Agenda

- Maxwell Render overview
- Maxwell for the GPU
  - Why?
  - Why now?
- Implementation details
- Demo (if there is time)
Maxwell Overview

• First physically based render in the market (2004)
• Ground-truth reference render
• Predictive rendering tool
• Light analysis tool
Maxwell in use Animation & VFX
Maxwell in use Architecture
Maxwell in use Design
Maxwell in use Science
Maxwell in use Others
Maxwell in use: two more images

**Photo**
- Nikon D70.
- Focal length: 24mm.
- CCD sensor size: 23.7x15.6mm.
- Manual settings:
  - ISO 200
  - f/stop 8
  - Shutter speed: 1/60
- WB set with 6500K lamp in Cornell box.
- Cornell box dimensions: 50x50x50cm.
- Compact fluorescent lamp Philips Master PL electronic: 865 (6500K daylight white).
- 27W / 1700 lumen / efficacy: 62.96 lumen/W.

**Maxwell render: 1.0**
- Maxwell camera.
- Focal length: 24mm.
- Film size: 23.7x15.6mm.
- Manual settings:
  - ISO 200
  - f/stop 8
  - Shutter speed: 1/60
- Wall Color: 666 / Burn 1 / Gamma 2.2
- Cornell box dimensions: 50x50x50cm.
- Maxwell emitter: 6500K.
- 27W / 1700 lumen / efficacy: 62.96 lumen/W.
Maxwell in use: two more images
Maxwell Philosophy

Correct \rightarrow Fast 😊
Fast \rightarrow Correct 😊
Why Maxwell for the GPU is interesting?

- Uses very accurate models everywhere
- Light transport algorithms
- Materials
- Sky, Cameras
- ...

- If Maxwell runs on the GPU, almost any render can
- Very generic, used across many industries
Why Maxwell for the GPU now?

- Technical reasons
  - Maxwell is not easy to port
    - Not a regular PT + MIS
- Emotional reasons
  - Previous bad experiences
- 2013 nVidia Roadmap → Fermi, Kepler, Maxwell...
  - Clearly a signal!
GPU Maxwell Render

- Motivations & Goals
- Engine architecture
- Features
GPU Maxwell Render

- Motivations & Goals
- Engine architecture
- Features
GPU Maxwell Render: Motivations

- We are very happy with the current quality
- More speed is welcome
- Our average users have good GPUs
GPU Maxwell Render: Goals

• Keep Maxwell:
  • Accurate
  • Reliable
  • Same workflow
  • Support as many GPU types as possible
  • Do not target high-end GPUs only
GPU Maxwell Render

- Motivations & Goals
- Engine architecture
- Features
GPU Maxwell Render: Implementation

- Full GPU implementation
- CUDA seems the only choice
- Not an easy task...
  - Maxwell Render is extremely accurate
  - Spectral & unbiased renderer
  - Complex materials
  - Generic, not specialized
• Three main modules (there are many more...)

GPU Maxwell Render: Implementation
GPU Maxwell Render: Implementation

- Three main modules

- Ray Generation
- Ray Tracing
- Materials Evaluation
GPU Maxwell Render

• Three main modules
GPU Maxwell Render

• Ray Generation Module
  • Primary Rays (PR)
    • Rays shot from camera
    • High degree of coherence
    • Two neighboring rays will likely hit nearby objects
  • Secondary Rays (SR)
    • Rays shot from surfaces
    • No coherence
    • Two neighbouring rays might hit different objects
Two problems

- Some rays finish before others
- Differences in execution between PR and SR

Two solutions

- Ray regeneration
- Ray sorting
• Ray Generation Module
  • Rays terminate when they...
    • Hit an emitter
    • Hit the sky
    • Russian Roulette

• Threads that are responsible for those terminated rays remain idle ➔ uncool
• Ray Generation Module

• Ray Regeneration

• Terminated ray won’t wait other rays within the same thread block
• Regenerate it and create a new PR

Same time more samples
Higher GPU workload
GPU Maxwell Render

- Ray Generation Module
  - Ray Sorting
    - Thread blocks with just PR
      - High degree of coherence
      - Best performance situation
    - Thread blocks with PR and SR
      - SR will hurt PR performance
      - The worst SR will drive the performance of the whole thread block
• Ray Generation Module
  • Ray Sorting
    • GPU Ray sorting by Ray Type

- PR₀
- PR₁
- SR₀
- PR₂
- SR₁
- PR₃
- SR₂
- PR₄

BLOCK 0

BLOCK 1
GPU Maxwell Render

- Ray Generation Module
  - Ray Sorting
    - GPU Ray sorting by Ray Type

```
<table>
<thead>
<tr>
<th>PR0</th>
<th>PR1</th>
<th>SR0</th>
<th>PR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>SR1</td>
<td>PR3</td>
<td>SR2</td>
<td>PR4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PR0</th>
<th>PR1</th>
<th>PR2</th>
<th>PR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR4</td>
<td>SR0</td>
<td>SR1</td>
<td>SR2</td>
</tr>
</tbody>
</table>
```

BLOCK 0

BLOCK 1
Ray Sorting

• Simple, yet powerful
• More useful for high-res images
• Use indirection buffers for sorting
GPU Maxwell Render

- Three main modules
GPU Maxwell Render

- Ray Tracing Module
  - Spatial partition data structure of the scene for ray traversal
    - Custom BVH structure packed into textures + arrays
    - Ray execution more homogeneous
    - Less storage per node
GPU Maxwell Render

• Ray Tracing Kernels
  • GPU architecture dependent
  • Fermi, Kepler, Maxwell
  • Use each architecture strengths
GPU Maxwell Render

- Three main modules
• Materials Evaluation Module
  • Maxwell materials are complex
  • Loads of code
  • Many layers + Many BSDFs / layer → very generic
Materials Evaluation Module

- Big kernels are harmful \(^{(\text{surprise!})}\)
- Samples evaluating different materials
  - Access different data
  - Execute different code
Materials Evaluation Module

- **How do we deal with it?** Materials Group Queue System (MGQS)
  1. One Material Group ID assigned to each material
  2. Queue system for Material Groups (MG)
  3. Every queue has specific kernels (avoid big kernels)
  4. Samples are queued to the corresponding MG Queue
  5. Samples are queued to the corresponding MG Queue
  6. All samples evaluating the same MG are executed together
    - Increased coherence in execution time
    - Increased coherence in data access
GPU Maxwell Render

- Motivations & Goals
- Engine architecture
- Features
Sampling:
• Light sampling
• BSDF sampling

Variance reduction techniques
• Monte-Carlo sampling
• MIS
• Russian Roulette
• Geometry:
  • Clones, instances and pretessellated displacement, etc.
Materials:
- Diffuse, specular and glossy materials
- Support for complex materials
- Many layers per material
- Many BSDFs per layer
- Complex effects (dispersion, etc)
GPU Maxwell Render

- Emitters
  - Area emitters
  - Sun
  - Image-based lighting (IBL).

- Environment
  - Constant sky
  - Physically-based sky
  - Image-based lighting (IBL).
GPU Maxwell Render

- Camera
  - Thin lens, pin hole.
  - Depth of field (Dof).
  - Bokeh
  - Motion blur.
GPU Maxwell Render

- CPU vs GPU:
  - Obviously very scene dependent
  - 3-5x in many cases
  - Slower in corner cases
  - More research needed
  - Good balance between speed and money
GPU Maxwell Render

- V1 almost ready
  - Q3 2016

- Currently in development:
  - SSS
  - Procedural geometries (Hair, particles, etc)
  - Volumetrics
  - Specific GPU friendly denoiser
  - Multi-GPU support
Maxwell summary

Thanks 😊
MAXWELL MEETS the GPU

Juan Cañada
Maxwell Render Head of Technology