

필름 스캐너에서 역전파 신경회로망을 이용한 색 보정

Color Correction Using Back Propagation Neural Network in Film Scanner

홍 승 범*, 백 중 환*

Seung-Bum Hong, Joong-Hwan Baek

요 약

필름 스캐너는 기존의 광학 필름으로부터 고해상도와 고화질의 디지털 영상을 취득할 수 있는 입력 장치로, 최근 인쇄 및 편집 전문가들의 사용이 증대되고 있다. 그러나 광원 및 센서의 비선형적 특성으로 인해 원 필름 영상의 컬러와 스캔된 영상의 컬러가 일치하지 않는 문제가 발생한다. 따라서 필름 스캐너에서는 스캔된 디지털 영상에 대한 색 보정이 필수적이다. 본 논문에서는 스캔된 RGB 색공간의 데이터를 $L^*a^*b^*$ 색 공간으로 변환한 후 역전파 신경회로망을 이용하여 색 보정 한다. 또한 TMS320C32 DSP 칩과 고해상도 라인 센서를 사용하여 R, G, B 각각 12 비트의 색분해도와 2400 dpi 급의 해상도를 갖는 필름 스캐너로 직접 구현하여 색 보정의 검증을 하였으며, 역 전파 신경망에 적용한 결과 평균 색 보정률이 79.8%로, 기존의 다항회귀법보다 43.5% 성능이 개선된 결과이다.

Abstract

A film scanner is one of the input devices for acquiring high resolution and high quality of digital images from the existing optical film. Recently the demand of film scanners have risen for experts of image printing and editing fields. However, due to the nonlinear characteristic of light source and sensor, colors of the original film image do not correspond to the colors of the scanned image. Therefore color correction for the scanned digital image is essential in film scanner. In this paper, neural network method is applied for the color correction to CIE $L^*a^*b^*$ color model data converted from RGB color model data. Also a film scanner hardware with 12 bit color resolution for each R, G, B and 2400 dpi is implemented by using the TMS320C32 DSP chip and high resolution line sensor. An experimental result shows that the average color correction rate is 79.8%, which is an improvement of 43.5% than our previous method, polygonal regression method.

Key Words : Color Correction, Film Scanner, Neural Net, Color Model, DSP

I. 서 론

최근 멀티미디어에 대한 수요의 증가로 디지털 영상 취득 장치에 대한 관심이 날로 높아지고 있다. 그중 필름 스캐너는 기존의 광학 필름으로부터 고화질·고해상도의 디지털 영상을 취득할 수 있는 입력 장치로 인쇄 및 편집 전문가들의 사용이 증대되고 있다. 보통의 네거티브 필름 스캐너에서는 CCD 센서를 통해 취득된 아날로그 영상신호가 AD 컨버터를 통해 디지털 데이터로 변환된 후 이를 다시 반전하여 포지티브 영상 데이터로 저장된

다. 이때 광원 및 센서의 비선형적 특성으로 인해 스캔된 영상의 컬러가 원 영상의 컬러와 불일치하는 현상이 발생한다. 이를 해결하기 위해 스캔된 영상 데이터에 대한 색 보정은 필수적이다.

기존의 필름 스캐너를 이용한 색 보정 방식으로는 이장원^[2]에 의해 제안된 다항회귀(polygonal regression) 방식을 이용하여 색 보정을 수행하였다. 하지만 XYZ 컬러 모델은 기기 종속적이며, 인간의 시각 특성에 적합한 특성 때문에 전체 색 공간을 모두 나타낼 수 없는 단점을 가진다. 김태현^[3]에 의해 [2]의 단점을 해결하기 위해 CIE $L^*a^*b^*$ 컬러 모델을 적용하여 색 보정 성능을 향상시켰으나 연산량이 많은 다항회귀방식을 채택하였다.

본 논문에서는 RGB 컬러 모델을 기기 독립적인 CIE $L^*a^*b^*$ 컬러 모델을 이용하고, 색 보정 알고리즘으로 신

*한국항공대학교 전자 정보통신 컴퓨터 공학부
접수 일자 : 2003. 8. 04 수정 완료 : 2003. 10. 07
논문 번호 : 2003-4-1

경 회로망(neural networks)의 사용을 제안한다. 즉, 표준 컬러 패치 필름을 스캔한 영상 데이터에서 각 컬러의 오차가 최소가 되도록 신경회로망을 학습하여 색 보정을 실현한다. 여기서, 신경회로망으로는 역 전파(back propagation) 신경회로망을 사용한다. 전체적인 색 보정 과정을 그림 1과 같이 입력받은 RGB 데이터를 XYZ 컬러 모델로 바꾸고 이를 다시 $L^*a^*b^*$ 컬러 모델로 바꾸는 전처리 과정과 색 보정 후에 다시 RGB 컬러 모델로 바뀌주는 후처리 과정을 거치게 된다. 또한 본 논문에서 TMS320C32 DSP 칩과 고해상도의 라인 센서를 이용하여 R, G, B 각각 12 비트의 색 분해도와 2400 dpi 급의 해상도를 갖는 필름 스캐너를 직접 제작하였다.

본 논문의 구성은 다음과 같다. 2장에서는 컬러 모델 변환에 대해 설명하고, 3장에서는 신경회로망을 이용한 색 보정 알고리즘을 기술한다. 4장에서는 제작한 스캐너 시스템의 구조를 보이고, 5장에서는 실험을 통해 제안된 색 보정 알고리즘의 성능을 평가하고 기존의 방법인 다항회귀법의 성능과 비교한다. 마지막으로 6장에서 본 연구에 대한 결론을 내린다.

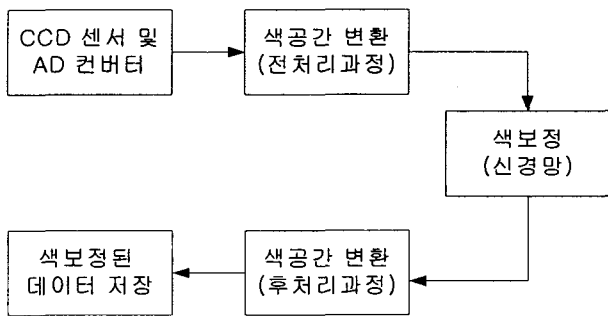


그림 1. 색 보정 과정
Figure 1. Color correlation procedure.

II. 컬러 모델 변환

컴퓨터 그래픽 등에서 가장 많이 사용하는 RGB 컬러 모델은 빛의 삼원색인 R(Red), G(Green), B(Blue)로 구성되며, 이들 컬러의 분광 요소들이 부가적으로 복합되어 결과적으로 컬러를 만들어 내는 가색 혼합 색체계이다. 즉, R, G, B의 3색광의 빛을 혼합하여 새로운 색을 재현하게 된다. 이 때, 혼합하는 빛의 혼합 강도에 따라 재현 색이 변화하고 피혼색광이 증가함에 따라 백색광에 가까워지는 특성이 있다^{[3][4]}. RGB 컬러 모델은 각 축의 모서리가 R, G, B인 3차원 입방체로 표현된다. Black은 원점, White는 입방체의 반대쪽 끝에 위치하여 두 점을 잇는 선을 따라 명암도(Gray Scale)가 표현된다.

본 논문에서 구현하는 필름 스캐너는 RGB 광원을 이용하여 필름에 순차적으로 R, G, B 광을 투사하고 필름을 투과한 광의 세기를 CCD 센서를 통해 전기적인 신호로 변환한다. 이 아날로그 전기 신호는 다시 AD 컨버터

를 통해 디지털 값으로 변환된 다음 RGB 컬러 모델로 최종 저장된다. 그러나 이러한 RGB 컬러 모델은 기기 종속적(device dependent)이어서 색 보정을 하기에는 부적합하다. 따라서 RGB 컬러 모델을 일단 XYZ 컬러 모델로 변환한 후 이를 다시 기기 독립적인 CIE $L^*a^*b^*$ 컬러 모델로 변환하는 색 보정 전처리 과정을 거친다. 변환된 $L^*a^*b^*$ 컬러 공간에서 색 보정을 수행하고 역순의 컬러 모델 변환인 색 보정 후처리 과정을 거쳐 다시 RGB 컬러 모델로 변환한다.

1. 색 보정 전처리

색 보정 전처리에서는 먼저 입력받은 RGB 색공간의 데이터를 변환공식에 대입할 수 있도록 0~1의 값으로 정규화(normalize)한다. 그리고 이 값을 아래 식 (1)에 대입해서 XYZ 컬러 모델로 변환한다.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.201 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.128 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

위 식 (1)에서 얻은 XYZ 색공간의 데이터를 아래 식 (2)에 대입해서 $L^*a^*b^*$ 컬러 모델로 변환한다^[5].

$$L^* = \begin{cases} 116 \left(\frac{Y}{Y_0} \right)^{\frac{1}{3}} - 16, & \frac{Y}{Y_0} > 0.008856 \\ 903.3 \frac{Y}{Y_0}, & 0.0 \leq \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (2)$$

$$a^* = 500 \left[f\left(\frac{X}{X_0} \right) - f\left(\frac{Y}{Y_0} \right) \right]$$

$$b^* = 200 \left[f\left(\frac{X}{X_0} \right) - f\left(\frac{Z}{Z_0} \right) \right]$$

여기서,

$$f(q) = \begin{cases} q^{\frac{1}{3}}, & q > 0.008856 \\ 7.787(q) + 0.1379, & 0.0 \leq q \leq 0.008856 \end{cases} \quad (3)$$

위 식 (2)에서 X_0, Y_0, Z_0 는 표준 백색면의 삼차극치 값이다. 즉, $R=G=B=1$ 일 때의 X, Y, Z 값이다.

2. 색 보정 후처리

색 보정을 끝낸 $L^*a^*b^*$ 색공간의 데이터를 다시 RGB 색공간의 데이터로 바꿔주기 위해서는 후처리가 필요하다. 후처리는 전처리의 반대 순서로 진행된다. 먼저 $L^*a^*b^*$ 색공간의 데이터를 아래 식(4)에 대입하여 XYZ 색공간의 데이터로 만들어준다.

$$\begin{aligned}
 X &= X_0 \left[g \left(\frac{L^* + 16}{25} \right) \right] \\
 Y &= Y_0 \left[g \left\{ \frac{a^*}{500} + f \left(\frac{Y}{Y_0} \right) \right\} \right] \\
 Z &= Z_0 \left[g \left\{ f \left(\frac{Y}{Y_0} \right) - \frac{b^*}{200} \right\} \right]
 \end{aligned} \tag{4}$$

여기서,

$$g(q) = \begin{cases} q^3, & q > 0.20681 \\ 0.1284(q - 0.1379), & 0.0 \leq q \leq 0.20689 \end{cases} \tag{5}$$

의 식 (3)에서 얻은 XYZ 색공간의 데이터를 아래 식(6)에 대입해서 RGB 컬러 모델로 변환한다.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.910 & -0.534 & -0.286 \\ -0.984 & 1.999 & -0.027 \\ 0.058 & -0.117 & 0.888 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{6}$$

의 식으로부터 얻어진 R, G, B 값은 0~255 사이의 값으로 다시 변환된다.

III. 역전파 신경회로망을 이용한 색보정

1. 신경회로망의 구조

신경회로망은 기본적으로 단순한 기능을 하는 유닛이 복잡하게 연결된 구조로 되어 있다. 신경회로망에서의 학습은 각각의 유닛이 환경에 적절히 반응하도록 그 자신의 입출력 형태를 어떠한 방법에 의하여 조정하는 것을 의미한다. 이러한 조정을 해주는 가장 보편적인 방법은 연결선에 매겨져 있는 가중치를 조정하는 것이다^[6]. 따라서 신경회로망은 각 입력에 연결된 강도를 조정함으로써 학습되며, 주어진 신경회로망의 학습법칙을 이용하여 풀고자 하는 문제의 주어진 입력에 대하여 올바른 출력을 내도록 연결강도를 어떻게 변화시킬 것인가를 결정하게 된다.

신경회로망의 구성은 셀(cells), 노드(nodes), 유닛(units)으로 불리는 많은 수의 뉴런으로 이루어지고, 각 유닛들은 특정 구조에 따라 가중치를 갖는 연결선으로 연결되어 있다. 그리고 각 유닛은 연결된 다른 유닛들로부터의 입력을 연결선의 가중치를 고려하여 더한 후 그 결과를 적당한 활성화함수(activation function)로 처리한 후 연결된 다른 유닛으로 출력한다^[6]. 신경망의 구조는 단층망과 다층망으로 나눌 수 있다. 단층망은 하나의 층으로 이루어져 있고, n개의 입력 유닛 X와 m개의 출력 유닛 Y가 가중치 w로 연결된 형태이다. n개의 층으로 이루어진 다층망은 하나의 입력 유닛층과 하나의 출력 유

닛층, 그리고 (n-1)개의 은닉층으로 구성된다. 결국, (n+1)개의 유닛층과 n개의 가중치집단을 갖는다. 본 논문에서는 n=2인 다층망을 사용한다. 즉, 1개의 은닉 유닛은 가중치 v를 통해서 n개의 입력 유닛 X를 받고, 이를 활성화함수를 통해서 다시 m개의 출력 유닛 Y에 가중치 w를 통해서 입력되게 된다. 그림 2는 2-layer일 때의 다층망의 구조를 나타낸다.

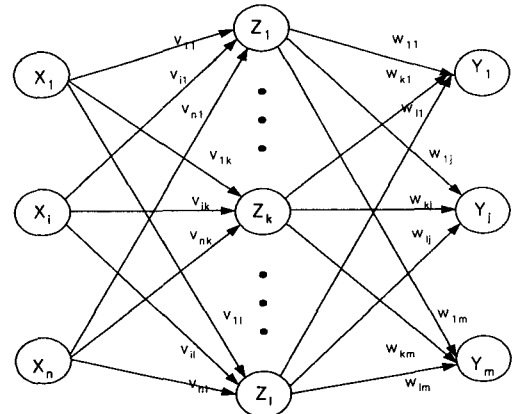


그림 2 다층망의 구조

Figure 2. Architecture of Multi-layer net

2. 활성화함수(Activation Function)

신경세포가 반응하는 조건은 외부로부터 일정 이상의 입력이 들어왔을 때 출력을 낸다. 또한 하나의 신경세포가 낼 수 있는 출력은 한정되어 있으므로 특정 값을 넘어갈 수 없다. 하나의 출력 유닛의 값은 여러 개의 입력 유닛들과 각 연결선의 가중치를 곱한 후 이 값들을 더해서 얻어지게 된다. 이 값을 활성화함수에 입력하게 되면 신경회로망 모델의 임계값을 갖는 출력을 얻게 된다. 이러한 동작은 아래와 같은 활성화함수들로 나타낼 수 있다^[7].

◆ Bipolar sigmoid

$$f(x) = \frac{1 - \exp(-\sigma x)}{1 + \exp(-\sigma x)}, \sigma > 0 \tag{7}$$

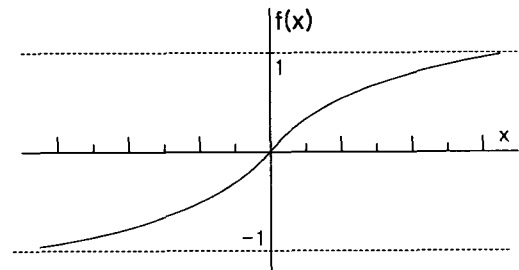


그림 3 Bipolar sigmoid

Figure 3. Bipolar sigmoid

3. Net Input

입력 유닛 X_1, X_2, \dots, X_n 과 출력 유닛 Y_1, Y_2, \dots, Y_m 이 가중치 $W = (w_{ij})$ 로 연결되어 있을 때, 유닛 Y_j 의 net input은 식 (8)과 같다.

$$y_in_j = x^T \cdot w_j = \sum_{i=1}^n x_i w_{ij} \quad (8)$$

이때 바이어스 b_j 를 추가할 때는 크기가 1인 유닛을 추가하고 이 유닛의 연결선의 가중치를 b_j 로 설정하면 된다. 바이어스가 추가된 net input은 식 (9)와 같다.

$$y_in_j = x^T \cdot w_j + b_j = \sum_{i=1}^n x_i w_{ij} \quad (9)$$

4. 역 전파 신경회로망

감독 학습 방법 중 델타 학습 법칙의 일종인 오류 역전파 규칙은 실제의 문제에 있어서 가장 널리 사용되는 학습법이다. 델타 학습 법칙의 기본은 현재 주어진 연결 가중치로 생성되는 오차값을 구하여 이를 감소시키는 방향으로 연결 가중치의 값을 조정하는 것이다. 하지만 간단한 XOR 문제도 해결하지 못하는 단점을 가진다. 오류 역전파 규칙은 이러한 문제를 해결하기 위한 방법의 일종으로 다층의 신경회로망을 학습시키는데 적합하다 [1][7][8][10].

신경회로망을 역 전파 방식으로 학습시키는 것은 크게 다음의 세 단계로 나눌 수 있다. 첫 번째는 신경회로망의 순방향으로 입력 유닛의 값을 가중치를 통해서 출력 유닛의 값을 계산한다. 두 번째는 올바른 출력값과 신경회로망을 통해서 얻은 출력값의 에러를 계산하고, 오류 역전파 알고리즘을 통해서 출력 유닛에서 발생한 오류 값을 최하단의 유닛까지 전파시켜 각 층의 가중치들의 에러를 계산한다. 세 번째는 이렇게 얻어진 값을 가지고 각 층의 가중치의 값을 갱신한다.

- Back propagation algorithm -

- Input

24개의 입력 학습패턴 : $x^{(p)} = (x_1^{(p)}, x_2^{(p)}, x_3^{(p)})$,

$1 \leq p \leq 24$

24개의 올바른 출력값 : $y^{(p)} = (y_1^{(p)}, y_2^{(p)}, y_3^{(p)})$,

$1 \leq p \leq 24$

초기 가중치:

$$v_{ij}^{(0)} ; 1 \leq i \leq 3, 1 \leq j \leq l$$

$$w_{jk}^{(0)} ; 1 \leq j \leq l, 1 \leq k \leq 3$$

허용오차: $\epsilon > 0$

최대 반복횟수: N

학습율: α

- Output

신경회로망의 최종 가중치 v_{ij} 과 w_{jk}

Step1. 가중치들을 초기화한다.

$$v_{ij} = v_{ij}^{(0)} ; 1 \leq i \leq 3, 1 \leq j \leq l$$

$$w_{jk} = w_{jk}^{(0)} ; 1 \leq j \leq l, 1 \leq k \leq 3.$$

그리고, 현재의 반복 횟수(it)를 초기화(0)한다.

Step2. $1 \leq p \leq 24$ 동안 Steps 3-4 (feed forward)와 Steps 5-7 (error back propagation)을 반복한다.

Step3. 각각의 은닉 유닛 $Z_j, 1 \leq j \leq l$ 에 대해서 입력 유닛과 가중치의 곱들의 합을 계산한다.

$$z_in_j = v_{0j} + \sum_{i=1}^n v_{ij} x_i^{(p)}, 1 \leq j \leq l$$

그리고 각 유닛에 대하여 활성화함수를 이용하여 은닉 유닛으로부터 출력신호를 얻어낸다.

$$z_j^{(p)} = f(z_in_j), 1 \leq j \leq l$$

Step4. 각각의 출력 유닛 $Y_k, 1 \leq k \leq 3$ 에 대해서 은닉 유닛과 가중치의 곱들의 합을 계산한다.

$$y_in_k = w_{0k} + \sum_{j=1}^l w_{jk} z_j^{(p)}, 1 \leq k \leq 3$$

그리고 활성화함수를 이용하여 출력신호를 얻어낸다.

$$y_k^{(p)} = f(y_in_k), 1 \leq k \leq 3$$

Step5. 최종 결과 클래스(true class, t_k)와 Step 4에서 얻은 출력 신호사이의 에러 term들을 계산한다.

$$\delta_k^{(p)} = (t_k^{(p)} - y_k^{(p)})f'(y_in_k), 1 \leq k \leq 3$$

가중치의 보정 term들은 아래와 같다.

$$\Delta w_{jk} = \alpha \delta_k^{(p)} z_j ; 1 \leq j \leq l, 1 \leq k \leq 3$$

bias의 보정 term들은 아래와 같다.

$$\Delta w_{0k} = \alpha \delta_k^{(p)} ; 1 \leq k \leq 3$$

Step6. 각 은닉 유닛에서 출력층으로부터 온 δ_k 입력들을 더한다.

$$\delta_in_j = \sum_{k=1}^m \delta_k^{(p)} w_{jk}, 1 \leq j \leq l$$

그리고 각각의 backward 에러 term을 구한다.

$$\delta'_j = \delta_in_j f'(z_in_j), 1 \leq j \leq l$$

그리고 가중치의 보정 term들을 구한다.

$$\Delta v_{ij} = \alpha \delta'_j x_i ; 1 \leq i \leq 3, 1 \leq j \leq l$$

그리고 bias의 보정 term들을 구한다.

$$\Delta v_{0j} = \alpha \delta_j^{(p)} ; 1 \leq j \leq l$$

Step7. 신경회로망의 가중치들을 갱신한다.

$$v_{ij} \leftarrow v_{ij} + \Delta v_{ij}; 1 \leq i \leq 3, 1 \leq j \leq l$$

$$w_{jk} \leftarrow w_{jk} + \Delta w_{jk}; 1 \leq j \leq l, 1 \leq k \leq 3$$

$$\Delta E = \frac{1}{3N} \sum_{k=L^*, a^*, b^*} \sum_{n=1}^N |k_m - k_n|, \quad (10)$$

Step8. 반복횟수를 증가한 후($it \leftarrow it+1$), 현재의 평균 제곱 오차를 구한다.

$$e = \sum_{p=1}^{24} \sum_{k=1}^3 (t_k^{(p)} - y_k^{(p)})^2$$

이때 $it \leq N$ 이고 $e > \epsilon$ 일 경우는 Step 2로 가고, $it \leq N$ 이고 $e < \epsilon$ 일 경우는 가중치 v_{ij} , w_{jk} 와 반복횟수 it 를 출력하고, 알고리즘을 끝낸다. 만약, $it > N$ 일 경우는 '수렴불가'를 출력하고 알고리즘을 끝낸다.

위의 알고리즘에서 가중치의 초기치는 -0.5 ~ 0.5 사이의 임의의 수를 정해서 사용한다.

본 논문에서 색 보정에 위의 방법을 사용할 때는 입력 X 로 스캔 받은 RGB 데이터를 전처리를 통해서 $L^*a^*b^*$ 로 변환한 데이터를 사용했다. 위의 알고리즘은 bipolar sigmoid를 사용했으며, 입력 유닛의 입력값의 입력범위 또한 -1 ~ 1 사이의 값을 사용하게 된다. 그러므로 0 ~ 100 사이의 L^* 값과 -128 ~ 127 사이의 a^* 와 b^* 값을 -1 ~ 1 사이의 값으로 정규화 시켜서 사용하게 된다.

입력값으로 L^*_i, a^*_i, b^*_i 세 개의 값을 사용하므로 입력 유닛의 수는 3개이며, 출력값으로 L^*_o, a^*_o, b^*_o 세 개의 값을 사용하므로 출력 유닛 또한 3개가 된다.

IV. 색보정 실험

본 논문에서 제안한 색 보정 알고리즘의 성능을 실험하기 위해 Macbeth Color Checker 24를 이용하였다. Macbeth Color Checker 24는 인간의 시각에 가장 민감하게 반응하는 자연 물체(Natural Objects)의 18가지 색과 6단계의 명암도, 총 24가지의 색으로 구성되어 있는 표준 컬러 영상을 이용하였다^[3]. 이 컬러 패치를 이용해 네거티브 필름을 만들고, 이 필름으로 제작된 스캐너로 취득한 RGB 데이터를 $L^*a^*b^*$ 컬러 모델 데이터로 변환하였다. 24개의 기준 $L^*a^*b^*$ 값과 제작한 스캐너를 통해서 얻은 RGB 데이터를 전처리를 통해서 $L^*a^*b^*$ 로 변환된 데이터는 표 1과 같다.

본 논문에서 컬러 오차 값은 L^*, a^*, b^* 각각에 대한 절대 오차 값의 합의 평균으로 전체 평균 오차 값을 정의하고 아래의 식으로 표현된다.

여기서 k_m 은 각 컬러의 기준 L^*, a^*, b^* 값이고, k_n 은 스캔한 컬러의 L^*, a^*, b^* 값이다. 또한 N 은 24이다. 24개의 컬러에 대한 각각의 오차 값과 전체 평균 오차 값은 표 2와 같다.

표 1. 24 컬러의 기준 $L^*a^*b^*$ 값과 스캔된 $L^*a^*b^*$ 값

Table 1. Standard $L^*a^*b^*$ values of the 24 colors.

컬러	$L^*a^*b^*$ 값			스캔된 $L^*a^*b^*$ 값		
	L^*	a^*	b^*	L^*	a^*	b^*
No.1 dark skin	23	34	29	17.12	20.40	15.81
No.2 light skin	73	20	25	60.65	19.33	13.59
No.3 blue sky	82	-1	-27	76.74	1.66	-13.49
No.4 foliage	63	-30	51	48.82	-22.98	37.18
No.5 blue flower	59	27	-61	56.61	4.76	28.85
No.6 bluish green	94	-28	1	70.83	-25.83	9.50
No.7 orange	63	41	65	57.75	28.17	72.46
No.8 purplish blue	54	20	-63	46.39	-1.13	-31.62
No.9 moderate red	62	58	46	60.06	35.04	20.79
No.10 purple	44	45	-71	40.54	17.09	-32.11
No.11 yellow green	92	-36	50	73.38	-18.51	29.43
No.12 orange yellow	86	8	41	77.62	12.86	33.31
No.13 blue	20	36	-68	16.31	22.06	-49.83
No.14 green	60	-52	56	38.59	-49.86	34.82
No.15 red	50	74	65	37.92	50.86	41.71
No.16 yellow	95	-11	86	79.59	8.36	79.84
No.17 magenta	56	78	-43	47.11	56.54	-23.99
No.18 cyan	75	-27	-37	44.69	-43.73	-33.94
No.19 white	99	-2	-1	91.05	0.04	-1.21
No.20 gray 1	91	-13	-6	75.20	-0.62	0.44
No.21 gray 2	73	-2	-7	62.72	-0.64	-0.08
No.22 gray 3	40	7	10	43.42	-1.00	-0.61
No.23 gray 4	20	7	15	26.89	-0.16	0.40
No.24 black	3	3	5	5.11	0.26	-2.68

표 2. 색 보정 전의 오차 값

Table 2. Error values before color correction

컬러 \ 오차 값	L^*	a^*	b^*
No.1 dark skin	5.88	13.60	13.19
No.2 light skin	12.35	0.67	11.41
No.3 blue sky	5.26	2.66	13.51
No.4 foliage	14.18	7.02	13.82
No.5 blue flower	2.39	22.24	32.15
No.6 bluish green	23.17	2.17	8.50
No.7 orange	5.25	12.83	7.46
No.8 purplish blue	7.61	21.13	31.38
No.9 moderate red	1.94	22.96	25.21
No.10 purple	3.46	27.91	38.89
No.11 yellow green	18.62	17.49	20.57
No.12 orange yellow	8.38	4.86	7.69
No.13 blue	3.69	13.94	18.17
No.14 green	21.41	2.14	21.18
No.15 red	12.08	23.14	23.29
No.16 yellow	15.41	19.36	6.16
No.17 magenta	8.89	21.46	19.01
No.18 cyan	30.31	16.73	3.06
No.19 white	7.95	2.04	0.21
No.20 gray 1	15.80	12.38	6.44
No.21 gray 2	10.28	1.36	6.92
No.22 gray 3	3.42	8.00	10.61
No.23 gray 4	6.89	7.16	14.60
No.24 black	2.11	2.74	7.68
$L^*a^*b^*$ 전체 평균 오차	L^* 평균 오차	a^* 평균 오차	b^* 평균 오차
12.41	10.28	11.91	15.05

색 보정으로 사용할 신경회로망 모델은 스캐너를 통해서 얻은 입력 데이터 L^* , a^* , b^* 를 세 개의 입력 유닛으로 갖고, 출력 유닛 또한 L^* , a^* , b^* 에 해당하는 세 개를 갖는다. 입력 유닛과 출력 유닛은 -1 ~ 1 사이의 값으로 정규화 시켜서 사용한다. 그리고 역전파 신경회로망 프로그램은 입력 데이터와 테스트 영상의 데이터를 입력 받아서 최소 오차를 결정하기 위해 반복회수, 은닉 유닛의 개수, 그리고 허용오차 등을 바꾸어가며 결정하였다.

본 논문에서는 허용오차(ϵ)를 0.05로 설정하고, 반복회수(N)를 150000으로 설정하였다. 그리고 은닉 유닛의 수는 56개로 선정하였다.

표 3 색 보정 후의 $L^*a^*b^*$ 값

Table 4. $L^*a^*b^*$ values after color correction.

컬러 \ $L^*a^*b^*$ 값	L^*	a^*	b^*
No.1 dark skin	17	19	14
No.2 light skin	67	19	9
No.3 blue sky	71	0	-11
No.4 foliage	51	-28	41
No.5 blue flower	60	1	-30
No.6 bluish green	72	-26	9
No.7 orange	57	30	73
No.8 purplish blue	50	0	-33
No.9 moderate red	55	39	28
No.10 purple	35	20	-33
No.11 yellow green	73	-18	31
No.12 orange yellow	77	8	29
No.13 blue	16	20	-46
No.14 green	37	-47	30
No.15 red	40	46	37
No.16 yellow	81	10	82
No.17 magenta	49	56	-23
No.18 cyan	45	-40	-32
No.19 white	85	6	0
No.20 gray 1	76	-7	0
No.21 gray 2	64	1	-6
No.22 gray 3	41	-1	4
No.23 gray 4	25	1	1
No.24 black	8	1	-5

신경망을 색 보정에 적용하는 것은 변환 문제로서 기존 신경망이 패턴 인식이나 분류에 쓰이는 것에 비해서 많은 복잡도(complexity)와 계산 시간이 소요된다. 그러나 일단 적정한 해로 학습이 된 후에는 새로운 데이터에 대해서는 빠른 색 보정을 수행한다.

오류 역전파 신경회로망 방법을 이용해서 색 보정을 수행한 후의 L^* , a^* , b^* 데이터는 표 4와 같다. 표 1의 테스트 영상의 데이터와 표 3에서 얻은 보정된 데이터간의 각각의 오차 값과 전체 평균 오차값은 표 4와 같다.

표 2 과 표 4의 결과를 비교해서 살펴보면, 색 보정 과정을 거치지 않은 영상의 경우는 테스트 영상의 데이터 값과 비교해서 L^* 은 10.28, a^* 는 11.91, b^* 는 15.05, 그리고 $L^*a^*b^*$ 평균 12.41의 오차값을 보이며, [2]의 다항회귀 방법을 이용한 색 보정을 수행한 영상은 테스트 영상의 데이터 값과 비교해서 L^* 은 6.8, a^* 는 8.70, b^* 는 8.215,

그리고 $L^*a^*b^*$ 평균 7.907의 오차값을 보인다.

표 4. 색 보정 후의 오차 값

Table 4. Error values after color correction.

번	색상	L^*	a^*	b^*
No.1	dark skin	0.1200	1.4000	1.8100
No.2	light skin	6.3500	0.3300	4.5900
No.3	blue sky	5.7400	1.6600	2.4900
No.4	foliage	2.1800	5.0200	3.8200
No.5	blue flower	3.3900	3.7600	1.1500
No.6	bluish green	1.1700	0.1700	0.5000
No.7	orange	0.7500	1.8300	0.5400
No.8	purplish blue	3.6100	1.1300	1.3800
No.9	moderate red	5.0600	3.9600	7.2100
No.10	purple	5.5400	2.9100	0.8900
No.11	yellow green	0.3800	0.5100	1.5700
No.12	orange yellow	0.6200	4.8600	4.3100
No.13	blue	0.3100	2.0600	3.8300
No.14	green	1.5900	2.8600	4.8200
No.15	red	2.0800	4.8600	4.7100
No.16	yellow	1.4100	1.6400	2.1600
No.17	magenta	1.8900	0.5400	0.9900
No.18	cyan	0.3100	3.7300	1.9400
No.19	white	6.0500	5.9600	1.2100
No.20	gray 1	0.8000	6.3800	0.4400
No.21	gray 2	1.2800	1.6400	5.9200
No.22	gray 3	2.4200	0	4.6100
No.23	gray 4	1.8900	1.1600	0.6000
No.24	black	2.8900	0.7400	2.3200
$L^*a^*b^*$ 전체 평균 오차		2.4096	2.4629	2.6588

그리고 역 전파 신경회로망 방법을 이용해서 색 보정을 수행한 영상은 테스트 영상의 데이터 값과 비교해서 L^* 은 2.4096, a^* 는 2.4629, b^* 는 2.6588, 그리고 $L^*a^*b^*$ 평균 2.5104의 오차 값을 보인다. 그림 4와 표 5의 결과를 살펴보면, 색 보정을 수행한 영상의 오차가 색 보정 과정을 거치지 않은 영상의 경우보다 오차가 줄어들었음을 알 수 있고, 색 보정을 수행한 두 가지의 결과 중에서 다항회귀 방법^[2]을 사용했을 때의 오차보다 역 전파 신경회로망 방법을 사용했을 때의 오차가 훨씬 적음을 알 수 있다. 역 전파 신경회로망 방법을 이용해서 색 보정을 수행

한 결과는 테스트 영상의 데이터와 거의 비슷함을 알 수 있다.

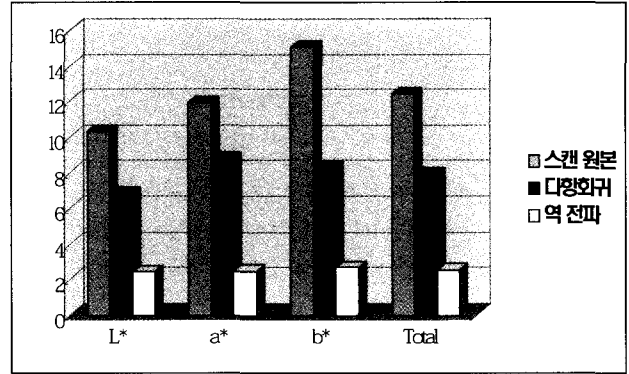


그림 4 컬러 오차값 비교 그래프 (Color Error Comparison Graph)

표 5. 컬러 오차값 비교

Table 5. Color Error Comparison

방법	평균 오차값	L^*	a^*	b^*	Total
보정 전		10.28	11.91	15.05	12.41
다항회귀(보정 후)		6.8	8.70	8.215	7.9
역 전파 (보정 후)		2.41	2.46	2.66	2.51
다항회귀의 보정율		33.9%	27.0%	45.4%	36.3%
역전파의 보정율		76.6%	79.3%	82.3%	79.8%

VI. 결론

본 논문에서는 고해상도와 고화질의 영상 데이터를 얻을 수 있는 필름 스캐너의 하드웨어를 범용 DSP 칩인 TMS320C32를 메인 컨트롤러로 채택하여 직접 제작하고, 색 재현 문제를 해결하기 위해 오류 역 전파 신경회로망 방법을 사용하여 색 보정을 수행하였다. Macbeth Color Checker 24색 보정용 테스트 필름을 사용하여 직접 제작한 스캐너에서 이 필름을 스캔해서 얻은 RGB 데이터를 색 보정에 적합한 $L^*a^*b^*$ 데이터로 변환하여 입력 데이터로 사용하고, 제작사에서 제공해주는 테스트 영상의 $L^*a^*b^*$ 값을 기준 값으로 사용하여 두 색 공간에서의 색 보정을 수행하였다. 본 논문에서 실험한 결과 색 보정을 하지 않은 데이터의 $L^*a^*b^*$ 평균오차는 12.414, 본 논문에서 제안한 역 전파 신경회로망 방법을 이용해서 색 보정을 한 후의 데이터의 평균오차는 2.51로 전체 평균 79.8%의 색 보정을 얻을 수 있었으며, 다항회귀 방식에 비하여 43%이상의 색 보정 능력이 있음을 확인하였다. 여기서 색 보정을 했을 때 오차가 줄어드는 것을 확인할 수 있었고, 다항회귀 방법에 비해서 제안한 역 전파 신경

회로망 방법이 더욱 강력한 색 보정 성능을 나타냄을 알 수 있었다.

참 고 문 헌

- [1] Maeng-Sub Cho, Byoung-Ho Kang and Luo, M.R., "Device Calibration of a Color Image Scanner Digitizing System by Using Neural Networks," Proceedings of IEEE International Conference on Neural Networks, Vol. 1, pp.59-62, 1995.
- [2] 김태현, 백중환, "다항회귀를 이용한 필름 스캐너에서의 색 보정", 대한전자공학회논문지 제40권 SP편 제 1호, pp.43-50, 2003.
- [3] 이 장원, "비디오 모니터를 이용한 사진 현상용 확대기의 구현", 한국항공대학교 석사학위논문, 2000
- [4] Charles A. Poynton, "A Guided Tour of Color Space," Proceedings of SMPTE Advanced Television and Electronic Imaging Conference, p.167-180, 1995.
- [5] Randy Crade, A Simplified approach to Image Processing, Prentice-Hall, pp.18-29, 1997.
- [6] William K. Pratt, Digital Image Processing, 3rd Edition, John Wiley & Sons, Inc., 2001.
- [7] Menahem Friedman and Abraham Kandel, Introduction to Pattern Recognition, World Scientific, pp.255-308, 1999.
- [8] Laurene Fausett, Fundamentals of Neural Networks, Prentice-Hall, 1994, pp.1-26, 289-316
- [9] Maureen Caudill and Charles Butler, Understanding Neural Networks, Massachusetts Institute of Technology, 1992, pp.172-179
- [10] M. J. Vrhel and H. J. Trussell, "Color Scanner Calibration via a Neural Network", IEEE International Conference on V.6, 1999, pp.3465-3468



홍 승 범(Seung-Bum Hong)

準會員

1995년 2월 : 한국항공대학교 항공통신정보공학과 졸업

1997년 2월 : 한국항공대학교 항공통신정보공학과 석사

2003년 8월 : 한국항공대학교 항공통신 정보공학과 박사

2001년 12월 ~ 현재 : 이레스페이스 연구원, 현 한서대학교 항공전자시뮬레이션학과 겸임교수

관심분야 : 컴퓨터 비전, 패턴 인식, 항공기 시뮬레이터



백 중 환(Joong-Hwan Baek)

正會員

1981년 2월 : 한국항공대학교 항공통신정보공학과 졸업

1987년 7월 : 미국 오클라호마주립대학교 전자공학과 석사

1991년 7월 : 미국 오클라호마주립대학교 전자공학과 박사

1992년 3월 ~ 현재 : 한국항공대학교 전자·정보통신·컴퓨터공학부 교수

관심분야 : 컴퓨터 비전 및 패턴 인식, 영상 처리 및 압축