Mesh Processing: From Creation to Comparison

Assoc. Prof. Yusuf Sahillioğlu

Computer Eng. Dept.  MIDDLE EAST TECHNICAL UNIVERSITY  Turkey
Surface Mesh

Representing geometry and topology as a polygon mesh.

```
1 v 1.000000 -1.000000 -1.000000
2 v 1.000000 -1.000000  1.000000
3 v -1.000000 -1.000000  1.000000
4 v -1.000000 -1.000000 -1.000000
5 v 1.000000  1.000000 -0.999999
6 v  0.999999  1.000000  1.000001
7 v -1.000000  1.000000  1.000000
8 v -1.000000  1.000000 -1.000000
9 f 2 3 4
10 f 8 7 6
11 f 5 6 2
12 f 6 7 3
13 f 3 7 8
14 f 1 4 8
15 f 1 2 4
16 f 5 8 6
17 f 1 5 2
18 f 2 6 3
19 f 4 3 8
20 f 5 1 8
```
Surface Mesh

Representing geometry and topology as a polygon mesh.
MP vs. CG

- Computer graphics // CENG 477
  - Rendering
    - Ray tracing
    - Rasterization

- Transformations
MP vs. CG

- Mesh Processing //CENG 789
  - Reconstruction
MP vs. CG

- Mesh Processing //CENG 789
  - Reconstruction

- Analysis
• Mesh Processing // CENG 789
  • Reconstruction

• Analysis

• Smoothing/Remeshing
MP vs. CG

- Mesh Processing // CENG 789
  - Parameterization
  - Deformation
  - Registration
  - Fabrication

- I’ll focus on my papers on Reconstruction and Analysis (Creation) (Comparison)
MP vs. CG

- Mesh Processing //CENG 789
  - Parameterization
  - Deformation
  - Registration
  - Fabrication

- I’ll focus on my papers on Reconstruction and Analysis (Creation) (Comparison)
**MP vs. IP**

- Image Processing //CENG 466
  - Regularity
    - IP: Every pixel has 4 neighbors
    - MP: Every edge is incident to 2 faces

- Exchange of ideas IP → MP

MP vs. IP

- Image Processing // CENG 466
  - Regularity
    - IP: Every pixel has 4 neighbors
    - MP: Every edge is incident to 2 faces

- Exchange of ideas MP $\rightarrow$ IP

Reconstruction

- Input: 3D point samples
- Output: Surface fit to the point samples

- 3D point samples can be acquired **passively** or **actively**
  - Stereoscopic images
  - Multiple silhouettes
  - Emitters
    - LIDAR, Laser Scanner, Kinect, ToF
Reconstruction

- Input: 3D point samples
- Output: Surface fit to the point samples

- 3D point samples can be acquired passively or actively
  - Stereoscopic images
  - Multiple silhouettes
  - Emitters
    - LIDAR, Laser Scanner, Kinect, ToF

- Alternative input: Scalar field defined over a 3D grid (CT)
  - Defines surface implicitly
  - Implicit methods, e.g., Marching Cubes, to extract the surface.
Reconstruction

• Input: 3D point samples; how to get them?
• Stereoscopy

Depth ambiguity handled by a second image
Reconstruction

- Input: 3D point samples; how to get them?
- Silhouettes

<table>
<thead>
<tr>
<th>Idea</th>
<th>Weakness</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Image]</td>
<td>[Image] Hidden Concavity</td>
</tr>
</tbody>
</table>
Reconstruction

• Input: 3D point samples; how to get them?
• Structured light

<table>
<thead>
<tr>
<th><strong>Idea</strong></th>
<th><strong>Weakness</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Diagram" /></td>
<td><img src="image" alt="Occlusion" /></td>
</tr>
</tbody>
</table>
Reconstruction

• Input: 3D point samples
• Output: Surface; how to get it?

Iterate

• Move each vertex \( P \) with \( v(P, B) \) in the direction of its normal \( N(P) \), as \( F_{\text{ext}} \) suggests: \( F_{\text{ext}}(P, B) = v(P, B) \cdot N(P) \)
• Regularize the mesh by \( F_{\text{int}} \)
• Collapse edges with length smaller than \( \varepsilon_{\text{min}} \)
• Split edges with length exceeding \( \varepsilon_{\text{max}} = 2\varepsilon_{\text{min}} \)
• Flip edges where necessary, favoring the vertices with valences close to 6

Till convergence

Reconstruction

- Input: 3D point samples
- Output: Surface; how to get it?

\[ \text{Fext: constant external force } (F(P) = -\varepsilon_{\text{min}}/2 \times N(P)) \]
Reconstruction

- **Input:** 3D point samples
- **Output:** Surface; how to get it?

**F_{ext}(P,B) = v(P,B) \cdot N(P)**

\[
v(P) = \varepsilon_{\min} \cdot f(P) = \varepsilon_{\min} \min_{r} \left\{ G \left( \text{Proj}_{r}(P) \right) - 0.5 \right\}
\]

\[
G(x', y') = (1 - \alpha)((1 - \beta)I(\lfloor x' \rfloor, \lfloor y' \rfloor) + \beta I(\lfloor x' \rfloor, \lfloor y' \rfloor + 1)) + \alpha((1 - \beta)I(\lfloor x' \rfloor + 1, \lfloor y' \rfloor) + \beta I(\lfloor x' \rfloor + 1, \lfloor y' \rfloor + 1))
\]

Reconstruction

- Input: 3D point samples
- Output: Surface; how to get it?

$F_{ext}$: force based on silhouettes
Reconstruction

• Input: 3D point samples
• Output: Surface; how to get it?

F_{ext}: force based on silhouettes

Reconstruction

- Input: 3D point samples
- Output: Surface; how to get it?
- Hidden concavity problem solved using range/laser surface

Reconstruction

- Input: 3D point samples
- Output: Surface; how to get it?
- Hidden concavity problem solved using range/laser surface.

Reconstruction

• Input: 3D point samples
• Output: Surface; how to get it?
• Mesh up laser point cloud as follows

  • Find local neighborhood $L_i$ of each point in the 3D point cloud input
  • For each $L_i$ compute tangent plane using PCA
  • Project all points in $L_i$ to the tangent plane and compute their 2D Delaunay triangulation
  • Merge all these local triangulations into a global one

• Input: 3D point samples
• Output: Surface; how to get it?
• Refine the silhouette-based mesh using an updated $F_{ext}(P)$ based on carvers assigned to triangles that share the vertex $P$
Comparison/Correspondence

• Once we have the meshes (reconstruction), we want to relate them with each other to enable nice apps, such as

• Shape interpolation:

• Deformation transfer:

• Attribute transfer:

• Shape registration:

• Shape matching:

• Statistical analysis:
Correspondence

• Solution idea
  • Quantify the quality of a given map
  • Then search the map space using this metric

\[ D_{iso}(\phi) = \frac{1}{|\phi|} \left( \sum_{(s_i, t_j) \in \phi} \frac{1}{|\phi'|} \sum_{(s_l, t_m) \in \phi'} |d_g(s_i, s_l) - d_g(t_j, t_m)| \right) \]

| .34 - .48 | = .64

A bad/high-distortion map.
Correspondence

• Looking at all N! permutations is infeasible
• Minimize this metric (or its variants) using:

  • Greedy optimization, Y. S., Y. Yemez, CVPR, 2010
  • Expectation-Maximization (EM), Y. S., Y. Yemez, PAMI, 2012
  • Deformation, Y. S., L. Kavan, Medical Image Analysis, 2015
  • Genetic optimization, Y. S., Transactions on Graphics 2018
  • Survey, Y. S., The Visual Computer, 2020
Correspondence

• Looking at all \( N! \) permutations is infeasible
• Minimize this metric (or its variants) using:

  • **Greedy optimization**, Y. S., Y. Yemez, CVPR, 2010
  • **Expectation-Maximization (EM)**, Y. S., Y. Yemez, PAMI, 2012
  • **Deformation**, Y. S., L. Kavan, Medical Image Analysis, 2015
  • **Genetic optimization**, Y. S., Transactions on Graphics, 2018
  • **Survey**, Y. S., The Visual Computer, 2020
Correspondence

- Looking at all $N!$ permutations is infeasible
- Minimize this metric (or its variants) using:
  
- Initial correspondence via MDS (left) is refined by greedy optimization based on neighbor voting (right).
Correspondence

• Looking at all $N!$ permutations is infeasible
• Minimize this metric (or its variants) using:

  • Isometric cost of matching $s_i$ to $t_j$ for all pairs (Q matrix in E-Step) guides graph matching and refinement which results in a better map to estimate Q (M-step). Repeat.

Minimum-Distortion Isometric Shape Correspondence Using EM Algorithm, Y. S., Y. Yemez, PAMI, 2012.
Correspondence

• Looking at all $N!$ permutations is infeasible
• Minimize this metric (or its variants) using:

• Represent a permutation as a chromosome and evolve many of them into the fittest one that yields the min-distortion map

Correspondence

- Looking at all N! permutations is infeasible
- Minimize this metric (or its variants) using:
  
  - Represent a permutation as a chromosome and evolve through genetic operators crossover and mutation

Correspondence

- Looking at all $N!$ permutations is infeasible
- Minimize this metric (or its variants) using:
  
  - Represent a permutation as a chromosome and evolve through genetic operators cross over and mutation

Correspondence

• Looking at all $N!$ permutations is infeasible
• Minimize this metric (or its variants) using:
  
  • Improved bilateral maps: scale-invariance, fuzzy voting.

Looking at all $N!$ permutations is infeasible
Minimize this metric (or its variants) using:

- Improved bilateral maps: scale-invariance, fuzzy voting.
Correspondence

- Looking at all $N!$ permutations is infeasible
- Minimize this metric (or its variants) using:
  - Improved bilateral maps: scale-invariance, fuzzy voting.

Looking at all N! permutations is infeasible
Minimize this metric (or its variants) using:

Improved bilateral maps: scale-invariance, fuzzy voting.
Interpolation

- Correspondence in action: shape interpolation
- Interpolate through the shortest path of inter-shapes

\[ v_{result}^k(\alpha) = \sum_{i=0}^{m} v_i^k b_{i,m}(\alpha), \quad b_{i,m}(\alpha) = \binom{m}{i} \alpha^i (1-\alpha)^{m-i} \]

Skeleton

- Correspondence in action: skeleton extraction/transfer
- Transfer the skeleton in source mesh to the target mesh using surface mesh correspondences

• Correspondence in action: inverse problem, skin extract
• Transfer the source mesh to the target skeleton using skeleton correspondences

Skeleton

Thanks

Papers, codes, executables, lectures, ..: http://ceng.metu.edu.tr/~ys

Y. S., Assoc. Prof.