Colorblind Image Correction Based on Segmentation and Similarity Judgement

Rong Ye\textsuperscript{1, a} and Ce Li\textsuperscript{2, b}

\textsuperscript{1}Room 1001, No.187 Hubinbei Road, Xiamen, China
\textsuperscript{2}No. Ding11, Xueyuan Road, Haidian District, Beijing, P.R. China

\textsuperscript{a} chelseayer@gmail.com; \textsuperscript{b} fanyan@wtedu.org

Abstract. Color blindness is the common disease of color vision deficiency (CVD), mainly caused by the heredity. Suffering from color blindness, a significant percentage of population faces the difficulty of recognizing color-based information. In this paper, we propose a colorblind image correction algorithm to accommodate digital images for color vision deficiency users. The algorithm has two parts: 1) image segmentation by K-means and system-clustering algorithms; and 2) image correction based on similarity judgement and multi-factor selection of color replacement. We then test the efficacy of our algorithm through simulations.

1. Introduction

Color vision deficiency is one of the most universal congenital diseases. Such a disorder affects 8% males and 0.5% females worldwide. Patients are reportedly troubled with their inability to distinguish certain colors [1, 2]. Human vision system perceives light through two receptors in the retina: rod cells which detect the intensity of light and cone cells which distinguish different colors of light. Cone cells are classified into three types according to the different ranges of light wavelength they are sensitive to. L-cones are sensitive to long wavelength light, M-cones are sensitive to middle wavelength light, and S-cones are sensitive to short wavelength light. The mixing signals of three cone cells construct the gamut of human color vision.

Color vision deficiency is caused by the loss of cone cells or shifts of sensitivity peaks [3]. It is categorized into three broad classes: monochromatism which only one type or none of the cones exists, dichromatism which only two types of cones are available and anomalous trichromats which all three types of cones exist but some have shifted peaks [4]. Modeling color vision deficiency is addressed as the foundation of simulating view of colorblind patients and further research. In [5], Gary W. Meyer brings up a problem in synthesizing the picture as it appears to dichromats. Meyer claims that, although people with normal color vision may accept the reproduction of a sense to be a simulated picture in dichromats’ vision, it is impossible to prove that the simulated colors are the same as perceived by colorblind people. Thus, it is not practical for individuals with normal vision to state what exact colorblind patients see. As it seems difficult for normal people to have the accurate data about the vision of colorblind patients, some people are found to be Unilateral Tritanopia [6], patients with one dichromatic eye and one normal eye. From their color matching measurements, the hues perceived by dichromats can be revealed. In the colorblind model proposed by Meyer in [5], Meyer applies the idea of “confusion line” and “major axis” in the XYZ color space. In [7], Farnsworth addresses the transformation of hue circuit for normal color vision into a hue line in the vision of
protanopes and deuteranopes. The endpoints of the line are approximately 470 nm which is yellow and 575 nm which is blue. Farnsworth thus defines such lines as a “major axis” on a chromaticity diagram. The confusion line was presented by J.C. Maxwell in 1855, defined as the line in the CIE diagram between two colors which a colorblind person cannot differentiate. Such line connects all colors which are indiscernible to the individual. The intersections between the major axis and each confusion line indicate the color perceived by the dichromat.

More recently, simulating the vision of CVD in digital images is possible by modeling color blindness in LMS color space. In [5], the model of computing LMS tristimulus value is

\[
S = \int E(\lambda) s(\lambda) d\lambda \\
M = \int E(\lambda) m(\lambda) d\lambda \\
L = \int E(\lambda) l(\lambda) d\lambda
\]

Where \(E(\lambda)\) is the light power spectral density function and \(s(\lambda)\), \(m(\lambda)\), and \(l(\lambda)\) are the fundamental spectral sensitivity functions for short, medium, and long wavelengths respectively. A colorblind simulating projection algorithm described in [8] replaces the tristimulus values with a reduced colorblind stimulus surface. Thus the simulation of color vision deficiency can be achieved by three steps: 1) convert the RGB data to LMS values; 2) apply the colorblind projection to the LMS values; and 3) convert the LMS coordinates back to RGB values.

In this paper, we present a colorblind image correction algorithm based on segmentation and similarity judgement. To implement the algorithm, we first make a colorblind simulation model for images in RGB color space, then segment the image by K-means clustering and system clustering, and then correct colors of the image by analyzing color similarity of each cluster, finally test the effectiveness of our algorithm on a simulated colorblind image for analysis and conclusion. The rest of the paper is structured as follows: Section 2 introduces the proposed implementation method. Section 3 describes the system implementation and conducts simulations to test the method. The last section concludes the paper.

2. Implementation Method

2.1 Colorblind Simulation Model

The L, M, S cone cells in normal retina constitute the human eye’s LMS space values. The visual system processes human eye’s RGB pixel values into the cone cells through the absorption signal. The transform matrix \(U\) implements this process from RGB space to LMS space as follows:

\[
\begin{pmatrix}
L \\
M \\
S
\end{pmatrix} = U \times \begin{pmatrix}
R \\
G \\
B
\end{pmatrix}
\]

(2)

Color blindness or color strength is due to the variation of absorption characteristics of the three cones. For red blind, when RGB space is converted to LMS space, the colors corresponding to the RGB cube space are mapped to the color plane of \(R = G\) along the L-axis. The mapping to color blindness in LMS color space is as follows, where \(T\) is the projection transformation matrix of the color blindness.

\[
\begin{pmatrix}
L_p \\
M_p \\
S_p
\end{pmatrix} = T \times \begin{pmatrix}
L \\
M \\
S
\end{pmatrix} = U \times \begin{pmatrix}
L_t & L_m & L_s \\
M_t & M_m & M_s \\
S_t & S_m & S_s
\end{pmatrix}
\]

(3)

Because the colorblind system projects the color along L-axis to a point and the R and G colors are just in this direction, they cannot be distinguished. For the same reason, the system projects the color
along M-axis to the R=B color plane, so the R and B colors cannot be distinguished. Similarly, the system also projects the color along S-axis to the B=G color plane.

Moreover, since the LMS color space is the interior perceived description of human eyes, difficult to observe on image, the color space is further converted back to RGB color space in order to simulate the model of new RGB color space in colorblind system as follows:

\[
\begin{align*}
R' &= U^{-1} \times T \times U \times R \\
G' &= U^{-1} \times \begin{pmatrix} L_R & L_G & L_B \\ M_R & M_G & M_B \\ S_R & S_G & S_B \end{pmatrix} \times U \times G \\
B' &= U^{-1} \times T \times U \times B
\end{align*}
\] (4)

Red colorblind simplification model from the above formula:

\[
\begin{pmatrix} R_p \\ G_p \\ B_p \end{pmatrix} = \begin{pmatrix} 0.14 & 0.86 & 0 \\ 0.14 & 0.86 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}
\] (5)

Green colorblind simplification model:

\[
\begin{pmatrix} R_d \\ G_d \\ B_d \end{pmatrix} = \begin{pmatrix} 0.33 & 0.67 & 0 \\ 0.33 & 0.67 & 0 \\ -0.02 & -0.02 & 1 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}
\] (6)

The steps in Figure 1 enable colorblind patients to obtain the normal color visual experience.

Figure 1 Colorblind simulation flow chart.

2.2 Colorblind Image Correction

2.2.1 Image Segmentation via Clustering. The local correction of colorblind space images consists of the clustering of colorblind images and analysis of each region to figure out the ones requiring correction. In this paper, we first use an improved K-means algorithm to divide the image into enough classes, and then use the system-clustering algorithm to merge n cluster centers obtained by K-means.

1) K-means clustering algorithm (K-means). The traditional K-means calculates the distance from all sample points to the center of the cluster. In a digital image, there is often a large amount of same data in image processing. Thus it is not necessary to repeat the calculation of the same sample point when calculating the distance between a sample point and the cluster center. This paper proposes a K-means clustering algorithm based on weighted sum. The algorithm is as follows.

Let the sample set \(X = \{x_1, x_2, \ldots, x_n\}\), where \(x_j\) represents the pixel value of image. Pixel values of all samples are counted, then a new sample set \(Y = \{y_1, y_2, \ldots, y_m\}\) and the corresponding frequency of all values \(F = \{f_1, f_2, \ldots, f_m\}\) are calculated for the objective function of error as follows:

\[
\min J(U, C) = \sum_{i=1}^{k} \sum_{j=1}^{m} (u_{ij}f_jy_j - c_i)
\] (7)

Where \(U = [u_{ij}]_{k \times m}\) represents a clustering matrix.

Update the cluster center according to the traditional K-means algorithm for iterative process and replace Equation (7) with the following formula.
$$c_i^{(l)} = \frac{\sum_{j=1}^{m} \left( u_{ij}^{(l)} f_{j} y_{j} \right)}{\sum_{j=1}^{m} \left( f_{j} u_{ij}^{(l)} \right)} \quad (8)$$

Table 1 illustrates the running time of images with five different sizes for K-means and weighted K-means algorithms, in which the image number 1 to 5 represent different images with resolution size from low to high. Running time has corresponding increase.

| Image Number | Image Size  | Running Time of K-means algorithm (unit: second) | Running Time of Weighted K-means (unit: second) |
|--------------|-------------|-------------------------------------------------|-----------------------------------------------|
| 1            | 200*150     | 0.1865                                          | 0.1514                                       |
| 2            | 480*233     | 0.3786                                          | 0.4037                                       |
| 3            | 543*358     | 0.9589                                          | 0.7731                                       |
| 4            | 1024*680    | 2.0029                                          | 1.5408                                       |
| 5            | 1024*768    | 4.7606                                          | 2.9303                                       |

Experiments show that the weighted K-means clustering algorithm is significantly faster than the traditional K-means in processing efficiency of image segmentation.

2) System clustering algorithm (Hierarchical Clustering). The algorithm of system clustering [9] separates all samples, initializes each as a cluster, and gradually merges clusters of samples by measuring the distance of sample clusters. In this paper, by combining K-means clustering with system clustering, the improved clustering results are obtained by following algorithm:

Step1: Given the convergence threshold $\varepsilon < 0$, input $n$ clustering centers obtained by K-means as the initial center of gravity of the system clustering;

Step2: Calculate the Euclidean distance between two different centers of gravity and find a minimum distance $m$. If $m < \varepsilon$, go to Step3. Otherwise, the algorithm stops.

Step3: Find the minimum value $m$ corresponding to the two categories $G_i$ and $G_j$ combined into a new class, and calculate the new center of gravity by formula (3).

$$x_r = \frac{1}{n_r} \left( n_p x_p + n_q x_q \right) \quad (9)$$

Among them $x_r$, $\{i = p, q, r\}$ denotes the center of gravity of class I. $n_p$, $n_q$, and $n_r$ are the corresponding samples of each class as $n_r = n_p + n_q$.

3) Image segmentation effect diagram. The original RGB image in Figure. 2 are segmented by K-means clustering and system clustering. Results are shown in Figure.3 and Figure.4 respectively.

Figure 2. The original RGB image.
In Figure 3, because the number of clusters k is large in K-means clustering algorithm, regions with similar colors are segmented to different clusters, such as green meadow and green trees. In Figure 4, these regions are merged in system clustering results to ensure that the differences between the various regions of the partition are large enough.

**Figure 3.** Segmentation results of K-means clustering by ten steps from (a) to (j).

**Figure 4.** Segmentation results of system clustering by eight steps from (a) to (h).

2.2.2 Image Correction via Similarity Judgement. 1) Similarity judgment. The image is divided into several regions by clustering algorithms. By measuring color similarity of such regions, color blindness is distinguished in all image regions. Lab color space is a uniform color system based on physiological characteristics. Due to the wide range in color gamut, the model in Lab color space is more suitable to simulate the human visual system color space and measure the color similarity because it can demonstrate all colors perceived by human naked eyes.

The color-matching center is represented by the color-clustering center of each segmentation region, and the similarity measure matrix $D$ is obtained by calculating the color similarity between the two regions. The matrix element $D(i, j)$ represents the LAB color space Euclidean distance between region $i$ and the region $j$ as:

$$D(i, j) = \sqrt{(l_i - l_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2}$$ (10)

The smaller $D(i, j)$ value indicates higher similarity of the two colors, which means that they are more difficult to be distinguished. After giving threshold $\varepsilon$, if $D(i, j) < \varepsilon$, the pixels included in the region represent the region area. By comparing the number of pixels contained in the two regions, the region containing fewer pixels is the one where the color needs to be replaced.
2) Multi-factor selection of color replacement. Assuming that the original color of the correction area is A and the color for replacement is B, in order to effectively make replacement and get a more friendly color for colorblind, B color must satisfy the following conditions. First, B color must be different from the color of other regions in the image, and the distance $d_\beta$ between B and other regions must be larger than the threshold $\beta$, which ensures that the colorblind patients can distinguish the region. Second, B color brightness and the original A color brightness must be consistent with the use of YUV color space. Y component, which determines the color brightness, is divided into multiple brightness levels, which are selected with the A brightness level for B color to replace consistently.

3. System Implementation and Testing

Based on Python platform, the process of color blindness correction system is shown in Figure.5.

![Flowchart of the correction algorithm.](image)

Figure 5 Flowchart of the correction algorithm.

A total of 10 red-blind and green-blind test charts and 10 red and green colors are selected for color blindness and green blindness. Color blindness test and natural image are simulated and corrected respectively. We then compare the test results of the images obtained by rotating the H component based on 120 degrees and the correction method based on the geometric transformation of the image.

Figure.6 (a) shows the original image for color blindness. Figure.6 (b) shows the red colorblind image. Figure.6(c) is the corrected colorblind images obtained in this paper. It is true that color blindness patients can accurately identify the images in the corrected images. Figure.6(d) is a corrected colorblind image obtained by using the rotation H component method. Figure.6(d) shows the color image in the natural image, and the corrected image is retained in the natural image. Figure.6(e) is the color blindness correction image obtained by image geometric mapping. Although all of these procedures from Figure.6(c) to Figure.6 (e) are aimed at the basic information of color blindness patients, all the color of the whole image is changed to a certain extent and thus is not conducive to real color perception.
4. Conclusion
This paper presents a colorblind image simulation and correction algorithm. The algorithm has the following features. First, it performs colorblind simulation filter for images in RGB color space. Second, it segments an image by K-means clustering and system clustering. Finally, it corrects colors of the image by analyzing color similarity of each cluster, ensuring the distance of replacement color to be larger than threshold, and maintaining consistent color brightness. The testing result proves the effectiveness of our algorithm, but also the drawback that it still affects the color vision deficiency patient’s perception of the real color in some degree. Thus the future work should further investigate the color replacement optimization that seeks for real color perception.

References
[1] McIntyre D. Color blindness: causes and effects [M]. Chester UK, Dalton Publishing. 2002.
[2] Tanuwidjaja, E., Huynh, D., Koa, K., Nguyen, C., Shao, C., Torbett, P. & Weibel, N. (2014). Chroma: a wearable augmented-reality solution for color blindness. Ubiquitous Computing, 799-810.
[3] Bao, J., Wang, Y., Ma, Y., & Gu, X. (2008). Re-coloring images for dichromats based on an improved adaptive mapping algorithm. International Conference on Audio, Language and Image Processing, 152-156.
[4] Jin, J., Di, S., Liu, P., Tang, G., Chen, X., & Du, R. (2014). The fabrication of a multi-spectral lens array and its application in assisting color blindness. International Journal of Optomechatronics, 10(1), 187-192.
[5] Meyer, G. W., & Greenberg, D. P. (1988). Color-defective vision and computer graphics displays. IEEE Computer Graphics and Applications, 8(5), 28-40.
[6] Alpern, M., Kitahara, K., & Krantz, D. H. (1983). Perception of colour in unilateral tritanopia. The Journal of Physiology, 335(1), 683-697.
[7] Farnsworth, D. (1943). The farnsworth-munsell 100-hue and dichotomous tests for color vision*. Journal of the Optical Society of America, 33(10), 568-574.
[8] Brettel, H., Vienot, F., & Mollon, J. D. (1997). Computerized simulation of color appearance for dichromats. Journal Of The Optical Society Of America A-optics Image Science and Vision, 14(10), 2647-2655.
[9] Zhang, W., Zhao, D., & Wang, X. (2013). Agglomerative clustering via maximum incremental path integral. Pattern Recognition, 46(11), 3056-3065.