

Lab 6 - Map

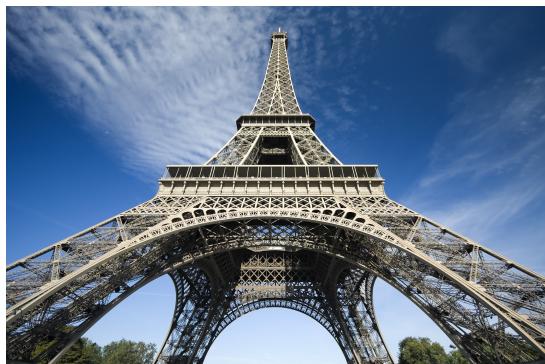
NGUYEN Duc Tung

1. Binarization

I get the binarization threshold from the command line, and process each pixel based on the threshold. The calculation is optimized a bit for GPU threads by using a single instruction:

```
1 binary = min(greyscale / threshold, 1) * 255;
```

Time elapsed: 21.9 ms



(a) Original image



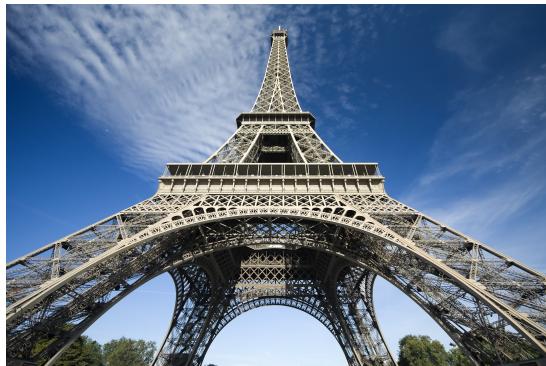
(b) Binarization with threshold = 128

2. Brightness control

The brightness task also get user's parameter for brightness changes. The 1-line max/min methods used to handle changes that exceed the range of color's value.

```
1 unsigned char r = min(max(input[globalId * 3] +
    brightnessChange, 0), 255);
2 unsigned char g = min(max(input[globalId * 3 + 1] +
    brightnessChange, 0), 255);
3 unsigned char b = min(max(input[globalId * 3 + 2] +
    brightnessChange, 0), 255);
```

Time elapsed: 21.8 ms



(a) Original image



(b) Brightness +50

3. Blending images

The task get the blending ratio from users, and then compute the average pixel's colors between two images by the ratio.

Here is the image blended between Eiffel image and national flag of France, with ratio = 0.5.

Time elapsed: 26.4 ms

