

색맹인 사람들을 도울 수 있는 스마트 폰 기반 색상 매칭 애플리케이션

정 명 범*

Color matching application which can help color blind people based on smart phone

Myoung-Beom Chung*

요 약

본 논문에서는 색맹인 사람들을 위한 스마트 폰 기반 색상 매칭 기법을 제안한다. 색맹인 사람들을 위해 기존 연구로 모바일 기반 애플리케이션들이 제공되기는 하였으나, 대부분의 연구가 사진 촬영 후 색상의 값, 이름만 제공할 뿐 동일 색상을 실시간으로 비교하지 못하는 불편함이 있다. 이러한 불편함을 해소하기 위해 우리는 스마트 폰의 카메라로 색상 비교를 위해 사진을 촬영하여 화면 옆에 두고, 실시간 입력되는 카메라 영상을 비교하여 유사 색상을 알려줌으로써 실시간 비교가 가능한 색상 매칭 알고리즘과 이를 이용한 애플리케이션을 개발하였다. 색상 매칭 알고리즘은 실시간 비교를 위해 Red, Green, Blue 그리고 Hue 값을 이용하여 코사인 유사도를 계산하며, 유사도 값에 따라 매칭 결과를 실시간으로 알려준다. 제안 방법의 성능을 판단하기 위해 색상 매칭 실험을 하였으며, 그 결과 매칭 성공률은 약 98%를 나타냈다. 따라서 제안 방법은 색맹인 사용자가 스마트 폰을 이용하여 자신이 원하는 색을 찾는데 효과적인 기법이 될 것이다.

▶ Keywords : 색상 매칭 애플리케이션, 색맹, 실시간 색상 매칭, 스마트 폰

Abstract

In this paper, we proposed the color matching application based on smart phone which can help color blind people. For color blind people, the existing methods and applications supported color matching application which based on mobile. However, because the most research only showed the color value and color name through capture image of mobile camera, those cannot compare with capture image color of mobile camera and color of real object in real-time. To solve those problem, we proposed the color matching algorithm and developed the color matching application that can compare with color of mobile camera's capture image and color of real object in real-time, because the proposed application divides

•제1저자, 교신저자 : 정명범

•투고일 : 2015. 3. 20, 심사일 : 2015. 4. 15, 게재확정일 : 2015. 5. 11.

* 성결대학교 컴퓨터공학부(Division of Computer Engineering, Sungkyul University)

screen of smart phone into two parts and it show one part as capture image of smart phone camera and the other part as real-time camera image of smart phone. Color matching algorithm calculate cosine similarity using Red, Green, Blue, and Hue value of each image for real-time comparing and show matching result according to similarity value in real-time. To evaluate the performance of the proposed application, we tested a color matching experiment using the proposed application and the matching result was 98% success rate. Therefore, the proposed application will be a useful application which can help color blind people.

▶ Keywords : Color matching application, Color blind people, Real-time color matching, Smart phone

I. 서 론

우리의 삶에서 색상은 매우 중요한 역할을 한다. 하나의 물체에 어떠한 색상이 칠해져 있는지에 따라 따뜻한, 차가움, 부드러운, 시원함 등과 같이 각기 다른 느낌을 전달할 수 있으며, 사람들이 입는 옷, 양말, 신발 등에도 각 색상의 조화에 따라 전달되는 느낌이 달라질 수 있다. 그러나 전 인구의 약 8%에게는 선천적으로 색상이 명확히 구별되지 않는 결함이 있다. 물론 이러한 결함이 일상생활에 큰 지장을 주지 않지만, 색맹인 사람의 경우 다양한 색상의 양말들 중 같은 색상의 양말을 고르는 것과 같은 사소한 일에서조차 스트레스를 받을 수 있다.

이러한 문제점을 해결하기 위한 기존 연구로 Ohkubo는 CCD, 컴퓨터, HMD를 이용하여 특정 색상에 약한 사람에게 색 보정에 의한 실제 색을 표현하는 기술을 제안하였다[1, 2]. Kondo는 피쳐 폰 기반 카메라를 이용하여 촬영한 영상의 특정 위치 색상 값을 Color wheel에 나타내어 색상 정보를 제공하였으며[3], Schmitt은 스마트 폰의 카메라의 촬영 이미지를 이용하여 제 2 색각 이상자 판별 Simulating과 Color wheel을 통한 색상 확인 애플리케이션을 제안하였다[4]. 그리고 최근 기술로 Manaf와 Harwahyu는 스마트 폰 카메라의 실시간 영상을 입력 받아 손가락이 가리키는 위치를 추적하여 특정 위치를 색상 값을 판단하여 말소리(Speech)로 알려주는 Augmented Reality를 사용한 연구가 제안되었다[5, 6].

그러나 기존의 연구들은 입력된 영상을 분석하여 사람들이 일반적으로 말하는 색상 값으로 표현하거나, 미리 정의되어 있는 색상을 찾아줄 뿐, 정의되지 않은 색상들에 대해서는 알

려주거나 찾아줄 수 없는 문제점이 있다. 한 예로 색맹인 사용자가 다양한 색상을 가진 여러 개의 양말들을 펼쳐 놓고, 동일한 색상의 양말들을 찾으려 하는 경우, 기존의 방법들은 각각의 양말을 모두 한번 이상 선택해야 하며, 선택한 양말의 색상들을 모두 기억해야지만 동일 양말을 찾을 수 있다. 게다가 손가락을 이용한 Augmented Reality를 적용한 애플리케이션의 경우 한 손은 스마트 폰을 들고, 다른 한 손으로 사물을 포인트 해야 하는 불편함이 있다.

따라서, 본 논문에서는 이러한 문제점을 해소하기 위해 찾고자 하는 색상을 스마트 폰 카메라로 촬영한 후, 촬영 이미지를 화면 왼쪽에 보이게 하고, 실시간으로 카메라 영상이 들어오는 화면을 오른쪽에 두어 색맹인 사용자가 보다 편리하게 자신이 원하는 색상을 찾을 수 있는 스마트 폰 기반 실시간 색상 매칭 기법을 제안한다. 제안 방법은 색상 비교를 위해 RGB 색상 값과 기존 연구에서 사용하던 HSI(H:Hue, S:Saturation, I:Intensity) 컬러 공간의 Hue 값을 사용하며, 이때 실시간 영상의 빠른 색상 분석을 처리하기 위해 입력 영상 중앙의 일정 영역을 이용한다. 비교를 위해 왼쪽에 있는 촬영 이미지는 사용자가 원하는 위치에 포인터를 옮길 수 있으며, 위치한 포인터 내의 색상 값과 실시간 영상에서 분석한 색상 값은 코사인 유사도(Cosine Similarity)를 이용한 색상 유사도를 비교하여 색상의 일치 여부를 알려준다. 즉, 제안 방법은 색맹인 사용자가 한 쌍의 동일한 색상 물체를 찾거나, 비교 할 때 유용하게 사용할 수 있다. 우리는 제안 방법의 성능을 판단하기 위해 iOS 기반 실시간 색상 매칭 애플리케이션을 제작하였으며, 다양한 색상을 지닌 여러 사물을 이용한 실험 결과 약 98%의 매칭 성공률을 나타냈다. 따라서 제안 방법은 색맹인 사용자들에게 손쉽게 동일 색상을 찾을 수 있는 효과적인 기법으로 활용 할 수 있을 것이다.

본 논문은 다음과 같이 구성된다. 2장에서는 영상의 RGB 값으로부터 HSI 값을 변환 하는 방법과 코사인 유사도를 이용한 색상 비교 방법을 설명한다. 3장에서는 제안 방법을 사용하기 위한 애플리케이션의 화면 구성과 색맹인 사람들에게 실시간으로 정보를 제공하는 효과적인 색상 매칭 알고리즘을 설명한다. 그리고 4장에서는 제안 방법을 적용한 애플리케이션의 성능 판단을 위한 실험 및 결과를 설명하며, 5장에서 결론을 맺는다.

II. 관련 연구

본 장에서는 색상 값 비교를 위한 이미지의 RGB 값을 HSI 컬러 공간으로 변환하는 방법과 코사인 유사도를 이용한 색상 값 비교 방법을 설명한다. 이미지의 RGB 값은 Red, Green, Blue 색상이 각각 0~255 사이의 값을 가지며, 이 값들의 조합에 의해 아래의 수식 1과 같이 하나의 픽셀 색상을 결정하는 것이다.

$$Color\ value\ of\ one\ pixel = R + G + B \quad (1)$$

수식 1에서 **R**은 Red 색상, **G**는 Green 색상, **B**는 Blue 를 나타내며, 컴퓨터와 스마트 폰에서 표현되는 이미지 픽셀 하나는 R, G, B 색상의 24bit format에서 그림 1과 같은 메모리 공간을 차지한다[7].

Red								Green								Blue							
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

그림 1. RGB 색상 표현을 위한 메모리 공간
Fig. 1. Memory space for RGB color expression

그림 1에서 Red, Green, Blue는 각각 8bit 공간을 차지 하며, Red는 16 ~ 23 위치에, Green는 8~15 위치에, Blue는 0~7 위치를 사용한다. 따라서 하나의 픽셀을 표현하는 RGB 색상은 수식 2와 같이 계산되어 메모리 공간에 입력 된다.

$$RGB\ Color = (R \times 256 \times 256) + (G \times 256) + B \quad (2)$$

수식 2에서 **Red**는 그림 1의 16~23 위치에 입력되기 위해 $256 \times 256 = 65536$ 을 곱하며, **Green**은 8~15 위치에 입력되기 위해 256을 곱한 것이다. 그리고 이 값은 16진수 표 기법으로 색상별 두 자리 수로 표현이 가능하다. 다음으로 이

미치로부터 추출한 RGB 색상은 수식 3-5에 의해 HSI 컬러 공간으로 변환 할 수 있다[8, 9].

$$H = \cos^{-1} \left[0.5 \times \frac{(R - G) + (R - B)}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right] \quad \text{if } (B > G) \quad H = 360 - H \quad (3)$$

$$S = \frac{1 - \min(R, G, B)}{I} \quad (4)$$

$$I = \frac{(R + G + B)}{3} \quad (5)$$

수식 3-5와 같이 RGB로 이루어진 이미지 픽셀을 HSI 컬러 공간으로 변환하는 이유는 다양한 주변 환경의 색상에서 사용자가 원하는 색상을 비교하기 위해서이다. 특히 색체를 나타내는 **H**와 **S**값이 밝기에 해당하는 **I**와 분리될 수 있으므로, 최근 많은 연구들이 **H**와 **S**를 이용하여 컬러 값 기반으로 화소들을 구분하는데 사용하고 있다.

색상 값 비교에 사용하는 코사인 유사도를 이용한 색상 값 비교 방법은 비교하고자 하는 색상들 간의 방향성을 고려한 거리 계산 방법이다. 비교 아이템간의 거리만을 계산하는 방법으로는 유클리디언 거리(Euclidean distance)를 주로 사용하지만, 이 방법은 방향성이 고려되지 않는 문제점이 있다. 코사인 유사도 방법은 유클리디언 거리에 각도를 고려한 계산 방법으로 아이템 검색 분야, 유사성 측정, 벡터 스페이스 모델 등에서 가장 많이 사용되며 수식 6과 같다.

$$sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{|\vec{i}| |\vec{j}|} = \frac{\sum_{k=1}^n i_k \cdot j_k}{\sum_{k=1}^n i_k^2 \sum_{k=1}^n j_k^2} \quad (6)$$

수식 6에서 **sim(i, j)**는 아이템 **i**와 **j**의 유사도 값이며 \vec{i} 와 \vec{j} 는 각 아이템의 값이 방향성을 가지고 있는 것을 나타낸다. 그리고 수식 6에 의해 **sim(i, j)** 값이 작을수록 두 아이템간의 유사성이 크며, 반대로 값이 클수록 두 아이템간의 유사성이 작은 것을 판단할 수 있다. 즉, 코사인 유사도 방법은 색상 기반 특징 값을 비교, 비슷하거나 동일한 색상, 이미지를 찾는데 주로 활용되는 알고리즘으로, Kaur은 피부 질병 이미지를 구별하기 위한 유사 이미지 검색에 활용되기도 하였으며[10], Qian의 경우 최근접 질의(nearest neighbor queries)를 위한 이미지 검색에 사용하기도 하였다 [11].

III. 색상 매칭 애플리케이션과 실시간 매칭 알고리즘

본 장에서는 색맹인 사용자를 위한 실시간 색상 매칭 방법에 관한 애플리케이션의 화면 구성을 소개하며, 색맹인 사람들에게 실시간으로 정보를 제공하는 효과적인 색상 매칭 알고리즘을 구체적으로 설명한다. 제안 애플리케이션의 화면은 그림 2와 같이 왼쪽에 사용자에 의해 촬영된 이미지를 배치하며, 오른쪽에 실시간으로 입력되는 카메라 영상을 배치한다.

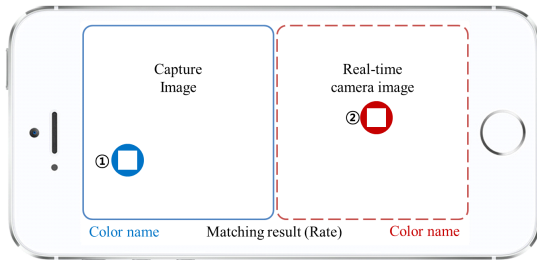


그림 2. 색상 매칭 애플리케이션의 화면 구성
Fig. 2. Screen arrangement of the color matching application

그림 2에서 왼쪽의 촬영된 이미지(Capture image)는 애플리케이션 실행 시 사용자가 색상 정보를 얻고자 하는 사진을 Photo Library에서 불러오거나, 스마트 폰의 카메라를 이용하여 촬영한 사진을 배치한다. 그리고 Capture image

위에는 pointing icon(그림 2. ①)을 배치하여 사용자가 색상을 매칭 하고자 하는 지점을 터치로 직접 옮겨 선택할 수 있게 한다. 오른쪽 실시간으로 입력되는 카메라 영상(Real-time camera image)은 스마트 폰 카메라에서 입력되는 실시간 영상을 보이며, 중앙에 포인터(그림 2. ②)를 표시하여 사용자에게 현재 비교하는 색상의 위치를 알려준다. 이때 왼쪽 선택 지점의 pixel 영역과 오른쪽 중앙 포인터의 pixel 영역은 동일하게 $n \times n$ 을 선택하여 각 pixel의 RGB 값을 HSI 컬러공간 값으로 변환한다. 각 pixel의 HSI 변환 값은 수식 7과 같이 배열로 저장하여 코사인 유사도 계산에 사용한다.

$$\begin{aligned} L_H[a(n-1)+b] &= \text{Hue}(a, b) \text{ of capture image} \\ R_H[a(n-1)+b] &= \text{Hue}(a, b) \text{ of real-time camera image} \end{aligned} \quad (7)$$

수식 7에서 a 와 b 는 해당 픽셀의 좌표 값을 의미하며, n 은 선택 영역의 행의 개수이다. L_H 와 R_H 는 각각 Capture image와 Real-time camera image의 Hue 배열이다. 그리고 수식 7과 같은 방법으로 L_S, R_S, L_I, R_I 를 각각 배열 형태로 저장하며, 각 pixel의 RGB 값 또한 수식 8과 같이 배열로 저장한다.

$$\begin{aligned} L_R[a(n-1)+b] &= \text{Red}(a, b) \text{ of capture image} \\ R_R[a(n-1)+b] &= \text{Red}(a, b) \text{ of real-time camera image} \end{aligned} \quad (8)$$

수식 8은 RGB 색상 값 중 해당 픽셀의 Red 값을 저장한 것으로 L_R 과 R_R 은 각각의 Red 색상 배열이며, 수식 8과 동일한 형태로 L_G, R_G, L_B, R_B 를 배열로 저장한다.

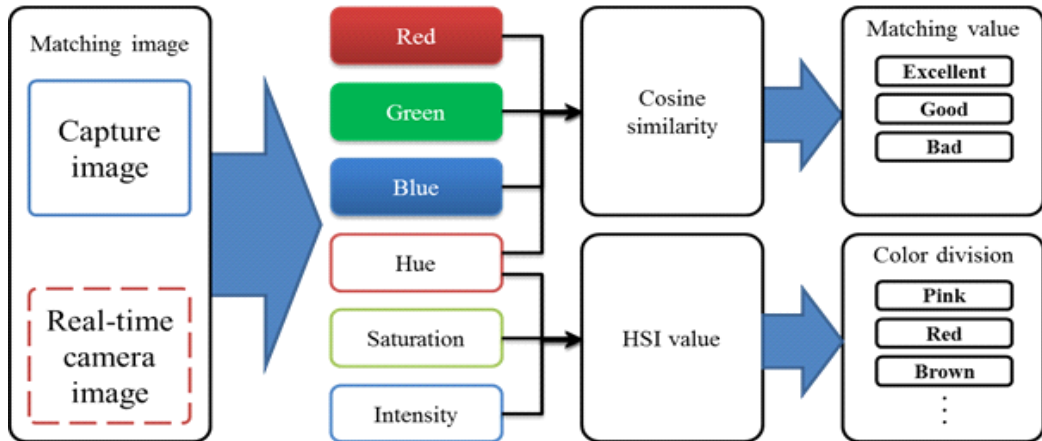


그림 3. 색상 매칭 애플리케이션의 매칭 알고리즘 흐름
Fig. 3. Matching algorithm flow of color matching application

색맹인 사람들에게 실시간으로 정보를 제공하는 효과적인 색상 매칭 알고리즘은 그림 3과 같이 앞서 저장한 각각의 RGB 색상 배열과 Hue 배열을 이용하여 코사인 유사도를 계산하고, 유사도 값에 따라 Excellent, Good, bad 중 하나의 매칭 결과 및 유사 비율을 표시한다. 그리고 저장한 HSI 값을 HSI value table의 Color names과 비교하여 사람들에게 친숙한 색상 정보를 동시에 제공한다. 그림 3에서 코사인 유사도(Cosine similarity)를 이용한 매칭 값(Matching value) 계산은 RGB 값과 Hue 값을 저장한 배열들의 평균 유사도와 pixel 각각을 비교한 유사도 평균을 사용하며, 매칭 방법은 그림 4의 슈도코드를 사용한다.

```

If ( Similarity value of RGB and Hue average is over  $T_1$  ) Then
    If ( Average of each pixel similarity of RGB and Hue is over  $T_1$  ) Then
        Return Excellent;
    Else If ( Average of each pixel similarity of RGB and Hue is over  $T_2$  ) Then
        Return Good;
    End If
Else If ( Similarity value of RGB and Hue average is over  $T_2$  ) Then
    If ( Average of each pixel similarity of RGB and Hue is over  $T_1$  ) Then
        Return Good;
    END IF
Else
    Return Bad;
    
```

그림 4. 색상 유사도 매칭 결과를 구하기 위한 슈도코드
Fig. 4. Pseudocode for getting color matching result

그림 4에서 T_1 과 T_2 는 색상 매칭 결과를 얻기 위한 각각의 임계값을 의미하며, 평균 유사도와 pixel 각각의 유사도 합계 평균 모두 T_2 를 넘을 경우에만 색상이 매칭 되는 것으로 판단한다. 다음으로 사람들에게 친숙한 정보를 제공하기 위한 색상 분류는 사용자 선택 영역의 Hue, Saturation, Intensity 값을 각각 평균 내어 Ruki가 사용했던, 표 1과 같이 총 16개의 Color name에 해당하는 값을 색상 매칭 애플리케이션 화면의 하단에 나타낸다.

표 1. HSI 값에 따른 각 색상 이름 정의
Table 1. HSI value for each color names defined

Color name	Hue	Saturation	Lightness
White	0-255	0-17	200-255
Gray	0-255	0-17	110-200
Black	0-255	0-255	0-50
Pink	0-3/175-182	70-120	230-255
Dark red	0-3/170-182	230-255	145-175
Red	0-10/165-190	0-255	0-255
Brown	15-17/190-197	79-255	155-190
Orange	10-23/191-200	0-255	0-255
Yellow	24-34/201-210	0-255	0-255
Light green	35-50/211-230	0-255	0-255
Green	51-65/231-250	0-255	0-255
Tosca	66-83/251-255	0-255	0-255
Light blue	84-107	0-255	0-255
Blue	108-136	91-255	0-255
Purple	108-140	0-90	0-255
Magenta	141-165	0-255	0-255

IV. 실험 및 분석

본 장에서는 제안 방법을 이용하여 구현한 실제 색상 매칭 애플리케이션을 소개하며, 제안 기술의 성능을 평가하기 위한 실험 및 결과를 설명한다. 제안 애플리케이션은 iOS 환경에 동작하도록 Apple사의 Xcode 6으로 제작하였으며, 실시간으로 입력되는 카메라 영상을 분석하기 위해 Apple에서 제공하는 AVCaptureVideoDataOutputSampleBufferDelegate를 사용하였다. 그림 5는 구현된 애플리케이션을 캡처한 것이다.



그림 5. 제안 방법을 적용한 색상 매칭 애플리케이션의 화면
Fig. 5. Screen of color matching application which applied the proposed algorithm

그림 5의 ①은 사용자가 색상 비교를 하려는 위치로 옮길 수 있는 포인터이며, 그림 5의 ②는 사용자가 실시간 입력되는 카메라 영상의 색상을 비교하기 위한 pixel 영역이다. 그림 5의 ③과 ④는 각 영상의 해당 픽셀 영역에 색상을 표 1에 분류에 따라 그 결과를 나타내는 것이며, 그림 5의 ⑤는 양쪽의 pixel 영역 색상이 일치하는지를 나타내는 것이다. 이때, 애플리케이션에 사용할 T_1 과 T_2 의 임계값 설정을 위해 T_1 과 T_2 값을 변화하며, 표 1의 16개 색상에 대한 색상 값 매칭을 각각 10회 시도하였으며, 결과는 아래 그림 6과 같다.

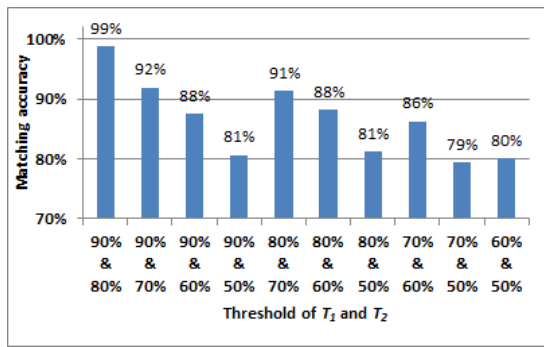


그림 6. T_1 과 T_2 임계값 설정에 따른 색상 매칭 결과
Fig. 6. Color matching results according to threshold value setting of T_1 and T_2

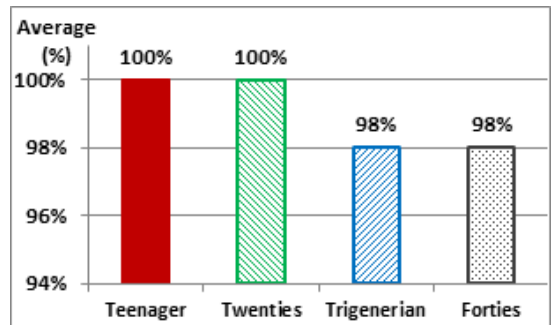
그림 6에서는 T_1 과 T_2 의 임계값을 90%, 80%로 설정하였을 때 가장 높은 색상 매칭 정확도를 나타냈다. 그리고 색상 매칭 결과에 직접적으로 관여하는 T_2 의 임계값을 낮춤에 따라 색상 매칭 정확도가 점차 감소하는 것을 볼 수 있다. 또한, T_2 의 값 같을 때 T_1 의 값이 차이가 있는 경우에도 색상 매칭 정확도가 유사함을 볼 수 있다. 이는 색상 매칭이 되었는지에 관한 결과를 최종적으로 T_2 에 의해 결정하기 때문이라 예상된다.

다음으로, 제안 방법의 성능 평가를 위한 실험은 다음과 같이 진행하였다. 실험 대상은 10대, 20대, 30대, 40대 각 연령층별로 5명이 참여하였으며, 실험자의 눈에 의한 색 구분을 방지하기 위해 Tomoyuki가 사용한 Simulation eyeglass를 제작하여 실험에 사용하였다. 성능 평가를 위한 색상 매칭 대상은 표 1의 16가지 색상 중 9개를 랜덤으로 선택하여 만들어진 3×3 표를 촬영하게 한 후 앞에 선택된 9개의 색상을 3×3 표에 재배열하여 동일한 색상을 각각 5초 이내에 찾게 하였다. 본 실험에 사용된 애플리케이션의 pixel 영역 크기는 20×20 pixel이며, 매칭 결과를 보여주기 위한 T_1 과 T_2 는 앞선 실험에서 높은 정확도를 나타낸 90%, 80%로 하였다.

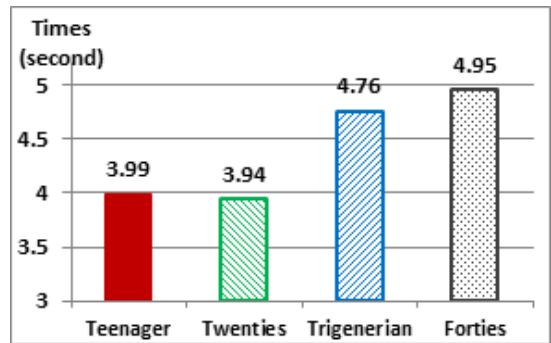
그림 7은 실험자가 제안 애플리케이션을 이용하여 색상을 촬영하고, 매칭 색상을 찾는 스마트 폰 화면을 캡처한 것이다. 그림 7의 Capture image는 왼쪽 위에서부터 차례대로 Green, Yellow, Light blue, Pink, Dark red, Light green, Red, White, Magenta 색상을 가지고 있으며, 실험자는 pointing icon을 Yellow에 옮겨서 동일 색상을 비교하여 찾고 있는 것이다.



그림 7. 색상 매칭 실험 진행 화면
Fig. 7. Application execution screen during color matching experiment



(a)



(b)

그림 8. 색상 매칭 실험 결과 (a) 연령대별 매칭 정확도, (b) 연령대별 색상 매칭 시간
Fig. 8. The result of color matching (a) Matching accuracy by age, (b) Time required for matching by age

그리고 실험 참가자는 같은 색을 찾기 위해 색상의 위치가 재배치된 다른 실험 종이를 이용하여 색상 매칭을 실행하고 있다. 현재의 화면에서는 Pink색을 나타내고 있어, Bad라는 매칭하지 않는 결과를 보여주고 있다. 그리고 그림 8은 제안 애플리케이션을 이용하여 각각의 동일 색상을 찾아낸 실험 결과를 나타낸 것이다. 그림 8(a)는 5초 동안 각 연령대별 동일 색상 검출 결과 평균을, 그림 8(b)는 실험자들이 동일 색상 검출 시 걸리는 시간 평균을 그래프로 나타낸 것이다.

그림 8(a)와 같이 10대와 20대 실험자들은 애플리케이션을 이용하여 제한 시간 안에 정확히 일치하는 색상들을 찾아낸 것을 알 수 있다. 반면 30대와 40대는 각각의 실험에서 1번씩 제한 시간 안에 원하는 색상을 찾지 못하는 것을 확인하였다. 그러나 제안 시간 안에 찾지 못한 것일 뿐, 제한 시간 이후 일치하는 색상을 각각 찾은 모습을 볼 수 있었다. 그림 8(b)는 색상 매칭 실험을 진행할 때, 실험자가 일치하는 색상을 찾을 때까지 걸리는 시간의 평균을 낸 것이다. 10대와 20대는 제한 시간보다 1초 빠른 약 4초가 걸리는 반면, 30대와 40대는 제한 시간에 가까운 것을 확인할 수 있다. 30대와 40대에 비해 10대와 20대가 색상 매칭에 빠른 이유는 10대와 20대의 경우 실험 자체를 게임 하는 느낌으로 집중해서 빨리 찾기를 하였다는 의견이 많았기 때문이다. 즉 실험 결과에서 제안 방법은 연령대에 관계없이 편리하게 색상을 비교하고, 원하는 색상을 찾을 수 있으므로 색맹인 사용자들에게 동일한 색상을 찾을 수 있는 효과적인 기법으로 활용 할 수 있을 것이다.

V. 결 론

본 논문에서 제안하는 스마트 폰 기반 색상 매칭 기법은 사용자들이 비교하고자 하는 색상을 실시간으로 비교할 수 있으며, 동시에 색상의 이름을 제공함으로써 색맹인 사람들에게 도움을 줄 수 있는 유용한 기술이다. 4장 실험 결과에서 볼 수 있듯이 제안 방법은 10~40대 전체적으로 98% 이상의 색상 매칭 성공률을 보였으며, 매칭을 위해 소요되는 시간 또한 평균 4.5초로 제안 애플리케이션은 실제 색맹인 사람들에게 즉시 서비스 할 수 있을 것으로 생각된다. 또한, 제안 방법을 이용하여 개발한 색상 매칭 애플리케이션은 3장에서 정의한 HSI 값에 따른 색상 이름을 더 다양하게 정의하면, 다양한 분야에 활용 할 수 있을 것이다. 색맹인 사람들 외에도, 어린 아이들에게 실제 색상과 그 색상의 이름을 설명해주고, 동일한 색상을 찾게 하는 교육 목적으로 활용이 가능할 것이다. 또한 개발한 애플리케이션의 화면에 정의된 색상 이름과 함께

RGB 값과 HSI 값을 함께 나타내어 준다면, 인쇄, 그래픽 디자이너들에게도 카메라 화면에 보이는 색상 값을 찾거나, 사용하고자 할 때 유용하게 쓰일 수 있을 것이다.

차후 연구로는 제안 방법에 사용되는 색상 유사도 매칭 알고리즘을 응용하여, 색맹인 사람들이 보다 편리하게 애플리케이션을 사용할 수 있도록 UI/UX 개선을 연구할 것이다. 그리고 실제 색맹인 사람들을 대상으로 한 실증 실험을 진행할 것이며, 기존 실험에서 사용한 16개의 색상 이외에 보다 다양한 색상을 접할 수 있는 실생활에서의 실험 연구를 할 것이다.

감사의 글

본 연구는 교육부 이공분야기초연구사업 중 기본연구지원사업의 연구비 지원(과제번호: NRF-2013R1A1A2061478)에 의해 수행되었습니다.

REFERENCES

- [1] T. Ohkubo, and K. Kazuyuki, "A color compensation vision system for color-blind people," *In: Proceedings of SICE Annual Conference 2008 IEEE*, pp. 1286-1289, August 2008.
- [2] T. Ohkubo, K. Kobayashi, K. Watanabe, and Y. Kurihara, "Development of a time-sharing-based color-assisted vision system for persons with color-vision deficiency," *In: Proceedings of SICE Annual Conference 2010 IEEE*, pp. 2499-2503, August 2010.
- [3] V. K. Y. V. S. Kondo, and V. Y. Tsuchiya, "Development of color-distinguishing application "ColorAttendant"," *FUJITSU Sci. Tech. J.*, Vol. 45, No. 2, pp. 247-253, 2009.
- [4] S. Schmitt, S. Stein, F. Hampe, and D. Paulus, "Mobile services supporting color vision deficiency," *In: Proceedings of 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM) IEEE*, pp. 1413-1420, May 2012.
- [5] A. S. Manaf, and R. F. Sari, "Color recognition system with augmented reality concept and

- finger interaction: Case study for color blind aid system," *In : Proceedings of 9th International Conference on ICT and Knowledge Engineering (ICT & Knowledge Engineering) IEEE*, pp. 118-123, January 2012.
- [6] R. Harwahu, A. S. Manaf, B. S. Ananto, B. A. Wicaksana, and R. F. Sari, "Implementation of color-blind aid system," *Journal of Computer Science*, Vol. 9, No. 6, pp. 794, 2013.
- [7] RGB Color Codes Chart, RapidTables.com, http://www.rapidtables.com/web/color/RGB_Color.htm
- [8] M. B. Chung, and I. J. Ko, "Intelligent copyright protection system using a matching video retrieval algorithm," *Multimedia Tools and Applications*, Vol. 59, No. 1, pp. 383-401, 2012.
- [9] S. M. Dominguez, T. Keaton, and A. H. Sayed, "Robust finger tracking for wearable computer interfacing," *In: Proceedings of the 2001 workshop on Perceptive user interfaces ACM*, pp. 1-5, November 2001.
- [10] S. Kaur, and D. Aggarwal, "Image Content Based Retrieval System using Cosine Similarity for Skin Disease Images," *Advances in Computer Science: an International Journal*, Vol. 2, No. 4, pp. 89-95, 2013.
- [11] G. Qian, S. Sural, Y. Gu, and S. Pramanik, "Similarity between euclidean and cosine angle distance for nearest neighbor queries," *In: Proceedings of the 2004 ACM symposium on Applied computing ACM*, pp. 1232-1237, March 2004.

저 자 소개



정 명 범

2004: 숭실대학교
미디어학부 공학사

2006: 숭실대학교
미디어학과 공학석사

2010: 숭실대학교
미디어학과 공학박사

현 재: 성결대학교
컴퓨터학부 조교수

관심분야: 콘텐츠,
모바일 애플리케이션,
신호처리, 멀티미디어통신

Email : nzin@ssu.ac.kr