

Ci1009 Lab 2 - ERE2 (2020) - versão 1.0

CUDA Image Color to Grayscale

(baseado no GPU Teaching Kit -- Accelerated Computing)

Objetivo

O objetivo deste laboratório é converter uma imagem RGB em imagem de escala de cinza. A entrada é uma imagem PPM em RGB e o aluno irá converter cada pixel dessa imagem em um único valor de intensidade de escala de cinza. Um pseudo-código da versão do algoritmo é mostrado abaixo:

```
for ii from 0 to height do
  for jj from 0 to width do
    idx = ii * width + jj
    # here channels is 3
    r = input[3*idx]
    g = input[3*idx + 1]
    b = input[3*idx + 2]
    grayImage[idx] = (0.21*r + 0.71*g + 0.07*b)
  end
end
```

Pré-requisitos

Antes de iniciar este laboratório, certifique-se de que:

- Você concluiu o lab1

Formato das imagens

A imagem de entrada está armazenada no formato PPM P6 enquanto a imagem em tons de cinza de saída é armazenada em Formato PPM P5. Os alunos podem criar suas próprias imagens de entrada exportando sua imagem em imagens PPM caso queiram. A maneira mais fácil de criar a imagem é por meio de ferramentas externas.

No Unix, existem programas para converter imagens para PPM.

A wlib já tem uma função para ler a imagem PPM RGB e para gerar PPM 5 (grayscale).

Instruções

Edite o código para realizar o seguinte:

- alocar memória do dispositivo
- copiar a memória do host para o dispositivo
- inicializar o bloco de threads e as dimensões da grade do kernel
- invocar kernel CUDA
- copiar os resultados do dispositivo para o host
- desalocar memória do dispositivo

As instruções sobre onde colocar cada parte do código estão demarcadas pelas // @@ linhas de comentário.

Instruções de configuração local

A versão mais recente do (programa) modelo/esqueleto para este laboratório (código-fonte e instruções) está em `~wagner/ci1009-ere2-2020/labs/lab2` na máquina orval ou macalan.

Você deve fazer uma cópia de cada diretório para sua conta pessoal e trabalhe com sua cópia.
O professor lhe dará instruções sobre onde/como entregar sua solução final, provavelmente por e-mail como o lab1.

O executável gerado como resultado da compilação de sua solução de laboratório deve ser executado usando o seguinte comando:

```
./ImageColorToGrayscale -i <input.ppm> -o <output.pbm>
```

```
./ImageColorToGrayscale -i <input.ppm> -o <output.pbm>
```

onde:

<input.ppm> é a imagem (do dataset), e <output.pbm> é a imagem de saída (também já está no dataset)

A biblioteca wb tem uma função para comparar a imagem gerada por seu kernel com a imagem de saída no dataset.

Code Template

O modelo de código no diretório do laboratório é sugerido como ponto de partida para estudantes. O código lida com a importação e exportação, bem como o verificação da solução.

Os alunos devem inserir seu código é as seções demarcadas com `// @@`.

Espera-se que os alunos não alterem a biblioteca, ou seja, apenas adicionem o seu código para fazer a transformação de imagem e para ativar as funções de CUDA necessárias.

Neste ponto, o wblib original não é funcional, então o professor está fornecendo uma versão "work-around" da lib.

This work is licensed by UIUC and NVIDIA (2016) under a Creative Commons Attribution-NonCommercial 4.0 License.

Modificada por W.Zola/UFPR