U.P.B. Sci. Bull., Series D, Vol. 69, No. 2, 2007 ISSN1453-2358

FROM WAVELENGTH TO R G B FILTER

Dragoş MIHAI[1], Eugen STRĂJESCU[2]

Lucrarea descrie implementarea unui filtru R G B bazat pe lungimea de undă a radiației luminoase. Pentru implementarea filtrului s-au folosit aproximații liniare pentru funcțiile de culoare CIE cu o bună aproximație în particular pentru domeniul roşu. Algoritmul se deosebeşte prin viteza bună de execuție şi capacitatea de implementare în aproape orice sistem.

The paper describes the implementation of an R G B filter based on the wavelength of illumination. For filter implementation linear approximations for the CIE color functions with a particular good approximation in the red domain have been used. The algorithm differs by a good execution velocity and by the capacity to be implemented almost in any virtual system.

Keywords: R G B, wavelength, filter, image processing, color function

1. Introduction

During the development of graphical processing of information has been established that it is very important to define a set of rules for conversion between the wavelength of a reflected light beam on various surfaces and the internal representation of colors on graphical devices.

The image acquisition process is complicated and in most cases the result is a numerical representation into a computer memory of an array, with the signification of red, green and blue. Up to the acquisition of this array there is nothing that can be done by the user to interfere with the acquisition process. Apart some external hardware filters, the user is a passive observer of the image acquisition process. After the R G B array acquisition, the user can start the complicated process of filtering and processing the collected data, so one can be sure that the obtained data are coherent with the investigated physical model. The base techniques for image manipulation in order to process data and get the useful information refer to a filtration on certain wavelengths and separation on wavelength intervals to eliminate image acquisition noises or environmental noises. For certain, this process needs functions or algorithms between light wavelength and the R G B code.

[1] Eng., ADG Design SRL, Bucharest, ROMANIA
[2] Prof., Dept. of Machine Tools and Production Systems, University "Politehnica" of Bucharest, ROMANIA

2. Theoretical considerations

A color model is a way to represent the colors and the relations between colors. Different color processing systems use different representations based on technical and economical considerations. The printing process uses the C M Y model. Monitors and most computerized graphical systems use the R G B model. Systems that manipulate mostly brightness, saturation and contrast use the H S I model.

Primary colors are rather associated with biological than physical process. The seeing is based on physiological response of human eye to light. The human eye contains photoreceptors (light sensitive cells) called cones which normally react to yellow-green colors (long wavelengths), green (medium wavelengths) and blue-violet (short wavelengths) which correspond to wavelengths spikes around 564 nm, 534 nm and 420 nm. The difference between signals that come from these three kinds of colors allows the brain to interpret a great variety of colors.

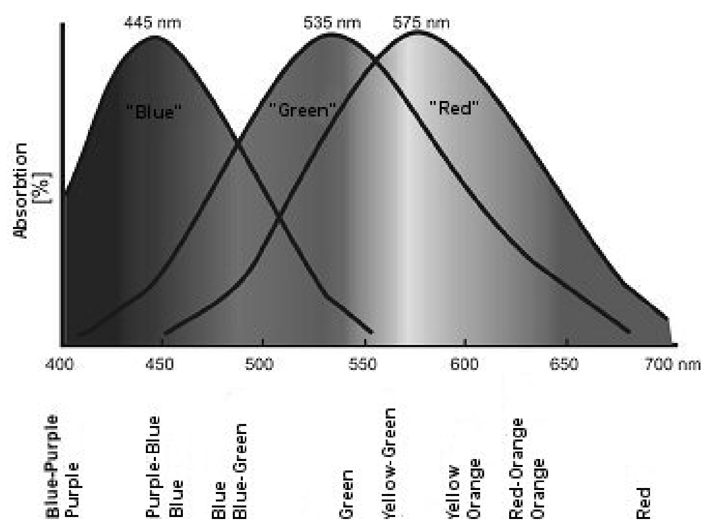

Fig. 1. Spectrum

In the twenties, David Wright and John Guild have independently conducted a series of experiments on human sight that ended in the elaboration of the specifications and parameters of the CIE XYZ color model. The experiments have been conducted using a circular beam with two active areas, projected on a screen so that the targeted area was approximately 2º. On one side of the field a sample color was projected while on the other side of the field a color was projected that could be changed by the observer. The adjustable color was made as a blend of three primary colors, each with a fixed chromatic, but with adjustable brightness. The observer's task was to modify the brightness of each of these three primary colors till he obtained a color identical with the sample one.

Though, not all of the colors in the spectrum can be synthesized by this method. When necessary, a variable quantity of the primary colors must be added to the composed color and the right combination with the other sample colors must be found. In these cases, the quantity of primary color added to the sample color was considered to have a negative value. By many experiments of this kind, the whole spectrum perceived by the human eye can be covered. For monochromatic colors, a correlation between the radiation's wavelength and the color can be established.

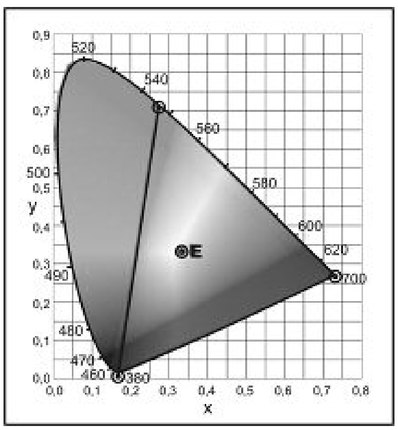

Fig. 2. Colour representation

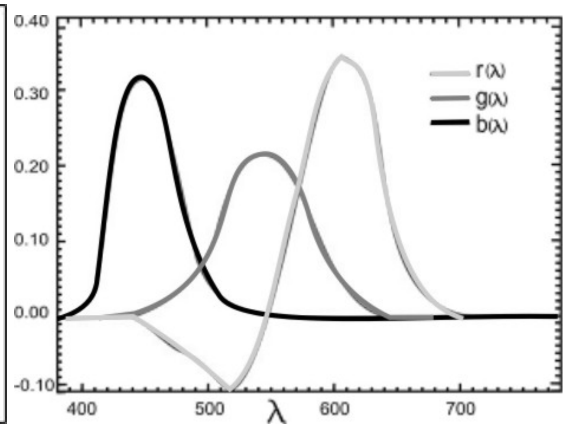

Fig. 3. Colour representation

The experiments of Wright and Guild have been conducted using various primary colors and different observers, so their results could be standardized by the "Commission Internationale d'Eclairage" (CIE), which defined a series of charts and functions for color definition, more specifically $r(\lambda)$, $g(\lambda)$, $b(\lambda)$. Fig. 2 shows null values on the chart at 435.8 for $r(\lambda)$ and $g(\lambda)$, at 546.1 for $r(\lambda)$ and $b(\lambda)$ and at 700 nm for $g(\lambda)$ and $b(\lambda)$. These color functions represent the quantity of each of the three standard monochromatic colors necessary to equal the sample color at the wavelength on the x axis. The three primary colors were standardized at the wavelengths of 700 nm (red), 546.1 nm (green) and 435.8 nm (blue). The last two were chosen as being easy to reproduce as monochromatic lines from a mercury vapor lamp. The wavelength of 700 nm, mode difficult to obtain at that time, was chosen because it represents the peak of the response of human eye to red light (small errors in the reproduction of this wavelength are not detected by the human eye).

The color functions were chosen by CIE after long debates. The left and the right limits of the spectrum were arbitrary chosen. The human eye can detect light with wavelengths up to 810 nm, but with a sensitivity thousands times smaller that the one for green light.

The R G B model for color representation uses the additive process, where the base components – red, green and blue – are combined in different ways in order to create the other colors.

The basic idea for the model itself and for the R G B abbreviation comes from the three primary colors in the additive color model.

The discussed application for the R G B color model is the recomposition of the colors on the normal computer screen. Each pixel can be represented in the memory as different numerical values for the red, green and blue levels. These values are converted into intensities.

The R G B model is represented by a three-dimensional cube having red, green and blue at the corners of each axis (fig. 2).

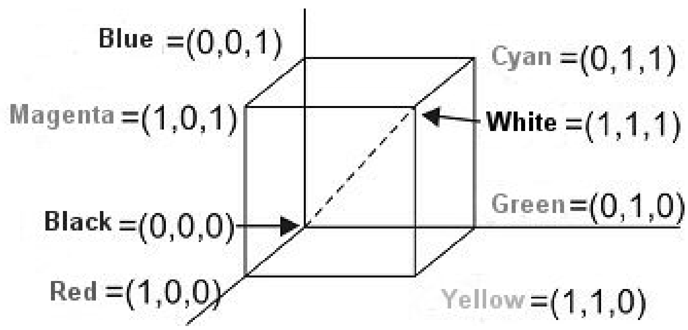

Fig. 2. RGB Cube

Black is the origin. White is in the opposed corner. The grayscale is represented on the black – white diagonal. In the 24 bit representation system, with 8 bit channels, red is (255, 0, 0). On the R G B cube this is (1, 0, 0).

The main factor that allows to operate with colors, respectively the addition of colors, is the almost linear interpretation of the human eye of color perception. This linearity is expressed by Grassmann's law: if the sample color is a combination of two monochromatic colors, then the value found by the observer for each monochromatic base color from the sample color will be the sum of each sample base color of the two monochromatic colors observed separately. In other words, if two rays of monochromatic color (1 and 2) are blended then, if the observer chooses as base color intensities for the color 1 the values R_1 G_1 B_1 and for color 2 the values R_2 G_2 B_2 then for the composed color the R G B values will be:

$$R = R_1 + R_2 \; ; \; G = G_1 + G_2 \; ; \; B = B_1 + B_2$$

In a more general way, Grassman's law, for a given color with a spectral distribution power $I(\lambda)$, can be expressed by the R G B coordinates given by:

$$R = \int_0^{\infty} I(\lambda) r(\lambda) d\lambda ; \quad G = \int_0^{\infty} I(\lambda) g(\lambda) d\lambda ; \quad B = \int_0^{\infty} I(\lambda) b(\lambda) d\lambda$$

3. Implementation of the algorithm

The implementation of the algorithm was made on the basis of a table containing R G B values, created by the following R G B functions of the wavelength.

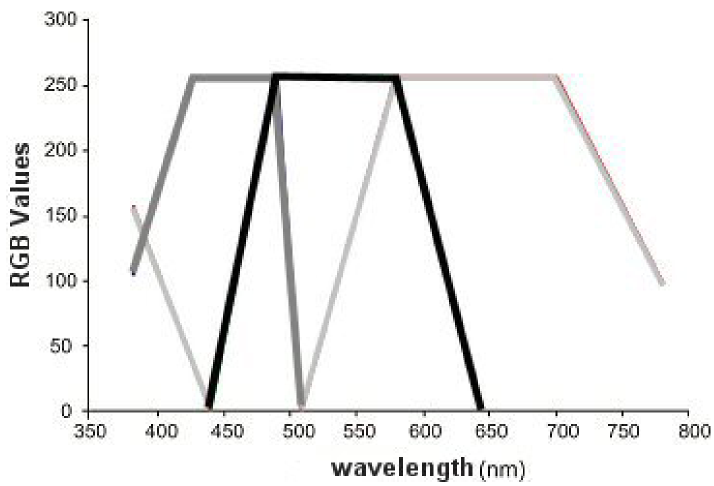

Fig. 4. Wavelength composition

Based on the above approximations, there were defined relations according to the following table (for each interval a set of rules to define the R G B, for a full visible spectrum between 380 nm and 780 nm):

Wavelength interval	R G B
380 – 410 nm	$R = 0.6 - 0.41 \cdot \frac{410 - WaveLength}{30}$ $G = 0$ $B = 0.39 + 0.6 \cdot \frac{410 - WaveLength}{30}$

Wavelength interval	**R G B**
410 – 440 nm	$R = 0.19 - 0.19 \cdot \frac{440 - WaveLength}{30}$ $G = 0$ $B = 1$
440 – 490 nm	$R = 0$ $G = 1 - \frac{490 - WaveLength}{50}$ $B = 1$
490 – 510 nm	$R = 0$ $G = 1$ $B = \frac{510 - WaveLength}{20}$
510 – 580 nm	$R = 1 - \frac{580 - WaveLength}{70}$ $G = 1$ $B = 0$
580 – 640 nm	$R = 1$ $G = \frac{640 - WaveLength}{60}$ $B = 0$
640 – 700 nm	$R = 1$ $G = 0$ $B = 0$
700 – 780 nm	$R = 0.35 - 0.65 \cdot \frac{780 - WaveLength}{80}$ $G = 0$ $B = 0$

The implementation of the algorithm is straightforward. It can be followed in the next table, where we have presented the code in the left column and the explanations in the right column. The algorithm was used for the interactive definition of filters for images acquired by a CCD camera. For interactivity and real-time processing of images a multi-thread code was used.

float R = 0.0;
float G = 0.0;
float B = 0.0;
float WaveLength = 480.0;
char x[11];
memset(x, 0, 11);
m_filter.GetWindowText(x, 10);
WaveLength = atoi(x);
if ((WaveLength >= 380.0) && (WaveLength <= 410.0)) {
R =0.6-0.41*(410.0-WaveLength)/30.0;
G = 0.0;
B = 0.39+0.6*(410.0-WaveLength)/30.0;}
if ((WaveLength >= 410.0) && (WaveLength <= 440.0)) {
R =0.19-0.19*(440.0-WaveLength)/30.0;
G = 0.0;
B = 1.0;}
if ((WaveLength >= 440.0) && (WaveLength <= 490.0)) {
R =0;
G = 1-(490.0-WaveLength)/50.0;
B = 1.0;}
if ((WaveLength >= 490.0) && (WaveLength <= 510.0)) {
R =0;
G = 1;
B = (510.0-WaveLength)/20.0;}
if ((WaveLength >= 510.0) && (WaveLength <= 580.0)) {
R =1-(580.0-WaveLength)/70.0;
G = 1;
B = 0;}
if ((WaveLength >= 580.0) && (WaveLength <= 640.0)) {
R =1;
G = (640.0-WaveLength)/60.0;
B = 0;}
if ((WaveLength >= 640.0) && (WaveLength <= 700.0)) {
R =1;
	/* R G B variables initialization*/
	
	/* wavelength variable initialization initializing reading of wavelength from the user */
	/* linear variation red and blue between 380 and 410 nm, green at zero */
	
	/* linear variation red between 410 and 440 nm, blue at maximum, green at zero */
	/* linear variation green between 440 and 490 nm, blue at maximum, red at zero*/
	/* linear variation blue between 490 and 510 nm, green at maximum, red at zero */
	/* linear variation red between 510 and 580 nm, green at maximum, blue at zero */
	/* linear variation green between 580 and 640 nm, red at maximum, blue at zero */
	/* between 640 and 700 nm red at maximum, green at

G = 0; B = 0;}	zero, blue at zero */
if ((WaveLength >= 700.0) && (WaveLength <= 780.0)) { R =0.35+0.65*(780.0-WaveLength)/80.0; G = 0; B = 0;}	/* linear variation red between 700 and 780 nm, green at zero, blue at zero */

if (picture) {
mCB->SetBitmapData(p_width, p_height, p_colordepth, picture);
mCB->Filter(R, G, B);
mCB->DisplayCapturedBits(&hwndStill);
set3DBuffer();}
	/* update in the picture after the R G B colors */

4. Conclusions

From the point of view of the implementation on computers, no image acquisition process is complete if there are no means of manipulating and quantification the color information. When it is implemented into an application that acquires images from a source in the red domain, the algorithm allowed the real-time filtration of environmental influences, more specifically, it allowed the selection of image information on the wavelength chosen by the user. Though not very accurate, especially in the domain purple – blue where the approximations largely differ from the CIE color functions, the algorithm based on the approximations is more close to reality in the red domain, where the approximation speed is also the greatest. The experimental results obtained confirmed the possibilities of using this algorithm and approximations.

In the case in which there are no critical execution times, the CIE color functions can be approximated on intervals by more complex functions and an algorithm with better response no matter the wavelength can be implemented.

R E F E R E N C E S

[1]. *D., Malacara*, Color vision and colorimetry: Theory and applications, SPIE Press, 2002.

[2]. *G., Wyszecki, W. S. Stiles*, Color Science : Concepts and Methods, Quantitative Data and Formulae, Wiley Series, 2000.

[3]. *** The basis of physical photometry, 2nd edition, CIE, 1983.

[4]. *** An analytic model for describing the influence of lighting parameters upon visual performance, 2nd edition, **Vol. 1** – Technical foundation, CIE, 1981.

[5]. *** An analytic model for describing the influence of lighting parameters upon visual performance, 2nd edition, **Vol. 2** – Summary and application guidelines, CIE, 1981.