Задание №2 Использование многопоточных вычислений при программировании видеопроцессоров

Написать программу на языке С/ или Python с использованием API CUDA или OpenCL. Произвести тестирование программы для разных размеров обрабатываемых файлов изображений. Для тестирования взять файлы размерами: 1024 x 768, 1280 x 960, 2048 x 1536. Получить среднее значение работы процедуры обработки каждого изображения при троекратном перезапуске программы. Сделать выводы об эффективности распараллеливания вычисления с помощью многопоточных вычислений на GPU. Оформить отчет.

Содержание отчета.

1. Титульный лист
2. Задача по полученному варианту.

(если последняя цифра 0 или 5: вариант 1;

если последняя цифра 1 или 6: вариант 2;

если последняя цифра 2 или 7: вариант 3;

если последняя цифра 3 или 8: вариант 4;

если последняя цифра 4 или 9: вариант 5

1. Листинг программы
2. Примеры входных и выходных файлов
3. Описание результатов тестирования
4. Выводы

**Вариант 1**

Загрузить цветное изображение.

Получить значения интенсивности ,

где – интенсивность пикселя *v*, – значение красной компоненты пикселя , – значение зелёной компоненты пикселя , – значение синей компоненты пикселя .

Установить значение скалярной величины - порога Threashold (от 1 до 255). Те значения интенсивности которые меньше порогового значения установить в 0, а которые больше Threashold установить в 1.

Выполнить операцию эрозии [https://habr.com/ru/post/113626/ или https://intuit.ru/studies/courses/10621/1105/lecture/17989?page=4] над полученной матрицей из 0 и 1. Примечание: Нужно задать шаг эрозии (1, 2 или 3). Получить изображение из результата путем установки вместо значений 0 – пикселей черного цвета (0, 0, 0), и вместо значений 1 – пикселей белого цвета. Сохранить результат в файл.

**Вариант 2**

Загрузить цветное изображение.

Установить шаг переноса по оси Х и оси Y [https://habr.com/ru/post/113626/]. Выполнить сдвиг всех пикселей на нужный шаг. Пограничные пиксели, содержавшие прежние пиксели, заменить цветом, определенным пользователем. Выполнить свертку с фильтром, добавляющим размытие, с каждым из цветовых каналов изображения. Ядро преобразования можно найти по ссылке: [https://docs.gimp.org/2.8/ru/plug-in-convmatrix.html]. Сохранить результат в файл.

**Вариант 3**

Загрузить цветное изображение.

Выполнить инвертирование цветов пикселей (новое значение цветового канала соответствует значению 255 – старое значение цветового канала). Выполнить свертку с фильтром, увеличивающим контраст, с каждым из цветовых каналов изображения. Ядро преобразования можно найти по ссылке: [https://docs.gimp.org/2.8/ru/plug-in-convmatrix.html]. Сохранить результат в файл.

**Вариант 4**

Загрузить цветное изображение.

Выполнить свертку с фильтром, придающим рельеф с каждым из цветовых каналов изображения. Ядро преобразования можно найти по ссылке: [https://docs.gimp.org/2.8/ru/plug-in-convmatrix.html]. Сохранить результат в файл.

**Вариант 5**

Загрузить цветное изображение.

Получить значения интенсивности ,

где – интенсивность пикселя *v*, – значение красной компоненты пикселя , – значение зелёной компоненты пикселя , – значение синей компоненты пикселя .

Установить значение скалярной величины - порога Threashold (от 1 до 255). Те значения интенсивности которые меньше порогового значения установить в 0, а которые больше Threashold установить в 1.

Выполнить операцию наращивания (диляции/ дилатации) [https://habr.com/ru/post/113626/ или https://intuit.ru/studies/courses/10621/1105/lecture/17989?page=4] над полученной матрицей из 0 и 1. Примечание: Нужно задать шаг наращивания (1, 2 или 3). Получить изображение из результата путем установки вместо значений 0 – пикселей черного цвета (0, 0, 0), и вместо значений 1 – пикселей белого цвета. Сохранить результат в файл.