Pixel-wise Illumination Correction Algorithms for Relative Color Constancy Under the Spectral Domain

Yunfeng Zhao

Queen's University Belfast
Belfast, Northern Ireland
y.zhao@qub.ac.uk

Chris Elliott

Queen's University Belfast
Belfast, Northern Ireland
chris.elliott@qub.ac.uk

Huiyu Zhou

University of Leicester
Leicester, England
hz143@leicester.ac.uk

Karen Rafferty

Queen's University Belfast
Belfast, Northern Ireland
k.rafferty@qub.ac.uk

Abstract—Achieving color constancy between and within images, i.e., minimizing the color difference between the same object imaged under nonuniform and varied illuminations is crucial for computer vision tasks such as colorimetric analysis and object recognition. Most current methods attempt to solve this by illumination correction on perceptual color spaces. In this paper, we proposed two pixel-wise algorithms to achieve relative color constancy by working under the spectral domain. That is, the proposed algorithms map each pixel to the reflectance ratio of objects appeared in the scene and perform illumination correction in this spectral domain. Also, we proposed a camera calibration technique that calculates the characteristics of a camera without the need of a standard reference. We show that both of the proposed algorithms achieved the best performance on nonuniform illumination correction and relative illumination matching respectively compared to the benchmarked algorithms.

Keywords—color constancy, algorithm, illumination correction, camera calibration.

I. INTRODUCTION

The ability to determine color of objects independent of the illumination source is known as color constancy. Applications of color constancy range from comparison of scene objects, e.g., colorimetric analysis, to object recognition [1]. Color constancy tries to ensure a minimum "color distortion" within and between images. The distortion of the color is mainly produced by the varied characteristics of illuminant and imaging systems.

White balancing is the most commonly used color constancy technique. Initially, the color of the illumination is estimated [2, 3]. Then, a 3×3 matrix is computed and applied to the image to correct any effect due to scene illumination, which is called the gamut-constraint method [4, 5]. Even though white balancing was originally designed to only ensure the correction of neutral colors, the work proposed by Cheng et al. has indicated that a diagonal 3×3 matrix is efficient to perform full color correction under broadband illumination [6]. However, existing techniques can only find ground truth color under camera-specific color spaces. The same color surface imaged by different cameras can hardly be mapped to the same pixel values.

Unlike the proceeding methods that mostly assume a single and uniform illumination, the retinex theory proposed by Land and McCann concluded that edges are important information for color constancy [7]. They suggest a random walk algorithm
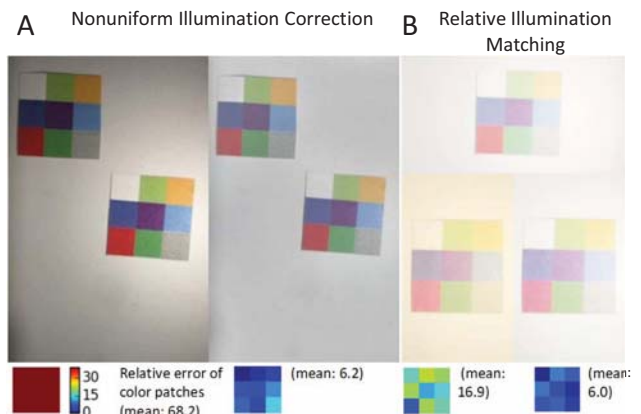

Fig. 1. (A) The image on the left: the input image taken under nonuniform illumination. The image on the right: the corrected image using the proposed spectral nonuniform illumination correction algorithm. (B) The image on the top: the image to be matched. The image at the bottom left: the original image to be corrected. The image at the bottom right: the corrected image using the proposed spectral illumination matching algorithm.

to normalize nonuniform illumination of an image based on the edge information. The major drawback of this method is the requirement of using a threshold to identify an edge from a smooth transition [8].

As standardized in ISO 17321-1:2012 [9], there are mainly two methods for camera color calibration: spectral method and target method. The spectral method measures image sensor's response in a continuous range of wavelength and requires elaborate laboratory equipment. The target method calibrates the camera by measuring the color patches on a standard color target, e.g., the Macbeth color chart. Both of these methods require standard color reference and are unpractical for normal end-users.

In this paper, we present the concept of relative color constancy, i.e., the ability to align colors of same objects between images independent of the illumination source and image sensor. Unlike color constancy, relative color constancy does not focus on mapping the color to its ground truth value. Instead, it targets on minimizing the color difference of the same object imaged under varied illumination condition and by different camera. We present the following contributions to promote the relative color constancy:

1. For the first time, two pixel-wise spectral transformation algorithms were proposed. These algorithms transform an image from perceptual color space to spectral domain to achieve relative color constancy under constrained conditions.
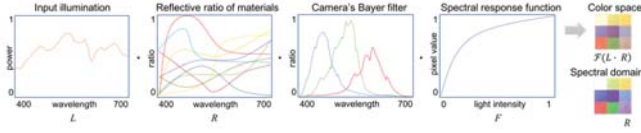
Fig. 2. Illustration of the RGB imaging process and the difference of produced image in perceptual color space and spectral domain.

The first algorithm corrects the nonuniform distribution of illumination, coined the Spectral Nonuniform Illumination Correction (SNIC) algorithm. The second algorithm aligns the illumination between images by comparing the colors of a common material that appears in both of the images, coined the Spectral Illumination Matching (SIM) algorithm.

2. Unlike most existing camera color calibration techniques that require standard color references, a new unsupervised calibration technique was proposed in this paper to calculate the characteristics of a camera without the need of any standard color reference. This technique works by finding the balance point between color difference of the same material across images and the ability of distinguishing colors in a single image.

3. Two datasets were created for both camera color calibration and evaluation of relative color constancy. To our best knowledge, the dataset consisting of images with nonuniform illumination is the first of its kind.

II. PROPOSED METHOD

The prior knowledge in image formation indicates that the light energy received by a camera is the light energy emitted from the light source subtracting that absorbed by the material in the imaged scene [8]. The Bayer filter on the image sensor separates the image sensor's sensitivity of light intensity into three regions: low, middle, and high frequency. These regions correspond to the red, green, and blue channels of the RGB imaging. Based on Abney's law [10], the total luminance of light composed of several wavelengths is equal to the sum of the luminances of its monochromatic components. Thus, the total light energy received by each color channel can be integrated by the light energy of each wavelength within the corresponding frequency region. Finally, the light intensity received by RGB receptors in the image sensor is nonlinearly mapped to the output pixel values [11]. This process is illustrated in Fig. 2.

Based on this knowledge, the mathematical modelling of RGB image formation, D, at pixel x can be represented as:

$$D(x) = \int_\omega \mathcal{F}_c[L(\lambda, x) \cdot R(\lambda, x)] \, d\lambda \qquad (1)$$

where λ denotes the wavelength of incident light, ω is the visible spectrum ranging from $400 - 700$ nm, $L(\lambda, x)$ is the illumination intensity received by the camera at wavelength λ and pixel x, $R_c(\lambda, x)$ is the reflective ratio of color surface at wavelength λ and pixel x, and $\mathcal{F}_c$ is the camera's spectral response function (CSRF), and c is the color channel r, g, b.

Equation (1) can be simplified for each color channel:

$$D = \mathcal{F}(L \cdot R) \qquad (2)$$

Two useful transformations of (2) for the algorithms in the following sections are:

$$R = \frac{\mathcal{F}^{-1}(D)}{L} \qquad (3)$$

$$L = \frac{\mathcal{F}^{-1}(D)}{R} \qquad (4)$$

As mentioned in [11], the CSRF can be modelled as:

$$D = \mathcal{F}(I) = a \cdot \log_{10}\left(\frac{I}{b}\right) \qquad (5)$$

where I is the input light intensity and a, b, γ are camera-specific constants.

The gamma correction is a common post-production procedure in digital cameras. It can be represented as:

$$D_{out} = D_{in}{}^\gamma \qquad (6)$$

where D_{in} and D_{out} are the input and output pixel values respectively, and γ is the gamma correction constant.

After considering the gamma correction, the CSRF model can be improved to:

$$D = \mathcal{F}(I) = \left[a \cdot \log_{10}\left(\frac{I}{b}\right)\right]^\gamma \qquad (7)$$

The inversed CSRF can be inferred as:

$$I = \mathcal{F}^{-1}(D) = b \cdot 10^{\frac{D^{1/\gamma}}{a}} \qquad (8)$$

A. The Nonuniform Illumination Correction Algorithm

We assume that there is an achromatic and Lambertian background surface where its reflective ratio is known. The background image under the same illumination condition as the image to be corrected is also provided.

It is not too hard to deduce the following equation from (3), (4), and (8) based on the assumption that illumination on the test image, L_T, is the same as that on the background image, L_B, at the same position:

$$R_T = \frac{\mathcal{F}^{-1}(D_T)}{L_T} = \frac{\mathcal{F}^{-1}(D_T)}{\frac{\mathcal{F}^{-1}(D_B)}{R_B}} = R_B \cdot 10^{\left(\frac{D_T^{1/\gamma_1}}{a_1} - \frac{D_B^{1/\gamma_2}}{a_2}\right)} \qquad (9)$$

where a_1, γ_1 and a_2, γ_2 are the constants of the first and second cameras, R_B is the reflective ratio of the background surface, D_T is the pixel value of the image to be corrected, and D_B is the pixel value of the background image. R_T is the reflective ratio of the material. The color of the material under the canonical illumination can be considered as multiplying R_T by a unit vector which can be eliminated. Therefore, R_T is also the color of the material under the canonical illumination and is illumination and camera irrelevant.

Based on (9), the SNIC algorithm is provided as in Algorithm 1:

Algorithm 1 Spectral nonuniform illumination correction algorithm
---
1: image1 ← The image to be corrected
2: image2 ← The background image
3: result ← The result image
4: a_1 ← The constant a of camera 1
5: a_2 ← The constant a of camera 2
6: γ_1 ← The constant γ of camera 1
7: γ_2 ← The constant γ of camera 2
8: R_B ← The reflective ratio of the background material
9:
10: **for** every pixel p of image1 **do**
11: **for** every color channel c of image1 **do**
12:    D_T ← pixel value of the image to be corrected at p and c
13:    D_B ← pixel value of the background image at p and c
14:    $R_T \leftarrow R_B \cdot 10^{\left(\frac{D_T^{1/\gamma_1}}{a_1} - \frac{D_B^{1/\gamma_2}}{a_2}\right)}$
15:    result ← set R_T to the result image at p and c

B. The Relative Illumination Matching Algorithm

We assume the illumination is broadband and uniform on the Lambertian surface to be imaged. There is a common material existing in both of the images to be illumination matched. The illumination in one image can be aligned to the other by reversely applying the illumination difference of the common material to the entire image:

$$\Delta L_C = \Delta L_T \tag{10}$$

$$L_{C1} - L_{C2} = L_{T1} - L_{T2} \tag{11}$$

where L_{C1} and L_{C2} are the light intensities of the common material in two images, and L_{T1} and L_{T2} are the light intensities of the pixels to be corrected.

After combing (11), (4), and (8), we can extend the equation as:

$$\frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{R_C} = \frac{10^{\frac{D_{T1}^{1/\gamma_2}}{a_2}} - 10^{\frac{D_{T2}^{1/\gamma_2}}{a_2}}}{R_T} \tag{12}$$

We attempt to match illumination of image 1 to that of image 2. If there is prior knowledge on the reflective ratio of the materials appearing in the two images, we can substitute them for R_C and R_T. However, it is mostly unpractical in real scenarios. We instead consider the pixel values of the first image after correction as the true reflective ratio of the materials. That is, $R_C = D_{C1}$ and $R_T = D_{T1}$. Then, we can have:

$$\frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{D_{C1}} = \frac{10^{\frac{D_{T1}^{1/\gamma_2}}{a_2}} - 10^{\frac{D_{T2}^{1/\gamma_2}}{a_2}}}{D_{T1}} \tag{13}$$

where a_1, γ_1 and a_2, γ_2 are the constants of the camera 1 and camera 2. If the two images are taken by a single camera, a_1 will equal a_2. Similarly, γ_1 will equal γ_2. D_{T1} is the pixel value of image 1 after illumination matching. D_{T2} is the pixel value of image 1 before illumination matching. And D_{C1} and D_{C2} are pixel values of the common material in image 1 and 2 respectively. Equation (13) is named as the Ideal Spectral Illumination Matching (iSIM) algorithm. However, there is no straight forward way to directly solve (13) in practice. Trying to solve this using traditional minimization methods will only result in large computational cost. Hence, we developed the Fast Spectral Illumination Matching (fSIM) algorithm to minimize the computational cost. That is, instead of assuming $R_T = D_{T1}$ as in the iSIM, we assume that $R_T = D_{T2}$ and let D_{T1} converge to R_C through iterations:

$$\frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{D_{C1}} = \frac{10^{\frac{D_{T1}^{1/\gamma_2}}{a_2}} - 10^{\frac{D_{T2}^{1/\gamma_2}}{a_2}}}{D_{T2}} \tag{14}$$

in this way, the number of iterations needed for calculating D_{T1} is reduced to around 3 to reach an error less than 1 out of 255 in pixel value.

Equation (14) can be transformed to:

$$D_{T1} = \left[a_2 \cdot \log_{10} \left(D_{T2} \cdot \frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{D_{C1}} + 10^{\frac{D_{T2}^{1/\gamma_2}}{a_2}} \right) \right]^{\gamma_2} \tag{15}$$

Based on (15), the fSIM algorithm can be described as in Algorithm 2.

Algorithm 2 Fast spectral illumination matching algorithm

1: image1 ← The image to be corrected
2: result ← The corrected image
3: a_1 ← The constant a of camera 1
4: a_2 ← The constant a of camera 2
5: γ_1 ← The constant γ of camera 1
6: γ_2 ← The constant γ of camera 2
7: D_{C1} ← The pixel value of the common material in image 2
8: D_{C2} ← The pixel value of the common material in image 1
9: N ← The number of iterations
10:
11: **for** every pixel p of image1 **do**
12:    **for** every color channel c of image1 **do**
13:       **for** i = 1 **to** N **do**
14:          D_{T2} ← pixel value of image1 at p and c
15:          $D_{T1} \leftarrow \left[a_2 \cdot \log_{10} \left(D_{T2} \cdot \frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{D_{C1}} + 10^{\frac{D_{T2}^{1/\gamma_2}}{a_2}} \right) \right]^{\gamma_2}$
16:          $D_{C1} \leftarrow \left[a_2 \cdot \log_{10} \left(D_{C2} \cdot \frac{10^{\frac{D_{C1}^{1/\gamma_1}}{a_1}} - 10^{\frac{D_{C2}^{1/\gamma_1}}{a_1}}}{D_{C1}} + 10^{\frac{D_{C2}^{1/\gamma_2}}{a_2}} \right) \right]^{\gamma_2}$
17:       result ← set D_{T1} to the result image at p and c

C. Camera Calibration

Camera calibration is the process of calculating the camera-specific constants a and γ for the SNIC and SIM algorithms. Unlike most camera calibration techniques, this method does not require a standard reference color. Instead, it assumes that there are shared color objects in all the training images with a varied range of color in the scene.

The proposed camera color calibration technique was based on the fact that there are mainly two criteria to be considered for the performance of a relative color constancy correction: 1) The color of the same material that appears in multiple images should be as close as possible. 2) The colors in a single image should be as distinguishable as possible. However, an overly enlarged camera constant a leads to an over exposed image which will result in a better compliance for the first criteria yet worse for the second one. On the other hand, decreasing a will saturate the image which promotes the second criteria yet downgrades the first. Therefore, a minimax game needs to be played to find the balance between the first and second criteria.

The proposed method calculates the optimal camera constants a and γ by minimizing the color difference of the same material, V, across different images, and maximizing the distinction of the different colors, S, in each single image. Mathematically, it can be represented as $\min\limits_{V} \max\limits_{S} \mathcal{J}(V,S) = V + \lambda S^{-1}$ and modelled as the problem of finding the optimal camera constants a and γ by minimizing the following cost function:

$$a, \gamma = \operatorname*{argmin}_{a,\gamma} \left(\left\| \frac{\sum_{j=1}^{m} \left(\{D_j\}_3 - \overline{\{D\}_3} \right)}{m}, \lambda \left(\frac{\sum_{i=1}^{n} \left(\{D_i\}_3 - \overline{\{D\}_3} \right)}{n} \right)^{-1} \right\| \right) \tag{16}$$

where m is the number of images, n is the number of colors, D is the fSIM formula as (15), and N is the number of iterations for fSIM. λ is the weight controlling the balance. ‖ denotes the L2-norm.

D. Dataset

A

B

Fig. 3. (A) Two example images in the first dataset. Two identical 9-patch color charts appear in different positions in every image. The illumination is nonuniform with varied colors. (B) Examples of images in the second dataset taken under dynamic illumination conditions. A uniform illuminated color chart appears in every image in the dataset.

Two datasets were created for camera calibration and for benchmark analysis of relative color constancy algorithms. Both the datasets were taken using an iPhone 6 Plus. The first was taken with the automatic white balance turned off while the second with default camera settings. The color charts imaged in both datasets were color printed using a WorkCentre 7845 (Xerox, U.S.) printer. Since the relative color constancy does not focus on correcting the color to its ground truth value, the ground truth value of the color patches that appeared in these two datasets are unknown. Both datasets were carefully labelled so that the same color patch in different images can be paired and compared accurately.

The first dataset consists of 106 nonuniform illuminated images under varied illumination colors. The distribution of the illumination is mostly radiational. Four different randomly generated 9-patch color charts were imaged in the dataset. Two identical color charts were randomly located in every image. A 3D-printed lightbox attached to the smartphone was used to fully block the ambient light. A fiber optical cable was used to pipe the smartphone rear flashlight into the lightbox and generate the radiational distributed illumination. This dataset is useful to evaluate the performance of nonuniform illumination correction algorithms. Two example images in the first dataset are shown in Fig. 3A.

The second dataset was constructed by 28 photos of a uniform illuminated color chart under different illumination conditions. The 24-patch color charts seen in all the photos have almost identical spectral characteristics. Fig. 3B shows some of the example images in the second dataset.

III. EXPERIMENT

Before the experiments, the characteristics of the iPhone 6 Plus rear camera were measured using the proposed camera calibration technique and the second dataset. Tensorflow [12] was chosen as the platform for performing the machine learning. The cost function (16) was processed for the optimal parameter a using the Adam Optimizer [13] with γ being set to 2.4 and λ to 10^3. We manually pick the optimal a based on the training output for later experiments.

To quantitatively estimate the performance of relative color constancy, the root mean square error (RMSE) and angular error were used to compare color values. The RMSE between two color charts can be represented as:

$$Err_{RMS} = \sum_{i=1}^{n} \sqrt{(\rho_i^T - \rho_i^C)^2} \qquad (14)$$

whilst the angular error between two color charts can be represented as:

$$Err_{angular} = \sum_{i=1}^{n} \cos^{-1}\left(\frac{\rho_i^T \cdot \rho_i^C}{\|\rho_i^T\|\|\rho_i^C\|}\right) \qquad (15)$$

where ρ^T and ρ^C are the compared color vectors, n is the number of color patches in the compared color chart. The RMSE measures the absolute distance between two color vectors while the angular error measures the difference between color values irrelevant of their magnitude.

Two experiments were conducted to demonstrate the effectiveness of the SNIC and fSIM algorithms. The first dataset was used for the first experiment to evaluate the performance of nonuniform illumination correction algorithms while the second dataset was used for the second experiment to verify the performance of relative illumination matching algorithms.

The first experiment was designed to benchmark the performance of the SNIC with the retinex algorithm [7] and naïve algorithm, i.e., averaging the reversed lightness channel of the background image, L_b, with the lightness channel of the test image, L_s, under LAB color space. Mathematically, it can be represented as $L_r = \frac{1}{2} \cdot (255 - L_b + L_s)$. The background images needed for the SNIC algorithm were generated directly from the images of the first dataset using the morphology open and close operations and Gaussian blur. A cross shape structure element with a size of 91 was used for the morphology operations, and a kernel size of 41 was used for the Gaussian blur. After performing the nonuniform illumination correction, the average RMSE and angular error between the two compared 9-patch color charts in a single image were evaluated.

In the second experiment, the performance of fSIM was compared with the gray world (GW) algorithm [14] on relative illumination matching. The GW algorithm is a commonly used benchmark algorithm for color constancy [3, 15]. A reversible handshake comparison strategy was used. That is, every image in the second dataset was bidirectionally matched and evaluated with every other image in the dataset without a duplication. Note that image A matching to image B is different from matching image B to image A. Therefore, there were in total 756 comparisons evaluated in this experiment using the 28 images in the second dataset. The white color
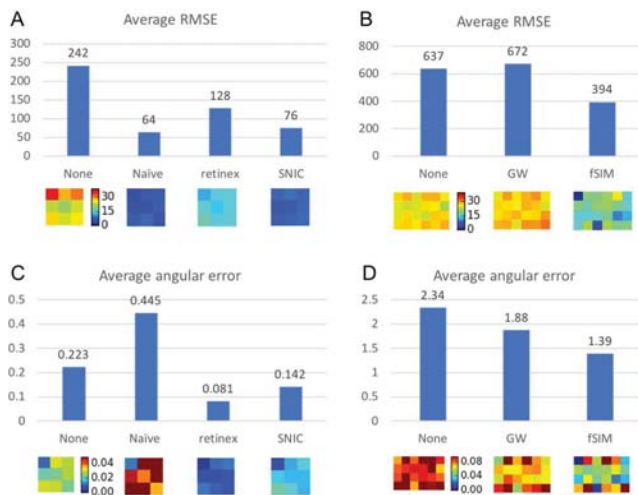
Fig. 4. This figure shows comparison results of the first experiment (first column) and second experiment (second column) in terms of average RMSE and angular error. The error distribution of color comparison between color charts is visualized in heatmap beneath each method applied. (A) Average RMSE between compared images using the original images and nonuniform illumination correction algorithms. (B) Average RMSE between compared images using the original images and relative color matching algorithms. (C) Average angular error of the comparisons as in (A). (D) Average angular error of the comparisons as in (B).

patch was used as the common material across images for illumination matching. The average RMSE and angular error of the two compared 24-patch charts appearing under two different illumination conditions were analyzed. To test the gray world algorithm, the algorithm was applied on both the image to be matched and the image to be corrected.

IV. RESULT AND DISCUSSION

The results of the first and second experiments are illustrated in Fig. 4. Overall, the results have shown that the proposed SNIC algorithm has achieved the second lowest average RMSE and angular error in the first experiment. On the other hand, the proposed fSIM algorithm has clearly outperformed the gray world algorithm in terms of both average RMSE and angular error in the second experiment.

Since there was no single algorithm that outperformed the others in terms of both average RMSE and angular error in the first experiment, it is necessary to take a closer analysis of the experimental results. In Fig. 4A, the naïve algorithm has output the best result in terms of average RMSE. However, the average angular error produced by this method has indicated the worst performance. This can be explained by visualizing the corrected image from the naïve algorithm in Fig. 5. The corrected image tended to be darker and less saturated than the images corrected using the other methods. And a less saturated image would surely provide a lower RMSE. Therefore, the naïve algorithm did not provide the optimal correction performance in the first experiment. In Fig. 4C, the retinex algorithm has produced the lowest angular error value. Yet, it was only ranked in third place as evaluated by using the average RMSE. Again, we explain this by visualizing the image output in Fig. 5. The nonuniform illumination was not

perfectly corrected using the retinex algorithm. The illumination at the center of the image is visually and distinguishably brighter than that at the corner. Since the angular error does not reflect difference in brightness, the retinex algorithm gained the lowest annular error value. However, the retinex algorithm did not outperform the SNIC as can be confirmed visually in Fig. 5.

Based on the results indicated in Fig. 4 and preceding analyses, we draw the conclusion that the proposed SNIC and fSIM algorithms achieved the best nonuniform illumination correction and relative illumination matching performance respectively compared to the other tested algorithms in the first and second experiments.

Thanks to the prior knowledge of reflective ratio of the background material provided to the SNIC algorithm, the illumination normalized image processed by the SNIC visually stands out because of its high contrast. This advantage will greatly benefit computer vision applications, e.g., object recognition. Based on [6] and our observation, the fSIM algorithm can produce the best result when bright neutral color patch and broadband illumination appear in the imaged scene.

In future research, the performance of the SNIC and SIM algorithms can be further improved by more accurate mathematical modeling of the camera's spectral response function, e.g., considering the tone mapping procedure in imaging system. Also, further improvement in the camera calibration technique will enable the handling of nonuniform illumination and calibration of the γ parameter.

V. CONCLUSION

To achieve relative color constancy, i.e., minimizing the color difference of same objects taken by different cameras under varied illuminations, this paper has proposed two illumination correction algorithms that work under the spectral domain: a nonuniform illumination correction algorithm named SNIC and a relative illumination matching algorithm named SIM. A camera calibration technique that does not require a standard color reference has also been proposed to serve for these two proposed algorithms. Two image datasets taken by a smartphone were created for benchmarking of relative color constancy algorithms. Through our experiments, we have demonstrated that the proposed algorithms produced the best overall performance compared to the other tested algorithms.

The proposed algorithms work under the spectral domain. Hence, they can be downgraded or upgraded for gray scale, multispectral, and hyperspectral images. Another benefit of working under the spectral domain is that color can be operated as in our physical world, e.g., predicting color of mixed pigments. Since these algorithms work pixel-wise, they can also be GPU accelerated. Therefore, these proposed algorithms are promising to be used in applications in computer vision.
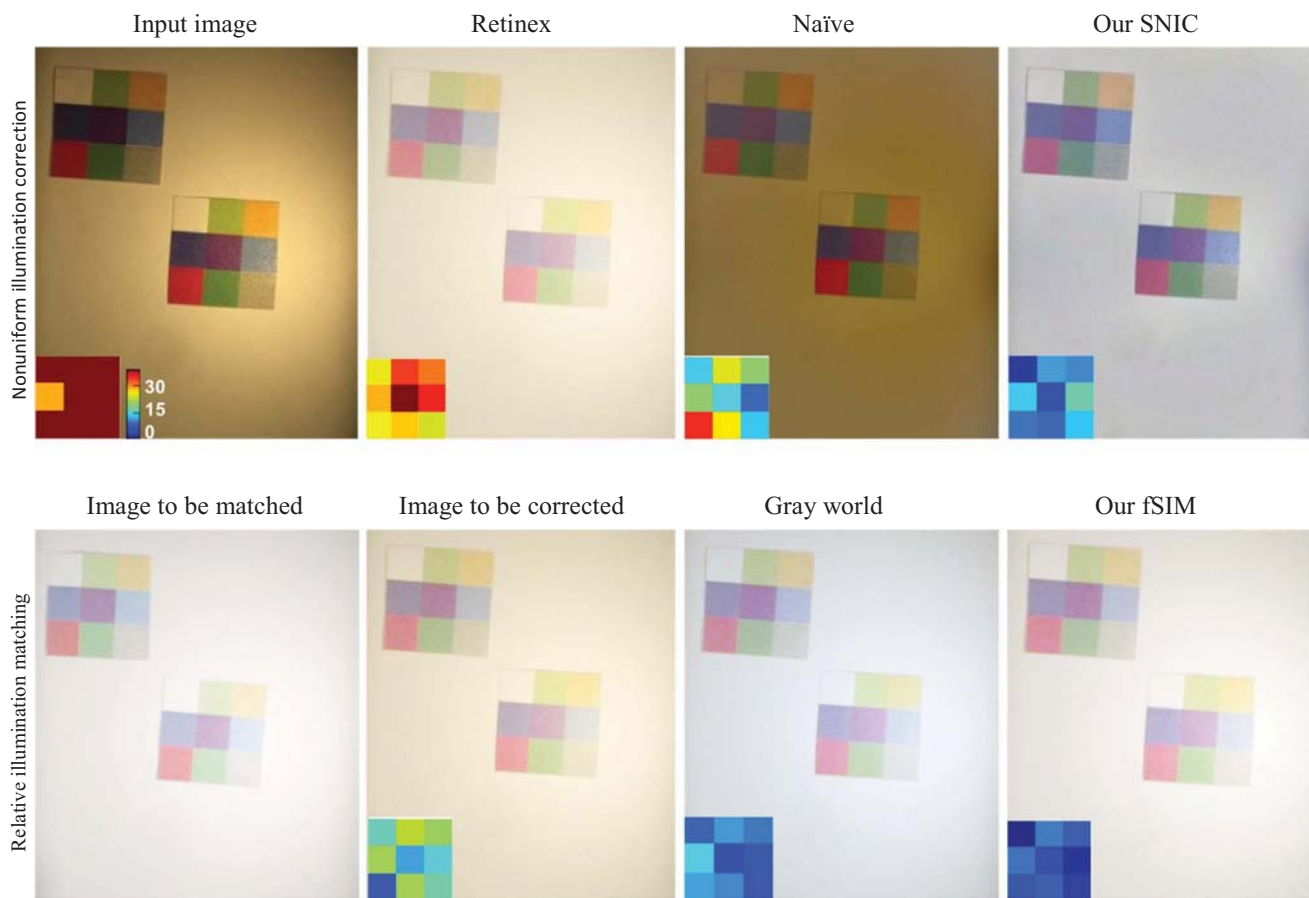
Fig. 5. This figure visualizes the results of nonuniform illumination correction (the first row) and relative illumination matching (the second row). In the first row, the first image shows the sample input image with nonuniform illumination. The following three images show the results after applying retinex, naïve, and SNIC algorithms respectively. In the second row, the first and second images are the sample images to be matched and corrected respectively. The third and fourth image show the results after applying the gray world and fSIM algorithms respectively. The heatmap of each image indicates the RSME of that specific test. The heatmaps in the first row evaluate the relative color constancy between the color patch at center and corner for each image. The heatmaps in the second row evaluate the relative color constancy between the color patch located at center of the illumination corrected image and image to be matched.

REFERENCES

[1] A. Gijsenij, T. Gevers, and J. Van De Weijer, "Computational color constancy: Survey and experiments," vol. 20, no. 9, pp. 2475–2489, 2011.

[2] B. V. Funt and G. D. Finlayson, "Color constant color indexing," vol. 17, no. 5, pp. 522–529, 1995.

[3] P. V. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp, "Bayesian color constancy revisited," presented at the Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1–8.

[4] D. A. Forsyth, "A novel algorithm for color constancy," vol. 5, no. 1, pp. 5–35, 1990.

[5] A. Gijsenij, T. Gevers, and J. Van De Weijer, "Generalized gamut mapping using image derivative structures for color constancy," vol. 86, no. 2–3, pp. 127–139, 2010.

[6] D. Cheng, B. Price, S. Cohen, and M. S. Brown, "Beyond white: Ground truth colors for color constancy correction," presented at the Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 298–306.

[7] E. H. Land and J. J. McCann, "Lightness and retinex theory," vol. 61, no. 1, pp. 1–11, 1971.

[8] M. Ebner, Color constancy, vol. 6. John Wiley & Sons, 2007.

[9] ISO 17321-1:2006 Graphic technology and photography Color characterization of digital still cameras-Part 1: Stimuli, metrology and test procedures.

[10] W. d. W. Abney, Researches in Colour Vision, Longmans, Green, London, 1913.

[11] P. Egan, F. Lakestani, M. P. Whelan, and M. J. Connelly, "Application of a logarithmic complementary metal–oxide–semiconductor camera in white-light interferometry," 2008.

[12] M. Abadi et al., "Tensorflow: a system for large-scale machine learning.," presented at the OSDI, 2016, vol. 16, pp. 265–283.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.

[14] G. Buchsbaum, "A spatial processor model for object colour perception," vol. 310, no. 1, pp. 1–26, 1980.

[15] V. Agarwal, B. R. Abidi, A. Koschan, and M. A. Abidi, "An overview of color constancy algorithms," vol. 1, no. 1, pp. 42–54, 2006.