Real Time Image Segmentation

Miklos Homolya, Ravikishore Kommajosyula, Gaurav Kukreja

Technical University of Munich

April 3, 2014

Overview

Problem Definition

OpenGL Interoperability

What is Interoperability?

- Mapping OpenGL Resources to CUDA, to enable CUDA to read/write
- Can be used to show output from CUDA kernel, straight from GPU saving time and bandwidth

How to use OpenGL Interop?

- Set current threads OpenGL context to use for OpenGL interop with CUDA **device**.

```
cudaGLSetGLDevice(device);
```

- Create OpenGL Pixel Buffer, and register to use as CUDA buffer.

```
gl.glGenBuffers(1, &pixels);
gl.glBindBuffer(GL_PIXEL_UNPACK_BUFFER, pixels);
size_t size = w * h * 4 * sizeof(unsigned char);
gl.glBufferData(GL_PIXEL_UNPACK_BUFFER, size, 0,
    GL_DYNAMIC_DRAW);
cudaGraphicsGLRegisterBuffer(&pixels_CUDA, pixels,
    cudaGraphicsMapFlagsWriteDiscard);
```

How to use OpenGL Interop?

- Before starting kernel, map pixel buffer to a CUDA pointer.

```
cudaGraphicsMapResources(1, &pixels_CUDA, 0);
cudaGraphicsResourceGetMappedPointer(&d_pixels, &size,
     pixels_CUDA);
```

References

John Smith (2012)
Title of the publication
Journal Name 12(3), 45 – 678.

The End