
MARCH 5-9
2007
SAN FRANCISCO
TAKE CONTROL
MOSCONE CENTER
www.gdconf.com
CMP

Precomputed Light Carving into Meshes

Stephan Etienne
High Moon Studios

Motivation

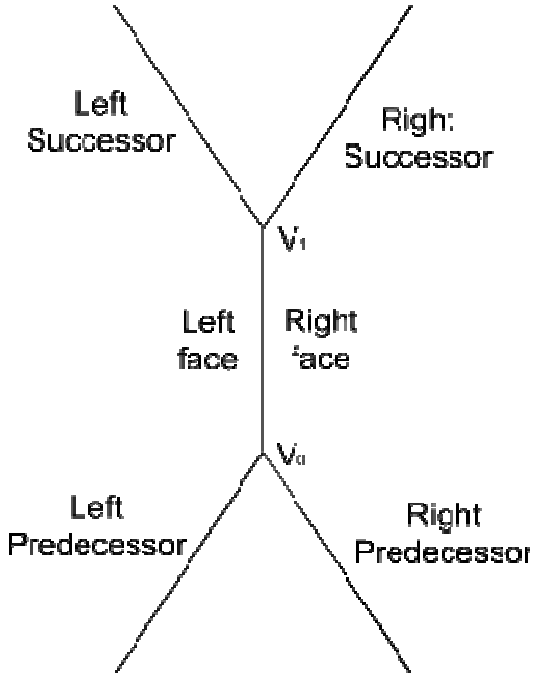
- Good looking pre-computed lighting without the need of big textures
- Cut down on rendering cost by removing a texture fetch

Highlights

- Shadow cutting
- Mesh tessellation
- Lighting
- Simplification
- Pros and cons

Use a Winged-Edge database

- Adding/removing vertices/edges/faces can be painful
- Use winged-edge data structure to minimize pain

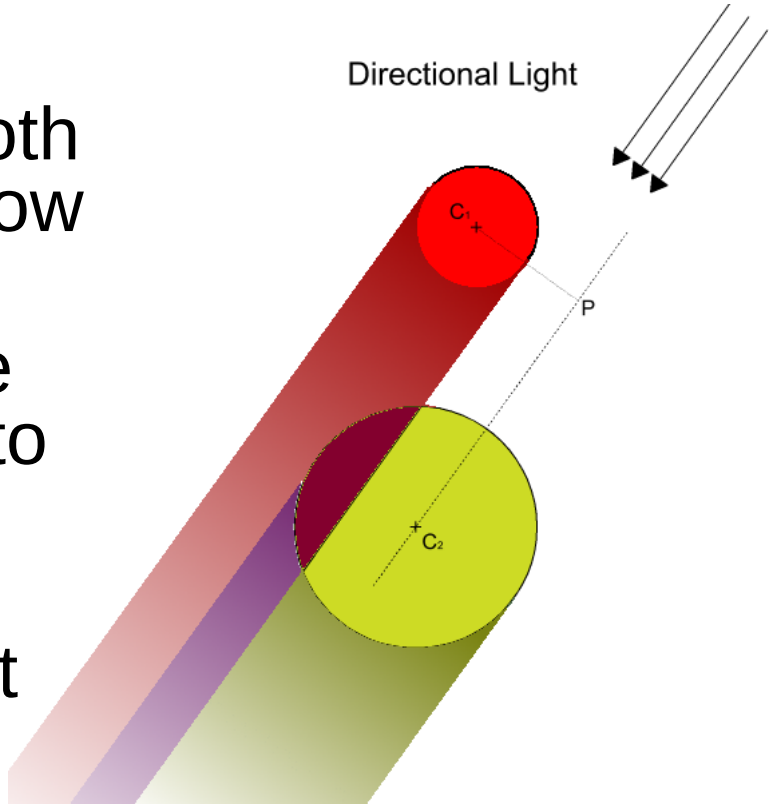


Shadow cutting

- Introduced by Alex Vlachos for the Animusic demo presented ay SIGGRAPH 2001
- Need to determine whether a mesh casts a shadow onto another
- Deal with directional lights and point lights separately

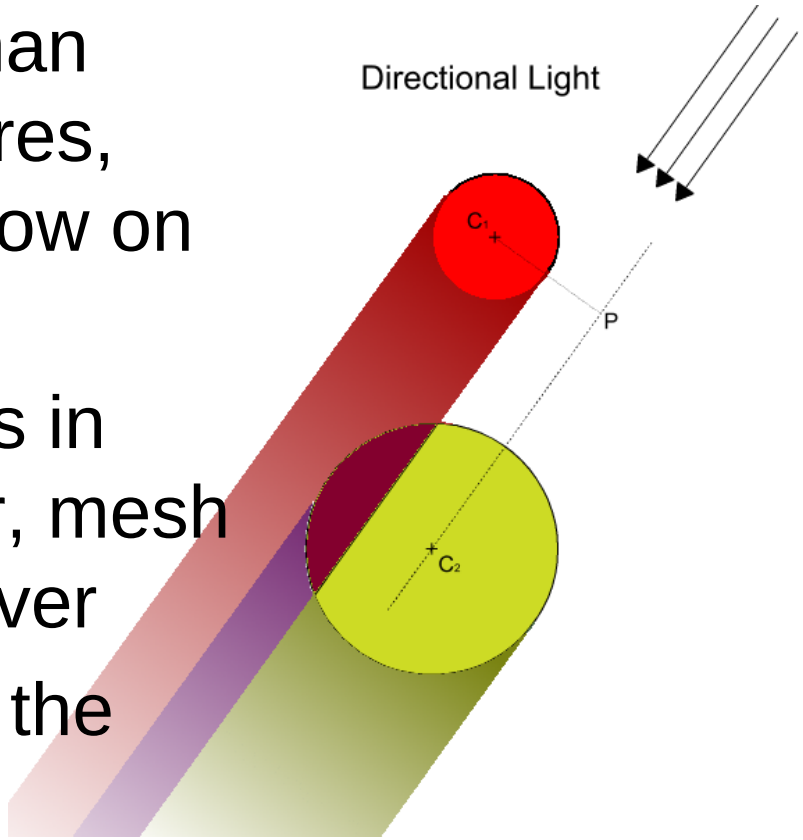
Shadow from directional lights

- If both bounding spheres intersect, both meshes cast a shadow on one another
- Project center of one bounding sphere onto other line
- Compute distance between center point and projected point



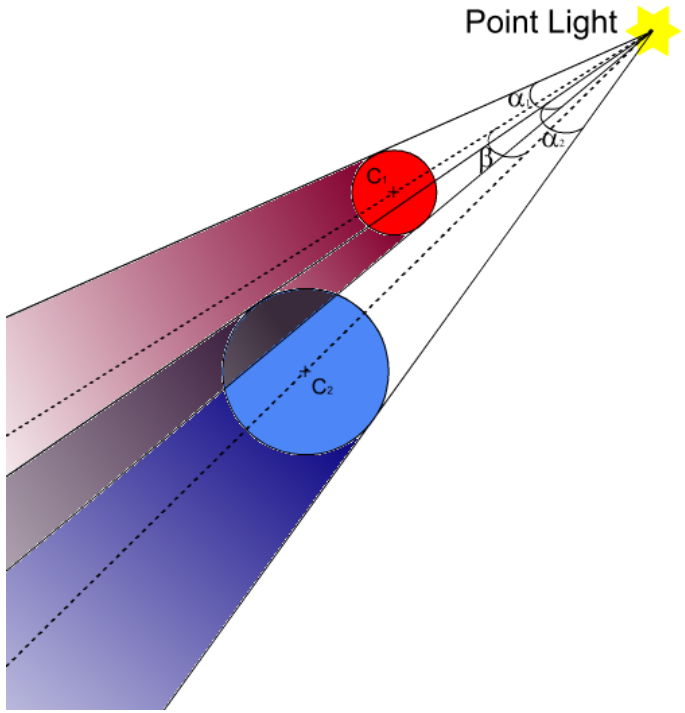
Shadow from directional lights

- If distance is less than radius of both spheres, one is casting shadow on the other
- If projected center is in front of other center, mesh is the shadow receiver
- Otherwise, mesh is the shadow caster



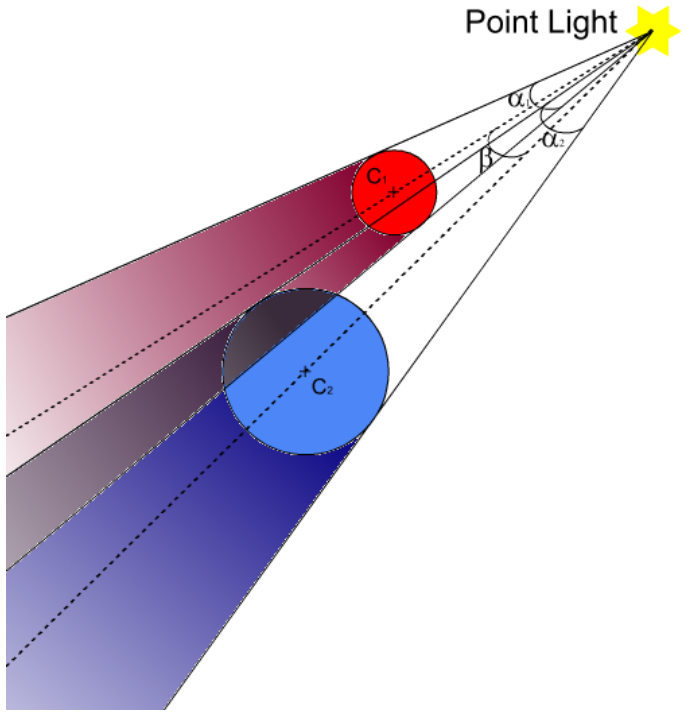
Shadows from Point Lights

- If both bounding spheres intersect, both meshes cast a shadow on one another
- Point light and bounding sphere form a conic section
- Compute α_1 and α_2, the angle of each conic section



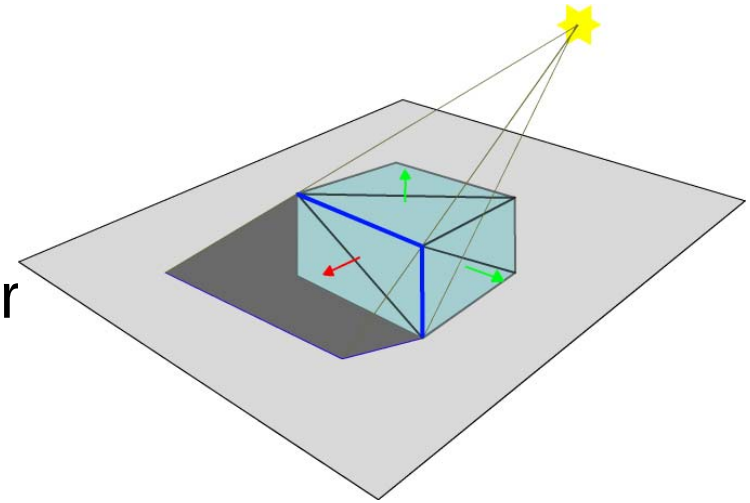
Shadows from Point Lights

- Compute β, the angle between the center line of each conic section
- If $(\alpha_1 + \alpha_2) * 0.5 >= \beta$, one is shadowing the other
- Mesh closest to the light shadows the other one



Cutting the shadow

- Project silhouette of shadow caster onto shadow receiver
- Silhouette edge has one face facing towards light and other facing away
- Construct list of silhouette edges



Cutting the shadow (continued)

- Project silhouette edge onto triangle plane
- If projected silhouette edge intersects triangle, split it along projected line
- Tip: Before projecting silhouette edge, translate edge along normal by small negative amount

Tessellation

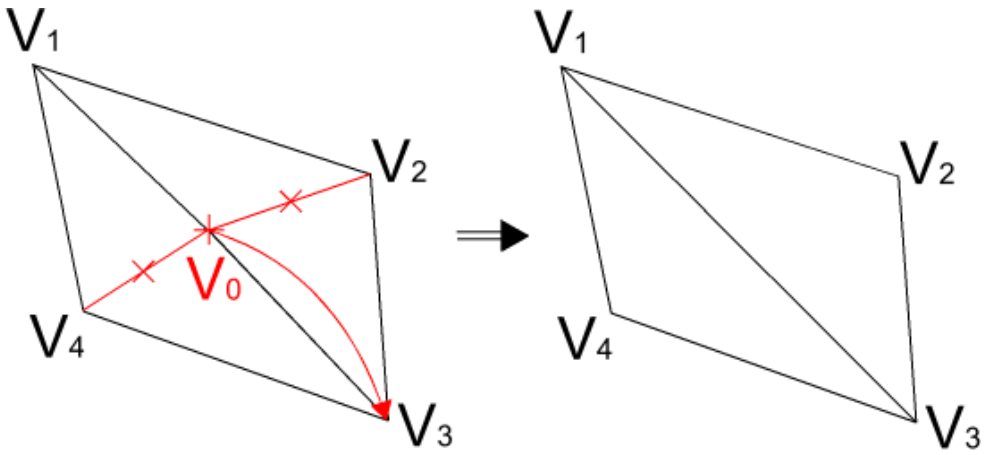
- Necessary to capture fine details in lighting changes
- Compute per triangle weight base on area size and edges length
- Subdivide until threshold reached
- Stop when max number of vertices have been created

Lighting

- Use your favorite technique to compute incoming lighting at each vertex
- A global illumination scheme is recommended

Simplification

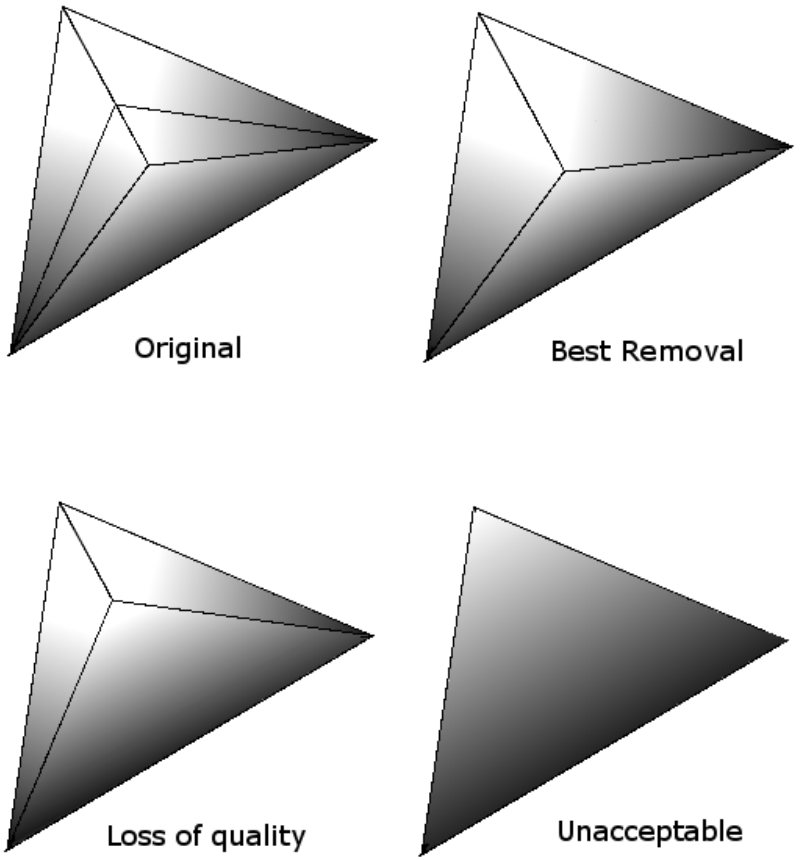
- A vertex is removed by collapsing it onto one of its neighbor



V_0 is collapsed at V_3. Edges to V_2 and V_4 and removed

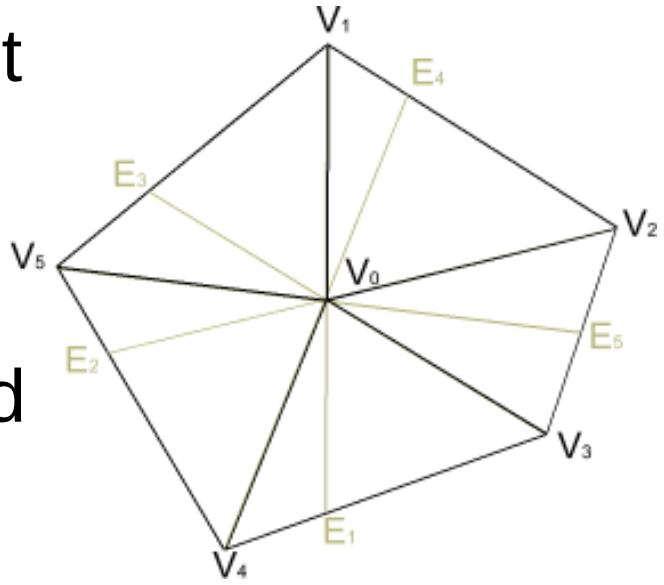
Simplification

- Remove least important vertices first
- It preserve quality
- It maximizes number of vertices that can be removed



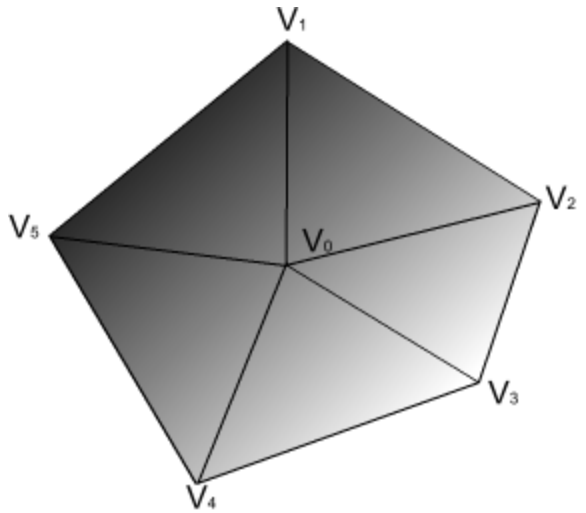
Simplification

- Compute color gradient
- Color gradient is max difference in color between interpolated vertex and extrapolated vertex
- V_1V_0 is projected onto V_4V_5 to create E_1



Simplification

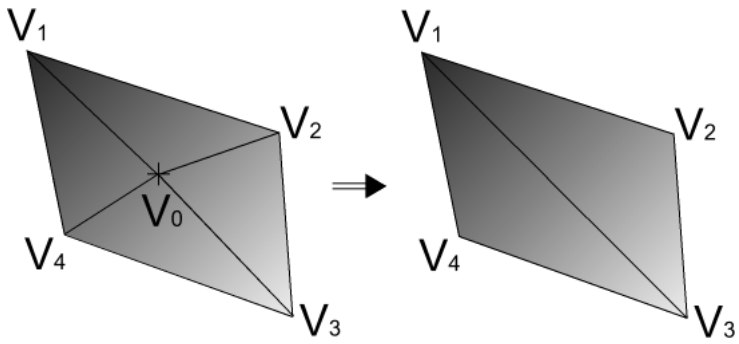
- Vertices at the center of a smooth triangle fan are removed



Smooth gradient triangle fan

Simplification

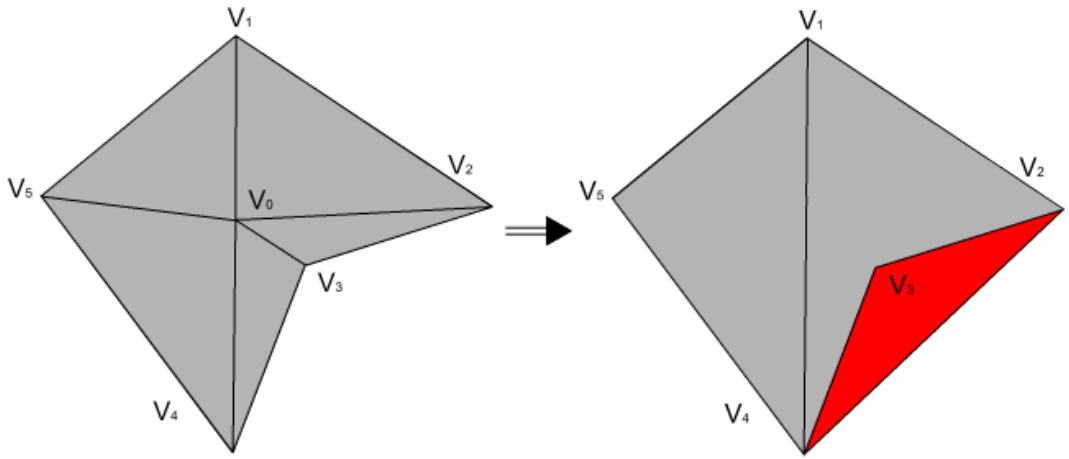
- If vertex is in middle of line segment
- Compute color gradient with interpolated vertex
- If color gradient is small, remove vertex



Removal of a vertex in the middle of a smooth edge

Simplification

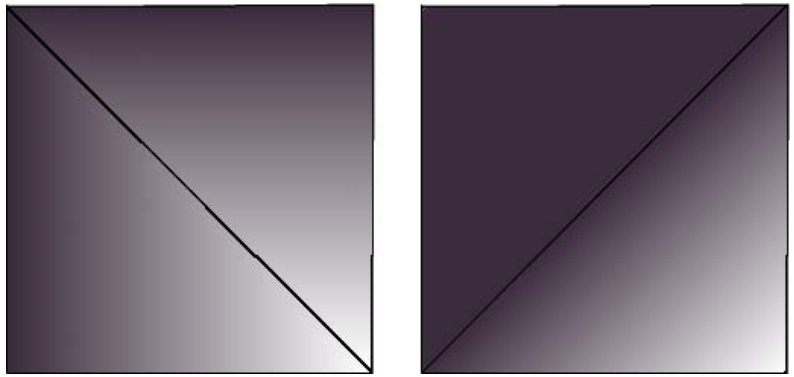
- Beware of creating degenerate triangles



Collapsing the edge of a concave triangle fan may result in degenerate shapes

Simplification

- There are two possible diagonals for a four sided polygon
- One orientation can look dramatically better than the other one



Benefits

- Pros:
  - Saves a lot of memory compared to light-maps
  - Improves fill-rate, in particular when up close
- Cons:
  - Increase vertex transform cost
  - Meshes viewed from afar can be more expensive to render

Paper

- Paper available at www.HighMoonStudios.com/Research/PreComputedLightCarving.pdf

Questions?