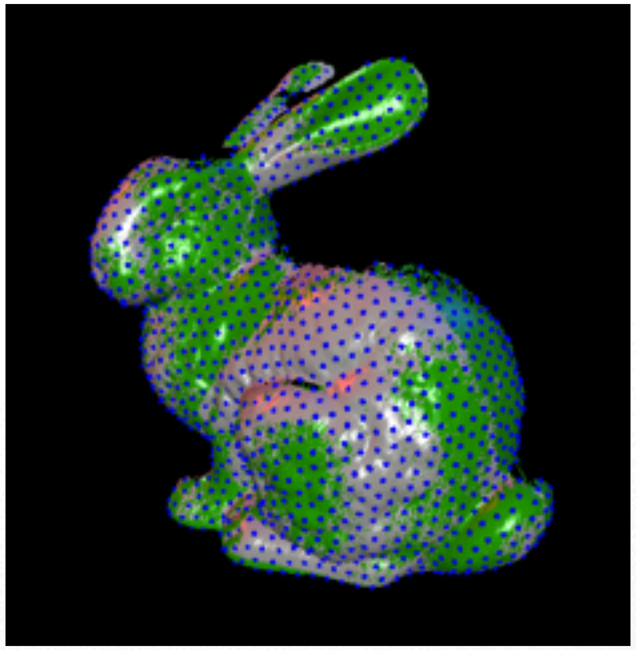
Exercise 2. Registration



Pei-Lun Hsieh

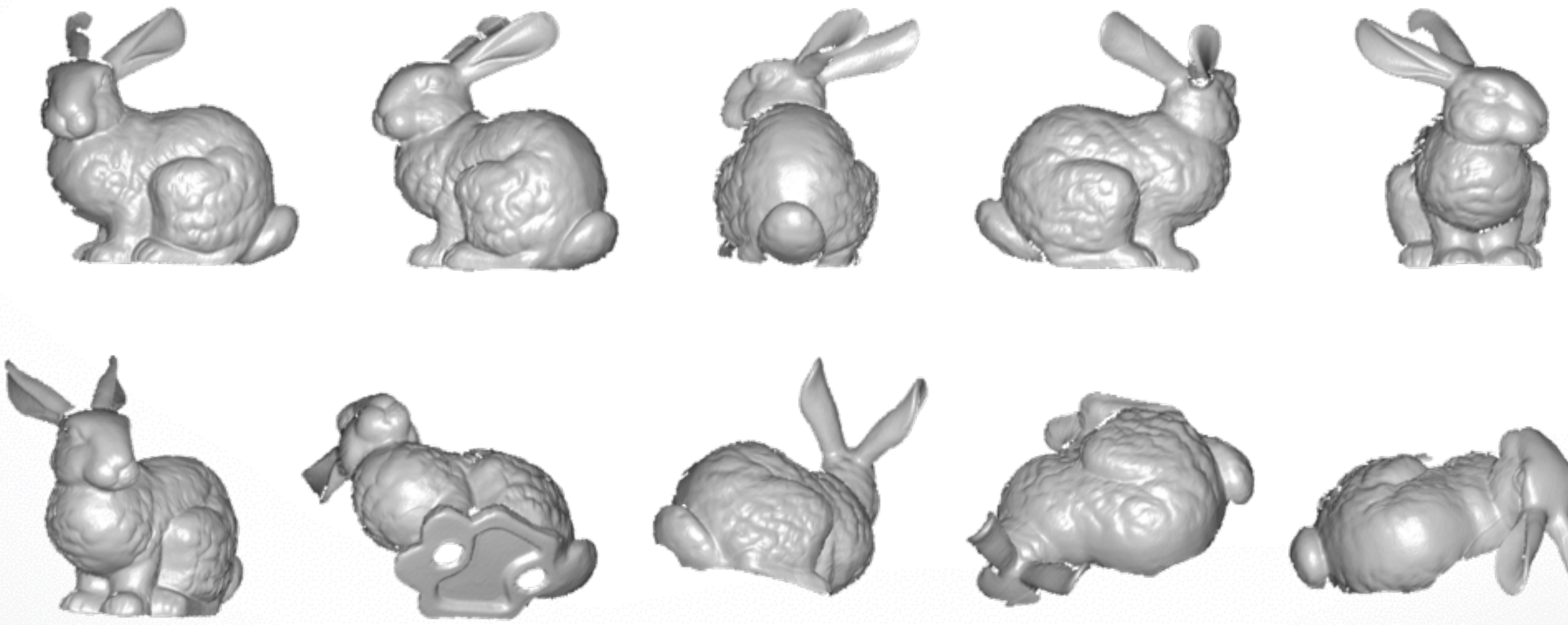
http://cs599.hao-li.com

Rigid Registration

- **Selecting** source points
- **Matching** points to the target mesh
- **Weighting** the correspondences
- **Rejecting** bad pairs
- Compute **error metric**
- **Minimize** error metric

Exercise 2

- Perform rigid registration between 10 scans of the Stanford bunny



Exercise 2

- Demo
  - `‘SHIFT’` + mouse controls: manual alignment for an initial transformation
  - `‘r’`: perform single registration step with point to point distance minimization
  - `‘SPACE’`: perform single registration step with point to plane distance minimization
  - `‘n’`: load next scan

Exercise 2

- Getting it compiled
- Subsampling
- Bad pairs rejection
- Point to point optimization
- Point to plane optimization

Getting It Compiled

- CMake, OpenGL, OpenMesh
- ANN (Approximate Nearest Neighbor)
  - efficient closest point lookup using kd-tree

Subsampling

- Uniform subsampling within a given radius `subsampleRadius`
- `RegistrationViewer::subsample()` in RegistrationViewer.cc

Bad Pairs Rejection

- Closest points are computed using ANN
- Prune correspondences based on
  - distance threshold
  - normal compatibility
- `RegistrationViewer::calculate_correspondences()` in RegistrationViewer.cc

Point to Point Optimization

- Minimize $E = \sum_{i=1}^{N} \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i\|_2^2$

  by solving a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$

- `Registration::register_point2point()` in Registration.cc

Euler Angles

- Three elemental rotations:

$$\mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \mathbf{R}_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \mathbf{R}_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Any rotation matrix can be decomposed as a product of elemental three rotation matrix

$$\mathbf{R} = \mathbf{R}_z(\gamma)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha) = \begin{bmatrix} c_\gamma c_\beta & -c_\alpha s_\gamma + c_\gamma s_\beta s_\alpha & s_\gamma s_\alpha + c_\gamma c_\alpha s_\beta \\ c_\beta s_\gamma & c_\gamma c_\alpha + s_\gamma s_\beta s_\alpha & c_\alpha s_\gamma s_\beta - c_\gamma s_\alpha \\ -s_\beta & c_\beta s_\alpha & c_\beta c_\alpha \end{bmatrix}$$

$$c_\alpha = \cos\alpha \quad s_\alpha = \sin\alpha$$

Linearized Transformation

- Linearized Euler angle

(assuming small rotation: $\cos\alpha = 1 \quad \sin\alpha = \alpha$)

$$\mathbf{R} = \begin{bmatrix} c_\gamma c_\beta & -c_\alpha s_\gamma + c_\gamma s_\beta s_\alpha & s_\gamma s_\alpha + c_\gamma c_\alpha s_\beta \\ c_\beta s_\gamma & c_\gamma c_\alpha + s_\gamma s_\beta s_\alpha & c_\alpha s_\gamma s_\beta - c_\gamma s_\alpha \\ -s_\beta & c_\beta s_\alpha & c_\beta c_\alpha \end{bmatrix} = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}$$

- Linearized transformation

$$\mathbf{x} = \begin{bmatrix} \alpha & \beta & \gamma & \mathbf{t}_x & \mathbf{t}_y & \mathbf{t}_z \end{bmatrix}^\top$$

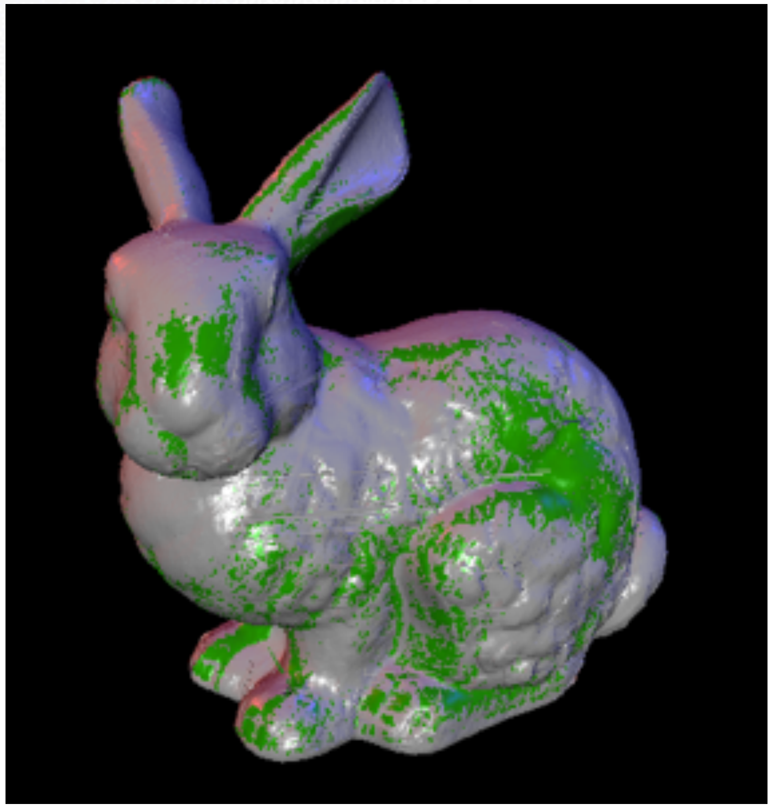
Point to Plane Optimization

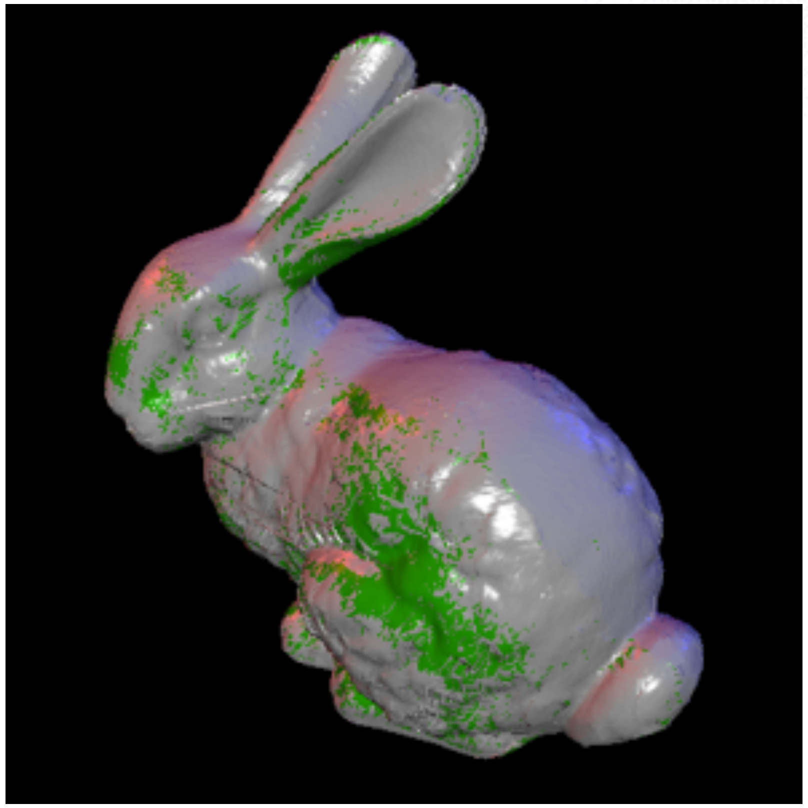
- Minimize $E = \sum_{i=1}^{N} \|\mathbf{n}_i^\top (\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i)\|_2^2$

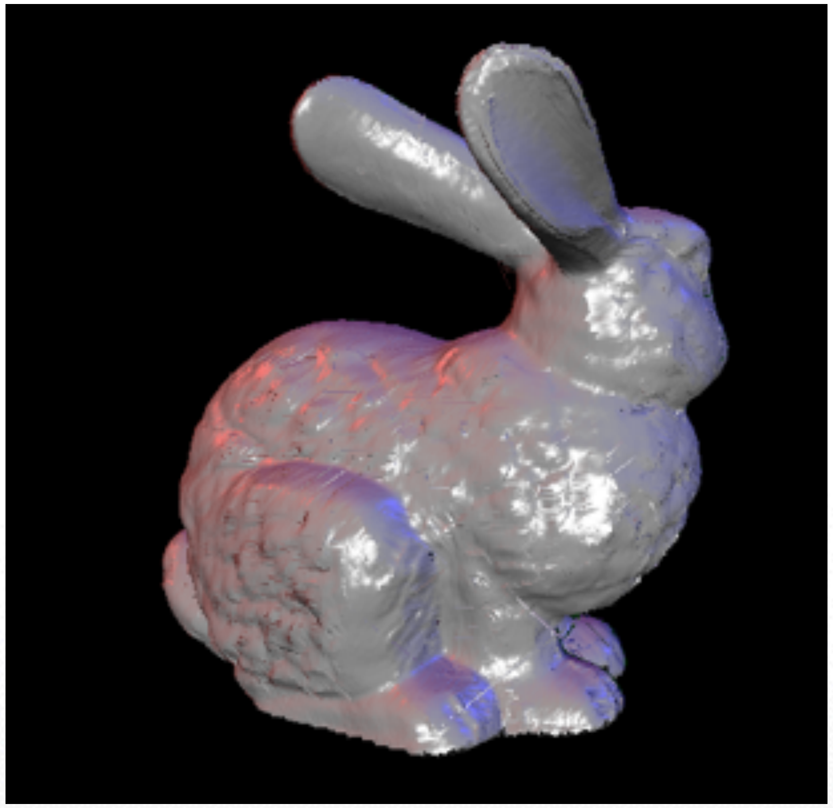
  by solving a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$

- `Registration::register_point2surface()` in Registration.cc

Results







Submission

- Deadline: **Feb 19, 2014 11:59pm**
- Upload a .zip compressed file named "Exercise2-YourName.zip" to
  - http://www.dropitto.me/usc-cs599dgp
  - password: `ididit`
- Include a "read.txt" file describing how you solve each exercise and the encountered problems

Contact

- Office Hours: Wednesday 11:30 - 13:30 SAL 219
- email: peilun.hsieh@usc.edu
- Highly recommended to post your question on Piazza:

  https://piazza.com/usc/spring2014/cs599dgp http://cs599.hao-li.com

Thanks!