Visibility

Tom Funkhouser
COS 526, Fall 2016

Slides mostly by Frédo Durand
Visibility

Compute which part of scene can be seen
Visibility

Compute which part of scene can be seen (i.e., line segment from source to point in scene)
Visibility Applications

Computer graphics

- Hidden surface removal
  - Shadow computation
  - Global illumination
  - Occlusion culling
Visibility Applications

Computer graphics

- Hidden surface removal
- Shadow computation
- Global illumination
- Occlusion culling
Visibility Applications

Computer graphics
- Hidden surface removal
- Shadow computation
- Global illumination
- Occlusion culling

\[ L(x', x'') = L_e(x', x'') + \int s f_r(x, x', x'') L(x, x') V(x, x') G(x, x') dA \]
Visibility Applications

Computer graphics

- Hidden surface removal
- Shadow computation
- Global illumination
- Occlusion culling
Visibility Applications

Computational Geometry
- Art galleries

Computer vision
- Object recognition
- 3D scene reconstruction
- Next best view planning

Robotics
- Motion planning
- Visibility-based pursuit-evasion
- Self-localization
Outline

Hidden surface removal
- Visibility from viewpoint

Shadow map
- Visibility from point light source

Aspect graph
- Visibility from any point in space

Visibility Skeleton
- Visibility between scene elements
Outline

Hidden surface removal
- Visibility from viewpoint

Shadow map
- Visibility from point light source

Aspect graph
- Visibility from any point in space

Visibility Skeleton
- Visibility between scene elements
Hidden Surface Removal

Compute which part of every primitive can be seen from a point
Hidden Surface Removal

Occlusion by a single occluder
Hidden Surface Removal Problem

Cumulative occlusion by multiple occluders

Fredo Durand
Hidden Surface Removal Problem

Sorting according to a distance is not enough
Hidden Surface Removal Methods

Figure 29: Characterization of ten opaque-object algorithms. A comparison of the algorithms.
Hidden Surface Removal Methods

Image-space
- Z-buffer
- Scan-line
- Warnock subdivision

Object-space
- Depth-sort
- Weiler-Atherton
- BSP

Line-space
- Ray casting
Image-space

Computation performed in the plane of the image
E.g. is triangle inside rectangle?
Usually discretized in pixels
Object-space

3D space where the scene is defined
E.g., triangle is occluded if it is inside the pyramid

Fredo Durand
Line space

Visibility expressed in terms of rays
E.g. are all rays between the eye and the triangle blocked by the rectangle?
Typical advantages and drawbacks

Image-space
+ Robust, easier to code, occluder fusion, can use hardware
- Limited to one viewpoint, aliasing, needs hardware

Object-space
+ Precision, can handle from-region visibility
- Often robustness problems, occluder fusion is harder

Line space
+ Natural space, simple atomic operation (ray-casting)
- 4D, often requires approximation, or too complex

Fredo Durand
Outline

Hidden surface removal
  • Visibility from viewpoint

Shadow map
  • Visibility from point light source

Aspect graph
  • Visibility from any point in space

Visibility Skeleton
  • Visibility between scene elements
Shadow Maps

Precompute image of depths from light

- Store image of distances from light
- Lookup depth of surface point in image when shade
Shadow Maps

Suitable for hardware pipeline

- Projection into light coordinate system is 4x4 matrix
- Shadow map stored in texture

Problems

- Field of view
- Aliasing

[Teller & Durand, MIT]
Outline

Hidden surface removal
- Visibility from viewpoint

Shadow map
- Visibility from point light source

Aspect graph
- Visibility from any point in space

Visibility Skeleton
- Visibility between scene elements
Aspect Graph

There are many possible views of any 3D object
Aspect Graph

Some produce topologically equivalent visibility solution

Qualitatively equivalent (same aspect)

Qualitatively different (different aspect)
Aspect Graph

Characterization of the set of possible views of an object

- [Koenderink and Van Doorn 79, Plantinga and Dyer 90, Gigus et al. 90-91, Petitjean et al. 92]

Fredo Durand
Aspect Graph

For a polygonal scene with \( n \) edges

- \( O(n^3) \) visual events
- \( O(n^6) \) for orthographic views
- \( O(n^9) \) for perspective views

More reasonable estimate may be

- \( O(n^4) \) and \( O(n^6) \)

Not practical to compute and store!

Fredo Durand
Outline

Hidden surface removal
  • Visibility from viewpoint

Shadow map
  • Visibility from point light source

Aspect graph
  • Visibility from any point in space

Visibility Skeleton
  • Visibility between scene elements
Visibility from Polygon

Umbra and Penumbra

- [Nishita et Nakamae 85, Heckbert 92, Teller 92, Lischinski et al. 93, Drettakis et Fiume 94, Stewart et Ghali 94]
Visual event

Appearance-disappearance of objects (qualitative change of a view)
Visual event

Appearance-disappearance of objects (qualitative change of a view)
« Wedge » defined by a vertex and an edge
Type EV

Fredo Durand
Visual event

Appearance-disappearance of objects

Limits of umbra
Critical line

Line going through $e$ and $v$
Critical lines

1D set of lines going through $e$ and $v$ (1 degree of freedom)
Extremal stabbing line

1D set of lines going through $e$ and $v$ (1 degree of freedom)

Extremity: extremal stabbing line (VV) (0 degree of freedom)
Extremal stabbing line

Type $VEE$ (0 degree of freedom)
Adjacent critical line set

Generated by the second edge

Same extremity $ve_1e_2$
Triple-edge event
Visibility skeleton

Encodes adjacencies of extremal stabbing lines and critical line sets
Visibility Skeleton

Extremal stabbing line = Node
Visibility Skeleton

Extremal stabbing line = Node
Critical line set = Arc
Visibility Skeleton

Idea:
• Graph representation of visual events

Complexity
• Memory: $O(n^4)$ in theory, $n^2$ observed
• Time: $O(n^5)$ in theory, $n^{2.4}$ observed

Results
• Scenes up to 1500 polygons
  ▪ 1.2 million nodes
  ▪ 32 minutes for computation
Radiosity with Visibility Skeleton

Exact computation of form-factors
  • point-polygon

Discontinuity meshing
  • scene subdivision along shadow boundaries
  • also for indirect lighting

Refinement criterion
  • perceptual metric
  • error estimation
Radiosity with Visibility Skeleton

492 polygons: 10 minutes 23 seconds
Radiosity with Visibility Skeleton

Fredo Durand
Summary

Object-space visibility
- Help understand the nature of visibility
- Offer insights about which algorithms will work well
- Generally difficult to code and make robust

Image-space visibility
- Usually only for visibility from a point
- Can be implemented with graphics hardware
- Usual benefits/problems of image-precision computation