

Acceleration of Image Processing Using New Color Model

¹Mahdi Alshamasin, ²Riad Al-kasasbeh, ²A. Khraiwish,
²Y. Al-shiboul and ³Dmitriy E. Skopin

¹Department of Mechatronics,

²Department of Electrical Engineering,

³Department of Computer Engineering,

Faculty of Engineering Technology, Al-Balqa'a Applied University,
P.O. Box 15008 Amman 11134, Jordan

Abstract: The theoretical outcomes and experimental results of new color model designed for accelerated image processing together with implementations of designed model to software of image processing are presented in the study. This model, as it will be shown below, may be used in modern real time video processing applications. The developed model allows to increase a speed of image processing algorithms using accelerated model of conversion to new decorrelated color space. Experimental results the proposed method can get a better performance than other existing methods.

Key words: Image processing, color model, RGB, HSV and HSI

INTRODUCTION

Digital image processing is a new and promptly developing field which finds more and more application in various information and technical systems such as: radar-tracking, communications, television, astronomy, etc. ^[1]. There are numerous methods of digital image processing techniques such as: Histogram processing, local enhancement, smoothing and sharpening, color segmentation, a digital image filtration and edge detection. Initially, these methods were designed specially for grey scale image processing^[2,3]. The RGB color model is standard design of computer graphics systems is not ideal for all of its applications. The red, green and blue color components are highly correlated. This makes it is difficult to execute the image processing algorithms. Many processing techniques and on the intensity component of an image only. These processes are standard implemented using the HSI or HSV color model that are not ideal in terms of performance.

The main target of this study is to develop a new color model that can be used in modern real time video processing applications. Achieving such a goal requires presenting this study which is organized from five sections. In section two, we describe some type of color model that is used in image processing dedicated to work on the intensity component of an image only. Section three deals with developing a new methodology

for object recognition. In section four, we develop an advanced approach for image processing that satisfies the target of this study. Results and conclusion are shown in section five and six correspondently.

MATERIALS AND METHODS

Image processing using standard color model: The data flow of standard color image processing is shown in Fig. 1. Any color image is a bitmap^[1], where each pixel represented as R, G, B color components. After reading pixel in standard algorithm of image processing it will be converted to other color model (step 2 in Fig. 1), where two components representing color pixel properties (for standard HSI model it's hue and saturation respectively), third one carrying intensity of current pixel. All methods of image processing deal especially with intensity components ^[1-3,4] (to change I component of new color model, step 3). After image processing, for results representation it's necessary to convert pixel to RGB model because only this format has hardware support (step 4).

The HSI color model, as it was mentioned before, is standard of color image processing. The idea of this model implementation is not so easy; usually this standard can be represented graphically in hue, saturation and intensity in 3 dimensional space^[3,4] as shown in Eq. 1-4.

Corresponding Author: Mahdi Salman Alshamasin, Department of Mechatronics, Faculty of Engineering Technology, Al-Balqa'a Applied University, P.O. Box 15008, Amman 11134, Jordan
Tel: +96265606017/+962777491552

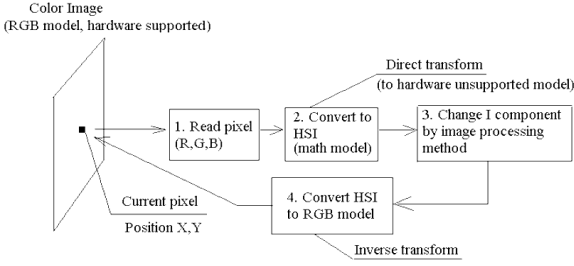


Fig. 1: The data flow of standard color image processing

The following equations show conversions from RGB space to HSI:

$$I = \frac{1}{3}(R + G + B) \tag{1}$$

$$S = 1 - \frac{3}{(R + G + B)}[\min(R, G, B)] \tag{2}$$

$$H = \cos^{-1} \left\{ \frac{0.5[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right\} \tag{3}$$

$$\text{If } B \text{ is greater than } G, \text{ then } H = 360^0 - H \tag{4}$$

where, R, G and B are three color component of source RGB image. H, S and I it's components of hardware unsupported HSI format.

As we can see, conversion from RGB to HSI is not optimal in terms of computing algorithm complexity because it's regarding minimum from three searching (expression 1, as minimum two operators of condition), external cosine function, square root, square computation, additional operation of condition (expression 4) during one pixel conversion. But such more difficulties to convert from HSI color space back to standard RGB, where the process depends on which color sector H lies in. For the RG sector ($0^0 \leq H \leq 120^0$), we have the following equations to convert RGB to HSI format:

$$B = I(1-S) \tag{5}$$

$$R = I \left[1 + \frac{S \cos H}{\cos(60^0 - H)} \right] \tag{6}$$

$$G = 3I - (R+B) \tag{7}$$

For the GB sector ($120^0 \leq H \leq 240^0$):

$$H = H - 120^0 \tag{8}$$

$$R = I(1-S) \tag{9}$$

$$G = I \left[1 + \frac{S \cos H}{\cos(60^0 - H)} \right] \tag{10}$$

$$B = 3I - (R+G) \tag{11}$$

For the BR sector ($240^0 \leq H \leq 360^0$):

$$H = H - 240^0 \tag{12}$$

$$G = I(1-S) \tag{13}$$

$$B = I \left[1 + \frac{S \cos H}{\cos(60^0 - H)} \right] \tag{14}$$

$$R = 3I - (G+B) \tag{15}$$

In point of view of computer performance inverse transformations represented in Eq. 5-15 meet additional condition operation to choose sector, floating point arithmetics and call of external cosine function, that means standard model for image processing is not optimal in terms of speed taking into account a high resolution of images in modern photo and video applications (image size more than 2000x1000). There are some other color models among which possible to emphasize the^[2,3] HSV space (hue, saturation, value), HSL space (hue, lightness, saturation), the CMY color space, YCbCr color model^[5-8], RG color space^[8]. CMY, YCbCr and RG are intensity correlated models (like RGB) and can't be used for image processing

Development of a new approach for image processing: The development of a new method is caused by necessity to decrease the time of image processing and to provide the possibility of real video processing techniques. It is known that majority methods of image processing working only with intensity part of color model^[9-11].

Requirement to new model: new model must be faster to compare with a standard, has two components for pixel color properties and one component for intensity, calculation of intensity can be inherited from HSI model (Eq. 1) for algorithms compatibility.

Also new color model must be in full basis^[12,13], it mean that model should allow direct and inverse transformations, transform must be evaluated on its gamut and decorrelation properties.

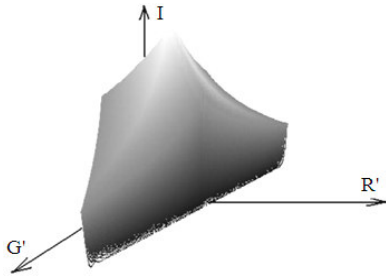


Fig. 2: The spatial configuration of new color model in 3D space (Perspective projection)

The pixel intensity calculation we inherited from HSI color model, calculations of color components from other color model widely known as "CIE 1931 color space"^[1,4] that was derived from a series of experiments done in the late 1920s by W. David Wright (Wright 1928) and John Guild (Guild 1931). In the CIE color model each color represented as three components, but the sum of it return constant one. Lets modify CIE model (exclude blue component and convert to integer arithmetic) and show equations for direct transforms:

$$R' = \frac{R \cdot 256}{R + G + B} \tag{16}$$

$$G' = \frac{G \cdot 256}{R + G + B} \tag{17}$$

$$I = \frac{R + G + B}{3} \tag{18}$$

Note: Multiplication by 256 is equivalent to shift left by 8 bits (fast logic operation), after shift we obtain integer division in equations 15-17 (operation "%" or "mod")

The graphical representation of R'G'I color model can be shown as cube in 3D color space as shown in Fig. 2.

The graphical representation look like p0079ramid (simulated by OpenGL), but it is rough evaluation because we can obtain here perspective projection only. Even in origin the value of light component can be non-zero since blue component, that removed from model may has non-zero value. The maximum value of intensity (255) is possible to obtain in position of grayscale axis (white color). Also in bottom of the model we can obtain non-solid elements or jets that appear because elaborated model operates with integer numbers and according to the equations 16 and 17

small values of R, G and B cause great increment of R' and G'.

Description of inverse transform (RGI to RGB) following:

$$R = \frac{3 \cdot R' \cdot I}{256} \tag{19}$$

$$G = \frac{3 \cdot G' \cdot I}{256} \tag{20}$$

$$B = \frac{3 \cdot (256 - R' - G') \cdot I}{256} \tag{21}$$

Note: Division by 256 is equivalent to shift right by 8 bits (fast logic operation), all operations can be executed in integer arithmetics.

Evaluation the transform on its gamut: Transforming from one space to another and processing in these spaces usually generate a gamut problem, i.e., the values of the variables may not be in their respective intervals. We study enhancement techniques for color images theoretically in a generalized setup. A principle is suggested to make the transformations gamut-problem free. Let's represent an example of simple image enhancement using R'G'I color space (Fig. 3).

Lets suppose that there is color pixel in some x,y position of image and color components of this pixel are R = 15, G = 50, B = 100 and we shall provide simple image processing named brightening (increment of brightness by 50 for example).The first operation is conversion from RGB color space to R'G'I according with equations: (16-18); second one is image processing, brightening 50, new intensity component become I = I+50 = 105; and the last one operation is inverse conversion to return the pixel to standard RGB model, according with equations 21-23 we can calculate that new pixel color is R = 28, G = 95, B = 192. Especially for this example no out of gamut problem, but it is only particular case of image enhancement. Gamut in transform space (R'G'I elaborated model) is out of our interest, because transform domain is mathematical representation of color and has infinite gamut range (all range of integer numbers). Gamut is interesting especially for inverse transform since RGB color space has hardware support with limited gamut (usually the range 0...255). Let's denote R_{max}, G_{max}, B_{max} to be a gamut of source RGB space, now we will apply transformations 16, 18 to show that after transformations gamut will be in range:

$$R' = \frac{R_{\max} \cdot 256}{R_{\max} + G_{\max} + B_{\max}}; I = \frac{R_{\max} + G_{\max} + B_{\max}}{3} \quad (22)$$

Equation 22 shows direct transform of gamut limited values in RGB color space to R'G'I color space. Now we can apply inverse transform according to Eq. 19.

$$R = \frac{3 \cdot \frac{256R_{\max}}{R_{\max} + G_{\max} + B_{\max}}}{256} \times \frac{R_{\max} + G_{\max} + B_{\max}}{3} = R_{\max} \quad (23)$$

Since R_{\max} in range of RGB gamut, result of transform also located in that range. Also it is possible to proof the same for G and B components. It's demonstrates that the transformations are gamut-problem free. But this fact is not quite enough, gamut problem can appear during image processing because after intensity correction any modification of I component after inverse transform can cause situation when some color component will be out of range. Let's use a same example as in Fig. 3, but coefficient of intensity increment supposes be equal 100:

- Source image: R = 15, G = 50, B = 100
- Direct transform (to RGI): R' = 23, G' = 77, I = 55
- Image processing: R' = 23, G' = 77, I = I+100 = 155
- Inverse transform: R = 41; G = 139; B = 283 > 255 out of gamut

To avoid gamut problem we designed a method of inverse transformation gamut correction (Fig. 4). Let's denote as R_{\min} and R_{\max} -gamut range of red color in RGB color space. Using equation 19 we will rewrite it as equality:

$$\frac{85 \cdot R_{\min}}{R'} \leq I \leq \frac{85 \cdot R_{\max}}{R'} \quad (24)$$

where, I and R' is intensity and red components in R'G'I color space respectively. Equation 24 describes all possible range of intensity component I. If equality is false, gamut problems will appear after inverse transform.

Since the range of $R_{\min} = 0$ and $R_{\max} = 255$ in most popular image formats (32 bits images) equality 24 can be rewritten as algorithm of gamut correction:

$$\text{if } (I > \frac{21760}{R'}) \text{ then } I = \frac{21760}{R'} \quad (23)$$

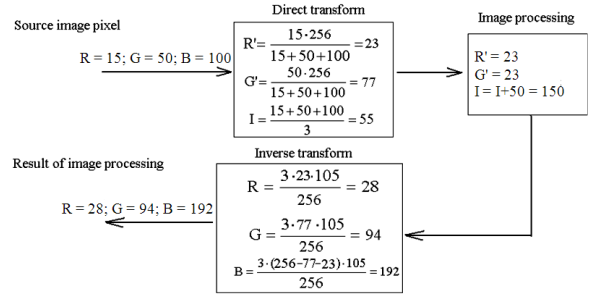


Fig. 3: Color image processing using elaborated color model



Fig. 4: Results of image processing (brightening+100) of color image. (a): Processing without correction (Corrupted image) and (b): Correction of gamut

$$\text{if } (I > \frac{21760}{G'}) \text{ then } I = \frac{21760}{G'} \quad (24)$$

$$\text{if } (I > \frac{21760}{256-R'-G'}) \text{ then } I = \frac{21760}{256-R'-G'} \quad (25)$$

This method allows checking the component of intensity and modifying it before applying of inverse transformation. To explain lets remove steps numbers 4 and 5 and for example apply inverse transform correction:

- for R: I = 155 < 944, OK
- For G: I = 155 < 283, OK
- For B: I=155 < 139: Overflow of Blue color; Use I = 139 (result of correction)
- Inverse transform: R = 38; G = 125; B = 255 (all colors in the range of it's gamut)

Comparison of image processing results with inverse transform gamut correction and without it represented in Fig. 4.

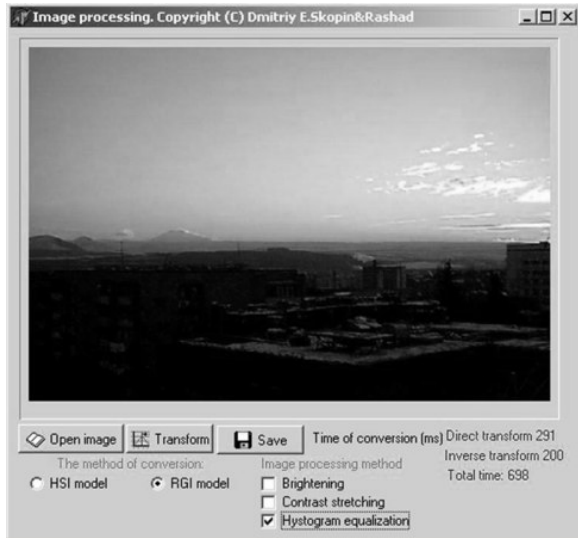


Fig. 5: Graphical user interface of the developed program

RESULTS

To study the practical approaches of new color model especially in image processing field the program implemented a RGI model and standard HSI model (for comparing) has been created with using Borland Delphi 6.0 compiler^[15,16]. The graphical user interface of program, that we created shown in Fig. 5. Produced by Borland International, Delphi is a powerful development environment used primarily to build client/server applications for Microsoft Windows. During program creation we have apply addition optimization of equations 16-18:

- The sum $S = R+G+B$ should be found ones per one transforming pixel together with ratio $1/S$
- Equation 16 and 17 was converted to left shift (8 bits), that is analog of multiplication by 256, then apply multiplication by $1/S$
- Division by 256 was represented as shift right (8 bits) in Eq. 19-21

As we can show in the Fig. 5, the program of image processing was created using both methods of conversion: standard HSI model and elaborated RGI. The program implemented such method of image processing as brightening, contrast stretching and histogram equalization; it's enough because our aim was evaluation of new model advantages. Both methods that used standard and designed produce the same visual results. To estimate a time of image conversion,

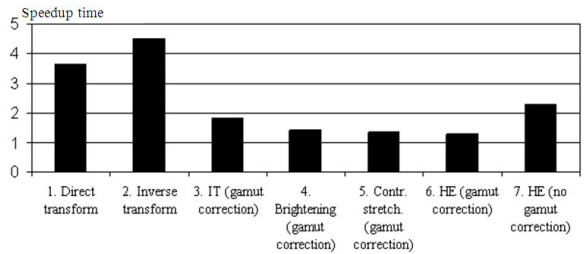


Fig. 6: The speedup of elaborated model

Table 1: Measured time of conversion

Test	R'G'I model (ms)	HSI model (ms)	Acceleration
Direct transform	280	1024	3.65
Inverse Transform (IT)	205	939	4.50
IT with gamut correction	315	939	1.84
Brightening with gamut correction	1546	2170	1.40
Contrast stretching (with gamut correction)	1680	2265	1.35
Histogram Equalization (HE) with gamut correction	1985	2568	1.29
HE without gamut correction	1116	2568	2.30

we create the test image represented as array (1000×2500 pixels) and filled by random pixels. After that, the time for conversion was measured. Results of evaluation are shown in the Table 1.

The same result but expressed as speedup (acceleration) as quotient of HSI conversion time over RGI conversion time shown in the Fig. 6:

DISCUSSION

The practical results show that the offered model at least few times faster in direct and inverse transform to compare with standard HSI color model. This acceleration attained owing to the fact that only integer arithmetics involved to transformations 16-21, also transformations are quite simple with minimum numbers of operations. Extremely good performance we can obtain for inverse transform (conversion from transform space to RGB brought acceleration around 450%), because in Eq. 19-21 no division (it is slow operation), inverse transform represented as integer multiplication and right logical shift. However other result we obtain after involving a gamut correction to calculations, as we can see acceleration reduced more than twice because during gamut correction we have three condition operations and three integer divisions (Eq. 22-24) per one pixel. But as it was mentioned before, gamut correction is important during increment of I component, for example if during image processing we have only decrement of image intensity we can

exclude gamut correction from algorithm. Also we can see that even gamut correction, speed up of image processing using new model has acceleration 29%) The data of Table 1 shows, that speed up of elaborated model decreasing during increment of computation difficulties of image processing algorithm. This fact can be explained by evaluation of speed up:

$$S = \frac{t_1}{t_2} = \frac{t_{c1} + t_p}{t_{c2} + t_p}$$

Speed up is ratio of image processing time for standard (HSI) model and offered (R'G'I) model. Image processing time is sum of conversion time t_c (direct and inverse) and time of image processing algorithm t_p . Last parameter is constant for both models, that's why during increment of t_p whole ratio decreased.

CONCLUSION

In this article we offer the new color model that can be used in color image processing techniques. The results of the test show that the time spent during HSI conversion method was approximately 4 times more to comparing with elaborated R'G'I color model. Implementation of new model to such methods of image enhancement as brightening, contrast stretching and histogram equalization can increase a speed of application up 400%. Carried out experimental results of RGI color model for image processing show their efficiency that allows to recommend to include developed method in software of modern digital image processing systems.

REFERENCES

1. Heijden, F., 1994. Image Based Measurement Systems. Wiley.
2. Gonzalez, R.C. and R.E. Woods, 1992. Digital Image Processing, Reading, Massachusetts. Addison-Wesley.
3. Castleman, K.R., 1996. Digital Image Processing. 2nd Edn., Prentice-Hall, Englewood Cliffs, New Jersey, USA.
4. Blake, R.E., 1987. Development of an incremental graph matching device. *Patt. Recog. Theor. Appl.*, 30: 355-366.
5. Pavlidis, T., 1982. Algorithms for Graphics and Image Processing. Computer Science Press.
6. Giardina, C.R. and E.R. 1988. Dougherty, Morphological Methods in Image and Signal Processing. Prentice-Hall, Englewood Cliffs, New Jersey, USA.
7. Boros, P. and R.E. Blake, 1995. The calculation of the aspect graph from a CAD model. *Proceedings of the 2nd Asian Conference on Computer Vision*, Singapore, pp: 852-863.
8. Dudgeon, D.E. and R.M. Mersereau, 1984. Multidimensional Digital Signal Processing. Prentice-Hall, Englewood Cliffs, New Jersey.
9. Papoulis, A., 1968. Systems and Transforms with Applications in Optics. McGraw-Hill, New York.
10. Russ, J.C., 1995. The Image Processing Handbook. 2nd Edn., CRC Press, Boca Raton, Florida.
11. Blake, R.E., 1994. Partitioning graph matching with constraints. *Patt. Recog.*, 27: 439-446.
12. Foley, J., A. van Dam, S. Feiner and J. Hughes, 1990. Computer Graphics: Principles and Practice, 2nd Edn., Addison-Wesley, Reading, MA.
13. Blake, R.E. and P. Boros, 1995. The extraction of structural features for use in computer vision. *Proceedings of the 2nd Asian Conference on Computer Vision*, Singapore.
14. Skopin, D.E., 2005. Optical Topographer for Diagnosis of Support-Locomotion System Diseases, medical and ecological information technology, Kursk, pp: 48-50.
15. Kerman, F. and C. Mitchell, 2001. Programming and Problem Solving With Delphi. International Edn., Pearson Higher Education, UK., ISBN: 0321204417, pp: 650.
16. Rachele, W., 2001. Learn Object Pascal with Delphi. Wordware Publishing Inc., ISBN: 1556227191.