GPGPU

Adrien Barens

Alexandre Lemonnier

Sarah Gutierez

Victor Simonin

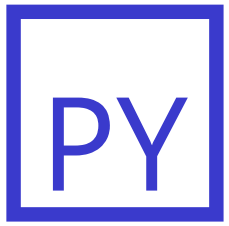
DESCRIPTION OF THE PROBLEM

- Detect objects on an image, a sequence of images
- Compare different methods of implementation:
  - CPU methods
  - GPU methods, applying data-parallelism concepts
- Benchmark the different methods

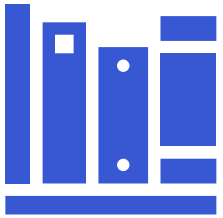
THE ALGORITHM

- Reference image and image(s) to test
- Grayscale
- Apply Gaussian blur
- Difference between reference and test image(s)
- Morphological opening and closing with a disk (or retangle)
- Threshold on image
- Connected components

CPU IMPLEMENTATION



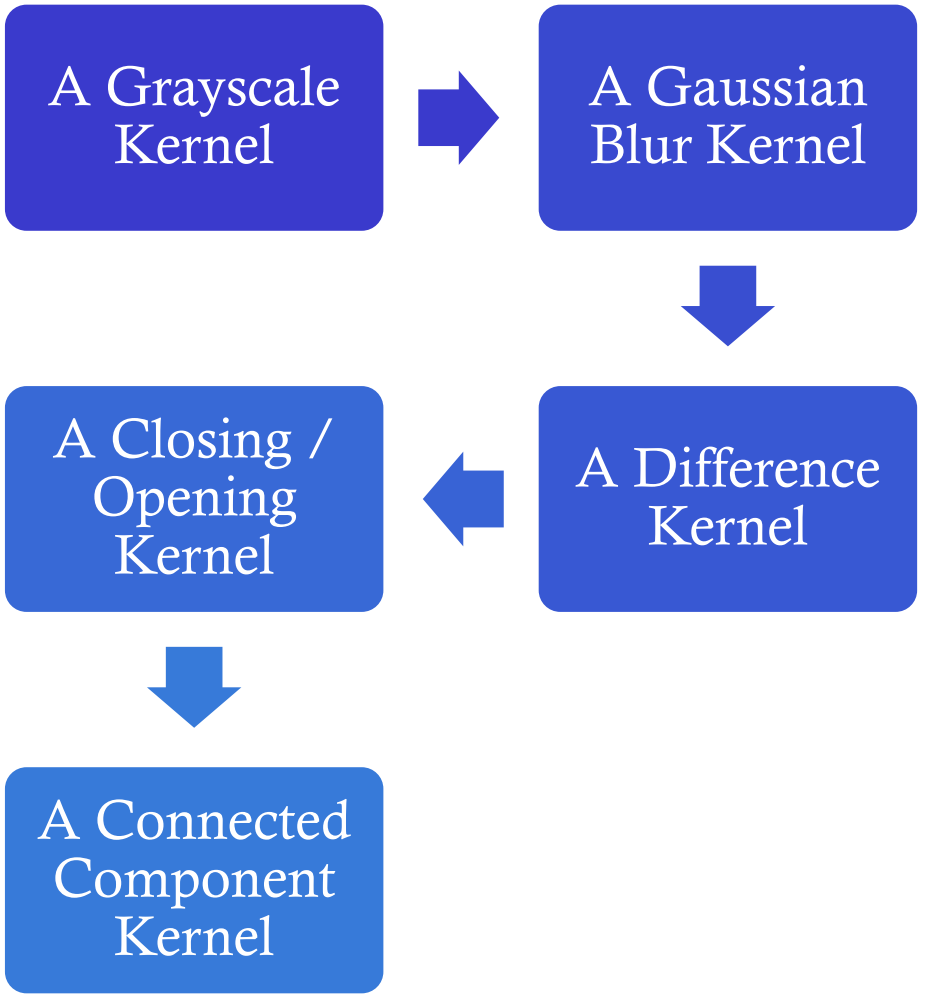
Python and OpenCV



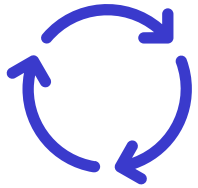
C++ and OpenCV

C++ from scratch

GPU IMPLEMENTATION



PERFORMANCE INDICATORS

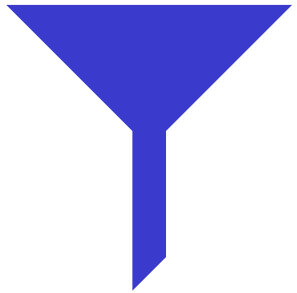


Number of iterations

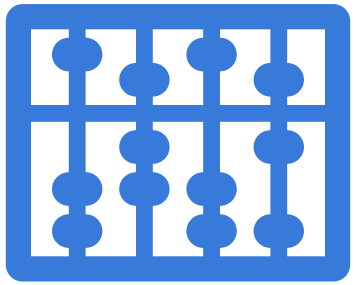
Time

FPS

BOTTLENECK

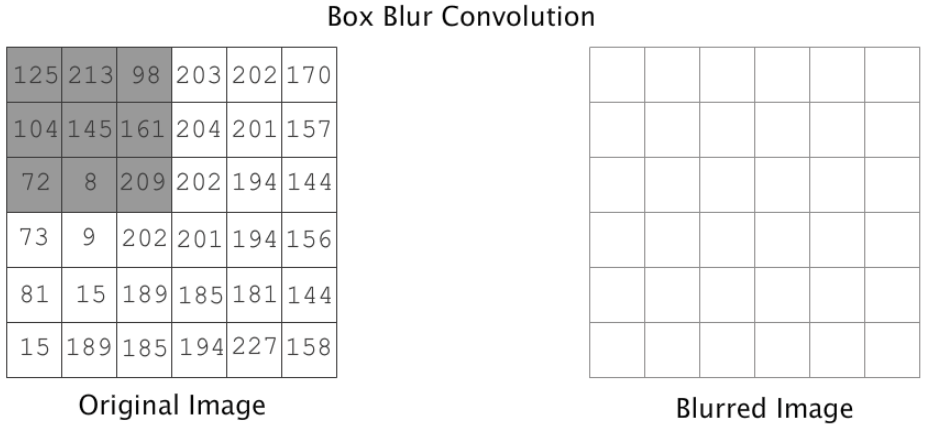


Gaussian Blur

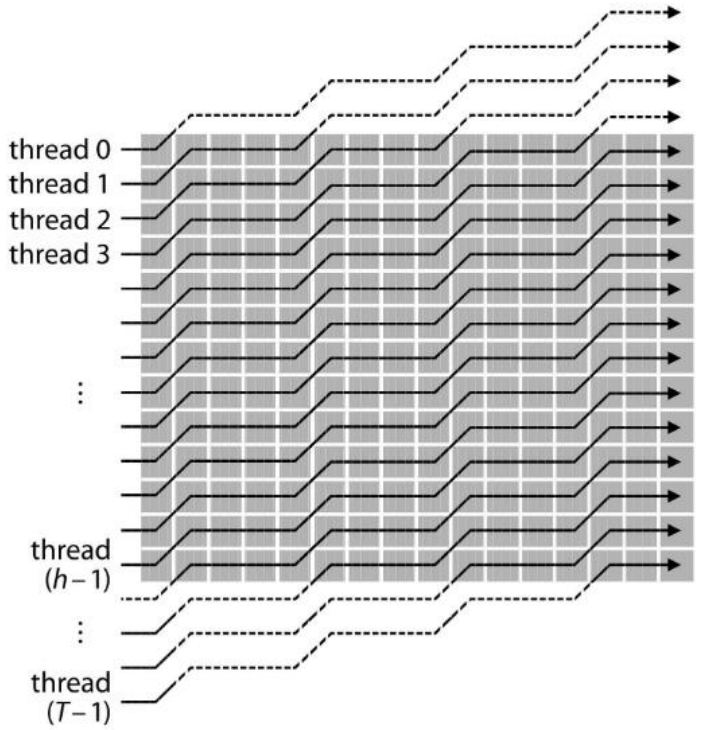


Morphological operation closing and opening

IMPROVEMENT OVER THE GPU IMPLEM.



Fast Blur



Morard and Bartovsky algorithms

SUMMARY TABLE OF PERFORMANCE

```
2022-11-29T18:13:56+01:00
Running ./benchmark
Run on (12 X 4641.35 MHz CPU s)
CPU Caches:
  L1 Data 32 KiB (x6)
  L1 Instruction 32 KiB (x6)
  L2 Unified 512 KiB (x6)
  L3 Unified 32768 KiB (x1)
Load Average: 1.80, 1.16, 0.50
```

Benchmark	Time	CPU	Iterations
BM_Grayscale	330781 ns	330548 ns	2113
BM_Grayscale_CV	27312 ns	27288 ns	25883
BM_Gaussian_Filter	19569900 ns	19551168 ns	36
BM_Gaussian_Filter_CV	34410 ns	34118 ns	20886
BM_Difference	44839 ns	44789 ns	12701
BM_Difference_CV	6242 ns	6236 ns	111919
BM_Opening_Closing	196314319 ns	196173976 ns	4
BM_Opening_Closing_CV	5166 ns	5163 ns	148774
BM_Threshold	59332 ns	59270 ns	11938
BM_Threshold_CV	4297 ns	4280 ns	166368
BM_Connected_Component	371191 ns	370818 ns	1888
BM_Connected_Component_CV	257539 ns	256218 ns	2720
BM_Pipeline	234961644 ns	234786074 ns	3
BM_Pipeline_CV	591758 ns	584706 ns	1236

DEMONSTRATION

THANKS FOR YOUR ATTENTION.

ANY QUESTIONS?