

Lab 8 - Scatter

NGUYEN Duc Tung

In this labwork, I implemented a RGB to HSV conversion, and a HSV to RGB conversion. Each pixel is converted using the given formula. The kernel's reliability is checked by switching the image back and forth between RGB and HSV without changing the image.

There is no dedicated optimization for Scatter yet. But I just tweaked a bit on the if/else to avoid 1 branch divergent.

```
1 if (delta == 0) {
2     hue = 0;
3     saturation = 0;
4 } else {
5     saturation = delta / max;
6
7     if (max == R) {
8         hue = 60 * (((G - B) / delta) % 6);
9     } else if (max == G) {
10        hue = 60 * ((B - R) / delta + 2);
11    } else {
12        hue = 60 * ((R - G) / delta + 4);
13    }
14 }
```

instead of

```
1 if (max == 0) {
2     saturation = 0;
3 } else {
4     saturation = delta / max;
5 }
6
7 if (delta == 0) {
8     hue = 0;
9 } else {
10    if (max == R) {
11        hue = 60 * (((G - B) / delta) % 6);
12    } else if (max == G) {
13        hue = 60 * ((B - R) / delta + 2);
14    } else {
15        hue = 60 * ((R - G) / delta + 4);
16    }
17 }
```

The image remains the same before and after doing color space conversion.

Time elapsed: 27.5 ms



(a) Original image



(b) Image after conversion