



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento Ciencia de la Computación
IIC3373 - Programación Concurrente

Tarea 3: Introducción a OpenCL

Entrega: Lunes 14 de Octubre

Introducción:

El objetivo de esta entrega es el de introducir al alumno al mundo de la programación concurrente asistida por GPU. Para ello, se utilizará el estándar más usado en la industria que permite aprovechar el poder de cómputo de las tarjetas gráficas en computadores de escritorio: OpenCL.

Descripción:

Usted deberá implementar un programa que reciba de entrada la ruta a un archivo de video ubicado en el computador y produzca un nuevo video con el contraste de color corregido mediante algoritmos de ecualización. Esta es una técnica muy común utilizada para mejorar la calidad en imágenes (o en este caso videos) de mala calidad cuyos colores sean demasiado claros o demasiado oscuros.

Existen varios algoritmos de ecualización disponible en la literatura. Debe elegir uno de ellos, con excepción de la ecualización RGB que produce resultados bastante malos, y aplicar el método cuadro por cuadro en su video.

Es requisito fundamental que este proceso aproveche toda la capacidad de su tarjeta gráfica para acelerarlo lo más posible. Para ello usted deberá hacer uso de OpenCL. Se deja a libertad del alumno la determinación de los elementos a paralelizar. Por ejemplo, usted puede despachar un worker thread por cada cuadro; pero tal vez otro alumno decida despachar un worker thread por cada línea de cada cuadro; etc.

Es imperativo el cumplimiento de este requisito. Los programas que no usen OpenCL obtendrán automáticamente un 1.

Restricciones:

Usted puede desarrollar su tarea en Java, C# o C++. Para trabajar con OpenCL puede hacer uso de cualquier “wrapper” disponible en la web.

Para el procesamiento del video, se autoriza a los alumnos a hacer uso de cualquier librería third-party que deseen.

Evaluación:

La evaluación constará de tres partes:

- Evaluación del funcionamiento del programa (5 puntos)
- Ingeniería de software (1 punto)
- Informe (1 punto)

Como usted notará, si a esto le agrega el punto base la nota máxima es un 8.

Para el primer punto, usted deberá seleccionar una de las siguientes alternativas. Si su programa cumple a cabalidad todas las funcionalidades descritas en dicha alternativa, se le asignará el puntaje asociado. En caso de que falle alguna, automáticamente obtendrá 0 puntos. Sea prudente en seleccionar una alternativa apropiada: no elija una que otorgue más puntos si su programa no la logra cumplir!

1. El programa recibe una imagen a color y la devuelve ecualizada (1 punto).
2. El programa recibe un conjunto de cuadros individuales a color y logra ecualizarlos todos (2 puntos).
3. El programa recibe un conjunto de los cuadros de un video y devuelve el video ecualizado, con o sin audio, a al menos 25 fps y con una duración de entre 5 a 30 segundos (3 puntos).
4. El programa recibe un video de entre 5 a 30 segundos y devuelve sus cuadros, al menos 25 por segundo, ecualizados (3 puntos).
5. El programa recibe un video a color de entre 5 a 30 segundos y lo devuelve ecualizado a al menos 25 fps (3.5 puntos).
6. El programa recibe un video a color de más de 30 segundos y lo devuelve ecualizado a al menos 25 fps, pero sin audio (4 puntos).
7. El programa recibe un video a color de más de 30 segundos y lo devuelve ecualizado a al menos 25 fps y con el mismo audio original (5 puntos).

El punto de ingeniería de software tiene que ver con la claridad del código, el seguimiento de buenas prácticas de programación y usabilidad.

Para el último punto, usted deberá entregar un informe de pequeña extensión donde explique el algoritmo de ecualización utilizado y justifique su elección. Recuerde incluir referencias bibliográficas.

Plazos y entrega:

El plazo para entregar esta tarea es el día lunes 14 de octubre a las 23:59. Se descontará un punto de la nota final por cada hora o fracción de atraso.

Los archivos que usted debe entregar son:

- El código escrito.
- El informe solicitado.
- Una imagen, conjunto de imágenes o video para probar su programa (esto depende de la alternativa que haya seleccionado). Se sugieren imágenes o videos de mala calidad para mejor apreciar los efectos de su programa.
- Un archivo de texto .readme en donde explique cómo utilizar su programa y donde precise la alternativa de evaluación seleccionada.

Pueden enviar estos entregables comprimidos en un .zip o un .rar via mail a jibenedettoc@gmail.com.

En caso de problemas de envío por cualquier motivo, se aceptarán vías secundarias de entrega como por ejemplo compartir un Dropbox o subir su archivo a un servicio de File Sharing en línea (rapidshare, etc.) siempre y cuando se realice antes del plazo límite. Recuerden en este caso enviar un mail al ayudante con copia al profesor detallando los problemas que tuvieron al realizar el envío.

Referencias:

[1] <http://www.khronos.org/opencv/>

[2] http://www.cmsoft.com.br/index.php?option=com_content&view=category&layout=blog&id=41&Itemid=75

[3] http://en.wikipedia.org/wiki/Histogram_equalization

[4] Cheng, Fan-Chieh, Shanq-Jang Ruan, and Chang-Hong Lin. "Color Contrast Enhancement Using Automatic Weighting Mean-Separated Histogram Equalization with Spherical Color Model." International Journal of Innovative Computing, Information and Control 7.8 (2011).

[5] <http://www.ffmpeg.org/>