Efficient Image-Based Methods for Rendering Soft Shadows

Maneesh Agrawala
Ravi Ramamoorthi
Alan Heirich
Laurent Moll

Pixar Animation Studios
Stanford University
Compaq Computer Corporation
Compaq Computer Corporation

SIGGRAPH 2001
Hard vs. Soft Shadows

Hard Shadows

Soft Shadows
Shadow maps

- Image-based hard shadows [Williams 78]
- Time, memory depend on image size, not geometric scene complexity
- Disadvantage: bias and aliasing artifacts
- Soft shadows [Chen and Williams 93]
  - View interpolate multiple shadow maps
IBR good for soft shadows

- IBR good for secondary effects
  - Artifacts less perceptible
- IBR works well for nearby viewpoints
- Shadow maps from light source
  - Light source localized area
  - Poorly sampled regions are also dimly lit
IBR good for soft shadows

- Poorly sampled regions are also dimly lit
Contributions

- Extend shadow maps to soft shadows
- Image-based rendering especially suitable
- Two novel image-based algorithms:
  - Layered attenuation maps (LAM)
  - Coherence-based raytracing (CBRT)
• **LAM**
  - Display: 5-10 fps
  - Some aliasing artifacts
  - Interactive applications
    - Games
    - Previewing

• **CBRT**
  - Render: 19.83 min
  - Speedup: 12.96x
  - Production quality images
Preliminaries

\[ E = \int_{A_{light}} \left[ \frac{L \cos \theta_i \cos \theta_l}{\pi r^2} \right] V \, dA \]

\[ ATT = \frac{1}{A} \int_A V \, dA \]
Refresher: LDIs

- Layered depth images [Shade et al. 98]
• Layered depth images [Shade et al. 98]
Refresher: LDIs

- Layered depth images [Shade et al. 98]
Precomputation

- Render views from points on light (hardware)
- Create layered attenuation map (software)
  - Warp views into LDI
  - Store (depth, attenuation)
- Objects in LAM visible in at least 1 view
Precomputation

1st viewpoint
Precomputation

2nd viewpoint

Attenuation = 2/2

Attenuation = 1/2
Precomputation

Warped 2\textsuperscript{nd} viewpoint

Attenuation = \frac{2}{2}

Not present

Attenuation = \frac{1}{2}
Display

• Render scene without shadows (hardware)
• Project into LAM (software)
  • Read off attenuation
  • Attenuation modulates shadowless rendering
Display

LAM (center of light)

Eye
Display

LAM (center of light)

Eye

Attenuation = 2/2
Color = Color * 2/2
Display

LAM (center of light)

Eye
Display

LAM (center of light)

Eye

Not in LAM
Attenuation = 0
Color = Color * 0
procedure Precompute
1   foreach light sample \( l_i \)
2       \textit{Viewpoint} \leftarrow l_i
3       Render(SCENE)
4   foreach pixel \((x, y)\)
5       \((x', y')\) \leftarrow \text{WarpCenter}(x, y, z(x, y))
6       \text{Insert}((x', y'), z, \epsilon)
7       Process Attenuation Maps
Layered images
Layered attenuation map
procedure Display
1 RenderWithLightingAndTextures(SCENE)
2 foreach pixel \((x, y)\)
3 \((x', y', z') \leftarrow \text{WarpLDI}(\((x, y, z(x, y))\))\)
4 \text{layer} \leftarrow \text{Layer}((x', y'), z', \epsilon)\)
5 \text{color} \leftarrow \text{color} \ast \text{AttMap}((x', y'), \text{layer})
Attenuation map and rendering

1st layer

one layer

2nd layer

attenuation map

rendering
• LAM size: 512 x 512
• Avg num depth layers: 1.5
• Precomp:
  • 7.7 sec (64 views)
  • 29.4 sec (256 views)
• Display: 5-10 fps
• LAM size: 512 x 512
• Avg num depth layers: 2
• Precomp:
  • 6.0 sec (64 views)
  • 22.4 sec (256 views)
• Display: 5-10 fps
• Layered attenuation maps – fast, aliases
• Coherence-based raytracing – slow, noise
Coherence-based raytracing

- Hierarchical raytracing through depth images
  - Time, memory decoupled from geometric scene complexity

- Coherence-based sampling
  - Light source visibility changes slowly
  - Reduce number shadow rays traced

- Also usable with geometric raytracer
• Represent scene with multiple shadow maps
- Represent scene with multiple shadow maps
Image-based raytracing

- Trace shadow ray through shadow maps

1st shadow map

2nd shadow map

Light
Light source visibility image

Visibility image

Light

$S_1$
Light source visibility image

Vis image for $s_1$

Visibility image

Light

$0$

$s_1$

$s_2$
Coherence-based sampling

• Compute visibility image at first point $s_1$

• Loop over following surface points $s_i$
  • Predict visibility image at $s_i$ from $s_{i-1}$
  • Trace rays where prediction confidence low
Predicting visibility

Blocker pts

Prediction
Predicting visibility

Blocker pts

Prediction
**Prediction confidence**

- Low confidence
  - Light source edges
  - Blocked/unblocked edges

- Trace rays in all X’ed cells
  - High confidence: 5
  - Low confidence: 31
  - Total cells: 36
  - Ratio: $\frac{5}{36} = 0.14$

Predicted visibility
## Prediction confidence

- **Low confidence**
  - Light source edges
  - Blocked/unblocked edges

- **Trace rays in all X’ed cells**
  - High confidence: 56
  - Low confidence: 88
  - Total cells: 144
  - Ratio: \( \frac{56}{144} = 0.40 \)

Predicted visibility
Propagating low confidence

• If traced ray $\neq$ prediction
  trace neighbor cells

• Similar to [Hart et al. 99]
Propagating low confidence

- If traced ray $\neq$ prediction
  trace neighbor cells

Prediction incorrect
• Light cells: 16 x 16 (256)
• Four 1024 x 1024 maps

• Precomp:  2.33 min
• Render: 19.83 min
• Rays:  79.86

• Speedup:  12.96x
  2.27x due to image-based raytracing accelerations
  5.71x due to coherence-based sampling
- Light cells: 16 x 16 (256)
- Four 1024 x 1024 maps
- Precomp: 3.93 min
- Render: 65.13 min
- Rays: 88.74
- Speedup: 8.52x
  2.16x due to image-based raytracing accelerations
  3.94x due to coherence-based sampling
LAM
Ray tracing
CBRT
Conclusions

- Two efficient image-based methods
- Layered attenuation maps
  - Interactive applications
- Coherence-based raytracing
  - Production quality images
- IBR ideal for soft shadows – secondary effects