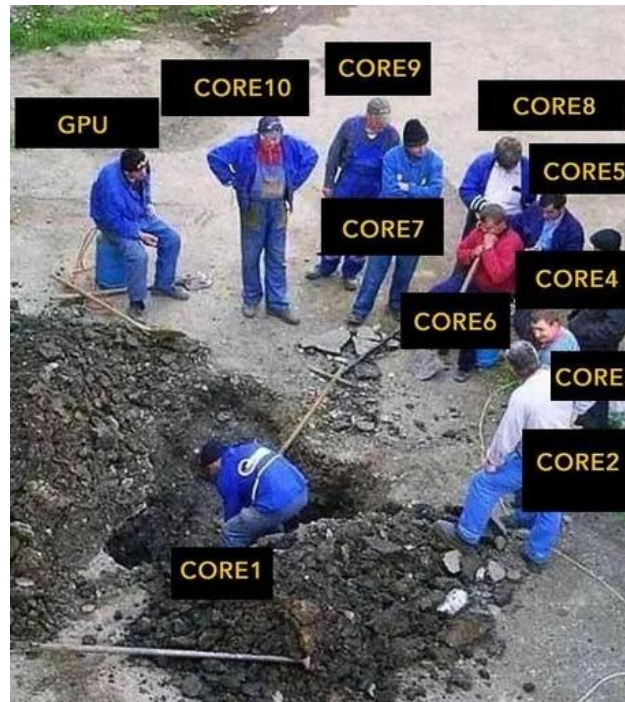


Soal Terkait:

## Fotosop (up to 5 poin + 30 poin bonus)



### Ketentuan

Buatlah sebuah program yang melakukan filtering terhadap sebuah gambar dengan ketentuan berikut

- Program memproses pixel pada gambar secara paralel
- Program menerima 1 input, yaitu *path* dari image yang akan dilakukan filtering (mungkin terdapat tambahan input untuk filter tertentu)
- Program bebas diimplementasikan menggunakan bahasa dan library bawaan untuk *multithreading* / *process*

Berikut filtering dasar yang dapat diimplementasikan:

- (Poin 1) Grayscale (mengubah warna menjadi skala abu-abu dengan perhitungan  $(R+G+B)/3$ )
- (Poin 2) Contrast (menambah kontras pada gambar dengan perhitungan  $\max(0, \min(255, c * (v - 128) + 128))$  dimana  $c$  merupakan nilai kontras yang diinginkan dari inputan pengguna dan  $v$  merupakan nilai dari pixel. Lakukan perhitungan untuk setiap channel  $v = \{R, G, B\}$ ).
- (Poin 2) Saturation (memperbesar saturasi warna dari gambar dengan mengonversi color space RGB menjadi HSL, mengalikan nilai saturation dengan nilai inputan pengguna dan kemudian mengembalikannya menjadi RGB)

### Bonus

- (Poin 1/2/3) Menggunakan GPU parallel dengan CUDA atau OpenCL (1 poin untuk setiap filtering dasar).
- (Poin 2) Real-time live preview edit (jika input filter diubah, maka gambar akan otomatis ditampilkan sesuai dengan input filter. Implementasikan dengan GUI)
- (Poin 2) Real-time filtering image dari feed camera (Implementasikan dengan GUI).
- Mengimplementasikan filtering lanjutan:
  - (Poin 5/8) Blur (Gaussian, Box, atau Directional, tipe/algoritmanya dibebaskan tetapi cukup salah satu saja, 8 poin jika dengan GPU)

- (Poin 10/15) Edge detection (misalnya dengan Sobel operator, 15 poin jika dengan GPU)

### Tujuan

- Belajar sistem paralel (seru guys)
- Simulasi menjadi software engineer di Adobe

### Berkas

- Source code
- Dokumen berisikan penjelasan singkat penggunaan program

Program diimplementasiin pake bahasa python dan library pycuda, algoritma utamanya ada di file sotosop.py terus bagian cuda\_kernel (full cuda cuy). Ada beberapa algoritma percobaan dll di obsolete, siapa tau perlu pengerjaan yang make threading (ada make library cupy juga yang agak ngecheat di situ sih ehehe)

```
10
11 cuda_kernel = """
12 _global_ void grayscale(const float* input, float* output, uint64_t size) {
13     int idx = blockIdx.x * blockDim.x + threadIdx.x;
14     if (idx < size) {
15         int chidx = idx * 3;
16         float gray = (input[chidx] + input[chidx + 1] + input[chidx + 2]) / 3.0;
17         output[chidx] = gray;
18         output[chidx + 1] = gray;
19         output[chidx + 2] = gray;
20     }
21 }
22
23 _global_ void contrast(const float* input, float* output, float value, uint64_t size) {
24     int idx = blockIdx.x * blockDim.x + threadIdx.x;
25     if (idx < size) {
26         float calculated = value * (input[idx] - 128) + 128;
27         if (calculated > 255) {
28             calculated = 255;
29         }
30         else if (calculated < 0) {
31             calculated = 0;
32         }
33         output[idx] = calculated;
34     }
35 }
36
37 _global_ void saturation(const float* input, float* output, float value, uint64_t size) {
38     int idx = blockIdx.x * blockDim.x + threadIdx.x;
39     if (idx < size) {
40         output[idx] = input[idx] * value;
41     }
42 }
43
44 _global_ void blur(const float* input, float* output, int value, int width, int height, uint64_t size) {
45     int idx = blockIdx.x * blockDim.x + threadIdx.x;
46     int limitval = value * 3;
47 }
```

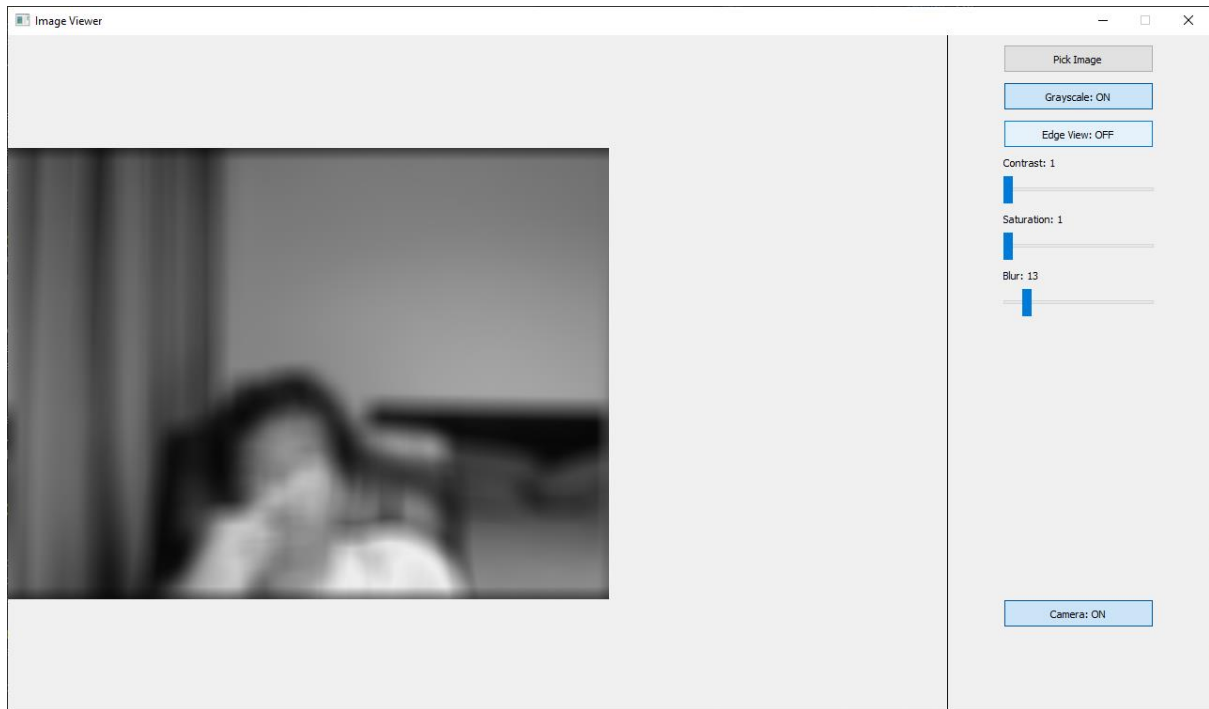
Dependensinya python (ditulis di 3.10.10), opencv, pycuda (make cuda 11.7), numpy, sama pyqt5

Buat Jalanin program utamanya langsung cus aja main.py, polosan banget tapi yang penting jalan lah ya



Langsung load image sama ganti ganti perubahan di kanan, harusnya self explanatory

Feed kamera jalan dan bisa diedit real time



Gui statik ga resize, banyak bug di gui (banyak bantuan chatgpt buat guinya ehe), tapi yaudahlahya yang penting algoritmanya gak sih 😞