Decomposing Polygon Meshes by Means of Critical Points

Yinan Zhou and Zhiyong Huang
School of Computing
National University of Singapore

Introduction

- Polygon mesh
  - the most widely available form of 3D models
  - a set of faces that share edges, or a set of vertices that are connected to neighbouring vertices by edges
  - no explicit higher-level structure

- Impose higher level structure
  - what features can be used to delineate components reliably?
  - how to detect these features?
  - how to decompose based on the features detected?

Related Work

- Computer vision-based decomposition
  - geometrical ions (geons), hierarchies of cylinder-like primitives, super-quadric
- Volume data decomposition
  - multi-resolution pyramids
- Polygon mesh decomposition
  - skeleton base methods

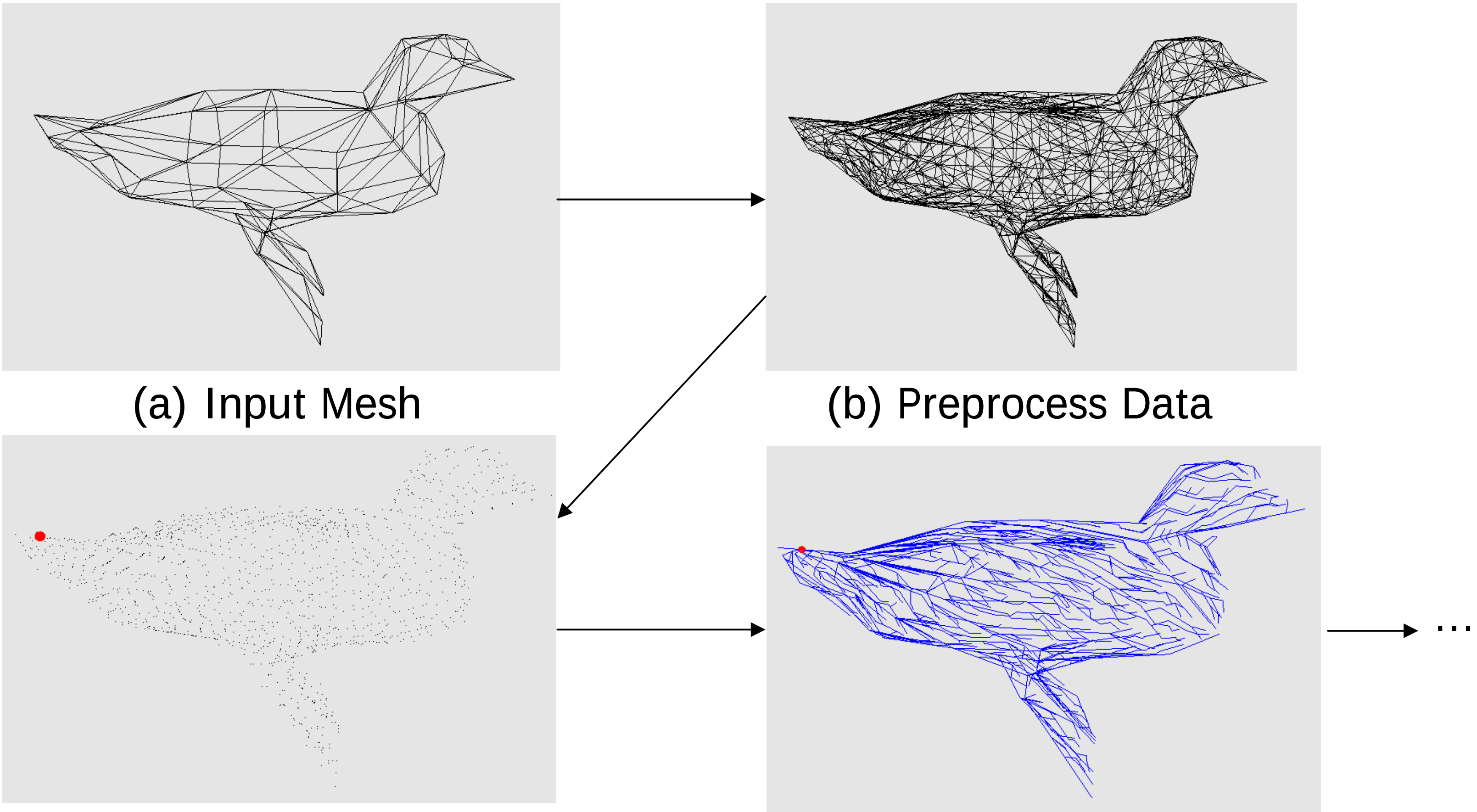
Our Approach

- A polygon mesh decomposition method
  - similar to Li et al. 01
    - by means of critical points
  - different from Li et al. 01
    - critical points are detected using the mesh surface information, geodesic distance, before the decomposition process.

Overview of the Method in Steps

1. Preprocess of input data
2. Select a root vertex
3. Capture geometric information of the object
4. Construct geodesic tree
5. Compute vertex sign change
6. Detect critical points
7. Order the critical points
8. Using backwards flooding to decompose the object

Illustration of Each Step



(a) Input Mesh

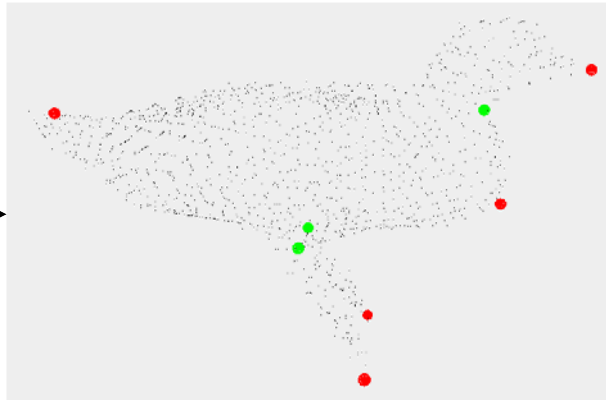
(b) Preprocess Data

(c) Define Root

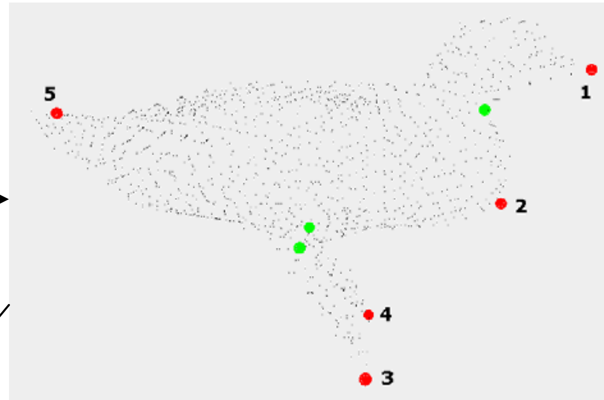
(d) Build Geodesic tree

Illustration of Each Step

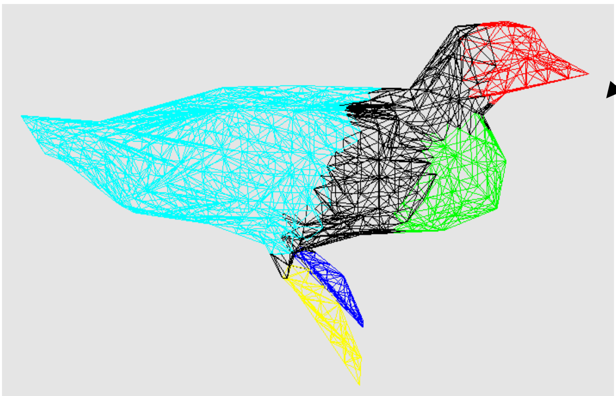
…



(e) Detect Critical Points

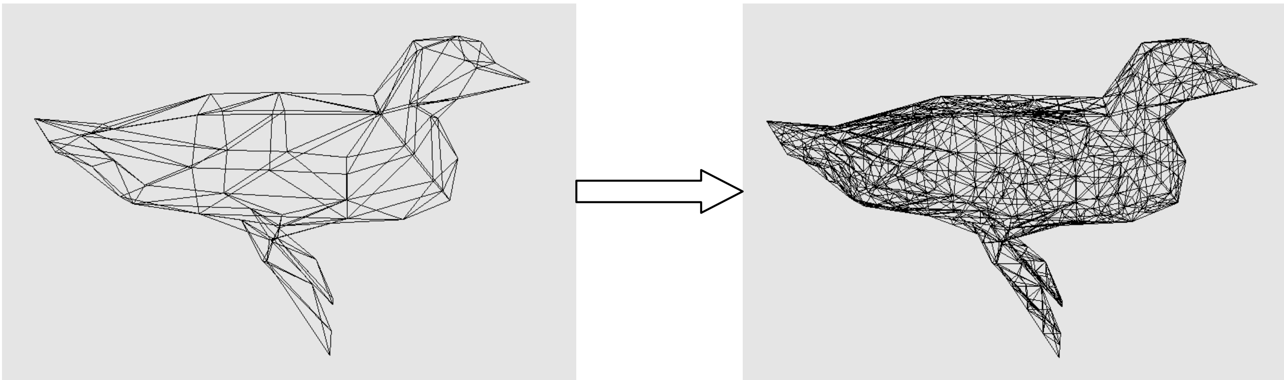


(f) Order Critical Points



(g) Decomposition by Backwards Flooding

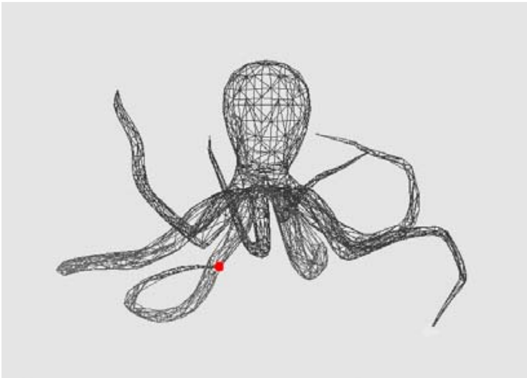
Step 1: Preprocess Input Data



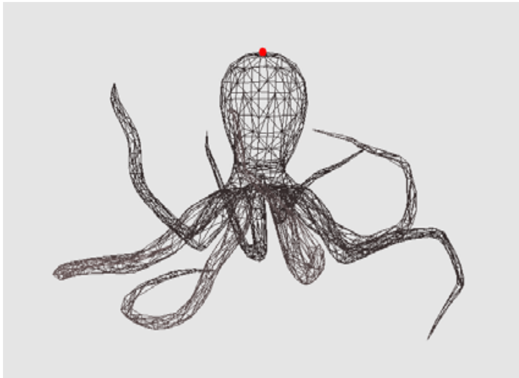
(a) 70 vertices, 136 triangles

(b) 916 vertices, 1828 triangles

Step 2: Define the Root Vertex

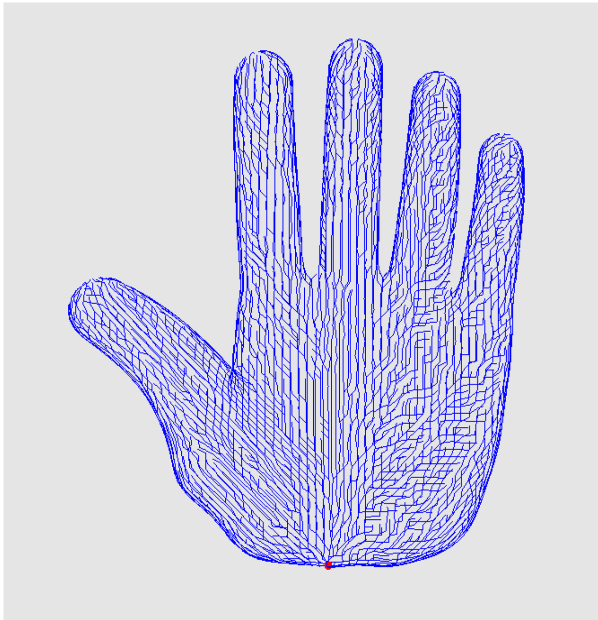


(a) Automatically selected root vertex



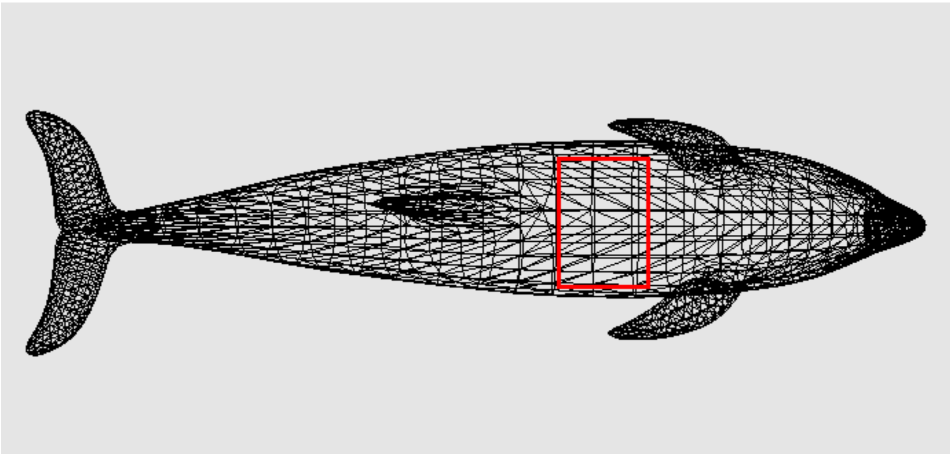
(b) Interactively selected root vertex

Step 3 & 4: Build the Geodesic Tree

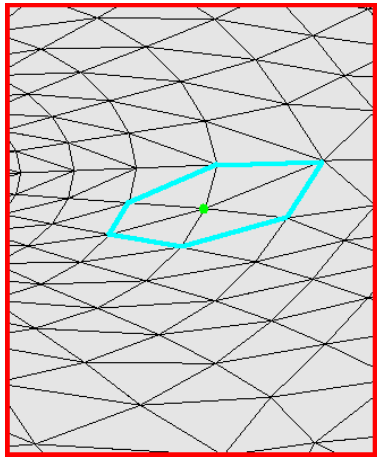


Example of a geodesic tree

Step 5: Detect Vertex Sign Change (sgc)



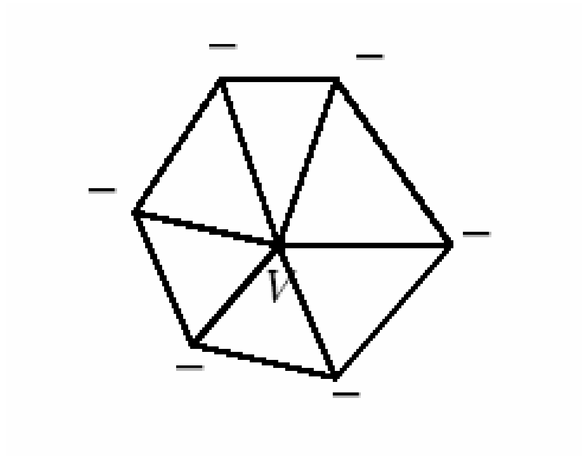
(a)A mesh with a small patch selected



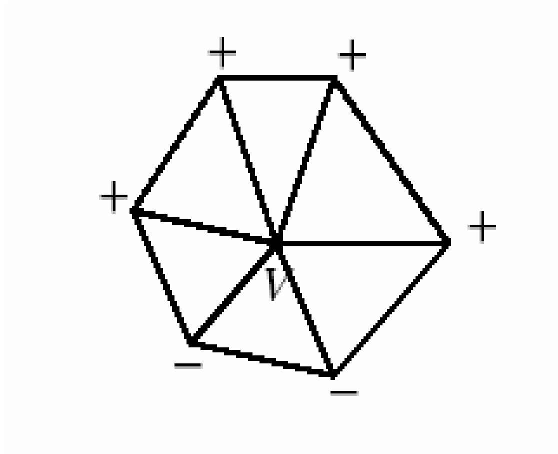
(b) Zoom-in of the small patch

Step 6: Detect critical points

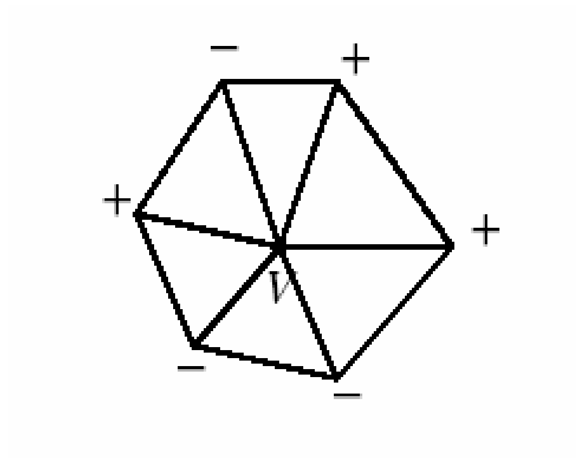
- Lazarus F. and Verroust A (1997)



(a) sgc(v) = 0, v is a local extremum

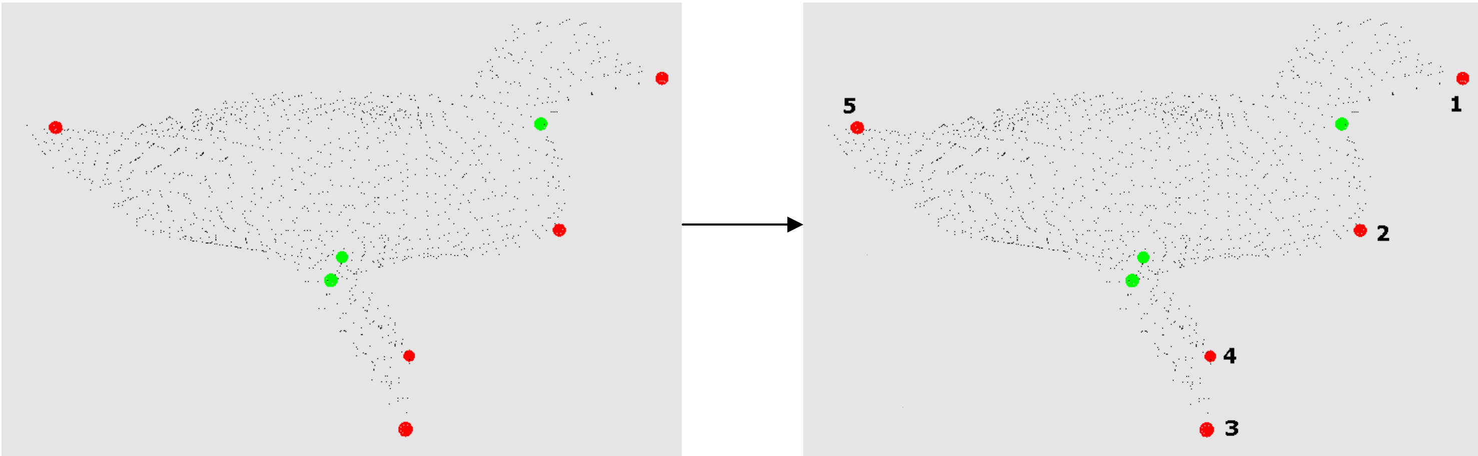


(b) sgc(v) =2, v is a regular vertex



(c) sgc(v) =2n with n=2, v is a saddle vertex

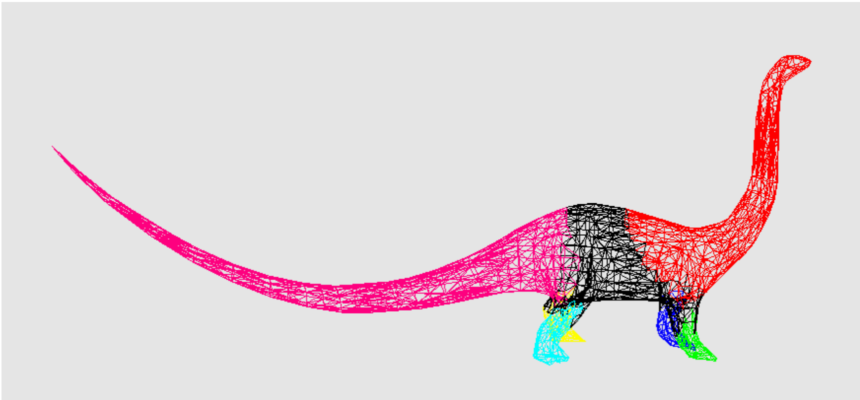
Step 7: Order the Critical Points

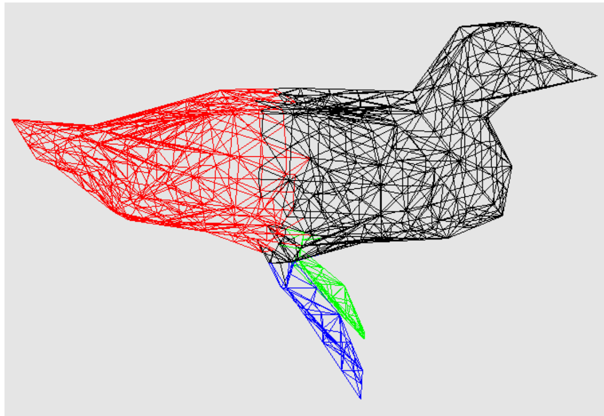


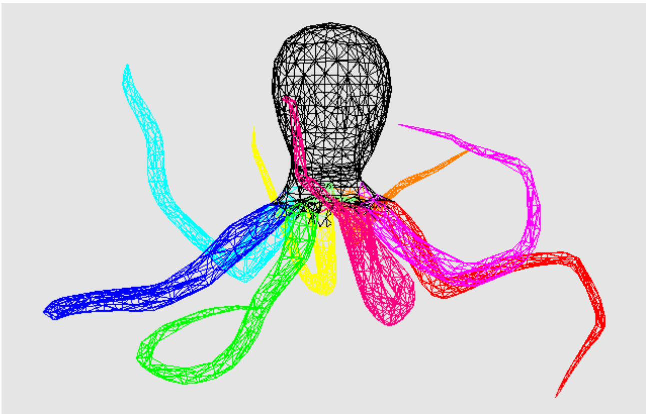
According to geodesic distance of a descending order

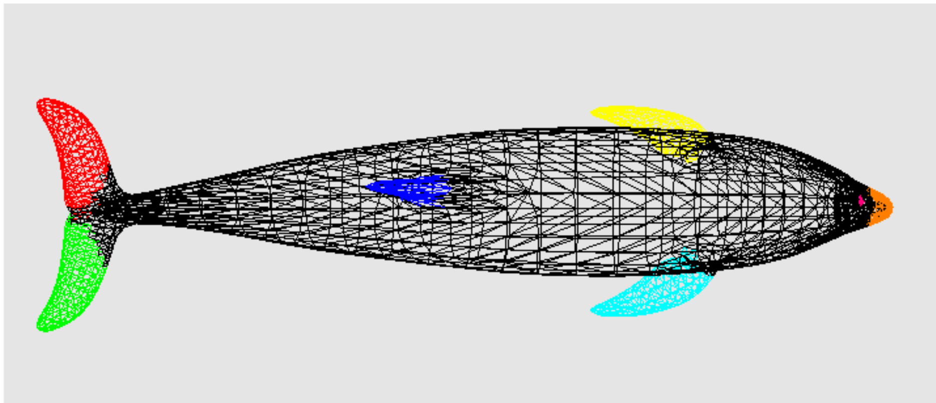
Step 8: Backwards Flooding to Decompose the Object

More Decomposition Results









Decomposition Time

	Dino-saur	Dolphin	Duck	Octopus
No. of vertices	1551	2101	916	1909
No. of faces	3098	4198	1828	3814
Decomposition time (second)				
Automatic allocation of root	2.00	2.00	1.00	3.00
Interactively assigned root	1.00	1.00	<1.00	2.00

Conclusion

- We propose a method for decomposing an object represented in polygon meshes into components by means of critical points
- We have implemented it and conducted experiments on a few 3D objects
- On the whole, it is effective and efficient

Future Work

- Applications
  - 3D model retrieval
  - Component-based morphing

Acknowledgement

- MMM 2004 anonymous reviewers
- NUS scholarship for undergraduate study
- NUS research grant R252-000-085-112