Design Considerations of Color Image Processing Pipeline for Digital Cameras

Wen-Chung Kao, *Member*, IEEE, Sheng-Hong Wang, Lien-Yang Chen, and Sheng-Yuan Lin

Abstract — *Although many individual image processing steps have been well addressed in the field, very few good image pipeline designs were proposed to integrate these processing stages. In this paper, a new color image processing pipeline (IPP), which processes the image raw data captured from CCD/CMOS sensors and converts to the final color with exposure corrected, is presented. It bridges the gap between theoretical color imaging sciences and practical IPP implementation issues found in digital imaging systems. We demonstrate the IPP by processing different scenes of pictures and the results show that the proposed pipeline performs well in diverse scenes and illuminants*[1].

Index Terms —Image processing pipeline, auto white balance, exposure adjustment, and color reproduction.

I. INTRODUCTION

The ordering as well as the algorithm adopted in each stage for processing a color image is usually called color image processing pipeline (IPP). Although many algorithms have been proposed to address the issues of color image processing, covering the stages from raw image data to the final JPEG file [1]-[3], only a few IPPs were presented to integrate these steps while compromising computational complexity and image quality [4]-[10]. A good IPP should consider the ordering as well as the algorithms for color reproduction, tone reproduction, noise filtering, and edge enhancement. Even though many digital cameras have been designed, the design of IPP is still kept confidential by most digital camera design companies. A typical IPP includes the following stages: color filter array (CFA) interpolation, auto white balance (AWB), color correction, noise filtering, tone reproduction, gamma correction, edge enhancement, and color saturation enhancement. Without a well-designed processing flow, it is impossible to get good and stable image quality, even if suitable algorithms for individual stages are selected. Assigning these processing steps into appropriate stages in the IPP is a complicated task, and the ordering of IPP stages will dominate the final picture quality.

The major design objective of IPP is to reproduce the colors such that the camera can obtain consistent result across changes in the scene illumination. The colors of the pictures captured by digital cameras are affected by the physical content of the scene, the illumination incident on the scene and the spectral sampling properties of the sensor. The color spectral response of the image sensor needs to match that of typical human eye, as defined by Commission Internationale de l'Eclairage (CIE). Real image sensors, however, cannot meet the requirement. Recently, it is widely accepted to adopt the standard RGB (*sRGB*) color space as the reference color space. Therefore, the designer should perform a calibration in order to map the color space of the sensor to linear *sRGB* space first. This process is also called color correction. Once this calibration is well performed, it is believed that much better color reproduction can be achieved. The remaining issues for color reproduction are estimating the color temperature of the illuminants and compensating for their effects on the surface of the objects. The step to solve these problems is usually called AWB [11]-[14].

In this paper, we aim at the flow design of IPP as well as new algorithms for exposure adjustment and color reproduction. The IPP design issues are well-addressed and a novel IPP to address these issues is proposed. The proposed system is based on analysis of the color effects and works on appropriate color spaces while compromising computational complexity and image quality throughout the entire IPP. Unlike traditional heuristic color reproduction algorithms, the proposed color reproduction algorithm combines macro edge detection with gray world assumption and fully utilizes the sensor spectral responses by calibrating color temperature curve. Highly accurate estimation of the illuminants can be achieved.

In the following sections, we will first describe the design issues of a color IPP. Then the proposed IPP architecture and its design considerations are stated. The exposure correction problem is discussed in Section IV. In Section V, we present a new approach to AWB that is based on color temperature curve calibration of the sensor. In Section VI, the color saturation enhancement approach is presented. Experimental results and conclusions of the proposed image pipeline are given in Section VII and VIII.

[1] This work was supported in part by the National Science Council, R.O.C., under Grants NSC 95-2221-E-003-018-MY2.

Wen-Chung Kao and Lien-Yang Chen are with Institute of Applied Electronic Technology and Department of Industrial Education, National Taiwan Normal University, Taipei, Taiwan, R.O.C (e-mail: jungkao@ntnu.edu.tw).

Sheng-Hong Wang is with the Computer & Information Networking Center, National Taiwan University, Taipei, Taiwan, R.O.C.

Sheng-Yuan Lin is now with Research and Development Department, Foxlink Image Technology Corporation, Taiwan, R.O.C. (email: malcolm.lin@foxlinkimage.com).

Contributed Paper
Manuscript received October 15, 2006

II. Problem Statement of Designing Color Image Processing Pipeline

The fundamental problems in the existing color IPPs come from the following facts: (a) The nonlinear transformations such as gamma correction and tone reproduction are usually misused and assigned in the early stages of image pipeline. The CCD sensor should be designed to work in linear region so that the raw data output can be a good representation of the intensity of lightness for each color component. However, some image pipelines perform gamma correction in the early stages, which tend to result in false colors because of the nonlinear transformations. (b) Many available algorithms of AWB only rely on heuristic ideas such as simple gray world assumption or white points/regions estimation [15]-[16]. These algorithms can work fine in typical scenes, but may fail in some particular scenes. This is because they do not fully utilize the sensor characteristics to predict the types of illuminants. (c) The post-processing steps such as noise filtering and edge enhancement may generate some undesired colors or artifacts if they are applied on unsuitable stages. (d) It is really difficult to enhance color saturation without affecting relative luminance of the processed image area. Even if the images are processed in *CIEXYZ* or *HSI* color space, the color difference in chromaticity diagram are not defined uniformly such that it is too complicated to handle the color balance problem in these spaces. (e) The radiometric/colorimetric relation between the raw image data and scene radiances/spectral distributions should be calibrated so that the raw data can be transformed to linear *sRGB* color space correctly. However, sometimes this transformation is not performed or it is assigned to an inappropriate stage.

III. The Proposed Color Image Processing Pipeline

In order to address the issues mentioned above, a new IPP is proposed in this paper. The design of the proposed IPP is based on the following considerations: (a) Auto white balance should be performed at early stage. This is because the linear processing in the later stages relies on correct ratio between the three channel values and the white balance result will not be affected too much. If white balance adjustment is applied after nonlinear processing, the colors tend to become incorrect. An additional advantage is as the color characteristics of the image sensor are calibrated, the raw data contains information about the color temperature of the light sources. After correct estimation of the color temperature, sensor white balance and light source white balance can be combined to reduce computation amount. (b) It is necessary to have complete tri-stimulus values for each pixel in order to perform color processing and other image enhancement steps. Raw data lack enough color information for each pixel before color interpolation is executed. In addition, color interpolation may also generate some artifacts, such as smoothing the edges or spreading noise, which need to be filtered or smoothed out in the noise filtering stage. Hence it would be better to carry out color interpolation before performing any other processing stages. (c) Most edge detection and noise filtering algorithms are executed on the *Y* components in *YCbCr* color space or on *L** component in *CIELAB* color space. Color psychophysics indicates that human eyes are more sensitive to noise in luminance but not so much for chrominance components. However, noise is produced in sensor *RGB* color space. It should be better if the noise can be filtered out in sensor *RGB* space before they are transformed to other color space for further processing. On the other hand, it is possible for some edges to locate on the boundary of two different chromatic blocks with similar luminance. The edge may not be detected if the edge detection algorithm only focuses on luminance components. (d) Color correction, which is usually realized by multiplying a 3×3 matrix, may also enlarge random noise. Hence noise filtering should be performed before color correction. (e) Typical noise filters remove random noise or artifacts in the image but also smooth out the edges as a side effect. This results in images of poor sharpness. It would be better for the noise filter to use the edge information extracted from the previous stage to prevent the edges from being blurred. (f) *CIELAB* is a relatively uniform color space with the luminance and chrominance components separated such that it is much easier to handle color saturation enhancement. (g) It is considered beneficial for the edge enhancement algorithm to be applied only on *L** component to avoid generating color artifacts. (h) Applying tone reproduction in the later processing stage can improve the rendition of the pictures. (i) The nature of gamma correction is nonlinear and it is used for compensating the nonlinear response of the CRT monitor or the LCD monitors. It is difficult to maintain the chrominance correctness after this type of nonlinear transformation. Hence gamma correction should be put at the later stage of IPP.

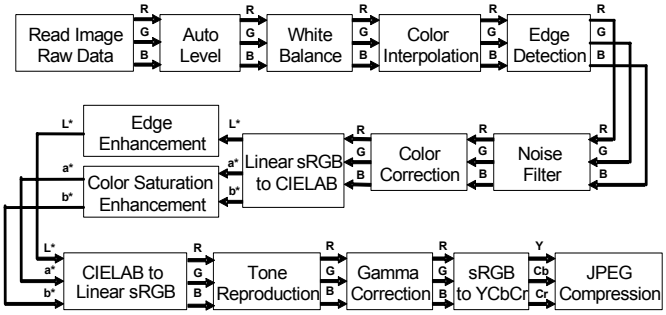

Fig. 1. The proposed image processing pipeline.

According to the above design considerations, we propose a new IPP shown in Fig. 1. The IPP includes the following stages: (a) Auto level stretch for clamping black level and correcting exposure value, (b) auto white balance that is based on the raw data captured from CCD/CMOS sensors, (c) color interpolation to produce the missing color components for each pixel, (d) edge detection on *RGB* channels, (e) noise filter that can refer to the information obtained from edge detection and keep the edge from being smoothed out, (f) the transformation of the tri-stimulus values from sensor RGB

color space to *CIELAB* color space, (g) edge enhancement on *L* components according to the edge extraction result, (h) color saturation enhancement by scaling the *a** and *b** components, (i) tone reproduction by stretching on *RGB* color space for enhancing rendition of the final pictures, and (j) gamma correction for compensating for the nonlinearity of the monitors.

Since the noise filtering as well as edge enhancement, color correction, and tone reproduction algorithms used in the proposed system have been discussed in our previous works [17]-[19], in this paper, we only focus on the algorithm design for exposure correction, auto white balance, and color saturation enhancement in the following sections.

IV. Exposure Correction by Auto Level Stretching

Step 1: Calculate the histogram $H_i(0 \le i \le \Phi)$ of the tristrmulus values *R*, *G*, and *B* for all pixels in the image.

Step 2: Find the control points α and β such that they satisfies (1):

$$\left(\sum_{i=0}^{\alpha-1} H_i\right) \le 0.001 \cdot \Omega \le \left(\sum_{i=0}^{\alpha} H_i\right)$$
$$\left(\sum_{i=0}^{\beta-1} H_i\right) \le 0.999 \cdot \Omega \le \left(\sum_{i=0}^{\beta} H_i\right) \quad (1)$$
$$\Omega = \sum_i H_i$$

Step 3: For the original tri-stimulus (*RGB*) values V of each pixel, stretch out their new values $\hat{V}$ with (2):

$$\hat{V} = \begin{cases} (V-\alpha) \times \left(\Phi / (\beta-\alpha)\right) & \alpha < \beta \end{cases} \quad (2)$$

Fig. 2. The exposure correction algorithm.

The circuit offset caused by dark current and camera flare resulted from lens imperfections must be corrected first before other processing steps. Even in some cases the actual dynamic range of the scene is narrower than the sensor dynamic range. Under such conditions we can perform auto level stretch in order to subtract the offset factors and allow the sensor to express more details of the scene.

The proposed algorithm shown in Fig. 2 first analyzes the histogram $H_i(0 \le i \le \Phi)$ of the raw data, where Φ denotes the maximum code of the imaging system. The next step is to find the two control points corresponding to 0.1% and 99.9%, respectively. These two values are selected based on experience and can be adjusted by the designer. The dynamic range between the two control points is stretched out to fit the entire dynamic range of the output codes. It should be noted that this process is done before color interpolation. Each pixel only contains one of the three color components. The exposure correction is done without considering what the color is in the pixel. In addition, the control point α is mainly used to remove the possible black level offset and flare caused by the lens imperfection. Thus the value α should be limited to a low level, even if the distribution of the pixels only occupies relatively higher levels. If the histogram spreads only on the upper portion of the input dynamic range and it is expected to actually increase the dynamic range of the final image. The easiest way is to modify the tone reproduction curve in the later stage to map the input dynamic to the output dynamic range. If the raw image data are subtracted by a higher constant value during stretching, the color will become incorrect since the ratios of the *RGB* components are also changed. One important thing in applying auto-level stretch is dynamic range stretching usually reduces data precision, or increase quantization noise significantly. Hence the input dynamic range should be much wider than the output. Since the output JPEG (Joint Photographic Experts Group) file only adopts 8-bit data, the final image will contain significant contour effect if the input image sensor only generates 8-bit of raw data. In practical camera system design, the sensor input should be equipped with a 10-bit or even 12-bit analog-to-digital-converter (ADC).

V. Auto White Balance

The proposed AWB algorithm combines three illuminant estimation technologies: color temperature curve (CTC) calibration, gray world assumption, and macro edge detection. Macro edge detection extracts object boundary information with higher noise immunity and prevents the statistical histograms from being dominated by larger areas of uniform color. According to gray world assumption, the statistics values of the three color channels for these macro edges are assumed to be very close in typical scenes. Finally, the color temperature of the illuminant is estimated by projecting the statistical values onto the CTC, which is calibrated for a given sensor to reflect the spectral property of gray objects illuminated by natural light sources.

In the following subsections, we will first give problem formulation of AWB, and then the sensor calibration for guiding AWB is proposed and the complete algorithm is stated.

A. The Problem Formulation of AWB

The measuring values of image sensors are dependent on the three factors: (a) spectral power distribution of the incident illuminant $L(\lambda)$ falling on the objects, where λ represents the wavelength, (b) surface reflectance $U(\lambda)$ of the objects in the scene in the direction of the camera, and (c) relative spectral response $P^{(k)}(\lambda)$ (k = *R, G,* or *B* channel) of the CCD/CMOS sensors. The image formation $I^{(k)}$ for each channel k of the sensors can be described by (3). The objective of AWB is to compensate for the color shift caused by the illuminant $L(\lambda)$.

$$I^{(k)} = \int P^{(k)}(\lambda) U(\lambda) L(\lambda) d\lambda \quad (3)$$

The aim of AWB is to guess the illumination under which the image is taken and compensate the color shift affected by the illuminate. The AWB problem is usually solved by adjusting the gains of the three primary colors *R, G*, or *B* of the image sensors to make a white object to appear as white under different illuminants.

Under different illuminants, typical AWB adjustment is to make the levels of the red (*R*), green (*G*) and blue (*B*) components balanced for the object whose nature color is gray or white. Many available algorithms for AWB rely on some heuristic ideas such as simple gray world assumption or Retinex theory [15]. These heuristic rules can attain good results for many scenes, but it is easy to enumerate the fail cases. For example, with gray world assumption, an image of a gray object in front of a large green grass field will more likely integrate to green rather than gray. Even if applying Retinex theory that detects the maximum values for *R*/*G*/*B* values in the captured image, it still can not work normally if parts of the regions are overexposed. The scenes mentioned above are typical cases because usually the dynamic range of the sensor can not cover the entire range of outdoor scenes. The drawbacks in these kinds of algorithms lie in the fact that they do not fully utilize the sensor characteristics to help predict the types of illuminants. That is, it lacks enough information in the raw data for predicting the color temperature of the ambient light. Other algorithms first calibrate the sensor characteristics and create correlation tables or gamut maps for different illuminants [14]. It is believed that the color reproduction result with sensor calibration should be better than the ones without sensor calibration. But these methods fully rely on the huge image database, which includes several thousands of pictures and each picture has been annotated with its illuminant condition. These labels are referred to as "truth" annotations and are used for training the correlation table or gamut map.

B. Color Temperature Curve Calibration of the Image Sensor

The proposed AWB process is guided by color temperature curve calibration. Assuming the photo-response non-uniformity (PRNU) and defects of the sensor and the vignette effect of the lens are negligible or well corrected, the fundamental step of AWB design is to find the color characteristic of the sensor. By using the standard Macbeth color checker [20] as the scene target with a uniform natural light source, the average values of six gray blocks in the bottom row are used to calculate the *R*, *G* and *B* components. With a simple linear approximation to the data points, the offset caused by lens flare can be removed and sensor white balance can be characterized.

As shown in Fig. 3, the horizontal axis represents the reflectance of the gray blocks in color checker, and the three measured *R*/*G*/*B* values corresponding to the gray blocks should be linear because the reflectance of these blocks have been calibrated as direct proportion. The offset values are denoted as s_R, s_G, and s_B for *R*, *G*, and *B* channels, respectively. Note that the sensitivity of *G* channel is usually higher than the other ones for typical image sensors. In addition, with light sources of higher color temperature, the stimulus values of *B* channel for these gray blocks are usually higher than the stimulus values of *R* channel.

After subtracting the offset values of s_R, s_G, and s_B for *R*, *G*, and *B* channels, respectively, the slopes p_R, p_G, and p_B of the three lines can be used to represent relative sensitivities of the colors and the ratios of p_G / p_R and p_G / p_B can be computed to characterize the sensor performance under a given illuminant. By taking raw images under sun light at different moments of the day, it is possible to get several values of p_G / p_R and p_G / p_B under different color temperatures. In real application, average values of *R*, *G* and *B* are used. It is possible then to take *G*/*R* and *G*/*B* as coordinates and plot these points for different light sources and the resulting curve is called color temperature curve (CTC) of the sensor.

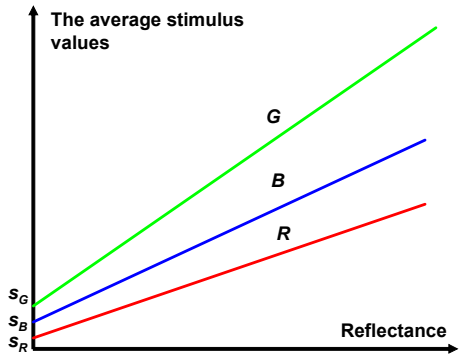

Fig. 3. The offset values calibration for R, G, and B channels with the six gray blocks in the color checker.

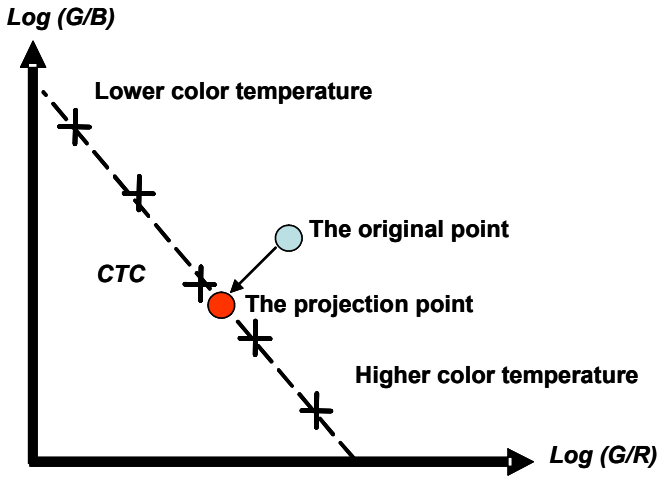

Fig. 4. Color temperature curve (CTC)

Since the data we are dealing with are ratios of two values, it is customary to take logarithm of the values calculated such that multiplication can be replaced with addition. The resulting curve is roughly linear for natural light sources [21]-[22] as shown in Fig. 4. In general, when the color temperature of the light source is higher, the blue component is stronger and the red component is weaker. The sensor output level of *R* is lower than *B*. Thus the ratio *G/R* is larger than *G/B*, and the data point lies on the lower right portion of the space. On the other hand, with lower color temperature illuminant, we will obtain opposite result and the data point will lie on the upper left corner of the space. From this feature, it can be concluded that essentially the slope of the curve is negative.

With proper constraints and statistical analysis performed, it is possible to derive an average color data point in the $\log(G/B)-\log(G/R)$ space. Note that the measurement point for gray objects or patches will locate in the CTC if they are illuminated by a natural light source. However, the real scenes are colorful. The approach to utilizing the calibrated CTC is combining macro edge detection and gray world assumption. The algorithm detects macro edges and then the average values of three color channels are calculated based on only these macro edges. The point in the $\log(G/B)-\log(G/R)$ plane can be determined accordingly. As shown in Fig. 4, the point may not locate on the CTC curve because the proposed approach based on macro edges detection and gray world assumption is still just an estimation of color temperature. This estimation point is then projected on the CTC, and the projection point is assumed as the color temperature of the ambient light source.

C. The Detailed AWB Algorithm

The detailed AWB algorithm is listed in Fig. 5. Compared to other approaches, the improvements include following items: (a) It fully utilizes the sensor characteristics with calibrated color temperature curve. (b) In order to prevent large uniform color blocks from dominating the AWB calculation, the proposed algorithm performs edge detection on coarse grids (16×16 macro blocks) instead of performing edge extraction on the pixel level. The noise immunity with macro blocks is much better than with pixel levels. (c) Unlike color by correlation, the color temperature curve can be calibrated with few images taken under different illuminants.

VI. Color Saturation Enhancement

Color saturation enhancement of the proposed IPP is realized in *CIELAB* color space. The reason is *sRGB*, *YCbCr*, *HSI* and *CIEXYZ* tri-stimulus values were not defined with explicit consideration of color differences. Two colors with a small difference in the tristimulus values may look very different or virtually indistinguishable depending on where the two values are located in the non-uniform color spaces. Among the various uniform color spaces, *CIELAB* is the most widely used.

CIELAB color space is based on a color vision model. This space is transformed from *CIEXYZ* tristimulus values into an achromatic lightness values *L** and two chromatic values *a** and *b** using the transformations shown in (9) – (11), where X_n, Y_n, and Z_n are the tristimulus values of the reference (neutral) white point, and $f(t)$ is defined in (12).

Input: An image raw data
Output: The AWB corrected image data

Step1: Divide the whole image raw data into a two-dimensional array of 16×16 macro blocks:
$A_{i,j}(1\le i\le m,\quad 1\le j\le n)$

Step2: For each macro block $A_{i,j}$, compute the average red ($R_{i,j}$), green ($G_{i,j}$), and blue ($B_{i,j}$) values.

Step3: For each macro block $A_{i,j}$, compute the color deviation index $D_{i,j}$ based on (4).

$$D_{i,j}=\frac{(R_{i,j}-R_{i,j+1})^2+(G_{i,j}-G_{i,j+1})^2+(B_{i,j}-B_{i,j+1})^2}{(R_{i,j}+R_{i,j+1})^2+(G_{i,j}+G_{i,j+1})^2+(B_{i,j}+B_{i,j+1})^2} \quad (4)$$

Step4: For each macro block, set the boundary flags $F_{i,j}$ by (5).

$$F_{i,j}=\begin{cases}1 & D_{i,j}\ge\theta\\ 0 & D_{i,j}<\theta\end{cases} \quad (5)$$

where θ is the pre-defined threshold value, which is set as 0.1 in the current implementation.

Step5: For each macro block, check for the distance $S_{i,j}$ between the point and the CTC curve with (6). If it is longer than a threshold ϕ, it is treated as a saturated color and is removed in the statistics.

$$T_{i,j}=\begin{cases}0 & S_{i,j}\ge\phi\\ 1 & S_{i,j}<\phi\end{cases} \quad (6)$$

Step6: Calculate the weighted average red (*WR*), green (*WG*), and blue (*WB*) values with (7).

$$\begin{aligned}WR&=\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}R_{i,j}\Big/\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}\\ WG&=\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}G_{i,j}\Big/\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}\\ WB&=\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}B_{i,j}\Big/\sum_{i=1,j=1}^{i=m,j=n}T_{i,j}F_{i,j}\end{aligned} \quad (7)$$

Step7: Calculate $\log(WG/WR)$ and $\log(WG/WB)$

Step8: Find the projection point (R_p , B_p) from ($\log_{10}(WG/WR)$, $\log_{10}(WG/WB)$) to the calibrated CTC curve

Fig. 5. The AWB algorithm

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R_s \\ G_s \\ B_s \end{bmatrix} \quad (8)$$

$$L^* = 116 f\left(\frac{Y}{Y_n}\right) - 16 \quad (9)$$

$$a^* = 500\left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right) \quad (10)$$

$$b^* = 200\left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right) \quad (11)$$

$$f(t) = \begin{cases} t^{1/3} & t > 0.00885\epsilon \\ 7.787t + \dfrac{16}{116} & t \le 0.00885\epsilon \end{cases} \quad (12)$$

The chromatic a^* is a red-green opponent channel in that positive values for a^* indicate redder colors and negative values indicate greener colors. Similarly, b^* is a yellow-blue opponent color channel. Note that the radial distance $\sqrt{(a^*)^2 + (b^*)^2}$ and angular position $\tan^{-1}(a^*/b^*)$ in $a^* - b^*$ plane represent chroma and hue, respectively. The uniformity in *CIELAB* can be further improved by decomposing the total *CIELAB* color difference into lightness, hue, and chroma differences, and then introduce different weighting factors for these differences that are determined by the lightness, hue, and chroma of the pixels. However, the computation complexity is too high and usually it is good enough for typical digital imaging systems by processing color images in *CIELAB* color space.

The goal of color saturation enhancement is to increase the color saturation while keeping the hue and brightness of the pixels. As described above, one of the inherent properties of *CIELAB* color space is that radial distance and angular position represent chroma and hue of the color, respectively. It is much easier to handle color consistency and brightness than other types of color space such as *sRGB*, *YCbCr*. For each pixel, by only applying the same scaling factor (1.1× - 1.3×) on both chroma components a^* and b^*, the color saturation can be enhanced, but hue and luminance can be kept almost unchanged. After scaling the chroma components for each pixel, the relative ratio of the hue and chroma values between two different colors are not changed. Thus picture white balance as well as color calibration results can also be kept after color saturation operations. Note that this approach is just a heuristic approach that can maintain the brightness of the pixel after color saturation enhancement better. It may not be desirable to boost the chroma of some colors such as skin colors too much.

VII. Experimental Result

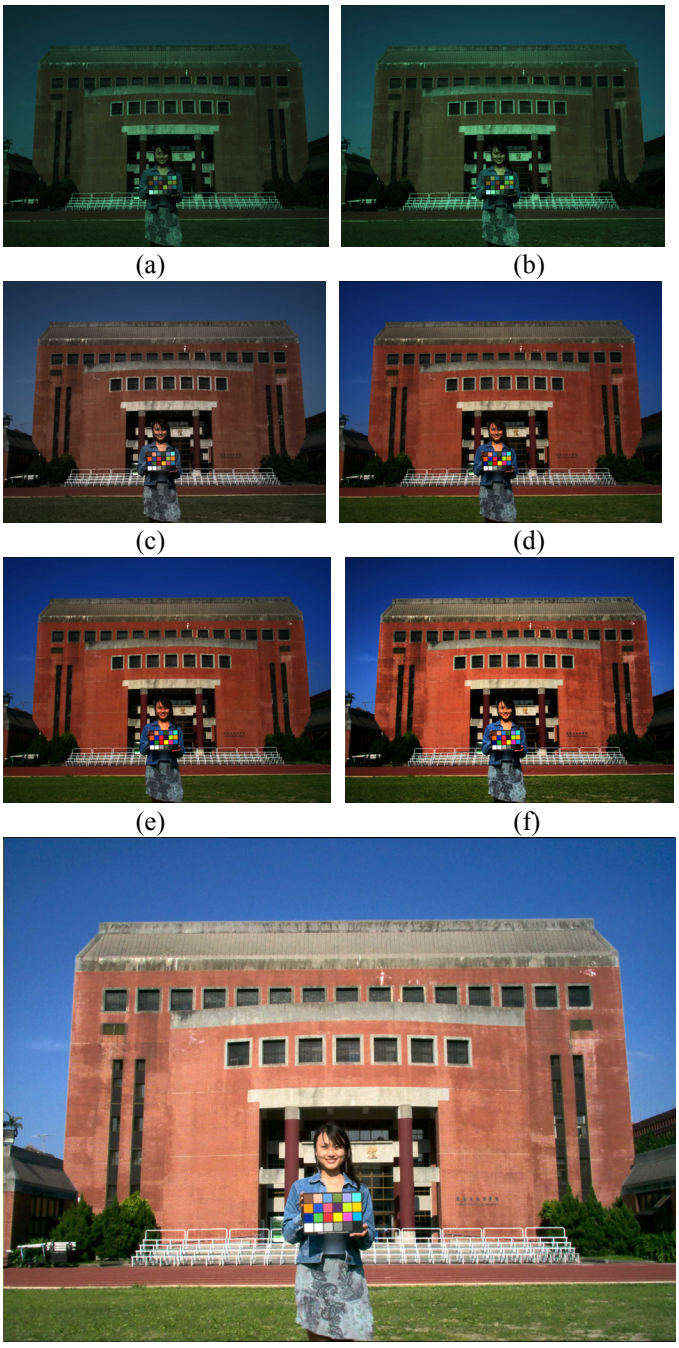

Fig. 6. An outdoor image. (a) the raw image, (b) the image after auto-level stretch, (c) the image after AWB, (d) the image after color correction, (e) the image with color saturation enhanced, (f) the image after tone reproduction, and (g) the final image after gamma correction.

In order to demonstrate the performance of the proposed image pipeline, we have taken several hundreds of pictures with a commercial digital camera, whose resolution is three mega pixels. The camera can store raw image data into flash memory so that we can test complete image pipeline from the raw data to the JPEG bit stream. Fig. 7 and 8 show the processing results of IPP for two pictures taken in different scenes, and the partial results are shown in the figures: (a) the

raw image, (b) the image after auto-level stretch, (c) the image after AWB, (d) the image after color correction, (e) the image color saturation enhanced, (f) the image after tone reproduction, and (g) the final image after gamma correction. It is worth noting that the algorithm also works fine with indoor environment and back-lit condition.

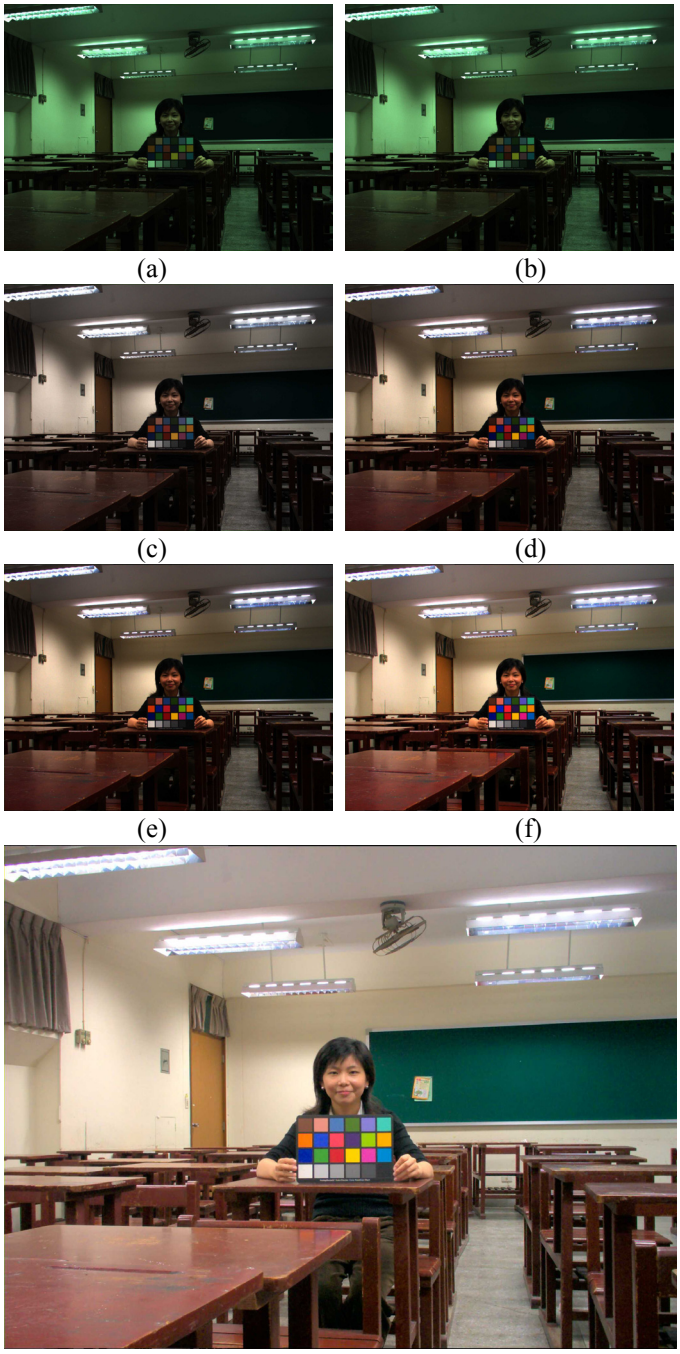

Fig. 7. An indoor image. (a) the raw image, (b) the image after auto-level stretch, (c) the image after AWB, (d) the image after color correction, (e) the image with color saturation enhanced, (f) the image after tone reproduction, and (g) the final image after gamma correction.

Since the scenes and light sources in the laboratory are much simpler than in real world, it can not be used to demonstrate the problems faced in real scenes. The taken pictures with raw data include not only the pictures taken in the laboratory but also in the real scenes, which include indoor and outdoor lighting conditions. By including a color checker in the scenes, it is easy to verify the color reproduction performance based on the color difference analysis for the 18 color blocks and 6 gray blocks of the color checker. Some test pictures as well as their processing results, which include the output images in each stage of the proposed IPP, can be found in the website [23].

The calibrated color temperature curve with a particular sensor is shown in Fig. 8 by measuring the six gray blocks in the color checker. In order to further demonstrate the performance of AWB, we use two pictures shown in Fig. 9. Most of AWB algorithms can not handle the white balance problem well if the pictures are taken under the scene with few colors. Just like these pictures, the scenes include large area with the same or similar colors. The images shown in Fig. 9-(a) and (d) are processed by the IPP but using gray world white balance. With color temperature curve and extracted macro edges shown in Fig. 9-(b) and (e), the resulting pictures that pass through entire pipeline are shown in Fig. 9-(c) and (f). It can be seen that the proposed AWB algorithm handle the colors well even if there exist large regions with red, green, and white colors.

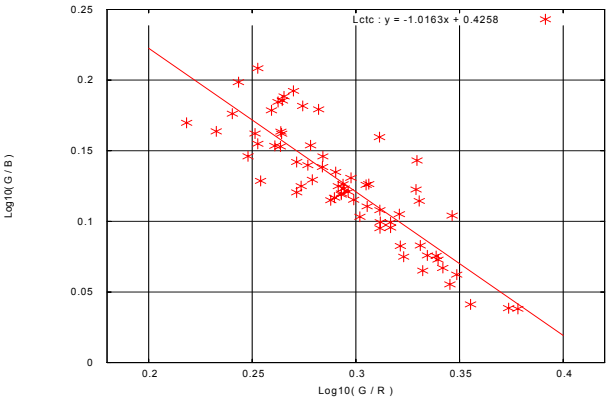

Fig. 8. An example of calibrated color temperature curve.

VIII. Conclusion and Discussion

We have proposed a new image processing pipeline (IPP) to address the issues of color and tone reproduction. With the advances in integrated circuit technology, the computation power is not always the major considerations for the design of image pipeline. On the contrary, the designers of high-end digital imaging systems usually struggle for the ideal image pipeline design instead of building a low cost imaging system with low performance. In this paper, the proposed image pipeline can be provided as a good reference, which can be optimized further based on the adopted hardware platforms. It is also possible to design an IPP chip with such a robust flow for real-time video and still image processing, which will be our future works.

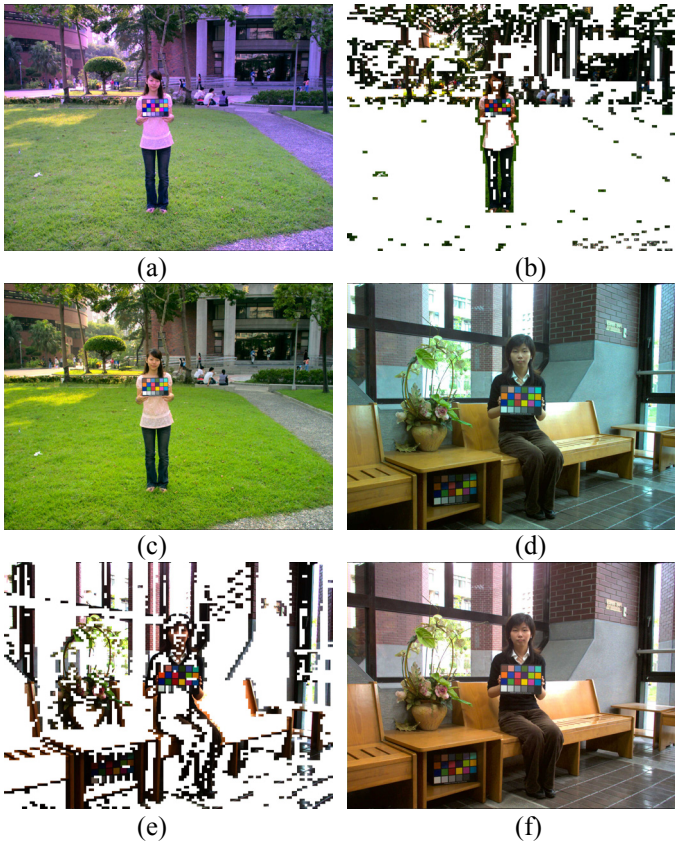

Fig. 9. The pictures with gray-world and with the proposed AWB algorithm. (a) the picture with gray world AWB, (b) the extracted macro edges of Fig. 9-(a), (c) the picture Fig. 9-(a) with the proposed AWB, (d) the other picture with gray world AWB, (e) the extracted macro edges of Fig. 9-(d), (f) the picture Fig. 9-(d) with the proposed AWB.

ACKNOWLEDGMENT

The authors would like to thank the helpful discussions with Dr. Hsien-Che Lee at Foxlink Peripherals Inc., Fremont, California.

REFERENCES

[1] G. Sharma and H. J. Trussell, "Digital color imaging," *IEEE Trans. Image Processing*, vol. 6, no, 7, pp. 901-932, Jul. 1997.

[2] G. Sharma, M. J. Vrhel, and H. J. Trussell, "Color imaging for multimedia," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1088-1108, Jun. 1998.

[3] H. C Lee, *Introduction to Color Imaging Science*, Cambridge University Press, 2005.

[4] K. Illgner, H. G. Gruber, P. Gelabert, J. Liang, Y. Yoo, W. Rabadi, R. Talluri, "Programmable DSP platform for digital still cameras", in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Mar. 1999, pp. 2235-2238.

[5] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew, "Color image processing pipeline," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 34 – 43, Jan. 2005.

[6] A. J. P. Theuwissen, "Image processing chain in digital still cameras," in *Proc. IEEE Symposium VLSI Circuits*, June 2004, pp. 2 – 5.

[7] A. Gentile, S. Vitabile, L. Verdoscia, and F. Sorbello, "Image processing chain for digital still cameras based on the SIMPil architecture," in *Proc. IEEE 2005 International Conf. Parallel Processing Workshops*, June 2005, pp. 215 – 222.

[8] J. C. Chen, C. F. Shen, and S. Y. Chien, "CRISP: Coarse-grain reconfigurable image signal processor for digital still cameras," in *Proc. IEEE International Symposium Circuits & Systems*, May 2006, pp. 4379 – 4382.

[9] C. Weerasinghe, W. Li, I. Kharitonenko, M. Nilsson, and S. Twelves, "Novel color processing architecture for digital cameras with CMOS image sensors," *IEEE Trans. Consumer Electronics*, vol. 51, no. 4, pp. 1092 – 1098, Nov. 2005.

[10] E. Y. Lam, "Image restoration in digital photography," *IEEE Trans. Consumer Electronics*, vol. 49, no. 2, pp. 269 – 274, May 2003.

[11] F. Gasparini and R. Schettini, "Color balancing of digital photos using simple image statistics," *Pattern Recognition*, vol. 37, no. 6, pp. 1201-1217, 2004.

[12] K. Barnard, V. Cardei, and B. Funt, "A comparison of computational color constancy algorithms-part I: methodology and experiments with synthesized data," *IEEE Trans. Image Processing,* vol. 11, no. 9, pp. 972-983, Sep. 2002.

[13] K. Barnard, V. Cardei, and B. Funt, "A comparison of computational color constancy algorithms-part II: experiments with image data," *IEEE Trans. Image Processing,* vol. 11, no. 9, pp. 985-996, Sep. 2002.

[14] G. D. Finlayson, S. D. Hordley, and P. M. Hubel, "Color by correlation: a simple, unifying framework for color constancy," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1209-1221, Nov. 2001.

[15] E. Y. Lam, "Combining gray world and Retinex theory for automatic white balance in digital photography," in *Proc. IEEE International Symposium Consumer Electronics*, June 2005, pp. 134 – 139.

[16] C. C. Weng, H. Chen, and C. F. Fuh, "A novel automatic white balance method for digital still cameras," in *Proc. IEEE International Symposium Circuits and Systems*, May 2005, pp. 3801 -3804.

[17] W. C. Kao, L. Y. Chen, and S. H. Wang, "Tone Reproduction in Color Imaging Systems by Histogram Equalization of Macro Edges," *IEEE Trans. Consumer Electronics*, pp. 682-688, vol. 52, no. 2, May 2006.

[18] W. C. Kao, S. H. Chen, T. H. Sun, T. Y. Chiang, and S. Y. Lin, "An integrated software architecture for real-time video and audio recording systems," *IEEE Trans. Consumer Electronics*, vol. 51, no. 3, pp. 879-884, Aug. 2005.

[19] W. C. Kao and Y. J. Chen, "Multistage bilateral noise filtering and edge detection for color image enhancement," *IEEE Trans. Consumer Electronics*, pp. 1346-1350, vol. 51, no. 4, Nov. 2005.

[20] C. S. McCamy, H. Marcus, and J. G. Davidson, "A color-rendition chart." *J. Applied Photographic Eng.*, vol. 2, no. 3, pp. 95-99, 1976.

[21] H. C. Lee, "Digital color image processing method employing constrained correction of color reproduction functions," *U.S. Patent*, 4,663,663, May 5, 1987.

[22] G.D. Finlayson and S.D. Hordley, "Color constancy at a pixel," *Journal of the Opt. Soc. Am. A*, vol. 18, no. 2, pp. 253-264, 2001.

[23] Color image database is available at http://soc.aet.ntnu.edu.tw/ImageData.htm.

Wen-Chung Kao (M'05) received the B.S. degree in electrical engineering from National Central University, Taiwan, R.O.C. in 1990, and the M.S. and Ph.D. degrees in electrical engineering from National Taiwan University, Taiwan, in 1992 and 1996, respectively. From 1996 to 2000, he was a Department Manager at SoC Technology Center, ERSO, ITRI, Taiwan. From 2000 to 2004, he was an Assistant Vice President at NuCam (Foxlink Image Technology) Corporation, Taipei, Taiwan, where he was responsible for leading embedded software team to develop digital still/video cameras. In 2002, he was also invited to form SiPix Technology Inc., Taiwan, where he was in charge of setting up the research team of the company and studying flexible electrophoretic display. In 2004, he became an Assistant Professor at Institute of Applied Electronic Technology and Department of Industrial Education, National Taiwan Normal University, Taipei, Taiwan. He is currently a member of Publications Review Committee of IEEE Transactions on Consumer Electronics. His current research interests include system-on-a-chip (SoC) with embedded software design, digital camera system, homecare system, pattern recognition, and color imaging science.

Sheng-Hong Wang was born in Taipei, Taiwan, on June 10, 1979. He received the M.S. degree in Department of Mechatronic Technology, National Taiwan Normal University, Taipei, Taiwan in 2006. He is currently with Computer & Information Networking Center, National Taiwan University, Taipei, Taiwan. His research interests are in the areas of color image science as well as the design of image processing pipeline and embedded system design.

Lien-Yang Chen was born in Tao-Yuan, Taiwan, on October 6, 1980. He is pursuing the M.S. degree in Institute of Applied Electronic Technology, National Taiwan Normal University, Taipei, Taiwan. His research interests are in the areas of color image processing chip design and embedded software system design.

Sheng-Yuan Lin (M'96) is Vice President of R&D Department at Foxlink Image Technology Corporation, Taipei, Taiwan. He received the B.S. and M.S. degrees in electrical engineering from National Taiwan University, Taiwan, R.O.C., in 1983 and 1987, respectively, and the Ph.D. degree in electrical engineering from University of Pennsylvania, Philadelphia, Pennsylvania, in 1994. He was with OES, ITRI, Taiwan, from 1995 to 1997, working on digital camera system characterization. He joined Foxlink Image Technology, which was named Nucam at the time, in 1997 and now is in charge of digital still camera system integration. His research interest is in optics, electronics and image processing integration. He is also a member of SPIE.