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.

Студент: А. А. Садаков
Группа: 8О-406Б
Преподаватели: К.Г. Крашенинников,
А.Ю. Морозов

Москва, 2022

Условие

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

Вариант 2: Метод расстояния Махаланобиса.

Программное и аппаратное обеспечение

ГРАФИЧЕСКИЙ ПРОЦЕССОР
Имя устройства: NVIDIA GeForce GTX 1650
Compute capability: 7.5
Размер графической памяти: 4242604032
Размер разделяемой памяти: 49152
Размер константной памяти: 65536
Максимальное количество регистров на блок: 65536
Максимальное количество потоков на блок: 1024
Количество мультипроцессоров: 14

ПРОЦЕССОР
Имя устройства: Intel Core I5-10300H
Архитектура: Comet Lake-H
Количество ядер (потоков): 4(8)
Базовая (максимальная) частота: 2,5(4,5)ГГц
Кеш 1-го уровня: 64Кб (на ядро)
Кеш 2-го уровня: 256Кб (на ядро)
Кеш 3-го уровня: 6Мб (всего)

ОПЕРАТИВНАЯ ПАМЯТЬ
Объём: 8Гб
Тип: DDR4
Частота:2933МГц

ЖЁСТКИЙ ДИСК
Имя устройства: Intel SSDPEKNW512G8L
Тип диска: SSD
Объём памяти: 512Гб
Скорость чтения/записи: 1500/1000 Мб/с

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
OS: Windows 10,Подсистема Ubuntu 20.04
IDE: Visual Studio Code
Компилятор: nvcc

Метод решения

Для преобразования изображения алгоритмом Махаланобиса, нужно предварительно вычислить вектора средних значений пикселей и ковариационные матрицы.

Описание программы

Для преобразования изображения алгоритмом Махаланобиса, нужно предварительно вычислить вектора средних значений пикселей и ковариационные матрицы. Их вычисление происходит на CPU и копируется в разделяемую память.

Далее при помощи формулы Малаханобиса находится класс с наименьшим номером, к которому был отнесен соответствующий пиксель.

Вызов $MahalanobisKernel$ происходит с количеством нитей на блок — 1024 и количеством блоков — 1024. В самом $MahalanobisKernel$ я вычисляю общий индекс исполняемой нити idx который и будет индексом пикселя, смещение $offset$, которое позволяет найти следующий обрабатываемый блок для нити.

Код функции:

```
1  __global__ void MahalanobisKernel (uint32_t w, uint32_t h, uint64_t nc,
       uint32_t *data) {
2      int idx = blockIdx.x * blockDim.x + threadIdx.x;
3      int offsetX = gridDim.x * blockDim.x;
4
5      for (uint64_t i = idx; i < w * h; i += offsetX) {
6          uint32_t color = data[i];
7          double max = getArg(color, 0);
8          uint32_t curr_class = 0;
9          for (uint64_t k = 1; k < nc; ++k) {
10             double tmp = getArg(color, k);
11             if (max < tmp) {
```

```
12                      max = tmp;
13                      curr_class = k;
14                  }
15              }
16              data[i] += curr_class;
17          }
18      }
```

Результаты

1. Замеры времени работы ядер с различными конфигурациями (время указано в микросекундах).

Конфи-гурации ядра \ Размер изо-бражений	16x16	256x256	1280x720	1920x1080	3840x2160
1, 32	453	158961	2199846	4599219	18851152
32, 32	68	5055	88657	168064	630839
32, 256	76	1426	25151	52567	217755
256, 256	75	1067	18587	38842	156823
1024, 1024	75	1146	18504	38516	160448

2. Сравнение с CPU

Размер входных данных	16x16	256x256	1280x720	1920x1080	3840x2160
GPU(256, 256)	75	1067	18587	38842	156823
CPU	381	134358	2317364	4917438	20208486

3. Примеры изображений

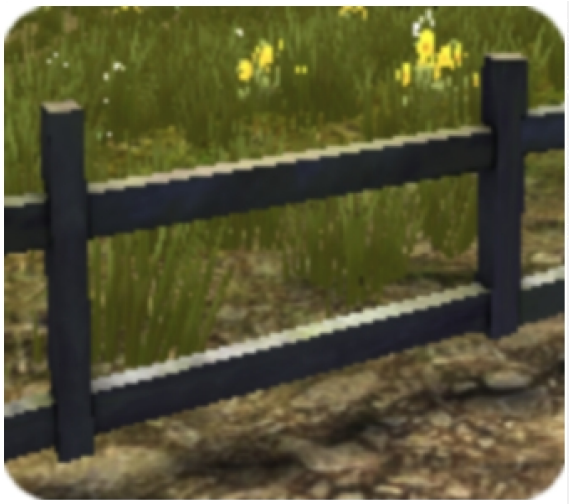

Рис. 1: До Махаланобиса

Рис. 2: После Махаланобиса

Рис. 3: До Махаланобиса

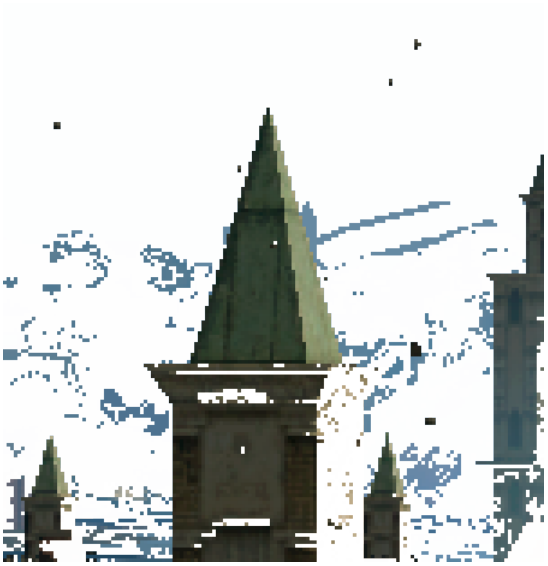

Рис. 4: После Махаланобиса

Выводы

Константная память кэшируется. Кэш существует в единственном экземпляре для одного мультипроцессора, а значит, общий для всех задач внутри блока. На хосте в константную память можно что-то записать, вызвав функцию $cudaMemcpyToSymbol$. Из устройства константная память доступна только для чтения.

Константная память очень удобна в использовании. Можно размещать в ней данные любого типа и читать их при помощи простого присваивания.