Dynamic Adaptive Shadow Maps on Graphics Hardware

Aaron Lefohn, UC Davis
Shubhabrata Sengupta, UC Davis
Joe Kniss, Univ. of Utah
Robert Strzodkka, CAESAR Institute
John Owens, UC Davis
Problem Statement

- **Goal**
  - Interactive high-quality GPU shadows
  - GPU-based adaptive shadow maps

- **Challenges**
  - GPU quadtree, trilinear filtering, dynamic
  - GPU adaptive refinement

- **Solution**
  - Page-table based dynamic adaptive data structure
  - Mix of GPGPU and graphics programming

Aaron Lefohn
University of California, Davis
Shadow Map History

- Williams, 1978
  - Depth image rendered from the light position
Shadow Map

- **Shadow lookup**
  - Transform camera pixels to shadow map
  - Compare pixel depth to value in shadow map
  - If shadow map value closer to light, pixel in shadow
Shadow Map History

• Pros
  • Simple, image-based approach
  • Fast: Hardware support
  • Support (fake) soft shadows
  • Often used in film renderers
Shadow Map History

- **Problems: Aliasing**
  - Incorrect shadow map sampling
    - Objects close to eye are far from light
    - Occluder parallel to light but perpendicular to eye
  - Depth quantization aliasing

*Image from NVIDIA GDC 2004 presentation, Gary King*
Previous Work

• Many attempts to fix shadow maps
  • Image-based
  • Hybrid image/object-based

• Problems
  • Special cases
  • Object-based calculations
Adaptive Shadow Maps (ASM)

- Fernando et al., SIGGRAPH 2001
- Solves sampling problems
  - Add shadow resolution where needed
    - Quadtree of small shadow maps
  - Dependent on light *and camera* position
- Robust solution but not possible in realtime
Adaptive Shadow Map
Our Solution

- ASM data structure
- ASM algorithm
- Results
ASM Data Structure Requirements

• Adaptive
• Multiresolution
• Fast, parallel random-access read
  • 2x2 native Percentage Closer Filtering (PCF)
  • Trilinear interpolated mipmapped PCF
• Fast, parallel write
• Fast, parallel insert and erase
Glift : Generic GPU Data Structures

- **GPU Data structure abstraction**
  - Upcoming TOG paper
  - “Glift : Generic, Efficient, Random-Access GPU Data Structures”

- **Factor data structure into**
  - Virtual domain
  - Physical domain
  - Address translator
  - Iterators

- **Glift template library**
ASM Virtual Domain

- Shadow map coordinates

(0,0) (1,0) (0,1) (1,1)

Aaron Lefohn
University of California, Davis
ASM Physical Domain

- Paged 2D texture memory
ASM Address Translator

- Mipmapped page table

Virtual Domain  ↓  Mipmapped page table  ↓  Physical Domain
ASM Data Structure

- Page table address translator
  - Common GPU structure, architecture literature
  - Coarse, uniform discretization of virtual domain
    - $O(N)$ memory
    - $O(1)$ computation
    - Uniform consistency
    - $O(1)$ insert
    - $O(1)$ erase
    - Partial mapping (sparse)
ASM Data Structure

- Page table example

\[
\begin{align*}
\text{vpn} &= \frac{\text{va}}{\text{pageSize}} \\
\text{off} &= \text{va} \mod \text{pageSize} \\
\text{pa} &= \text{ppa} + \text{off}
\end{align*}
\]
ASM Data Structure

- Adaptive Page Table
  - Map multiple virtual pages to single physical page

\[
vpn = \frac{va}{pageSize}
\]

\[
ppa = \text{pageTable}(vpn).ppa()
\]

\[
s = \text{pageTable}(vpn).s()
\]

\[
off = \frac{va \times s}{pageSize}
\]

\[
pa = ppa + off
\]
ASM Data Structure

- Multiresolution Page Table
ASM Refinement Algorithm

- Initialize ASM

- For each frame
  1. Identify eye pixels needing shadow refinement
  2. Transfer refinement request to CPU
  3. Draw new shadow pages into ASM

- Other than this...standard shadow mapping
1. Identify Pixels Needing Refinement

- **Refinement criteria**
  - Pixel is on shadow boundary
  - ASM does not contain required resolution
    - Required resolution
      - Projected area of screen space pixel into light space
    - Lookup resolution in current ASM

- **Result**
  - Shadow page location and resolution
  - Null
2. Transfer Refinement Request to CPU

- Compress request stream
  - Stream compaction: Daniel Horn, GPU Gems II, 2005

- Transfer small message to CPU
  - Usually 10s - 100s elements
3. Draw New Shadow Pages into ASM

- Allocate new pages
  - Draw into page tables

- Create new shadow pages
  - Draw into physical memory
ASM Shadow Lookup

- **Bilinear PCF lookup**
  - Duplicate 1 column and 1 row of texels in each page

```c
float4 main( uniform VMem2D asm,
              float3 shadowCoord,
              float4 litColor ) : COLOR
{
    float isInLight = asm.vTex2Ds( shadowCoord );
    return lerp( black, litColor, isInLight );
}
```

- **Trilinear mipmap lookup**
  - Two bilinear lookups
Demo
ASM Performance Results

- **Fernando Results**
  - 5 fps while moving camera
    - Many optimizations
      - Frustum culling, guaranteed frame rate, progressive refinement
  - Fixed light
  - 31K polys, $512^2$ image, $65K^2$ - $524K^2$ ASMs

- **Our results**
  - 15-20 fps while moving camera including refinement
  - 7-12 fps while moving light
  - 45k polys, $512^2$ image, $131K^2$ ASM

- Lookup time compared to $2048^2$ shadow map:
  - Bilinear filtered: 90%
  - Trilinear filtered mipmapped: 73%
ASM Memory Results

• Effective shadow resolution up to $131,072^2$

  $16^2 - 64^2$ page size
  $512^2 - 2048^2$ page table
  $2048^2 - 4096^2$ physical memory
  $20 - 80$ MB
Limitations

• **Performance**
  - Stream compaction (50% - 85% frame rate)
  - No frustum culling
  - No framerate guarantee / progressive refinement

• **Memory consumption**
  - 2-level page table

• **Correctness**
  - Refinement algorithm can miss edges
Summary

- **Dynamic, GPU-based adaptive shadow maps**
  - Possible at interactive rates on current GPUs
  - Trilinear filtering removes popping
  - Ready for use in relighting renderers
  - May be viable for games in the near future

- **Dynamic, adaptive GPU data structure**
  - Mix of GPGPU and graphics programming
Acknowledgements

- Craig Kolb, Nick Triantos  
  NVIDIA

- Fabio Pellacini  
  Dartmouth

- Adam Moerschell, Yong Kil  
  UC Davis
  Serban Porumbescu, Chris Co, ....

- Karen and Kaia Lefohn

- National Science Foundation Graduate Fellowship
- Department of Energy
- Pixar Animation Studios
More Information

- **Google “Adaptive shadow maps GPU”**

- **ACM Transactions on Graphics (TOG) paper**
  - “Glift: Generic, Efficient, Random-Access GPU Data Structures”
  - To appear...

- **GPGPU Siggraph 2005 course notes**