Reducing the Cost of Lookup Table Based Color Transformations

Raja Balasubramanian
Xerox Digital Imaging Technology Center
Webster, New York

Abstract

Color transformations in digital imaging systems are often implemented with lookup tables (LUTs) that require some form of multidimensional interpolation. Such LUT based transformations typically involve a trade-off between the computational cost, required storage and/or memory, and the resulting accuracy of the transform. In this paper, novel methods are proposed for improving some of these quality-cost trade-offs. The methods fall in two categories: i) those that improve the trade-off between computational cost and quality; and ii) those that enhance the trade-off between LUT size and quality. Results show that promising trade-offs can be achieved by exploiting the properties of the human visual system, as well as the characteristics of the function being approximated by the LUT.

Introduction

An important goal of a color management system is to be able to achieve consistent color output across a wide variety of devices in an imaging system. This necessitates the use of color transformations that capture the color characteristics of the devices in the system, as well as properties of the human visual system (HVS). The derivation of such transforms is an important focus of this conference, and will be addressed by other authors.

This paper deals with the efficient implementation of these transforms. Often, the color transforms are complex multidimensional functions that make the real-time processing of image data a computationally prohibitive task. To reduce the computational cost, the functions are typically implemented as multidimensional lookup tables (LUTs). A LUT is essentially a rectangular grid that spans the input color space of the transform. Figure 1 shows an example of a 3-D LUT that maps L*a*b* input to printer CMYK. Output values corresponding to each of the grid intersections, or nodes, are pre-computed and stored in the LUT. Input colors are processed through the LUT by i) finding the cube to which the input color belongs, and ii) performing an interpolation among a subset of the 8 cube vertices to compute the output color value. Many interpolation schemes exist. In 3 dimensions, these include trilinear, prism, pyramid, and tetrahedral interpolation.[1] Of these, tetrahedral interpolation requires the fewest computations, and all three require comparable storage and memory. The cost required for these schemes can be prohibitive for some applications, particularly in the low volume market. In this paper, methods are proposed to reduce the cost with acceptable loss in quality.

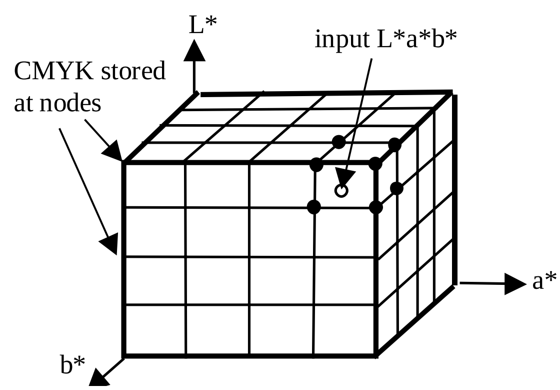


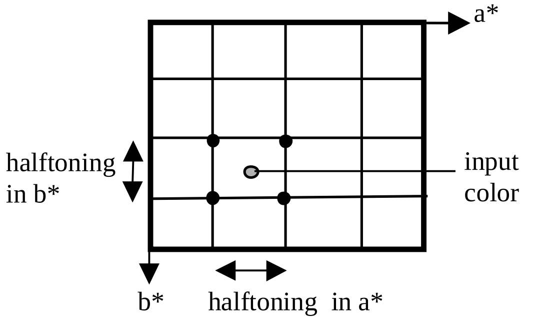
*Figure 1. Example of 3-D LUT interpolation from L*a*b* to CMYK.*

Three algorithms will be described in this paper. The first one reduces the computational cost of the LUT interpolation, while the remaining two offer savings in LUT size and memory requirements. The idea in all three cases is to keep the loss in LUT accuracy at a visually acceptable level.

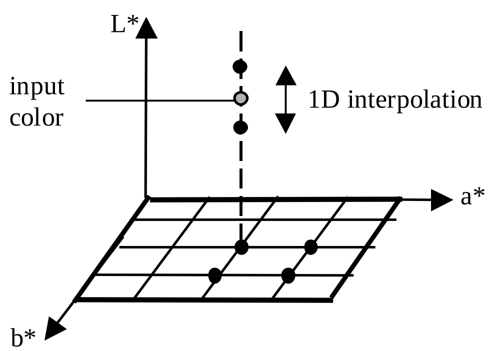
Reduction in computational cost via chrominance halftoning

Basic idea

It is well known that the HVS is less sensitive to errors in chrominance than errors in luminance at high spatial frequencies[2]. This fact may be utilized to our advantage in the case where the input to the LUT is a luminance-chrominance color space. Consider again the example of a printer color correction LUT from L*a*b* to CMYK. The aforementioned property of the HVS implies that it is possible to introduce high frequency distortions in the a* and b* channels in a manner that is not objectionable to a human observer. The type of distortion we choose for our purpose is multilevel halftoning[3] along each of the a* and b* axes. The cost savings is achieved by choosing the halftone levels to coincide with the LUT node locations along a* and b*, as shown in Figure 2a. After the chrominance halftoning step, the input color maps to one of the 4 neighboring LUT nodes in the a*-b* plane (denoted by black circles in Fig 2a). Calculation of the output CMYK now requires only a 1-D interpolation along the L* dimension, as shown in Fig 2b.



(a)



(b)

Figure 2. (a) Multilevel halftoning along a and b* followed by (b) 1-D interpolation along L**

The chrominance halftoning may be performed inexpensively using multilevel screening, wherein each of the a* and b* input values are compared to a threshold in a periodic halftone screen, and assigned to one of two neighboring node values. For our experiment, an 8 x 8 Bayer dot[3] was used, as this effectively contains the halftone signature within high spatial frequencies. Other screens, *e.g.* stochastic screens can also be used.

Experimental Results

Table 1 compares the computational complexity of the proposed approach with standard trilinear and tetrahedral interpolation. The latter is the fastest known 3-D interpolation algorithm. The table shows that the cost savings are significant. Timing tests on a Sun SPARC 20 workstation showed that the time taken to convert an image from L*a*b* to CMYK with a 3-D LUT using the proposed method was approximately half of the time required by tetrahedral interpolation.

Table 1. Cost analysis for trilinear, tetrahedral, and proposed schemes for N output signals. M, A, C, and S denote multiplications, additions, comparisons, and shift operations, respectively.

	M	A	C	S
1. Trilinear	7N	7N+2	0	2
2. Tetrahedral	3N	3N+2	2.5	2
3. Proposed method	N	N+4	0	2
4. % Savings from (2) to (3)	67	43	100	0

The quality of the proposed technique is best evaluated by comparing printed samples generated by this method with a standard method such as tetrahedral interpolation, and observing how much degradation is brought about by the chrominance halftoning. To augment the visual evaluation, a quantitative error metric was sought that would predict the visual difference between the standard and the new interpolation methods. Since the new approach introduces a distortion that is dependent on spatial frequency, it is meaningful to use a color difference metric that is also sensitive to spatial frequency. To this end, the sCIELAB model[4] was used as a visual difference metric.

The experiment was performed as follows. A 3-D LUT from L*a*b* to CMYK was constructed for a Xerox 5795 laser printer. The LUT comprised 17 uniformly spaced nodes along each of the 3 axes. The input L*a*b* image was transformed to printer CMYK with this 3-D LUT, using both the new technique and tetrahedral interpolation. The two CMYK images were then converted back to CIELAB using a printer model, and a visual difference image computed using the sCIELAB model. The mean and 95th percentile values from the difference image are reported in Table 2 for 4 images of natural scenes. In interpreting this data, it is useful to remember that for an image consisting of a single color (*i.e.* an image containing only the zero frequency component), the sCIELAB error reduces to the CIE 1994 color difference metric.[5] As a point of reference, the errors shown in Table 2 are often within page-to-page and day-to-day print variations, which lie between 3 and 5 ΔE units for many printers. Furthermore, for binary printers, the high frequency errors introduced by the multilevel halftoning are often masked by the binary halftoning applied to the C, M, Y, K signals. This masking effect is not accounted for in Table 2. Finally, if LUT size is not a critical issue, a denser node sampling along a* and b* will certainly reduce the visibility of the chrominance halftoning. Hence we believe that the quality achieved by the proposed technique is acceptable for many applications.

Table 2 sCIELAB error introduced by chrominance halftoning technique

Images	sCIELAB error	
	Average	95th Percentile
Image 1	2.59	4.4
Image 2	3.38	6.0
Image 3	3.48	6.4
Image 4	4.0	8.1

Efficient node sampling using output values

The next technique samples the LUT nodes efficiently according to the characteristics of the color transformation being approximated.

Basic Idea

The idea is based on the observation that the actual interpolation calculation from an N-dimensional input space to an M-dimensional output space is really a series of M independent interpolations from N-dimensional input to 1-dimensional output. Consider for example, a LUT transform from monitor RGB to printer CMY. The output cyan value is calculated by interpolating the cyan values at the surrounding nodes. Similarly, the magenta and yellow outputs are independently calculated by interpolating the nearby magenta and yellow node values, respectively. This being the case, there is no reason to use the same grid to interpolate the C, M, and Y. For example, one might expect that C has a strong dependence on its complementary input R, and a weak dependence on G and B. Likewise, M may show stronger dependence on input G than on R and B. This has indeed been observed with several printer characterization transforms. Hence, a suitable grid for interpolating C would have a dense sampling along R and coarse sampling along G and B. Equivalently, the M signal requires a finer sampling along G than along R and B. The idea is demonstrated in Fig 3 for a 2-D transform from R-G to C-M. Note that the operation of finding the enclosing sub-cube needs to now be repeated for each output separation. This is a relatively small computational overhead.

Experimental Results

An experiment was conducted with a transformation from sRGB to CMY for a Tektronix Phaser 440 dye sublimation printer. A standard 3-D LUT consisting of a uniformly spaced grid of 16x16x16 nodes was used as the reference

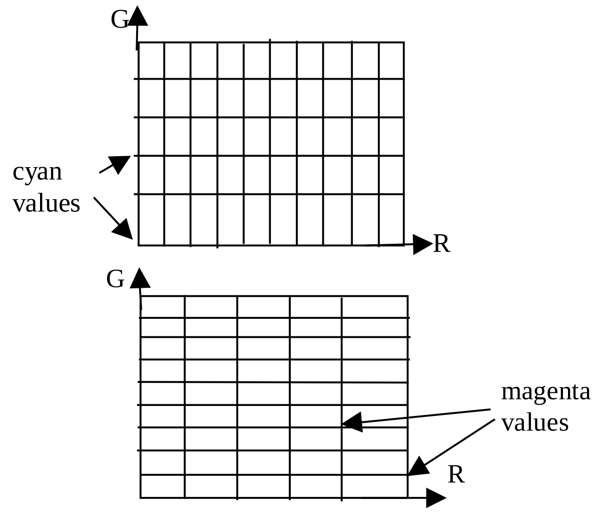


Fig 3 Different grids for interpolating cyan and magenta output signals

transform. The new LUT structure used 3 different grids of size 16x8x8, where for a given output signal, the finer node sampling occurred along the complementary color axis (as described above). Each axis was uniformly sampled. A set of 1000 sRGB values distributed within the printer gamut was used as an independent test set. This data was transformed to CMY using both the reference and new LUTs, then converted to L*a*b* using a printer model. The CIE 1994 ΔE metric was used to calculate the error between the new and the reference transforms. The average and 95th percentile errors were 1.05 and 3.82 respectively. Alternatively, if the same 16x8x8 grid structure was used for all output signals, the best average and 95th percentile errors were 2.1 and 6.2 respectively. Clearly, the variable grid structure improves the trade-off between LUT size and accuracy.

Efficient node sampling using input values

The following method allows the LUT grid structure to change based on the values of the input signals. This contrasts with the method just described, which changes the grid structure based on the output signal.

Basic Idea

To describe the concept, it is desirable to think of the 3-D grid in Fig 1 as a family of identical 2-D grids, two of which are shown in Fig 4. One form of 3-D interpolation is achieved by i) projecting the R and G values of the input RGB point on each of the two neighboring 2-D planes B_j and B_{j+1}, ii) performing 2-D interpolation among vertices V_1-V_4 within the first plane, and among vertices V_5-V_8 within the second plane, and iii) using the third input coordinate B_{in} to perform 1-D interpolation on the 2 intermediate outputs from ii). This is illustrated in Fig 4. When seen from this perspective, it is readily seen that the 2-D grids in Fig 4 do not necessarily have to all be identical. If, for example, we know that the output signal variation decreases as input B increases, then we could design R-G grids that become coarser as B increases. This is shown in Fig 5.

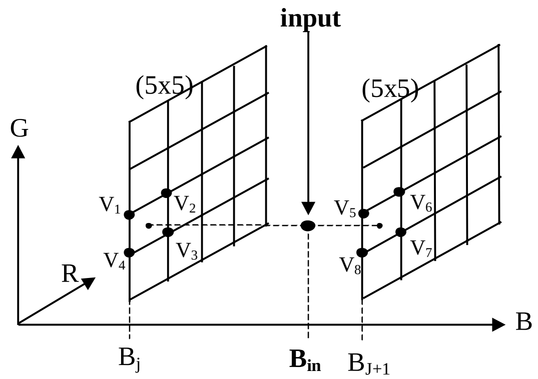


Figure 4. Standard 3D LUT structure

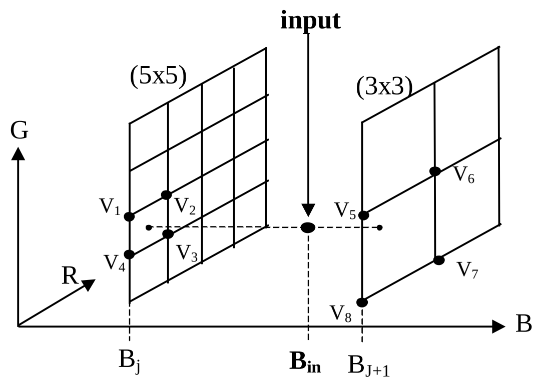


Figure 5. Proposed LUT structure where grid sizes changes as a function of the B coordinate.

This technique can be used to significantly reduce the LUT size, especially when the input dimension is greater than 3. An application where this approach is particularly useful is in implementing printer characterization transformations from CMYK to L*a*b*. Let us think about the CMYK hypercube as a series of CMY cubes at different levels of K. We know that as K increases, the overall variation in L*a*b* decreases as a function of CMY. At the extreme, when K =100%, there is usually very little change in L*a*b* values as a function of CMY. Hence, one can use the proposed grid structure to reduce the resolution of the CMY grid as K increases.

Note that this approach splits an N-dimensional input space into an N-1 dimensional and a 1-dimensional subspace. The same idea can be used again to decompose the N-1 dimensional subspace into subspaces of dimension N-2 and 1. It is interesting to note that N-1 recursive applications of this approach will result in sequential linear interpolation[6], where all the interpolations are performed in 1-D subspaces.

Experimental Results

The proposed technique was used to implement a transform from CMYK to L*a*b* for a Xerox 5765 laser printer. A 4-D LUT using a 5x5x5x5 uniformly spaced grid was used as the reference LUT. To test the new approach, different 3-D CMY grids were used for four levels of K = 0%, 33%, 66%, and 100%, where the CMY grid sizes were 5^3, 4^3, 3^3, and 2^3, respectively. 1000 CMYK samples distributed throughout the printer gamut were used as a test set. This data was converted to L*a*b* using both the reference and the new methods, and the differences computed using the CIE 94 ΔE metric. The average and 95th percentile errors were 1.8 and 6.25, respectively. Given the dramatic savings in LUT size from 625 to 224 nodes, this error is acceptable for many applications.

Conclusion

Three techniques have been proposed to reduce the cost of LUT transforms while minimizing the loss in accuracy. The first uses chrominance halftoning to reduce the 3-D interpolation problem to 1-D interpolation. The technique introduces high frequency chrominance errors that are not easily perceived by the human visual system. The last two approaches exploit characteristics of the particular function being approximated to generate efficient node sampling schemes that enable the overall LUT size to be reduced with minimal loss in accuracy. All 3 techniques show considerable promise in improving quality-cost trade-offs.

References

1. H. R. Kang, *Color Technology for Electronic Imaging Devices*, SPIE, Bellingham, WA., 1997, Chapter 4.
2. P. K. Kaiser and R. M. Boynton, *Human Color Vision*, OSA, Washington, DC., 1996, pp. 399-409.
3. R. Loce, P. Roetling, Y-w. Lin, *Electronic Imaging Technology*, SPIE, Bellingham, WA., 1999, Chapter 7.
4. X. M. Zhang and B. A. Wandell, A spatial extension to CIELAB for digital color image reproduction, *Proc. SID Symposium.* (1996).
5. CIE, Industrial Color-Difference Evaluation, Technical Report CIE 116-1995.
6. J. P. Allebach, James Chang, and Charles Bouman, Efficient implementation of nonlinear color transformations, *IS&T and SID's 1st Color Imaging Conference*, pp. 143-148. (1993).