

**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: В. В. Бирюков
Группа: 8О-407Б
Преподаватель: А. Ю. Морозов

Москва, 2022

Условие

Цель работы: Научиться использовать GPU для классификации и кластеризации изображений. Использование константной памяти.

Вариант: 1. Метод максимального правдоподобия.

Программное и аппаратное обеспечение

Характеристики графического процессора:

- Наименование: GeForce GT 545
- Compute capability: 2.1
- Графическая память: 3150381056 Б
- Разделяемая память на блок: 49152 Б
- Количество регистров на блок: 32768
- Максимальное количество потоков на блок: (1024, 1024, 64)
- Максимальное количество блоков: (65535, 65535, 65535)
- Константная память: 65536 Б
- Количество мультипроцессоров: 3

Характеристики системы:

- Процессор: Intel(R) Core(TM) i7-3770 CPU 3.40GHz
- Память: 15 ГБ
- HDD: 500 ГБ

Программное обеспечение:

- ОС: Ubuntu 16.04.6 LTS
- IDE: Visual Studio Code 1.72
- Компилятор: nvcc 7.5.17

Метод решения

Для метода максимального правдоподобия необходимы три величины для каждого класса i : выборочное среднее avg_i , обратная матрица и определитель матрицы ковариации $conv_i$. Все эти величины предподсчитываются заранее, для вычисления определителя и обратной матрицы используются явно заданные формулы, так как матрицы имеют размер 3×3 . Для определения класса пикселя p используется дискриминантная функция следующего вида:

$$-(p - avg_i)^T \cdot conv_i^{-1} \cdot (p - avg_i) - \ln(|\det conv_i|)$$

пиксель принадлежит тому классу, при котором дискриминантная функция максимальна. Сложность алгоритма — $O(n \cdot w \cdot h)$, где n — количество классов.

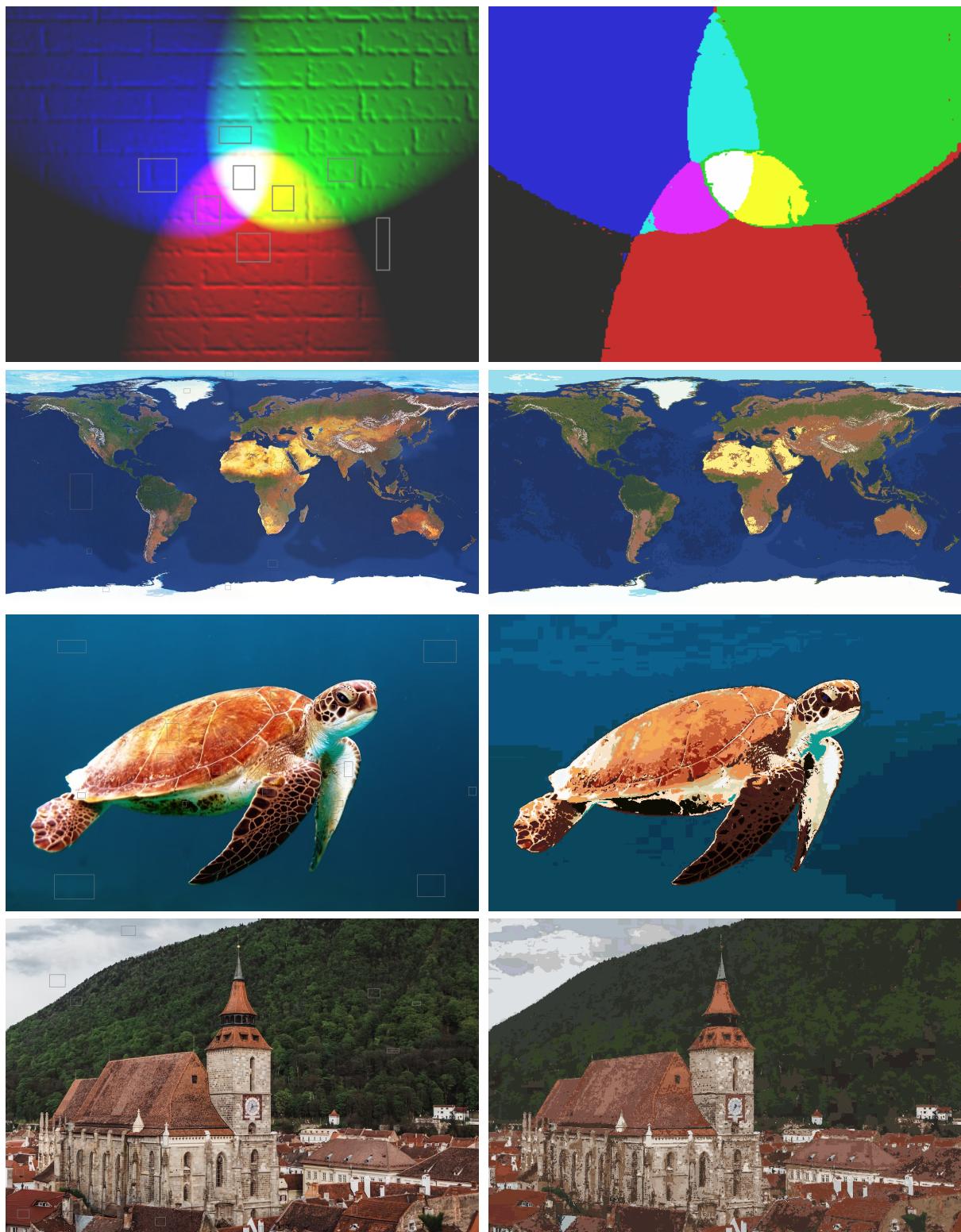
Описание программы

В файле `lab3.cu` расположен основной код программы. Функция `maximum_likelihood` — вычислительное ядро — осуществляет описанный выше алгоритм и принимает три аргумента: указатель на массив данных и размер изображения и количество классов. Нужные алгоритму данные о классах — выборочное среднее и обратная матрица ковариации и определитель матрицы ковариации — записаны в константную память. Файл `matrix.hpp` содержит классы векторов (для CPU реализации) и матрицы с необходимыми функциями и операторами.

Результаты

Примеры работы программы

Серыми прямоугольниками отмечены области, из которых составлены выборки классов.



Сравнение времени работы

Размер теста	100×100	500×500	1000×1000	1000×1000	2000×2000	2000×2000	5000×5000	5000×5000
Число классов	10	10	10	32	10	32	10	32
Конфигурация	Время выполнения, мс							
CPU	9.1379	189.2646	529.117	1545.433	1937.266	5923.956	11940.333	36821.500
1, 32	1.4703	38.0822	152.122	442.399	608.409	1769.516	3802.423	11058.966
32, 32	0.1618	3.4147	13.548	40.965	54.131	164.102	338.453	1028.226
64, 64	0.1383	2.7711	11.035	34.528	44.074	138.093	275.482	862.861
128, 128	0.1385	2.6698	10.624	33.326	42.452	133.250	265.203	832.737
256, 256	0.1378	2.6618	10.618	33.315	42.408	133.211	264.936	832.471
512, 512	0.1647	2.7680	10.757	33.754	42.873	134.950	267.780	843.309
1024, 1024	0.2754	2.9208	11.234	34.477	43.730	136.688	271.044	852.250

Выводы

В ходе выполнения лабораторной работы я познакомился с классификацией пикселей изображения. Алгоритмы из данной лабораторной могут использоваться как непосредственно для классификации, так и для сжатия изображения путем ограничения палитры. Однако, использование не более 32 оттенков приводит к не очень хорошим результатам.

Метод максимального правдоподобия (как и другие методы, использующие матрицу ковариации) отличается тем, что ему необходимо минимум четыре различных (не лежащих на одной плоскости в пространстве цветов) пикселя для каждого класса — нельзя задать конкретные цвета и классифицировать только по ним. Это усложняет тестирование на реальных изображениях. Один из выходов — ручное выделение областей, из которых составляются выборки классов.

Замеры времени в этот раз отличаются тем, что, начиная с конфигурации 128,128, при дальнейшем увеличении числа потоков время изменяется очень медленно. 128,128 является оптимальной конфигурацией.