MOHAMED BIN ZAYED UNIVERSITY OF ARTIFICIAL INTELLIGENCE

METAVERSE CENTER

CV 804: 3D RENDERING & GEOMETRY PROCESSING

WEEK 10 / EXERCISE 5:Surface Registration

Lecturer: Hao Li

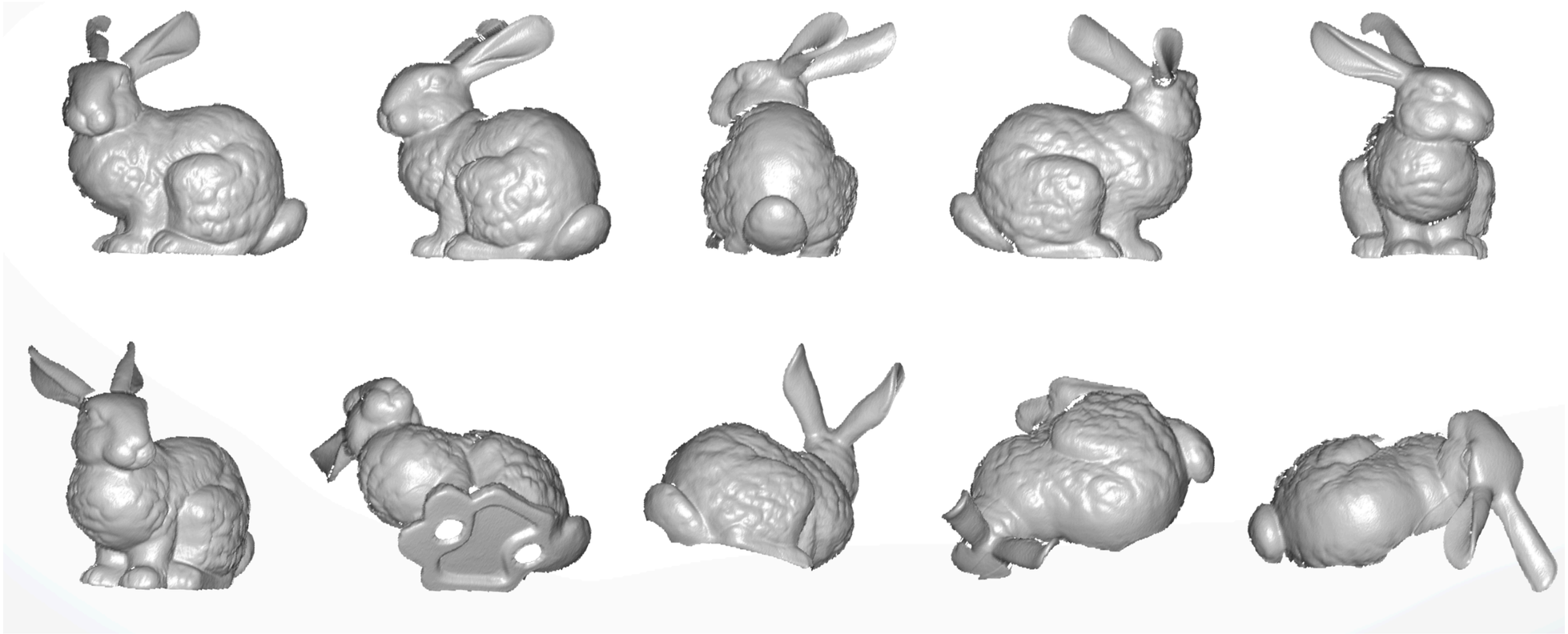
TAs: Phong Tran, Long Nhat

01/31/24

Exercise 5

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

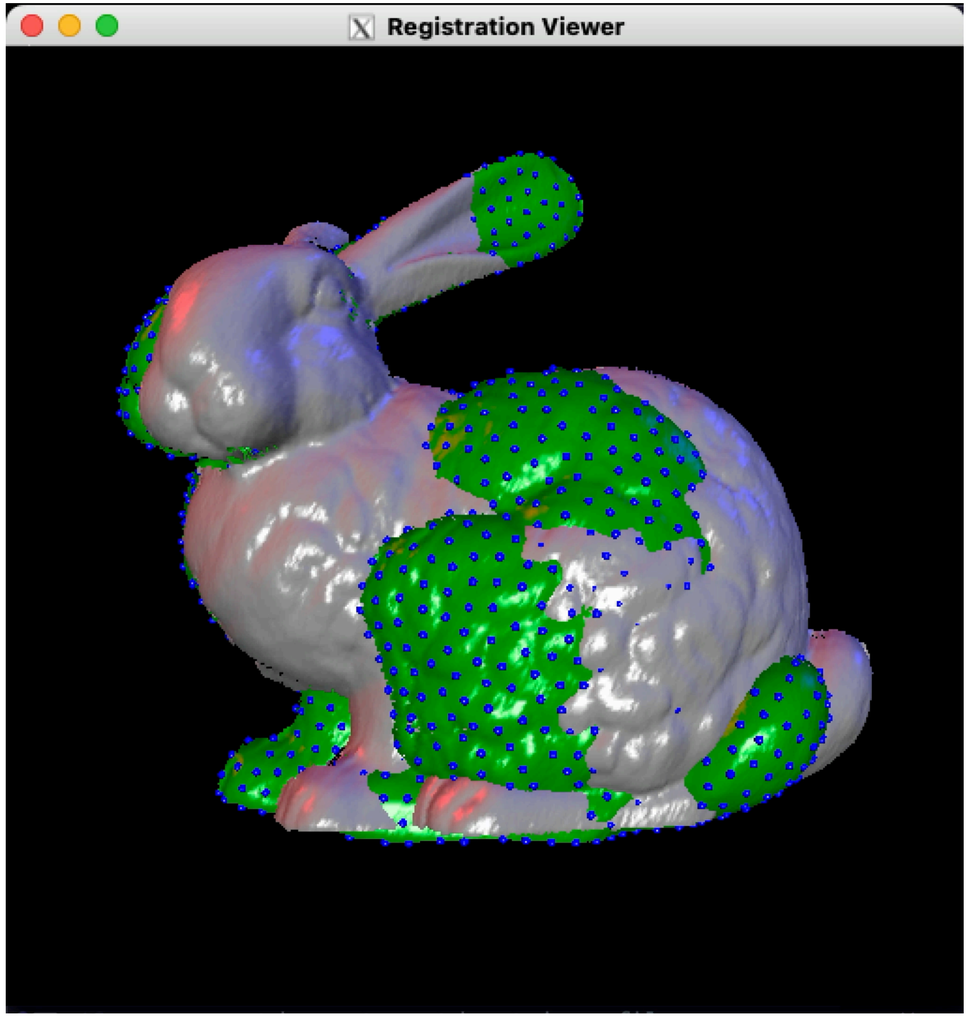
Surface Registration: Bunny Example



Perform rigid registration between 10 scans of the Stanford bunny

Surface Registration: GUI

- 'SHIFT' + mouse control: manual alignment for an initial transformation
- 'r': Perform a single registration step using point-to-point distance minimization
- 'SPACE': Perform a single registration step using point-to-plane distance minimization
- 'n': Move to next scan

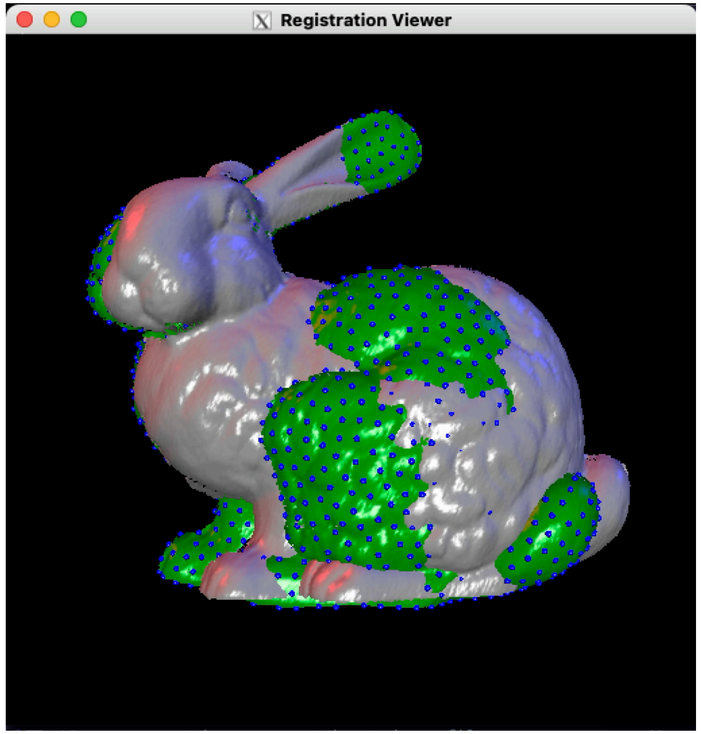


Task 1: Compiling the code

- Same dependencies as the second exercise
  - CMake, OpenGL, OpenMesh
- ANN (Approximate Nearest Neighbor)
    - Efficient closest point lookup using kd-tree

Task 2:

- Uniform subsampling within a given radius 'r'
- The sampled points are visualized in blue on the source mesh (the moving mesh)



Task 3: Bad Pairs Rejection

- Closest points are computed using ANN
- Prune correspondences based on
  - distance threshold
  - normal compatibility

Task 4: Point-2-point optimization

- Minimize

$$E = \sum_{i=1}^{N} \|Rp_i + t - q_i\|_2^2$$

- Decompose R using euler rotations:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & 0 & -\sin\gamma \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- $R = R_z(\gamma)R_y(\beta)R_z(\beta)$

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

Task 4: Point-2-point optimization

- Assuming the rotation angle is small:

$$c_\alpha = \cos\alpha \approx 1$$

$$s_\alpha = \sin\alpha \approx \alpha$$

$$s_x s_y \approx 0$$

- R can be linearized :

$$R = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}$$

Task 4: Point-2-point optimization

- Assuming the rotation angle is small:

$$c_\alpha = \cos\alpha \approx 1$$

$$s_\alpha = \sin\alpha \approx \alpha$$

$$s_x s_y \approx 0$$

- R can be linearized :

$$R = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}$$

- The task is to minimize ||Ax - b|| wrt x:

$$x = \begin{bmatrix} \alpha & \beta & \gamma & t_x & t_y & t_z \end{bmatrix}$$

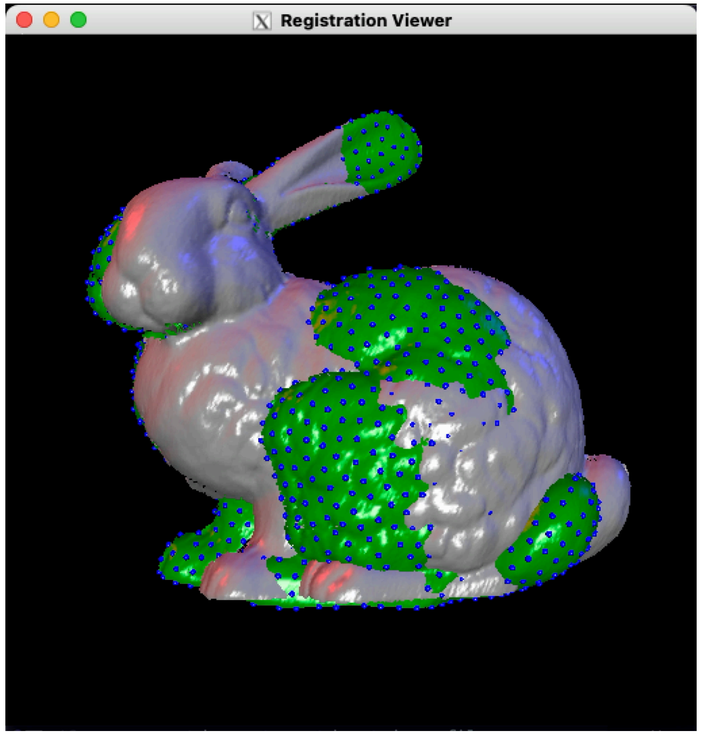
Task 5: Point-2-plane optimization

- Minimize

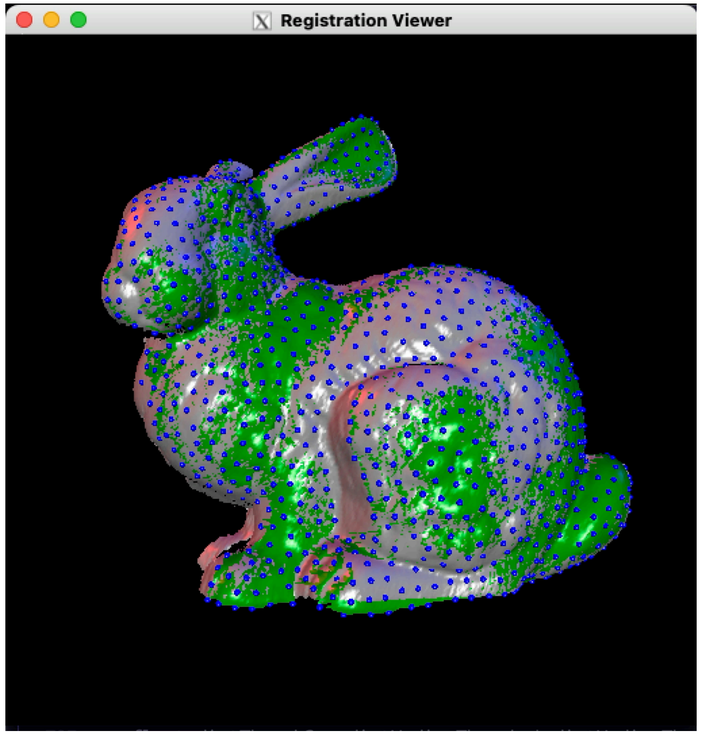
$$E = \sum_{i=1}^{N} \| n^T \left(R p_i + t - q_i \right) \|_2^2$$

- Same formula as point-2-point, but A and b are multipled by n^T

Results



point-2-point



point-2-plane

Submission

- Deadline: **Tuesday 9th April, 11:59 pm**
- Upload a .zip compressed file named “Exercise2-YourName.zip” to Moodle
- Include your code with comments
- Include a readme file:
  - Describe how you solved each exercise, using same exercise numbers (1.1 - 1.3) and titles as in handout
  - Describe problems you encountered
- Include JPEG frames or a video

Contact

INSTRUCTOR

- Hao Li, hao.li@mbzuai.ac.ae
- Office: Building 1B, 1st floor (please schedule first)

TEACHING ASSISTANTS

- Phong Tran, the.tran@mbzuai.ac.ae
- Long-Nhat Ho, long.ho@mbzuai.ac.ae

OFFICE HOURS

* Office Hours: TBD, will be posted soon
* Emails (include “CV804” in title)

QUESTIONS?

www.hao-li.com

THANKS!

MOHAMED BIN ZAYED
UNIVERSITY OF
ARTIFICIAL INTELLIGENCE

hao.li@mbzuai.ac.ae