

**МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)**

**Институт №8 «Компьютерные науки и прикладная математика»
Кафедра 806 «Вычислительная математика и программирование»**

**Лабораторная работа №2
по курсу «Программирование графических процессоров»**

Обработка изображений на GPU. Фильтры.

**Выполнил: К.М. Воронов
Группа: 8О-407Б
Преподаватель: А.Ю. Морозов**

Москва, 2022

Условие

Цель работы: научиться использовать GPU для обработки изображений.

Использование текстурной памяти и двухмерной сетки потоков.

Вариант 6. Выделение контуров. Метод Превитта.

Программное и аппаратное обеспечение

GPU:

- Название NVIDIA GeForce GTX 1050
- Compute capability: 6.1
- Графическая память: 4236378112
- Разделяемая память: 49152
- Константная память: 65536
- Количество регистров на блок: 65536
- Максимальное количество нитей: (1024, 1024, 64)
- Максимальное количество блоков: (2147483647, 65535, 65535)
- Количество мультипроцессоров: 5

Сведения о системе:

- Процессор: Intel Core i5-8300H 2.30GHz
- ОЗУ: 32 ГБ
- SSD 1ТБ
- HDD 1ТБ

Программное обеспечение:

- OS: Kubuntu 20.04
- Текстовый редактор: Sublime text
- Компилятор: nvcc

Метод решения

Для решения этой задачи я создаю две матрицы: m_x и m_y . Далее прохожусь по всем пикселям, получаю значение яркости и с помощью матриц считаю градиенты, сначала по x и по y , потом общий. Заполняю цвета значениями градиента.

Описание программы

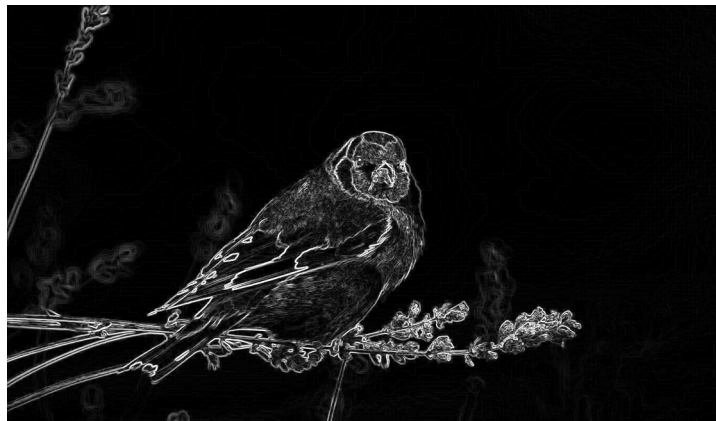
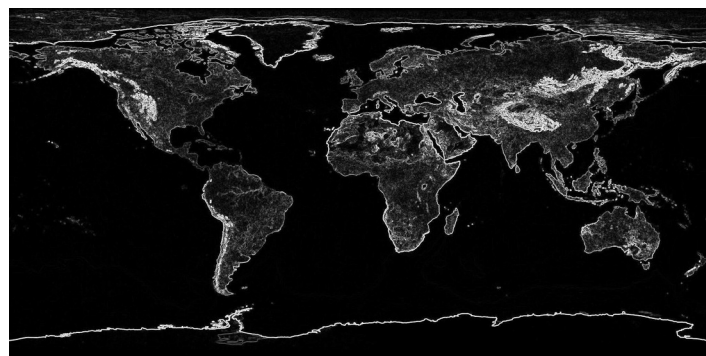
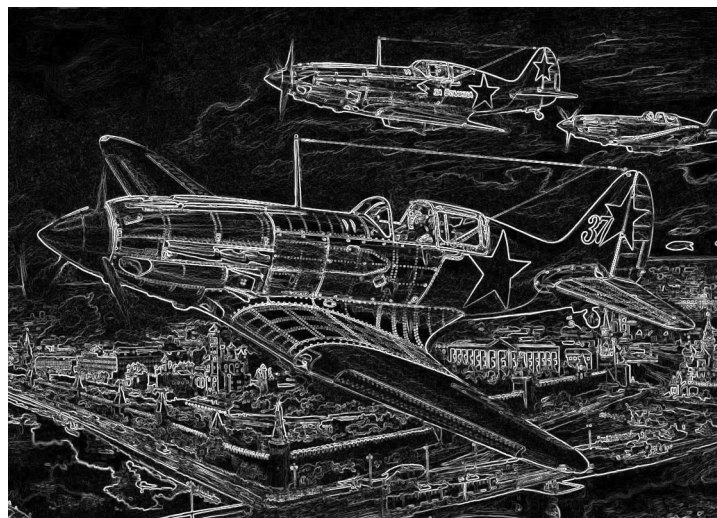
Для хранения картинки на видеокарте я использую объект текстуры `cudaTextureObject_t`. Ядро `kernel` принимает текстуру, выходной массив и размеры изображения. Для удобства использую двухмерную структуру блоков и потоков. После определения индекса потока, прохожусь по соответствующему окну и считаю градиент. Для подсчета компоненты яркости использовал функцию `conv`, которая работает на видеокарте. Она принимает 3 компоненты цвета и выдает компоненту яркости. Всё имеет тип `double`.

Результаты

Входное изображение



Выходное



CPU:

w*h	Время, мс
100	0.047000 ms
10000	1.550000 ms
1000000	153.206000 ms
100000000	17735.725000 ms

GPU:

<<< dim3(16, 16), dim3(32, 32) >>>

w*h	Время, мс
100	0.028544 ms
10000	0.059904 ms
1000000	3.196704 ms
100000000	294.330414 ms

<<< dim3(32, 32), dim3(32, 32) >>>

w*h	Время, мс
100	0.045920 ms
10000	0.086944 ms
1000000	3.199552 ms
100000000	293.121063 ms

<<< dim3(128, 128), dim3(8, 8) >>>

w*h	Время, мс
100	0.090688 ms
10000	0.114464 ms
1000000	3.135744 ms
100000000	279.188507 ms

<<< dim3(128, 128), dim3(32, 32) >>>

w*h	Время, мс
100	0.555776 ms
10000	0.595840 ms
1000000	3.744992 ms
100000000	279.351471 ms

Выводы

Выполнив данную лабораторную, я научился обрабатывать изображения с использованием технологии CUDA. Как видно из результатов, обработка изображения на видеокарте происходит намного быстрее, чем на процессоре. Во время выполнения работы я не столкнулся с особыми сложностями.