

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

**Факультет прикладной математики и физики
Кафедра вычислительной математики и программирования**

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

**Обработка изображений на GPU.
Фильтры.**

Выполнил: Н.И. Забарин

Группа: 8О-408Б

Преподаватели: К.Г. Крашенинников,
А.Ю. Морозов

Москва, 2017

Условие

1. Цель работы:

Научиться использовать GPU для обработки изображений.
Использование текстурной памяти.

2. Вариант задания:

Вариант 2. Медианный фильтр.

Программное и аппаратное обеспечение

Спецификации GPU

Name:	GeForce GT 620M
Compute capability:	2.1
Warp size:	32
Max threads per block:	1024
Clock rate:	1250000
Multiprocessor count:	2
Max threads dim:	1024 1024 64
Max grid size:	65535 65535 65535

Спецификации видеопамати

Total global memory:	1024 MB
Shared memory per block:	48 KB
Registers per block:	32 KB
Total constant memory:	64 KB

Спецификации CPU

Процессор	Intel Core i5-3317U
Ядер	4
Базовая частота	1.7 GHz

Спецификация оперативной памяти

Объем памяти	10 Гб
Частота	1600 МГц

Спецификация жесткого диска

Тип	SSD
Интерфейс	M.2
Объем	240Gb

Спецификация программного обеспечения

CUDA Toolkit	7.5
OS	Ubuntu 16.10
IDE	Vim
Compiler	nvcc V7.5.17

Метод решения

Изображение хранится в текстурной памяти. Каждый поток обрабатывает один пиксель (если кол-во потоков меньше кол-ва пикселей, то каждый поток берет на себя больше пикселей). Для этого он строит гистограмму по каждому цвету и выбирает медиану. Медиана ищется с помощью подсчета префиксных сумм по гистограмме.

Описание программы

В программе имеется одно ядро

```
__global__ void kernel(uchar4 *dst, uint32_t w, uint32_t h, uint32_t radius).
```

В нем проходятся все пиксели, соотнесенные с данным потоком, и в каждый записывается посчитанные значения цвета в

```
__device__ void medpx(uchar4 &px, int x0, int y0, uint32_t radius, uint32_t w, uint32_t h).
```

В функции medpx подсчитывается тройной массив

```
uint16_t cnt[3][256];
```

- кол-во пикселей в окне с радиусом radius и центром в пикселе x, y с определенной красной, зеленой и синей компонентой цвета соответственно(0, 1, 2).

Для каждого массива вызывается функция

```
__device__ uint8_t medc(uint16_t *cnt, int mid),
```

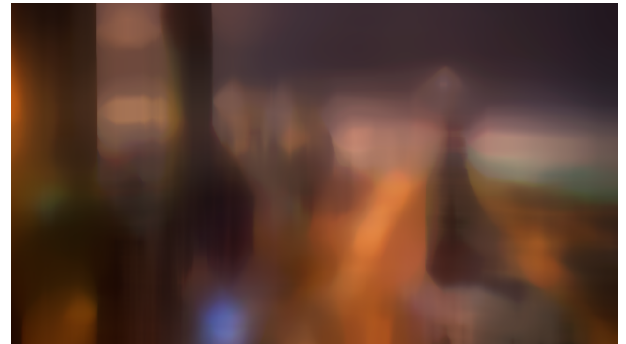
которая возвращает медиану поданного на вход массива cnt. $mid = s / 2$, где s – кол-во пикселей в окне. Медианой считается первый элемент, имеющий порядковый номер больше mid (начало счета с единицы).

Результаты

Замеры времени работы для изображения 648x365

Радиус	Параметры ядра	Время
1	dim3(8, 8), dim3(8, 8)	118.073570
10	dim3(8, 8), dim3(8, 8)	1117.185425
20	dim3(8, 8), dim3(8, 8)	4081.904297
1	dim3(16, 16), dim3(16, 16)	95.151108
10	dim3(16, 16), dim3(16, 16)	2101.118408
20	dim3(16, 16), dim3(16, 16)	7783.942871

Пример работы фильтра:



Выводы

Данный фильтр может быть применен для уменьшения шума на изображении. Он убирает мелкие детали на изображении, как бы “сглаживая его”. Во многих программах для обработки растровых изображений данный фильтр присутствует (пример - Adobe Photoshop).