# Study on the 3D Interpolation Models Used in Color Conversion

Bangyong Sun and Shisheng Zhou

***Abstract*—To find a better model in color conversion, different 3D interpolation models are evaluated, and the reason is analyzed. Firstly, the main principle of 3D look-up-table that interpolation technique based on is introduced; and then the geometric structure and expression of four different interpolation models-trilinear, tetrahedron, pyramid, prism-are analyzed; at last the precision of these four algorithms is tested by calibrating a Toshiba LCD monitor. In the experiment the test shows that all 3D interpolation methods get better calibration result than regression method and BP network; and among the four interpolation methods, prism and pyramid models get smaller average errors and take up less calibration time, while prism and trilinear models have better error distribution than others, thereby the best calibration model is prism interpolation by evaluating synthetically, which indicates that this method may be the firstly-selected of color conversion.**

***Index Terms*—3D interpolation, look-up-table, color conversion.**

## I. Introduction

In the printing or textile industry, the main goal of color reproduction is achieving the vision consistence between original and reproduced products. However, the color devices used in the workflow have different color-mixing models and color gamuts, which makes that goal hard to accomplished[1]. For example, the scanner or monitor uses the additive primary colors RGB, while the printers use subtractive CMYK colors which often have smaller color gamut than RGB devices. What's more, even the same values are sent to two different devices, the resulting colors are usually quite different[2], that's why some color images printed on papers or clothes look different as shown on the monitor screen. To get the consistence between different color devices, a profile connection space(PCS) is introduces by International Color Consortium (ICC)[3]. Using the PCS space, all the devices' color space are converted into PCS values, and then the PCS values are transformed to other devices' color space, thus the two colors may look consistent as long as the precision of color conversion is guaranteed.

Now there are mainly three models used in color conversions, 3D interpolation, polynomial regression and neural network. Because the 3D interpolation method often takes up least time and gets acceptable result, now it is used most widely. Actually there are many models of 3D interpolation method; the aim of this paper is to find the best model in color conversion. In the paper different models of 3D interpolation are analyzed, and in the experiment the precision of them are evaluated from average error, maximum error, times used, and so on. Finally the experiment shows the prism interpolation gets the best result, which indicates that it can be firstly selected as the interpolation modes in color conversion.

Manuscript received October 19, 2011; revised October 31, 2011. This work was supported in part by research foundation of Department of Education of Shanxi province (No11JK0541) and Youth Foundation of Xi'an University of Technology (No.104-211009)F.

Authors are with Xi'an University of Technology, China (email: sunbangyong@xaut.edu.cn).

## II. 3D Interpolation Modes

As both the RGB additive color-mixing mode and the CMY subtractive mode, all have three variables, if the CIE $L^*a^*b^*$ color space is chosen as PCS space, then the color conversion between PCS and RGB(or CMY) can be resolved with 3D interpolation method.

During color conversion, the interpolation is always done on a 3D look-up-table, which contains some matrix of color conversion data for mapping the source color space into the destination color space[4]. For example, the 3D look-up-table during display characterization[5] is shown in Fig.1, with the grids on it that contain two groups of color information-the RGB values and its corresponding $L^*a^*b^*$ values. For a given point $p$ whose position is determined by the RGB values, one can finds eight points around it which constitute a cube. With the cube's vertex data, the $L^*a^*b^*$ value of point $p$ can be computed using the 3D interpolation method.

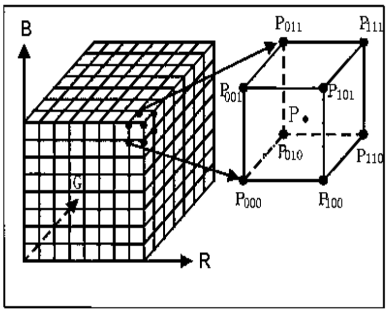


Fig. 1. The framework of 3D look-up-table and its grids

There are many models of 3D interpolation, and the four geometrical method trilinear, prism, pyramid and tetrahedral are used most widely, which are analyzed in the following.

### A. Trilinear Interpolation

The trilinear interpolation [6] is derived from the linear interpolation and 2D bilinear interpolation. As in Fig.1, if the coordinate mechanism is *XYZ*, and the values stored in grids are represented as $f(p)$, if the coordinates of eight vertices on the cube are set as:

$$\begin{aligned}&p_{000}(x_m, y_m, z_m),\ p_{010}(x_m, y_M, z_m),\\ &p_{100}(x_M, y_m, z_m),\ p_{110}(x_M, y_M, z_m)\\ &p_{001}(x_m, y_m, z_M),\ p_{011}(x_m, y_M, z_M),\\ &p_{101}(x_M, y_m, z_M),\ p_{111}(x_M, y_M, z_M)\end{aligned} \tag{1}$$

where m and M is used to describe the *XYZ* values according its position, so it is obviously: $x_m< x_M$; $y_m< y_M$ and $z_M<z_M$ . Thus for a given point $p$ with coordinate $(x,y,z)$, its corresponding function values $f(p)$ can be determined as:

$$\begin{aligned}f(p) &= c_0 + c_1\Delta x + c_2\Delta y + c_3\Delta z + c_4\Delta x\Delta y + c_5\Delta x\Delta z\\ &+ c_6\Delta y\Delta z + c_7\Delta x\Delta y\Delta z\end{aligned} \tag{2}$$

where $\Delta x, \Delta y, \Delta z$ represent relative distance between $p$ and $p_{000}$ in the $x$, $y$ and $z$ directions.

$$\begin{aligned}\Delta x &= (x - x_m)/(x_M - x_m);\\ \Delta y &= (y - y_m)/(y_M - y_m);\\ \Delta z &= (z - z_m)/(z_M - z_m)\end{aligned} \tag{3}$$

while $c_0 \sim c_7$ are the correlative coefficients:

$$\begin{aligned}c_0 &= f(p_{000});\\ c_1 &= f(p_{100}) - f(p_{000});\\ c_2 &= f(p_{010}) - f(p_{000});\\ c_3 &= f(p_{001}) - f(p_{000});\\ c_4 &= f(p_{110}) - f(p_{010}) - f(p_{100}) + f(p_{000});\\ c_5 &= f(p_{101}) - f(p_{001}) - f(p_{100}) + f(p_{000});\\ c_6 &= f(p_{011}) - f(p_{001}) - f(p_{010}) + f(p_{000});\\ c_7 &= f(p_{111}) - f(p_{011}) - f(p_{101}) - f(p_{110})\\ &+ f(p_{100}) + f(p_{001}) + f(p_{010}) - f(p_{000})\end{aligned} \tag{4}$$

With this method, all the eight vertices' information are used to compute $f(p)$, so there is no need for a search mechanism to find which vertices are selected.

## B. Tetrahedral Interpolation

The tetrahedral interpolation[7] slices a cube into six tetrahedrons, and each tetrahedron has four flat triangle surfaces. Actually there are many ways to divide a cube into tetrahedrons, one of the way used in this paper is shown in Fig.2.

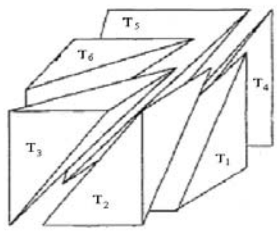


Fig. 2. The cube divided into 6 tetrahedrons

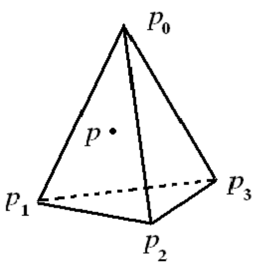


Fig. 3. The tetrahedral interpolation

The point $p$ will be in one of the six tetrahedrons, and the number of the tetrahedron can be determined by the condition as below:

$$\begin{aligned}&T1: \Delta x > \Delta y > \Delta z \quad T2: \Delta x > \Delta z > \Delta y \quad T3: \Delta z > \Delta x > \Delta y\\ &T4: \Delta y > \Delta x > \Delta z \quad T5: \Delta y > \Delta z > \Delta x \quad T6: \Delta z > \Delta y > \Delta x\end{aligned} \tag{5}$$

As in Fig.3, if point $p$ is inside the tetrahedron which is consist of four vertices $p_0$, $p_1$, $p_2$, $p_3$, then the value of $p$ can be expressed as:

$$\begin{aligned}&f(p) = f(p_0)*(V_0/V) + f(p_1)*(V_1/V) + f(p_2)*(V_2/V) +\\ &f(p_3)*(V_3/V)\end{aligned} \tag{6}$$

where $V$ represents the volume of the tetrahedron $p_0p_1p_2p_3$, and $V_0$ is the volume of the sub-tetrahedron $pp_1p_2p_3$, $V_1$ is the volume of the sub-tetrahedron $pp_0p_2p_3$, $V_2$ is the volume of the sub-tetrahedron $pp_0p_1p_3$, $V_3$ is the volume of the sub-tetrahedron $pp_0p_1p_2$, obviously we find that $V= V_0+ V_1+ V_2+ V_3$

## C. Prism Interpolation

The prism interpolation[8] in Fig.4 is a six-point prismatic extraction from the cubical packing, and it is a symmetrical structure. As defined in equation.3, for point $p$ inside the cube, if $\Delta x > \Delta y$, it is contained by *Prism*1, and the value of $f(p)$ can be expressed as:

$$\begin{aligned}f(p) = f(p_{000}) &+ (f(p_{100}) - f(p_{000}))\Delta x + (f(p_{110}) - f(p_{100}))\Delta y\\ &+ (f(p_{001}) - f(p_{000}))\Delta z + (f(p_{101}) - f(p_{001}) - f(p_{100}) +\\ &f(p_{000}))\Delta x\Delta z + (f(p_{111}) - f(p_{101}) - f(p_{110}) + f(p_{100}))\Delta y\Delta z\end{aligned} \tag{7}$$

Otherwise, the point $p$ is inside of *Prism*2, then the function $f(p)$ is:

$$\begin{aligned}f(p) = f(p_{000}) &+ (f(p_{110}) - f(p_{010}))\Delta x + (f(p_{010}) - f(p_{000}))\Delta y\\ &+ (f(p_{001}) - f(p_{000}))\Delta z + (f(p_{111}) - f(p_{011}) - f(p_{110}) +\\ &f(p_{010}))\Delta x\Delta z + (f(p_{011}) - f(p_{001}) - f(p_{010}) + f(p_{000}))\Delta y\Delta z\end{aligned} \tag{8}$$

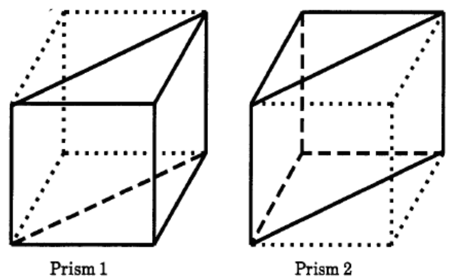


Fig. 4. The cube is divided into 2 prisms

## D. Pyramid Interpolation

As shown in Fig.5, a cube can be divided into 3 pyramids, and the function $f(p)$ can be determined similar to the prism interpolation. Firstly the number of prism which contains the point $p$ should be determined, and the next task is to compute the value of $f(p)$. Kang[9] lists the conditions and coefficients of pyramid interpolation:

| *pyramid* | test | $c_1$ | $c_2$ | $c_3$ | $c_4$ |
|---|---|---|---|---|---|
| 1 | $\Delta y > \Delta x, \Delta z > \Delta x$ | $p_{111} - p_{011}$ | $p_{010} - p_{000}$ | $p_{001} - p_{000}$ | $p_{011} - p_{001} - p_{010} + p_{000}$ |
| 2 | $\Delta x > \Delta y, \Delta z > \Delta y$ | $p_{100} - p_{000}$ | $p_{111} - p_{101}$ | $p_{001} - p_{000}$ | $p_{101} - p_{001} - p_{100} + p_{000}$ |
| 3 | $\Delta x > \Delta z, \Delta y > \Delta z$ | $p_{100} - p_{000}$ | $p_{010} - p_{000}$ | $p_{111} - p_{110}$ | $p_{110} - p_{100} - p_{010} + p_{000}$ |

At last, the value $f(p)$ will be determined accordingly as below.

$$\begin{aligned}
&\text{Pyramid1: } f_1(p) = f(p_{000}) + c_1\Delta x + c_2\Delta y + c_3\Delta z + c_4\Delta y\Delta z \\
&\text{Pyramid2: } f_2(p) = f(p_{000}) + c_1\Delta x + c_2\Delta y + c_3\Delta z + c_4\Delta x\Delta z \\
&\text{Pyramid3: } f_3(p) = f(p_{000}) + c_1\Delta x + c_2\Delta y + c_3\Delta z + c_4\Delta x\Delta y
\end{aligned} \tag{9}$$

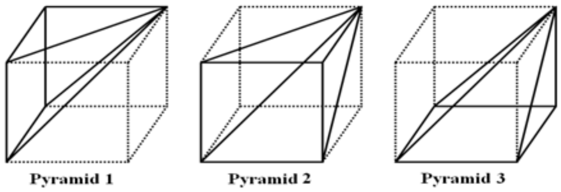


Fig. 5. The cube is divided into 3 pyramids

## III. Experiment

In the experiment, a TOSHIBA M5 LCD monitor is selected as the calibration object, and the color conversion from RGB to $L^*a^*b^*$ is done using the different interpolation models above. Firstly, the full RGB channels are divided into 1000 patches which are displayed on the monitor, and then the corresponding $L^*a^*b^*$ values are recorded using the spectrophotometer X-Rite DTP94 at D50/2$^o$, while all these color values will be used as modeling data to construct the 3D look-up-table. In addition, another 343 groups color values are collected as the testing data, while the CIE$\Delta E_{ab}^*$ color difference formula [10-12] is chosen as evaluation function, which is expressed as:

$$\Delta E = \sqrt{(L_1^* - L_2^*)^2 + (a_1^* - a_2^*)^2 + (b_1^* - b_2^*)^2} \tag{10}$$

where $L_1^*a_1^*b_1^*$ is the measure $L^*a^*b^*$ value, and $L_2^*a_2^*b_2^*$ is the computed $L^*a^*b^*$ value.

To compare the calibration result of different method, not only the above four 3D interpolation models but also the polynomial regression method and BP neural network are tested in the experiment. The selected polynomials consist of 20 coefficients, and the highest order of polynomial is 3$^{rd}$, which is expressed as:

$$\begin{aligned}
&p_3(x,y,z) = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 z + \alpha_4 xy + \alpha_5 yz + \alpha_6 zx + \\
&\alpha_7 xyz + \alpha_8 x^2 + \alpha_9 y^2 + \alpha_{10} z^2 + \alpha_{11} x^2 y + \alpha_{12} x^2 z + \alpha_{13} y^2 x + \\
&\alpha_{14} y^2 z + \alpha_{15} z^2 x + \alpha_{16} z^2 y + \alpha_{17} x^3 + \alpha_{18} y^3 + \alpha_{19} z^3
\end{aligned} \tag{11}$$

While for the constructed BP network, the input/output layer all have three variables, and the hidden layer have 16 nerve units, at last the simulation degree is set as 100 times.

In the experiment, the evaluation parameters comprise four items: average error, maximum error, 95% ascending-sort error, and the calibration time used. Among them, average error is the most import for judging the result of algorithm; and the calibration time used may indicates whether the algorithm will be used widely; while the other two items explain the distribution of errors. Finally, the experiment result is shown in TABLE I.

As shown in table.1, it is clearly that the calibration result of the four interpolation methods is better than regression method and BP network, as the latter two algorithms get bigger errors and take up more time. For the four 3D interpolation models, the prism and pyramid algorithm have smaller average errors and use less time, while the prim and trilinear models have better error-distribution, at last by evaluating synthetically the prism interpolation is defined as the best calibration model for color conversion.

TABLE I: The Errors of Different Methods Used in Color Characterization

| Algorithm | Average(error) | Maximum(error) | 95%error | Time used(s) |
|---|---|---|---|---|
| BP network | 1.9001 | 12.2824 | 3.6353 | 83.2030 |
| Regression | 1.6410 | 6.5535 | 3.0549 | 0.5000 |
| Trilinear | 1.2182 | 6.2040 | 1.8727 | 0.3440 |
| Tetrahedron | 1.2045 | 10.6243 | 2.4860 | 0.4850 |
| Pyramid | 1.2002 | 10.1702 | 2.4303 | 0.2810 |
| Prism | 1.1547 | 9.1992 | 1.9259 | 0.2820 |

## IV. Conclusions

Now the three most commonly used models for color conversion are 3D interpolation, polynomial regression and neural network. In the paper, the principle of these methods is analyzed and tested, while from the experiment, the conclusions about computing mechanism, time-using, precision and sample data can be made below:

(1) When the regression and neural network method are used in color conversion, the coefficients among $L^*a^*b^*$ and RGB is computed by using the sample data, and any unknown color values are determined by these coefficients. But there are always some measuring or recording errors in the sample data, and they also bring in errors for the computed coefficients, then all the converted color values are influence by these coefficients. While for the 3D interpolation method, the converted color values are determined by the eight points in cube, so all wrong points outside of the cube will not bring errors here, they can only influence a little proportion colors surrounds the according cubes.

(2) As a whole, the time of 3D interpolation used in color conversion is less than other two methods. The reason is because most of the time is spent for searching cubes in 3D interpolation, while the searching process often use little time with the development of storage-and-finding technology. But for the other two methods, most time is taken up during computing coefficients. Especially for the neural network, as the nonlinear feature between RGB and $L^*a^*b^*$ color space, the algorithm must be executed many times to ensure the precision, thus much time are used. While for the four kinds of 3D interpolation methods, tetrahedron interpolation uses more time than the others, because the corresponding cube are sliced into 6 parts which is complex than other slicing method, thus the trilinear, pyramid and prism methods use about the same time with the simple slicing structure in cube.

(3) From the experiment result, prim and trilinear models get the better error distribution and higher precision. By analyzing the interpolation expression, we find the reason is their expression all have more terms of polynomials with high order. Take the trilinear modes for example, if the terms of $\triangle x\triangle y$, $\triangle x\triangle z$, $\triangle y\triangle z$ and $\triangle x\triangle y\triangle z$ are ignored, the average error will rises 24.9% from 1.2182 to 1.5215. However, although prim and trilinear models have complicate expressions, the time used doesn't increase

clearly.

(4) One condition for all color conversion methods must be fulfilled: the sampling data should distribute around all of the device spaces. For the interpolation method, the sampling data should have a regular structure, and more data will improve the precision. While for the other methods, more redundant data sometime bring in errors during calculating coefficients.

From the analysis above, we can find the reason of why the 3D interpolation method is used most widely in color management and image processing. In this paper, the regression method and neural network method are compared with the interpolation method, and we find the interpolation method is precise and time-saving. What’s more, in the test of four kinds of interpolation models, the prism interpolation has the best performance which indicates that we can choose this model during color conversion.

## ACKNOWLEDGMENT

This work is supported by "13115"creative foundation of science and technology, Shanxi province of china (No2009ZDGC-06).

## REFERENCES

[1] R H Liu, P Zeng, and Y F Wang. “A color calibration method of printer using adaptive regression,”.*Acta Electronica Sinica*, 2007, vol. 35, no.11, pp. 2201-2204.

[2] J C King. “Why color management,” in *the 9th Congress of the International Colour Association, Rochester*, 2001, pp.439-445

[3] E R Ding, P Zeng, and H Liu, et al. “Color calibration based on boosting kernel partial least squares regression,” in *Chinese Journal of Scientific Instrument*, 2008, vol.29, no.1, pp.79-84.

[4] K D Gennetten. “RGB to CMYK conversion using 3D barycentric interpolation,” in *proc. Device-Independent Color Imaging and Imaging Systems Integration, San Jose*, 1993, pp.116-126 .

[5] J B Thomas, J Y Hardeberg, and I Foucherot et al. “The PLVC Display Color Characterization Model Revisited.” *Color Research and Application*, 2008, vol.33, no.6, pp.449-460

[6] S I Nin, J M Kasson, and W Plouffe. “Printing CIELAB images on a cmyk printer using trilinear interpolation,” *Color Hard-copy and Graphics Arts, San Jose*, 1992, pp.316-324.

[7] J M Kasson, W Plouffe, and S I Nin. “Tetrahedral interpolation technique for color space conversion.” Device-Independent Color Imaging and Imaging Systems Integration, San Jose, 1993:126-138.

[8] K Kanamori and T Fumoto, H Kotera. “A color transformation algorithm using prism interpolation,” in *ISandT 8th International Congress on Advances in Non-Impact Printing Technologies. Virginia*, 1992, pp.477-482・

[9] H R Kang. “Color technology for electronic imaging devices.” Washington: SPIE Optical Engineering Press, 1997: 55-63

[10] R G KUEHNI. “Color difference formulas: an unsatisfactory state of affairs.” *Color Research and Application*, 2008, 33(4):324-326

[11] M R Luo, G Cui, and B Rigg. “The development of the CIE 2000 colour-difference formulae: CIEDE2000.” *Color Research and Application*, 2001, 26(5): 340-350

[12] N Sándor, T Ondró, and J Schanda. “Spectral interpolation errors.” *Color Research and Application*, 2005, 30(5): 348-353.

**Bangyong Sun** was born in Shan Dong Province, China in 1980. He is currently a Ph.D. candidate in mechanical engineering at Xian University of Technology. And now he is a lecturer and a research fellow at the faculty of printing and packing in Xian University of Technology. He is doing some researches on color management such as conversion algorithms and gamut mapping algorithms.

**Shisheng Zhou** was born in Shan Dong Province, China in 1963 and he is a Professor and Ph.D. Supervisor with the Xi’an University of Technology. And He is now the dean of the Faculty of Printing and Packaging Engineering, Xi’an University of Technology. He is a expert in the field of color science.