

# Term-Project 3, Serial implementation

Yacine Sahli

2019

## Program's inputs

`./main.exe input kernel output`

None of the parameters are optional.

- input can be a .jpeg .jpg .png image or a folder containing multiples images in the same format.
- kernel as to be a file containing a kernel with the dimensions of the kernel in the first line, for example a box blur kernel file

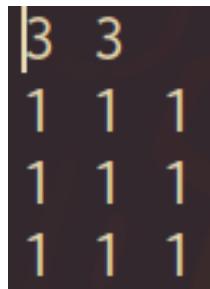


Figure 1: Kernel example

- output can be a .jpeg .jpg .png image or a folder.

## Execution time

reading and writing is very slow because of the compression/uncompression necessary for .png and .jpg

image:1620x1080 kernel:3x3 convolution time:0.443081 total time:1.94417

image:1620x1080 kernel:5x5 convolution time:1.07776 total time:2.57822

image:1620x1080 kernel:7x7 convolution time:2.00735 total time:3.5001

```
image:1620x1080 kernel:9x9 convolution time:3.25197 total time:4.74417
image:9865x3120 kernel:3x3 convolution time:7.83599 total time:33.3938
image:9865x3120 kernel:5x5 convolution time:20.6961 total time:46.2597
image:9865x3120 kernel:7x7 convolution time:37.1036 total time:63.7677
image:9865x3120 kernel:9x9 convolution time:62.328 total time:92.3733
```

## Output

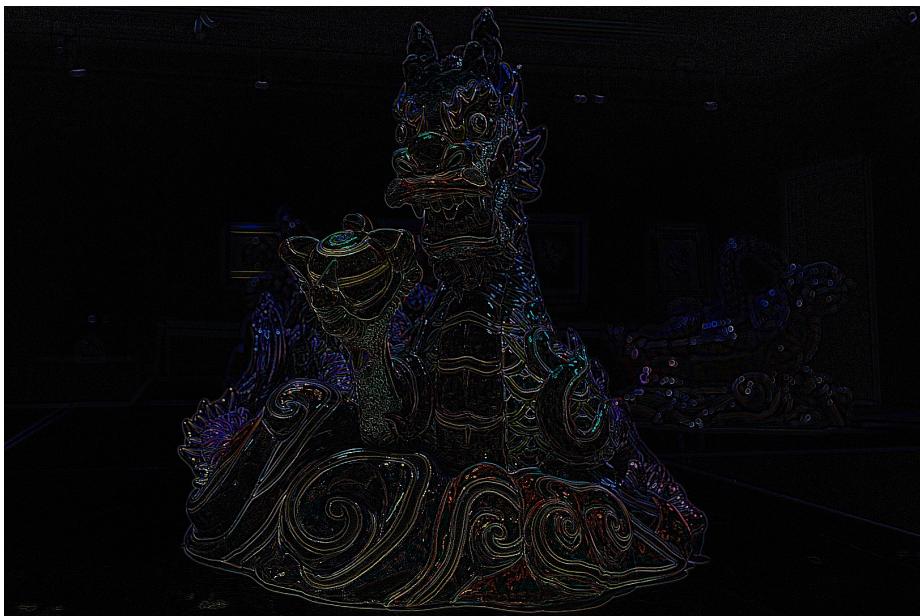
Here is the same image with 3 differents kernels used for testing my program.



Original dragon picture



Same picture after Gaussian blur 5x5



Same picture after High pass filter



Same picture after emboss filter

## Makefile

The program and his Makefile is on the group cluster at `/e10715005/final_project/ImageConvolutionOpenMP`  
`make test` will run a quick test.

`make valgrind` will run the same test with valgrind.

`make speed` will run multiple small test.

`make folder` will run a test on a folder

## Validations

See Output, resulting images are as expected by the filter, blurred, embossed or high passed.:wq