! https://zhuanlan.zhihu.com/p/646056979

! https://zhuanlan.zhihu.com/p/645495938

PS中可选颜色的数学原理

可选颜色

所谓的可选就是当前图像中哪些颜色会受到接下来的操作影响

可选颜色一共有9个颜色选项

分别是
红R,绿G,蓝B
青C,品红M,黄H
白W,灰G,黑K

下方有连个选项，相对和绝对

还有四个滑块
青色，洋红，黄色，黑色

控制逻辑

如果一个像素点$Pix=[R,G,B]$
$$\left{\begin{aligned}
被红色控制& & max(R,G,B)=R\\
被绿色控制& & max(R,G,B)=G\\
被蓝色控制& & max(R,G,B)=B\\
被黄色控制& & min(R,G,B)=B\\
被洋红控制& & min(R,G,B)=G\\
被青色控制& & min(R,G,B)=R\\
被黑色控制& & R<128且G<128且B<128\\
被灰色控制& & [R,G,B]\neq [0,0,0]且[R,G,B]\neq [255,255,255]\\
被白色控制& & R>128且G>128且B>128\
\end{aligned}\right.$$
并且调整范围都是$[-1,1]$
找到了被谁控制的逻辑
接下来就是如何计算

计算方法如下
如果是绝对
如果是相对
$$m=\left{\begin{aligned}
1&&绝对\
1-\dfrac{v}{N}&&相对
\end{aligned}\right.$$

$$clip(value,min,max)=\left{\begin{aligned}
max &&value>max\
min&&value<min\
value&&else
\end{aligned}\right.$$

$$range=\left{\begin{aligned}
max-med &&选“红绿蓝”控制\
med-min &&选“洋红、青、黄”控制\
255-(|max-128|+|min-128|)&&选“灰”控制\
2\times min - 255&&选“白”控制\
255-2\times max &&选“黑”控制\
\end{aligned}\right.$$
总体公式为
$$SelectiveColor(value,\alpha,\alpha_K)=clip(((-1-\alpha)\times \alpha_K-\alpha)\times m,-\dfrac{value}{255},1-\dfrac{value}{255})\times range$$
此处计算的是变动的大小
最终结果是
对于每个通道结果
$result = value+SelectiveColor(value,\alpha,\alpha_K)$

其中α代表对应被控制选项的数值大小，可以是CMY中任何一个，α_K表示黑色的大小
此处代表每个通道数值，如果是R通道，则计算第一个滑块和黑色滑块，其他以此类推。

于是，可以使用java编码验证

```
public static BlendColor selectiveColor(BlendColor inColor, String
selectiveColor, boolean isAbs,
            double ratioCyan, double ratioMagenta, double ratioYellow,
double ratioBlack) {
        double range = 0;
        if (inColor.getMax().name == "red" && selectiveColor == "red"
                || inColor.getMax().name == "blue" && selectiveColor ==
"blue"
                || inColor.getMax().name == "green" && selectiveColor ==
"green") {
            range = inColor.getMax().get255Value() -
inColor.getMid().get255Value();
        }

        if (inColor.getMin().name == "blue" && selectiveColor ==
"yellow"
                || inColor.getMin().name == "green" && selectiveColor ==
"magenta"
                || inColor.getMin().name == "red" && selectiveColor ==
"cyan") {
            range = inColor.getMid().get255Value() -
inColor.getMin().get255Value();
        }

        if (inColor.red.get255Value() > 128 && inColor.red.get255Value()
> 128 && inColor.red.get255Value() > 128
                && selectiveColor == "white") {
            range = 2 * inColor.getMin().get255Value() - 255;
        }

        if (inColor.red.get255Value() < 128 && inColor.red.get255Value()
< 128 && inColor.red.get255Value() < 128
                && selectiveColor == "black") {
            range = 255 - 2 * inColor.getMax().get255Value();
        }

        if (inColor.red.get255Value() != 0 && inColor.red.get255Value()
!= 0 && inColor.red.get255Value() != 0
                && selectiveColor == "gray"
                || inColor.red.get255Value() != 255 &&
inColor.red.get255Value() != 255
                        && inColor.red.get255Value() != 255 &&
selectiveColor == "gray") {
            range = 255
                    - (Math.abs(inColor.getMax().get255Value() - 128)
                            + Math.abs(inColor.getMin().get255Value() -
128));
        }

        return new BlendColor(
```

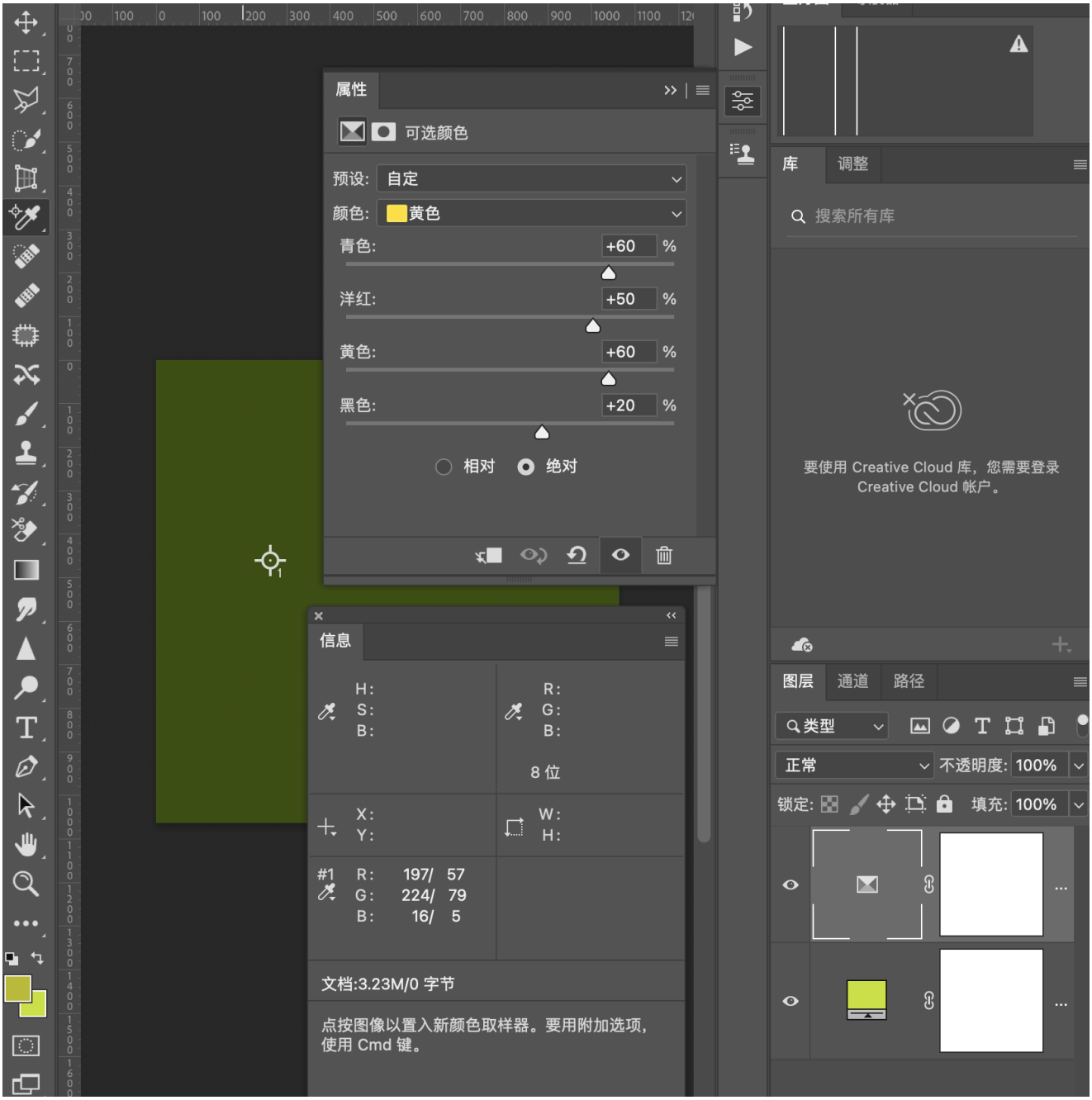
```
                inColor.red.get255Value()
                        +
selectiveColorChannel(inColor.red.get255Value(), range, isAbs,
ratioCyan, ratioBlack),
                inColor.green.get255Value()
                        +
selectiveColorChannel(inColor.green.get255Value(), range, isAbs,
ratioMagenta, ratioBlack),
                inColor.blue.get255Value()
                        +
selectiveColorChannel(inColor.blue.get255Value(), range, isAbs,
ratioYellow, ratioBlack));
    }

    public static double selectiveColorChannel(double channelValue,
double range, boolean isAbs, double arg,
            double karg) {
        double m = isAbs ? 1 : 1 - channelValue / 255;
        return ColorUtils.round(((-1 - arg) * karg - arg) * m, 1 -
channelValue / 255, -channelValue / 255)
                * range;
    }
```

验证结果

```
RGB[ 57.17,  79.20,   4.64]~ HSY[162.27,  74.56,  64.39]~ HSB[162.27,
94.14,  31.06]
```

再看看PS的结果



完美验证

参考文献

http://blog.pkh.me/p/22-understanding-selective-coloring-in-adobe-photoshop.html