International Journal of Innovative Computing, Information and Control
Volume **8**, Number **1(A)**, January **2012**

ICIC International ©2012 ISSN 1349-4198
pp. **167–186**

A GLOBAL COLOR TRANSFER SCHEME BETWEEN IMAGES BASED ON MULTIPLE REGRESSION ANALYSIS

Chen-Chung Liu*

Department of Electronic Engineering
National Chin-Yi University of Technology
No. 35, Lane 215, Section 1, Taiping, Taichung 411, Taiwan
*Corresponding author: ccl@ncut.edu.tw

Received August 2010; revised December 2010

Abstract. *The image color transfer process enhances images' color features for wide applications in different fields. This paper presents a multiple regression analysis based algorithm for global color transfer between images. Both the target and source images are first transformed into the CIELAB color space. Multiple regression analysis (MRA) is then conducted on the L^*, a^* and b^* planes to find the corresponding best fitting functions for each plane, respectively. The new value of each pixel on each plane is then evaluated using the best fitting function. These new values are then combined into the LAB color image result. The result image is then transformed into the RGB color space. The experimental results show that the presented approach has three major advantages: (i) the proposed algorithm is manual free, simple, effective and accurate in transferring color between images without any change in the image details; (ii) the proposed algorithm saves time and the time consumption is independent of the number of bins selected and the degree of regression; (iii) there are no restrictions in the image dynamic color ranges in the proposed algorithm.*
Keywords: Color transfer, CIELAB, Multiple regression analysis

1. **Introduction.** Digital color images are the most important media for efficient information presentation and communication. They are widely applied in the multimedia, biomedical, Internet and video and push people to pay more attention to color image processing to meet human requirements [1]. Color is a significant source of information for image analysis [2], classification [3], segmentation, recognition and retrieval [4]. It provides artists with the ability to show creativity and style [5] and also offers doctors a means to describe tissue pathology and aesthetics in dermatology and dentistry [6]. Because some of the original colors in an image might not be suitable for analysis, the colors must be adjusted. The appearance of an image can be adjusted manually to obtain specific effects but that requires advanced image manipulation techniques which consume a great deal of time. Color transferring is one of the most popular techniques in image color processing for changing an image's color and preserving the image's original details and natural look at the same time. This means replacing the target image's colors using the source image's color without any change in the text and certain visual qualities [7].

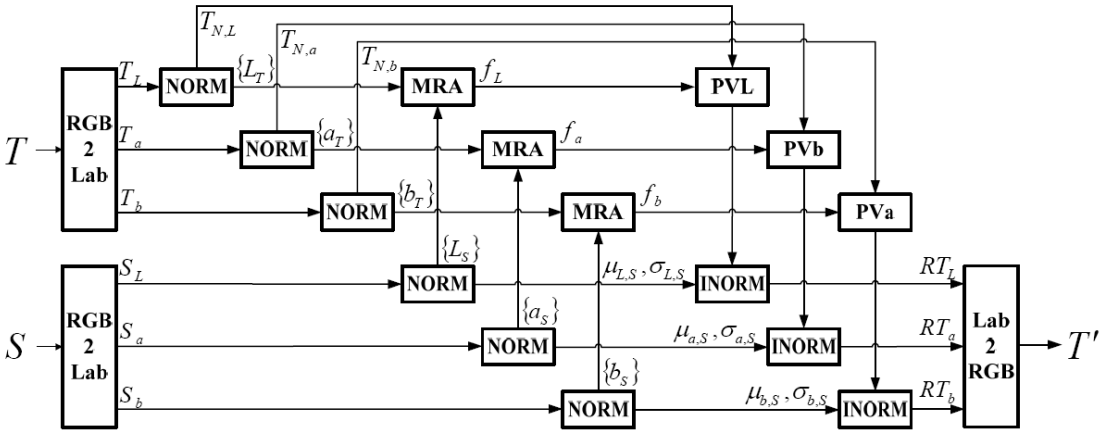
Color transfer between images is a challenging task because both the source and target images may consist of various colors, causing multiple shades in the transfer results [8]. Several schemes have been proposed for global color transfer between images. Reinhard et al. [9] proposed a statistical based scheme. Their idea is based on transferring the original RGB color images into the $l\alpha\beta$ color space proposed by Ruderman et al. [10], in which the correlation between channels is minimized and the color perception can be evaluated better. They assumed that the pixel values of each channel in the $l\alpha\beta$

color space was a Gaussian distribution and the color transfer process shifted and scaled each channel's pixel values from the target image to match the corresponding source image mean and variance because of the uncorrelated $l\alpha\beta$ color space axes. The color transferred $l\alpha\beta$ image is finally converted back into the RGB color space. Reinhard et al.'s scheme allows operating the transfer independently in each channel to convert a potentially complex three-dimensional problem into three much simpler one-dimensional problems. Although this technique is simple and efficient for a large range of images, it has three disadvantages: (i) it is time consuming because it must perform color space conversion three times to transfer color from a source image into a target image; (ii) the color transfer quality depends on the composition of the target and source images [5]; (iii) overflow will appear when the image hue is oversaturated and may not be good for images with a big gap between different chromaticity's [7]. S. Xu et al. [7] transformed target and source images into the $l\alpha\beta$ color spaced first. They used the Gaussian Mixture Models-Exception Maximum (GMM-EM) method to cluster the source color image into specified areas and used the K-means algorithm to segment the target image. They then imposed mean chromaticity values from the source image onto corresponding areas in the target image. C. C. Liu and G. N. Hu [11] considered the histogram in each plane of a color space as a probability density function. They first transformed the target and source images into LAB color space and then computed the probability density characteristics of each plane in each image to find the correlation function for every two corresponding planes between the target and source images. The transferred pixel values for each plane are determined by conducting the corresponding correlativity function on each target image plane. The color transferred target image is finally constructed by replacing each plane with its corresponding transferred plane. G. R. Greenfield and D. H. House [12] proposed an approach to recolor a target image according to the color scheme from the source image. The target and source images are first decomposed into clusters of pixels with similar color. The color palette for each image is constructed by selecting the most typical colors from the above decomposition. The color transfer from source image to the target image is evaluated by matching the decomposed areas between the target and source decomposition using a Euclidean metric. The resulting quality of the color transferred image is highly dependent on the previous segmentations. Zhou et al. [13] synthesized a natural scene using multiple labeled level eigen-spaces to generate a depth map, using multiple color level eigen-spaces to generate natural textures to fill in the label maps. Their approach can model natural textures and common depth layouts without learning large numbers of parameters to synthesize perceptually natural scene images. However, they often needed more than fifty sample images (depending on the complexity of the sample image) to train their system to extract sample textures and eigen-spaces. The goal of this paper is to find a simple and efficient global color transferring approach and a measurement metrics to evaluate color transferring scheme performance.

Multiple regression analysis is a statistical method used to model the relationships between several independent variables and a dependent variable by fitting an equation to the observed data. Numerous experiments have shown that multiple regression models can be used to make accurate predictions. Multiple regression procedures are very widely used in the social and natural sciences today [14]. It is also a suitable technique for us in global color transferring from a source image to a target image. The RGB color space is the most popular space used in sensor and display devices. It is reasonable to take use of multiple regression analysis on each component in a color image in the RGB domain. The RGB pixels of an image are often correlative. On the other hand, the uncorrelated CIELAB defines colors that are more closely related to human color perception and is often used in colored products quality control. Multiple regression analysis is conducted

in this work on the LAB and RGB domains of two images for global color transferring. The main stages in the proposed algorithm are: (i) best fitting functions are determined from the target and source histograms using multiple regression analysis; (ii) the color transfer is turned into a pixel value evaluation by conducting best fitting functions on the target image. The experimental results show that the proposed algorithm is effective and validated. The remainder of this paper is organized as follows: Section 2 presents the proposed algorithm; Section 3 describes the empirical results; Section 4 concludes this paper.

2. **The Proposed Algorithm.** This paper presents a way to transfer the colors from a source RGB color image to a target RGB color image. Figure 1 is the flow chart of the recolor for a target RGB color image with multiple regression analysis. In order to obtain a more accurate color transfer result, both the target image and source image are first transformed into the CIELAB color space from the original RGB color space. Multiple regression analysis is then conducted on each plane (L^*, a^* and b^* planes) to find the corresponding best fitting functions for each plane. These best fitting functions are used to determine the new values for each pixel in the target image on each plane. These new values are then combined into the LAB color image result. The final result RGB color image is obtained by transforming the LAB color image result into the RGB color space. Detailed descriptions of the proposed scheme are illustrated as follows.



T: Target (RGB color) Image
S: Source (RGB color) Image
INORM: Inverse normalization operation
T_L: L-plane of target LAB color image
T_b: b^*-plane of target LAB color image
RGB2Lab: color space transformation from RGB to Lab
Lab2RGB: color space transformation from LAB to RGB

T': Transferred target RGB color image
NORM: Normalization operation
MRA: Multiple regression analysis
T_a: a^*-plane of target LAB color image

Figure 1: The flow chart of the proposed color transfer algorithm

2.1. **Color space transform.** In this step, the original target image and source image are transformed from the original RGB color space into the CIELAB color space. RGB is the most popular and natural color space conventionally used in computer monitors and other raster devices to store, process, display and analyze color images. However, the RGB model cannot accurately and suitably describe human perception of color [15]. Due to the high correlation of R, G and B, a respective modification of the red, green and

blue channels may result an out-estimated color. Therefore, measures such as the color-difference and intensity-difference defined in the RGB color space are not appropriate to quantify the perceptual differences between images. CIELAB model is a standard measure of color defined by CIE in 1976. It is more suitable and effective for description of human color perception than in RGB space. It is irrelative with equipment and is also effective in separating color and brightness of the image. So, the CIELAB model is great appropriate for color processing of color images.

2.1.1. *The RGB color space.* The color of an object can be explained scientifically as the color of light reflected by the particular object. Although the range of colors is infinite, a full range of perceivable colors by human eyes is able to be reproduced by combining different proportions of the three primarily colors: red, green and blue [16].

In televisions, computer monitors and colored image projection systems, by using only the three colors are enough to adequately represent any of the unlimited visible colors [17]. The image in RGB color space is the most suitable for the color representation of a color image. The RGB color model is the most popular and natural color model, because it can compose any color adequately. R, G and B component of a color in RGB color space are given by:

$$R, G, B = k \int_{400}^{700} I(\lambda)\Phi(\lambda)S_{R,G,B}(\lambda)d\lambda, \tag{1}$$

where k is a constant that defines the total overall brightness response of the human eyes, $I[\lambda]$ is the illumination spectral intensity of a color, $\Phi[\lambda]$ is the object spectral reflectivity, $S_{R,G,B}[\lambda]$ is the spectral sensitivity of the R or G or B channel of the detector and λ is the wavelength. The normalized red, green and blue coordinates are defined as follows:

$$r = R/(R + G + B) \tag{2}$$

$$g = G/(R + G + B) \tag{3}$$

$$b = B/(R + G + B), \tag{4}$$

where R, G and B are the intensities of red, green and blue light at a given pixel [18].

2.1.2. *The CIELAB color system.* CIE 1976 $L^*a^*b^*$ (CIELAB) is an approximately uniform color space developed by the International Commission on Illumination Commission International de l'Eclairage (CIE) in 1976 [19]. The L^*-component represents the lightness of the color, a^*-component represents the position between green and magenta, and b^*-component represents the position between blue and yellow. It is more perceptually uniform than other color space to be the most complete color model for describing all the colors visible to the human eyes [15]. CIE LAB color space is used in this paper because it has accurate color representation.

The transformation from the RGB color space to CIELAB color space consists of three steps: (i) converting nonlinear RGB (or called gamma-corrected RGB) values into standard RGB values; (ii) converting standard RGB values to CIEXYZ values; (iii) converting CIEXYZ values to CIELAB values [20]. The original color space of the images that we frequently obtain is nonlinear RGB space: the nonlinear RGB is always represented with three 8-bit unsigned integers ranging from 0 to 255. In this paper, the inverse gamma correction (ITU-R BT.709) is conducted to convert the three integer values from 0 to 255 into floating-point numbers from 0 to 1 with the following equations [20,21].

$$R_S = \begin{cases} R_N/1147.5, & 0 \le R_N < 21 \\ (((R_N/255) + 0.099)/1.099)^{1/0.45}, & 21 \le R_N \le 255 \end{cases} \tag{5}$$

$$G_S = \begin{cases} G_N/1147.5, & 0 \le G_N < 21 \\ (((G_N/255) + 0.099)/1.099)^{1/0.45}, & 21 \le G_N \le 255 \end{cases} \tag{6}$$

$$B_S = \begin{cases} B_N/1147.5, & 0 \le B_N < 21 \\ (((B_N/255) + 0.099)/1.099)^{1/0.45}, & 21 \le B_N \le 255\,, \end{cases} \tag{7}$$

where R_N, G_N and B_N are red, green and blue component value in nonlinear RGB space, respectively. Moreover, R_S, G_S and B_S are red, green and blue component value in standard RGB space, respectively. The corresponding converting from standard RGB space to nonlinear RGB space is given by the following equations.

$$R_N = \begin{cases} fix(1147.5 \times R_N), & 0 \le R_S < 0.01830 \\ fix(255 \times (1.099 \times R_S^{0.45} - 0.99)), & \text{otherwise} \end{cases} \tag{8}$$

$$G_N = \begin{cases} fix(1147.5 \times G_N), & 0 \le G_S < 0.01830 \\ fix(255 \times (1.099 \times G_S^{0.45} - 0.99)), & \text{otherwise} \end{cases} \tag{9}$$

$$B_N = \begin{cases} fix(1147.5 \times B_N), & 0 \le B_S < 0.01830 \\ fix(255 \times (1.099 \times B_S^{0.45} - 0.99)), & \text{otherwise}\,. \end{cases} \tag{10}$$

The converting from the standard RGB color space to CIEXYZ (ITU-RBT.709) space is given by the following equations [20-22].

$$X = 0.412391 \times R_S + 0.357584 \times G_S + 0.180481 \times B_S \tag{11}$$

$$Y = 0.212639 \times R_S + 0.715169 \times G_S + 0.072192 \times B_S \tag{12}$$

$$Z = 0.019331 \times R_S + 0.119195 \times G_S + 0.950532 \times B_S\,. \tag{13}$$

Finally, the converting from the CIEXYZ space to CIELAB space is given by the following equations [20,21].

$$L^* = 116 \times f(Y/Y_n) - 16 \tag{14}$$

$$a^* = 500 \times (f(X/X_n) - f(Y/Y_n)) \tag{15}$$

$$b^* = 200 \times (f(Y/Y_n) - f(Z/Z_n)) \tag{16}$$

$$f(u) = \begin{cases} u^{1/3}, & \text{if } 0.008856 < u \\ 7.787 \times u + 16/116, & \text{otherwise} \end{cases}\,. \tag{17}$$

where X_n, Y_n and Z_n are the CIEXYZ tristimulus values of the reference white point, they are 0.950456, 1.0 and 1.089058, respectively, for the CIEXYZ (D65). The corresponding inverse converting from the CIELAB space to CIEXYZ space is given by the following equations.

$$X = \begin{cases} X_n \times v_x^3, & v_x > \delta \\ 3\delta^2 \times (v_x - 16/116) \times X_n, & \text{otherwise} \end{cases} \tag{18}$$

$$Y = \begin{cases} Y_n \times v_y^3, & v_y > \delta \\ 3\delta^2 \times (v_y - 16/116) \times Y_n, & \text{otherwise} \end{cases} \tag{19}$$

$$Z = \begin{cases} Z_n \times v_z^3, & v_z > \delta \\ 3\delta^2 \times (v_z - 16/116) \times Z_n, & \text{otherwise} \end{cases}\,, \tag{20}$$

where $\delta = 6/29$, $v_y = (L^* + 16)/116$, $v_x = v_y + a^*/500$ and $v_z = v_y - b^*/200$.

2.2. **Normalization.** In the CIELAB color space, colors are represented by varying values of L^*-component, a^*-component and b^*-component. The values of each of the L^*-component, a^*-component and b^*-component are represented with 8-bit unsigned integers (uint8) on a scale from 0 to 255. The higher degree polynomial regression should not work since it becomes impossible to avoid the overflow when the data are represented with uint8. In multiple regression stage, the values of L^*-component, a^*-component and b^*-component are normalized to compress and limit the dynamic range to avoid the overflow. The mean and standard deviation of L^*-plane, a^*-plane and b^*-plane are evaluated, respectively. And then the values of L^*-component, a^*-component and b^*-component of a $M \times N$ CIELAB color image are normalized by the following equations.

$$\mu_X = \sum_{i=1}^{M}\sum_{j=1}^{N} X(i,j)/(M \times N), \quad X \in \{L^*, a^*, b^*\} \tag{21}$$

$$\sigma_X = \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} (X(i,j) - \mu_X)^2/(M \times N)}, \quad X \in \{L^*, a^*, b^*\} \tag{22}$$

$$X_N(i,j) = (X(i,j) - \mu_X)/\sigma_X, X \in \{L^*, a^*, b^*\}, \quad 1 \le j \le M, \quad 1 \le j \le N\,, \tag{23}$$

where $X(i,j)$, $X \in \{L^*, a^*, b^*\}$ is the pixel value of pixel $p(i,j)$ in the X plane of original CIELAB color image, and $X_N(i,j)$, $X \in \{L^*, a^*, b^*\}$ is the corresponding pixel value of pixel $p(i,j)$ in the X plane of normalized CIELAB color image.

2.3. **Multiple regression analysis (MRA).** When performing experiments, data are frequently tabulated in the form of ordered pairs $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$ with each x_i distinct. Given the data, it is then usually desirable to be able to predict y from x by finding a mathematical model, that is a function $y = H(x)$ that fits the data as closely as possible. One way to determine how well the function $y = H(x)$ fits these order pairs $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$ is to measure the sum of squares of the errors (SSE) between the predicted values of y and the observed values y_i for all of the n data points.

Multiple regression is one of the widely used statistical techniques [14]. This technique is used to find a polynomial function of degree k, $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_k x^k$ as the predicting function, that has the minimum of the sum of squares of the errors (SSE) between the predicted values of y and the observed values y_i for all of the n data points $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$. The values of β_0, β_1, β_2, $\cdots$, β_k that minimize

$$SSE(\beta_0, \beta_1, \cdots, \beta_k) = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_k x_i^k)]^2 \tag{24}$$

are obtained by setting the $k+1$ first partial derivatives $\frac{\partial}{\partial\beta_0}SSE(\beta_0,\beta_1,\cdots,\beta_k)$, $\frac{\partial}{\partial\beta_1}SSE(\beta_0,\beta_1,\cdots,\beta_k)$, $\cdots$, $\frac{\partial}{\partial\beta_k}SSE(\beta_0,\beta_1,\cdots,\beta_k)$ equal to zero, and solving the resulting simultaneous linear system of the so-called normal equations:

$$n\beta_0+\beta_1\sum_{i=1}^{n}x_i+\beta_2\sum_{i=1}^{n}x_i^2+\cdots+\beta_k\sum_{i=1}^{n}x_i^k=\sum_{i=1}^{n}y_i \tag{25}$$

$$\beta_0\sum_{i=1}^{n}x_i+\beta_1\sum_{i=1}^{n}x_i^2+\beta_2\sum_{i=1}^{n}x_i^3+\cdots+\beta_k\sum_{i=1}^{n}x_i^{k+1}=\sum_{i=1}^{n}x_iy_i \tag{26}$$

$$\beta_0\sum_{i=1}^{n}x_i^k+\beta_1\sum_{i=1}^{n}x_i^{k+1}+\beta_2\sum_{i=1}^{n}x_i^{k+2}+\cdots+\beta_k\sum_{i=1}^{n}x_i^{2k}=\sum_{i=1}^{n}x_i^ky_i, \tag{27}$$

and the matrix form solution of the normal equations system be

$$\begin{bmatrix}\beta_0\\ \beta_1\\ \beta_2\\ \vdots\\ \beta_k\end{bmatrix}=\mathrm{B}=\left[X^TX\right]^{-1}\left[X^TY\right], \tag{28}$$

where

$$X=\begin{bmatrix}1 & x_1 & x_1^2 & \cdots & x_1^k\\ 1 & x_2 & x_2^2 & \cdots & x_2^k\\ 1 & x_3 & x_3^2 & \cdots & x_3^k\\ \vdots & \vdots & \vdots & \cdots & \vdots\\ 1 & x_n & x_n^2 & \cdots & x_n^k\end{bmatrix},\quad Y=\begin{bmatrix}y_1\\ y_2\\ \vdots\\ y_n\end{bmatrix}. \tag{29}$$

2.4. **Best fitting functions determination.** At the beginning of the process, the L^*, a^* and b^* planes of the $M_T\times N_T$ normalized target LAB color image and the $M_S\times N_S$ normalized source LAB color image are uniformly split into l non-overlapping blocks, $4\le l\le\min(M_T,N_T,M_S,N_S)$, respectively. The mean of each block is then served as the input data for multiple regression analysis. Figure 2(a1) shows the target RGB color image with size 853×602 pixels, it is a traditional Chinese painting. Figure 2(a2) is the histogram of the target RGB color image, Figure 2(a3) is the histogram of the normalized target LAB color image, and Figure 2(a4) is the histogram of the 100 non-overlapping blocks of the normalized target LAB color image. Figure 2(b1) shows the source RGB color image with size 1024×768, it is an outdoor scene taken in autumn. Figure 2(b2) is the histogram of the source RGB color image, Figure 2(b3) is the histogram of the normalized source LAB color image, and Figure 2(b4) is the histogram of the 100 non-overlapping blocks of the normalized source LAB color image. The shapes of Figure 2(a3) and Figure 2(a4) are almost the same, the major difference between them is the height (number of each pixel value). The comparison result between Figure 2(b3) and Figure 2(b4) is like that between Figure 2(a3) and Figure 2(a4). Tables 1-3 are respectively the L^*, a^* and b^* components of the 100 non-overlapping blocks of the normalized target LAB color image. On the other hand, Tables 4-6 are respectively the L^*, a^* and b^* components of the 100 non-overlapping blocks of the normalized target LAB color image.

The multiple regression analysis is used to find the best fitting functions (polynomials) of the three elements L^*, a^* and b^* of these color-block images. These best fitting functions of MRA are used to find out the transferred L^*, a^* and b^* values of each pixel of the target image. These transferred values are conducted in a transform from LAB to RGB, and

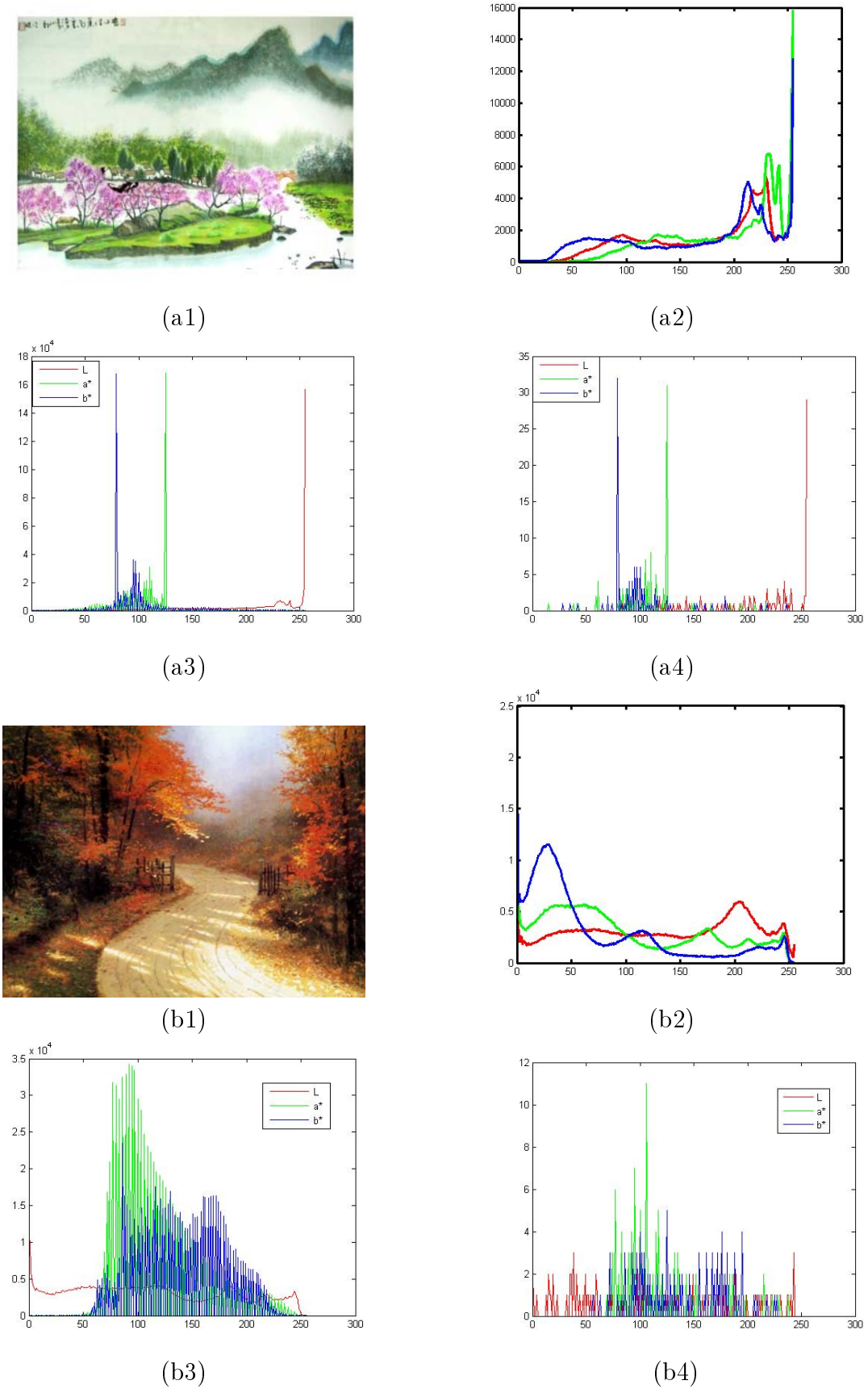


Figure 2: The target image and source image

Table 1: The 100 non-overlapping blocks' L^* values of the normalized target LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	0.843145	0.226178	0.305787	0.4252	0.285884	0.206276	0.106765	0.106765	–0.33108	0.843145
Row 2	0.843145	0.24608	0.345591	0.305787	0.305787	0.345591	–0.5102	–1.12717	–2.10237	0.843145
Row 3	0.843145	0.544612	0.166472	–0.66942	–0.58981	–1.50531	–1.24658	–1.3859	–0.31118	0.843145
Row 4	0.843145	0.465004	0.564515	0.445102	0.106765	0.4252	0.763536	0.803341	–0.21167	0.843145
Row 5	0.843145	–1.54511	0.544612	0.465004	0.504808	0.325689	0.564515	0.544612	0.4252	0.843145
Row 6	0.843145	–0.72913	–1.32619	–2.56012	–1.02766	–1.62472	–0.19177	0.086863	–1.88345	0.843145
Row 7	0.843145	–0.11216	–2.44071	–0.13206	–0.80873	–0.90824	–0.31118	–1.3859	–0.39079	0.843145
Row 8	0.843145	0.146569	–1.72423	–1.12717	–1.10727	–0.13206	–0.21167	0.823243	–0.01265	0.843145
Row 9	0.843145	0.803341	–2.08247	–1.88345	–1.78394	–1.82374	0.4252	0.843145	0.823243	0.843145
Row 10	0.843145	0.843145	0.843145	0.843145	0.843145	0.843145	0.843145	0.843145	0.843145	0.843145

Table 2: The 100 non-overlapping blocks' a^* values of the normalized target LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	0.559141	0.204141	0.02664	0.470391	–0.06211	0.02664	0.470391	0.381641	0.204141	0.559141
Row 2	0.559141	–0.15086	–0.15086	–0.06211	0.02664	0.02664	–0.86086	–0.77211	–0.94961	0.559141
Row 3	0.559141	–0.15086	–0.41711	–1.03836	–0.59461	–0.94961	–0.86086	–0.59461	–0.41711	0.559141
Row 4	0.559141	–0.23961	0.02664	–0.15086	–0.50586	0.02664	0.204141	0.470391	–0.15086	0.559141
Row 5	0.559141	–1.74836	–0.06211	–0.15086	0.02664	–0.41711	–0.06211	–0.15086	0.204141	0.559141
Row 6	0.559141	–1.65961	3.399145	–1.21586	–1.65961	–1.65961	–0.68336	–0.68336	–0.77211	0.559141
Row 7	0.559141	1.979143	1.535393	2.600394	2.955395	3.754146	1.357892	–0.50586	–3.25711	0.559141
Row 8	0.559141	1.446642	–0.77211	0.02664	–2.28086	–1.74836	–1.65961	0.559141	–2.45836	0.559141
Row 9	0.559141	0.204141	–0.68336	–0.41711	–0.86086	–0.50586	–0.32836	0.559141	0.559141	0.559141
Row 10	0.559141	0.559141	0.559141	0.559141	0.559141	0.559141	0.559141	0.559141	0.559141	0.559141

Table 3: The 100 non-overlapping blocks' b^* values of the normalized target LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	–0.57398	–0.23345	–0.23345	–0.30156	–0.02913	0.038972	–0.30156	–0.23345	0.038972	–0.57398
Row 2	–0.57398	–0.09724	–0.09724	–0.02913	–0.02913	–0.09724	0.107077	0.447604	–0.8464	–0.57398
Row 3	–0.57398	–0.09724	–0.02913	0.243288	0.175182	0.038972	–0.16534	–0.16534	0.311393	–0.57398
Row 4	–0.57398	–0.09724	–0.16534	–0.09724	–0.02913	–0.36966	–0.30156	–0.43777	–0.16534	–0.57398
Row 5	–0.57398	2.354554	0.107077	0.175182	0.038972	0.515709	–0.02913	0.038972	–0.23345	–0.57398
Row 6	–0.57398	2.014027	–207229	0.788131	1.809711	1.53729	0.515709	0.447604	0.038972	–0.57398
Row 7	–0.57398	–0.50587	–0.71019	–0.98261	–1.86798	–1.66366	–0.57398	1.605395	4.057188	–0.57398
Row 8	–0.57398	–0.8464	0.175182	0.107077	2.354554	2.42266	3.376135	–0.57398	3.512345	–0.57398
Row 9	–0.57398	–0.50587	0.447604	0.379498	0.788131	0.856236	–0.57398	–0.57398	–0.50587	–0.57398
Row 10	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398	–0.57398

Table 4: The 100 non-overlapping blocks' L^* values of the normalized source LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	–0.99812	–0.72969	–0.74381	–0.81445	0.753761	1.686214	1.855751	1.559061	0.315791	–0.09392
Row 2	–1.12527	–1.30894	–0.53189	0.160382	0.146254	1.34714	1.502549	1.431909	1.092835	–1.02638
Row 3	–1.57737	–0.85684	–0.99812	0.329919	0.739633	0.527712	1.064579	0.852658	–0.36236	–0.98399
Row 4	–1.33719	–1.08289	–1.39371	–0.8851	0.132126	0.527712	0.767889	0.4712	–0.433	–1.36545
Row 5	–1.57737	–1.25243	0.245151	–0.03741	–0.22108	–0.13631	–0.20695	–0.30584	0.188638	–0.91335
Row 6	–1.0405	–1.36545	–1.06876	–0.77207	–0.39061	0.993938	0.83853	–1.35132	0.033229	–1.52086
Row 7	–0.95574	–1.30894	–1.08289	0.117998	0.075614	0.965682	0.923298	–0.17869	–0.84271	–1.56324
Row 8	–0.10805	–0.74381	–0.10805	0.654865	0.923298	1.064579	1.488421	–0.22108	–1.0405	–1.0405
Row 9	–0.7862	1.361268	–0.17869	1.785111	1.234116	1.573189	1.855751	0.824402	0.287535	–1.06876
Row 10	–0.09392	–0.56015	–0.8851	0.61248	1.092835	1.403653	1.855751	1.770983	1.813367	–0.02328

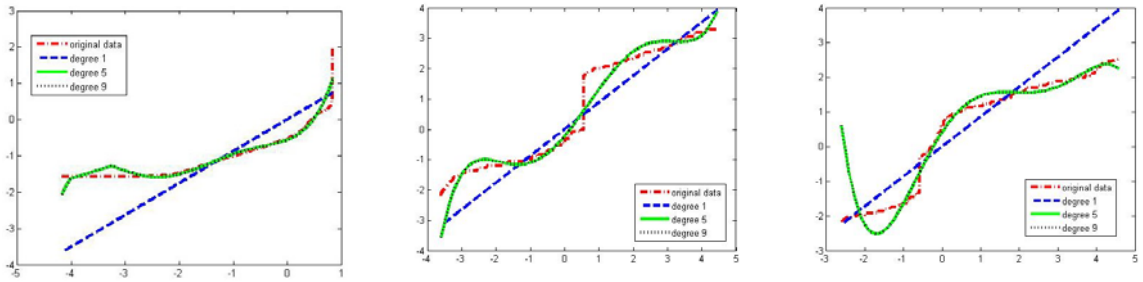
Table 5: The 100 non-overlapping blocks' a^* values of the normalized target LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	–0.97082	–0.07755	0.443522	–0.30087	–0.67306	–1.19413	–1.41745	–1.11969	0.890154	1.78342
Row 2	–0.37531	–0.44974	0.369083	2.527808	3.123318	–0.59862	–0.89638	–0.82194	0.815716	0.071328
Row 3	–1.04525	0.145766	–0.15199	2.602246	1.857859	1.78342	–0.59862	–0.59862	0.443522	–0.00311
Row 4	–1.19413	–0.89638	–0.07755	–0.07755	1.708981	–0.30087	–0.59862	–0.30087	1.113471	0.220205
Row 5	–1.04525	–0.59862	2.081175	2.37893	1.262348	–0.00311	0.666838	0.443522	2.900002	0.964593
Row 6	–0.30087	–0.59862	–0.67306	–0.52418	–0.00311	–0.44974	–0.52418	–0.00311	2.751124	–0.7475
Row 7	–0.30087	–0.67306	–0.30087	2.527808	0.294644	–0.44974	–0.37531	0.369083	–0.15199	–0.97082
Row 8	0.369083	–0.52418	1.708981	0.071328	–0.7475	–0.67306	–1.04525	0.666838	–0.44974	–0.44974
Row 9	–0.00311	–1.04525	–0.30087	–1.04525	–0.89638	–1.04525	–1.19413	–0.07755	0.890154	–0.59862
Row 10	–0.22643	–0.30087	–0.30087	–0.30087	–0.7475	–0.89638	–1.11969	–1.26857	–1.11969	–0.30087

Table 6: The 100 non-overlapping blocks' b^* values of the normalized target LAB color image in Figure 2

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10
Row 1	–0.75448	–0.05185	–0.34461	–0.05185	–1.04724	–1.74987	–1.51566	–2.10118	1.704731	0.943548
Row 2	–0.81303	–0.87158	0.416575	1.411968	1.763283	–1.69132	–1.92553	–1.69132	0.709337	–0.34461
Row 3	–1.34	–0.28606	–0.98869	0.943548	2.231704	1.997494	–1.69132	–0.87158	–0.1104	–0.57882
Row 4	–1.10579	–1.16434	–1.16434	–0.75448	0.826443	–0.40316	–0.34461	–0.34461	0.884995	–0.93013
Row 5	–1.34	–0.87158	1.411968	1.060653	0.709337	–0.16895	0.182364	0.123812	1.411968	–0.28606
Row 6	–1.04724	–1.2229	–1.16434	–0.2275	0.006707	0.709337	0.182364	–0.93013	1.23631	–1.39855
Row 7	–0.34461	–1.10579	–1.04724	1.23631	0.416575	0.592232	0.53368	0.943548	–0.69592	–1.34000
Row 8	0.76789	–0.63737	0.826443	1.060653	0.592232	0.826443	1.353416	0.240917	–0.98869	–0.98869
Row 9	–0.52027	2.173151	–0.16895	–0.98869	1.060653	0.592232	–0.81303	1.411968	1.0021	–0.93013
Row 10	1.0021	0.006707	–0.46171	0.76789	0.943548	1.23631	–0.57882	–0.46171	0.416575	0.884995

then constructed into the transferred target RGB color image. Figure 3 shows the curves of the degree 1, degree 5 and degree 9 best fitting functions of component L^*, a^* and b^* between source image and target image. Tables 7-9 are respectively the tables of the best fitting functions' coefficients of component L^*, a^* and b^*.



L^*-compotents a^*-compotents b^*-compotents

Figure 3: The curves of degree1, 5 and 9 best fitting functions

Table 7: Coefficients of best fitting functions for L^* component

	β_9	β_8	β_7	β_6	β_5	β_4	β_3	β_2	β_1	β_0
Degree 1	0	0	0	0	0	0	0	0	0.87682	–5.70e–15
Degree 2	0	0	0	0	0	0	0	0.33983	1.2507	–0.33981
Degree 3	0	0	0	0	0	0	0.14093	0.6874	1.1911	–0.5323
Degree 4	0	0	0	0	0	0.075242	0.46947	0.8809	0.8878	–0.60033
Degree 5	0	0	0	0	0.039776	0.32632	0.82987	0.7227	0.57363	–0.56125
Degree 6	0	0	0	0.017063	0.17485	0.62811	0.89836	0.41333	0.47728	–0.51124
Degree 7	0	0	0.013664	0.14732	0.57393	0.96348	0.58115	0.005104	0.53043	–0.46366
Degree 8	0	0.013864	0.16889	0.75244	1.4323	0.86048	–0.44713	–0.27407	0.77922	–0.44087
Degree 9	0.011261	0.15988	0.86564	2.1483	2.0572	–0.56507	–1.7347	0.064714	1.0855	–0.47083

Table 8: Coefficients of best fitting functions for a^* component

	β_9	β_8	β_7	β_6	β_5	β_4	β_3	β_2	β_1	β_0
Degree 1	0	0	0	0	0	0	0	0	0.88168	–1.22e-14
Degree 2	0	0	0	0	0	0	0	0.077088	0.86619	–0.07708
Degree 3	0	0	0	0	0	0	–0.05749	0.14161	1.1625	–0.13004
Degree 4	0	0	0	0	0	–0.013283	–0.044292	0.25699	1.15	–0.17663
Degree 5	0	0	0	0	0.008457	–0.028272	–0.14399	0.35893	1.3424	–0.22992
Degree 6	0	0	0	0.002250	0.004328	–0.064943	–0.10487	0.49875	1.3152	–0.26551
Degree 7	0	0	–0.000949	0.004609	0.02206	–0.097519	–0.1931	0.58943	1.3999	–0.29663
Degree 8	0	–0.000532	0.000472	0.016991	–0.002987	–0.18747	–0.086218	0.7949	1.341	–0.33601
Degree 9	0.000141	–0.000980	–0.003142	0.026609	0.027287	–0.24603	–0.17513	0.88859	1.3905	–0.35861

3. **Experiment Results.** These experiments were conducted on a computer with a 2.8 GHz Intel Pentium processor and 2 GB RAM running Matlab version 7.6. For 853×602 target image and 1024×768 source image, the color transferring procedure lasted about 0.05 seconds while the scheme was performed in RGB color space. About 3.17 seconds were required for the scheme to operate in the LAB color space. The reason is that the original input images and the output image are RGB color images. Additional procedures

Table 9: Coefficients of best fitting functions for b^* component

	β_9	β_8	β_7	β_6	β_5	β_4	β_3	β_2	β_1	β_0
Degree 1	0	0	0	0	0	0	0	0	0.8602	2.741e–14
Degree 2	0	0	0	0	0	0	0	–0.22694	1.2909	0.22693
Degree 3	0	0	0	0	0	0	–0.00256	–0.21976	1.2954	0.2246
Degree 4	0	0	0	0	0	0.04562	–0.16885	–0.34925	1.7012	0.34661
Degree 5	0	0	0	0	–0.01304	0.10415	–0.13243	–0.60732	1.695	0.41196
Degree 6	0	0	0	–0.003504	0.006702	0.10639	–0.25123	–0.55923	1.8078	0.41988
Degree 7	0	0	0.0040672	–0.028953	0.004521	0.30803	–0.35421	–0.92933	1.8736	0.49496
Degree 8	0	–0.000687	0.0091706	–0.031305	–0.04293	0.36372	–0.2341	–1.0449	1.7999	0.50457
Degree 9	–0.000530	0.003767	0.0050341	–0.075644	0.034932	0.48979	–0.45215	–1.153	1.8873	0.52539

are required to transform the input images from the RGB color space into the LAB color space and transform the output image from the LAB color space into the RGB color space while the scheme is conducted in the LAB color space.

This section presents experimental results under various conditions to illustrate the utility and efficiency of the proposed scheme. The input target RGB color image is a traditional Chinese painting with size 853×602 pixels. The input source RGB color images with different sizes are outdoor scenes shot in autumn (1024×768), a mountain scene shot on a winter sunny day (700×525), flower (499×396), watercolor painting (550×322) and city scene shot on a cloudy day (1024×689).

In order to give a detailed description of the color transfer results, several measurement metrics are conducted to measure the color transfer performance of the proposed algorithm. They are the difference in mean values of the color transferred image from the source image in lightness/darkness (L^*), in red-shade/green-shade (a^*), in yellow-shade/blue-shade (b^*), in chromaticity (C^*), in hue (H^*) and in total color (E^*) [20,21]. These performance measures are based on two images: a source color image and the color transferred image. These performance measures are described as follows:

$$C^* = \sqrt{(a^*)^2 + (b^*)^2} \tag{30}$$

$$E^* = \sqrt{(L^*)^2 + (a^*)^2 + (b^*)^2} \tag{31}$$

$$H^* = (180 \times \arctan 2(b^*/a^*))/\pi \tag{32}$$

$$\bar{X} = \sum_{i=1}^{M} \sum_{j=1}^{N} X(i,j)/(M \times N), X \in \{L^*, a^*, b^*, C^*, H^*, E^*\} \tag{33}$$

$$\Delta\bar{X} = \bar{X}_t - \bar{X}_s \tag{34}$$

$$\Delta\bar{X}(\%) = 100 \times (|\Delta\bar{X}|/\bar{X}_s), \tag{35}$$

where $X(i,j)$, $X \in \{L^*, a^*, b^*, C^*, H^*, E^*\}$ is the pixel value of pixel $p(i,j)$ in the X plane of original CIELAB color image, $M \times N$ is the size of the test image, $\bar{X}$ is the mean of X, $\bar{X}_t$ is the mean of X component of color transferred image, $\bar{X}_s$ is the mean of X component of source image, $\Delta\bar{X}$ is the difference of X component mean of the transferred image from the source image, $\Delta\bar{X}(\%)$ is the difference ratio of X component mean of the transferred image from the source image and $\tan 2$ is a more novel version (four quadrant) of the arc-tangent function that returns the angle in the full range $(-\pi, \pi]$, and is defined

as the following equation [36]:

$$\arctan 2\left(\frac{y}{x}\right) = \begin{cases} \arctan(y/x), & x > 0 \\ \pi + \arctan(y/x), & x < 0, y \geq 0 \\ -\pi + \arctan(y/x), & x < 0, y < 0 \\ \pi/2, & x = 0, y > 0 \\ -\pi/2, & x = 0, y < 0 \\ \text{undefined}, & x = y = 0. \end{cases} \tag{36}$$

For testing images, these measurement metrics are evaluated by applying the proposed algorithm in the RGB domain and LAB domain, respectively. These measurement metrics are listed in tables for the performance analysis. Figures 4, 5 and Table 4 are used to demonstrate the color transfer results corresponding to the variation in the degree of best fitting polynomials (functions) and the color transfer result corresponding to different color spaces. Figure 4(T) is the original target image and Figure 4(S) is the source image. Figure 4(aj), $j = 1, 3, 5, 7, 9$, represent the color transferred image obtained with the j best fitting function in the LAB color space. Figure 4(bj), $j = 1, 3, 5, 7, 9$, indicate the color transferred image obtained with the j best fitting function in the RGB color space. Figure 5 shows the box-plots of L^*, a^* and b^* for the target image, source image, and color transferred images obtained by applying the best fitting functions in the RGB domain and in the LAB domain, respectively. The upper row shows the LAB component distributions of color transferred images obtained by applying the best fitting functions on the target LAB image. The lower row is the LAB component distributions of color transferred images obtained by applying the best fitting functions on the target RGB image. Column 1 indicates the L^* component, Column 2 indicates the a^* component and Column 3 indicates the b^* component. We can make the following conclusions based on the above box plots: (i) Processing in the LAB color space, the performance of any degree best fitting functions are almost the same on the L^*, a^* and b^* components; (ii) Processing in the RGB color space, the performances of the higher degree (larger than 2) best fitting functions are almost the same on the L^*, a^* and b^* components; (iii) The higher degree multiple regressions are superior to the lower multiple regressions for color transfer in the RGB color space. The measurement metrics for the target image, source image and color transferred images in Figure 4 are arranged into Table 10. Table 10 shows the same conclusions as Figure 4 and Figure 5.

Table 10: Statistical values of CIELAB components of target image, source image and color transferred images

CIELAB Components		L^*		a^*		b^*		C^*		H^*		E^*	
		MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
Target		212.64	50.25	121.70	11.27	136.43	14.68	183.60	7.47	41.84	5.13	283.45	34.10
source		111.65	70.78	142.04	13.43	150.89	17.08	207.45	19.43	43.34	2.68	243.66	39.01
Degree 1	LAB	92.56	57.48	140.17	11.78	144.90	10.91	202.07	8.31	44.04	3.85	228.18	26.54
	RGB	82.08	57.50	139.77	12.06	148.03	12.98	204.00	12.15	43.38	3.60	225.75	29.02
Degree 3	LAB	92.69	57.50	140.18	11.78	144.92	10.91	202.09	8.32	44.04	3.84	228.26	26.58
	RGB	76.98	50.67	142.20	12.19	150.12	12.94	207.04	14.44	43.46	2.83	225.25	28.83
Degree 5	LAB	92.44	57.45	140.13	11.77	144.92	10.89	202.05	8.31	44.03	3.84	228.11	26.53
	RGB	76.17	50.55	143.08	12.01	150.00	13.06	207.54	14.73	43.66	2.70	225.38	29.15
Degree 7	LAB	92.81	57.55	140.20	11.75	144.90	10.90	202.07	8.30	44.05	3.84	228.31	26.58
	RGB	75.26	48.98	143.44	12.36	150.27	13.17	207.98	15.08	43.68	2.71	225.16	29.18
Degree 9	LAB	92.61	57.52	140.18	11.77	144.89	10.90	202.07	8.31	44.05	3.84	228.21	26.56
	RGB	74.40	47.77	143.95	12.40	150.26	13.24	208.32	15.20	43.79	2.69	224.99	28.73

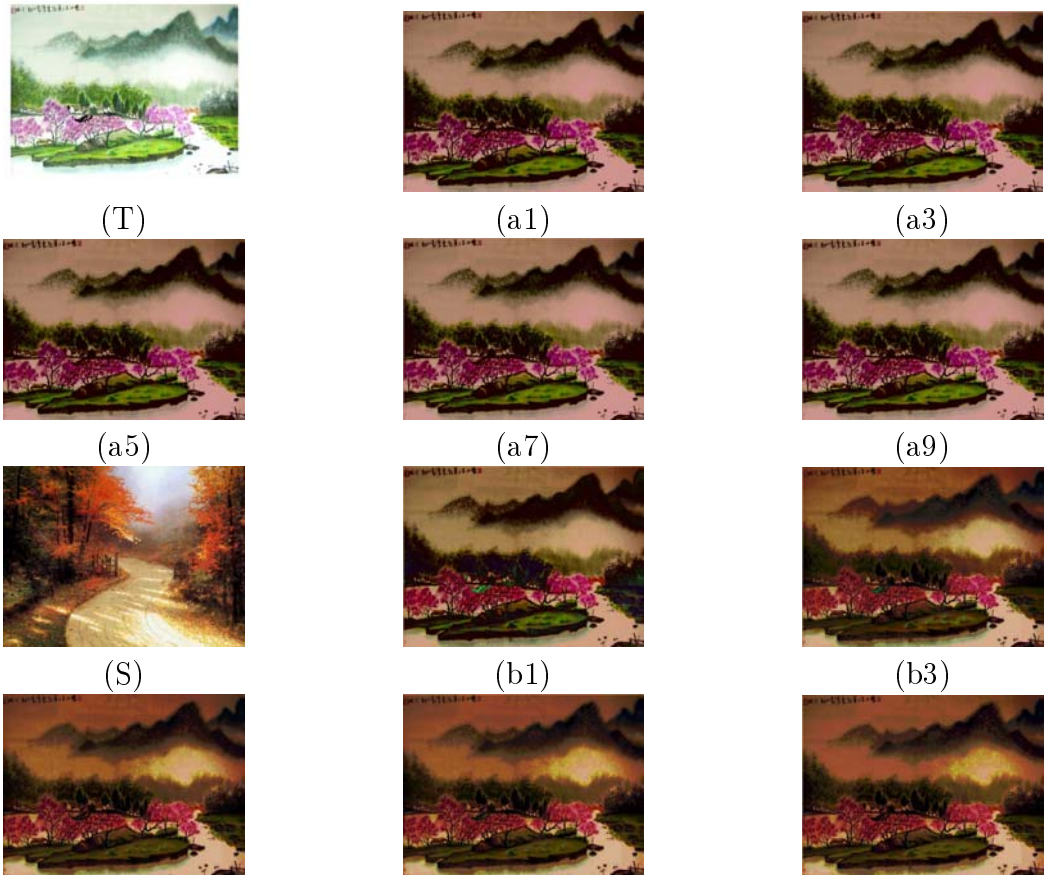


Figure 4: The color transferred images

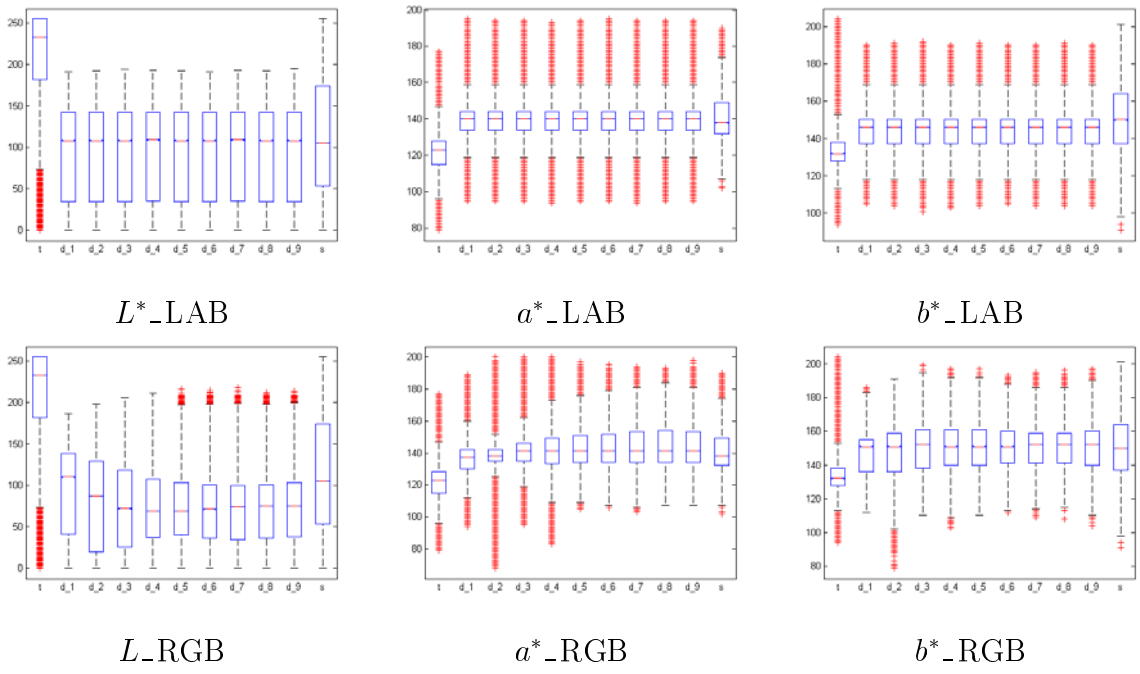
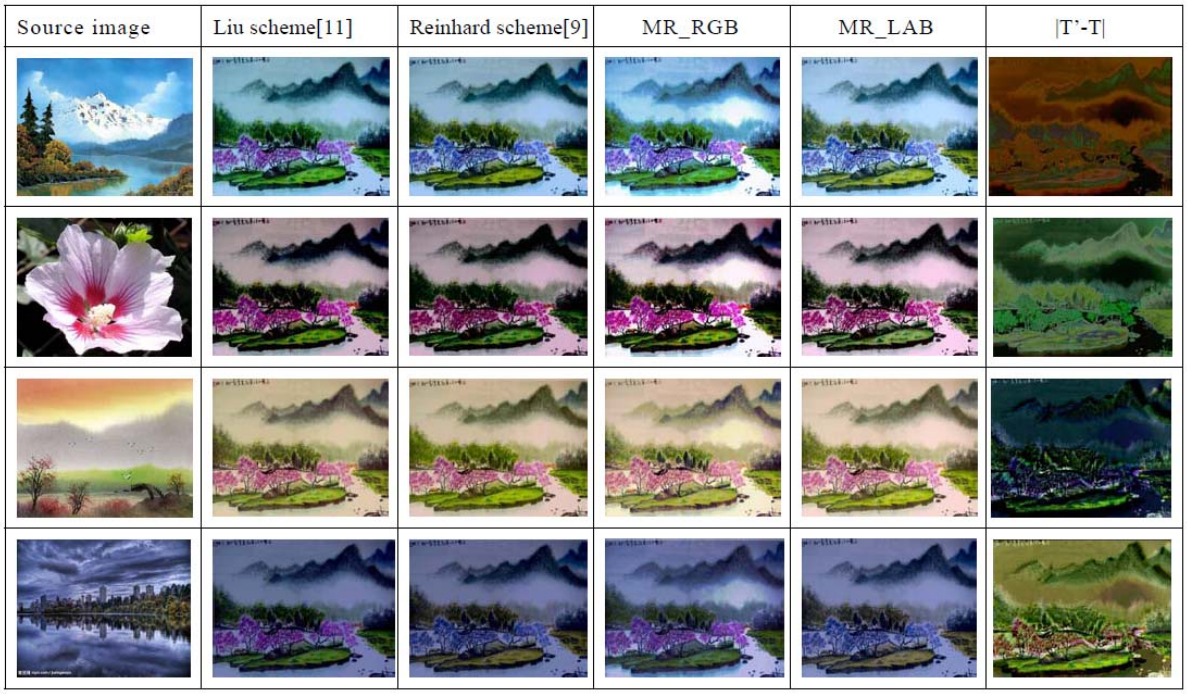


Figure 5: The box-plots of target image, source image and color transferred images



Here, |T′–T| is the absolute difference images between the original target image (T) and the color transferred image (T′) obtained using the MRA scheme in the LAB color space.

Figure 6: The performance in visual result for different techniques

To evaluate our approach, we compared our results with two color transfer algorithms: the color transfer technique proposed by Reinhard et al. [9] and the approach proposed by Liu et al. [11]. Figure 6 is used to demonstrate the color transfer visual results with different schemes for four source images. The items in Figure 6, Column 1 show the input source color images, Column 2 shows the color transferred images obtained by using Liu scheme, Column 3 shows the color transferred images obtained by using Reinhard et al. scheme, column 4 shows the color transferred images obtained using MRA scheme on RGB color space, column 5 shows the color transferred images obtained using MRA scheme on LAB color space and column 6 shows the absolute difference images between the original target image and the color transferred image obtained using the MRA scheme in the LAB color space. The absolute difference images show that the target image colors are almost completely replaced with the source image colors. In the visual result, Figure 6 clearly shows that the presented algorithm provides superior results over that of the other two schemes.

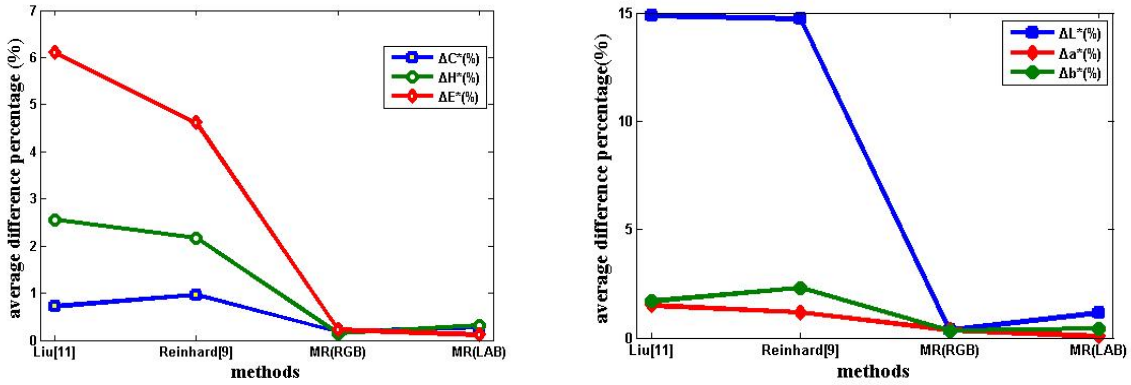
The statistical values (mean, standard deviation (STD)) of the measurement metrics for each image in L^*, a^*, b^*, C^*, H^* and E^* are arranged into Table 11. Table 11 demonstrates that these statistical values are almost the same between the source images and the corresponding transferred images obtained using the proposed scheme. Due to the numerical differences between colors in the CIELAB system is very consistent with human visual perceptions, the color distance in terms of CIELAB components really indicates how much the color transferred image differs from the source image. To present the color distance between the transferred image and source image with CIELAB component units is the most suitable way to measure the performance of color transferring schemes. In this paper, the color distance measured with CIELAB component units was conducted

Table 11: Statistical values of CIELAB components of target image, source image and result mages

CIELAB Components		L^*		a^*		b^*		C^*		H^*		E^*	
		MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
Target image		212.64	50.25	121.70	11.27	136.43	14.68	183.60	7.47	41.84	5.13	283.45	34.10
Test 1	source	151.33	60.06	119.40	6.67	114.10	22.24	165.69	18.99	46.76	4.59	230.38	35.28
	Reinhard [9]	129.36	58.00	118.85	7.59	118.62	21.90	169.04	12.60	45.45	6.40	219.05	29.11
	Liu [11]	128.09	60.06	116.82	13.88	117.59	19.72	167.20	10.03	45.01	7.25	217.34	28.71
	MR_LAB	151.46	58.84	119.43	6.71	115.34	21.09	167.04	12.40	46.37	6.05	230.86	34.09
	MR_RGB	151.33	59.52	118.36	14.72	113.36	23.69	165.71	13.32	46.58	8.17	229.82	35.52
Test 2	source	121.42	85.60	135.87	15.51	127.13	11.44	186.64	12.64	46.83	4.34	234.83	43.82
	Reinhard [9]	99.81	67.13	132.97	13.96	128.38	9.62	185.47	6.98	45.93	4.68	219.08	30.36
	Liu [11]	100.40	68.72	133.17	14.51	127.63	13.24	185.32	8.24	46.18	5.36	219.65	31.11
	MR_LAB	126.48	75.91	135.80	13.93	126.67	10.10	186.37	6.96	46.92	4.75	234.68	38.32
	MR_RGB	122.33	83.85	135.58	17.25	126.82	15.89	186.78	11.36	46.86	6.05	234.82	43.26
Test 3	source	170.21	47.71	131.05	10.97	146.97	15.14	197.33	13.60	41.81	3.72	263.49	30.75
	Reinhard [9]	153.13	47.46	128.52	11.69	149.38	14.24	197.79	7.04	40.78	4.84	253.34	26.35
	Liu [11]	152.38	48.18	129.99	10.27	147.12	11.17	196.81	6.21	41.49	3.96	252.07	27.84
	MR_LAB	170.21	47.62	131.04	10.74	146.78	14.78	197.47	7.57	41.85	4.71	263.59	28.45
	MR_RGB	170.06	47.52	130.72	9.79	146.49	12.80	196.85	7.63	41.81	4.08	262.89	29.54
Test 4	source	96.57	45.48	131.75	2.95	107.11	12.48	170.15	6.80	51.02	3.58	199.82	21.51
	Reinhard [9]	80.64	41.62	131.48	3.70	109.90	13.23	171.78	6.92	50.25	3.87	193.85	14.59
	Liu [11]	80.84	42.63	130.45	10.14	110.51	14.88	171.80	6.40	48.81	5.47	194.20	13.92
	MR_LAB	96.93	44.09	132.00	3.30	106.97	12.37	170.26	6.43	51.10	3.65	200.11	18.02
	MR_RGB	97.17	44.65	131.86	9.89	107.24	14.75	170.76	6.73	50.95	5.38	200.60	21.18

Table 12: Difference from transferred image to source image

Distance		ΔL^*	ΔL^* (%)	Δa^*	Δa^* (%)	Δb^*	Δb^* (%)	ΔC^*	ΔC^* (%)	ΔH^*	ΔH^* (%)	ΔE	ΔE^* (%)
Liu [11]	Test 1	–23.24	–15.36	–2.58	–2.16	3.49	3.06	1.51	0.91	–1.75	–3.74	–13.04	–5.66
	Test 2	–21.02	–17.31	–2.70	–1.99	0.50	0.39	–1.32	–0.71	–0.65	–1.39	–15.18	–6.46
	Test 3	–17.83	–10.48	–1.06	–0.81	0.15	0.10	–0.52	–0.26	–0.32	–0.77	–11.42	–9.46
	Test 4	–15.73	–16.29	–1.30	–0.99	3.40	3.17	1.65	0.97	–2.21	–4.33	–5.62	–2.81
	Abs-Avg	**19.46**	**14.86**	**1.91**	**1.488**	**1.885**	**1.680**	**1.250**	**0.713**	**1.233**	**2.558**	**11.315**	**6.098**
Reinhard [9]	Test 1	–21.97	–14.52	–0.55	–0.46	4.52	3.96	3.35	2.02	–1.31	–2.80	–11.33	–4.92
	Test 2	–21.6	–17.80	–2.9	–2.13	1.25	0.98	–1.17	–0.63	–0.90	–1.92	–15.75	–6.70
	Test 3	–17.08	–10.03	–2.53	–1.93	2.41	1.64	0.46	0.23	–1.03	–2.46	–10.15	–3.85
	Test 4	–15.93	–16.50	–0.27	–0.20	2.79	2.60	1.63	0.96	–0.77	–1.51	–5.97	–2.99
	Abs-Avg	**19.15**	**14.71**	**1.563**	**1.180**	**2.743**	**2.295**	**1.653**	**0.960**	**1.003**	**2.173**	**10.80**	**4.615**
MR (RGB)	Test 1	0.00	0.00	–1.04	–0.87	–0.74	–0.65	0.02	0.01	–0.18	–0.38	–0.56	–0.24
	Test 2	0.91	0.75	–0.29	–0.21	–0.31	–0.24	0.14	0.08	0.03	0.06	–0.01	–0.01
	Test 3	–0.15	–0.09	–0.33	–0.25	–0.48	–0.33	–0.48	–0.24	0.00	0.00	–0.60	–0.23
	Test 4	0.60	0.62	0.11	0.08	0.13	0.12	0.61	0.36	–0.07	–0.14	0.78	0.39
	Abs-Avg	**0.415**	**0.365**	**0.443**	**0.353**	**0.415**	**0.335**	**0.313**	**0.173**	**0.070**	**0.145**	**0.488**	**0.218**
MR (LAB)	Test 1	0.13	0.09	0.03	0.03	1.24	1.09	1.35	0.81	–0.39	–0.83	0.48	0.21
	Test 2	5.06	4.17	–0.07	–0.05	–0.46	–0.36	–0.27	–0.14	0.09	0.19	–0.15	–0.06
	Test 3	0.00	0.00	–0.01	–0.01	–0.19	–0.13	0.14	0.07	0.04	0.10	0.10	0.04
	Test 4	0.36	0.37	0.25	0.19	–0.14	–0.13	0.11	0.06	0.08	0.16	0.29	0.15
	Abs-Avg	**1.388**	**1.158**	**0.090**	**0.070**	**0.508**	**0.428**	**0.468**	**0.270**	**0.150**	**0.320**	**0.255**	**0.115**



(a) (b)

Figure 7: The absolute difference ratio from transferred image to source image

to measure the color transferring performances among the presented MRA approach, Liu approach [11] and Reinhard et al. scheme [9]. The color distance from the transferred images to the source image measured with CIELAB component units are arranged into Table 12. The comparisons in terms of the absolute distance ratios among previous schemes are plotted in Figure 7. Both Table 12 and Figure 7 illustrate that the presented MRA approach can transfer the color between images better than the other two schemes.

4. **Conclusions.** Color is the most important feature in color images and is applied in many different areas, such as medical image analysis, video object extraction, image compression, tracking systems. Transferring a source image's color to a target image involves changes the target image's color to enhance the target image's color features. This paper presented a multiple regression analysis based algorithm for global color transfer between images is presented. The above experimental results and performance analyses validate that the presented approach has three major advantages: (i) the proposed algorithm is manual free, simple, effective and accurate in transferring color between images without any change in the image details; (ii) the proposed algorithm saves time and the time consumption is independent of the number of bins selected and the degree of regression; (iii) there are no restrictions in the image dynamic color ranges in the proposed algorithm. In the future, we are going to combine the proposed scheme with image segmentation schemes to improve the color transfer results.

Acknowledgment. This work is partially supported by National Chin-Yi University of Technology, Taiwan under contract NCUT 11-R-CL-001. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] X.-H. Han, Y.-W. Chen and J. M. Lei, A spatio-chromatic ICA based noise reduction in color images, *International Journal of Innovative Computing, Information and Control*, vol.4, no.3, pp.661-669, 2008.

[2] Y.-S. Lin, M.-Z. Rau and S.-J. Lee, Applying self-constructing clustering to color image quantization, *ICIC Express Letters,* vol.3, no.3(B), pp.813-818, 2009.

[3] S.-C. Huang, A new scheme for generating initial palettes of color quantized images, *International Journal of Innovative Computing, Information and Control*, vol.6, no.7, pp.3011-3022, 2010.

[4] M. V. Sudhamani and C. R. Venugopal, Nonparametric classification of data viz. clustering for extracting color features: An application for image retrieval, *ICIC Express Letters*, vol.1, no.1, pp.15-20, 2007.

[5] T. Pouli and E. Reinhard, Progressive histogram reshaping for creative color transfer and tone reproduction, *Proc. of the Symposium on Non-Photorealistic Animation and Rendering*, pp.81-90, 2010.

[6] S. De, A. Dagan, P. Roan, J. Rosen, M. Sinanan, M. Gupta and B. Hannaford, CIELab and RGB color values of in vivo normal and grasped porcine liver, *Proc. of the 15th Medicine Meets Virtual Reality*, pp.109-111, 2007.

[7] S. Xu, Y. Zhang, S. Zhang and X. Ye, Uniform color transfer, *Proc. of IEEE International Conference on Image Processing*, pp.940-943, 2005.

[8] D. Freedman and P. Kisilev, Object-to-object color transfer: Optimal flows and SMSP transformations, *IEEE Conference on Computer Vision and Pattern Recognition*, pp.287-294, 2010.

[9] E. Reinhard, M. Ashikhmin, B. Gooch and P. Shirley, Color transfer between images, *IEEE Computer Graphics and Applications*, vol.21, no.5, pp.34-41, 2001.

[10] D. L. Ruderman, T. W. Cronin and C. C. Chiao, Statistics of cone responses to natural images: Implications for visual coding, *J. Optical Soc. of America*, vol.15, no.8, pp.2036-2045, 1998.

[11] C. C. Liu and G. N. Hu, A re-coloring algorithm for a color image using statistic scheme in CIE L^* a^* b^* color space, *International Symposium on Computer, Communication, Control and Automation*, pp.240-243, 2010.

[12] G. R. Greenfield and D. H. House, Image recoloring induced by palette color associations, *Journal of WSCG*, vol.11, no.1, pp.189-196, 2003.

[13] Q. Zhou, L. Ma, M. Celenk and D. Chelberg, Natural scene synthesis using multiple eigenspaces, *International Conference on Image Processing*, vol.3, pp.II-121-4, 2003.

[14] C. C. Liu and W. Y. Chen, Screw pitch precision measurement using simple linear regression and image analysis, *Applied Mathematics and Computation*, vol.178, pp.390-404, 2006.

[15] F. Lopez, J. M. Valiente, J. M. Prats and A. Ferrer, Performance evaluation of soft color texture descriptors for surface grading using experimental design and logistic regression, *Pattern Recognition*, vol.41, pp.1761-1772, 2008.

[16] S. S. Deeb, Genetics of variation in human color vision and the retinal cone mosaic, *Current Opinion in Genetics & Development*, vol.16, no.3, pp.301-307, 2006.

[17] C. Clausen and H. Wechsler, Color image compression using PCA and back propagation learning, *Pattern Recognition*, vol.33, pp.1555-1560, 2000.

[18] I. Andreadis and Ph. Tsalides, Analog computation of image chromaticity, *Real-Time Imaging*, vol.3, pp.1-6, 1997.

[19] M. Melgosa, Testing CIELAB-based color-difference formulas, *Color Research and Application*, vol.25, no.1, pp.49-55, 2000.

[20] M. E. Celebi, H. A. Kingravi and F. Celiker, Fast colour space transformations using minimax approximations, *IET Image Processing*, vol.4, no.2, pp.70-80, 2010.

[21] Y. Chen and P. Hao, Optimal transform in perceptually uniform color space and its application in image retrieval, *The 7th International Conference on Signal Processing*, vol.2, pp.1107-1110, 2004.

[22] Y. Han, S. Lee, J. Choi and S. Lee, A simple and efficient color recovering system for content sharing website, *IEEE Transactions on Consumer Electronics*, vol.56, no.2, pp.863-869, 2010.

[23] P. Djeu, M. Quinlan and P. Stone, Improving particle filter performance using SSE instructions, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.3480-3485, 2009.