COLOR CORRECTION METHOD USING GRAY GRADIENT BAR FOR MULTI-VIEW CAMERA SYSTEM

Jae-Il Jung and Yo-Sung Ho

School of Information and Mechatronics
Gwangju Institute of Science and Technology (GIST)
1 Oryong-dong Buk-gu, Gwangju, 500-712, Korea
Email: {jijung, hoyo}@gist.ac.kr

ABSTRACT

Due to the different camera properties of the multi-view camera system, the color properties of captured images can be inconsistent. This inconsistency makes post-processing such as depth estimation, view synthesis and compression difficult. In this paper, the method to correct the different color properties of multi-view images is proposed. We utilize a gray gradient bar on a display device to extract the color sensitivity property of the camera and calculate a look-up table based on the sensitivity property. The colors in the target image are converted by mapping technique referring to the look-up table. Proposed algorithm shows the good subjective results and reduces the mean absolute error among the color values of multi-view images by 72% on average in experimental results.

Keywords: color correction, color mismatch problem, multi-view camera, gray gradient bar

1. INTRODUCTION

The technology related to three-dimensional (3-D) information is attracting much attention due to its various applications such as free viewpoint TV (FTV), games, simulations and educational tools. Many researches are going in progress to obtain 3-D information of objects. One of the main researches is the method based on multi-view images. This method is widely used because it can acquire geometrical and color information of objects simultaneously. At the beginning of the research, people utilized a single camera to obtain multi-view images due to the limitation of technology and expensive cost [1]. However, these methods have the grave limitation that it cannot be applied a dynamic scenes. Therefore, researchers have begun to use increased numbers of cameras to extend the technology to dynamic scene [2]. It is called multi-view camera system. The example of this system is shown in Fig. 1. By using multi-view cameras, the multi-view images captured at same time but different viewpoints can be obtained. We can reconstruct 3-D information of objects, after several post-processes such as depth estimation and view synthesis.

Despite its various advantages, the multi-view camera system has two significant problems caused by increased number of cameras. The one is geometrical mismatch problem induced by the misalignment of multiple cameras. The other is the color mismatch problem. The color property of multi-view images depends not only on the reflectance properties of objects but also the properties of the each camera. Hence, even though we capture the same object by the multi-view camera, the color of the object in each image can be different. These variations come from the different property of charge-coupled device (CCD) or complementary metal-oxide-semiconductor (CMOS) in each camera, jitter of shutter speed and aperture, or the variation of angle between objects and camera. These variations disturb the post-process we mentioned already.

Fig. 1: The multi-view camera system which has 10 cameras in parallel array.

Therefore, color correction plays an important role in multi-view image processing fields. Many researchers have made efforts to overcome the color mismatch problem and putted up several results. According to whether pre-process exists or not, these algorithms can be classified into two parts. Most algorithms which have a pre-process make use of the known target such as authorized color chart to compare the color distribution of the multiple images. However, these authorized color chart is very expansive and have only few color samples with a narrow intensity range. Few samples hamper the accuracy of the algorithm. Other part of algorithms is the method without a pre-process. Most algorithms belonged to this part utilize a correspondence of neighboring image to compare color property. However, the performance of the process finding correspondence is not satisfied yet. In some case that the degree of the color distortion is large, we cannot detect an appropriate correspondence. The conventional algorithms for color correction are unable to solve the color mismatch problem effectively.

In this paper, to solve the color mismatch problems effectively, we propose the color correction method using a gray gradient bar. The contribution of this work is that we correct the color distortion on base of color sensitivity property of the camera. The color sensitivity property means the degree of camera's response when a certain intensity of light comes into the camera. These color sensitivity property is obtained by a gray gradient bar and has wide coverage which can cover all pixel values of captured image, because samples are captured in dark room by using a display device. We calculate the look-up table based on the extracted color sensitivity property. Then, the main scene is acquired without any adjusting the setting up of the multi-view cameras. The colors in the target image are converted by mapping technique using the look-up table.

2. Related Knowledge and Work

2.1 Color Mismatch Problem

Multi-view images are captured by several cameras at the same time but different positions. Because of the variance caused by different cameras having dissimilar photoelectric characteristics, significant color mismatch problem is often induced as you can see in Fig. 2. This is the standard sequences (a) *objects* and *(b) race* provided by KDDI Laboratory. Even though cameras are of the same type and we have set their registers to the same values, captured images can be quite different.

This problem is one of the key problems in multi-view image processing field. Because it disturbs the essential post-process such as depth estimation and view synthesis for various applications.

(a)

(b)

Fig. 2: Color mismatch problem of the multi-view images without any correction: (a) *objects* and (b) *race* sequences.

2.2 Color Correction

Previous researches for color correction are classified into two categories: processing images after acquisition with and without pre-processing. In pre-processing, a known target, such as color chart, is usually used to calibrate cameras.

Ilie *et al.* carried out pre-process which captures the color pattern board and corrected images [3]. Joshi *et al.* also used pre-processing and performed correction in channels of red, green, and blue, individually [4]. However, they only consider linear transformation for correction. And, the number and the range of samples are not enough to calibrate the characteristic of a camera.

Fecker *et al.* and Chen *et al.* proposed the method for the compensation of luminance and chrominance variation using histogram matching [5-6]. These methods are not able to cover occlusion area. Therefore, the quality of correction is determined by the occlusion area. Other algorithms without pre-processing utilize a correspondence of a neighboring image to compare color property. In order to find correspondence, Jiang et al. and Yamamoto et al. use Expectation-Maximum algorithm and Gaussian filtering, respectively [7-8]. However, the performance of process finding correspondence is not satisfied yet. In some case that the degree of the color difference is large, we cannot detect appropriate correspondence.

Like this, the conventional algorithms for color correction are unable to solve the color mismatch problem effectively.

3. PROPOSED ALGORITHM

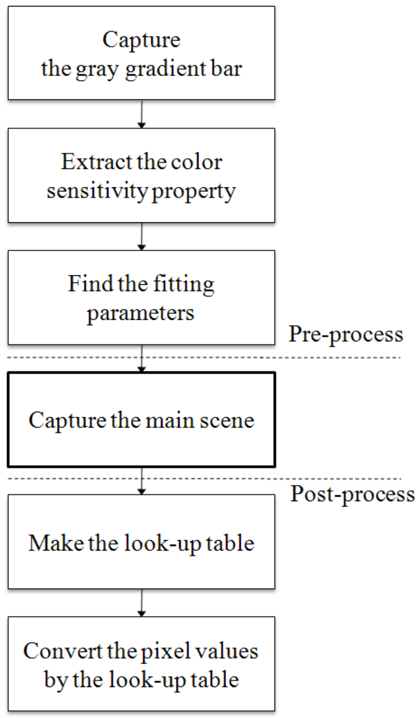


Fig. 3: The flow chart of the proposed algorithm consisting of pre-process and post-process.

To solve the color mismatch problems effectively, we propose the color correction method using a gray gradient bar. Figure 3 outlines the entire procedure of our algorithm. In order to obtain the color sensitivity property of camera, we capture the gray gradient bar on a monitor in a dark room. By using this method, we can obtain information related to the color sensitivity of the camera in a wide intensity range. Camera's sensitivity properties of RGB channels are extracted in Step 2. We fit the extracted data to an appropriate fitting line in Step 3. Then, the multi-view images of the main scene are captured. Among them, the one view which has a natural color distribution is selected as a reference view. Other views are called target views. In Step 5, we make the look-up table for converting process by considering the relation between the camera properties of reference view and target views. Finally, we correct the color of target views by converting pixel values on the basis of the look-up table.

3.1 Capturing Gray Gradient Bar

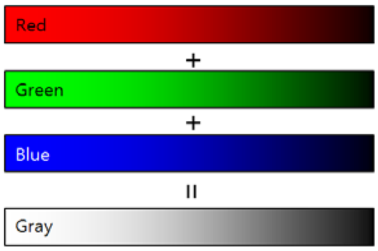


Fig.4: Gray gradient bar consisting of red, green, and blue gradient bars

As we mentioned, we use the gray gradient bar for our algorithm. The gray gradient bar consists of three gradient bars of red, green and blue colors as shown in Fig. 4. By capturing this bar, we can achieve the same effect capturing the bars of three colors individually. We display this bar on a liquid crystal display in a dark room and capture the bar. Similar methods are used to measure the contrast ratio of display devices [9]. This method has various advantages. Many color samples which have information of different intensity values can be obtained. It helps to improve the accuracy of the color correction. And, when capturing in a dark room, we can acquire whole range of intensity from 0 to 255. Captured image of the bar functions as a basis of the color sensitivity properties.

3.2 Extraction of Sensitivity Property

For simple explanation, we show an experimental example of images captured by two neighboring cameras: camera 1 and camera 2. As you can see captured gray gradient bars in Fig. 5, although we captured the same bar, the captured images have the different distribution of the color. It means that the color sensitivity of camera 1 is different from that of camera 2.

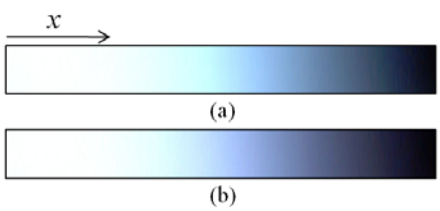


Fig. 5: The pre-captured gray gradient bars by (a) camera 1 and (b) camera 2.

From these captured images of the bar, 256 color samples are extracted with the same interval in accordance with x direction. We calculate average value of 10 pixels located on the perpendicular line of x direction and take this value, for the authenticity of samples. Each sample has 3 values representing the intensity of red, green, and blue.

By using these samples, we obtain the color sensitivity properties of cameras in red, green, and blue channels, respectively. These properties do not need to be absolute. The relative properties are enough to be a basis of the look-up table. Figure 6 shows the color sensitivity property of each camera. As the position of x becomes more distant from the starting point, the intensity of each color is decreased together, due to the feature of the gradient bar. However, the tendency of the diminution is different according to cameras. In this example, the difference is more prominent in green and blue channels. These differences induce the color inconsistency of output images.

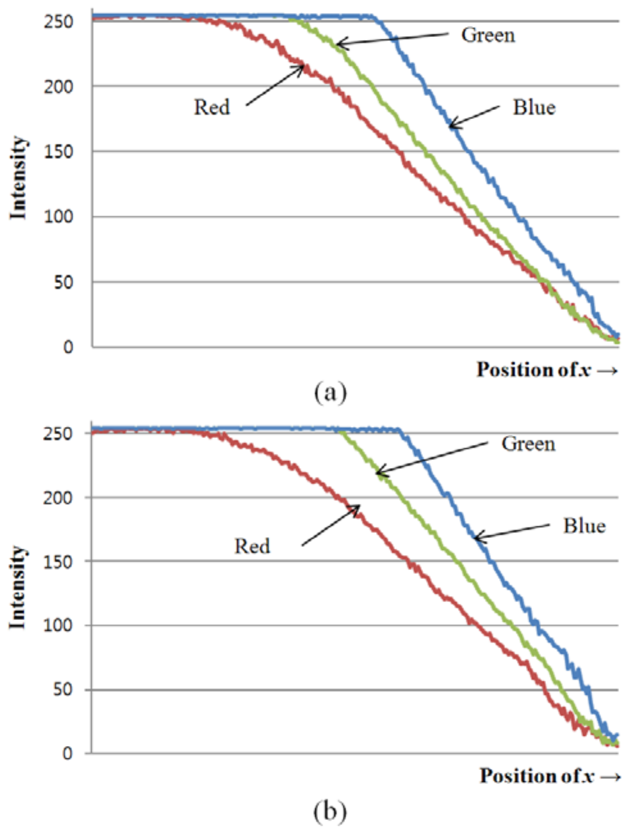


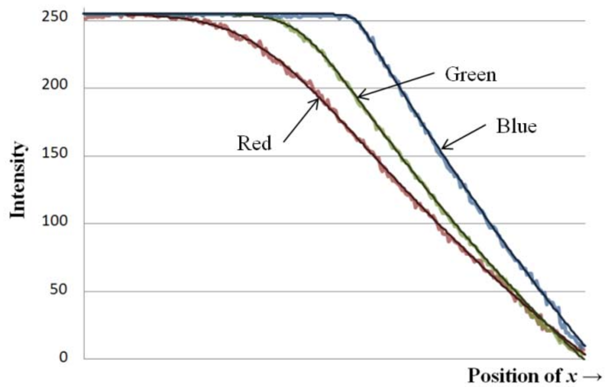
Fig. 6: The color sensitivity property of (a) camera 1 and (b) camera 2.

3.3 Fitting Extracted Data

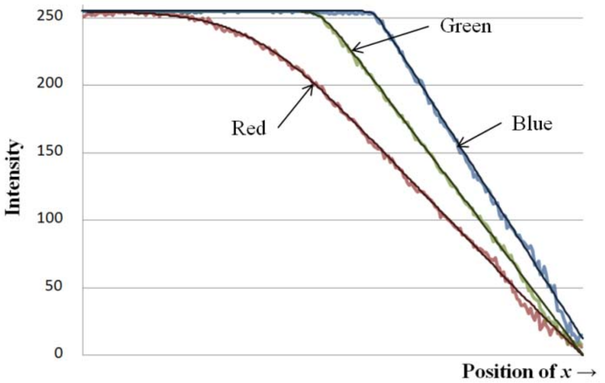
As you can see in Fig. 6, sampled data have fluctuation because of noise in the image. This fluctuation induces decrease of look-up table's accuracy. In order to reduce fluctuation of data, we fit extracted data to a sigmoid line having an "S" shape. The sigmoid line is appropriate to reflect the shape of the sensitivity property and the area of saturation of the color sensitivity data and can be represented as follow:

$$Data_f = a + \frac{b}{\left[1 + e^{-\left(\frac{Data_o - c}{d}\right)}\right]^e} \quad (1)$$

where $Data_o$ and $Data_f$ are original and fitted data, respectively. The small letters represent fitting parameters. We find fitting parameters to minimize the error between $Data_o$ and $Data_f$.



(a)



(b)

Fig. 7: The fitting result of the color sensitivity properties of (a) camera 1 and (b) camera 2.

Figure 7 represents the fitting result of color sensitivity data. Light color lines and deep color lines represent original data and fitting line, respectively. After fitting, the fluctuation of data is removed and only the sensitivity property of camera is remained. It means that the sigmoid fitting method is appropriate to fitting the color sensitivity data. We carry out the fitting process to whole extracted data and use fitted values when making a look-up table.

3.4 Capturing Main Scene

After data fitting, we capture the main scene without any change of the camera setting up. It is not necessary to maintain the condition of a dark room. This condition is only required when we capture the gray gradient bar to obtain samples of the wide range.

Figure 8 shows the captured images of the main scene. Because the sensitivity of blue of camera 2 is lower than that of camera 1, the image captured by camera 2 is reddish. To correct the color distribution of image captured by camera 2 on the basis of the color sensitivity property of camera 1, we select the camera 1 and camera 2 as a reference view and a target view, respectively.

(a)

(b)

Fig. 8: The original images captured by (a) camera 1 and (b) camera 2.

3.5 Look-up Table and Conversion

For the converting process of RGB channels, we make the 3 look-up tables which have information of the relation between properties of the target and the reference views. The example of converting process is shown in Fig. 9. The value in the figure is RGB values for one pixel. As you can see, target image has different value of red as compared with reference image. The look-up table has information that the red value of 200 is matched to that of 100. Like this, pixel value of target image is converted referring to information of the look-up table. This process is carried out to pixels in red, green, and blue channels, individually.

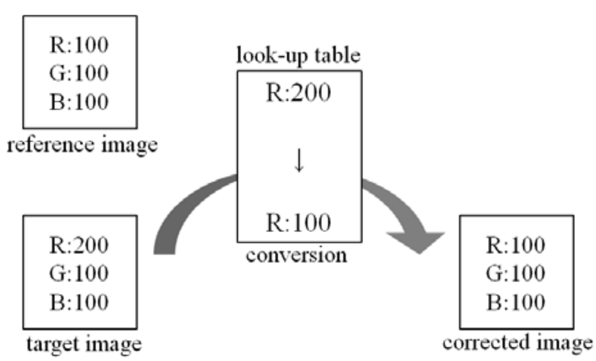


Fig. 9: The converting process using information of the look-up table.

After the converting process is applied to all pixels in target image, we obtain the corrected image having similar color distribution to reference view. The corrected target image is shown in Fig 10.

Fig. 10: The corrected image

From Fig. 10, we can subjectively recognize that the color distribution of image captured by camera 2 becomes similar to that by camera 1. To verify the corrected color sensitivity property of camera 2, we apply proposed algorithm to the image of a gray gradient bar of camera 2 and re-extract the color sensitivity property. The result is shown in Fig. 11. From this result, we can know that the distortion in green and blue channel is compensated by our algorithm.

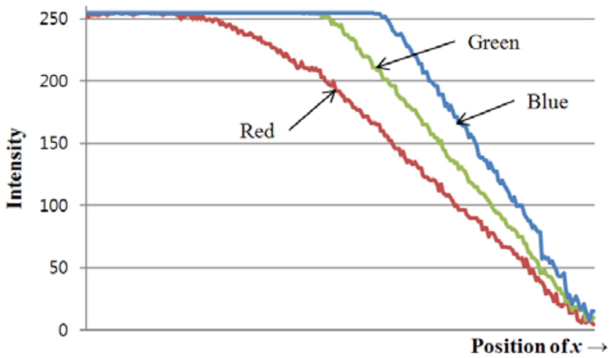


Fig. 11: The corrected color sensitivity property of camera 1.

To check similarity between the color sensitivity properties of camera 1 and corrected camera 2, the mean absolute error (MAE) is calculated on base of fitting results of each color sensitivity property by:

$$MAE\_sensitivity = \sum_{i=1}^{256} \left| d_{cam1,i} - d_{cam2,i} \right| \quad (2)$$

where d_{cam1} and d_{cam2} are sensitivity data of camera 1 and camera 2. i represents the number of sample. The result is shown in Fig. 12. A light gray bar and a deep gray bar represent the MAE value of color sensitivity properties before and after correction, respectively. This result means that the color sensitivity properties of camera 1 and camera 2 become similar after correcting process. This algorithm can be extended to multi-view camera system. The detail and extended experimental results to multi-view image are in next section.

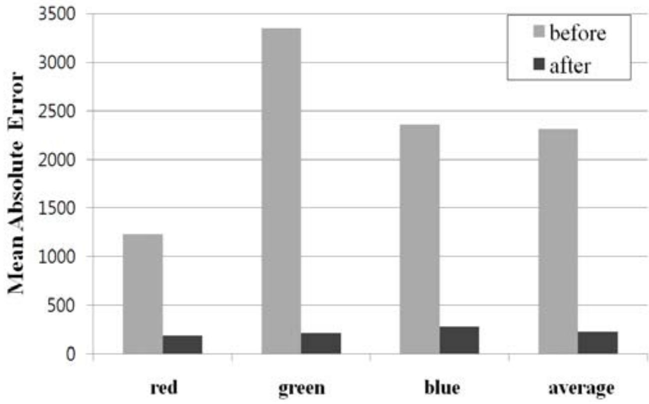
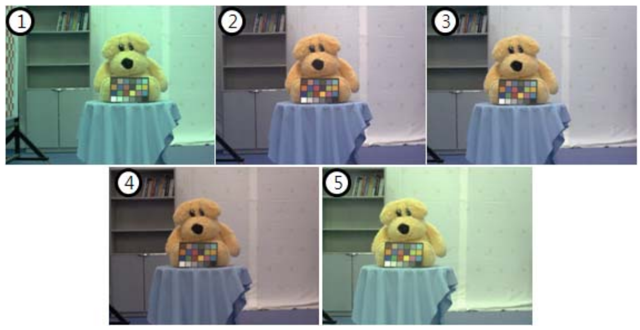


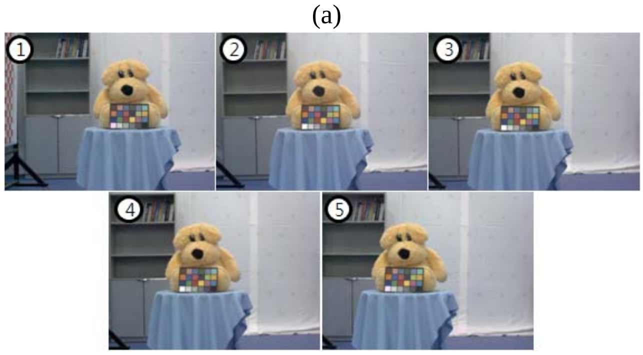
Fig. 12: Comparison of MAE between color sensitivity properties: before and after correction.

4. Experimental Results

In order to evaluate the performance of the proposed algorithm, we used 5 multi-view images captured by 1-D parallel multi view camera system. This system consists of 5 Point Grey Research Flea IEEE-1394 cameras with 1/3-inch Sony CCD. We captured gray gradient bar by 5 cameras, individually. Then, we captured the main scene which contains a known target (a 24-sample *GretagMacbeth* [10] *ColorChecker^{TM}*) as shown in Fig. 13 (a) for objective measurement. To check that our algorithm can operate in severely distorted image, we roughly adjust camera setting up of camera 1 and camera 5. The image captured by camera 3 is selected as a reference image. After applying our algorithm to these images, the corrected images shown in Fig. 13 (b) are obtained. With the naked



(a)



(b)

Fig. 13: Multi-view images captured by 5 cameras: (a) before correction and (b) after correction.

eyes, we can infer that the distorted color distributions of target views are corrected well.

For objective measurement, we extract the 24 color values of Macbeth chart from every image and calculate MAE value comparing with reference view. In order to obtain high accuracy, 100 pixels in one color patch are extracted and averaged to be used as the representative value. We define the formula for MAE:

$$MAE\_Macbeth = \frac{1}{24}\left(\sum_{i=1}^{24}\left|d_{di} - d_{ri}\right|\right) \tag{3}$$

where i is the number of sample in Macbeth chart. d_d and d_r are extracted values of the ptarget view and the reference view, respectively. The results are shown in Table 1 with the exception of camera 3, since camera 3 is reference camera. The ratio is calculated by Eq. (4).

$$Ratio(\%) = \frac{MAE_{after}}{MAE_{before}} \times 100 \tag{4}$$

The proposed algorithm achieved about 72% quality improvement compared to the original images. As you can see in Fig. 14, the MAE values of corrected images are evenly distributed. It means that the proposed algorithm successfully corrects the color mismatch problem.

Table 1: The result of proposed algorithm. The values in the table are MAE.

camera	correction	red	green	blue	average	ratio (%)
1	before	19.04	21.08	4.17	14.76	26.53
	after	4.58	3.92	3.25	3.92	
2	before	3.29	9.38	4.92	5.86	50.95
	after	3.75	1.92	3.29	2.99	
4	before	17.29	27.42	27.42	24.04	11.90
	after	3.17	2.67	2.75	2.86	
5	before	7.75	26.08	4.00	12.61	20.59
	after	2.71	2.42	2.67	2.60	

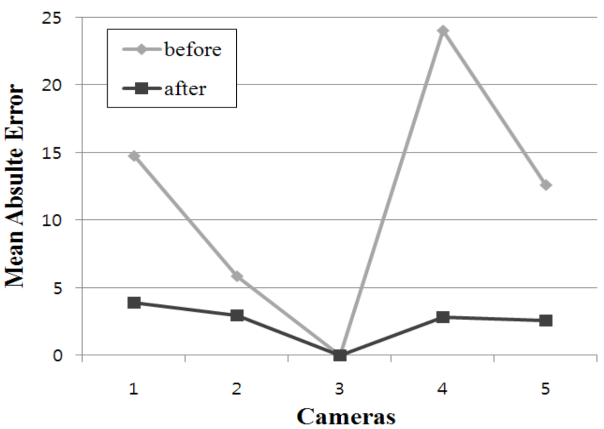


Fig. 14: Comparison of the color distribution of multi-view images: before and after correction.

5. CONCLUSIONS

In this paper, we have proposed a color correction algorithm for multi-view images. We captured a gray gradient bar displayed on a display device in a dark room. This method can help to acquire abundant samples having sensitivity information of a wide range. On the basis of these samples, the color sensitivity property of the camera is extracted. Then, we made the look-up table by considering the relation of color sensitivity properties of each camera and converted values of target view. In experimental results, proposed algorithm shows the good subjective results and reduces the mean absolute error among the color values of multi-view images by 72% on average.

ACKNOWLEDGEMENTS

This work was supported in part by ITRC through RBRC at GIST (IITA-2008-C1090-0801-0017).

REFERENCES

[1] M. Levoy and P. Hanrahan, “Light field rendering,” SIGGRAPH 1996, pp. 33-42, Aug. 1996.

[2] A. Majumder, W. Seales, M. Gopi, and H. Fuchs, “Immersive teleconferencing: A new algorithm to generate seamless panoramic video imagery,” the Seventh ACM International Conference on Multimedia, pp 169-178, Oct. 1999.

[3] A. Ilie and G. Welch, “Ensuring color consistency across multiple cameras,” IEEE international Conference on Computer Vision, pp. II: 1268-1275, Oct. 2005.

[4] N. Joshi, B. Wilburn, V. Vaish, M. Levoy, and M. Horowitz, “Automatic color calibration for large camera arrays,” in UCSD CSE Tech. Rep. CS2005-0821, May 2005.

[5] U. Fecker, M. Barkowsky, and A. Kaup, “Improving the Prediction Efficiency for Multi-View Video Coding Using Histogram Matching,” Picture Coding Symposium, pp. 2-16, April 2006.

[6] Y. Chen, J. Chen, and C. Cai, “Luminance and chrominance correction for multi-view video using simplified color error model,” Picture Coding Symposium, pp. 2-17, April 2006.

[7] G. Jiang, F. Shao, M. Yu, K. Chen, and X. Chen, “New Color Correction Approach to Multi-view Images with Region Correspondence,” Lecture Notes in Computer Science, Vol. 4113, pp.1224-1228, Aug. 2006.

[8] K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, M. Tanimoto, S. Shimizu, K. Kamikura, and Y. Yashima, “Multiview Video Coding Using View Interpolation and Color Correction,” IEEE Transactions on Circuits and Systems for Video, Vol.17, No.11, pp. 1436-1449, Nov. 2007.

[9] P. A. Boynton and E.F. Kelley, “Measuring the contrast ratio of displays,” Information Display, Vol. 11, pp. 24-27, Sep. 1996.

[10] GretagMacbeth Color Management Solutions, http://www.gretagmacbeth.com.