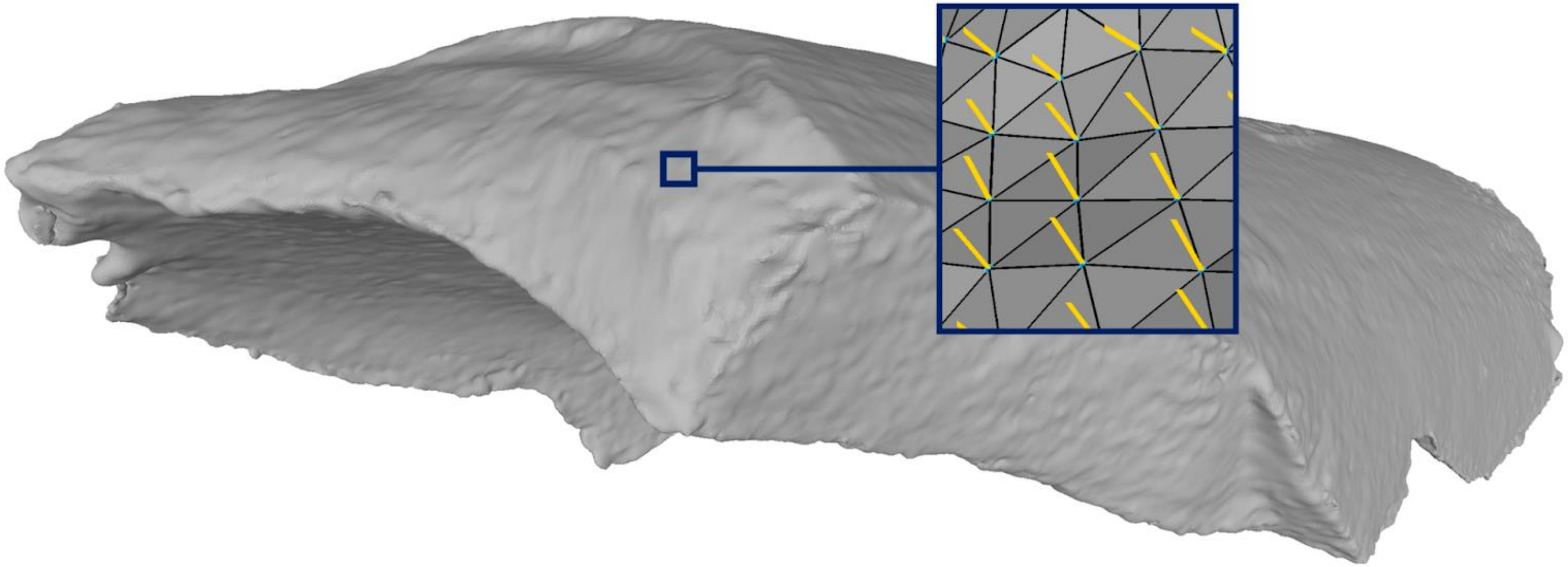
Output aligned bones!

Our input data

Provided by the **AMAAZE Consortium at the University of Minnesota**

.ply “point cloud” file format

Vertices
Edges
Normal vectors
Triangular faces



Our input data

“Archaeological” faces

Only **break faces** are used in our algorithm

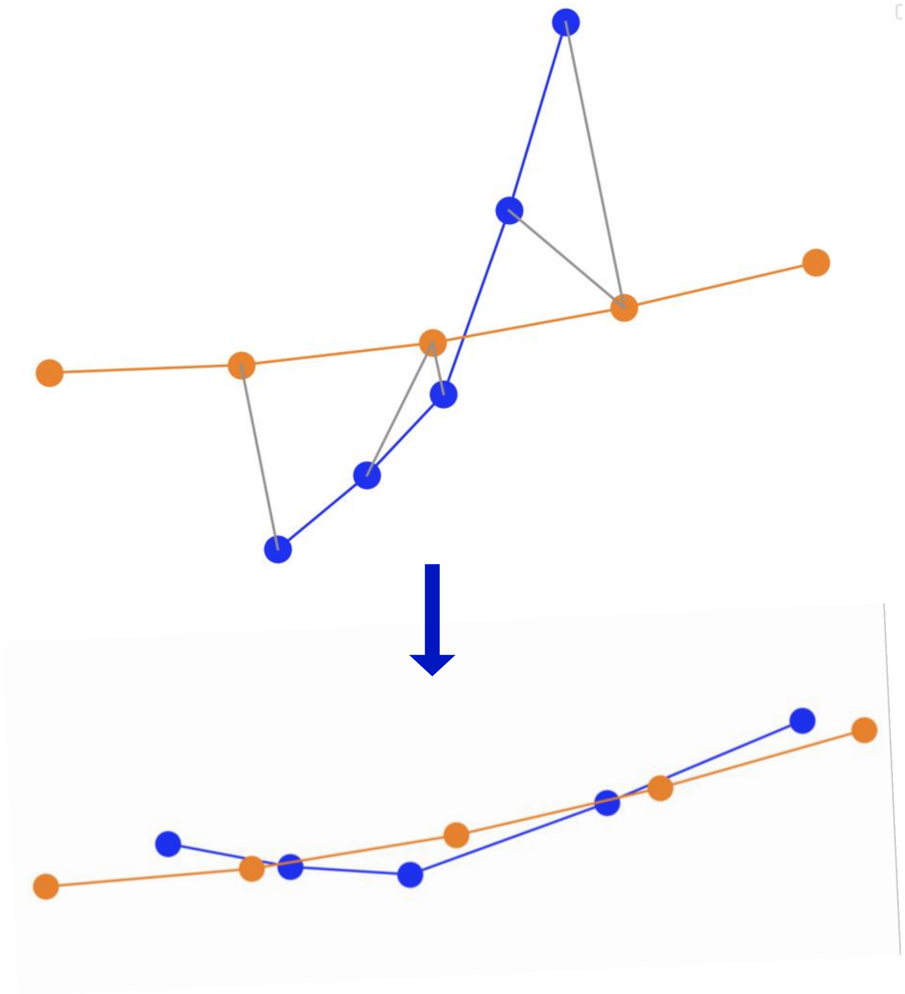
Iterative Closest Point (ICP)

Let one shape be **stationary** and let the other be **moving**.

Pair points.

Find the **minimizing transformation**.

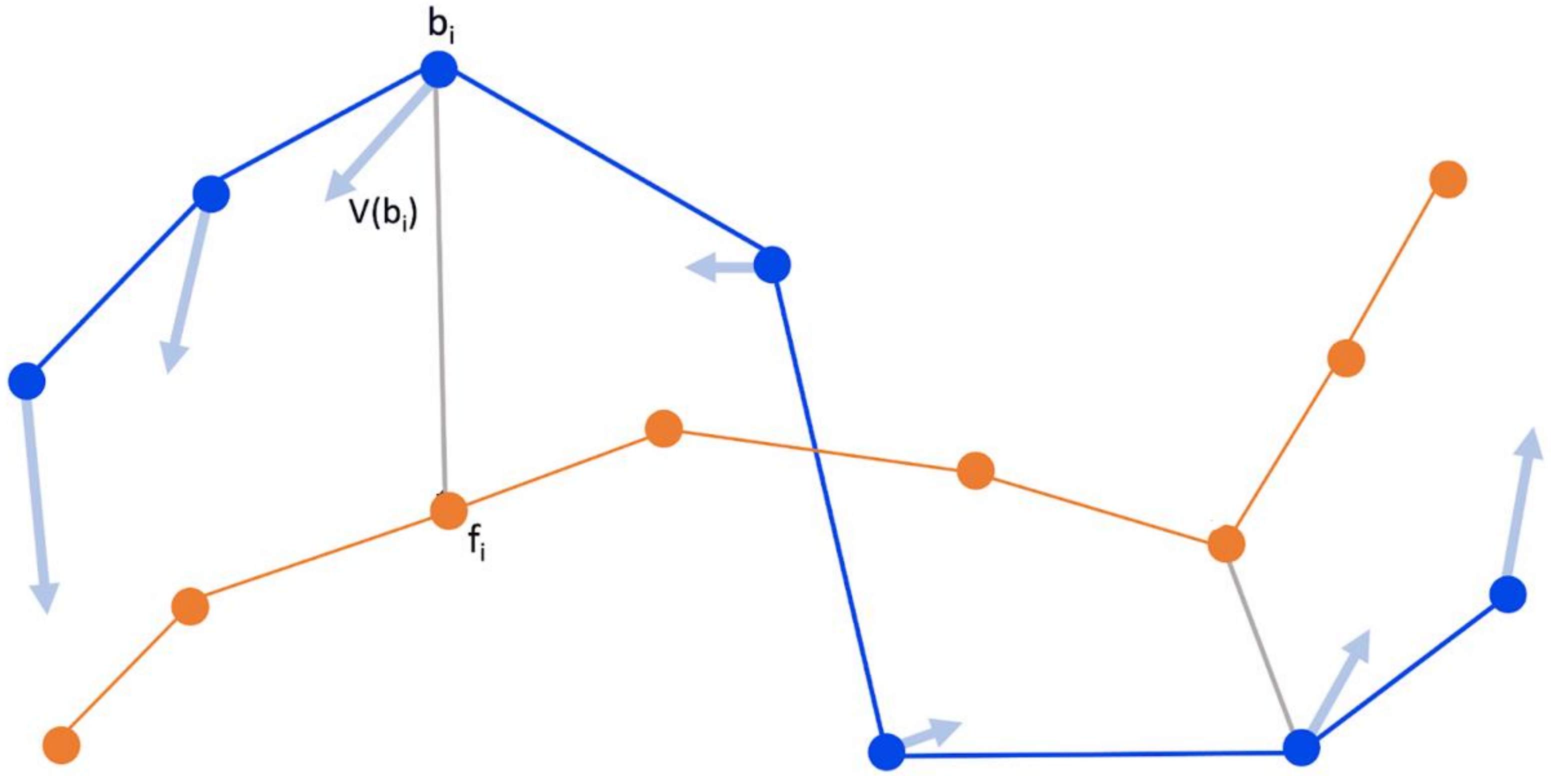
Iterate!



Iterative Closest Point (ICP)

Motion found using an optimization problem:

$$\min_{(\mathbf{c},\bar{\mathbf{c}})} \sum_{\mathbf{i}=1}^{\mathbf{n_B}} ||\mathbf{b_i} + \mathbf{v}(\mathbf{b_i}) - \mathbf{f_i}||^2 \qquad \mathbf{v}(\mathbf{b_i}) = \bar{\mathbf{c}} + \mathbf{c} \times \mathbf{b_i}$$

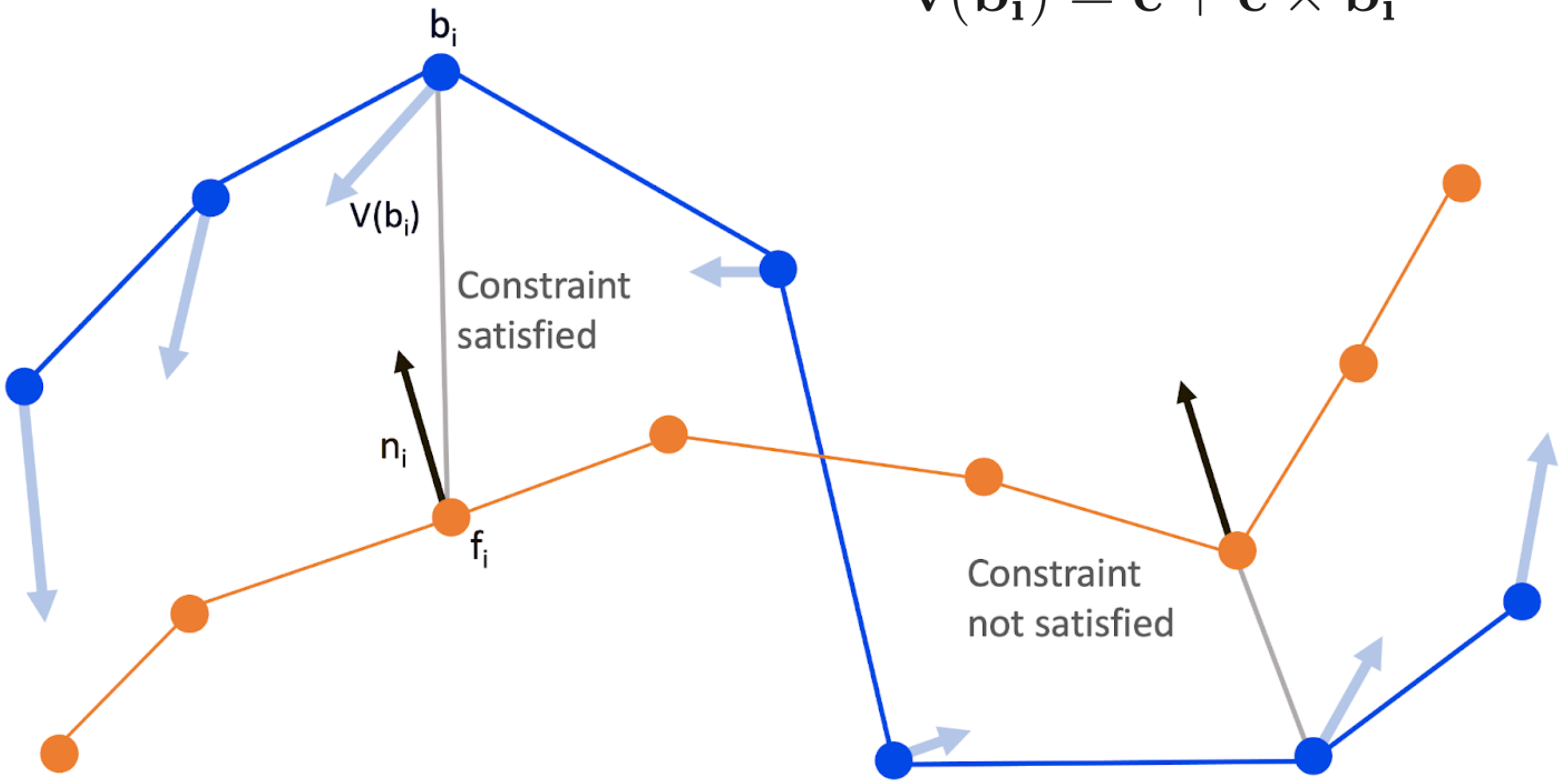


Overlap Problem

Solving the overlap problem

$$\min_{(\mathbf{c},\bar{\mathbf{c}})} \sum_{\mathbf{i}=1}^{\mathbf{n_B}} ||\mathbf{b_i} + \mathbf{v}(\mathbf{b_i}) - \mathbf{f_i}||^2 \ \mathbf{s.t.}\ \mathbf{n_i^T} \cdot (\mathbf{b_i} + \mathbf{v}(\mathbf{b_i}) - \mathbf{f_i}) > \mathbf{0}, \forall \mathbf{i}$$

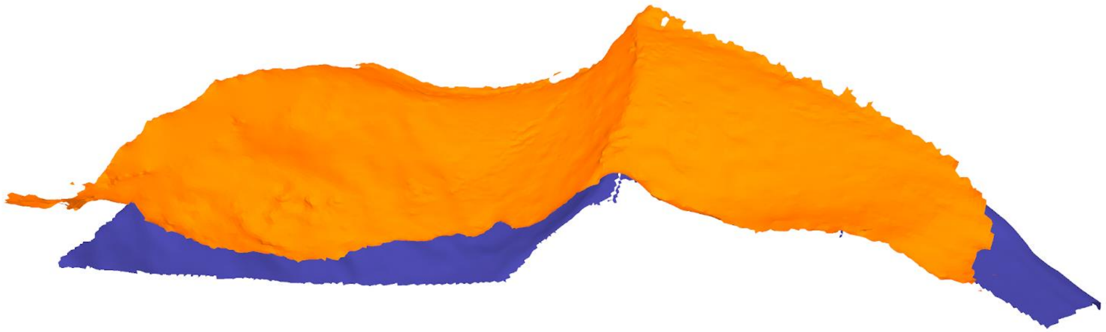
$$\mathbf{v}(\mathbf{b_i}) = \bar{\mathbf{c}} + \mathbf{c} \times \mathbf{b_i}$$



Overlap problem solved!

Without constraint

With constraint

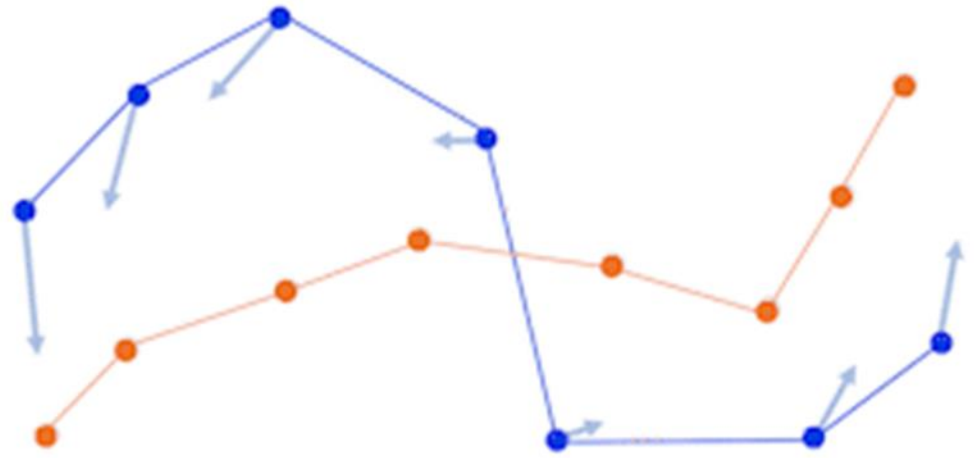


Finding the rigid motion

Any rigid transformation is given by a uniform helical motion.

Our optimization finds a **velocity vector field**:

$$\mathbf{v}(\mathbf{b_i}) = \overline{\mathbf{c}} + \mathbf{c} \times \mathbf{b_i}$$



We “integrate” to find a rigid motion.

Finding a helical motion

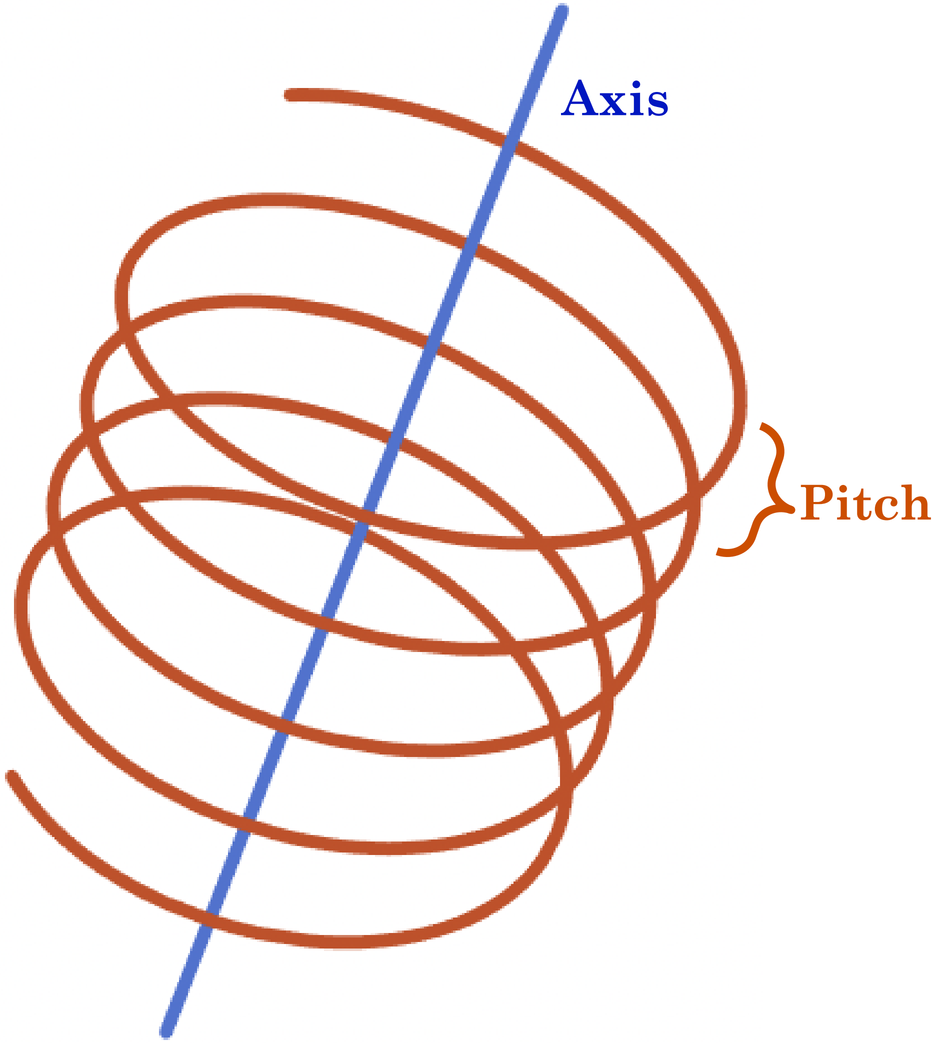
Direction vector: $\mathbf{c}$

Distance between turns of the helix: $\mathbf{p} = \frac{\mathbf{c}\cdot\bar{\mathbf{c}}}{\mathbf{c}\cdot\mathbf{c}}$

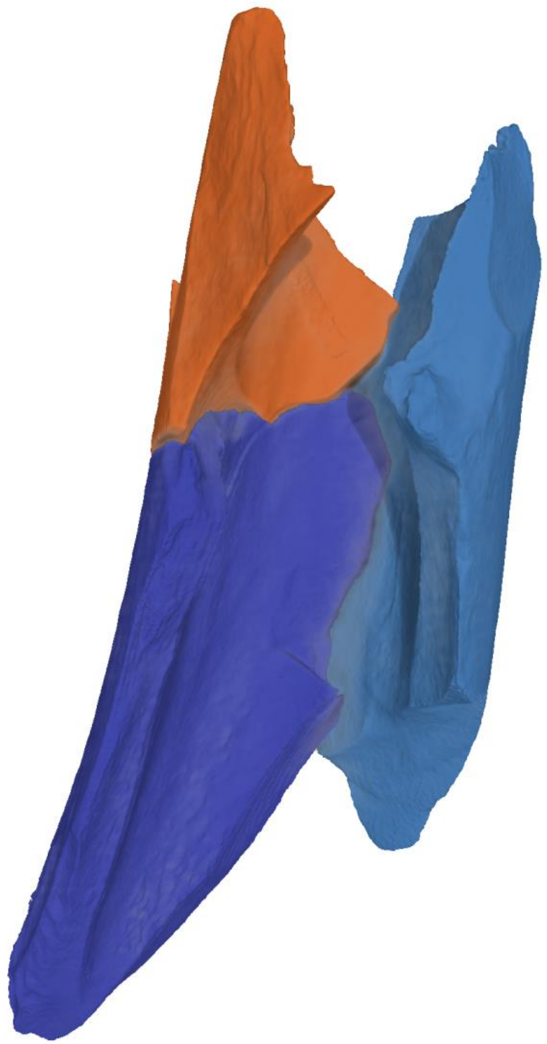
Uniform helical motion around the z axis:

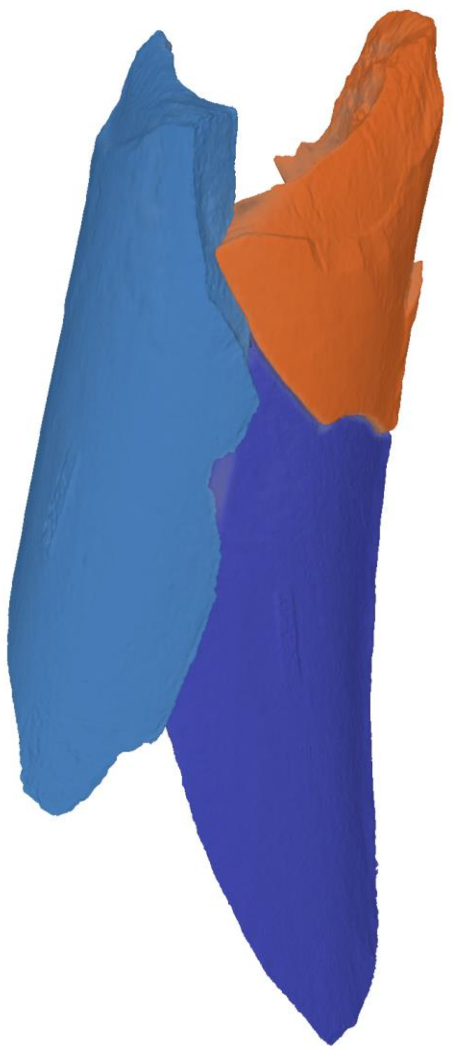
$$\mathbf{x} \mapsto \mathbf{x}(t) = \begin{bmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ p \cdot t \end{bmatrix}$$

We **transform, apply motion, transform back**.



Results





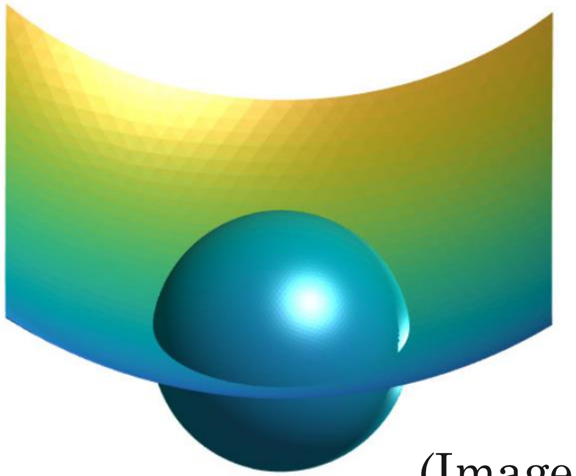
Ongoing Work

Multipiece refits:

$$\min_{(\bar{\mathbf{c}}_1,\mathbf{c}_1,\ldots,\bar{\mathbf{c}}_n,\mathbf{c}_n)} \sum_{(j,k)\in\mathcal{A}} \sum_{\mathbf{b}\in A_j} \left[\left(\mathbf{b}+\mathbf{v}_{jk}(\mathbf{b})-\mathbf{f}_b^k\right)^T \cdot \mathbf{n}_b^k\right]^2$$

$$\text{subject to} \quad \left(\mathbf{b}+\mathbf{v}_{jk}(\mathbf{b})-\mathbf{f}_b^k\right)^T \cdot \mathbf{n}_b^k > 0 \qquad \forall (j,k)\in\mathcal{A},\ \mathbf{b}\in A_j.$$

Spherical volume invariant:



(Image from O'Neill et al.)

Acknowledgements

A sincere “thank you” to:

My research advisor, Professor Rob Thompson
My research partner Evelene Zhang
The AMAAZE Consortium at the University of Minnesota
Dr. Katrina Yezzi-Woodley and Dr. Peter Olver

The Carleton College Summer Science Fellowship, the Provost’s Office, and the Harry A. and Margaret D. Towsley Foundation

All others who helped support a wonderful summer of research!

References

Pottmann, Helmut, et al. "Geometry and convergence analysis of algorithms for registration of 3D shapes." *International Journal of Computer Vision* 67.3 (2006): 277- 296.

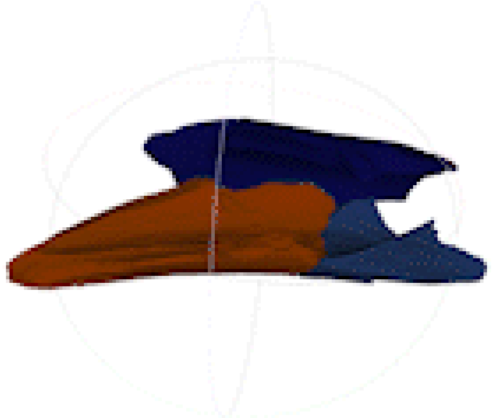
Hofer, Michael, et al. "3D shape recognition and reconstruction based on line element geometry." *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. IEEE, 2005.

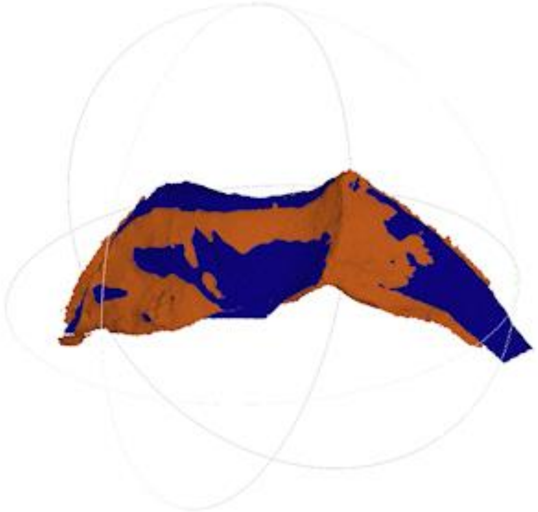
Flöry, Simon. *Constrained matching of point clouds and surfaces*. na, 2009.

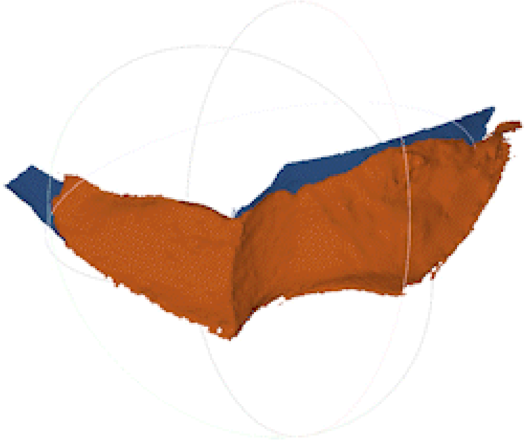
Pottmann, Helmut, and Johannes Wallner. *Computational line geometry*. Berlin-Heidelberg, New York: Springer, 2001.

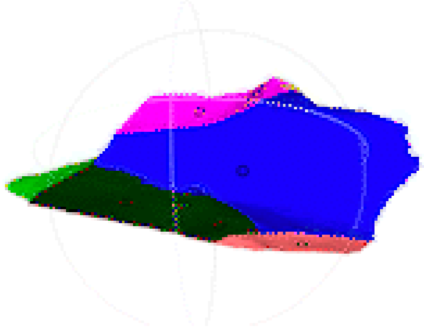
Calder, Jeff, AMAAZETools, (2020), jwcalder, https://github.com/jwcalder/AMAAZETools/tree/main/amaazetools

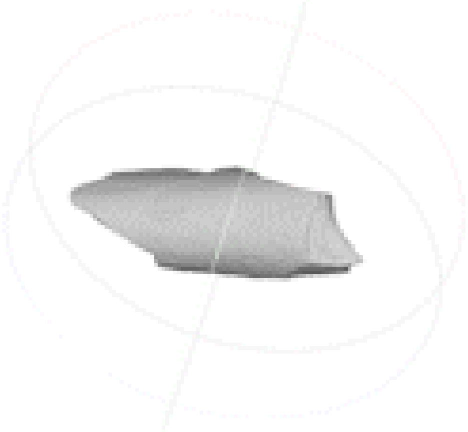
O'Neill, Riley CW, et al. "Computation of circular area and spherical volume invariants via boundary integrals." SIAM Journal on Imaging Sciences 13.1 (2020): 53-77.











Thank you!

Any questions?