



# Modification of Colors in Images for Enhancing the Visual Perception of Protanopes

Polyxeni Sgouroglou, Christos-Nikolaos Anagnostopoulos

## ► To cite this version:

Polyxeni Sgouroglou, Christos-Nikolaos Anagnostopoulos. Modification of Colors in Images for Enhancing the Visual Perception of Protanopes. 10th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), Sep 2014, Rhodes, Greece. pp.73-84, 10.1007/978-3-662-44722-2\_9 . hal-01391031

**HAL Id: hal-01391031**

**<https://inria.hal.science/hal-01391031>**

Submitted on 2 Nov 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Modification of Colors in Images for Enhancing the Visual Perception of Protanopes

Polyxeni Sgouroglou<sup>1</sup>, Christos-Nikolaos Anagnostopoulos<sup>2</sup>

<sup>1</sup> School of Science & Technology, Hellenic Open University, Patras, Greece  
std041740@ac.eap.gr

<sup>2</sup> Cultural Technology & Communication Dpt., University of the Aegean, Mytilene, Lesvos, Greece, 81 100  
canag@ct.aegean.gr

**Abstract.** The purpose of this paper is to design and to implement proper color modification methods for enhancing visual perception of digital images for people with protanopia (one of the types of Color Vision Deficiency). This paper proposes one simulation method and two daltonization methods for the modification of digital colored images, implementing the additional step of quantization for a faster modification of colors which are not correctly perceived. In order to avoid color confusion a color checking module is implemented and optimal daltonization parameters are selected. The ultimate objective of this paper is to minimize the total processing time to such an extent that it is possible to apply the proposed methods to real-time sequences (online video). The entire application is tested in a set of artificial and real images, while the performance of the proposed algorithms is evaluated in an appropriately selected set of colored images.

**Keywords:** Color Modification, Color Quantization, Real-Time Processing

## 1 Introduction

Color Vision Deficiency (CVD) is quite common since about 8 - 12% of the male population and 0.5% of the female European population show in a certain degree, some type of CVD. Even though there is no known medical method which corrects such deficiencies in human vision, these people are not considered to suffer from a serious malfunction. Nevertheless there are certain cases in everyday life where color confusion may become annoying (e.g. reading a website) or even critical (e.g. road signs). Consequently there is intense research interest around the problematic color perception, resulting in the development of various techniques and algorithms for the digital simulation of such malfunctions. In addition several enhancement/modification methods (daltonization techniques) for the color information in digital images have been proposed in order to reduce in certain cases color confusion.

Many researchers have conducted research for appropriately modeling the visually impaired vision, presenting algorithms to simulate what individuals with vision deficiency see. The problem of color adaptation according to user's perception is also addressed in [1,2]. In [1] paper, among other issues, the problem of tailoring visual content within the MPEG-21 Digital Item Adaptation (DIA) framework to meet users' visual perception characteristics was addressed. The image retrieval aspect for people with CVD was discussed in [3]. Even a physiologically motivated human color visual system model which represents visual information with one brightness component and two chromatic components was proposed for testing the color perception of people suffering from CVD [4].

One of the most common techniques is presented in [6], in which the daltonization parameters are selected with a trial-and-error process. In [5], which was based on the work of [6], a method was proposed for the automatic modification of the daltonization parameters in such a way that no conflict among colors that were daltonized and colors that remained intact is ensured.

Although many algorithms have been proposed for the color modification of images in order to become more visible to people with CVD, relatively few steps have been made in the field of video and specifically in the field of real time processing, due to the high computational time required for the color modification of a single image/frame. For this purpose a color quantization technique is implemented by this paper.

In this paper the proposed methods are subject to the modification of the methods presented in [6] and [5] and their parallel combination with a color quantization technique. In fact the additional step of color quantization is implemented in [5] by this paper. Other methods where a color quantization technique has been applied is [7] although no RGB similarity checking routine is implemented and [8] although still relatively slow for use in real-time sequences due to the complexity of the selected clustering Fuzzy-C-means algorithm.

In the proposed methods, as presented in [9], a color reduction technique is initially applied to the image and a smaller set of colors/clusters is created. This smaller set of colors is then divided into two main groups: Colors that a person with protanopia can perceive right and don't need modification and colors that a person with protanopia cannot perceive right and do need modification. The colors of the second set are modified using specific initialization daltonization parameters and are placed in a new set. Then to avoid confusion an rgb similarity checking module is applied, so that there are no similar colors between the set of colors created after the modification and the set of colors that remained intact. If a similarity is found then the daltonization parameters are iteratively modified until similarity is eliminated.

The proposed algorithms were implemented in the programming environment MATLAB. As mentioned, the ultimate objective of this paper is to minimize the total processing time under the perspective of future application in real-time sequences. For this reason as appropriate quantization method the technique used in [10] was selected, as simpler and faster to use and with relatively reliable and comparable to k-means results. The daltonization methods described in this paper are focused only on a specific type of color vision deficiency, namely protanopia.

## 2 Proposed method

The original adaptive algorithm contains three steps [5]. Since the original adaptive algorithm is quite time consuming, in order to resolve this problem, the methods of this paper are proposed.

### 2.1 Simulation

Initially the number of the colors in the image is reduced using the color quantization method proposed in [10]. The resulting palette is an  $N \times 3$  matrix 'map', where the number of rows is equal to the number of quantized colors. Similarly to [5,6], the proposed methods follow the transformation matrices from RGB to LMS and vice versa. The improvement which is introduced in this paper is that in the methods proposed in this paper the simulation is applied only in the matrix 'map', and not in every pixel of the image [6]. After performing the simulation for each row in the matrix 'map' (i.e.  $N$  times total), the matrix 'map\_sim' is produced, which is the color palette of the quantized color image as perceived by a person with protanopia.

For each row of matrix 'map', namely for every color of the color palette, the following steps are repeated:

**Transformation of RGB values to LMS values.** The transformation from RGB to LMS is achieved by a multiplication with matrix  $A_1$  (1):

$$A_1 = \begin{bmatrix} 17.8824 & 43.5161 & 4.1193 \\ 3.4557 & 27.1554 & 3.8671 \\ 0.02996 & 0.18431 & 1.4670 \end{bmatrix} \quad (1)$$

The linear transformation from RGB to LMS is actually the multiplication described in (2):

$$\begin{bmatrix} L & M & S \end{bmatrix}^T = A_1 \cdot \begin{bmatrix} R & G & B \end{bmatrix}^T \quad (2)$$

**Finding the modified L'M'S' values.** Then a linear transformation [11] is applied, in order to reduce the normal color domain (LMS) to the protanope color domain ( $L_p M_p S_p$ ). The values that correspond to the loss of color information in the protanope vision are subtracted from the LMS values.

**Reverse transformation of L'M'S' values to R'G'B' values.** The conversion from  $L_p M_p S_p$  to  $R_p G_p B_p$  is again achieved through a multiplication proposed in [11]. The new modified  $R_p G_p B_p$  values demonstrate the way a protanope perceives each color represented by the corresponding row of the color palette.

After the execution of the described simulation sub-steps for every row of matrix 'map' (i.e.  $N$  times), the result is a matrix which represents the color palette of the color-quantized image, as it is perceived by a protanope.

## 2.2 Daltonization

It consists of the following sub-steps:

**Calculation of the palette's error matrix.** The error matrix is the absolute value of the subtraction between the color palette of the quantized image and the color palette that emerged after the simulation ( $\text{map} - \text{map\_sim}$ ). Per row we have the following equations (3), (4), (5).

$$E_R = |R - R_p| \quad (3)$$

$$E_G = |G - G_p| \quad (4)$$

$$E_B = |B - B_p| \quad (5)$$

After performing the subtractions for all the rows of the palette (N times total), we get the table 'error\_map'.

$$\text{error\_map} = \begin{bmatrix} |R_1 - R_1'| & |G_1 - G_1'| & |B_1 - B_1'| \\ \vdots & \vdots & \vdots \\ |R_N - R_N'| & |G_N - G_N'| & |B_N - B_N'| \end{bmatrix} \quad (6)$$

**Creating sets 'S1' and 'S2'.** The first set is called 'S1' and contains all the colors of the quantized image that do not need modification and are correctly perceived by a person with protanopia. The second set is called 'S2' and contains the colors of the quantized image that require modification. The separation is based on (6) and (7).

$$\text{rightPerceived}(RGB, R_p, G_p, B_p) = \begin{cases} \text{true if } (E_R < c, E_G < c, E_B < c) \\ \text{false otherwise} \end{cases} \quad (7)$$

There is a subtraction of the components of a color before and after the simulation (RGB and  $R_p, G_p, B_p$  respectively). If the subtraction of the three components is less than 20.48, then the RGB color is correctly perceived by a person with protanopia, not modified and placed in the set 'S1'. Conversely, if the subtraction of even one of the three components is bigger than the value 20.48, then the RGB color is not properly perceived and placed in the set 'S2' for daltonization. Basically we check if for all three values of each row of the error matrix (error\_map) applies  $< 0.08$ . Simultaneously the matrix 'error\_S2' is created. If a row of the color palette must be placed in 'S2', then the corresponding row of the error matrix is added into the error matrix of 'S2' (error\_S2).

**Daltonization of 'S2'.** Key to the daltonization procedure is not the error matrix of the whole color palette as in [6] and [5], but the error matrix only of those rows that need modification, namely the rows that belong to 'S2' (error\_S2). Actually the values of the error matrix of 'S2' are added to the initial values of the color palette of the quantized image, so that the values that indicate a high error level are redistributed to the blue side of the spectrum. This process is achieved with the help of the parameter

matrix ‘M’ of [6]. In the proposed methods the parameter matrix ‘M’ is always initiated according to the values presented in [5], as seen in (8).

$$M = \begin{bmatrix} m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 \\ m_7 & m_8 & m_9 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad (8)$$

Initially a multiplication of matrix ‘M’ with every row of ‘error\_S<sub>2</sub>’ is executed.

$$\begin{bmatrix} E_R' & E_G' & E_B' \end{bmatrix}^T = M \cdot \begin{bmatrix} E_R & E_G & E_B \end{bmatrix}^T \quad (9)$$

The modified values E<sub>R</sub>’, E<sub>G</sub>’, E<sub>B</sub>’ which arise after the multiplication in (9), are added to the values of the three components of the elements of ‘S<sub>2</sub>’, resulting the final daltonized values R<sub>d</sub>, G<sub>d</sub>, B<sub>d</sub> (10).

$$\begin{bmatrix} R_d & G_d & B_d \end{bmatrix} = \begin{bmatrix} R & G & B \end{bmatrix} + \begin{bmatrix} E_R' & E_G' & E_B' \end{bmatrix} \quad (10)$$

The colors R<sub>d</sub>, G<sub>d</sub>, B<sub>d</sub> that result after the modification of the elements of ‘S<sub>2</sub>’ are placed in a new set named ‘S<sub>3</sub>’.

### 2.3 RGB similarity checking module

The RGB similarity checking module consists of the following sub-steps:

**Simulation of ‘S<sub>3</sub>’.** For all the colors in ‘S<sub>3</sub>’ a simulation is performed and the new colors which result from the simulation are placed in the set ‘S<sub>3\_sim</sub>’.

**RGB similarity checking module.** Then the RGB similarity checking module is applied. It must be ensured that among new daltonized colors there is no color perceived by a person with protanopia, similar to a color that from the beginning remained intact. In other words, we must ensure that there is no similarity between the elements of the sets ‘S<sub>1</sub>’ and ‘S<sub>3\_sim</sub>’. This requirement is technically expressed by equations (11), (12).

$$similar(RGB, R'G'B') = \begin{cases} true & if \ (|R-R'| < d, |G-G'| < d, |B-B'| < d) \\ false & otherwise \end{cases} \quad (11)$$

$$confusion(S_1, S_{3\_sim}) = \begin{cases} true & if \ (\exists RGB \in S_1, R'G'B' \in S_{3\_sim} : similar(RGB, R'G'B')) \\ false & otherwise \end{cases} \quad (12)$$

As ‘d’ we denote an appropriately chosen threshold, which in this paper has the same value as in [5]. As similar are considered all the combinations for which applies (r±10, g±10, b±10). This value in MATLAB corresponds to 0.04 (in the range 0-1). We check the subtraction between the components of each color of ‘S<sub>1</sub>’ and ‘S<sub>3\_sim</sub>’ (RGB and R’G’B’ respectively). If the subtraction of the three components is below the value 0.04 then the two colors are perceived as identical. Conversely, if the subtraction of even one of the three components is bigger than the value 0.04 then the

two colors are perceived as distinct. If even one color in  $S_1$  is found similar to a color in  $S_{3\_sim}$  then the procedure of daltonization is repeated with modified parameters of the matrix  $M$ . At each subsequent iteration the parameter  $m_4$  is reduced by amount  $s$ , while the parameter  $m_7$  increases by the same amount. In this paper we used the same value as in [5], namely  $s = 0.05$ . Thereby the redistribution of the red color in the blue channel increases, while the same amount of green is transferred to the blue channel.

**Repetition if necessary.** For the repetition of the daltonization routine when the RGB similarity check locates a pair of similar colors between  $S_1$  and  $S_{3\_sim}$  two methods were developed in [9] and are presented in this paper.

*Method 1: Daltonization by modification of entire  $S_2$  and  $error\_S_2$ .* In the first proposed method the entire modified matrix  $M$  and the matrix  $error\_S_2$  are subjected to the daltonization routine. This method results into more rapid changes of the image, as each time that a pair of similar colors is found, the entire content of  $S_2$  is being modified.

*Method 2: Daltonization by modification of  $S_2$  and  $error\_S_2$  per row.* In the second proposed method the entire modified matrix  $M$  and only one row of matrix  $error\_S_2$  are subjected to the daltonization routine. This row is the one that appears similar to some element of  $S_1$ . The results of this method show milder coloration results in the final image.

**Termination condition.** The iterative RGB similarity checking routine ends when there is no similar color found between  $S_1$  and  $S_{3\_sim}$  or when the parameter  $m_4$  of matrix  $M$  is  $\leq 0.05$ .

**Image reconstruction.** Before termination the algorithm proceeds to the creation of the final color palette  $map_1$  and reconstructs the image by replacing in the original color palette map all the colors that belong to the set  $S_2$  with the respective colors in  $S_3$ . Finally, the color palette  $map_1$  is subjected to a final simulation in order to obtain the final daltonized image, as it will be perceived by a person with protanopia.

### 3 Experiments

At first the algorithm was tested on the artificial image “test.png” for a 4-cluster quantization without dithering. It has to be noted that in this specific image the results of the algorithm remain the same even for a 256-cluster quantization. When the original image includes fewer colors than those specified from the user through the quantization method, the produced color palette will also include less colors and the final image will include all the colors of the original image. Furthermore, in this specific image both Method 1 and Method 2 produce the same results, since the error matrix contains only one row. The results are presented in Figure 1.

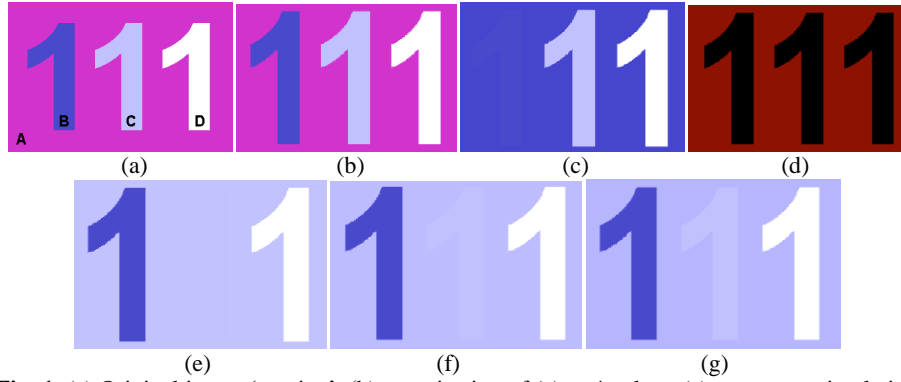
Subsequently, the simulation routine is executed for figure 1b and figure 1c is created, while a simulated color palette of figure 1b with the name  $map\_sim$  is produced (13). As shown in figure 1c, left ‘1’ is almost invisible for a protanope, as the

respective color for the background (color A) is now perceived as (69,69,205), very close to the values of the color of the left '1' (color B).

$$map\_sim = \begin{bmatrix} 69 & 69 & 205 \\ 193 & 193 & 255 \\ 73 & 73 & 203 \\ 255 & 255 & 255 \end{bmatrix} \quad (13)$$

$$error\_map = \begin{bmatrix} 141 & 18 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (14)$$

At this point we calculate the error of the color palette 'map' by finding the absolute value of the difference between the palette 'map\_sim' and the original palette 'map' of the quantized image. The result is stored in the matrix 'error\_map'.



**Fig. 1.** (a) Original image 'test.jpg', (b) quantization of (a) to 4 colors, (c) protanope simulation of (b): left '1' is hardly visible, (d) error\_map: the area of the image which presents high error value and needs daltonization appears red, (e) protanope simulation of (b) after the 1<sup>st</sup> iteration of the proposed algorithm: middle '1' is not visible, (f) protanope simulation of (b) after the 2<sup>nd</sup> iteration of the proposed algorithm: middle '1' remains imperceptible for protanopia, (g) protanope simulation of (b) after the 3<sup>rd</sup> iteration of the proposed algorithm: middle '1' becomes visible.

Having defined the error palette, the algorithm continues with the classification of colors to those which are correctly perceived by a protanope and do not need daltonization ( $S_1$ ) and to those which are not correctly perceived and do need daltonization ( $S_2$ ). It is obvious in (14) that only the R component of the first row in the 'error\_map' matrix shows a high value of error and therefore daltonization is necessary. In this way the colors of palette 'map' are divided into the sets ' $S_1$ ' and ' $S_2$ ', while the error matrix 'error\_ $S_2$ ' of the set ' $S_2$ ' is produced.



$$S_1 = \begin{bmatrix} 193 & 193 & 255 \\ 73 & 73 & 203 \\ 255 & 255 & 255 \end{bmatrix} \quad (15)$$

$$S_2 = [210 \quad 51 \quad 204] \quad (16)$$

$$error\_S_2 = [141 \quad 18 \quad 1] \quad (17)$$

$$M = \begin{bmatrix} m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 \\ m_7 & m_8 & m_9 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad (18)$$

In fact the set 'S<sub>2</sub>' contains the color A which is the color of the background, since that color is confused with the color B of the left '1' in the case of a protanope.

Thereafter the first iteration of the proposed daltonization algorithm is executed. The 'error\_S<sub>2</sub>' matrix (17) is subjected to the daltonization routine as well as the original values of the 'M' matrix (18). After daltonization the set 'S<sub>2</sub>' is renamed into 'S<sub>3</sub>' and its simulation is named 'S<sub>3\_sim</sub>'.

$$S_3 = [69 \quad 210 \quad 255] \quad (19)$$

$$S_{3\_sim} = [194 \quad 194 \quad 254] \quad (20)$$

As we can see after the first iteration the color of the background is changed into (69,210,255). This color is perceived by a protanope as (194,194,254) very close to the color of the middle '1', which is now, as shown in figure 1e, not visible.

At this point the rgb similarity checking routine examines if there is any similarity between the one and only element (194,194,254) of 'S<sub>3\_sim</sub>' and the elements of the set 'S<sub>1</sub>'. The routine detects similarity with the element (193,193,255) of the set 'S<sub>1</sub>'. It is obvious that if the step of the rgb similarity checking routine was omitted then the final image would be that of figure's 1e and the middle '1' would be invisible by a protanope. The parameters of the 'M' matrix are modified, as shown in (21). Thereafter comes the second iteration of the algorithm, by subjecting to the daltonization routine whether the whole 'error\_S<sub>2</sub>' matrix (Method 1) or only that row of the matrix which shows the similarity (Method 2). To the daltonization routine we also subject the new matrix 'M' with the modified parameters (21).

$$M_2 = \begin{bmatrix} m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 \\ m_7 & m_8 & m_9 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0.95 & 1 & 0 \\ 1.05 & 0 & 1 \end{bmatrix} \quad (21)$$

The results of the second iteration are described in the following:

$$S_3 = [69 \quad 203 \quad 255] \quad (22)$$

$$S_3 - sim = [188 \ 188 \ 254] \quad (23)$$

After the second iteration the color of the background is transformed into (69,203,255). This color is perceived by a protanope as (188,188,254). However, once again there is no significant difference compared to the color C of the middle '1' which, as shown in figure 1f, remains indistinguishable.

Once more the rgb similarity checking routine is activated and examines whether the color (188,188,254) of the new matrix ' $S_{3\_sim}$ ' is noticeably different from the elements of the set ' $S_1$ '. Once more a similarity with the element (193,193,255) of the set ' $S_1$ ' is detected. We should remind that according to the rgb similarity checking routine as similar we consider all the color combinations for which stands  $(r \pm 10, g \pm 10, b \pm 10)$ . The parameters of matrix ' $M$ ' are once more adapted as shown in equation (24) which results to the third iteration of the algorithm.

$$M_3 = \begin{bmatrix} m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 \\ m_7 & m_8 & m_9 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0.9 & 1 & 0 \\ 1.1 & 0 & 1 \end{bmatrix} \quad (24)$$

The results of the third iteration of the algorithm are:

$$S_3 = [69 \ 196 \ 255] \quad (25)$$

$$S_3 - sim = [182 \ 182 \ 254] \quad (26)$$

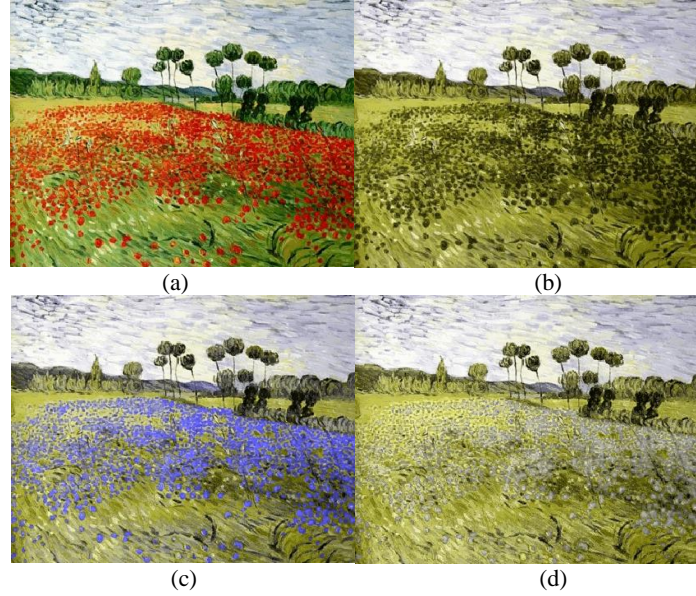
As we see, after the third iteration of the proposed algorithm, the color of the background (A) is changed into the color (69,196,255) which a protanope perceives as (182,182,254).

Once more the rgb similarity checking routine examines if the shade (182,182,254) of ' $S_{3\_sim}$ ' matrix is considerably different from the elements of the set ' $S_1$ '. This time fortunately no similarities are found and the algorithm continues with the reconstruction of the color palette 'map' by replacing the elements of the set ' $S_2$ ' with the elements of the set ' $S_3$ '. In this way the palette 'map<sub>1</sub>' of the final image is produced (27) which is perceived by a protanope as shown in (28).

$$map_1 = \begin{bmatrix} 69 & 196 & 255 \\ 193 & 193 & 255 \\ 73 & 73 & 203 \\ 255 & 255 & 255 \end{bmatrix} \quad (27)$$

$$Final\_sim = \begin{bmatrix} 182 & 182 & 254 \\ 193 & 193 & 255 \\ 73 & 73 & 203 \\ 255 & 255 & 255 \end{bmatrix} \quad (28)$$

Figure 2 demonstrates the result of applying both methods in Van Gogh's "Field of Poppies". It is evident that Method 1 has more a distinguishable effect from Method 2 and therefore it is considered more adequate to model the color difference for the specific image.



**Fig. 2.** For quantization to 256 colors: (a) Initial image, (b) Protanope perception, (c) Method 1: Protanope perception, (d) Method 2: Protanope perception.

## 4 Experimental results

The implementation of the proposed algorithm resulted in a very good computational time, which was drastically reduced compared to that of [5]. Indicatively, in an Intel Core Duo, 2 GHz, 2 GB, each iteration of the proposed algorithm's Method 2 in MATLAB when applied in a 24 bit 400x300 image with quantization to 256 colors, lasts about 0.04 seconds.

Since the computational time show up to  $15 \times 10^{-3}$  variance in each execution of the algorithm and in order to determine more clearly if it is suitable for real-time processing in image sequences with the PAL/SECAM standard (25 frames/sec), a function that executes the algorithm 25 times iteratively was created. The results show that the application of Method 2 of the proposed algorithm with quantization to 256 colors, can achieve real-time processing in a live broadcasted video, as for the processing of 25 frames they remain under the threshold of 1 second.

From these results we also find that Method 2 of the proposed algorithm is slightly faster than Method 1 and this is especially evident in quantization with more than 128 colors. This is expected as in Method 1 many matrix multiplications are performed at

each iteration of the rgb similarity checking module, in contrast to Method 2 where in each iteration only one multiplication is performed. Moreover, Method 1 after each iteration requires simulation of the entire ' $S_3$ ' matrix, since the entire ' $S_2$ ' changes, while in Method 1 simulation needs to be performed only at one row of ' $S_3$ ' matrix, namely the one that corresponds to that row in ' $S_2$ ' which was modified.

Consequently, the minimization of the overall processing time in order to establish real time processing is successfully achieved.

The proposed algorithm could be modified in the future so after the initial daltonization of the image, the iterative daltonization of the RGB similarity checking module would only be performed if there is similarity and adjacency of relatively large areas among similar colors of ' $S_1$ ' and ' $S_{3\_sim}$ '.

In conclusion, the proposed algorithm presents several desirable features such as: (i) It is fast and deterministic (i.e. it always produces the same result for a given input image), (ii) It satisfies a global consistency property (i.e. all pixels of the same color in the original image will be mapped to the same recoloration in the final image), (iii) It preserves the original image luminance and natural appearance (i.e. false colors are avoided) and (iv) It can be efficiently implemented on modern GPUs.

## 5 Conclusions

In this paper an improved daltonization algorithm for protanopia is proposed. An intelligent iteration technique is suggested for the selection of the adaptation parameters, incorporating a color quantization technique. Computational time is drastically reduced. Consequently, the minimization of the overall processing time in order to establish real time processing in live broadcasted/streaming videos is successfully achieved.

## References

1. Nam J., Ro Y.M., Huh Y., Kim M., "Visual Content Adaptation According to User Perception Characteristics", IEEE Trans. Multimedia, vol. 7, no. 3, 2205, pp. 435-445.
2. Yang S., Ro Y. M., "Visual contents adaptation for color vision deficiency, Image Processing", Proc. Of Int. Conf. on Image Proc., vol. 1, 2003, pp. I - 453-6.
3. Kovalev V., "Towards Image Retrieval for Eight Percent of Color-Blind Men", Proc. Of 17th Int. Conf. on Pattern Recognition (ICPR'04), vol. 2, 2004, pp. 943 - 946.
4. Martin C. E., Keller J. G., Rogers S. K. and Kabrisky M.: Color Blindness and a Color Human Visual System Model, IEEE Trans. on Systems, Man, and Cybernetics—Part A: Systems and Humans, vol. 30, No. 4, (2000) 494-500
5. Anagnostopoulos C., Anagnostopoulos I., Tsekouras G., Kalloniatis C., "*Intelligent modification for the daltonization process of digitized paintings*", Proc. of 5<sup>th</sup> International Conference on Vision Systems, ICVS'07, Bielefeld, Germany, 21-24 March 2007, accessible on line: <http://biecoll.ub.uni-bielefeld.de/volltexte/2007/52/>
6. Fidaner O, Poliang L., Ozguven N., "*Analysis of color blindness*". Available at: [http://scien.stanford.edu/pages/labsite/2005/psych221/projects/05/ofidaner/colorblindness\\_project.htm](http://scien.stanford.edu/pages/labsite/2005/psych221/projects/05/ofidaner/colorblindness_project.htm), last date of access: 21/06/2014.

7. Kuhn, G. R., “*An Efficient Naturalness – Preserving Image – Recoloring Method for Dichromats*”, IEEE Trans. On Visualization and Computer Graphics, Vol. 14, No. 6, November/December 2008, pp. 1747 – 1754.
8. Doliotis P., Tsekouras G., Anagnostopoulos C. A., Athitsos V., “Intelligent Modification of Colors in Digitized Paintings for Enhancing the Visual Perception of Color-blind Viewers”, Proc. of the 5<sup>th</sup> IFIP Conference on Artificial Intelligence Applications and Innovations (AIAI'2009), 2009.
9. Sgouroglou, P., “*Modification of Colors in Colored Images for Enhancing the Visual Perception of Protanopes*”, Thesis, Hellenic Open University, 2013.
10. Spencer W.T., “*Efficient Inverse Color Map Computation*”, Graphics Gems II, (ed. James Arvo), Academic Press: Boston. 1991.
11. Vienot F., Brettel H., Mollon J., “*Digital Video Colourmaps for Checking the Legibility of Displays by Dichromats*”, Inc. Col. Res. Appl., vol. 24, no.4, 1999, pp. 243-252