LETTER

Color Constancy Based on Image Similarity

Bing LI^{†a)}, Student Member, De XU^{†b)}, Member, Jin-Hua WANG[†], and Rui LU[†], Nonmembers

SUMMARY Computational color constancy is a classical problem in computer vision. It is an under-constrained problem, which can be solved based on some constraint. Existing algorithms can be divided into two groups: physics-based algorithms and statistics-based approaches. In this paper, we propose a new hypothesis that the images generated under a same illumination have some similar features. Based on this hypothesis, a novel statistics-based color constancy algorithm is given and a new similarity function between images is also defined. The experimental results show that our algorithm is effective and it is more important that the dimension of the features in our algorithm is much lower than many former statistics-based algorithms.

key words: color constancy, image similarity

1. Introduction

The image recorded by a camera depends on three factors: the physical content of the scene, the illumination incident on the scene, and the characteristics of the camera [1]. The goal of color constancy is to recognize the color of objects invariant of the color of the light source. It generally includes two steps [3]. Firstly, estimate the color of light source from image data. Secondly, the illuminate-invariant descriptors are computed, which is used to adjust image in order to get the objects' colors under a known light source. The color constancy is important for a large variety of applications such as object recognition, scene understanding, image reproduction as well as digital photography [1].

So far, there are a number of leading color constancy algorithms. These algorithms can be divided into two groups: physics-based algorithms which are based on an understanding of physical imaging processes, and statistics-based approaches that exploit statistical knowledge of common lights and surfaces [4]. The Grey World (GW) algorithm, which is a classical algorithm for color constancy, assumed that the average reflectance in the scene is achromatic. Another simple color constancy method, called max-RGB, estimated the light source color from the maximum response of the different color channels [3]. In addition, Forsyth et al., proposed a gamut-mapping approach [5]. This method forms a set of all possible (R, G, B) due to surfaces in the world under a known "canonical" illumination at first. Then the set is represented by a convex hull. Secondly, the set of all possible

is represented under the unknown illumination by its convex hull. These two hulls are a unique diagonal mapping. The diagonal mapping is estimated at last. All these algorithm can be classified as physics-based approaches. The following algorithms are all statistics-based approaches. Such as Finlayson et al. proposed a Color by Correlation method to improve the Color in Perspective method [6]. Moreover, neural network was also used to estimate the color of the illumination [7], which can get the relationship between an image of a scene and the chromaticity of scene illumination through the training of neural network. In addition to the two statistical methods, there are illumination color by voting, probabilistic algorithms and so on.

Statistics-based algorithms are very important for color constancy, but its disadvantage is also very obvious. For many algorithms, the chromaticity space (r, g) is divided into $N \times N$ sub-spaces, then the chromaticity distribution in this space is used for color constancy [6]; Consequently, the dimension of the feature matrix is N^2 , which is too high for computation. In this paper, a low-dimensional feature is extracted for color constancy, both effect and efficiency are improved.

This paper is organized as follows. In Sect. 2, computational color constancy is described. Then we will explain the details of the proposed algorithm in Sect. 3. The experimental results and the comparison results are presented in Sect. 4. Section 5 concludes this paper.

2. Computational Color Constancy

According to the lambertian reflectance model, the image $f = (R, G, B)^T$ can be computed as follow:

$$f(x) = \int_w e(\lambda) s(x, \lambda) c(\lambda) d\lambda \quad (1)$$

where x is the spatial coordinate, λ is wavelength and w represents the visible spectrum. $e(\lambda)$ is color of light source, the surface reflectance is denoted as $s(x, \lambda)$ and the camera sensitivity function $c(\lambda) = (R(\lambda), G(\lambda), B(\lambda))$. The goal of color constancy is to estimate e .

$$e = \int_w e(\lambda) c(\lambda) d\lambda \quad (2)$$

Because both $e(\lambda)$ and $c(\lambda)$ are unknown, it can not be solved without further assumptions. Existent algorithms for color constancy are nearly all based on some assumptions.

Manuscript received June 4, 2007.

Manuscript revised July 31, 2007.

[†]The authors are with Institute of Computer Science and Engineering, Beijing Jiaotong University, Beijing, China.

a) E-mail: bjtulb@gmail.com

b) E-mail: dxu@bjtu.edu.cn

DOI: 10.1093/ietisy/e91-d.2.375

In this paper, we propose a new assumption (similarity assumption): the images generated under the same or similar illumination have some similarities. Based on this hypothesis, a novel color constancy algorithm is proposed, which is called color constancy based on image similarity (CCBIS).

3. Color Constancy Based on Image Similarity

Based on the image similarity hypothesis, the color constancy computation process includes three steps: Firstly, we must have got the similar feature matrix of each image to compute the similar degree between images. Secondly, suppose that the illumination of an image (Noted as I_u) is unknown and being estimated. The n images, which are most similar to the I_u , can be found out from the image database, in which the images' illumination are known. At last, based on the similarity assumption, the illuminations of the n images may be also similar to the I_u 's illumination. Consequently, the I_u 's illumination is estimated according to the n images' illumination, then we map the image I_u to the canonical illumination to finish the color constancy. The flow chart of the algorithm is shown as Fig. 1.

3.1 Feature Extraction

The first step of CCBIS is feature extraction from images, but before this step, a preprocess step for images is necessary. At first, the pixels whose color values are too low must be removed, because the object which has dark color can't reflect the light well and will affect the algorithm performance.

$$newI = \{I - P|R(P) + G(P) + B(P) \leq \delta\} \quad (3)$$

where the δ is the threshold. δ can't be very large, if so, many useful pixels will be removed. According to experiments, $\delta = 6$ is suitable for most images. The $newI$ is the updated image of I after removing the dark pixels. P represents the pixel in the image I , The $R(P)$, $G(P)$ and $B(P)$

respectively represent the Red, Green, Blue 3 channel values of every pixel.

After having got $newI$, the region-growing method is used to segment the image on the constrain that all pixels in a region were within a certain absolute tolerance of each other. Because a region-growing algorithm just segments the images into regions in this algorithm, how to select initial seeds has nearly no affect to the performance. We simply use scanning image method to get seeds for region-growing (color distance threshold between regions is 25). The image $newI$ is segmented into Num regions as $A_1, A_2, \dots, A_{Num}$.

$$newI = A_1 + A_2 + \dots + A_{Num} = \sum_{i=1}^{Num} A_i \quad (4)$$

After the preprocess, the next step is extracting image features. In many existing algorithms (e.g. color by correlation [6]), chromaticity feature is usually used for estimate illumination. They map the $(R, G, B)^T$ 3-D color space to the $(r, g)^T$ 2-D chromaticity using Eq.(5) or (6). Then the chromaticity space is partitioned into $N \times N$ uniform regions as Fig. 2 shown. According to the image's chromaticity distribution in the r - g space, a N^2 dimension feature matrix can be extracted [2], [6]. The detail process can be found in [6].

$$r = R/B, g = G/B \quad (5)$$

$$r = R/(R + G + B) \quad g = G/(R + G + B) \quad (6)$$

The most obvious disadvantage of chromaticity feature above is that the dimension of the feature matrix is too high. So the computational cost is also time consuming. For this reason, the chromaticity feature is not considered in our algorithm. It is our goal to get a kind of low-dimensional and effective features for CCBIS. As we known, the Minkowski-norm is usually used for illumination estimation in color constancy. In this paper, Minkowski-norm is also used to compute the feature from image.

$$f(p) = \left(\frac{\int \bar{f}^p(j) dj}{\int dj} \right)^{\frac{1}{p}} \quad (7)$$

where the $\bar{f}(j)$ is the average color of the j th region in the image $newI$. Each $f(p)$ is one of the image features, many

Fig. 1 The flow chart of the algorithm CCBIS.

Fig. 2 The partition of the chromaticity space.

physics-based algorithms make $p = 1, 6, \infty$ for color constancy. We also compose image feature matrix with different p as follow:

$$F = (f_1, f_2, f_3)^T = \left(\frac{f(1)}{|f(1)|}, \frac{f(6)}{|f(6)|}, \frac{f(\infty)}{|f(\infty)|} \right)^T \quad (8)$$

When $p = \infty$, Eq. (7) results in computing the maximum of $\tilde{f}(j)$. Each f_i in F is a normalized R, G, B 3-D vector. So F is a 3×3 feature matrix, which represents a certain image.

3.2 Definition of the Similarity

In this section, we'll define the distance between two images, which indicates the similar degree between two images. Many color constancy algorithms discard intensity information of color [6]. The color vector's direction is much more important than its intensity. Consequently, we define the angle between two features as the distance function between two images.

$$\begin{cases} Dis(F_i, F_j) = \sum_{k=1}^3 \lambda_k \cos^{-1}(f_k^i \bullet f_k^j) \\ \sum_{k=1}^3 \lambda_k = 1 \end{cases} \quad (9)$$

where the F_i and F_j are the feature matrix of two images as Eq. (8). f_k^i is the k th feature vector of the F_i . λ_k is the weight value. Because both f_k^i and f_k^j are normalized vector, their product $f_k^i \bullet f_k^j$ is just the cos value of the angle formed by the two vectors. The $\cos^{-1}$ is a monotone decreasing function, so we can remove the $\cos^{-1}$ computation in order to simplify the computation and reduce the computational cost. We define a similarity degree function as follow:

$$\begin{cases} Sim(F_i, F_j) = \sum_{k=1}^3 \lambda_k (f_k^i \bullet f_k^j) \\ \sum_{k=1}^3 \lambda_k = 1 \end{cases} \quad (10)$$

So the larger is the $Sim(F_i, F_j)$, the smaller is the $Dis(F_i, F_j)$, the closer are the directions of the two vector. It means that the features of the two images are more similar. In the physics-based algorithms, $p = 6$ make the performance of algorithm much better than $P = 1$ and $p = \infty$ [3]. So in the experiment, we generally make λ_2 largest in the λ_k , that is, $\lambda_2 > \lambda_1$ and $\lambda_2 > \lambda_3$.

3.3 Illumination Estimation

Based on the features extracted above, illumination estimation is introduced in this section. Given an image, whose illumination is unknown. At first, according to the Eqs. (7)(8), the feature matrix of this image is as F_e . Secondly, we will find 3 most similar images in the image database, in which the images' illumination are known and normalized as $\hat{e}_i$. It means that we find 3 images with the features F_1, F_2, F_3 , which makes the similar degrees with the F_e are largest

according to Eq. (10). At last, the 3 images are used to estimate the illumination e_e of the image with the feature F_e . Suppose the illumination vectors of the 3 images are $\hat{e}_1, \hat{e}_2, \hat{e}_3$ respectively. The e_e is estimated as:

$$\begin{cases} e_e = \sum_{i=1}^3 k_i \hat{e}_i \\ k_i = \frac{Sim(F_i, F_e)}{\sum_{i=1}^3 Sim(F_i, F_e)} \end{cases} \quad (11)$$

where k_i is weight value, which is computed according to the similar degree function Sim . The larger is the Sim , the larger is the k_i . The $\hat{e}_e = e_e / |e_e|$ is the normalized the illumination vector of the image with unknown illumination. Then based on the diagonal model [1], we can map the image to the image under the canonical illumination.

4. Experiments

A large data set [1] of colorful object under varying light sources is used to test CCBIS. The data set contains 429 images. For all images, the correct illumination of them are measured labelling as e_l . Among them, 108 images with different illumination are used as the image database. The other 321 images with varying light source over 30 different scenes are used as CCBIS testing images. We will estimate the illumination e_e of each image using CCBIS and compute the error between e_e and e_l . The angular error between the estimated light source e_e and the measured light source e_l is used as an error measure.

$$\text{angular error} = \cos^{-1}(\hat{e}_l \bullet \hat{e}_e) \quad (12)$$

where $\hat{e}_l \bullet \hat{e}_e$ is the dot product of the two normalized vectors e_l and e_e . Ideally, the angle is 0, which is the case when both vectors have the same direction. The parameters λ_k in Eq. (10) are evaluated as: $\lambda_1 = 0.2, \lambda_2 = 0.6, \lambda_3 = 0.2$. In order to comparison, we also test the classic Color-by-Correlation method [6] on the same data set. In the Color-by-Correlation experiment, we make $N = 25$, so the chromaticity space is partitioned into 25×25 uniform regions. The mean angular error and feature dimension data are shown in Table 1. The "Mean" is the mean angular error of each algorithm.

From Table 1, we can get a conclusion that the CCBIS algorithm performs better than the Color-by-Correlation on this data set. And what is most important that the feature dimension in the CCBIS (9-D) is much lower than the Color-by-Correlation (625-D).

The following Fig. 3 shows an example of the CCBIS. We estimate the illumination of the (A) according to the illumination of the images in (D). Using diagonal model [1],

Table 1 Comparison result.

	Feature Dimension	Mean
Color-by-Correclation	25×25	9.0
CCBIS	3×3	7.4

Fig. 3 An example of the CCBIS.

Fig. 4 Examples of the CCBIS.

(A) is mapped to the images under the canonical illumination using the estimated illumination and measured illumination respectively (shown as (B) (C)). (A) is reddish over the image because of the illumination. After process with CCBIS, the color of white paper approaches white in (B), its color in (C) is much whiter than (B). But the effect of (B) is much closer to the (C).

Figure 4 shows another 3 experimental results. (A) is reddish and (D)(G) are bluish. After illumination estimation

with CCBIS and mapping using diagonal model, (B)(E)(H) are nearly the images under the canonical white illumination, which are almost as same as (C)(F)(I).

Generally, through the experiments and analysis, the CCBIS has at least three advantages: Firstly, the image feature's dimension in the CCBIS is much lower than many other color constancy algorithms. Secondly, a novel similarity function and a illumination estimation equation are defined. The experiments show that they are effective. At last, the low-dimensional image feature matrix can be expanded to other statistics-based color constancy approaches, which maybe improves their performances.

5. Conclusion

In this paper, we propose a novel color constancy algorithm based on the image similarity hypothesis. The algorithm is tested over a large data set, and results show that our algorithm is effective. What is more important is that the feature used in this algorithm has a much lower dimension than many former statistics-based approaches. We want to introduce the relevant feedback into the computation of color constancy to improve its performance in the future work.

References

- [1] K. Barnard, V. Cardei, and B.V. Funt, "A comparison of computational color constancy algorithms-part 1: Methodology and experiments with synthesized data," *IEEE Trans. Image Process.*, vol.11, no.9, pp.972-983, Sept. 2002.
- [2] K. Barnard, L. Martin, A. Coath, and B.V. Funt, "A comparison of computational color constancy algorithms-part 2: Experiments with image data," *IEEE Trans. Image Process.*, vol.11, no.9, pp.985-996, Sept. 2002.
- [3] J. van de Weijer and Th. Gevers, "Color constancy based on the grey-edge hypothesis," *IEEE Conf. Image Processing*, pp.722-725, Sept. 2005.
- [4] G. Schaefer, S. Hordley, and G. Finlayson, "A combined physical and statistical approach to colour constancy," *IEEE Conf. Computer Vision and Pattern Recognition*, pp.148-153, June 2005.
- [5] D. Forsyth, "A novel algorithm for color constancy," *Int. J. Comput. Vis.*, vol.5, pp.5-36, 1990.
- [6] G. Finlayson, S. Hordley, and P. Hubel, "Color by correlation: A simple, unifying framework for color constancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.23, no.11, pp.1209-1221, Nov. 2001.
- [7] B.V. Funt, V. Cardei, and K. Barnard, "Learning color constancy," *Proc. IS&T/SID Fourth Color Imaging Conference: Color Science, Systems and Applications*, pp.58-60, 1996.