International Journal of New Technology and Research (IJNTR)
ISSN: 2454-4116, Volume-3, Issue-3, March 2017 Pages 56-62

Color Image Processing in Digital Image

Dr. Mir Mohammad Azad, Md Mahedi Hasan , Mohammed Naseer K

***Abstract*— The use of color in image processing is motivated by two principal factors. First, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Color image processing is divided into two major areas: *full-color* and *pseudo-color* processing. In the first category, the images in question typically are acquired with a full-color sensor, such as a color TV camera or color scanner. In the second category, the problem is on of assigning a color to a particular monochrome intensity or range of intensities. Until recently, most digital color image processing was done at the pseudo color level. However, in the past decade, color sensors and hardware for processing color images have become available at reasonable prices. The result is that full-color image processing techniques are now used in a broad range of applications, including publishing, visualization, and the Internet.**

***Index Terms*—Image processing, Color Image, Digital Image.**

I. Introduction

Although the process followed by the human brain in perceiving and interpreting color is a physic psychological phenomenon that is not yet fully understood, the physical nature of color can be expressed on a formal basis supported by experimental and theoretical results.

In 1666, Sir Isaac Newton discovered that when a beam of sunlight passes through a class prism, the emerging beam of light is not white but consists instead of a continuous spectrum of colors ranging from violet at one end to red at the other. The color spectrum may be divided into six broad regions: violet, blue, green, yellow, orange, and red. When viewed in full color, no color in the spectrum ends abruptly, but rather each color blends smoothly into the next.

Basically, the colors that humans and some other animals perceive in an object are determined by the nature of the light reflected from the object. As illustrated in visible light is composed of a relatively narrow band of frequencies in the electromagnetic spectrum. A body that reflects light that is balanced in all visible wavelengths appears white to the observer, however, a body that favors reflectance in a limited range of the visible spectrum exhibits some shades of color. For example, green objects reflect light with wavelengths

primarily in the 500 to 570 nm range while absorbing most of the energy at other wavelengths.

Characterization of light is central to the science of color. If the light is achromatic (void of color), its only attribute is its *intensity*, or amount. Achromatic light is what viewers see on a black and white television set, and it has been an implicit component of our discussion of image processing. The term gray level refers to a scalar measure of intensity that ranges from black, to grays, and finally to white.

II. Color Models

The purpose of a color model (also called color space or color system) is to facilitate the specification of colors in some standard, generally accepted way. In essence, a color model is a specification of a coordinate system and a subspace within that system where each color model is a specification of a coordinate system and a subspace within that system where each color is represented by a single point. In terms of digital image processing, the hardware-oriented models most commonly used in practice are the RBG (red, green, blue) model for color monitors and a broad class of color video cameras; The CMY (clay, magenta, yellow) and CMYK (cyan, magenta Yellow, Black) models for color printings and the his (hue, saturation, intensity) model, which corresponds closely with the way humans describe and interpret color.

A. The RGB color Model

In the RGB model, each color appears in its primary spectral components of red, green, and blue. This model is based on a Cartesian coordinate system. The color subspace of interest is the cube shone in Figure: 4(a) in which RGB values are at three corners; cyan, magenta, and yellow are at three other corners; black is at the origin; and white is at the corner farthest from the origin.

Images represented in the RGB color model consist of three component images, one for each primary color. When fed into an RGB monitor, these three images combine on the phosphor screen to produce a composite color image. The number of bits used to represent each pixel in RGB space is called the *pixel depth.* Consider an RGB image in which each of the red, green, and blue images is an 8-bit image. Under these conditions each RGB *color* pixel [that is, a triplet of values (*R,G,B*)] is said to have a depth of 24 bits (3 image planes times the number of bits per plane). The term *full-color* image is used often to denote a 24- bit RGB color image. The total number of colors in a 24-bit RGB image is (2 to the power 8) whole cube = 16,777,216. RGB color cube corresponding to the diagram in Figure 1.

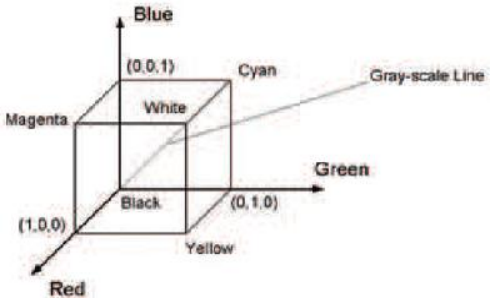


***Fig 1:** Schematic of the RGB color cube. Points along the main diagonal have gray values from black at the origin to white at point (1, 1, 1).*

Dr Mir Mohammad Azad, Department of Computer Science and Engineering, Department of Computer Science and Information Technology, Shanto Mariam University of Creative Technology, Dhaka, Bangladesh
Md Mahedi Hasan, Department of Business Administration, Prime University, Dhaka, Bangladesh,
Mohammed Naseer K, Department of Computer Science and Engineering , Maulana Azad National Urdu University, Campus Bangaluru, India,

Fig 2 shows that an image of the cross-sectional plane is viewed simply by feeding the three individual component images into a color monitor. In the component images, 0 represents black and 255 represents white (note that these are gray scale images).

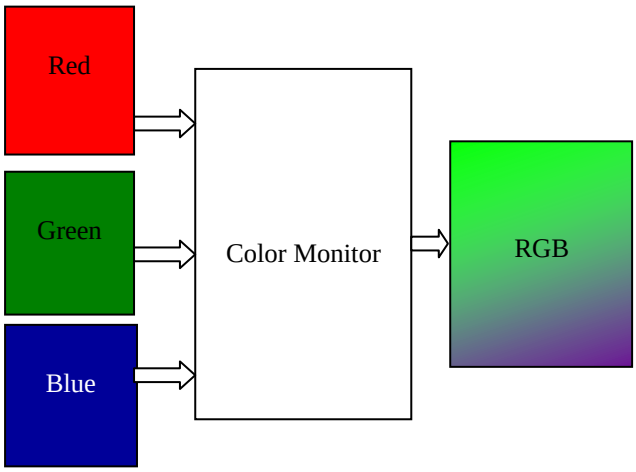
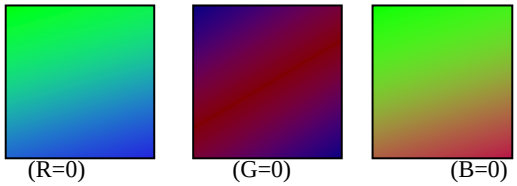


***Fig2: Generating** the RGB image of the cross-sectional color plane (127, G, B).*



*Finally **Fig 3:** shows the three hidden surface planes of the cube generated in the same manner.*

It is of interest to note that acquiring a color image is basically the process shown in **Fig 2, 3** in reverse. A color image can be acquired by using three filters, sensitive to red, green, and blue, respectively. When we view a color scene with a monochrome camera equipped with one of these filters, the result is a monochrome image whose intensity is proportional to the response of that filter

B. The CMY and AMYK color Models

As indicated cyan, magenta and Yellow are the secondary colors of light, alternatively, the primary colors of pigments. For example, when surface coated with cyan, pigment is illuminated with white light, no red light is reflected from the surface. That is, cyan subtracts red light from reflected white light, which itself is composed of equal amount of red, green and blue light.

Most devices that deposit colored pigments on paper. Such as color printers and copies require CMY data input or perform an RGB to CMY conversion internally.

This conversion is performed using the simple operation

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad (1)$$

Where, again, the assumption is that all color values have been normalized to the rang with pure cyan does not contain red (this is, C=1-R in the question).

Similarly, pure magenta does not reflect green and pure yellow does not reflect blue. Equation (1) also reveals that RGB values can be obtained easily from a set of CMY values by subtracting the individual CMY values from 1.

C. The HIS Color Model

The modal are about to present, called the HSI (hue, saturation, intensity) color model, decouples the intensity component from the color-carrying information (hue and saturation) in a color image. As a result, the HSI is an ideal tool for developing image processing algorithms based on color description that are natural and intuitive to humans, who after all the developers and users of these algorithms. We can summarized by saying that RGB is ideal for image color generation (as in image capture by a color camera or image display in a monitor screen).but its use for color descriptions much more limited.

III. IMAGE PROCESSING AND COLOR TRANSFORMATION

A. Pseudo color image processing

Pseudo color (also called as *false* color) image processing consists of assigning colors to gray values based on a specified criterion The term pseudo or false color is used to differentiate the process of assigning colors to monochrome image from the process associate with true color for human visualization and interpretation of gray-scale events in an image or sequence of image. One of the principal motivations for using color is the fact that human can discern thousands of color shades and intensities, compared to only two dozen or so shades of gray.

B. Gray level to color transformation

This transformation is more general and thus is capable of achieving a wider range of pseudo color enhancement result than the simple slicing technique discussed in the preceding section. An approach tat is particularly attractive is shown in **Fig 4** .Basically, the idea underling this approach is to perform three result are than fed separately into the red, green and blue channels of a color content is modulated by nature of the transformation function. Note that these are transformation on the gray-level values of image and are not function of position.

The method discussed in the previous section is a special case of the technique just described. There, piecewise linear function of the gray levels is used to generate colors. The method discussed in this section, on the other hand, can be based on smooth, nonlinear function, which as might be expected gives technique considerable flexibility.

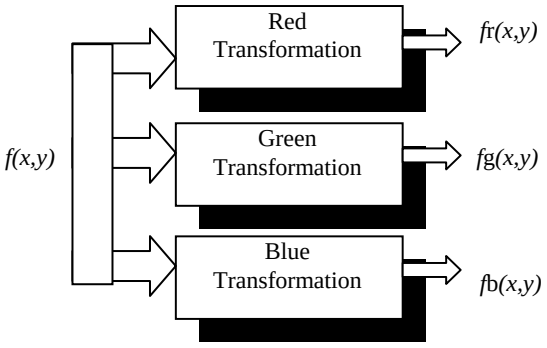


Fig 4: *Functional block diagram for pseudo color image processing fr, fg, and fb are fed into the corresponding red, green and blue inputs of an RGB color monitor.*

The type of processing just illustrated is quite powerful in helping visualized events of interest in complex image, especially when those events our beyond our normal sensing capabilities. **Fig 5** is an excellent illustration of this. These are image of the Jupiter moon Io, shown in pseudo color by combining several of the sensor images from the *Galileo* spacecraft, some of which are in spectral regions not visible to the eye. However, by understanding the physical and chemical processes likely to affect sensor response, it is possible to combine the sensed image into a meaningful pseudo color map.

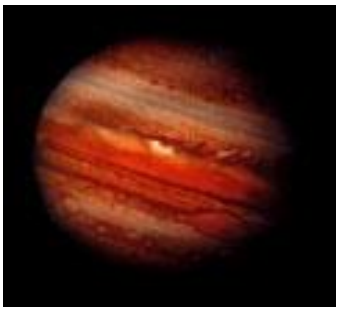


Fig 5: *pseudo color rendition Jupiter Moon Io.*

Fig 6: A *close-up (Courtesy of NASA)*

One way to combine sensed image data is by how they show either differences in surface chemical composition or change in the way the surface reflects sunlight. For example, in the pseudo color image in **Fig 6**, bright red depict material newly ejected from an active volcano on Io, and surrounding yellow materials are older sulfur deposits. This image conveys these characteristics much more readily than would be possible by analyzing the component images individually.

C. Basics of Full color image processing

In this section be being study of processing techniques applicable to full color image. Although they are far from being exhaustive, the techniques developed in the section follow are illustrative of how full-color image are handled for a variety of image processing tasks. Full-color image processing approaches fall into two major categories.

a) In first category, we process each component image individually and then from a composite processed color image from the individually processed components.

b) In Second category, we work with color pixels directly.

Because full-color image have at least three components, color pixels really are vectors. For example, in the RGB system, each color point in the RGB coordinates system (In figure RGB color Model).

Let $\boldsymbol{C}$ represent an arbitrary vector in RGB color space:

$$\boldsymbol{c} = \begin{bmatrix} Cr \\ Cg \\ Cb \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \qquad (1)$$

The equation indicates that the components of C are simply the RGB components of a color image at a point. We take into account the fact that the color components are a function of coordinates (x,y) by using the notation

$$C(x,y) = \begin{bmatrix} Cr(x,y) \\ Cg(x,y) \\ Cb(x,y) \end{bmatrix} = \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix} \qquad (2)$$

For an image of size $M \times N$,there are MN such vector, $C(x,y)$,for $x=0,1,2,.....M-1$; $y= 0,1,2,.....N-1$.

It is important to keep clearly in mind that Eq.(2) depicts a vector whose components are *spatial* variables in x and y. The fact that the pixels are now color pixels introduces a factor that, in its easiest formulation, allows us to process a color image by processing each of its components image separately, using standard gray-scale image processing method.

The results of individual color component processing are not always equivalent to direct processing in color vector space, in which case we must formulate new approaches.

D. Color Transformation

Color transformations; deal with processing the components of a color image within context of a single color model.

Formulation

As with gray-level transformation techniques of model color transformation using the expression

$$g(x,y) = T[f(x,y)] \qquad (1)$$

Where $f(x,y)$ is a color input image, $g(x,y)$ is the transformed or processed color output image, and T is an operator on f over a spatial neighborhood of $f(x,y)$.

The pixel values here are triplets or quartets (i.e., groups of three or four values) from the color space chosen to represent the images.

Analogous to the approach we used to introduce the basic gray-level transformations. We will restrict attention in this section to color transformations of the form.

$$si = T(r1,r2,.....,rn), \quad i = 1,2,.....,n \qquad (2)$$

where, for notational simplicity, ri and si are variables denoting the color components of $f(x,y)$ and $g(x,y)$ at any point (x,y), n is the number of color components, and $\{T1,T2,.....,Tn\}$ is a set of *transformation* or *color mapping functions* that operate on ri to produce si. Note that n transformations, Ti, combine to implement the single transformation function, T, in Equation (1). The color space chosen to describe the pixels of f and g determines the value of n. If the RGB color space is selected, for example, $n=3$ and $r1,r2$, and $r3$ denote the red, green, and blue components of the input image, respectively. If the CMYK or his color spaces are chosen, $n=4$ or $n=3$.

Fig 7 shows a high-resolution color image of a bowl of strawberries and cup of coffee that was digitized from a large format (4" x 5") color negative. The second row of the figure contains the components of the initial CMYK scan. In these images, black represents 0 and white represents 1 in each CMYK color component. This we see that the strawberries are composed of large amounts of magenta and yellow because the images corresponding to these two CMYK components are the brightest. Black is used sparingly and is generally confined to the coffee and shadows within the bowl of strawberries. When the CMYK image is converted to RGB, as shown in the third raw of the figure, the strawberries are seen to contain a large amount of red and very little (although some) green and blue. The last row or **Fig. 7** shows the components of **Fig. 7—**computed using Equation, (2) through (4). As expected, the intensity component is a monochrome rendition of the full-color original. In addition, the strawberries are relatively pure in color; they possess the highest saturation or least dilution by white light of any of the hues in the image, finally, we note some difficulty in interpreting the hue component.

The problem is compounded by the fact that;

- there is a discontinuity in the his model where 0° and 360° meet, and
- hue is undefined for a saturation of 0 (i.e., for white, black, and pure grays). The discontinuity of the model is most apparent around the strawberries, which are depicted in gray level values near both black (0) and white (1). The result is an unexpected mixture of highly contrasting gray levels to represent a single color—red.

Any of the color space components in **Fig. 7** can be used in conjunction with Equation (2). In theory, any transformation can be performed in any color model. In practice, however, some operations are better suited to specific models. For a given transformation, the cost of converting between representations must be factored into the decision regarding the color space in which to implement it. Suppose, for example, that we wish to modify the intensity of the image in **Fig. 7** using.

$$g(x, y) = kf(x, y) \qquad (3)$$

Where $0 < k < 1$. In the color space, this can be done with the simple transformation.

$$S3 = kr3 \qquad (4)$$

where $s1 = r1$ and $s2 = r2$. Only his intensity component $r3$ is modified. In the RGB color space, three components must be transformed:

$$si = kri \qquad i = 1,2,3. \qquad (5)$$

The CMY space requires a similar set of linear transformations:

$$si = kri + (1 - K) \qquad i = 1,2,3. \qquad (6)$$

Although the his transformation involves the fewest number of operations, the computations required to convert an RGB or CMY (K) image to the his space more than offsets (in this case) the advantages of the simpler transformation

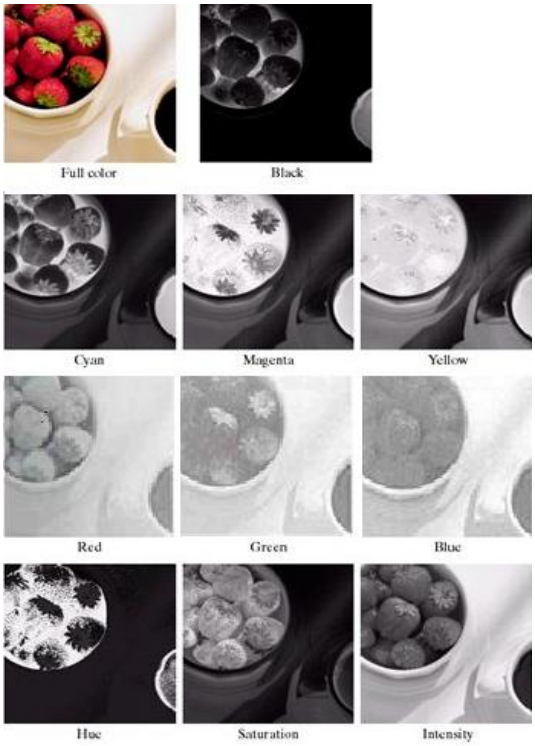


Fig 7: *A full-color image and its various color-space components (Original image courtesy Med-data Interactive.)*

The conversion calculations are more computationally intense than the intensity transformation itself. Regardless of the color space selected, however, the output is the same. **Figure 8** shows the result of applying any of the transformations in Equation (4) through (6) to the image of

Fig. 8 using $k = 0.7$.

Fig 8: Left side original image *and right one Result of decreasing its intensity by 30 %*

It is important to note that each transformation defined in Equation (4) through (6) depends only on one component within its color space.

For example, the red output component, *s1*, Equation (5) is independent of the green *(r2)* and blue *(r3)* inputs; it depends only on the red *(r1)* input. Transformations of this type are among the simplest and most used color processing tools and can be carried out on a per-color-component basis, as mentioned at the beginning of our discussion. In the remainder of this section we examine several such transformations and discuss a case in which the component transformation functions are dependent on all the color components of the input image and, therefore, cannot be done on an individual color component basis.

IV. Color Complements and Slicing

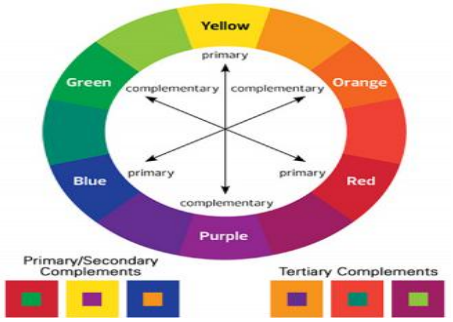


Figure 9: *Complements on the color circle*

The hues directly opposite one another on the *color circle*† of **Fig. 9** are called *complements.* Our interest in complement stems from the fact that they are analogous to the gray-scale negatives. As in the gray-scale case, color components are useful for enhancing detail that is embedded in dark regions of a color image—particularly when the regions are dominant in size.

Color Slicing

Highlighting a specific range of colors in an image is useful for separating objects from their surroundings.

Color slicing is either to

- display the colors of interest so that they stand out from the background or ,
- Use the region defined by the colors as a mask for further processing.

The most straightforward approach is to extend the gray-level slicing techniques. Because a color pixel is an *n*-dimensional quantity, however, the resulting color transformation functions are more complicated than their gray-scale counterparts. In fact, the required transformations are more complex than the color component transforms considered thus far. This is because all practical color slicing approaches require each pixel's transformed color components to be a function of all *n* original pixel's color components.

One of the simplest ways to "slice" a color image is to map the colors outside some range of interest to a non prominent neutral color. Neutral color. If the colors of interest are enclosed by a cube (or *hypercube* for $n > 3$) of width W and centered at a prototypical (e.g., average) color with components $(a1, a2,.....,an)$, the necessary set of transformations is

$$Si = \begin{cases} 0.5 & \text{if } [\,|rj - aj| > W/2\,] \text{ any } 1 \le k \le n \\ ri & \text{otherwise} \end{cases}, \; i = 1, 2,....., n. \quad (7)$$

These transformations highlight the colors around the prototype by forcing all other colors to the midpoint of the reference color space (an arbitrarily chosen neutral point). For the RGB color space, for example, a suitable neutral point is middle gray or color (0.5, 0.5, and 0.5).

If a sphere is used to specify the color of interest, Equation (7) becomes

$$\mathbf{Si} = \begin{cases} 0.5 & \text{if } \sum_{j=1}^{n} (rj - aj)^2 > R^2 0 \\ ri & \textbf{otherwise} \end{cases}, \; i=1,2,...,n. \quad (8)$$

Here, $R0$ is the radius of the enclosing sphere (or hyper sphere for $n > 3$)and $(a1, a2,....., an)$ are the components of its center (i.e., the prototypical color). Other useful variations of Equation (7) and (8) include implementing multiple color prototypes and reducing the intensity of the colors outside the region of interest—rather than setting them to a neutral constant.

V. Tone and Color Corrections

Color transformations can be performed on most desktop computers. In conjunction with digital cameras, flatbed scanners, and inkjet printers, they turn a personal computer into a *digital darkroom*—allowing tonal adjustments and color corrections, the mainstays of high-end color reproduction systems, to be performed without the need for traditionally outfitted wet processing (i.e., dark-room) facilities. Although tone and color corrections are useful in other areas of imaging, the focus of the current discussion is on the most common uses-photo enhancement and color reproduction.

The effectiveness of the transformations examined in this section is judged ultimately in print. Since these transformations are developed, refined, and evaluated on monitors, it is necessary to maintain a high degree of color consistency between the monitors used and the eventual output devices. In fact, the colors of the monitor should represent accurately any digitally scanned source images, as well as the final printed output. This is best accomplished with a *device-independent color model* that related the color gamut's of the monitors and output devices, as well as any other devices being used, to one another. The success of this approach is a function of the quality of the *color profiles* used to map each device to the model and the model itself. The model of choice for many *color management system* (CMS) is the CIE *L*a*b* model, also called CIELAB (CIE [1977]). THE *L*a*b* color components are given by the following equations:

$$L^* = 116.h(Y/YW) - 16 \quad (9)$$

$$a^* = 500[\,h(X/Xw) - h(Y/YW)\,] \quad (10)$$

$$b^* = 200[\,h(Y/Yw) - h(Z/ZW)\,] \quad (11)$$

where

$$h(q) = \begin{cases} \sqrt[3]{q} & q > 0.008856 \\ 70787q + 16/116 & q > 0.008856 \end{cases} \quad (12)$$

and Xw , Yw, and Zw are reference white tri stimulus values—typically the white of a perfectly reflecting diffuser under CIE standard D65 illumination (defined by x = 0.3127 and y = 0.3290 in the CIE chromaticity diagram of **Fig 10**.

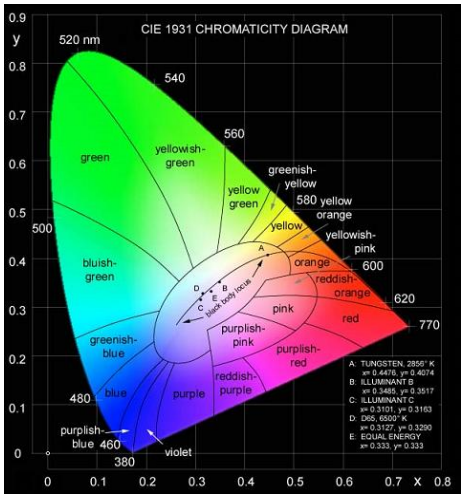


Fig. 10: *chromaticity diagram*

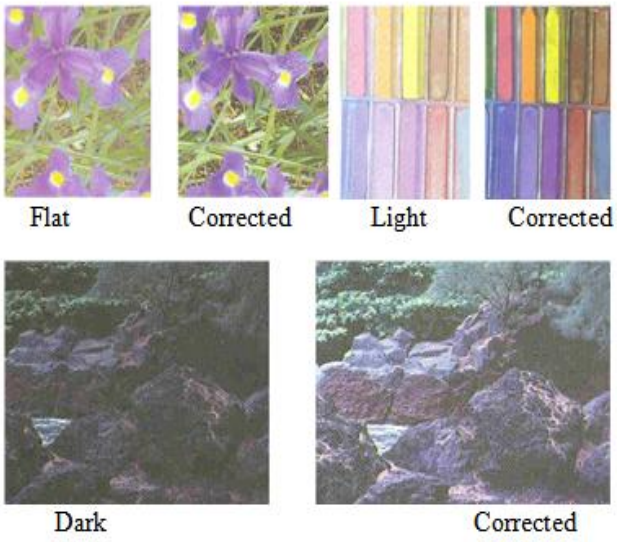


Fig 11: *Tonal correcting for Flat, light (high key), Dark (high key) color image.*

The *L*a*b* color space is *colorimetric* (i.e., colors perceived as matching are encoded identically), *perceptually uniform* (i.e., color differences among various hues are perceived uniformly-- see the class paper by Mac Adams [1942]), and *device independent.* While not a directly displayable format (conversion to another color space is required), its gamut encompasses the entire visible spectrum and can represent accurately the color of any display, print, or input device.

Like the his system, the *L*a*b* system is an excellent decoupled of intensity (represented by lightness *L**) and color (represented by a* for red minus green and b* for green minus blue), making it useful in both image manipulation (tone and contrast editing) and image compression applications. The principal benefit of calibrated imaging systems is that they allow tonal and color imbalances to be corrected interactively and independently—that is, in two sequential operations. Before color irregularities, like over- and under-saturated colors, are resolved, problems involving the image's tonal range are corrected. The *tonal range* of an image, also called its *key type*, refers to its general distribution of color intensities. Most of the information in *high-key* images is concentrated at high (or light) intensities; the colors of *low-key* images are located predominantly at low intensities; *middle-key* images lie in between. As in the monochrome case, it is often desirable to distribute the intensities o a color image equally between the highlights and the shadows. The following examples demonstrate a variety of color transformations for the correction of tonal and color imbalances.

Eg:- Transformations for modifying image tones normally are selected interactively. The idea is to adjust experimentally the image's brightness and contrast to provide maximum detail over a suitable range of intensities. The colors themselves are not changed. In the RGB and CMY(K) spaces, this means mapping all three (or four) color components with the same transformation function; in the his color space, only the intensity component is modified.

Fig 11 shows typical transformations used for correcting three common tonal imbalances—flat, light, and dark images. The S-shaped curve in the first row of the figure is ideal for boosting contrast. Its midpoint is anchored so that highlight and shadow areas can be lightened and darkened, respectively. (The inverse of this curve can be used to correct excessive contrast). The transformations in the second and third rows of the figure correct light and dark images and are reminiscent of the power-law transformations. Although the color components are discrete, as are the actual transformation functions, the transformation functions themselves are displayed and manipulated as continuous quantities—typically constructed from piecewise linear or higher order (for smoother mappings) polynomials. Note: that the keys of the images in **Fig 11** are directly observable; they could also be determined using the histograms of the images color components.

VI. Conclusion

The study in this is an introduction to color image processing and covers topics selected to give the reader a solid background in the techniques used in this branch of image processing. Our treatment of color fundamentals and color models was prepared as foundation material for a field that is in its own right wide in technical scope and areas of application. In particular, we focused on color models that we felt are not only useful in digital image processing but would also provide the tools necessary for further study in this area of color image processing. The discussion of pseudo color and full-color processing on an individual image basis provides a tie to techniques.

References

[1] Goutsias, J, Vincent, L., and Bloomberg, D. S. (eds.) [2000]. *Mathematical Morphology and Its Applications to Image and Signal Processing*,Kluwer Academic Publishers, Boston, MA.

[2] . Mallot, A.H. [2000]. *Computational Vision*, The MIT Press, Cambridge, MA.

[3] . Marchand-Maillet, S. and Sharaiha, Y. M. [2000]. *Binary Digital Image Processing: a discrete Approach*, Academic Press, NY.

[4] . Mitra, S. K. and Sicuranza, G. L. (eds.) {2000]. *Nonlinear Image Processing*, Academic Press, NY.

[5] . Edelman, S. [1999]. *Representation and Recognition in Vision*, The MIT Press, Cambridge, MA.

[6] . Lillesand, T. M. and Kiefer, R. W. [1999]. *Remote Sensing and Image Interpretation*, John wiley & Sons, NY.

[7] . Mather, P. M. [1999]. *Computer Processing of Remotely Sensed Images: An Introduction*, John Wiley & Sons, NY.

[8] . Petrou, M. and Bosdogianni, P. [1999]. *Image Processing: The Fundamentals*, John Wiley & Sons, UK.

[9] . Russ, J. C. [1999]. *The Image Processing Handbook*, 3rd ed., CRC Press, Boca Raton, FL.

[10] . Smirnov, A. [1999]. *Processing of Multidimensional Signals*, Springer- Verlag, NY.

[11] . Sonka, M., Hlavac, V., and Boyle, R. [1999]. *Image Processing, Analysis, and Computer Vision*, PWS Publishing, NY.

[12] . Umbaugh, S. E. [1998]. *Computer Vision and Image Processing: A Practical Approach Using CVIPtools*, Prentice Hall, Upper Saddle River, NJ.

[13] . Haskell, B.G. Nd Netravali, A.N. [1997]. *Digital Pictures: Representation, Compression, and Standards*, Perseus Publishing, NY.

[14] .Jahne, B. [1997]. *Digital Image Processing: Concepts, Algorithms, and Scientific Applications*, Springer-Verlag, NY.

[15] .Castleman, K.R. [1996]. *Digital Image Processing*, 2nd ed., Prentice Hall, Upper Sanddle River, NJ

AUTHORS PROFILE

Dr Mir Mohammad Azad was born in Village – Korer Betka; Post Office – Mirrer Betka; Police Station - Tangail; District - Tangail, Bangladesh on 10th October, 1982.He received PhD in Computer Science, 2008 from Golden State University, Master of Computer Application, 2006 from Bharath Institute of Higher Education and Research Deemed University (Bharath University) and Bachelor of Computer Application,2004,Bangalore University, India. He was working as a **lecturer** and **head of computer science** in various colleges in Bangalore and also worked as an **Assistant professor** and **Vice Principal** in different colleges in Bangalore during the year (2005-2009).He worked as an **Assistant Professor** and **Head of CSE & CSIT** at **Shanto Mariam university of Creative Technology (2010-2014).** He is having **18** publications in international journal in various countries like UK, USA, FRANCE, KOREA, PAKISTAN and INDIA. At present he is working as an **Associate Professor**, Department of **Computer Science and Engineering & Computer Science and Information Technology** in **Shanto Mariam university of Creative Technology**, Uttara, Dhaka, Bangladesh**.** His areas of interest include Computer Architecture, E-commerce, Digital Image processing, Computer Network, Wireless communication and MIS.

Md. Mahedi Hasan.was born in Vill: Anantupur, Post:Bonarpara, P.S: Saghata,Dist: Gaibandha, Bangladesh on 13th April 1983.He received **Master of Business Administration (M.B.A.) Major in MIS, 2009** Degree from *Sikkim Mannipal University Of Health Medical & Technological Science and* Bachelor of Computer Application (BCA) 2006 Degree from Bangalore University, India, He was working as a Lecturer, Department of Computer Science From January 2008 to March 2009 at Dum Dum Motijheel College (Honors Programs), Kolkata,India(Affiliated to the University of Calcutta (C.U.) and accredited by the National Assessment and Accreditation Council (N.A.A.C)) also worked as a **Lecturer** From 23rd December 2009 to August 2012 and As a **Senior Lecturer** from September 2012 to August 2013 **Prime University**, Dhaka. Presently Working as an **Assistant Professor**, **Prime University**, Dhaka, Bangladesh. His areas of interest include E-commerce, Digital Image, and MIS.

Mohammed Naseer K was born in Village Gangondanahalli , Nayandahalli **(Post),** Police Station – Chandra Layout; District - Bangalore, Karnataka, India on 12th January,1981.He is pursuing M.Tech in Computer Science Engineering at Jawaharlal Nehru Technological University (JNTU)**.** He received Master of Computer Application, 2007 from Bangalore University, also received M.Tech in Information Technology, 2007 from Karnataka State Open University and Bachelor of Computer Application, 2004 from Bangalore University, India. He worked as **Instructor (CSE)** from November 2009 to February 2010 at Maulana Azad National Urdu University. Worked as a Software Developer for the period of 2 years in CoreBits Technologies and also worked as a Tradesman (MR/AC) for the period of 2 years in ISRO. He worked as an **Assistant Professor / Guest faculty member of CSE** at **Maulana Azad National Urdu University (2010-2014).** At present he is working as a **Assistant professor of CSE** at **Maulana Azad National Urdu University,** India**.** His areas of interest include Programming, Digital Image processing, and Computer Network.