Linearization

The first step in color correction is to linearize the detected colors. Because the input color space has not been calibrated, we usually use some empirical methods to linearize. There are several common linearization methods. The first is identical transformation, the second is gamma correction, and the third is polynomial fitting.

Linearization is generally an elementwise function. The mathematical symbols are as follows:

C: any channel of a color, could be R, G or B.

R, G, B: R, G, B channels respectively.

G: grayscale;

s, sl: subscript, which represents the detected data and its linearized value, the former is the input and the latter is the output;

d, dl: subscript, which represents the reference data and its linearized value

Identical Transformation

No change is made during the Identical transformation linearization, usually because the tristimulus values of the input RGB image is already proportional to the luminance. For example, if the input measurement data is in RAW format, the measurement data is already linear, so no linearization is required.

The identity transformation formula is as follows:

$$C_{sl} = C_s$$

Gamma Correction

Gamma correction is a means of performing nonlinearity in RGB space, see the Color Space documentation for details. In the linearization part, the value of γ is usually set to 2.2. You can also customize the value.

The formula for gamma correction linearization is as follows:

$$\begin{matrix} C_{sl} = C_s^{\gamma}, & C_s \geq 0 \\ C_{sl} = -(-C_s)^{\gamma}, & C_s < 0 \end{matrix}$$

Polynomial Fitting

Polynomial fitting uses polynomials to linearize. Provided the polynomial is:

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$$

Then:

$$C_{sl} = f(C_s)$$

In practice, $n \leq 3$ is used to prevent overfitting.

There are many variants of polynomial fitting, the difference lies in the way of generating $f(x)$. It is usually necessary to use linearized reference colors and corresponding detected colors to calculate the polynomial parameters. However, not all colors can participate in the calculation. The saturation detected colors needs to be removed. See the algorithm introduction document for details.

Fitting Channels Respectively

Use three polynomials, $r(x), g(x), b(x)$, to linearize each channel of the RGB color space[1-3]:

$$
\begin{aligned}
R_{sl} &= r(R_s)\\
G_{sl} &= g(G_s)\\
B_{sl} &= b(B_s)
\end{aligned}
$$

The polynomial is generated by minimizing the residual sum of squares between the detected data and the linearized reference data. Take the R-channel as an example:

$$R = \arg min_f(\Sigma(R_{dl} - f(R_S)^2)$$

It's equivalent to finding the least square regression for below equations:

$$
\begin{aligned}
f(R_{s1}) &= R_{dl1}\\
f(R_{s2}) &= R_{dl2}\\
&\ldots
\end{aligned}
$$

With a polynomial, the above equations becomes:

$$
\begin{bmatrix}
R_{s1}^n & R_{s1}^{n-1} & \ldots & 1\\
R_{s2}^n & R_{s2}^{n-1} & \ldots & 1\\
\ldots & \ldots & \ldots & \ldots
\end{bmatrix}
\begin{bmatrix}
a_n\\
a_{n-1}\\
\ldots\\
a_0
\end{bmatrix}
=
\begin{bmatrix}
R_{dl1}\\
R_{dl2}\\
\ldots
\end{bmatrix}
$$

It can be expressed as a system of linear equations:

$$AX = B$$

When the number of reference colors is not less than the degree of the polynomial, the linear system has a least-squares solution:

$$X = (A^TA)^{-1}A^TB$$

Once we get the polynomial coefficients, we can get the polynomial r.

This method of finding polynomial coefficients can be implemented by numpy.polyfit in numpy, expressed here as:

$$R = polyfit(R_S, R_{dl})$$

Note that, in general, the polynomial that we want to obtain is guaranteed to monotonically increase in the interval [0,1] , but this means that nonlinear method is needed to generate the polynomials(see [4] for detail). This would greatly increases the complexity of the program. Considering that the monotonicity does not affect the correct operation of the color correction program, polyfit is still used to implement the program.

Parameters for other channels can also be derived in a similar way.

Grayscale Polynomial Fitting

In this method[2], single polynomial is used for all channels. The polynomial is still a polyfit result from the detected colors to the linear reference colors. However, only the gray of the reference colors can participate in the calculation.

Since the detected colors corresponding to the gray of reference colors is not necessarily gray, it needs to be grayed. Grayscale refers to the Y channel of the XYZ color space. The color space of the detected data is not determined and cannot be converted into the XYZ space. Therefore, the sRGB formula is used to approximate[5].

$$G_s = 0.2126R_s + 0.7152G_s + 0.0722B_s$$

Then the polynomial parameters can be obtained by using the polyfit.

$$f = polyfit(G_s, G_{dl})$$

After f is obtained, linearization can be performed.

Logarithmic Polynomial Fitting

For gamma correction formula, we take the logarithm:

$$ln(C_{sl}) = \gamma ln(C_s), \qquad C_s \ge 0$$

It can be seen that there is a linear relationship between $ln(C_s)$ and $ln(C_{sl})$. It can be considered that the formula is an approximation of a polynomial relationship, that is, there exists a polynomial f, which makes[2]:

$$\begin{gathered} ln(C_{sl}) = f(ln(C_s)), \qquad C_s > 0 \\ C_{sl} = 0, \qquad C_s = 0 \end{gathered}$$

Because $exp(ln(0)) \to \infty$, the channel whose component is 0 is directly mapped to 0 in the formula above.

For fitting channels respectively, we have:

$$\begin{gathered} r = polyfit(ln(R_s), ln(R_{dl})) \\ g = polyfit(ln(G_s), ln(G_{dl})) \\ b = polyfit(ln(B_s), ln(B_{dl})) \end{gathered}$$

Note that the parameter of ln cannot be 0. Therefore, we need to delete the channels whose values are 0 from R_s and R_{dl}, G_s and G_{dl}, B_s and B_{dl}.

Therefore:

$$\begin{gathered} ln(R_{sl}) = r(ln(R_s)), \qquad R_s > 0 \\ R_{sl} = 0, \qquad R_s = 0 \\ ln(G_{sl}) = g(ln(G_s)), \qquad G_s > 0 \\ G_{sl} = 0, \qquad G_s = 0 \\ ln(B_{sl}) = b(ln(B_s)), \qquad B_s > 0 \\ B_{sl} = 0, \qquad B_s = 0 \end{gathered}$$

For grayscale polynomials, there are also:

$$f = polyfit(ln(G_{sl}), ln(G_{dl}))$$

and:

$$\begin{matrix} ln(C_{sl}) = f(ln(C_s)), & \quad C_s > 0 \\ C_s l = 0, & \quad C_s = 0 \end{matrix}$$

References

1. http://www.its.bldrdoc.gov/pub/ntia-rpt/04-406/
2. https://www.imatest.com/docs/colormatrix/
3. http://im.snibgo.com/col2mp.htm
4. https://math.stackexchange.com/questions/60610/polynomial-fitting-where-polynomial-must-be-monotonically-increasing
5. https://en.wikipedia.org/wiki/Grayscale