

# A NEW COLOR TRANSFORM FOR RGB CODING

Hyun Mun Kim, Woo-Shik Kim, and Dae-Sung Cho

Samsung Advanced Institute of Technology, Multimedia Lab,  
P.O. Box 111, Suwon 440-600, Korea

## Abstract

*This paper presents a new color transform (YSbSr) that increases the coding efficiency for RGB space by reducing the conversion error and coding error propagation that happen during the color space conversion. Traditionally the use of color transforms has been limited to maintain the compatibility with black-and-white transmission. But for high quality video for professional applications, the focus should change accordingly. Recently during the course of development for Professional Extensions of the MPEG-4 AVC H.264 standard, new color transforms have been proposed. These transforms are focusing on the decorrelation and reversibility using lifting scheme for integer implementation. But in real coding scenario, lossless coding is rare and coding error is inevitable. So we have to deal with the propagation of coding error due to the backward conversion to RGB space. Even though integer mapping is a charming factor, it is more desirable not to employ integer mapping if we lose the original goal, high decorrelation gain with small conversion error. Since the conversion equation takes up very small portion of coder complexity, the difference between integer mappings and floating point operation is diluted.*

*Based on this observation, we propose a new color transform that gives: high decorrelation gain with small rounding and conversion error. The simulation results show that we can achieve almost up to 2 dB gain compared with the traditional YCbCr color space using the proposed transform. It makes the proposed color transform a good candidate for applications that need to preserve the original color fidelity.*

## I. INTRODUCTION

Recently many new color transforms have been proposed including ORCT, RDCT and SHIRCT [1]. Also during the course of development for Professional Extensions of the MPEG-4 AVC/H.264 standard [2], new color transforms have been proposed. They include YCoCg-R [3] and YFbFr [4]. Both transforms employ integer mapping and guarantee the reversibility using a lifting scheme [5]. These transforms are claiming high coding gain and hardware

friendly implementation using integer mapping. But color transform involves the color space conversion error and the coding error propagation through the backward conversion. So far, none of them has addressed these errors.

Having mentioned this, this paper presents a new color transform (YSbSr) that increases the coding efficiency for RGB space by reducing the conversion error and coding error propagation. It starts with the well-known KL transform and approximates it to ensure the similar structure with the existing color transforms. By analyzing the relationship between the conversion formula and the conversion error that depends on the conversion coefficients, we decrease the error greatly by increasing the dynamic range of the color-converted space.

Using this new color transform, we achieved almost up to 2 dB gain compared with the traditional YCbCr color space. It also gives the better coding efficiency than the other color transforms that claim high coding gain theoretically. This dynamic range expansion concept can also be applied to any other color space easily to achieve the better coding efficiency.

This paper is organized as follows. In Section II, we show how color space conversion creates color distortion depending on conversion coefficients and describe the proposed color transform in detail. The test results are summarized in Section III. We conclude in Section IV.

## II. A NEW COLOR TRANSFORM

### A. Color Distortion Due to Color Space Conversion

In this section, we analyze the color space conversion error and show that there exists an achievable PSNR limit due to this rounding error [6]. This analysis assumes that we use 8 bit fixed-point RGB data ranging from 0 to 255 and use the same number of bits of precision to map the RGB data to another color space and to come back to the RGB space. We can generalize this analysis for any N-bit data.

The RGB to YCbCr conversion using the Rec. BT. 601 is defined as follows:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ -0.1146 & -0.3854 & 0.5 \\ 0.5 & -0.4542 & -0.0458 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

The rounding operations here introduce 1/12 rounding error due to the uniform rounding quantization error. The YCbCr to RGB conversion is performed as follows:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.5748 \\ 1.0 & -0.1873 & -0.4681 \\ 1.0 & 1.8556 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \quad (2)$$

The rounding operations here introduce  $\frac{1}{12}a^2$  rounding error from each component (Y, Cb, Cr) where  $a$  is the coefficient of the color conversion matrix in (2). It is due to the inherent rounding error from RGB to YCbCr conversion. For example we have the following rounding error for the G component:

$$Error = \frac{1}{12}(1^2 + 0.1873^2 + 0.4681^2)$$

So the overall conversion error for the G component,  $E_G$ , is:

$$E_G = \frac{1}{12}(1^2 + 1^2 + 0.1873^2 + 0.4681^2) = 0.1878.$$

We can calculate the conversion error for the other components using the same method.

$$E_R = \frac{1}{12}(1^2 + 1^2 + 0^2 + 1.5748^2) = 0.3733,$$

$$E_B = \frac{1}{12}(1^2 + 1^2 + 1.8556^2 + 0^2) = 0.4536.$$

Since we use 8 bits for each component, the achievable PSNR value for each component is

$$PSNR_G = 10 \cdot \log \frac{255^2}{E_G} = 55.4db$$

$$PSNR_R = 10 \cdot \log \frac{255^2}{E_R} = 52.4db$$

$$PSNR_B = 10 \cdot \log \frac{255^2}{E_B} = 51.6db$$

**Table 1: Achievable PSNR values when color space conversion (RGB→YCbCr→RGB) is performed using ITU-R Rec. BT. 601 for 8-bit images.**

|        | Theoretical analysis | Experimental Results |
|--------|----------------------|----------------------|
| PSNR-G | 55.4 dB              | 57.2 dB              |
| PSNR-R | 52.4 dB              | 52.2 dB              |
| PSNR-B | 51.6 dB              | 51.5 dB              |

According to this analysis the backward conversion coefficients are more important to the PSNR since the square sum of each row in (2) is greater than in (1). We also confirmed these PSNR values by conducting an experiment doing RGB to YCbCr and back to RGB conversion and measuring the PSNR values. For this experiment, we assumed the same probability for all possible color combinations from 0 to 255. So there are  $256^3$  equi-probable cases in this experiment. Table 1 summarizes this theoretical analysis along with the experimental results. It confirms the validity of the theoretical analysis.

## B. A New Color Transform YSbSr

In the previous section, we analyzed how the color conversion introduces achievable PSNR limit. In this section, we introduce a new color transform that generates small conversion and propagation error.

Recently, Malvar and Sullivan have proposed a new color transform, YCoCg-R, as defined below [3].

$$\begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1 & 0 & -1 \\ -1/2 & 1 & -1/2 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3)$$

Also Topiwala and Tu have proposed a new color transform, YFbFr, as defined below [4].

$$\begin{bmatrix} Y \\ Fb \\ Fr \end{bmatrix} = \begin{bmatrix} 5/16 & 3/8 & 5/16 \\ -1/2 & 1 & -1/2 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4)$$

Looking at the coefficients of (3) and (4), we can easily see that they are very similar transforms. Both transforms are derived using KL transform. After KL transform, they are approximated using the dyadic coefficients to guarantee the integer reversibility without considering the error propagation. They guarantee the lossless conversion using the lifting scheme and need one extra bit depth expansion for the chroma components due to the increased dynamic range.

But this is based on the assumption that there is no coding error involved. The lossless conversion is meaningless when coding noise is involved. The error propagation is more important factor to consider. If coding is involved, the coding error in the transformed space is propagated through the backward conversion to the RGB space. It is proportional to the sum of the square of each backward conversion coefficient as the rounding error analyzed in the previous section.

To cope with these problems, we devise a new color transform that gives high decorrelation gain along with small conversion and propagation error. We start with the KL transform. For that purpose, we use the Kodak image set [3] to estimate the correlation matrix between the color components. We normalize each component to have zero mean and unit variance as follows:

$$R = \frac{r - E[r]}{std(r)}, G = \frac{g - E[g]}{std(g)}, B = \frac{b - E[b]}{std(b)}.$$

Next we estimate the autocorrelation matrix as follows:

$$R_x = \begin{bmatrix} \text{var}(R) & E[RG] & E[RB] \\ E[RG] & \text{var}(G) & E[GB] \\ E[RB] & E[GB] & \text{var}(B) \end{bmatrix}. \quad (5)$$

Using the given training data set, we obtaine the following autocorrelation matrix,

$$R_x = \begin{bmatrix} 1 & 0.8525 & 0.7545 \\ 0.8525 & 1 & 0.9225 \\ 0.7545 & 0.9225 & 1 \end{bmatrix}. \quad (6)$$

Next, we find the eigenvectors and the eigenvalues of  $R_x$  using the following equation,

$$R_x \Phi = \Phi \Lambda \quad (7)$$

where  $\Phi = [\phi_1 \ \phi_2 \ \phi_3]$  are the set of eigenvectors and  $\Lambda$  is the diagonal matrix where diagonal terms are the set of corresponding eigenvalues ordered in decreasing value.

We obtained the following eigenvectors and eigenvalues,

$$\Phi^T = \begin{bmatrix} 0.5587 & 0.5968 & 0.5758 \\ -0.7860 & 0.1597 & 0.5972 \\ -0.2644 & 0.7863 & -0.5584 \end{bmatrix} \quad (8)$$

$$\Lambda = \begin{bmatrix} 2.6882 & 0 & 0 \\ 0 & 0.2536 & 0 \\ 0 & 0 & 0.0582 \end{bmatrix}$$

Now we can implement the KL transform to decorrelate the redundancy among the color components using  $\Phi^T$ . As shown in (8),  $\Phi^T$  is unitary matrix, i.e., it is normalized by L2 norm. So we need to scale the each row using L1 norm to guarantee the same dynamic range after color transform. Since the basis vectors are just scaled, it still maintains the characteristics of the KL transform.

The resulting transform is,

$$\Phi_{norm}^T = \begin{bmatrix} 0.3227 & 0.3447 & 0.3326 \\ -0.5095 & 0.1035 & 0.3870 \\ -0.1643 & 0.4887 & -0.3470 \end{bmatrix}. \quad (9)$$

Even though (9) guarantees the same dynamic range before and after the conversion, we slightly change (9) to use the same bias terms as in the YCbCr transform that makes the dynamic range of the chroma components ranging from 0 to 255. By setting the sum of negative coefficients equal to the sum of positive coefficients, we can use the same bias term 128 as YCbCr does. We can implement this by adjusting coefficients based on their absolute values. This approximation gives the following conversion,

$$\Phi_{norm}^T = \begin{bmatrix} 0.3223 & 0.344 & 0.333 \\ -0.5 & 0.106 & 0.394 \\ -0.161 & 0.5 & -0.339 \end{bmatrix} \quad (10)$$

The conversion formula in (10) is very close to the KL transform while maintaining the same dynamic range and bias terms of the well-known YCbCr transform.

As mentioned before, the transforms in (3) and (4) use one extra bit depth expansion for chroma components. It is based on the assumption that we can use separate bit depth for each component as defined in MPEG-4 AVC/H.264 Professional Extension. It means we can transform N bit data into (N+1) bit data. We take the advantage of this by scaling the forward transform by 2 in (10). It reduces the backward transform coefficients by half. Since the rounding error or coding noise propagation error is proportional to the sum of the square of each backward conversion coefficient, it reduces that error by quarter. So we propose to use the following transform, YSbSr:

$$\begin{bmatrix} Y \\ Sb \\ Sr \end{bmatrix} = \begin{bmatrix} 0.6460 & 0.6880 & 0.6660 \\ -1.0 & 0.2120 & 0.7880 \\ -0.3220 & 1.0 & -0.6780 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (11)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 0.5 & -0.6077 & -0.2152 \\ 0.5 & 0.12 & 0.6306 \\ 0.5 & 0.4655 & -0.4427 \end{bmatrix} \begin{bmatrix} Y \\ Sb \\ Sr \end{bmatrix}. \quad (12)$$

This new transform gives the high decorrelation gain as KL transform does while reducing the conversion error by increasing the dynamic range of the new color space. We can generalize this for any bit depth by employing (N+k) bit codec for N bit data.

### III. SIMULATION RESULTS

This section summarizes the simulation results. For this test we modified MPEG-4 AVC/H.264 reference software to support 4:4:4 chroma format and N-bit input data. The test materials are the widely used “Kodak set” of 24 images of size 512x768, captured with a high-quality 3-CCD camera [3]. We coded this 24 images using the above modified reference software. Since there is no temporal correlation, we coded the test images using intra prediction mode only. To compare the efficiency of each color transform, we first converted the original input RGB data into the new color space. Then we code the images using the above software and converted back the coded data into RGB space.

Figure 1 shows the simulation results for each color transform. It shows the efficiency of the proposed YSbSr transform since it reduces the conversion error most. The coding gain difference between YCoCg-R and YFbFr is very negligible even though YFbFr approximates the KL transform more closely. The well-known YCbCr performs worst as shown in Figure 1. The input data for our simulation have different bit-depth for each component depending on color transform. So we need to adjust the quantizer for each component to compare the coding efficiency. For that purpose, we applied a quantizer that is independent of input data bit depth [7].

### IV. CONCLUSIONS

In this paper, we introduced a new color transform that focuses on high coding gain and low conversion error for RGB data. We showed that we can achieve better coding efficiency by reducing the conversion and propagation error. The proposed coding technology reduces the color conversion error and coding error propagation that are important for professional applications by employing large dynamic range in the coding space. We can generalize this for any bit depth by employing (N+k) bit codec for N bit data. It makes the proposed color transform a good candidate for applications that need to preserve the original color fidelity.

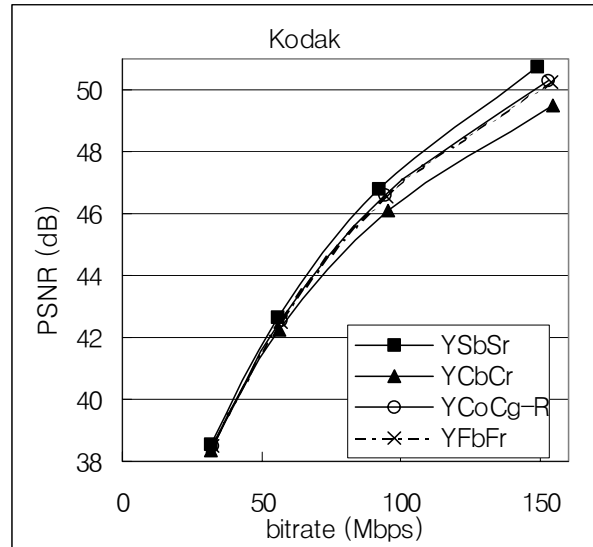


Figure 1: Simulation results for the Kodak images using various color transforms.

### REFERENCES

- [1] P. Hao, Q. Shi, “Comparative study of color transforms for image coding and derivation of integer reversible color transform,” *IEEE Int’l Conf. Pat. Recog. (ICPR’00)*, vol. 3, Barcelona, Spain, Sep. 2000.
- [2] “Professional extensions amendment to ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC,” *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6*, Document JVT-J047, Dec. 2003.
- [3] H. Malvar and G. Sullivan, “YCoCg-R: A color space with RGB reversibility and low dynamic range,” *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6*, Document JVT-I014, Sep. 2003.
- [4] P. Topiwala and C. Tu, “New invertible integer color transforms based on lifting steps and coding of 4:4:4 video,” *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6*, Document JVT-I015, Sep. 2003.
- [5] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1997.
- [6] G. Sullivan, “Approximate theoretical analysis of RGB to YCbCr to RGB conversion error,” *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6*, Document JVT-I017, Sep. 2003.
- [7] W. Gish and H. Yu, “Extended sample depth: Implementation and characterization,” *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6*, Document JVT-H016, May 2003.