An Improved Dynamic Daltonization for Color-Blinds

Anika Tasnim
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Mohammad S. Hasan
School of Computing and Digital Technologies
Staffordshire University
Stoke-on-Trent, UK

***Abstract*—Color-blindness is an inherited condition which leads to the sufferer not being able to distinguish between certain colors. One of the most widely used algorithms for enhancing a color-blind's color perception of images is Daltonization. Non-iterative Daltonization leaves the possibility of modifying imperceptible colors into colors that already exist in an image. Iterative Daltonization method provides better results by overcoming this problem. This paper presents a new iterative Daltonization method for protanopes which uses hue to ensure imperceptible colors are not modified into colors similar to existing perceptible ones. Experimental results show that the proposed method outperforms similar existing methods.**

Keywords—Color-Blindness, Daltonization, Hue, LMS color space, Protanope, RGB color space.

I. INTRODUCTION

Color-Blindness is a genetic condition which reduces the ability of a person to differentiate between certain colors. About 8% of men and 0.5% of women suffer from color-blindness worldwide [1]. Protanopia and Deuteranopia are two of the most common forms of color-blindness. Daltonization is one of the most widely used techniques for enhancing a color-blind's color perception. Non-iterative Daltonization methods use manually defined parameters which often lead to an imperceptible color being modified into another existing perceptible color of the image [2]. Existing iterative Daltonization methods [3], [4] and [5] use RGB values to check similarity between modified imperceptible colors and existing perceptible colors.

In this paper, an iterative Daltonization method is proposed which modifies a number of parameters for Daltonization and uses hue values to make sure modified unperceivable colors are different from existing perceivable colors. In RGB space color information is split among all three channels whereas, hue alone contains all information about colors. That is why hue values are used in the proposed algorithm.

The rest of the paper is organized as follows. A brief review of existing works is provided in section II, section III explains the proposed algorithm, section IV presents experimental results obtained with the proposed algorithm along with results of other existing algorithms for comparison. Finally, the paper concludes in section V.

II. EXISTING WORK

An extensive amount of research has been done to enhance color-blind's perception of colors. In [6] and [7] simulation of color deficient vision is proposed. Gradient mapping technique is used in [8] on an image and its color-blind perception to indicate regions that are not distinguishable to color-blinds. In [9] two methods are proposed to provide more accurate color information to dichromats and allow anomalous trichromats to perceive color as it is. Recoloring methods are applied in [10] and [11]. Paper [10] focuses on mapping colors to a smaller gamut dimension. The authors propose a method in [11] that emphasizes on the recolored image looking natural to the color-blind. A method is proposed in [12] that uses threshold technique to create a red mask which modifies the image into a perceivable form for a color-blind. However, the algorithm proposed in [12] fails to distinguish multiple shades of red. In [13] an interactive interface is provided to enhance the color perception of color-blind computer users. A technique of making videos more visually pleasing to color-blinds is proposed in [14].

One of the most popular methods for enhancing the color perception of color-blinds is Daltonization. It simulates how a color-blind views an image. Then an error image is generated taking the absolute difference between the original image and the color-blind's perception. Using the error values the original image is modified to enhance the color-blind's color perception of the image. In this paper, a color deficient view of an image is simulated. Firstly, the image is transformed from RGB to LMS space. LMS space represents the response of human eye's three cone cells to long (L), middle (M) and short (S) wavelengths. This LMS image is then converted to a reduced LMS space. Since color-blinds lack one or more functioning cones the reduction of LMS space is necessary to simulate their perception of colors. Depending on the type of color-blindness the reduced LMS space varies. The reduced LMS space is then converted back to RGB. Then an error image is generated containing the difference between the original image and color deficient vision of the images. Using the error values the original image is then modified to provide a better color view to the color-blind.

Daltonization is used as an iterative approach in [3], [4] and [5]. Paper [3] and [4] generate a mask based on a low threshold set on the error value. In addition to this mask, [5] considers another mask depending on the value of red pixels. Taking the intersection of these two masks, the final mask is generated in [5]. The papers [3], [4] and [5] then use their mask to divide the image into two parts, protanope perceptible pixels and protanope imperceptible pixels. Daltonization is run iteratively on protanope imperceptible part until the daltonized image does not have similar colors as initial protanope perceptible part. Color Checking Module or Similarity Checking Module checks similarity of the two images using RGB color values. When the colors of these two images are distinguishable, the final output is produced. Paper [4] and [5] use color clustering which reduces their computational complexity drastically compared to [3].

III. PROPOSED ALGORITHM

Initially, the proposed algorithm simulates protanope vision of an image. Then it divides the original image into two images depending on the color perceptibility of protanope. The imperceptible image is then iteratively Daltonized depending on the hue similarity of modified image and perceptible image. When the two images are different in terms of hue the final output is generated. Figure 1 shows the steps of the proposed algorithm.

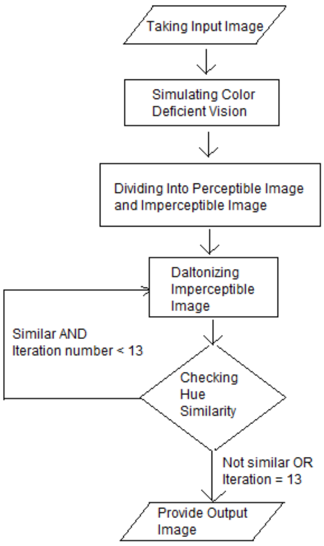

Figure 1: Flowchart of the proposed algorithm

A. Simulating Protanopia

This module simulates a protanope's perception of an image. It takes original an image I as an input and provides a protanope's perception of original image I_p as output. The input image is first converted from RGB color space into LMS color space using (1) [15].

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 17.8824 & 43.5161 & 4.1193 \\ 3.4557 & 27.1554 & 3.8671 \\ 0.02996 & 0.18431 & 1.4670 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

The protanopes do not have properly functioning L cones [16]. So, the normal color space L, M, S is then reduced to Protanope color space L_p, M_p, S_p with (2). Protanope color space L_p, M_p, S_p is then converted back to RGB color space R_p, G_p, B_p with (3). Finally, combining the three channels R_p, G_p, B_p the Protanope's view I_p is obtained.

$$\begin{bmatrix} L_p \\ M_p \\ S_p \end{bmatrix} = \begin{bmatrix} 0 & 2.02344 & -2.53581 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} R_p \\ G_p \\ B_p \end{bmatrix} = \begin{bmatrix} 0.0809 & -0.1305 & 0.1167 \\ -0.0102 & 0.0540 & -0.1136 \\ -0.0003 & -0.0041 & 0.6935 \end{bmatrix} \begin{bmatrix} L_p \\ M_p \\ S_p \end{bmatrix} \tag{3}$$

B. Categorizing Image

This module divides I into two images; $I_{correct}$ which contains pixels whose color is perceived correctly by protanope and $I_{incorrect}$ which contains pixels whose color is not perceived correctly by protanope. The inputs for this module are I and I_p. The outputs are $I_{correct}$ and $I_{incorrect}$.

In order to categorize the image two binary masks I_E and I_R are created. An error image E is calculated using (4).

$$\begin{aligned} E_R &= | R - R_p | \\ E_G &= | G - G_p | \\ E_B &= | R - R_p | \end{aligned} \tag{4}$$

Applying image binarization on E with a very low threshold the first mask I_E is obtained. When a pixel in E is equal or higher than the threshold, the corresponding pixel in I_E is assigned 1 and when it is below the threshold corresponding I_E is assigned 0 [3]. For achieving the second mask every pixel of I is scanned. Whenever a pixel's red component is higher than green and blue I_R is assigned 1. When the green or blue component is higher, I_R is assigned 0 [5]. By calculating the intersection of the two masks I_E and I_R the final mask I_{mask} is obtained. A logical AND between I_{mask} and I, $I_{incorrect}$ is generated. For generating $I_{correct}$, a logical AND of inverse I_{mask} and I is calculated.

C. Daltonizing

In this module, the image $I_{incorrect}$ is modified with a matrix M_i which converts colors of $I_{incorrect}$ into colors that are perceivable to protanopes and generates a modified image I_{dalton}. The initial value of M is given by (5).

$$M_1 = \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{1,4} & m_{1,5} & m_{1,6} \\ m_{1,7} & m_{1,8} & m_{1,9} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0.9 & 0.1 & 0 \\ 0.9 & 0 & 0.1 \end{bmatrix} \tag{5}$$

After generating I_{dalton}, protanope simulation is run on I_{dalton} to obtain I_{Dp} which shows how a protanope sees the Daltonized image. I_{Dp} and $I_{correct}$ are then sent for hue similarity checking. If the two images have similar hue then the matrix M is modified to M_i, where i represents iteration number. $I_{incorrect}$ is then modified again with the new matrix M_i which is given by (6).

$$M_i = \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{1,4} - (i-1) * x & m_{1,5} & m_{1,4} \\ m_{1,7} + (i-1) * x & m_{1,8} & m_{1,5} \end{bmatrix} \tag{6}$$

Here x is predefined to 0.07. Paper [3], [4] and [5] have used the value of 0.05. Reducing the value of x requires more iterations of Daltonization. So, the value is tuned to a higher value that provides a better result in less number of iterations.

The matrix M_i modifies a pixel's RGB values by using the red value to modify the green and blue values. With every iteration, a smaller portion of the red value is added to the green value and a larger portion of the red value is added to the blue value. In the 14^{th} iteration, the portion of the red value to be added to the green value becomes negative. To keep this red value to be added positive Daltonization is stopped at 13^{th} iteration.

After each iteration of Daltonization, Hue Similarity Check (*HSC*) module is used to check similarity of colors between I_{Dp} and $I_{correct}$.

D. Hue Similarity Checking (HSC)

HSC checks whether $I_{correct}$ and I_{Dp} have a significant amount of pixels having the same hue. If they have a significant amount of pixels having same hue another iteration of Daltonization is applied otherwise final output is generated.

First, hues $H_{correct}$ and H_{Dp} are calculated from corresponding R, G, B values for both $I_{correct}$ and I_{Dp}. The occurrence of each hue value is calculated by creating a histogram of hue values $H_{correct}$ and H_{Dp}. Then for every hue value, its occurrence is checked in both images. Hue occurrence under a very low threshold (e.g. 25) is ignored in *HSC*.

In this paper, the smallest size of an image is 200 pixel x 200 pixel, i.e. having a total of 40,000 pixels. Compared to the total number of pixels, the color of 25 pixels may be ignored as its color has a very less significant impact on the whole image's appearance. If any image's occurrence of that hue value is greater than the threshold, both of the image's occurrence of the hue is added to *SameHuePixel*. After scanning all the hue values, *HSC* returns 0 or 1 depending on the value of *SameHuePixel.* The return value of *HSC* is calculated using (7). Here, m and n are the length and width of image *I*, respectively

$$HSC(I_{correct}, I_{Dp}) = \begin{cases} 0, & SameHuePixel < 0.01 * m * n \\ 1, & Otherwise \end{cases} \quad (7)$$

If *HSC* returns 1 and the number of iterations of Daltonization is less than 13, another iteration of Daltonization is applied. When *HSC* returns 0, it indicates the number of pixels having the same hue in I_{Dp} and $I_{correct}$ is negligible compared to the total pixel number of *I*. When *HSC* returns 0 or the number of iterations of Daltonization is 13 Daltonization is stopped. A logical union of I_D and $I_{correct}$ is applied to produce the final processed image I_{final} which contains colors that are more perceivable to a protanope.

IV. Results and Analysis

The proposed algorithm has been implemented in Matlab R2015a using the image processing toolbox. In this section, the results achieved by using the proposed algorithm are shown alongside the results of conventional Daltonization algorithms for same images.

A. Algorithm's performance

Figure 2 shows the image "Improvisation 7" drawn by Wassily Kandinsky and color perception of a protanope. In Figure 3 the output of [3] is shown along with the proposed algorithm's output. The proposed algorithm makes red and white more distinguishable than that of [3]. The modified red part of the image becomes more noticeable using the proposed method. It also makes the image more vibrant.

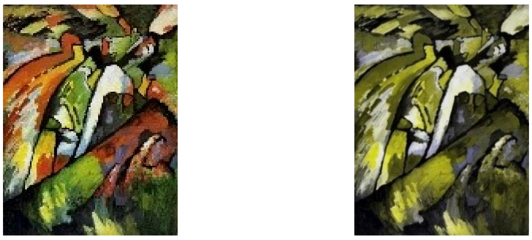

Figure 2: Original image [17] (left), protanope perception of the image (right)

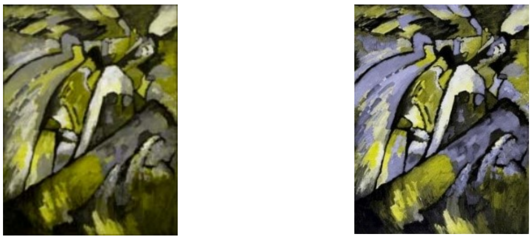

Figure 3: Protanope perception of Daltonization algorithm's output [3] (left), protanope perception of the proposed algorithm's output (right)

Figure 4 represents the image "Prism" and its protanopic perception. Figure 5 shows the output of [4] as well as the output of the proposed algorithm. The proposed algorithm is able to make a better distinction between red and grey for a protanope.

Figure 4: Original image [18] (left), protanope perception of the image (right)

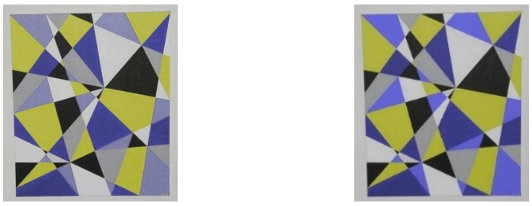

Figure 5: Protanope perception of Daltonization algorithm's output [4] (left), protanope perception of proposed algorithm's output (right)

An image of a collection of fruits along with its protanope perception is represented in Figure 6. The output of [5] is shown in Figure 7 along with the output of the proposed algorithm. The proposed algorithm makes the image look sharper than the other image due to better color separation. It also keeps the pineapple's leaves greener while the other image makes them appear more blueish.

Figure 6: Original image (left), protanope perception of the image (right)

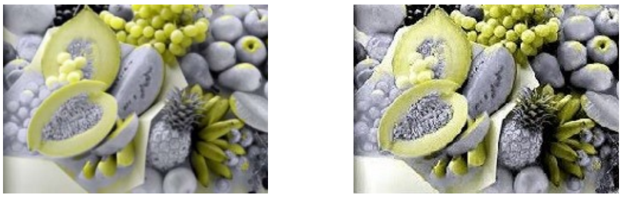

Figure 7: Protanope perception of Daltonization algorithm's output [5] (left), protanope perception of proposed algorithm's output (right)

In all the cases the proposed algorithm outperforms other existing Daltonization algorithms due to modification of Daltonization parameters and checking color similarity using hue values instead of RGB values. The definition of Hue Saturation Value (HSV) color space is quite similar to how humans perceive colors. In RGB color space image intensity and color information are mixed in all the three channels whereas, HSV color space separates image intensity and color information. While Saturation decides how pure the color is and Value decides how bright the color is, only Hue decides whether a color is red, green or any other color. That is the reason why checking similarity between colors using hue values provide better results.

B. *Time complexity*

If the input image has a width of m and a height of n, the algorithm proposed in [3] has a time complexity of $O(m^2n^2)$. As the algorithm proposed in [4] uses a Fuzzy-C-Mean algorithm for clustering RGB colors, it has a time complexity of $O(mndc^2i)$ where d is the number of dimensions (number of components in each point), c is the number of clusters and i is the number of iterations required for clustering. Paper [5] also uses color clustering without providing any details about the clustering method. So, the time complexity of the algorithm proposed in [5] cannot be calculated. The proposed algorithm has a drastically lower time complexity compared to that of [3]. The proposed algorithm has a time complexity of O(mnj) where j is the number of iterations of Daltonization. Here the upper bound for j is set to 13 as explained in III.C.

V. Conclusion

This paper has proposed a new technique of Daltonization for improving a protanope's perception of images. The proposed algorithm uses iterative Daltonization until the number of pixels having the same hue in protanope perceived part of the image and protanope unperceived part of the image is below a very low threshold. In most cases, the proposed algorithm is able to provide a significantly better result than existing similar algorithms. Though the algorithm is proposed for protanopes with slight modifications this algorithm can be used for people suffering from other color-blindness as well.

Acknowledgment

The authors would like to acknowledge an Erasmus+ International Credit Mobility (ICM) fund for Bangladesh awarded to Dr Mohammad Hasan at Staffordshire University, UK in 2016.

References

[1] "Types of Colour Blindness — Colour Blind Awareness." [Online]. Available: http://www.colourblindawareness.org/colour-blindness/types-of-colour-blindness/. [Accessed: 25-Aug-2017].

[2] "Vischeck: About Vischeck." [Online]. Available: http://www.vischeck.com/vischeck/. [Accessed: 29-Aug-2017].

[3] C. Anagnostopoulos, G. Tsekouras, and I. Anagnostopoulos, "Intelligent modification for the daltonization process of digitized paintings," no. Icvs, 2007.

[4] P. Doliotis, "Intelligent Modification of Colors in Digitized Paintings for Enhancing the Visual Perception of Color-blind Viewers Intelligent Modification of Colors in Digitized Paintings for Enhancing the Visual Perception of," no. April, 2009.

[5] H. Kim, J. Jeong, Y. Yoon, Y. Kim, S. Ko, and S. Member, "Color Modification for Color-blind Viewers Using the Dynamic Color Transformation," no. Ccm, pp. 602–603, 2012.

[6] G. W. Meyer and D. P. Greenberg, "Color-Defective Vision and Computer Graphics Displays," *IEEE Comput. Graph. Appl.*, vol. 8, no. 5, pp. 28–40, 1988.

[7] S. Yang, Y. M. Ro, E. K. Wong, and J. H. Lee, "Quantification and standardized description of color vision deficiency caused by anomalous trichromats - Part II: Modeling and color compensation," *Eurasip J. Image Video Process.*, vol. 2008, 2008.

[8] J. P. Srividhya, P. Sivakumar, and D. M. Rajaram, "The Color Blindness Removal Technique in Image by Using Gradient Map Method," *(Icsccn*, no. Icsccn, pp. 24–29, 2011.

[9] S. Yang and Y. M. Ro, "Visual contents adaptation for color vision deficiency," *ICIP 2003 Proc. Int. Conf. Image Process.*, vol. 1, pp. 453–456, 2003.

[10] K. Rasche, R. Geist, and J. Westall, "Re-coloring images for gamuts of lower dimension," *Comput. Graph. Forum*, vol. 24, no. 3, pp. 423–432, 2005.

[11] J. Bin Huang, Y. C. Tseng, S. I. Wu, and S. J. Wang, "Information preserving color transformation for protanopia and deuteranopia," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 711–714, 2007.

[12] R. Kulshrestha and R. K. Bairwa, "Removal of Color Blindness using Threshold and Masking," vol. 3, no. 6, pp. 218–221, 2013.

[13] L. Jefferson and R. Harvey, "An interface to support color blind computer users," *Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI '07*, p. 1535, 2007.

[14] J. Nam, Y. M. Ro, Y. Huh, and M. Kim, "Visual content adaptation according to user perception characteristics," *IEEE Trans. Multimed.*, vol. 7, no. 3, pp. 435–445, 2005.

[15] H. Brettel and J. D. Mollon, "Digital Video Colourmaps for Checking the Legibility of Displays by Dichromats ¸ oise Vie," vol. 24, no. 4, 1999.

[16] "Color Vision Deficiencies." [Online]. Available: http://psych.fullerton.edu/eriko/research/ColorVision.html. [Accessed: 29-Aug-2017].

[17] "Wassily Kandinsky - Improvisation 7, 1910." [Online]. Available: http://www.wassilykandinsky.net/work-74.php. [Accessed: 26-Aug-2017].

[18] "Sculpture and Mobiles by Bruce Gray - kinetic art, metal sculpture." [Online]. Available: http://www.brucegray.com/. [Accessed: 26-Aug-2017].