Advanced Computer Architecture (ACA)
HW: GPU (CUDA) Image Filters

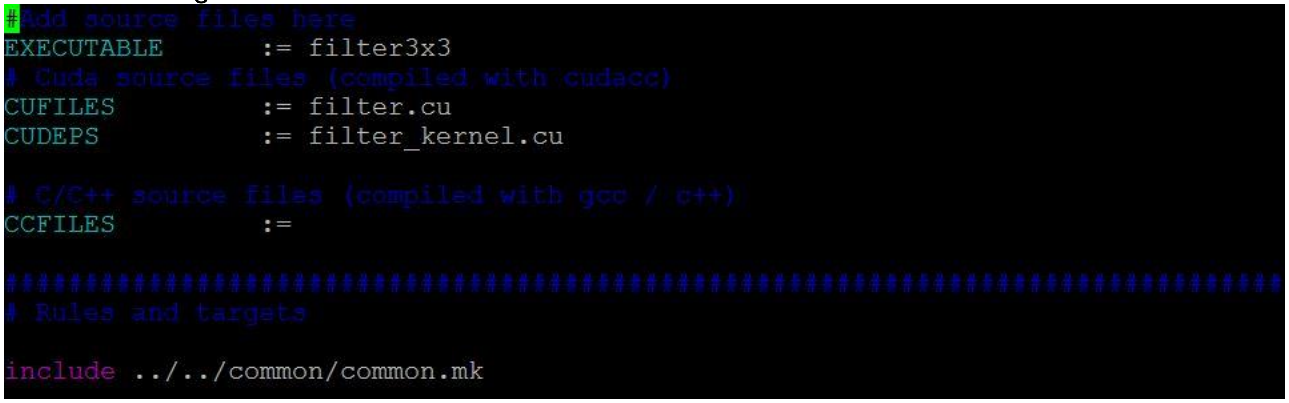
Vijoy Sunil Kumar

Files

3x3 kernel:	filter.cu	(sobel/average/boost)
	filter_kernel.cu	
5x5 kernel:	sobel5x5.cu	(sobel)
	filter_kernel5x5.cu	

Usage

Makefile changes to build 3x3 filter kernel



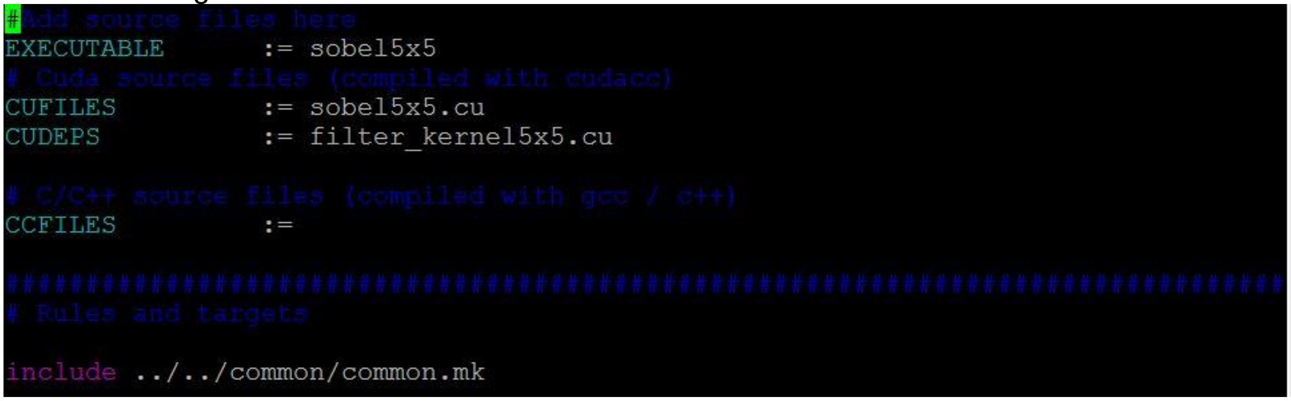
```
#Add source files here
EXECUTABLE      := filter3x3
# Cuda source files (compiled with cudacc)
CUFILES         := filter.cu
CUDEPS          := filter_kernel.cu

# C/C++ source files (compiled with gcc / c++)
CCFILES         :=

################################################################################
# Rules and targets

include ../../common/common.mk
```

Makefile changes to build 5x5 sobel kernel



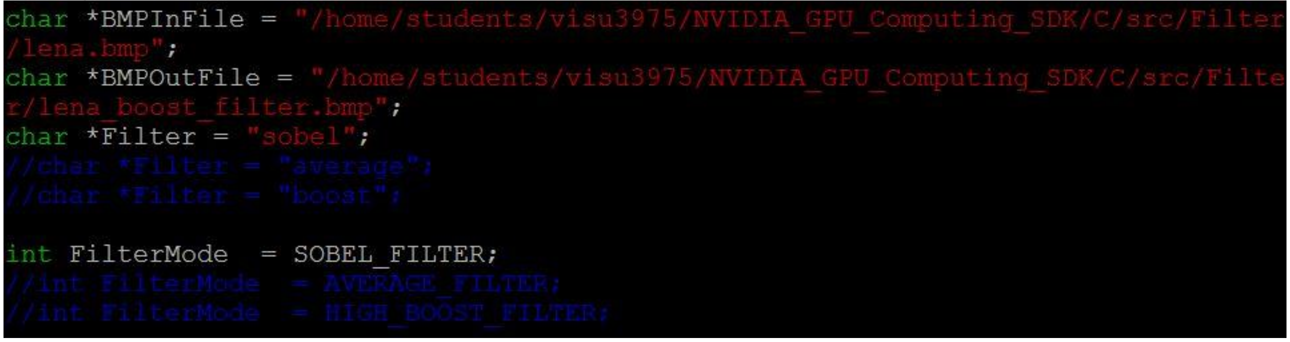
```
#Add source files here
EXECUTABLE      := sobel5x5
# Cuda source files (compiled with cudacc)
CUFILES         := sobel5x5.cu
CUDEPS          := filter_kernel5x5.cu

# C/C++ source files (compiled with gcc / c++)
CCFILES         :=

################################################################################
# Rules and targets

include ../../common/common.mk
```

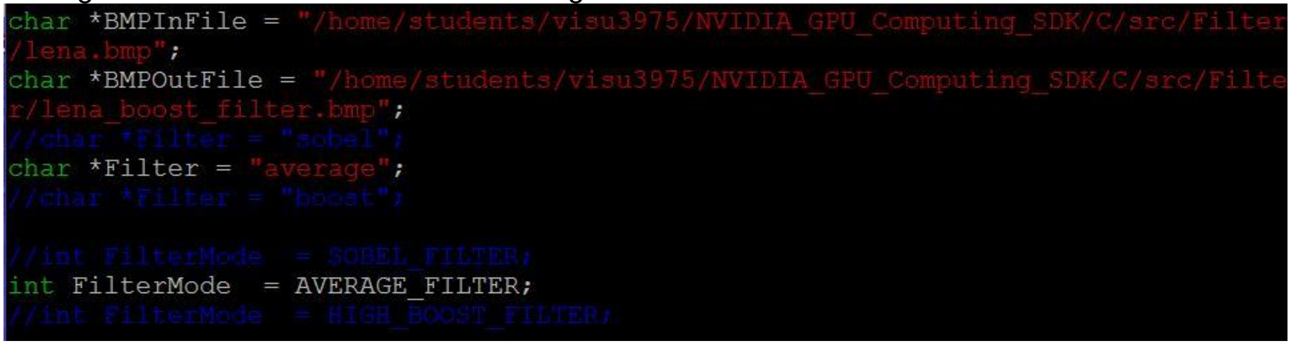
Changes in filter.cu file to build 3x3 Sobel filter



```
char *BMPInFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filter
/lena.bmp";
char *BMPOutFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filte
r/lena_boost_filter.bmp";
char *Filter = "sobel";
//char *Filter = "average";
//char *Filter = "boost";

int FilterMode  = SOBEL_FILTER;
//int FilterMode  = AVERAGE_FILTER;
//int FilterMode  = HIGH_BOOST_FILTER;
```

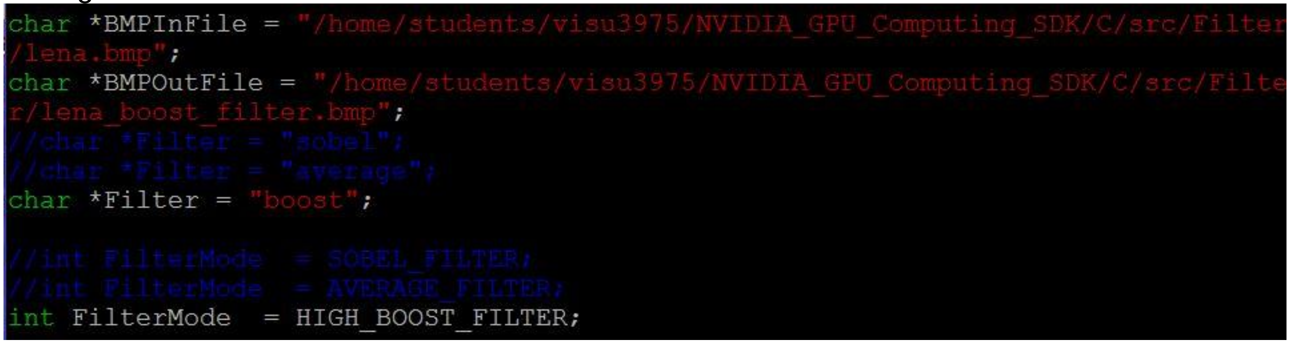
Changes in filter.cu file to build 3x3 Average filter



```
char *BMPInFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filter
/lena.bmp";
char *BMPOutFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filte
r/lena_boost_filter.bmp";
//char *Filter = "sobel";
char *Filter = "average";
//char *Filter = "boost";

//int FilterMode  = SOBEL_FILTER;
int FilterMode  = AVERAGE_FILTER;
//int FilterMode  = HIGH_BOOST_FILTER;
```

Changes in filter.cu file to build 3x3 Boost filter



```
char *BMPInFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filter
/lena.bmp";
char *BMPOutFile = "/home/students/visu3975/NVIDIA_GPU_Computing_SDK/C/src/Filte
r/lena_boost_filter.bmp";
//char *Filter = "sobel";
//char *Filter = "average";
char *Filter = "boost";

//int FilterMode  = SOBEL_FILTER;
//int FilterMode  = AVERAGE_FILTER;
int FilterMode  = HIGH_BOOST_FILTER;
```

Running 3x3 kernel sobel/average/boost filter

```
./filter3x3
```

Running 5x5 kernel sobel filter

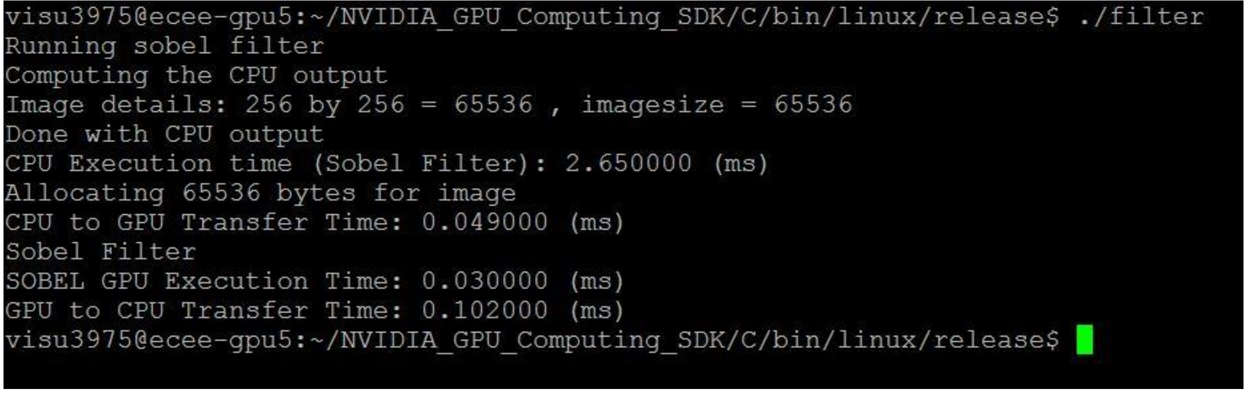
```
./sobel5x5
```

[Item 2] - Provide the execution time for the Sobel filter:

Input Size	GPU Execution Time (ms)	CPU Execution Time (ms)	Speedup %
lena.bmp	0.030	2.65	98.8
dublin.bmp	0.035	63.52	99.9

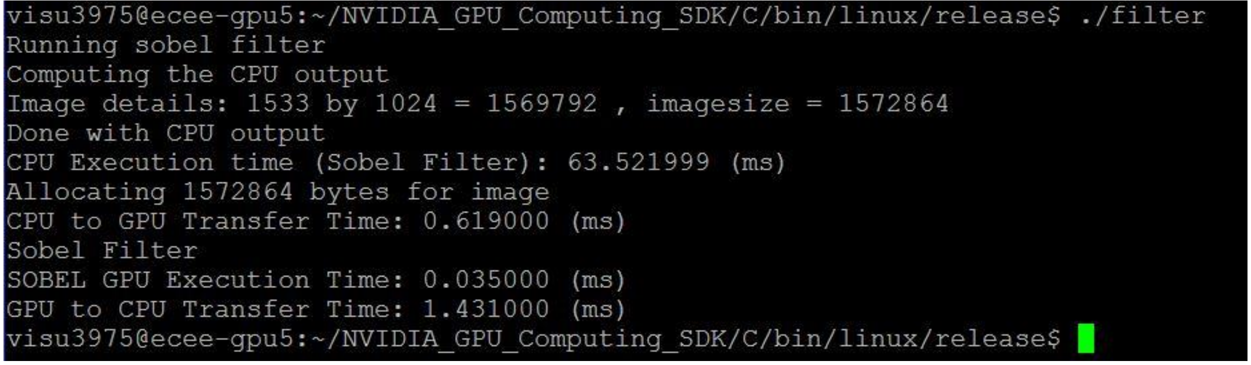
Screenshots

Sobel Filter run on lena.bmp in GPU



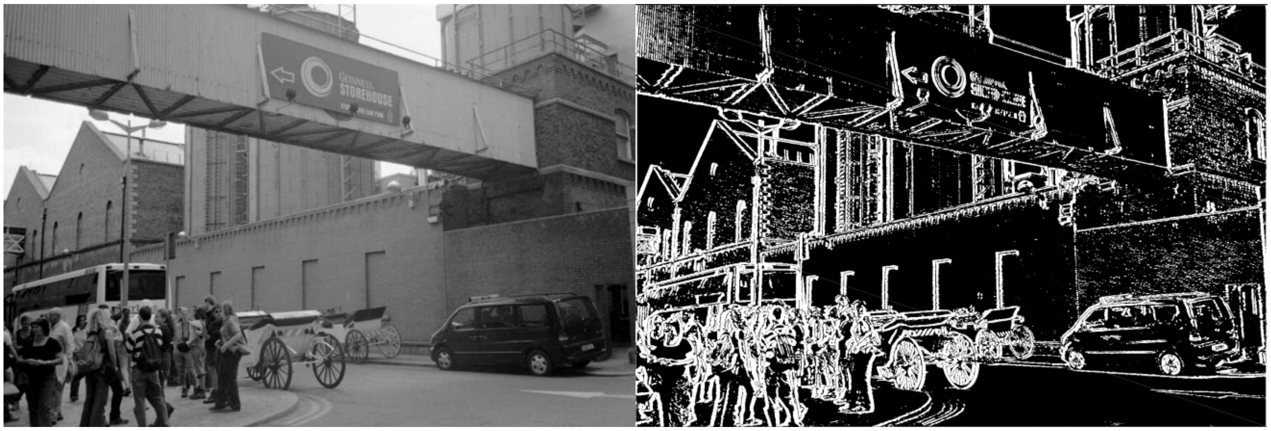
```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running sobel filter
Computing the CPU output
Image details: 256 by 256 = 65536 , imagesize = 65536
Done with CPU output
CPU Execution time (Sobel Filter): 2.650000 (ms)
Allocating 65536 bytes for image
CPU to GPU Transfer Time: 0.049000 (ms)
Sobel Filter
SOBEL GPU Execution Time: 0.030000 (ms)
GPU to CPU Transfer Time: 0.102000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$
```

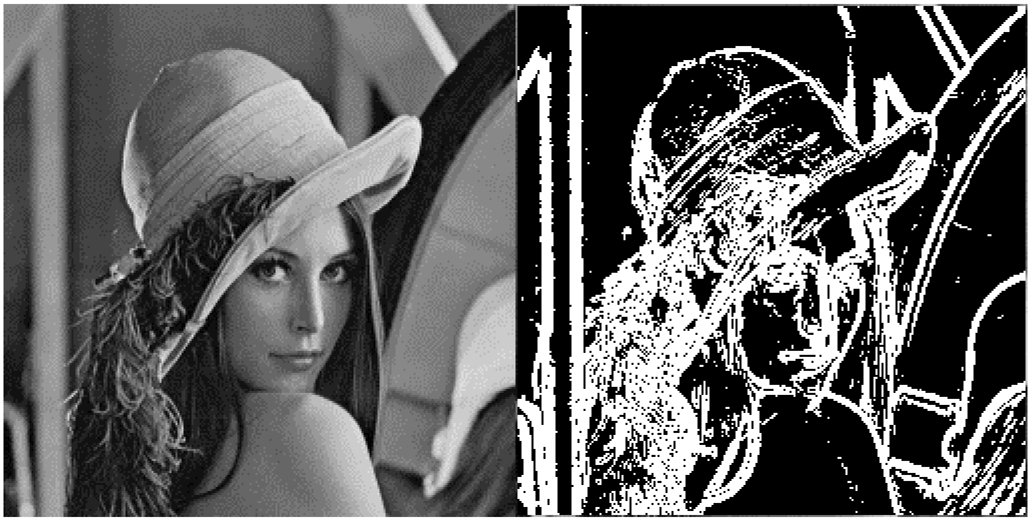
Sobel Filter on dublin.bmp in GPU



```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running sobel filter
Computing the CPU output
Image details: 1533 by 1024 = 1569792 , imagesize = 1572864
Done with CPU output
CPU Execution time (Sobel Filter): 63.521999 (ms)
Allocating 1572864 bytes for image
CPU to GPU Transfer Time: 0.619000 (ms)
Sobel Filter
SOBEL GPU Execution Time: 0.035000 (ms)
GPU to CPU Transfer Time: 1.431000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ 
```

Filter output

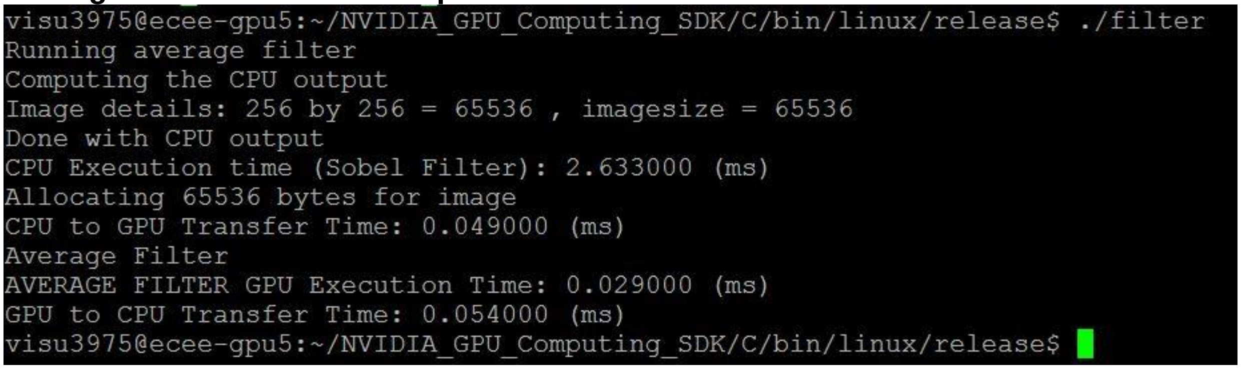




[Item 3] – Implement the Average Filter

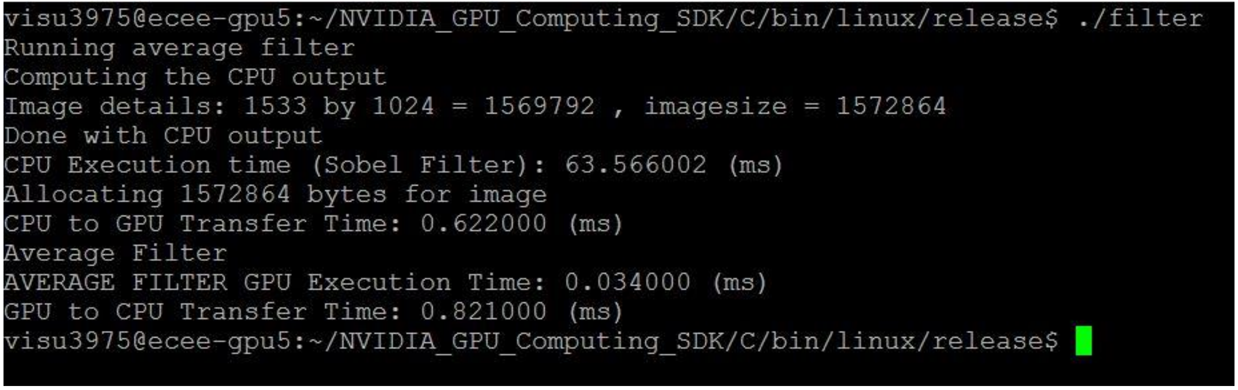
Screenshots

Average Filter run on lena.bmp in GPU



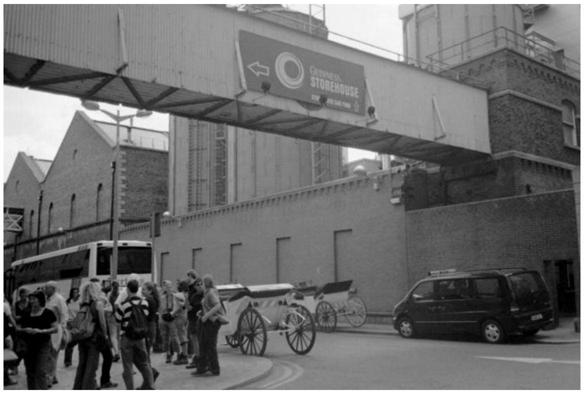
```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running average filter
Computing the CPU output
Image details: 256 by 256 = 65536 , imagesize = 65536
Done with CPU output
CPU Execution time (Sobel Filter): 2.633000 (ms)
Allocating 65536 bytes for image
CPU to GPU Transfer Time: 0.049000 (ms)
Average Filter
AVERAGE FILTER GPU Execution Time: 0.029000 (ms)
GPU to CPU Transfer Time: 0.054000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ █
```

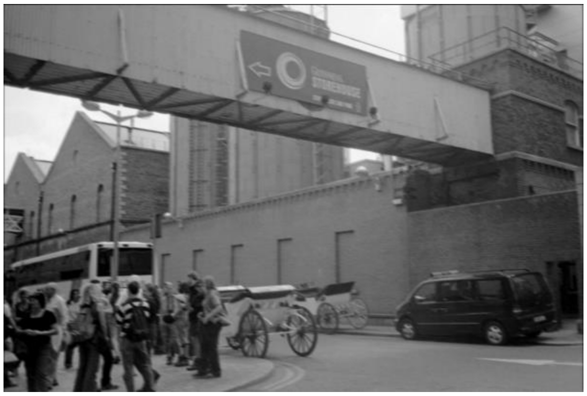
Average Filter run on dublin.bmp in GPU

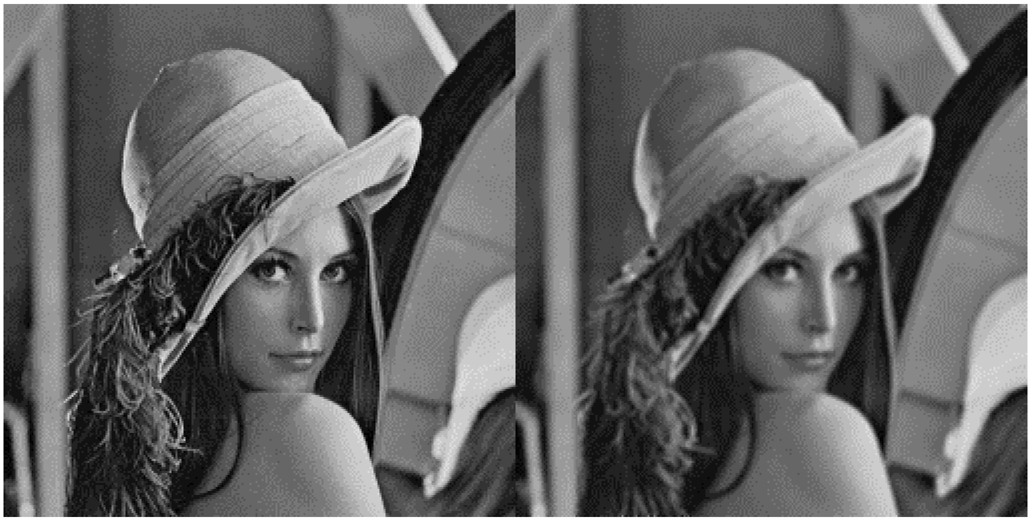


```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running average filter
Computing the CPU output
Image details: 1533 by 1024 = 1569792 , imagesize = 1572864
Done with CPU output
CPU Execution time (Sobel Filter): 63.566002 (ms)
Allocating 1572864 bytes for image
CPU to GPU Transfer Time: 0.622000 (ms)
Average Filter
AVERAGE FILTER GPU Execution Time: 0.034000 (ms)
GPU to CPU Transfer Time: 0.821000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ █
```

Filter output

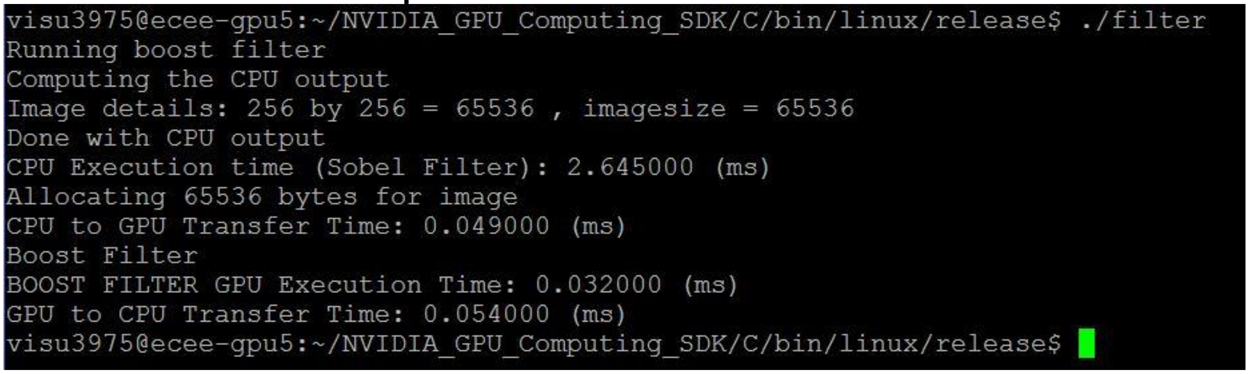






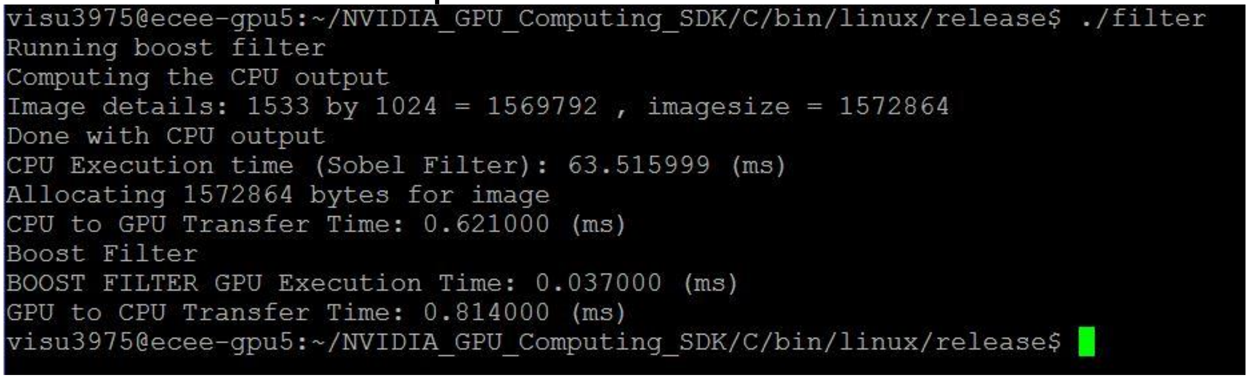
[Item 4] – Implement the Boost Filter

Boost Filter run on lena.bmp in GPU



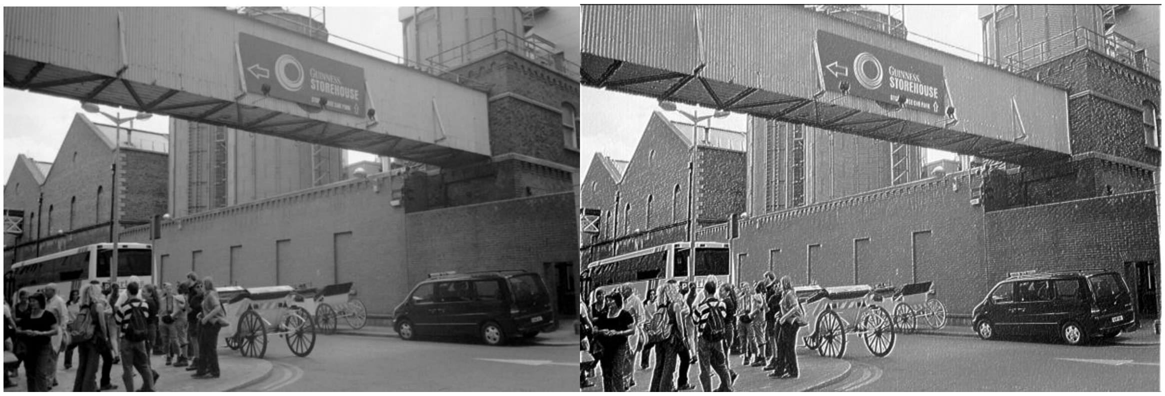
```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running boost filter
Computing the CPU output
Image details: 256 by 256 = 65536 , imagesize = 65536
Done with CPU output
CPU Execution time (Sobel Filter): 2.645000 (ms)
Allocating 65536 bytes for image
CPU to GPU Transfer Time: 0.049000 (ms)
Boost Filter
BOOST FILTER GPU Execution Time: 0.032000 (ms)
GPU to CPU Transfer Time: 0.054000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$
```

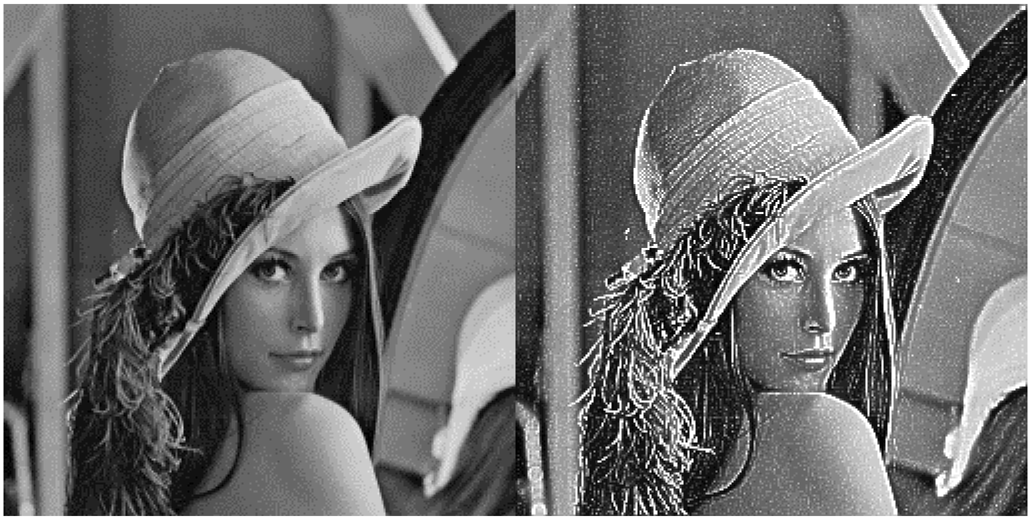
Boost Filter run on dublin.bmp in GPU



```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./filter
Running boost filter
Computing the CPU output
Image details: 1533 by 1024 = 1569792 , imagesize = 1572864
Done with CPU output
CPU Execution time (Sobel Filter): 63.515999 (ms)
Allocating 1572864 bytes for image
CPU to GPU Transfer Time: 0.621000 (ms)
Boost Filter
BOOST FILTER GPU Execution Time: 0.037000 (ms)
GPU to CPU Transfer Time: 0.814000 (ms)
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$
```

Filter output





[Item 4] – Graduate Students- You must rewrite the sobel filter to accept a 5x5 array kernel, using the following fomats:

$$\begin{bmatrix} 1 & 2 & 0 & -2 & -1 \\ 4 & 8 & 0 & -8 & -4 \\ 6 & 12 & 0 & -12 & -6 \\ 4 & 8 & 0 & -8 & -4 \\ 1 & 2 & 0 & -2 & -1 \end{bmatrix} \begin{bmatrix} -1 & -4 & -6 & -4 & -1 \\ -2 & -8 & -12 & -8 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 12 & 8 & 2 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

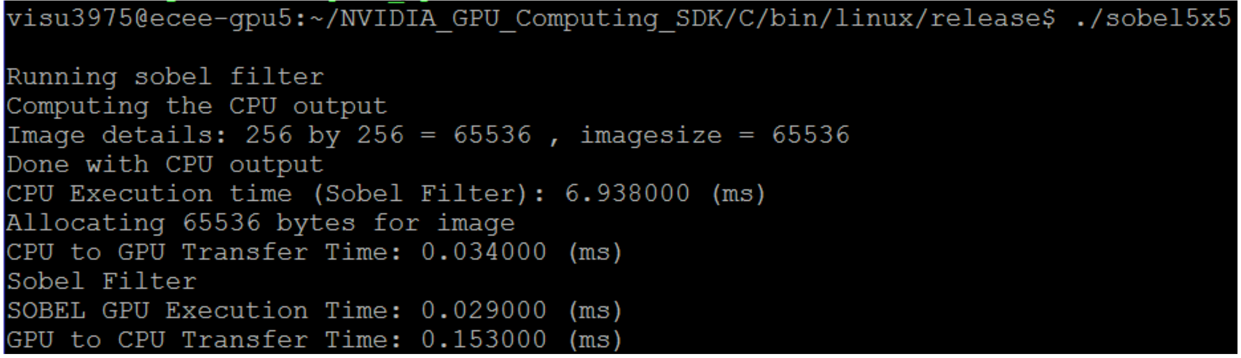
Your solution MUST pass the 5x5 array as an argument to the CUDA kernel. Currently the kernel uses a local variable array to implement the 3x3 filter kernel, which is very costly in terms of execution performance since each thread has an identical duplicated 3x3 array. Hint: FILTER_RADIUS is now 3 and FILTER_DIAMETER is 5.

You must submit your .cu to include a sobel5by5 kernel, and calculate the execution times and speedups:

Input Size	3x3 GPU Execution Time	5x5 GPU Execution Time	Speedup %
lena.bmp	0.030	0.029	3.33
dublin.bmp	0.035	0.034	2.85

Screenshots

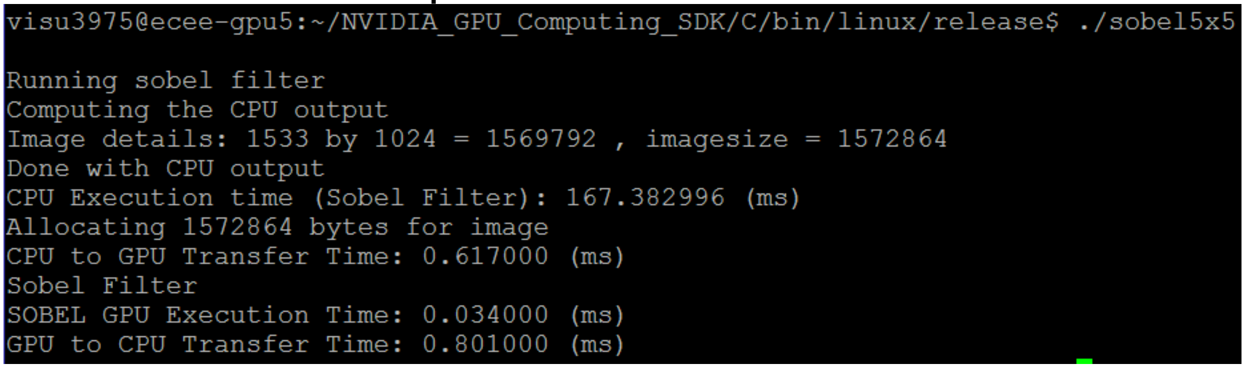
Sobel Filter run on lena.bmp in GPU with 5x5 Sobel Matrix



```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./sobel5x5

Running sobel filter
Computing the CPU output
Image details: 256 by 256 = 65536 , imagesize = 65536
Done with CPU output
CPU Execution time (Sobel Filter): 6.938000 (ms)
Allocating 65536 bytes for image
CPU to GPU Transfer Time: 0.034000 (ms)
Sobel Filter
SOBEL GPU Execution Time: 0.029000 (ms)
GPU to CPU Transfer Time: 0.153000 (ms)
```

Sobel Filter run on dublin.bmp in GPU with 5x5 Sobel Matrix



```
visu3975@ecee-gpu5:~/NVIDIA_GPU_Computing_SDK/C/bin/linux/release$ ./sobel5x5

Running sobel filter
Computing the CPU output
Image details: 1533 by 1024 = 1569792 , imagesize = 1572864
Done with CPU output
CPU Execution time (Sobel Filter): 167.382996 (ms)
Allocating 1572864 bytes for image
CPU to GPU Transfer Time: 0.617000 (ms)
Sobel Filter
SOBEL GPU Execution Time: 0.034000 (ms)
GPU to CPU Transfer Time: 0.801000 (ms)
```

Filter output