Single Viewpoint Symmetry-Based Model Completion for Efficient 3D Acquisition

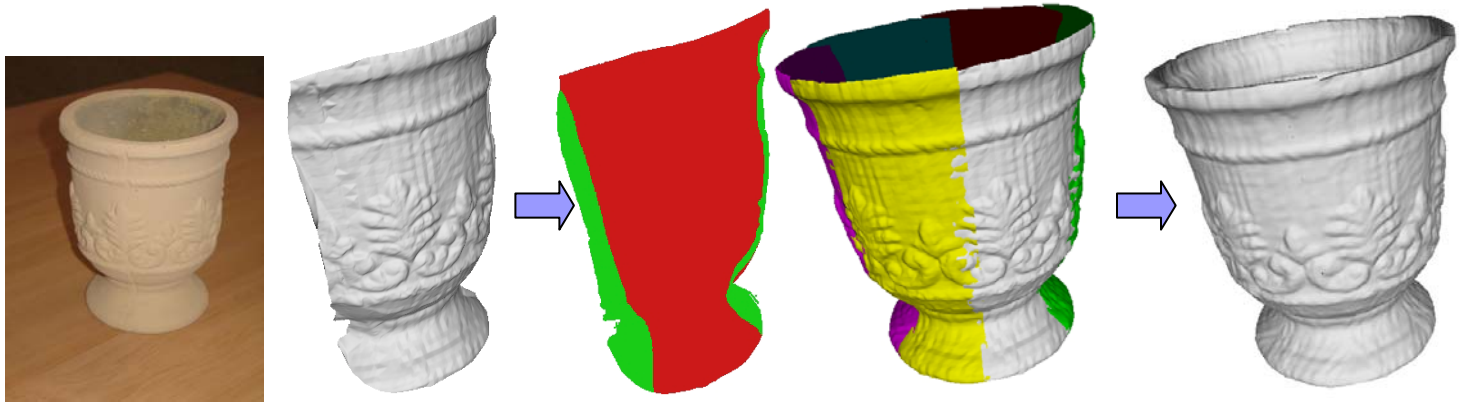
Alvin Law
Daniel Aliaga
Purdue University

Motivation

- Goal:
  - Capture a complete model from a single viewpoint
- Problem:
  - No information about occluded regions of an object
  - Need many images to obtain a complete model

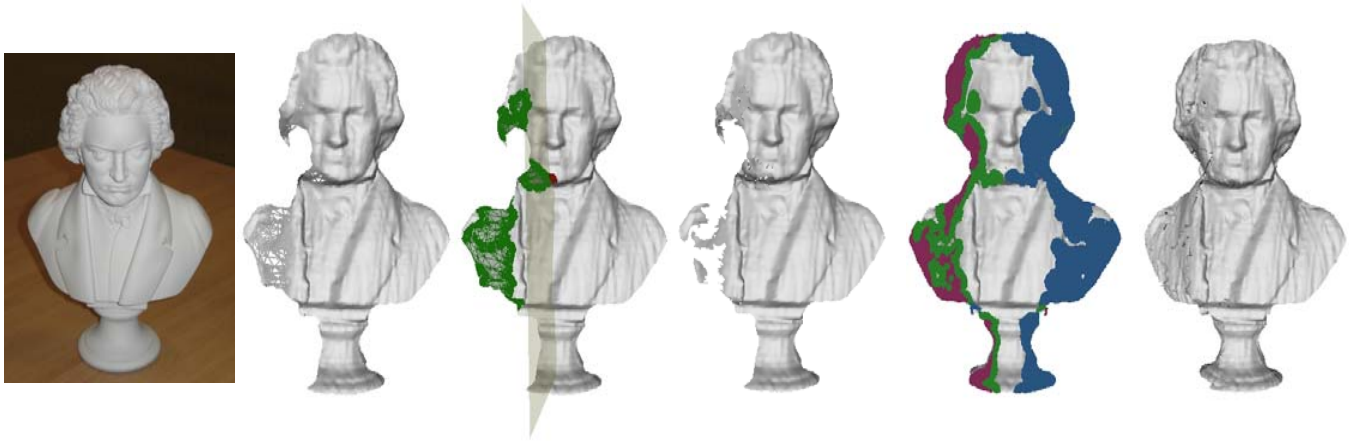
Observation

- Discovering and using the symmetry can help capture complete object models
  - Fill-in interior holes
  - Extend the object “border”
  - Complete the backside

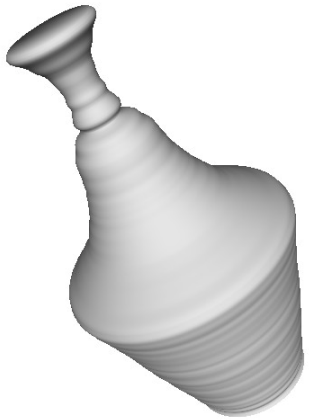


Model Completion

- Steps
  - Input: initial capture from single viewpoint
  - Discovery symmetry of object
  - Use symmetry to add geometry to unseen regions of model
  - Zip/watertight added regions of model



Symmetry

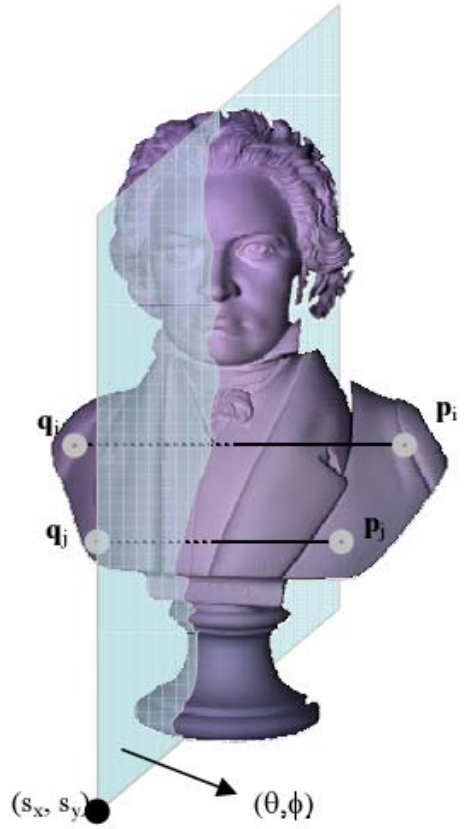


- Three cases supported
  - Bilateral
  - Radial
  - Surface-of-Revolution

Bilateral Symmetry Detection

- Compute symmetry plane (s_x, s_y, theta, phi)
  - Minimize: total symmetric distance between (**p**, **q**) pairs

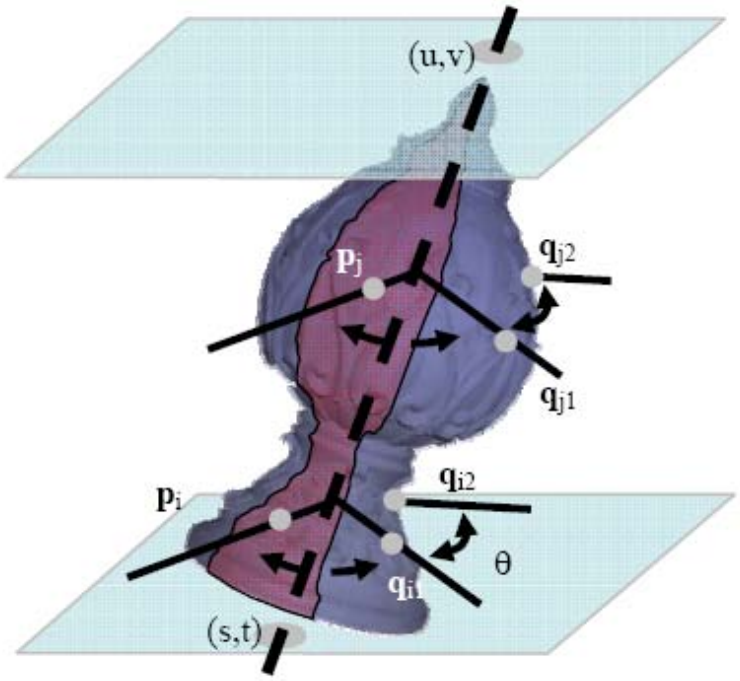
$$\sum_i \left\| q_i - \left(p_i + 2\left(\begin{bmatrix} \cos\theta\sin\phi \\ \sin\theta\sin\phi \\ \cos\phi \end{bmatrix} \cdot \left(\left(\begin{bmatrix} s_{b_x} \\ s_{b_y} \\ 0 \end{bmatrix} - p_i \right) \right) \begin{bmatrix} \cos\theta\sin\phi \\ \sin\theta\sin\phi \\ \cos\phi \end{bmatrix} \right) \right) \right\| \to 0$$



Radial Symmetry Detection

- Compute axis of rotation (s, t, u, v)
  - Minimize: total distance between original point and replicated point on proposed model

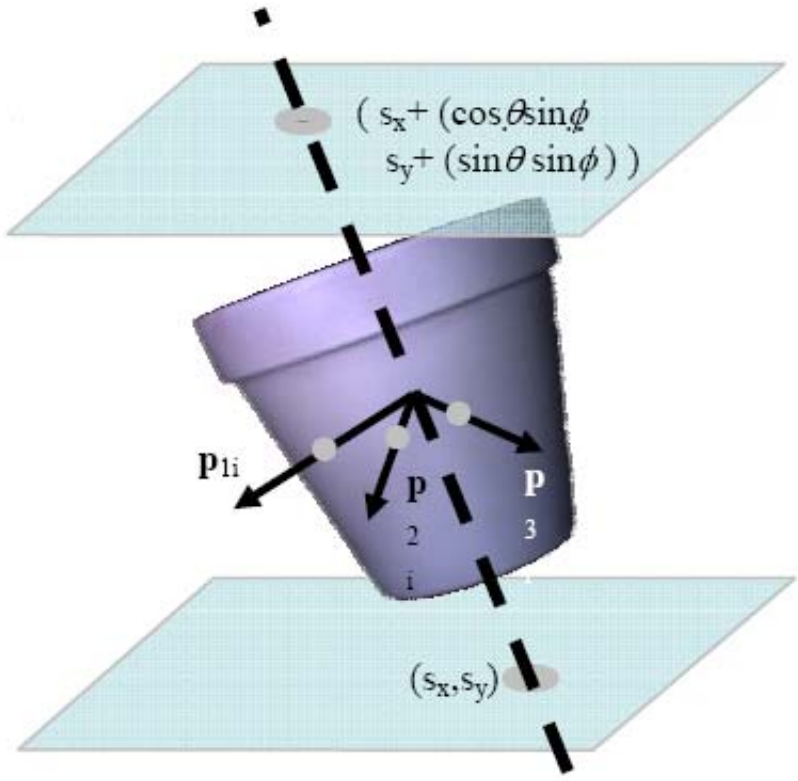
$$\sum_i \sum_j \left\| q_{ij} - R_{j\theta_n}(u,v)(p_i - [s,t,0]) \right\| \to 0$$



SOR Symmetry Detection

- Compute axis of rotation and disc radii
- Minimize: radial distance error between disc and points on disc

$$\sum_i \sum_j \left(\frac{\left\| (s_2 - s_1) \times (s_1 - p_{ij}) \right\|}{\left\| s_2 - s_1 \right\|} - r_j \right)^2 \to 0$$

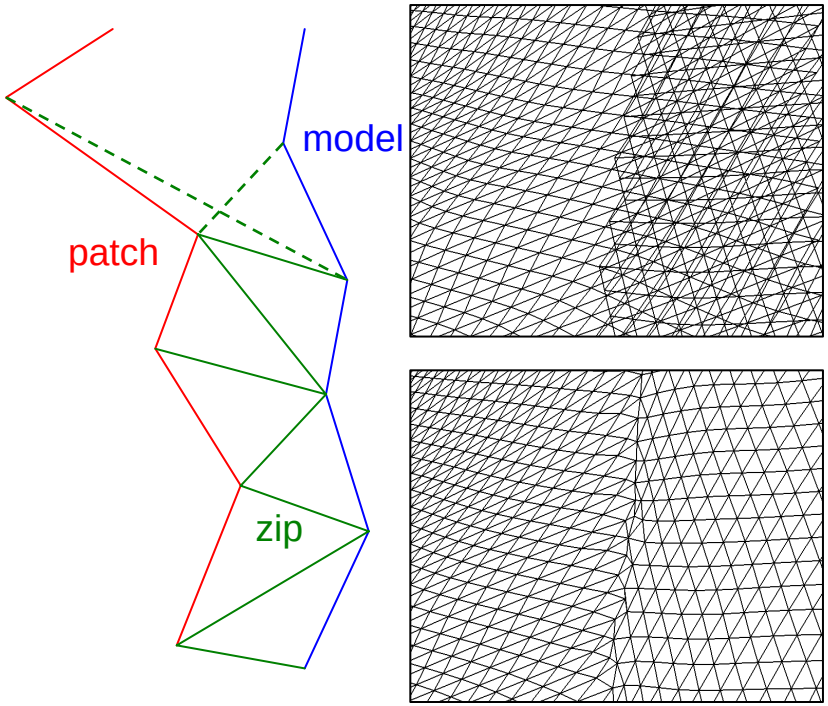


Adding Geometry

- Bilateral
  - Interior patches
  - Extending borders
- Radial
  - Repeat face n times
- Surface of Revolution
  - Create synthetic object

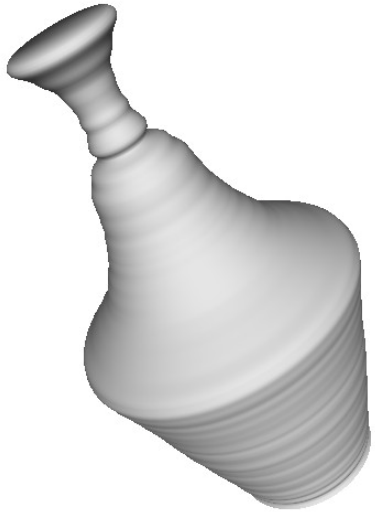
Zipping

- Merge together patches and model
  - Classify all patch triangles
    - Inside
    - Outside
    - Split
  - Add zipping triangles
    - Minimize gap distance



Watertight

- Two means
  - Seal ends of model
  - Create an inset



Resulting Models