

전자문서용 색각 장애 보정 소프트웨어 개발

장 영 건[†]

요 약

본 연구는 컴퓨터에서 칼라 전자문서를 사용할 때 나타나는 색맹 및 색약 장애자의 혼란을 경감시키기 위하여 컴퓨터 디스플레이에 적용되는 보조 기술에 관한 것이다. 본 연구에서는 보조 기술을 마이크로소프트의 윈도우즈를 256색 모드로 사용할 때로 제한하였으며, 여러 개의 창을 열었을 때 나타나는 색의 왜곡을 줄이는 방식을 사용하였다. 색맹 및 색약 장애자용 팔레트는 christine이 표준으로 제안한 웹에서 안전한 216색상의 팔레트를 기준으로 모든 전자 문서에 적용할 수 있도록 256색 팔레트로 확장하여 구현하였다. 구현된 색각 보정 소프트웨어를 직록 백맹 시뮬레이터를 사용하여 시험한 결과 색각 보정 효과가 있음이 검증되었다. 구현된 색각 보정 S/W는 전자문서의 설계과정에서 사용하면 좀 더 큰 색각 보정 효과를 거둘 수 있다.

Development of Assistive Software for color blind to Electronic Documents

Jang Young Gun[†]

ABSTRACT

This study is concerned with an assistive technology which reduces color blinds's confusion when they access electronic documents including color objects in their computers. In this study, I restrict the assistive technology would apply to windows operating system, 256 color mode and implement to minimize color distortion which occurs in multi window environments because of color approximation process. As a basic palette, I use a 216 colors web safe palette which the Christine proposed as a standard for color blind, expand it to 256 colors to apply all computer displays using Microsoft Windows as its operating system and implement it as windows application. To test its effectiveness, I use a simulator for dichromats, as results of the test, the developed color vision deficiency correction S/W is effective to reduce the confusion. It is more effective to use the implemented S/W in both of design and client process for electronic documents.

키워드 : 색맹 및 색약(Color Blind), 보조 기술(Assistive Technology), 256색 팔레트(256 Color Palette), 색각 보정(Color Vision Deficiency Correction), 전자문서(Electronic Document)

1. 서 론

컴퓨터의 보급의 증가와 이에 따른 사용자의 증가로 인해 교육, 게임, 업무 등 모든 사회활동에 컴퓨터가 활용되고 있으며, 마이크로소프트사에서 개발한 윈도우즈 환경과 X-window 등에서 지원하고 있는 GUI 환경의 등장과 인터넷에서 하이퍼링크 기능과 멀티미디어를 지원하는 WWW의 출현으로 과거 텍스트모드에서 컴퓨터를 활용하던 시기보다 다양한 색상을 갖는 환경에서 컴퓨터를 활용하게 되었다. 문서에서도 전자문서의 비중이 점점 증가하고 있다. 색상은 단순히 지장의 용도로도 쓰이지만 때때로 그 정보를 어떻게 해석하고, 사용해야 하는지에 대한 중요한 정보를 전달하는 중요한 수단이 되고 있으며, 시각적으로 사용자의 시선을 집중시키거나 사용자의 활용에 있어 편리한

아이콘을 제공하는 등 색상을 통한 정보 제공의 중요성이 증가하고 있다. 이에 따라 색각에 이상이 있는 전자문서 사용자의 경우 예전보다는 더욱 문서의 접근에 어려움을 겪고 있다.

색각 이상이란 색상의 식별능력이 없는 상태, 즉 우리가 흔히 알고 있는 색약이나 색맹을 말한다. 이는 추상체 종류 중 하나 이상이 정상 수준 이하이거나 전혀 기능하지 않는 경우에 발생한다. 색상 지각에 영향을 미치는 유전자는 X 염색체의 일부이기 때문에 색 기능 장애는 여성들보다 남성들에게 훨씬 많은 영향을 미친다. 한국의 경우 남자는 4.16 [1]~5.9[2]%, 여자는 0.31[1]~0.76[3]가 색상 지각에 장애가 있는 색각 이상자로 남자의 경우 100만명이 넘는 사람이 색각 이상을 갖고 있지만 전자 문서나 웹 문서의 저작자가 이들을 의식하여 문서를 작성하거나 설계하는 경우가 거의 없다. 세계적으로는 인구 12명 당 1명이 색각 이상을 갖고 있는 것으로 발표된 바 있다[4,5]. 색각 이상은 칼라

[†] 성희원 : 청주대학교 컴퓨터정보공학과 교수
논문접수 : 2003년 5월 21일, 심사완료 : 2003년 8월 21일

전자문서의 비중이 급증하고 있고, 특히 웹 문서의 접근이 일반화되고 있는 시점에서 문서의 일부분이 잘못 인식될 수 있어서, 문서 접근의 오류 및 오인식이 커다란 문제를 야기시킬 수 있다.

미국에서는 재활현장의 섹션 508에서 표준으로 웹 기반 정보와 응용에서 색상만으로 정보를 전달하지 말 것을 규정하고 있으며[6], W3C에서는 WCAG 1.0 Checkpoint 2.1로써 동일한 내용을 규정하고, 그 준수 여부를 검사할 수 있는 소프트웨어를 무상으로 제공하고 있으며, 준수 정도에 평가하여 등급을 표시하는 로고를 해당 웹사이트에 부여하고 있다[7]. 마이크로소프트는 윈도우즈에서 동작하는 색각 이상자를 위한 프로그램을 발표하였는데, 이 소프트웨어는 배경과 내용에 사용된 색상을 흑백으로 만들거나 고대비로 변환하여 원래의 색 정보가 무시된다[8]. 따라서 문자 정보에 대한 최소한의 접근 능력을 제공하지만 그래픽, 영상 정보 등은 심하게 왜곡되는 문제가 발생한다. 그럼에도 불구하고, 문자 정보에 있어서는 배경과 고대비로 색상이 지정됨으로써 배경과의 확실한 구분이 가능하다. 색각 이상자에 대한 컴퓨터 상의 고려와 관련된 연구로는 영국의 Christine 이 웹에서 사용되는 표준 팔레트 색상에 대하여 제 1색맹, 제 2색맹과 제 3색맹을 모사하기 위한 216개의 팔레트 색상을 정의하였다[9]. 이 연구는 웹 상에서 색각 장애인이 보는 화면을 정상인이 검사해 볼 수 있어 웹 설계자가 자신이 만든 웹 페이지가 색각 장애인에게 문제를 일으킬 수 있는 지를 검사하는데 유용하다. 또한 2000년 7월 웹에서 적록 색맹에게 적용할 수 있는 216개의 팔레트 색상을 표준으로 제안하였다[5]. 그러나 이 팔레트는 웹 저작자가 권고 수칙에 따라 웹 페이지에 작성하고, 웹 사용자가 같은 팔레트를 장착하고 있는 경우에는 효과가 크지만 그것과 관계없이 작성된 웹 문서의 경우에는 효과가 경감되며, 윈도우즈에 적용할 경우 여러 창을 활성화할 경우에 일관성을 유지하는 소프트웨어가 필요하며, 특히 그래픽이 근사화 때문에 왜곡되는 문제가 나타나는 것으로 본인의 실험 결과 밝혀졌다. 1996년 미국의 Holly G. Atkinson 등은 컴퓨터 상에서 색각 이상 검사를 하여 색각 이상이 있으면 대응되는 색상 팔레트로 해당 응용 프로그램에서 제공되는 팔레트를 변환하는 방식과 보정 시스템을 그들의 특허에서 제시하였다[10]. 그러나 이 특허에서는 어떤 방식으로 색상 팔레트를 변환하는지에 대한 설명은 없다. 미국의 Donald P. Greenberg는 1988년 색맹이 느끼는 색상에 대한 연구를 수행하여 적록 색맹 및 청색맹에 대하여 그들이 인지하는 색의 스펙트럼과 실제 색상들의 관계를 이론적으로 정립하였으며, 실제 색각 이상자에 대하여 실험을 수행하여 그 타당성을 입증하였다[11]. 이 연구는 이후 색각 이상자의 시각을 모사하는 Hans Brettel의 연구[12]와 같은 여러 연구와 구현에서 그 근거가 되고 있으며, 정상인이 색각 이상자의 시각 현상

을 이해하고, 평가하는데 큰 기여를 하였다. 상업적으로는 영국의 British Telecommunication의 디비전인 BtExact Technologies에서 웹 디자인에 관련한 상업적인 색맹 관련 서비스와 컨설팅을 업무로 하고 있고[13], 미국에서는 스탠포드 대학의 벤처기업인 Vischeck에서 TV 등 비디오 현시장치에 대한 컨설팅 및 특허 판매를 하고 있으며, 대상 화면을 분석하여 보정을 행하는 것이 특징이다[14]. 이시하타 시험을 통과하지는 못하지만 구분이 가지 않는 그림을 구별 가능하게 한다. 그러나 분석에 많은 시간이 소요되어 웹 서핑 같은 작업에는 적합하지 않다.

국내에서는 1998년 보건복지부의 지원으로 서한전산화 색각 검사 시스템이라는 컴퓨터를 이용한 색각 검사에 대한 연구[15]가 있으나 결과로써 나타나는 수치들이 전문화된 용어에 대한 해석 수치로써 일반인이 해석할 수 있는 범주에 속하지는 않는다. 이 시스템은 기초검사용(40색 단추모델)과 전문가용(85색 단추모델)이 있으며 모니터에 나타나 있는 여러 개의 색 단추를 유사한 색 순서대로 끼워 넣기만 하면 이상 여부 및 이상 정도를 바로 알 수 있다. 컴퓨터 또는 통신 단말기에서의 색각 보정에 대한 연구는 2002년 청주대에서 발표한 “색각 장애자를 위한 색각 보정 시스템의 설계 및 구현”이 유일하지만 색상을 어떻게 변환하는지에 대한 제시가 구체적으로 나타나지 않았다[16].

본 연구에서는 세계적으로 유일하게 웹 문서에 대한 색맹 보정 방식을 구체적으로 제시한 크리스틴의 연구 결과를 이용하여 모든 전자문서로 확장하는 방법을 제시하고, 구현할 것이다. 구현 환경은 윈도우즈 운영체제를 사용하는 개인용 컴퓨터로 한정하였고, 구현의 타당성과 적용성을 평가하기 위하여 vischeck의 색맹 모사 소프트웨어를 사용한 모의 실험을 수행하여 그 타당성을 검증할 것이다.

2. 색각 이상의 현상과 컴퓨터에서의 색맹 모사

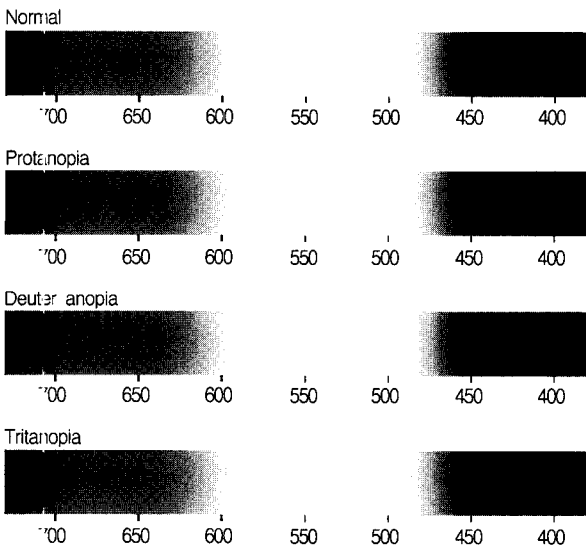
눈에는 빛을 받아 색과 형태를 느끼는 기능을 하는 것이 망막이며, 망막에 있는 신경세포에서 색을 느끼는 기능을 하는 세포는 추상체와 간상체가 있다. 추상체는 빛이 강할 때 주로 작용하며, 간상체는 어둠 속에서 색을 인식하는 데 사용되는 것으로 알려져 있다. 추상체는 파장에 따라 흡수도가 다르며, 단파(419nm), 중파(513nm), 장파(558nm)의 3가지 파장에 대하여 반응하며, 각각 S, M, L 추상체라 부르고 해당 파장에서 흡수도가 제일 크다. 이 추상체 중에 일부가 없거나 장애가 있는 사람을 보통 색맹이나 색약이라고 하며, 구체적인 색각 결합의 종류, 형태 및 원인을 <표 1>에 표시하였다.

정상인과 제 1색맹, 제 2색맹, 제 3색맹이 가시광선의 스펙트럼에 대하여 느끼는 색상을 (그림 1)에 표시하였다. 제 1색맹과 제 2색맹은 붉은 색에 대해서는 황갈색으로 유사

하게 인식하며 녹색은 각각 황색과 옅은 황색으로 인식하여 유사한 색감을 갖는 것을 알 수 있다. 색각 이상의 종류에 따른 성별 발생률을 <표 2>에 표시하였다. 색각 이상은 특히 남자에게서 유전적으로 많이 발생하며, 백인 남자는 약 8%, 동양인 남자는 약 6%, 아프리카 남자는 약 4% 정도가 발생한다.

<표 1> 색각 결함

종 류	형 태	원 인
적록색 결함		
지 1색약(적색약)	3색형 색각	L 추상체 장애
지 1색맹(적색맹)	2색형 색각	L 추상체 없음
지 2색약(녹색약)	3색형 색각	M 추상체 장애
지 2색맹(녹색맹)	2색형 색각	M 추상체 없음
청황색 결함		
지 3색맹(청색맹)	2색형 색각	S 추상체 없음



(그림 1) 정상인과 색맹의 가시광선에 대한 색각

색맹을 근본적으로 치료하는 방법은 없으며, 보정 방법으로 영국의 데이비드 해리스가 개발한 크로마젠(Chroma Gen) 렌즈나 색각 보정 안경과 같은 보조 기구가 사용될 수 있으며, 약간의 보정 효과를 갖는 것으로 알려져 있다[17].

<표 2> 색각 이상의 종류에 따른 성별 발생률

종 류	발 생 율(%)	
	남 자	여 자
제 1색약	1	0.02
제 1색맹	1	0.02
제 2색약	5	0.4
제 2색맹	1	0.01
제 3색맹	0.005	0.005
전 색맹	0.005	0.005

색은 빛의 3원색인 적색(R), 녹색(G), 청색(B)의 3가지 광선을 조합하여 받아들여 인식된다. 각각의 R, G, B는 식(1)과 같이 표현할 수 있으며, $E(\lambda)$ 는 각 광선의 임의적 스펙트럼의 에너지 분포이고 $\hat{r}, \hat{g}, \hat{b}$ 는 각각의 대응되는 색매칭 함수이다.

$$\begin{aligned} R &= \int E(\lambda) \hat{r}(\lambda) d\lambda \\ G &= \int E(\lambda) \hat{g}(\lambda) d\lambda \\ B &= \int E(\lambda) \hat{b}(\lambda) d\lambda \end{aligned} \quad (1)$$

일반적으로 색을 지정할 때 가장 보편적으로 통용되는 1931 CIE XYZ 좌표계로 색을 표현하면 식 (2)와 같으며 $\hat{x}(\lambda), \hat{y}(\lambda), \hat{z}(\lambda)$ 는 관찰자 매칭 함수이다. 이때 $X + Y + Z = 1$ 이 성립한다.

$$\begin{aligned} X &= \int E(\lambda) \hat{x}(\lambda) d\lambda \\ Y &= \int E(\lambda) \hat{y}(\lambda) d\lambda \\ Z &= \int E(\lambda) \hat{z}(\lambda) d\lambda \end{aligned} \quad (2)$$

R, G, B와 CIE XYZ간의 관계는 선형적이며, 변환 소프트웨어를 사용하여 CRT에 주어진 R, G, B 자극에 대한 XYZ를 계산할 수 있다. 인간의 스펙트럼 민감도 함수는 빛의 개별적 파장에 대한 시각 계의 신체적 반응을 특성화한 것이다. 신체물리학적 관점에서 스펙트럼 민감도 함수는 이미 언급한 색 매칭 함수의 선형적 조합이어야 한다. 스펙트럼 민감도 함수에 대한 색 매칭 함수의 선형적 변환은 색각 이상을 가진 사람에 대한 색 매칭 실험의 결과로부터 도출될 수 있다. 단, 중, 장파에 대한 스펙트럼 민감도 함수를 $\hat{s}(\lambda), \hat{m}(\lambda), \hat{l}(\lambda)$ 이라 하면 각 파장의 자극치 S, M, L은 식 (5)에 의하여 계산될 수 있다. 색맹은 보통 3개의 기본 스펙트럼 민감도 함수 중에서 2개만을 가진 사람을 말한다. 즉 적색맹에 있어서 색 공간은 SM 평면이 되고, 색도 다이어그램은 $S + M = 1$ 인 선이 된다.

$$\begin{aligned} S &= \int E(\lambda) \hat{s}(\lambda) d\lambda \\ M &= \int E(\lambda) \hat{m}(\lambda) d\lambda \\ L &= \int E(\lambda) \hat{l}(\lambda) d\lambda \end{aligned} \quad (5)$$

$$\begin{aligned} s &= \frac{S}{S+M+L} \\ m &= \frac{M}{S+M+L} \end{aligned} \quad (6)$$

CIE XYZ 공간에서 색도 혼란 점들이 색도 좌표로 주어지면 CIE XYZ 공간으로부터 SML 공간으로 변환하는 것이 유도될 수 있다. 이 변환 식은 식 (7)과 같다. x_p, y_p, x_d, y_d, x_l 와 y_l 는 혼란 점들이고, k_p, k_d 와 k_l 는 표준화 요소다.

식 (8)에 표시한 값들은 Estevez[18]에 의하여 제안된 값들이다.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_t & x_d & x_p \\ y_t & y_d & y_p \\ 1-x_t-y_t & 1-x_d-y_d & 1-x_p-y_p \end{bmatrix} \begin{bmatrix} k_t S \\ k_d M \\ k_p L \end{bmatrix} \quad (7)$$

$$\begin{matrix} x_p = 0.735 & x_d = 1.14 & x_t = 0.171 \\ y_p = 0.265 & y_d = 0.14 & y_t = -0.003 \end{matrix} \quad (8)$$

이 값들을 식 (7)에 대입하여 S, M, L에 대한 X, Y, Z의 관계를 도출하면 식 (9)와 같은 관계를 얻을 수 있다.

$$\begin{bmatrix} S \\ M \\ L \end{bmatrix} = \begin{bmatrix} 0.0000 & 0.0000 & 0.5609 \\ -0.4227 & 1.1723 & 0.0911 \\ 0.1150 & 0.9364 & -0.0203 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (9)$$

따라서 식 (9)를 사용하여 CIE XYZ로 주어진 색 자극에 대한 이색각자의 망막 좌표 SML을 계산할 수 있다.

다른 방식으로는 망막 좌표계인 LMS 공간에 색맹의 특성을 규정하고, 원래의 주어진 영상의 R, G, B 값을 대입하여 CRT 화면에 나타나는 규정된 색맹을 모사하는 R, G, B 값을 계산하는 Brettel의 연구 결과가 1997년에 발표된 바 있다[12]. 그의 연구는 기하학적인 해석 방식을 도입한 것이 특징이다.

3. 마이크로소프트 윈도우에서의 팔레트 처리

비디오 카드는 보통 아날로그 방식의 CRT 모니터를 필요로 한다. 때문에 디스플레이 어댑터 내부에서는 컴퓨터의 디지털 신호를 아날로그 방식의 비디오 신호로 변환시키는 장치가 내장되어 있는데, 이것을 DAC(Digital Analog Converter)라고 부른다. 이 DAC에는 색참조 테이블이라고 하는 램 메모리가 내장되어 있으며, 팔레트란 이 색참조 테이블을 말한다. 화면에 표시하고자 하는 데이터는 비디오 램이라는 프레임 버퍼에 저장된다. 이 값은 단순한 인덱스 값일 뿐, 자체가 색을 의미하지는 않는다. 실제 색에 관한 정보는 DAC의 색 참조 테이블에 저장된다. 비디오 램의 인덱스 값에 해당하는 색 참조 테이블의 색 정보가 바로 실제 색이 된다. 팔레트를 사용하는 이유는 데이터의 입출력 양을 줄이고 동시에 디스플레이를 위한 메모리를 줄이기 위한 것이다. 윈도우 시스템은 (그림 2)와 같은 구조를 갖는 시스템 팔레트 색상 중 20개의 시스템 색상을 미리 정의하여 제공한다. 이 그림에서 위에 표시된 숫자는 팔레트의 인덱스 번호이다.

0~9	10~245	246~255
시스템 색상	사용 가능 색상	시스템 색상

(그림 2) 시스템 팔레트 구조

마이크로소프트는 자원 메커니즘을 사용하지 않는다. 대신에 윈도우 배경, 메뉴 버튼의 텍스트 및 다양한 다른 엔티티들의 색상을 규정하는 색상 스킴을 정의한다. 사용자는 시스템 수준에서 제어판 애플릿을 사용하여 색상 스킴을 규정한다. 프로그램은 버튼, 메뉴 및 다른 아이টে 사용되는 디폴트 색상에 대한 직접적인 제어를 하지 않고, 시스템 팔레트에 위치한 몇몇 정적 색상을 다시 프로그램하는 색상 스킴을 구현하였다. 응용 프로그램들은 기능적으로 색상 스킴을 참조한다. 마이크로소프트는 COLOR_WINDOW, COLOR_MENU, TEXT 및 COLOR_SCROLLBAR와 같은 색상 스킴에 대한 미리 결정된 기능의 이름을 제공한다. 따라서 순응적 응용 프로그램은 윈도우의 배경 색을 채우는 데에 특정한 RGB를 규정하는 대신에 COLOR_WINDOW를 사용한다. 이것은 윈도우의 배경이 사용자에게 의하여 규정된 스킴 색을 표시하게 한다. 마이크로소프트가 메커니즘을 제공하지 않는 대신에 응용 프로그램 사용자는 개별적 버튼과 같은 것에 대하여 개별적 색을 선택할 수 있다. 그럼에도 불구하고 마이크로소프트의 응용 프로그램들은 디폴트 색 스킴을 중첩시키도록 쓰여질 수 있거나 X와 같은 색 제어를 사용자에게 제공할 수 있다[19, 20].

20개의 정적 시스템 색 이외의 색을 사용할 필요가 있으면 그 응용에 대한 논리적 팔레트를 생성해야 한다. 팔레트 엔트리는 다음과 같이 정의된다.

```
typedef struct tagPALETTEENTRY
{
    Byte peRed ;
    Byte peGreen ;
    Byte peBlue ;
    Byte peFlag ;
} PALETTEENTRY ;
```

응용은 다음과 같은 구조를 사용하여 팔레트를 형성한다.

```
typedef struct tagLOGPALETTE
{
    WORD          palVersion ;
    WORD          palNumEntries ;
    PALETTEENTRY palPalEntry [1] ;
} LOGPALETTE, *PLOGPALETTE, *LPLOGPALETTE ;
```

LOGPALETTE 구조는 논리적 팔레트이면서 색표와 관련된 정보를 포함하는 메모리의 가변 길이 블록의 헤더로써 동작하도록 구성되었다. 그런고로, 응용은 보통 다음과 같이 그것을 생성할 것이다.

```
LOGPALETTE *pPalette =
(LOGPALETTE *)
malloc (sizeof (LOGPALETTE) + (numEntries-1)
        * sizeof (PALETTEENTRY));
pPalette->palNumEntries = numEntries;
```

SelectPalette()에 대한 호출은 다음에 그리기 동작어 사용

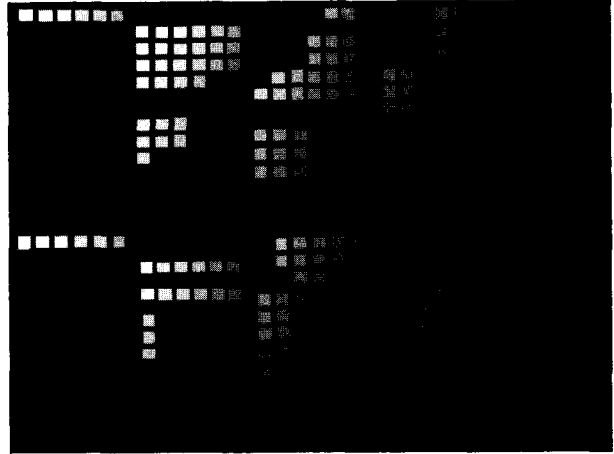
할 팔레트를 규정한다. RealizePalette은 선택한 응용 팔레트로부터 현재의 시스템 팔레트에 색을 매핑하는 실제적인 작업을 수행한다. 과거에는 기술적 이유 때문에 이 동작을 두 개의 별도의 함수에 위임하였지만 현재에는 이 둘이 항상 쌍으로 호출되어야 한다. SelectPalette()은 장치 컨텍스트에 이전에 장착된 팔레트에게 핸들을 넘겨준다. 보여준 경우에는 이것은 항상 시스템 팔레트가 될 것이다. 장치 컨텍스트를 파괴할 때 응용 팔레트를 선택된 대로 남겨두는 것은 오류이다. 그러므로 시스템 팔레트의 핸들은 최초에 SelectPalette()을 호출할 때 저장되고, 응용 팔레트를 통한 그리기, 표시하기가 완료된 후에 재 선택된다. 그리기 프로세스 중에 팔레트를 장착하는 것에 덧붙여서 응용은 시스템에게 윈도우즈의 Z 순서가 변하는 어떤 때라도 원하는 팔레트를 알려주어야 한다. 윈도우즈는 시스템 팔레트를 직접적으로 접근 할 수 없기 때문에 20가지 정적 시스템 색을 단축시키지 못하는 응용은 X에서보다 윈도우즈에서 보다 많은 작업을 해 주어야 한다. 그럼에도 불구하고, 응용 프로그램이 직접적으로 시스템 팔레트를 제어할 수 없게 하고 그것을 유지시키는 것은 몇 가지 장점이 있다. PALETTEINDEX 매크로를 사용할 때, 응용 프로그래머는 일어나는 잠재적인 색 재매핑을 고려할 필요가 없다. 규정한 인덱스 값이 응용 팔레트에 대한 인덱스이지 시스템 팔레트에 대한 것은 아니다. 이 엔트리의 한 색이 시스템 팔레트에 있는 다른 색으로 매핑된다면 이 변환은 위에서 언급한 두 번째 변환에도 일어난다.

4. 크리스틴의 색맹용 Web Safe Palette

크리스틴은 1999년 정상인, 특히 웹 문서의 저작자가 색맹이 그 문서를 보았을 때 문제를 야기 시키는지를 검증할 수 있는 제 1색맹용, 제 2색맹용, 제 3색맹용 색맹 모사 팔레트를 발표하였으며, 2002년 웹 문서의 저작자가 색맹을 고려한 웹 문서를 작성하는데 도움을 주는 색맹용 팔레트와 사용 가이드라인을 제시하고, 그 팔레트를 표준으로 할 것을 제안하였다. 이것은 일반적인 웹 안전 팔레트로 문서를 표현할 때 표현 방식에 따라 색각 이상자가 혼란을 일으킬 수 있는 여지가 있다는 것을 의미한다. 또한 색맹 사용자가 그 팔레트를 이용하여 해당 문서를 접근하면 웹 문서를 오류 없이 인식할 수 있다고 주장하였으며, 실제 많은 색맹을 대상으로 실험하여 제안 방식의 효과를 검증하였다.

그러나 그는 제안한 팔레트에 대한 수학적 모델을 제시하거나 수학적 검증을 제시하지는 않았다. 그는 각 모니터에서 R, G, B의 음영과 강도가 각각 다르고, 감마의 지정이 잘 변화하는 등 모든 컴퓨터에 대한 단일한 해답은 존재하기 어렵지만, 색상의 정확도의 차이에서 발생하는 문제점은 색맹이 색 차이를 구별하지 못하여 생기는 문제점에 비해

서는 큰 문제가 아니며, 실생활에서도 엄밀한 정밀성이 요구되는 경우는 드물다고 주장하였다.



(그림 3) 크리스틴의 색맹용 웹 안전 팔레트

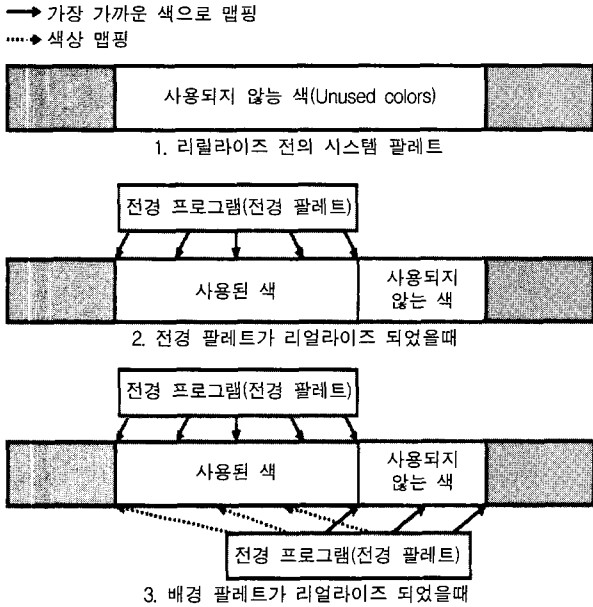
그는 팔레트의 색 선정을 적색맹과 녹색맹의 미묘한 차이에 근거하여 색이 청색으로 변환되는지, 금색으로 변환되는지에 따라 따뜻한 색과 차가운 색 군으로 분류하였고, 각 군에서는 밝기에 따라 5개의 부군(sub group)으로 분류하였다. 그가 제시한 팔레트는 (그림 3)에 표시하였다. 이 팔레트의 특징은 실제 제 1색맹과 제 2색맹의 차이가 분명하지 않고 미묘한 차이만이 있음을 이용하였고, (그림 1)에서 보듯이 정상인에 비하여 느끼는 색이 매우 제한적이므로 정상인에 대한 팔레트의 색의 다양성을 포기하는 대신에 색 혼란 선을 회피하도록 색을 그룹화하고, 선정하였으며, 혼란을 일으킬 만한 색은 각 군과의 거리 즉 밝기의 차이를 기준으로 색을 그룹화 한 것이 특징이다. 실제 혼란 선을 회피하는 방법으로써 대부분의 연구자들은 밝기와 대조를 언급하고 있으나, 임의의 그림을 분석하지 않고 대조 효과를 주는 것은 불가능하므로 결국 선택할 수 있는 회피 방안은 밝기 밖에 없는 셈이다. 크리스틴은 216개의 웹 안전 팔레트만을 언급하여 일반적인 전자문서에 대한 적용은 언급하지 않았다. 따라서 일반적인 전자문서에 대해서는 크리스틴의 팔레트를 확장하거나 수정할 필요가 있다. 본 연구에서는 크리스틴의 웹 안전 팔레트에 무채색 40개를 추가하여 일반적인 전자문서에 적용할 수 있도록 확장하였다.

5. 구현 및 시험

윈도우 3.0 버전 이후부터는 팔레트를 사용할 수 있도록 GDI (Graphic Device Interface)를 확장되었고, 윈도우즈는 각각의 어플리케이션에 팔레트를 할당해 줄 수 있도록 설계되었으며, 이 할당된 팔레트를 논리적 팔레트라 한다. 논리 팔레트는 매핑 알고리즘에 의해 필요할 때마다 시스템 팔

레트에 매핑하여 사용될 수 있다. 논리적인 팔레트는 256개의 항목을 가질 수 있으며, 각 항목은 0에서 255까지의 값을 갖는 RGB(red, green, blue)로 표현된다.

논리적 팔레트를 시스템 팔레트로 맵핑하는 과정을 수행하는 팔레트 리얼라이즈 과정을 아래 그림을 통해 표현하였다.



(그림 4) 팔레트 리얼라이즈 과정

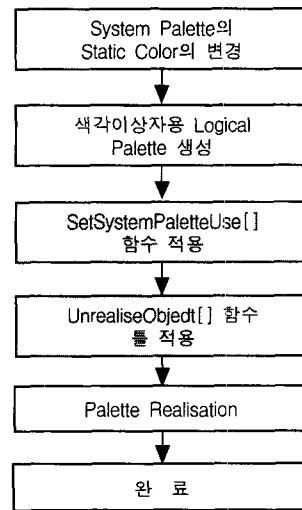
시스템 팔레트를 구성하는 엔트리는 정적인 상태, 사용된 상태, 사용되지 않은 상태 중 하나의 속성을 갖고, 리얼라이즈 과정에서 영향을 받지 않는 고정적인 색(20개)은 정적인 상태를 갖는다. 리얼라이즈 과정을 통해 사용되지 않은 엔트리가 구체적인 값을 갖게 되면 사용된 상태로 전환된다. 이렇게 사용된 엔트리로 설정되면 배경 팔레트의 리얼라이즈 과정 중에 이 색상을 변경되지 않는다.

이와 같은 팔레트 맵핑으로 시스템 팔레트 영역을 모두 사용하면 나머지 배경 윈도우가 사용할 엔트리가 없어진다. 이때, 팔레트 매니저는 배경 윈도우가 사용하는 색과 가장 유사한 색을 시스템 팔레트에서 찾아 맵핑하게 된다. 논리적 팔레트를 사용하지 않는 프로그램은 정적인 색상(20개)를 사용하게 되며, 논리적 팔레트를 사용한 경우 나머지 236개에 임의의 색을 설정하여 사용할 수 있다. 이때 사용되는 API 함수가 CreatePalette(), SelectPalette(), Realize Palette()이며, 논리적 팔레트를 활성화시키는 것은 다음 3단계로 구성된다.

- ① 응용 프로그램이 사용할 팔레트의 모든 엔트리의 컬러 설정은 배열에 저장하고, CreatePalette() 함수를 사용해서 팔레트로 변환한다. CreatePalette()는 팔레트의 핸들을 반환하고 이것은 보통 프로그램에서 쓰일 것에 대비해서 변수에 저장된다.

- ② 팔레트는 SelectPalette()를 이용해서 디바이스 컨텍스트로 선택된다.
- ③ 팔레트는 RealizePalette()를 이용해서 인식된다. RealizePalette() 함수는 사용 중인 컬러 지정을 바꾸며, 논리 팔레트 지정을 하드웨어 팔레트에 기록한다. 활성화된 윈도우는 하드웨어 컬러 지정에 우선 순위를 주며, 나머지 하드웨어 팔레트 내용은 비 활성화된 윈도우에 의해 사용될 수 있다. 비디오 하드웨어의 능력이 이상이 되는 팔레트 엔트리는 이용 가능한 가장 근접한 색상으로 맵핑된다.

이상의 3단계 절차 중간에 응용 프로그램이 시스템 팔레트의 정적인 색 영역을 변경할 지를 설정하는 기능이 필요하며, 이를 SetSystemPaletteUse() 함수로써 처리하였다. 이 함수는 시스템 팔레트 변경 이전에 사용된다. 또 하나의 중간 단계로써 적용되는 팔레트가 시스템의 작업에 관계없이 지속적으로 적용되게 하기 위하여 UnrealizeObject() 함수를 사용하여 지금의 팔레트 변경이 시스템에 임의적인 첫 번째 팔레트인 것처럼 처리되도록 하였다. 이상의 색맹 보정 팔레트 적용 절차를 (그림 5)에 간략하게 표시하였다. 그림에서 음영을 처리한 부분은 본 연구에서 일반적인 절차에 추가한 절차이다.



(그림 5) 색맹 보정 팔레트 적용 절차

웹 문서의 경우 매킨토시와 윈도우용 운영체제에서 서로 정확히 동일한 팔레트를 지원하지는 않는다는 것을 고려해야만 한다. 대체로 256색상의 팔레트에서, 두 운영체제의 브라우저가 공통으로 지원하는 색상은 216색이며, 나머지 40색은 어느 한쪽에서라도 디더링이 필요한 영역이 된다.

구현 단계에서는 API 함수를 활용하여 C++로 제작하였으며, 색각 이상자를 위한 팔레트의 항목으로 LOGPALETTE 형 데이터의 값을 채운 후, 팔레트를 생성하고, 데스크탑의 DC를 얻어 이 DC에 생성된 팔레트를 할당하여 시스템 팔

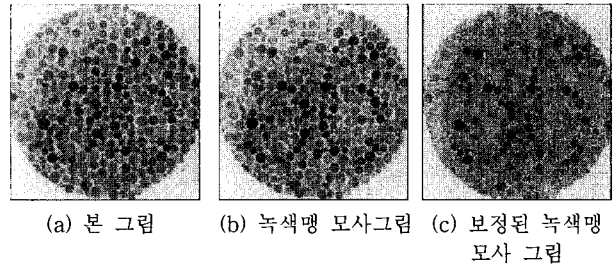
레트와 매핑하고, 윈도우 화면의 색상을 변경한다. 변경 전에 현재 팔레트를 저장하여, 종료 후 구 팔레트는 복구된다.

이때 기존에 발표되어있는 색각 이상사용 팔레트는 웹브라우저에서 사용되는 팔레트를 기준으로 216색만을 지정하도록 구성되어 있었으며, 이 경우 40개의 빈 시스템 팔레트 영역에 다른 어플리케이션에서 디폴트 팔레트에서 자신이 필요한 색을 갖고 다시 채우는 현상이 나타나는 문제점이 표출되었으며, 해결 방법으로 무채색을 활용해 40개의 색을 채워 넣는 방법을 활용하였다. 또한 색의 충실도를 고려하여 팔레트의 색 배열을 조정하였다.

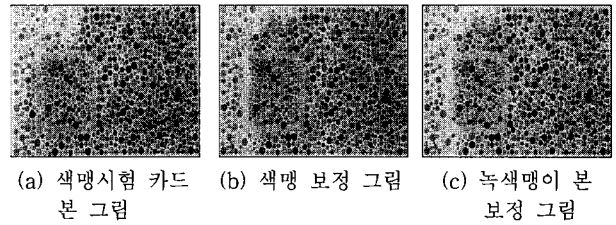
개발된 소프트웨어를 이시하라 색맹시험, 와고너 박사의 색맹 시험[21]과 일반 전자문서에 적용한 화면을 대상으로 vischeck[14]의 색맹 시뮬레이터를 사용하여 모의 실험을 수행하였다. 색맹 시뮬레이터는 완전한 이색각자를 모사하는 것으로 비정상적인 삼색각자의 시각을 모사하지는 못하며, 실제 색맹·색약 장애는 개인적으로 나타나는 정도의 차이가 매우 큰 것으로 보고되고 있어, 다양한 색각 장애자를 대상으로 색각 이상 정도와 현상을 측정하여 규명한 다음에 시험에 참여하도록 하여야 정확한 분석을 할 수 있다.

이시하라 팔레트 중 하나를 시험한 결과를 (그림 6)에 표시하였고, 와고너 박사의 색맹 시험 카드 중 하나를 시험한 결과를 (그림 7)에 표시하였다. (그림 6)에서 보정된 결과는 배경 색이 숫자에 비하여 좀 더 구분이 잘 가도록 대비가 되는 것을 알 수 있으며, (그림 7)의 경우 우측 상단에 작은 색 동그라미로 보이는 부분이 보정된 그림에서는 실제로 강화되어 뚜렷하게 나타나지만, 모사 프로그램을 적용하면 역시 해당 부분이 잘 보이지 않는 것을 알 수 있다. 따라서 완전 이색각자는 이 보정 프로그램을 사용하여도 좀 더 개선된 식별 능력을 보여 줄 순 있어도 이시하라 시험이나 다른 종류의 색맹 시험을 완전히 통과하지는 못할 것이 예상된다. 그러나 보정 효과가 뚜렷한 만큼, 혼란도가 덜한 여러 다른 문서에서는 효과가 있을 것으로 추정되며 실제 운전 면허 시험에서 색맹 검사에 불합격한 사람이 신호등 등 다른 방법에 의한 실제 판단 과정에서 59.4%가 적합 판정을 받아 면허를 교부 받은 것이 보고된 바 있다[22]. 특히 비정상 삼색각자에게는 상당한 효과가 있을 것으로 추정된다. 또한 색맹을 고려하지 않고 작성된 전자문서의 경우에 해당 문서를 분석하지 않고 적용하여 그림의 구성에 따라 혼란선이 발생할 경우, 팔레트의 적용만으로는 보정 효과가 제한적이기 때문이다. 따라서 충분한 보정 효과를 얻기 위해서는 화면 분석을 하여 밝기 조절과 대조 효과 기법을 병행하는 방식을 사용하던지, 전자 문서의 저작자가 해당 팔레트를 사용하여, 색맹을 위한 가이드 라인을 지키면서 작을 하여야 할 것이다. 화가가 그린 그림을 대

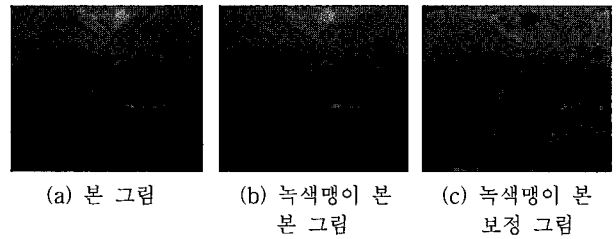
상으로 시험한 화면을 (그림 8)에 표시하였다. 본 (그림 8) (a)에서 녹색 풀밭과 구분되는 길이 모사된 (그림 8)(b)과 같이 적록색맹에게는 잘 보이지 않으나, 보정된 모사 (그림 8)(c)에서는 (그림 8)(b)에 비하여 비교적 풀밭과 구분이 잘 되는 것으로 나타났다.



(그림 6) 이시하라 색맹시험 결과



(그림 7) 와고너의 색맹시험 카드 시험 결과



(그림 8) 화가의 그림에 대한 보정 결과

6. 결 론

윈도우즈의 256 색 모드에서 동작하는 색맹 보정 소프트웨어를 설계하고, 구현하였다. 색맹 모사 프로그램을 사용하여 시험한 결과 색각 보정에 효과가 있는 것이 판명되었다. 이 소프트웨어는 크리스틴의 색맹용 웹 팔레트와는 달리 각종 워드 프로세서, 스프레드 시트, 프리젠테이션 응용 프로그램 등 멀티미디어 환경을 이용하는 모든 전자문서에 광범위하게 사용될 수 있으며, 한번의 수행으로 별도의 중지 명령이 없는 한 다중 창 환경에서 지속적으로 사용할 수 있어 편리하다. 기능 및 성능을 시험한 결과 256색 모드에서만 개발된 응용 프로그램이 동작하고, 윈도우즈가 근사 색으로 합성하기 때문에 세밀한 칼라 영상이 손상되는 경향이 있으며, 색의 충실도가 떨어지는 문제점과 문서 자체가 많은 혼란 선을 포함하고 있는 경우 보정의 효과가 미미하게 나타나는 문제가 있다. 전자의 문제는 윈도우즈 커넬을 수정하지 않는 한 나타날 수밖에 없으며, 후자의 문제

는 해당 문서를 분석하여 보정하는 방식을 사용해야 한다. 그러나 이 방식은 현재의 컴퓨터의 컴퓨팅 파워를 감안하면 실시간 처리 문제가 일어날 수 있다.

이 소프트웨어는 전자문서의 비중이 급격하게 늘어나고 있고, 미래에는 더욱 그 비율이 늘어날 것이 예상되는 시점에서 국내의 100만명 이상의 색각 이상자들이 전자 문서를 액세스 할 때 발생할 수 있는 혼란과 실수를 줄이는 데 기여할 수 있을 것이다. 추후에는 색의 충실도를 높이는 방법과 비교적 개인차가 크다고 알려져 있는 색각 이상을 정확히 진단하고, 개인별 색 보정을 수행할 수 있는 방식에 대한 연구와 핸드폰과 같은 단말기에 색각 보정 소프트웨어를 포팅하는 연구가 필요하다고 본다.

참 고 문 헌

[1] 남명화, 손무식, "선천성 색각 이상자 100명에 있어서 Nagel 씨 Anomaloscope에 의한 색각 이상의 분류", 한국안과학회지, 제21권, pp.511-515, 1980.

[2] HB Kim, SY Lee et al, "The incidence of congenital deficiency among Koreans," J korean medical Science, Vol. 4, pp.117-122, 1989.

[3] 이상우, 권정윤, "초·중등학생에 있어서의 선천성 적녹 색각 이상 시의 빈도", 한국안과학회지, 제27권, pp.211-215, 1986.

[4] <http://www.iamcal.com/toys/colors/stats.php>.

[5] Christine Rigden, "Now You See It, Now You Don't," IEEE Computer, Vol.35, No.7, pp.104-105, July, 2002.

[6] <http://www.section508.gov>.

[7] www.w3.org/TR/WAI-WEBCONTENT/.

[8] Jeff Carter, Mike Markel, "Web Accessibility for people with disabilities," IEEE Transactions of Professional Communication, Vol.44, No.4, pp.225-233, Dec., 2001.

[9] Christine Rigden, "The Eye of the Beholder-Designing for Colour-Blind Users," British Telecommunications Engineering, Vol.17, pp.291-295, Jan., 1999.

[10] Holly G. Atkinson, Bridgewater, Conn., "Method and System for Color Vision Deficiency Correction," United States Patent, Dec., 1996.

[11] G. W. Meyer & D. P. Greenberg, "Colour defective vision and computer graphics displays," IEEE Computer Graphics and Applications, Vol.8, No.5, pp.28-40, 1988.

[12] H. Brettel, F. Vienot & J. D. Mollon, "Computerized simulation of color appearance for dichromats," Journal of the Optical Society of America A, Vol.14, No.10, pp. 2647-2655, 1997.

[13] <http://www.btexact.com/>.

[14] <http://www.vischeck.com/>.

[15] 김민섭, 노문남, 이 건, 위원량, 이진학, 서한전산화 색각 검사 (1) : 전산화된 색각 검사의 개발 및 시험연구 대한안과학회지, 제41권 제1호, pp.205-214, 2000.

[16] 장영건 등, "색각 장애자를 위한 색각 보정 시스템의 설계 및 구현", 한국정보과학회 2002 가을 학술발표논문집, 제 29권 제2호, pp.181-183, 2002.

[17] <http://www.optometry.co.uk/articles/20001117/hodd.pdf>.

[18] O. Estevez, "On the Fundamental Data-Base of Normal and Dichromatic Color Vision," doctoral dissertation, Univ. of Amsterdam, Krips Repro Meppel, 1979.

[19] Theo Pavlidis et al, "Using Color in the X Window System versus Microsoft Windows : Part 1," Computer Graphics and Applications, IEEE, Vol.18, Issue 6, pp.64-73, 1998.

[20] Theo Pavlidis et al, "Using Color in the X Window System versus Microsoft Windows : Part 2," Computer Graphics and Applications, IEEE, Vol.18, Issue 6, pp.75-83, 1998.

[21] Dr. Terrace Waggoner, Color Vision Testing Made Easy, 2003.

[22] 주영숙 등, "운전면허 응시자 중 색각 이상자에 대한 신호등 검사" 대한안과학회잡지, 제23권 제1호, pp.135-137, 1982.



장 영 건

e-mail : ygjang@chongju.ac.kr
 1980년 인하대학교 전자공학과 학사
 1979년~1983년 국방과학연구소 연구원
 1983년~1994년 대우중공업 중앙연구소 책임연구원
 1991년 인하대학교 전자공학과 석사 (정보공학)

1995년 인하대학교 전자공학과 박사(정보공학)
 1995년~1996년 고등기술연구원 책임연구원
 1996년~현재 청주대학교 컴퓨터정보공학과 부교수
 2003년~현재 UC Davis visiting Professor
 관심분야 : HCI, CTI, 음성정보처리를 이용한 웹 프로그래밍, 보조기술, 재활공학