CS 380 - GPU and GPGPU Programming
Lecture 1: Introduction

Markus Hadwiger, KAUST
Goals

- Learn GPU architecture and programming; both for graphics and for compute (GPGPU)
- Shading languages (GLSL, Cg, HLSL), compute APIs (CUDA, OpenCL, DirectCompute)

Time and location

- Monday + Wednesday, 10:30 – 12:00, Building 9, Room 4140

Webpage:

http://faculty.kaust.edu.sa/sites/markushadwiger/Pages/CS380.aspx

Contact

- **Markus Hadwiger:** markus.hadwiger@kaust.edu.sa
- **Peter Rautek** (main contact assignments): peter.rautek@kaust.edu.sa

Prerequisites

- **C/C++ programming** (!), basic computer graphics, basic linear algebra
Lecture Structure

Lectures

• Part 1: GPU Basics and Architecture (both: graphics, compute)
• Part 2: GPUs for Graphics
• Part 3: GPUs for Compute

Some lectures will be on research papers (both seminal and current)

Assignments

• 5 programming assignments
• Weekly reading assignments (required; also some optional)

Quizzes

• 6 quizzes, 30 min each, ~every second Wednesday
  (tentative dates: Sep 11, Sep 25, Oct 9, Oct 30, Nov 13, Nov 27)
  • From lectures and (required) reading assignments

Semester project + final presentations, but no mid-term/final exam!

Grading: 40% programming assignments; 30% semester project; 30% quizzes
Resources (1)

Textbooks

- GPUs for Graphics: OpenGL 4.0 Shading Language Cookbook, 2nd ed.
Resources (1)

Textbooks

- GPUs for Graphics: OpenGL 4.0 Shading Language Cookbook, 2nd ed.
Long list of links on course webpage:

http://faculty.kaust.edu.sa/sites/markushadwiger/Pages/CS380.aspx

- www.opengl.org
- www.gpgpu.org
- www.nvidia.com/cuda/
- www.khronos.org/registry/cl/
- ...

Very nice resources for examples:

- GPU Gems books 1-3 (available online)
- GPU Computing Gems, Vol. 1 + 2 (Emerald/Jade edition)
- Ray Tracing Gems: High-Quality and Real-Time Rendering with DXR and Other APIs
Resources (3)

**OpenGL Programming Guide** *(red book)*

http://www.opengl-redbook.com/

Computer graphics and OpenGL

Current edition: 9\textsuperscript{th}
OpenGL 4.5
contains extended chapters on GLSL

Available in the KAUST library
also electronically
Resources (4)

OpenGL Shading Language (orange book)

Current edition: 3rd
OpenGL 3.1, GLSL 1.4
no geometry shaders

Available in the KAUST library
also electronically
Resources (5)

CUDA by Example: An Introduction to General-Purpose GPU Programming, Jason Sanders, Edward Kandrot

See reference section of KAUST library
Syllabus (1)

GPU Basics and Architecture (~August, September)

• Introduction
• GPU architecture
• How shader cores work
• GPU shading and GPU compute APIs
  – General concepts and overview
  – Learn syntax details on your own!
    – GLSL book
    – CUDA book
    – Online resources, ...
Syllabus (2)

GPUs for Graphics (~October)

- GPU texturing, filtering
- GPU (texture) memory management
- GPU frame buffers
- Virtual texturing
Syllabus (3)

GPU Computing (~November)

- GPGPU, important parallel programming concepts
- CUDA memory access
- Reduction, scan
- Linear algebra on GPUs
- Deep learning on GPUs
- Combining graphics and compute
  - Display the results of computations
  - Interactive systems (fluid flow, ...)

Semester project presentations
Example: Fluid Simulation and Rendering

- Compute advection of fluid
  - (Incompressible) Navier-Stokes solvers
  - Lattice Boltzmann Method (LBM)

- Discretized domain; stored in 2D/3D textures
  - Velocity, pressure
  - Dye, smoke density, vorticity, …

- Updates in multi-passes

- Render current frame

Courtesy Mark Harris
Example: Volumetric Special Effects

• NVIDIA Demos
  – Smoke, water
  – Collision detection with voxelized solid (Gargoyle)

• Ray-casting
  – Smoke: direct volume rendering
  – Water: level set / isosurface

Courtesy Keenan Crane
Example: Ray Tracing

Ray tracing in CUDA kernels, or ray tracing cores

- Microsoft DXR (DX12 API), Vulkan, NVIDIA OptiX / RTX
- NVIDIA Turing: “World’s First Ray Tracing GPU“ Quadro RTX, GeForce RTX

Epic Games Unreal Engine 4 with MS DXR
Example: Particle Simulation and Rendering

- NVIDIA Particle Demo
Example: Level-Set Computations

- Implicit surface represented by distance field
- The level-set PDE is solved to update the distance field
- Basic framework with a variety of applications
Example: Diffusion Filtering

De-noising

- Original
- Linear isotropic
- Non-linear isotropic
- Non-linear anisotropic
Example: Linear Algebra Operators

Vector and matrix representation and operators

• Early approach based on graphics primitives
• Now CUDA makes this much easier
• Linear systems solvers

Courtesy Krüger and Westermann
Example: Machine Learning / Deep Learning

Perfect fit for massively parallel computation

- NVIDIA Volta Architecture: Tensor Cores (mixed-prec. 4x4 matrix mult plus add)
- NVIDIA Turing Architecture: Improved Tensor Cores, ...

Frameworks

- TensorFlow,
  Caffe, Pytorch,
  Teano, ...

WHY ARE GPUs GOOD FOR DEEP LEARNING?

<table>
<thead>
<tr>
<th></th>
<th>Neural Networks</th>
<th>GPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherently Parallel</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Matrix Operations</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>FLOPS</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

GPUs deliver --
- same or better prediction accuracy
- faster results
- smaller footprint
- lower power
- lower cost

[Lee, Ranganath & Ng, 2007]
Example: GPU Data Structures

Glift: Generic, Efficient, Random-Access GPU Data Structures
• “STL” for GPUs
• Virtual memory management

Courtesy Lefohn et al.
Reading Assignment #1 (until Sep 2)

Read (required):

• Orange book, chapter 1 (*Review of OpenGL Basics*)
• Orange book, chapter 2 (*Basics*)
Thank you.