

논문 2004-41SD-1-9

보정된 X-색도 좌표를 이용한 색온도 변환 시스템의 성능 개선 및 구현

(Performance Improvement and Implementation of Color-Temperature Conversion System using Compensated X-Chromaticity Coordinate)

변형수*, 강봉순**, 양훈기***

(Hyungsoo Byun, Bongsoon Kang, and Hoongee Yang)

요약

본 논문은 보정된 X-색도 좌표를 이용한 색온도 변환 방법을 제안하고자 한다. 제안된 방법은 기존의 색온도 변환 시스템에서 문제점인 색온도 계산값의 오차를 줄이기 위하여 회귀방정식을 사용한다. 또한, 광범위한 영역에서 색온도 변환을 실현하기 위하여 색온도 영역을 1,667K에서 25,000K까지 확대하고자 한다. 제안한 방법은 Robertson 방법 및 기존의 색온도 계산 방법과 성능의 비교 분석을 통하여 타당성을 보이고자 한다. 본 논문에서 제안된 방법은 하드웨어로 설계하여 Xilinx FPGA 인 XCV2000E-BG560를 통하여 TV 시스템에서 실시간으로 동작을 검증한다.

Abstract

In this paper, we propose the color-temperature conversion system with compensated X-chromaticity coordinate. It uses a linear regression to improve the error of color-temperature calculation in conventional systems. It also extend the color-temperature range from 1,667K to 25,000K to provide a wider color-temperature range. We show the effectiveness of the proposed method by comparing the performance of the proposed method with those of the Robertson's and the existing methods. The proposed method is experimentally verified by displaying the results on a TV system through the Xilinx FPGA XCV2000E-BG560.

Keywords: 색온도, 색도 좌표, CIE-XYZ, Robertson, white point line

I. 서론

사람이 정경을 바라볼 때, 조명의 영향으로 느껴지는 색조를 인지 광원이라고 한다^[1-3]. 백열등에서는 전

체적으로 붉은 색조를 느끼며, 일광에서는 백열등에 비하여 푸른 색조를 느끼는 것을 말한다. 지금까지의 칼라 변환 시스템은 RGB의 변환기술(예, saturation, contrast 등)을 통하여 칼라를 변환하는데^[4], 이 변환 기술은 입력 영상에서 느껴지는 색조를 고려하지 않고 특정 값으로 RGB를 고정함으로써 칼라의 왜곡현상과 영상의 부자연스러움을 가져온다. 또한, 동일한 영상이라고 하더라도 원 영상의 특징을 고려하지 않고 디스플레이 매체의 특성에 따른 칼라 왜곡보정(Gamma Correction)을 적용하고 있다^[5]. 따라서 이러한 문제점을 보완하고 사용자가 원하는 다양한 색조로 변환할

* 정회원, 다들 멀티미디어 주식회사 연구원
(Tamul Multimedia Co. Ltd.)

** 정회원, 동아대학교 전기전자컴퓨터 공학부
(School of Electrical, Electronic, and Computer Eng., Dong-A Univ.)

*** 정회원, 광운대학교 전자공학과
(Dept. of Radio Science and Eng., Kwangwoon Univ.)

※ 본 논문은 2000년도 한국학술진흥재단의 지원에 의하여 연구되었음(과제 번호 : KRF-2000-041-E00185).

접수일자 : 2003년1월6일, 수정완료일 : 2003년12월26일

수 있는 변환 기술이 필요하다고 할 수 있다.

색온도는 광원을 수치적으로 표시한 것인데, 이상적인 완전 흑체가 방출하는 빛은 플랑크의 복사법칙에 의해 온도가 정해진다^[6]. 물체가 발광하는 색이 어떤 온도의 흑체가 복사하는 색과 같다면, 그 흑체의 온도를 물체의 색온도라고 한다. 가령, 전구의 빛은 2,800 K, 형광등의 빛은 4,500~6,500 K, 정오의 태양 빛은 5,400 K, 흐린 날의 낮빛은 6,500~7,000 K, 맑은 날의 푸른 하늘빛은 12,000~18,000 K 정도의 색온도이다. 색온도에 대한 연구는 입력 영상에 대하여 사용자가 원하는 색온도(예: 일광 효과, 석양 효과 등등)를 반영함으로써 다양한 색의 변환을 추구한다^[7]. 색온도는 인간의 감성에 많은 영향을 준다. 색온도가 낮은 붉은 색조의 영상은 사람에게 따뜻한 느낌을 갖게 하고, 색온도가 높은 푸른 색조의 영상은 시원한 느낌을 준다.

색온도에 대한 표시 방법이 Robertson 알고리즘에서 정량화 되어 있는데, CIE-XYZ 색도 좌표계에서 XY의 2차원 정보를 이용하여 수치화하고 있다^[8]. 이 2차원에 대한 정보는 복잡한 식으로 표시되어 있어 많은 용량의 하드웨어가 요구되기에 구현의 실용성이 문제가 되었다. 그리고 이를 해결하기 위해 입력 영상의 X-색도 좌표만을 계산하는 기존의 색온도 변환 방법이 제안되었지만^[9], 특정 단일 영상에 대한 잘못된 색온도 계산 및 낮은 색온도 계산의 제한(3,000 K 이하) 등의 문제가 있었다. 그 색온도 오차의 이유는, 여러 가지 물체를 포함하는 영상의 색온도는 대부분 white point 선 부근에 위치하고 있어서 X-색도 좌표만으로 색온도를 계산하여도 Robertson 알고리즘에서 계산된 색온도와 차이가 ± 100 K 이하로 상대적으로 작은 오차를 유발하고, 이 정도의 오차는 동영상에서 사람의 눈으로 감지하기에 상대적으로 작은 수치였다. 실험적으로는 입력 색온도가 10,000 K 이상에서는 ± 300 K 정도의 오차가 눈으로 감지되지 않았다^[10]. 색온도 하한치에 대한 3,000 K 설정 이유는 X-색도 좌표를 이용한 색온도 계산 방법에서는 색온도의 범위가 3,000~7,000 K에 적용할 수 있는 수식을 이용하였기에 나타난 문제로 색온도가 3,000 K 이하인 Yellow와 Red 계통의 영상이 입력될 때에는 3,000 K로 설정되어 색온도 계산의 오차를 유발하였다. 따라서, 보다 선명한 화질을 요구되는 응용분야에 있어서는 보다 정확한 색온도 계산이 필요하다고 할 수 있다. 이에, 본 논문에서는 기존의 색온도 계산 방법에 대한 문제점을 제시하고, 이를 해결하기 위한 방법을 제안하고자 한다.

본 논문의 II장에서는 기존 색온도 변환 시스템의 문제점을 지적하며, III장에서는 기존 색온도 변환 시스템의 문제점 해결 방법과 제안된 색온도 변환 시스템의 구성 내용 서술한다. IV장에서는 제안된 색온도 변환 시스템의 구현 및 결과 비교하고, 하드웨어 검증을 수행하면서 본 연구에서 제안한 방법의 타당성을 다루고자 한다.

II. 기존 색온도 변환 방법의 문제점

먼저 하드웨어로 구현된 색온도 변환 시스템의 방법은 기존의 논문인 참고문헌 [9]에 나타나 있다. 간단히 시스템의 과정을 나타내면 입력 영상의 RGB 신호를 CIE-XYZ 좌표로 변환하고, 이를 이용 영상에 대한 색도 (X, Y) 좌표를 구한다. 구해진 색도 좌표를 사용하여 색온도를 계산하고, 원하는 색온도로 변환을 수행한다. 그리고 최종적으로 다시 RGB 신호의 출력 영상으로 변환하는 과정을 거치게 된다. 기존의 논문에 언급되어 있는 색온도 변환 방법에서 몇 가지의 문제점이 있다^[10]. 색도값 중 x_s 값만을 사용하기 때문에 색도값이 white point 선보다 멀리 떨어진 영상이 입력될 때, 예를 들면, 녹색의 영상(x_s, y_s)이 입력될 경우 아래 그림 1과 같이 정확한 색온도를 검출하지 못하는 문제점을 가진다. 여기서 1D는 X-색도 좌표를 이용한 색온도 계산 방법을, 2D는 Robertson 알고리즘에 의한 방법을 나타낸다. X-색도 좌표를 이용한 방법에서는 정확한 색온도 위치인 '정확한 x_s '를 이용하여 색온도를 계산하지 않기에 색온도 오차를 유발한다고 할 수 있다.

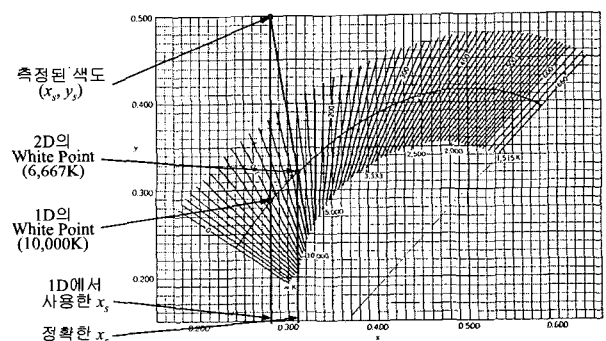


그림 1. X-색도 좌표를 이용한 색온도 계산 방법
Fig. 1. Method of color-temperature calculation using X-chromaticity coordinate.

또한, 기존 방법에서 사용한 식이 아래와 같이 색온도 3,000~25,000 K의 범위를 가지고 있기 때문에 색온도가

낮은 영상이 입력될 경우 색온도는 3,000 K로 고정되어 있다. 색온도 3,000~7,000 K와 7,000~25,000 K 영역에서 사용한 수식은 식 (1)과 (2)에 나타내었다. 하지만 붉은 색 계통의 영상은 3,000 K 이하의 색온도를 가진다. 이로 인해 색온도가 낮은 붉은 영상이 입력될 경우 제대로 검출하지 못하게 되어 사용자가 원하는 색온도로 변환시에 오차를 발생시키는 문제가 있다.

$$3,000\sim 7,000\text{ K} : x_s = -4.6070 \frac{10^9}{T_c^3} + 2.9678 \frac{10^6}{T_c^2} + 0.09911 \frac{10^3}{T_c} + 0.244063 \quad (1)$$

$$7,000\sim 25,000\text{ K} : x_s = -2.0064 \frac{10^9}{T_c^3} + 1.9018 \frac{10^6}{T_c^2} + 0.24748 \frac{10^3}{T_c} + 0.237040 \quad (2)$$

위의 두 가지 문제로 인하여 입력되는 영상이 심하게 왜곡되는 경향이 발생하게 되고 사용자가 원하는 색온도로 정확하게 영상을 변환하지 못하게 된다. 아래의 표 1에서는 단일 색을 가지는 영상을 사용하여 색온도를 계산한 경우이다. 결과와 같이 white point 선에서 멀리 떨어져 있는 Green와 Magenta 계통의 영상과 색온도가 3,000 K 부근 및 이하의 Red와 Yellow 계통의 영상에서는 색온도 계산에서 커다란 오차를 유발시키는 것을 볼 수 있다.

표 1. 단일 영상에서 계산된 색온도 차이
Table 1. Differences in color temperatures obtained from single patterns.

Input Images	2D (Robertson 방법)	1D 색온도 계산 방법	색온도 차이
Red	1,667 K	3,000 K	-1,333 K
Green	7,245 K	25,000 K	-17,755 K
Blue	25,000 K	25,000 K	0 K
Cyan	25,000 K	25,000 K	0 K
Yellow	3,629 K	3,000 K	629 K
Magenta	1,979 K	4,773 K	-2,794 K

III. 제안된 색온도 변환 방법

앞서 언급한 것과 같이 기존 색온도 계산방법에서의 문제점을 해결하기 위한 방법을 본 장에서 다룬다. 본 논문에서는 색도 (x_s, y_s) 좌표를 사용하여 X-색도 좌표만 사용하는 단점을 보완한 보정된 X 좌표를 이용하는 색온도 계산 방법을 제안한다. 제안하는 색온도 계산방법은 위에서 설명한 문제점인 색도 x_s 값만을 가지고 색온도를 추출, white point 라인보다 멀리 떨어진 곳에 색도

가 위치할 경우 정확한 색온도를 검출하지 못하는 문제와 색온도 범위가 정해졌기 때문에 색온도가 낮은 영상이 입력될 경우 원 영상이 왜곡되는 문제를 해결하기 위해 방안을 제시하고자 한다.

본 논문에서 제안하는 색온도 계산 방법의 기본적인 원리는 그림 2와 같다. 입력 영상의 색도 (x_s, y_s) 좌표일 경우 기존의 1D 색온도 계산 방법에 의하면 white point 라인상의 (x_s, y_{dd}) 좌표의 색온도가 검출되게 된다. 그러나 실제 Robertson 방법에서는 (pix, piy) 좌표의 색온도가 검출되어야 한다. 즉 *differ_pix* 만큼의 색온도 오차가 존재하게 된다. 여기서 색도가 white point 라인보다 멀리 떨어진 특정 영상이 입력될 경우 *differ_pix*가 커지게 되고, 따라서 색온도 오차 역시 커지게 되어 원 영상이 왜곡되게 된다. 따라서 제안하는 색온도 계산 방법은 이 *differ_pix* 값을 구하여 입력된 색도 x_s 좌표값에 *differ_pix* 값을 더하여 보정하게 된다.

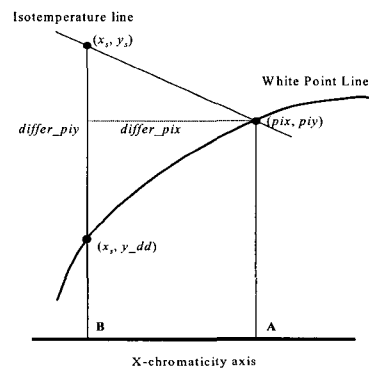


그림 2. 제안하는 색온도 계산 방법
Fig. 2. Method of proposed color-temperature calculation.

그림 3은 X-색도 좌표의 0.15~0.58 범위에서 0.02의 간격으로 데이터를 입력하여 구한 *differ_piy*와 *differ_pix*와의 관계를 그래프로 나타낸 것이다. 여기에서 0.15는 Cyan에 해당되는 값이고 0.58은 색온도의 새로운 하한치에 해당되는 값으로 색온도로 표시하면 25,000 K와 1,667 K에 해당된다. 그림에 나타난 것과 같이 각 라인 사이에 일정한 규칙이 있음을 알 수 있다. 이 규칙을 이용하여 *differ_pix* 값을 구할 수 있다. 여기서 각 라인은 색도 x_s 좌표를 나타낸다. 그래프에서 *differ_piy*에 대한 *differ_pix* 값을 ROM의 데이터로 저장하고 색도 x_s 값과 *differ_piy* 값을 이용하여 그림 2에 표시한 $x_s, differ_piy$ 범위를 찾아, 그 주위의 4개의 점에 대한 데이터를 ROM에서 사용하여 데이터 사

이의 비례관계를 이용하여 연산을 하면 $differ_pix$ 를 구할 수 있다.

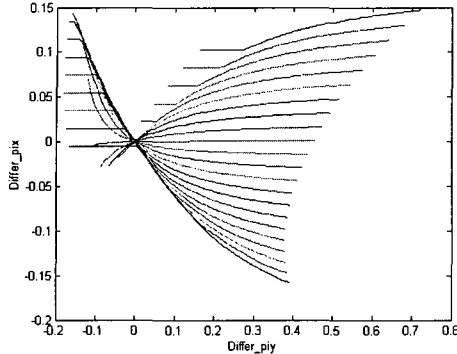


그림 3. $differ_piy$ 와 $differ_pix$ 와의 관계 그래프
Fig. 3. Graph of relationship between $differ_piy$ and $differ_pix$.

앞에서 언급한 색온도 범위 제약에 따른 왜곡현상은 3,000 K 이하의 색온도를 가지는 영상이 입력될 때 3,000 K로 제한을 두었기 때문에 생기는 현상이다. 식 (1)과 (2)에서와 같이 사용할 경우 3,000 K에서 변곡점이 발생, 실제 색온도 계산에 적용하기 어려웠다. 따라서 참고문헌 [8]에서 제시한 1,667~25,000 K 영역의 31개 데이터를 사용하여 회귀방정식을 적용하여 새로운 식을 도출하였다^[11]. 식 (3)과 같이 pix 를 3차 방정식 수식으로 구성을 하고, 식 (4)와 같이 31개의 데이터와 식 (3)과의 평균오차를 사용하였다.

$$pix = a_3 T_c^{-3} + a_2 T_c^{-2} + a_1 T_c^{-1} + a_0 \quad (3)$$

$$S = \frac{1}{31} \sum_{i=0}^{31} \{pix - f(T_c^{-1})\}^2 = \frac{1}{31} \sum_{i=0}^{31} \{a_3 T_c^{-3} + a_2 T_c^{-2} + a_1 T_c^{-1} + a_0 - f(T_c^{-1})\}^2 \quad (4)$$

식 (4)의 평균오차를 최소화함으로써 계수(a_k)의 값을 구할 수 있는데, 아래 식 (5)와 같이 계수에 따른 편미분 값을 '0'으로 설정하면 된다.

$$\frac{\partial S}{\partial a_0} = \frac{\partial S}{\partial a_1} = \frac{\partial S}{\partial a_2} = \frac{\partial S}{\partial a_3} = 0 \quad (5)$$

이의 방법으로 구한 새로운 수식은 색온도에 따른 적절한 값을 설정하면 식 (6)과 (7)과 같이 구할 수 있다.

$$\begin{aligned} & i) 4,000\text{K} \leq T_c \leq 25,000\text{K} \\ & \quad pix = -3.0258469 \frac{10^9}{T_c^3} + 2.1070379 \frac{10^6}{T_c^2} + 0.2226347 \frac{10^3}{T_c} + 0.24039 \\ & ii) 1,650\text{K} \leq T_c \leq 4,000\text{K} \\ & \quad pix = -0.2661239 \frac{10^9}{T_c^3} - 0.2343580 \frac{10^6}{T_c^2} + 0.8776956 \frac{10^3}{T_c} + 0.179910 \end{aligned} \quad (6)$$

(6)

$$\begin{aligned} & i) \quad pix \leq 0.38405 \quad (4,000\text{K} \leq T_c \leq 25,000\text{K}) \\ & \quad piy = 3.0817580pix^3 - 5.8733867pix^2 + 3.75112997pix - 0.37001483 \end{aligned}$$

$$\begin{aligned} & ii) \quad 0.38405 \leq pix \leq 0.50338 \quad (2,222\text{K} \leq T_c \leq 4,000\text{K}) \\ & \quad piy = -0.9549476pix^3 - 1.37418593pix^2 + 2.09137015pix - 0.16748867 \end{aligned}$$

$$\begin{aligned} & iii) \quad otherwise \quad (1,667\text{K} \leq T_c \leq 2,222\text{K}) \\ & \quad piy = -1.1063814pix^3 - 1.34811020pix^2 + 2.18555832pix - 0.20219683 \end{aligned}$$

(7)

그림 4에 회귀방정식을 이용하여 구한 식과 Robertson 방법을 이용하여 구한 식을 그래프로 나타내었다. 그림에서 보듯이 2개의 그래프가 거의 일치함을 알 수 있다.

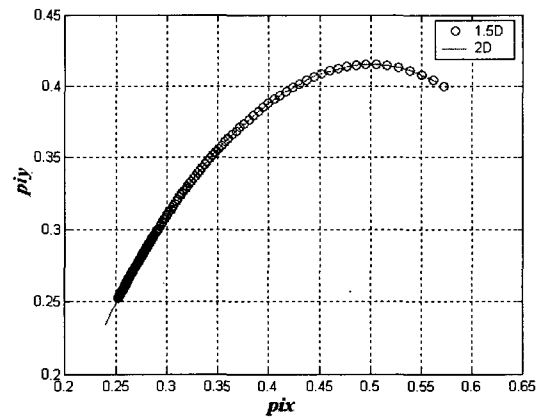


그림 4. Robertson 방법과 회귀방정식으로 구한 데이터
Fig. 4. Data obtained from Robertson and linear regression methods.

여기서 2D는 Robertson 방법에 의한 것을 나타내며, 1.5D는 제안된 방법의 회귀방정식을 이용하여 구한 식에 의한 것을 나타낸 것이다.

이상과 같이 제안된 방법을 이용하여 기존의 두 가지 문제점인 white point 선과 멀리 떨어진 영상에서 잘못된 색온도 계산이나, 색온도 계산을 3,000 K 이하에서는 수행할 수 없는 문제점을 해결할 수 있었다.

IV. 하드웨어 구조

그림 5는 본 논문에서 제안하는 색온도 변환 시스템의 블록도를 나타낸 것이다. 기존의 색온도 변환 시스템에서 X-색도를 보정하기 위한 블록이 새로 추가되고, 색온도의 제한 범위를 1,667 K로 늘리기 위해 색온도 계산하는 블록이 수정되었으며, 이들을 위해 관련 블록들도 수정되었다. 본 논문에서 제안하는 시스템은 그림 5와 같이 13개의 기능 블록으로 구성되어 있다. I/OFC_1,

EffPixel_sel, Chroma_Calc, Temperature_Calc, TP_Coeff, XYZtoRGB_Coeff, I/OFC_2, Time_alignment, I2C의 9개 블록은 기존 블록과 동일한 기능을 수행하며, 나머지 4개 블록은 새로 추가되거나 기존 블록

에서 분리된 블록이다. 본 장에서는 기존의 시스템에서 수정 및 추가된 기능 블록의 구조에 대해서만 다루며, 나머지 기존의 블록은 참고문헌 [9]를 참조하면 된다.

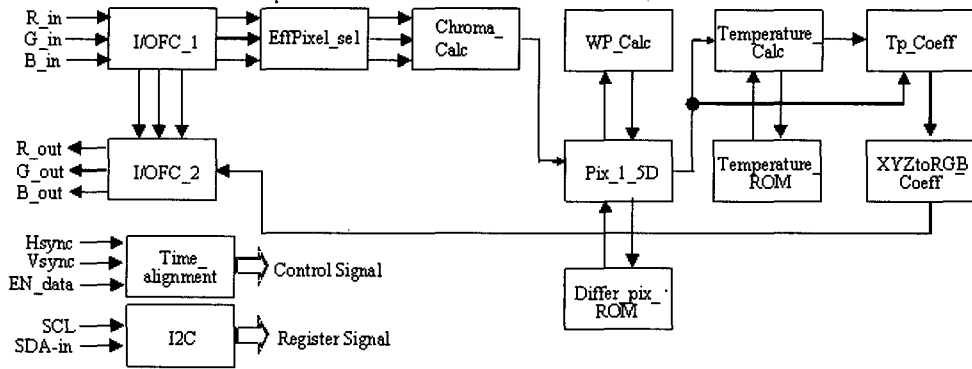


그림 5. 제안하는 색온도 변환 시스템의 블록도
Fig. 5. Block diagram of proposed color-temperature conversion system.

X-색도를 보정하기 위해 추가된 기능 블록으로는 그림 5에서 Pix_1_5D, WP_Calc, 그리고 Differ_pix_ROM의 세 블록이다. Pix_1_5D 블록은 색도 x_s 를 pix 로 보정하기 위한 주 기능을 담당하는 블록이고, 보조 기능의 블록으로는 식 (7)을 계산하는 WP_Calc 블록과 그림 3에서의 $differ_pix$ 값이 저장되는 Differ_pix_ROM 블록이 담당하고 있다.

그림 6은 WP_Calc 블록도를 나타낸 것이다. Chro-

ma_Calc 블록에서 계산된 X-색도 좌표인 pix 값을 이용하여 white point 라인상의 Y-색도 좌표를 계산하는 블록이다. 기능적으로는 회귀방정식을 사용하여 구한 식 (7)을, 색온도 계산은 영상의 한 필드(혹은 프레임) 동안에 연산하면 되기에 곱셈기(Mul) 한 개를 사용하여 구현할 수 있도록 식 (8)와 같이 인수분해형으로 변환하여 구현하여 하드웨어 복잡도를 줄이고자 하였다.

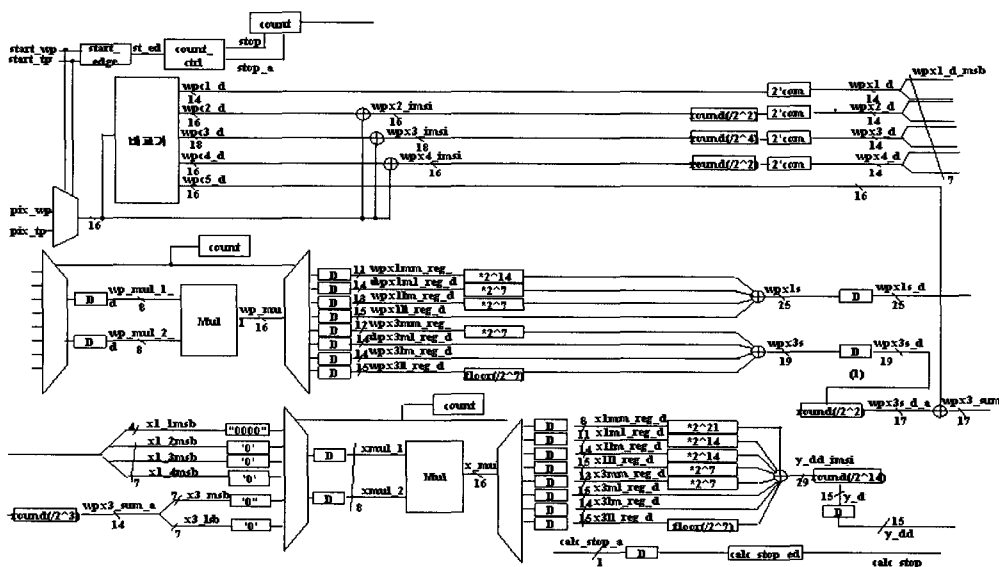


그림 6. WP_Calc의 블록도
Fig. 6. Block diagram of the WP_Calc.

- i) $pix \leq 0.38405$ (4,000 K $\leq T_c \leq 25,000$ K)
 $piy = 3.0817580 \times (pix - 0.11964896) \times ((pix - 0.89310347)^2 + 0.20585296)$
- ii) $0.38405 \leq pix \leq 0.50338$ (2,222 K $\leq T_c \leq 4,000$ K)
 $piy = -0.9549476 \times (pix - 0.8630554) \times (pix + 2.3872016) \times (pix - 0.0851291)$
- iii) otherwise (1,667 K $\leq T_c \leq 2,222$ K) (8)
 $piy = -1.1063814 \times (pix - 0.8508221) \times (pix + 2.1683678) \times (pix - 0.0990599)$

그림 7은 Pix_1_5D의 블록도이다. Pix_1_5D 블록은 WP_Calc 블록에서 계산된 white point 라인 상의 Y-색도 좌표값을 사용하여 Chroma_Calc 블록에서 구한 piy 좌표와의 거리의 차를 이용하여 pix 좌표를 보정하는 블

록이다. 또한 구간별 pix 좌표와 white point 라인의 x 축 좌표와 거리를 저장하고 있는 Differ_pix_ROM 블록으로 메모리 번지를 발생시킨다. 최종적으로 16 bit의 보정된 X-색도 좌표인 pix를 출력으로 내보낸다.

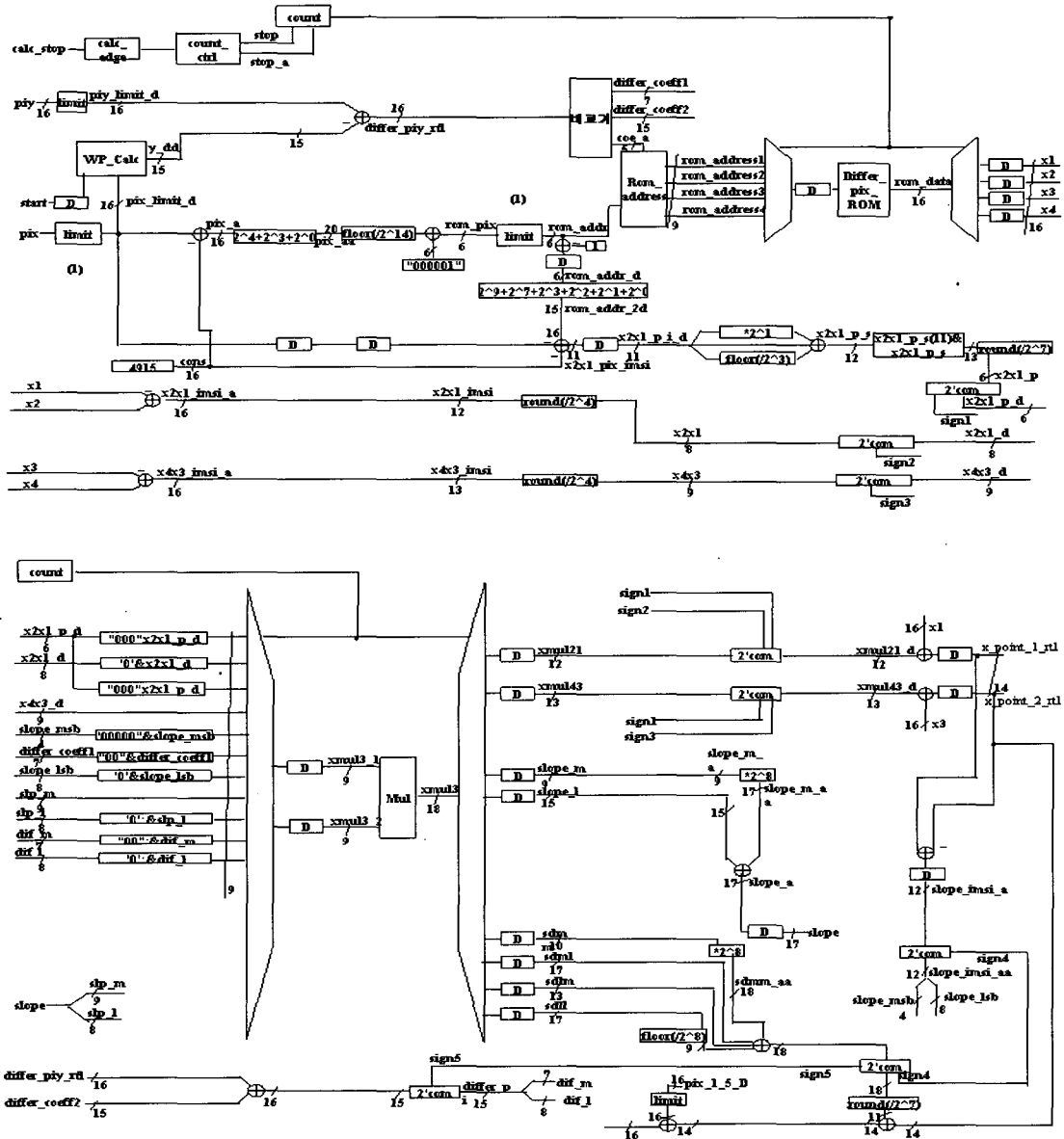


그림 7. Pix_1_5D의 블록도
 Fig. 7. Block diagram of the Pix_1_5D.

Differ_pix_ROM 블록은 구간 별 *pix* 좌표와 white point 라인 상의 거리를 저장하는 기능을 가지며, ROM 형태로 구성되어 있는 블록이다. 전체 크기는 352×16이며, 순차적으로 4개의 메모리 번지를 입력으로 하여 16 bit의 데이터인 *x1*, *x2*, *x3*, *x4*를 순차적으로 발생시킨다. 그리고 색온도의 계산 범위를 1,667 K로 늘리기 위해 수정된 기능 블록으로는 Temperature_Calc와 Temperature_ROM 블록이다. Temperature_Calc 블록은 입력 영상의 한 프레임의 색도 *pix*를 가지고 그 영상 프레임의 색온도를 구하는 블록이다. 여기서 구한 색온도를 기준

으로 사용자가 원하는 만큼의 색온도 변화를 줄 수 있다. Temperature_Calc 블록과 Temperature_ROM 블록은 그림 8과 같이 2개의 블록이 상호 연관하여 동작하게 되어 있다. 기존 시스템과 비교해서 단지 색온도의 계산 범위가 1,667 K로 늘어났으므로 메모리 번지가 약간 수정되었을 뿐이다. 그리고 Temperature_ROM이 기존에 비해 크기가 커졌다. Temperature_ROM 블록은 색온도를 계산하기 위해 필요한 데이터를 ROM 형태로 저장되어 있는 블록이다. ROM의 크기는 512(depth)×46(length)으로 구성되어 있다.

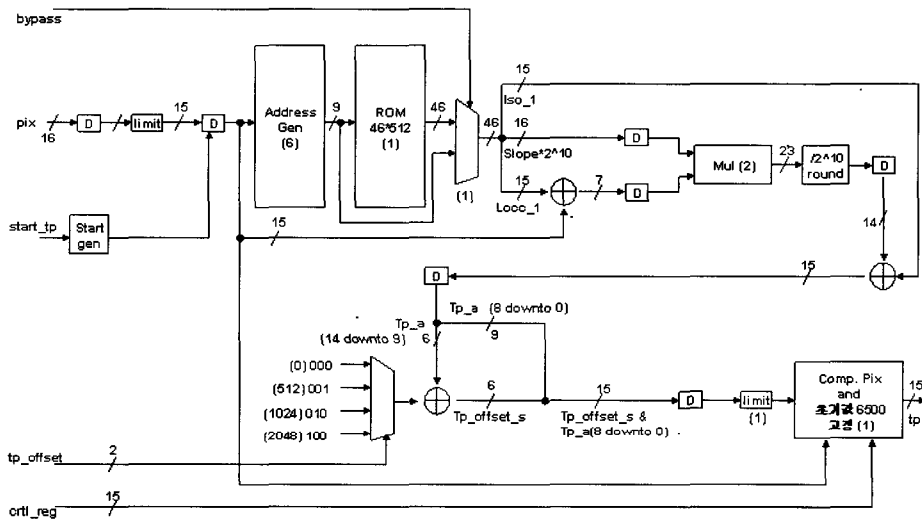


그림 8. Temperature_Calc와 Temperature_ROM 블록도
Fig. 8. Block diagram of the Temperature_Calc and the Temperature_ROM.

V. 구현 결과

표 2는 단색 영상을 입력하여 시뮬레이션한 결과이다. 표 2에서 Robertson 방법을 이용하여 구한 색온도는 1,667~25,000 K를 계산 영역으로 사용하였다. 기존의 색온도 계산 방법은 식 (1)과 (2)에서와 같이 3,000 K까지의 범위를 사용하였으나 제안하는 색온도 계산 방법은 회귀방정식으로 구한 식을 적용하여 1,667~25,000 K의 색온도 범위를 가지게 하여 색온도를 검출하였다. X-색도 좌표를 이용한 색온도 계산방법에 있어서 표 1에 나타낸 것과 같이, Green 영상은 -17,755 K의 오차를 가지고 있었고 Yellow와 Magenta에서 각각 629 K와 -2,794 K의 많은 오차를 가지고 있었다. 표 2는 제안한 방법을 이용하여 얻은 색온도 차이를 보여준다. X-색도 좌표를 이용한

방법에서는 색온도의 하한값이 3,000 K로 설정이 되어 있어서, Red 영상에서는 Robertson 방법의 결과와 -1,333 K의 차이를 보였지만, 본 논문에서 제안한 회귀방정식을 적용하여 보정된 색온도 계산 방법에서는 하한값을 1,667 K로 확대하였기에 색온도 차이가 없었다. 또한, Green, Yellow 및 Magenta 영상에서도 색온도 차이가 -9 K, 9 K 및 -201 K로 X-색도 좌표를 이용한 방법과 비교하여 상당히 개선됨을 확인하였다. 또한, 제안한 방법에서 사용한 회귀방정식을 이용한 새로운 수식을 이용하였을 경우에는 그림 6에서 보인 것과 같이 논리회로로 구현이 가능하였다. 따라서 X-색도 좌표 연산에서는 이 기능을 구현하기 위하여 ROM을 사용하였는데 235,578 bits의 용량이 필요하였는데, ROM이 필요 없는 논리회로로 대체하여 소요 ROM 용량을 매우 줄였다고 할 수 있다.

표 2. 제안된 방법을 이용한 단일 영상에서 계산된 색온도 차이

Table 2. Differences in color temperatures obtained from single patterns using the proposed method.

Input Images	2D (Robertson 방법)	기존 1D 색온도 계산 방법	제안된 색온도 계산 방법	2D와 제안 방법의 색온도 차이
Red	1,667 K	3,000 K	1,667 K	0 K
Green	7,245 K	25,000 K	7,254 K	-9 K
Blue	25,000 K	25,000 K	25,000 K	0 K
Cyan	25,000 K	25,000 K	25,000 K	0 K
Yellow	3,629 K	3,000 K	3,620 K	9 K
Magenta	1,979 K	4,773 K	2,180 K	-201 K

본 논문에서는 VHDL을 제안한 방법을 설계하였으며, Synopsys simulator를 사용하여 검증을 수행하였다. 하드웨어 복잡도를 살펴보기 위해, TSMC 0.25 um ASIC library를 이용하여 디지털 논리회로로 합성하였다. 제안한 방법의 동작 검증을 위하여 HDTV급의 fLCD TV 시스템을 사용하였는데, 하드웨어는 14.82 ns(67.43 25 MHz)의 동작 클록 주기를 사용하였다. 그 결과 본 논문에서 제안하는 방법의 하드웨어 복잡도는 189,982 게이트이다. X-색도 좌표를 이용한 방법은 37 ns(27 MHz)의 동작 클록으로 구현된 시스템이기에 제안한 방법과 하드웨어 복잡도를 바로 비교를 할 수 없었지만, 필요 ROM 용량이 전체 267,752 bits에서 32,174 bits로 약 88%를 감소할 수 있었다.

제안한 색온도 변환 방법의 실시간 검증을 위해 Xilinx Virtex-E FPGA XCV2000E-6BG560을 사용하였다. 그림 9는 본 논문에서 제안한 보정된 X-색도 좌표를 이용한 색온도 변환 시스템의 성능을 확인하기 위하여 삼성전자의 해당분야 전문가의 협조를 통하여 제작된 검증용 데모 보드이며, 그림 10은 데모 보드를 내장한 43" fLCD TV 시스템이다. 그림 10의 TV 세트 화면의 왼쪽 반은 입력 영상을 그대로 보여주고(bypass mode), 화면의 오른쪽 반은 본 논문에서 제안한 방법을 사용하여 색온도가 25,000 K로 변환된 영상이 실시간으로 나타나게 하였다. 실시간 검증에서도, 입력 영상(왼쪽 반)보다 더 높은 25,000 K로 변환된 영상(오른쪽 반)에서는 푸른 색조가 나타남을 볼 수 있다. 이를 통하여 본 논문에서 제안된 색온도 변환 방법의 타당성 및 실용성을 다시 검증하였다.

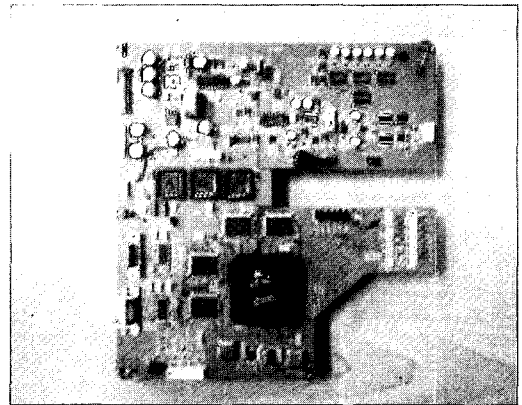


그림 9. 제안하는 색온도 변환 시스템 구현을 위한 보드
Fig. 9. Board for implementation of proposed color-temperature conversion system.

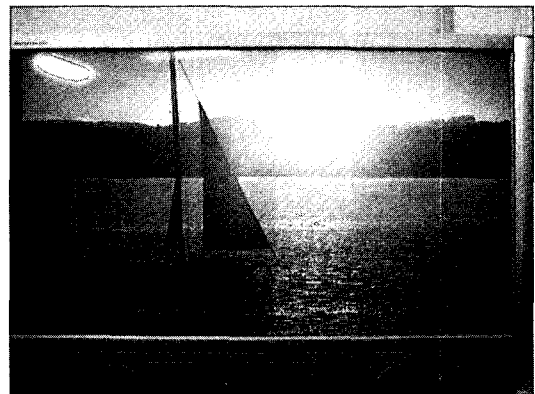


그림 10. 제안하는 색온도 변환 시스템의 데모 TV 세트
Fig. 10. Demonstration TV set of proposed color-temperature conversion system.

VI. 결 론

X-색도 좌표를 이용한 색온도 계산에서는 3,000~

25,000 K의 색온도 범위로 한정되어 있고, white point 선과 멀리 떨어진 영상이 입력되는 경우에 색온도의 오차가 많이 발생하였다. 이런 문제점들을 해결하기 위하여 본 논문에서는 색온도 계산에서 색도 (x_s, y_s) 좌표를 이용하여 색도 x_s 좌표값을 회귀방정식을 이용하여 새로운 색온도 수식인 보정된 pix 를 사용하여 보다 정확한 색온도를 구할 수 있었고, 색온도의 계산 범위를 1,667~25,000 K로 늘릴 수도 있었다. 실시간 검증을 위하여 Xilinx Virtex-E FPGA XCV 2000E- 6BG560를 사용하여 제안한 방법의 타당성을 검증하였다. 따라서 본 논문에서 제안한 방법은 기존 방법보다 개선된 나아진 성능으로 색온도 변환 시스템을 구현함으로써 다양한 영상 매체에서의 적용 범위를 더욱 넓힐 수 있었다.

감사의 글

본 논문에서 제안한 방법의 타당성을 검증하기 위한 검증용 데모 보드 제작에 아낌없는 협조를 해주신 삼성전자의 이봉근연구원, 조봉환선임, 김영선수석, 홍창완상무님께 감사를 드리고자 합니다. 또한, 본 연구는 IDEC에서 제공한 Synopsys와 Xilinx tool을 사용하여 수행하였습니다.

참고 문헌

[1] A.P. Petrov, C.Y. Kim, Y.S. Seo and I.S. Kweon, "Perceived illuminant measured, Color Research

- and Application," Vol. 23, No. 3, pp. 159-168, 1998.
- [2] 강봉순, 장근식, "색온도 변환장치 및 방법," 대한민국 출원 특허, 출원번호: 10-2001-0063418, 2001
- [3] 강봉순, "색온도를 이용한 디스플레이 시스템의 칼라 변환 기술", IDEC News Letter, pp. 34-37, 2002
- [4] K. Jack, Video Demystified: A Handbook for the Digital Engineer, HighText Pub., 1996.
- [5] C.A. Poynton, A Technical Introduction to Digital Video, John Wiley & Sons Inc., 1996.
- [6] W.N. Sproson, Colour Science in Television and Display Systems, Adam Hilger Ltd., 1983.
- [7] H. Lee, B. Lee, O. Moon and B. Kang, "Design and Implementation of Color Management System with X-chromaticity Coordinate," 2001-Summer IDEC Conference, pp. 157-158, 2001.
- [8] G. Wyszecki and W.S. Stiles, Color Science, Concepts and Methods, Quantitative Data and Formulae, John Wiley & Sons Inc., 1982.
- [9] 이호남, 이봉근, 문오학, 강봉순, "X-색도 좌표를 이용한 색온도 변환 시스템 구현," 대한전자공학회논문지, 제39권 SD편 제8호, pp. 64-73, 2002
- [10] 강봉순, fLCD용 Magic Color Hardware Implementation, 최종연구보고서, 삼성전자, 2002.6.
- [11] G.J. Borse, Numerical Methods with MATLAB, a Resource for Scientists and Engineers, PWS Publishing Company, 1997.

저 자 소 개



변형수(정회원)

2001년 2월 : 동아대학교 공대 電子工學科(공학사). 2003년 2월 : 동아대학교 대학원 전자공학과(공학석사). 2003년 1월 ~ 현재 : 다들 멀티미디어 연구원.
<주관심분야 : System IC Design,

DSP System Design>



강봉순(정회원)

1985년 2월 : 연세대학교 전자공학과(공학사). 1987년 8월 : 미국 University of Pennsylvania 전기공학과(공학석사). 1990년 8월 : 미국 Drexel University 전기 및 컴퓨터공학과(공학박사). 1989년 12월 ~ 1999년 2월 : 삼성전자 반도체 수석연구원. 1999년 3월 ~ 현재 : 동아대학교 전기전자컴퓨터공학부 조교수. <주관심분야 : VLSI Design, ASIC Design>



양훈기(정회원)

1985년 2월 : 연세대학교 전자공학과(공학사). 1987년 5월 : SUNY at Buffalo, 전기 및 컴퓨터공학과(공학석사). 1992년 5월 : SUNY at Buffalo 전기 및 컴퓨터공학과(공학박사). 1993년

3월 ~ 현재 : 광운대학교 전파공학과 교수. <주관심분야 : 무선통신시스템, UWB>