

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

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

**Лабораторная работа №3
по курсу «Программирование графических процессоров»**

Классификация и кластеризация изображений на GPU.

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

Москва, 2022

Условие

Цель работы. Научиться использовать GPU для классификации и кластеризации изображений. Использование константной памяти и одномерной сетки потоков.

Вариант 5. Метод k-средних.

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 создал цикл, в котором проверял, поменялись ли центры классов или нет. Если они не поменялись — обработка картинки закончена. В цикле происходит классификация пикселей изображения относительно центров с помощью метода минимального расстояния, а также обновление центров.

Описание программы

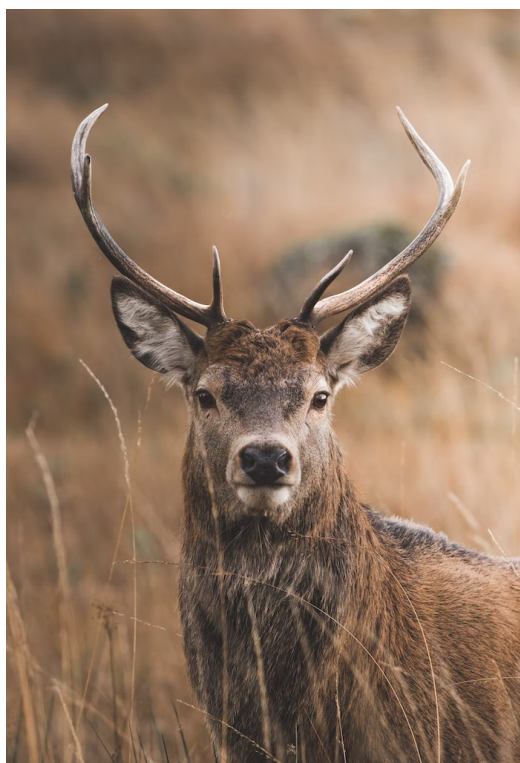
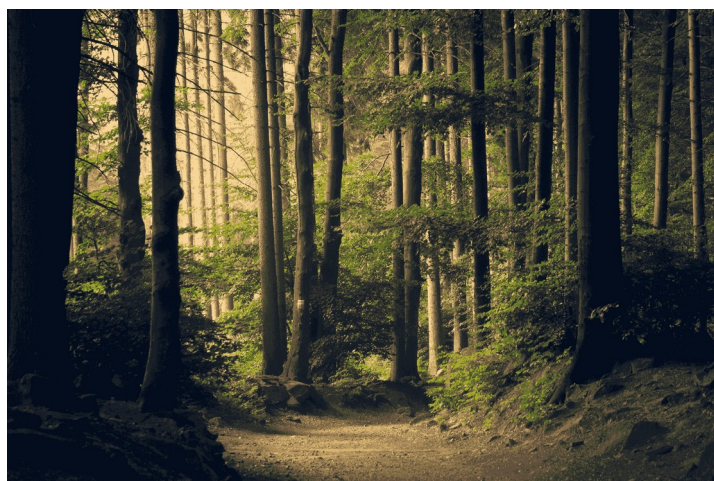
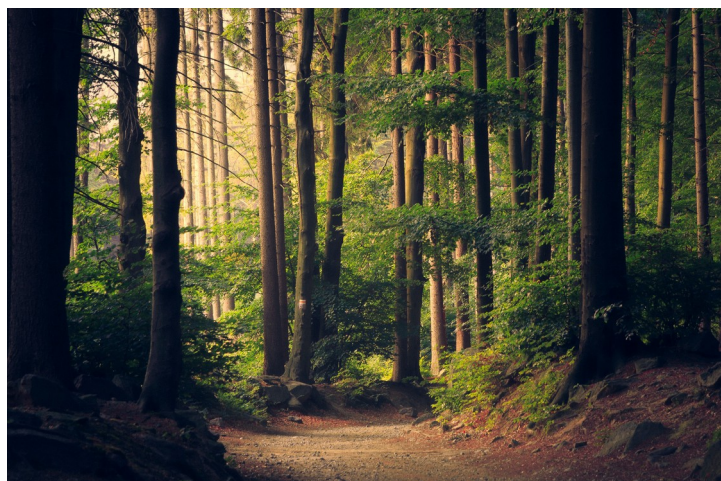
Для хранения значений цветов пикселей я использую типы uchar и double. Для сравнения цветов пикселей написан оператор !=, который принимает на вход все три цвета(double3) и поочередно сравнивает каждый цвет. Ядро kernel принимает на вход картинку, количество пикселей в ней, а также количество кластеров. Кластеры записываются в констанную память в цикле. Картинка также каждый раз перезаписывается.

Результаты

Входное изображение



Выходное



16 центров**CPU:**

w*h	Время, мс
100	0.089 ms
10000	8.487 ms
1000000	845.104 ms
100000000	72445.3 ms

GPU:

<<<1, 32>>>

w*h	Время, мс
100	0.538336 ms
10000	102.037437 ms
1000000	9153.844727 ms
100000000	1430341.125000 ms

<<<256, 256>>>

w*h	Время, мс
100	0.127424 ms
10000	13.422880 ms
1000000	4713.576660 ms
100000000	439568.656250 ms

<<< 512, 512 >>>

w*h	Время, мс
100	0.247136 ms
10000	11.932160 ms
1000000	5566.277344 ms
100000000	1251575.625000 ms

<<<1024, 1024>>>

w*h	Время, мс
100	0.251872 ms
10000	16.995487 ms
1000000	4532.570312 ms
100000000	278425.562500 ms

Выводы

Выполнив данную лабораторную, я научился классифицировать пиксели изображения с использованием технологии CUDA. Во время выполнения столкнулся с проблемой переполнения, во время подсчёта новых центров unsigned long long переполнялся, поэтому пришлось использовать тип double. Также из результатов замеров времени видно, что алгоритм с использованием видеокарты работает дольше, чем с использованием только процессора. Это объясняется тем, что нам приходится каждый раз копировать картинку, а это дорогостоящая операция по времени.