Ranking-based Color Constancy with Limited Training Samples

Bing Li, Haina Qin, Weihua Xiong, Yangxi Li, Songhe Feng, Weiming Hu, Stephen Maybank

Abstract—Computational color constancy is an important component of Image Signal Processors (ISP) for white balancing in many imaging devices. Recently, deep convolutional neural networks (CNN) have been introduced for color constancy. They achieve prominent performance improvements comparing with those statistics or shallow learning-based methods. However, the need for a large number of training samples, a high computational cost and a huge model size make CNN-based methods unsuitable for deployment on low-resource ISPs for real-time applications. In order to overcome these limitations and to achieve comparable performance to CNN-based methods, an efficient method is defined for selecting the optimal simple statistics-based method (SM) for each image. To this end, we propose a novel ranking-based color constancy method (RCC) that formulates the selection of the optimal SM method as a label ranking problem. RCC designs a specific ranking loss function, and uses a low rank constraint to control the model complexity and a grouped sparse constraint for feature selection. Finally, we apply the RCC model to predict the order of the candidate SM methods for a test image, and then estimate its illumination using the predicted optimal SM method (or fusing the results estimated by the top k SM methods). Comprehensive experiment results show that the proposed RCC outperforms nearly all the shallow learning-based methods and achieves comparable performance to (sometimes even better performance than) deep CNN-based methods with only 1/2000 of the model size and training time. RCC also shows good robustness to limited training samples and good generalization crossing cameras. Furthermore, to remove the dependence on the ground truth illumination, we extend RCC to obtain a novel ranking-based method without ground truth illumination (RCC_NO) that learns the ranking model using simple partial binary preference annotations provided by untrained annotators rather than experts. RCC_NO also achieves better performance than the SM methods and most shallow learning-based methods with low costs of sample collection and illumination measurement.

Index Terms—Color constancy, Illumination estimation, Ranking, Feature selection.

1 Introduction

Color, as one of the most important visual cues, has been used in many computer vision applications, such as white balancing, image understanding, and image matching [1], [2], [3]. The color signal (i.e., RGB values) from any imaging device is affected by three factors: the color of light incident on the scene, the object's surface reflectance, and the sensor sensitivity functions of the camera [2], [3], [4]. Therefore, the same surface, under different lights, usually has a different color. Computational color constancy aims to remove the effect of illumination and stably perceive color of the surface. The crucial step is illumination estimation that tries to estimate the color of the light [4], [5].

The illumination estimation methods are usually divided into two categories: statistics-based methods (SM) and learning-based methods (LM) [6], [7]. The SM methods estimate the illumination using the nature of the image itself without any training stage, such as MaxRGB (also called White Patch, WP) [8], Grey World (GW) [9], and Grey Edge (GE) [10]. The LM methods find a mapping function between the image (or its visual features) and its illumination using machine learning methods. According to which kind of machine learning technique is used, the LM methods can be further divided into shallow LM methods and deep LM methods.The methods belonging to the former one always use the traditional shallow learning methods, such as Support Vector Regression (SVR) [11] and Bayesian model [12], to learn the relationship between these handcrafted features and illumination values. The deep LM methods try to build up an end-to-end deep convolutional neural networks (CNN) model [13], [14], [15], [16], [17] between input images and illumination values. Recently, the deep LM methods [13], [14], [15], [16], [17] achieve better results than shallow LM methods [11], [12]. Although past decades have witnessed great development of both SM and LM methods, they still have the following limitations:

- The robustness of the SM methods is limited because their models depend on some very specific assumptions. Once the model is determined, it remains fixed for all test images. It is difficult for a fixed model to be robust and effective enough to handle the wide

- *Bing Li is with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China, also with People AI Inc., Beijing 100080, China.*
- *Haina Qin is with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China, also with the School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS), Beijing 100490, China.*
- *Weihua Xiong is with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China.*
- *Yangxi Li is with the National Computer Network Emergency Response Technical Team Coordination Center of China (CNCERT/CC), Beijing 100029, China.*
- *Songhe Feng is with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China.*
- *Weiming Hu is with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.*
- *Stephen Maybank is with the Department of Computer Science and Information Systems, Birkbeck College, UK.*
- *Corresponding Author: Weiming Hu (wmhu@nlpr.ia.ac.cn)*

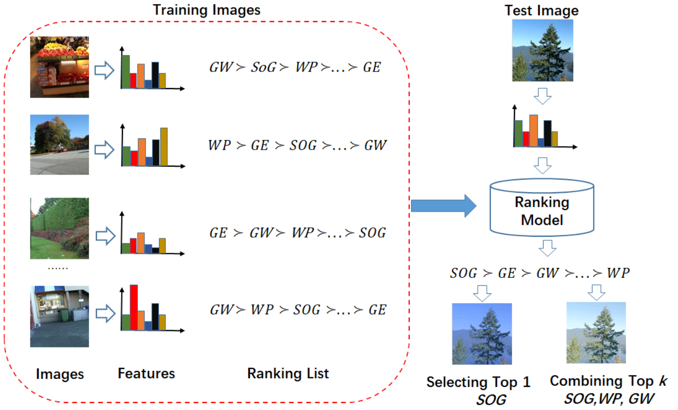


Fig. 1. An overview of the Ranking-based Color Constancy. The binary relation $M1 \succ M2$ means that $M1$ method achieves better performance than $M2$ on this image.

variety of images in real-world scenarios.

- The deep LM methods need a large number of labelled training samples. Unlike some other computer vision tasks, it is expensive and time-consuming to collect training images with accurate illumination values for color constancy research.
- Most deep LM methods should run on high-performance Graphics Processing Units (GPUs) during both the training and inference stages. The high computational cost makes these methods unsuitable for some real-time systems, especially for image signal processors (ISP) in cameras and cellphones.

In order to overcome the aforementioned limitations and achieve comparable performance to the deep LM methods, we design an efficient method based on existing simple SM methods with limited training samples. Given a number of images and a certain number of simple SM methods, we try to select the optimal one for each image. In this way, significant improvements in performance can be obtained. Consequently, we propose a novel ranking-based color constancy method (RCC). The paradigm of the proposed RCC method lies in the fact that the optimal SM method selection for an image should be a relative ranking problem, rather than an exclusive selection problem. As shown in Fig. 1, for each training image, M candidate SM methods are firstly ranked according to their estimation results with the ground truth illumination of the image. We then propose a novel ranking model with a specific loss function. The low rank and grouped sparse constraints are also introduced to explicitly control the model complexity so that a reliable prediction model can be learned even when the number of training samples is limited. The ranking model is trained using images and their corresponding rankings of the SM methods. During the test stage, the ranking model is used to predict the order of the candidate SM methods for a test image, and then estimate its illumination using the predicted optimal SM method (or fusing the results estimated by the top k SM methods).

The method in the literature most similar to RCC is the natural image statistics-based method (NIS) [18]. NIS models the optimal SM method selection as an 'exclusively selection' through a multi-class classification strategy. Compared with NIS, RCC has the following advantages:

- RCC takes the ordinal relationship of all SM candidates into account during the training procedure, while NIS only considers the best one but ignores relative ranking and the relationship among different candidates.
- The ranking model in RCC makes it less sensitive to the imbalanced number of training examples. The multi-class classification in NIS easily suffers from an imbalanced data distribution that becomes more severe with the increase of the number of classes.
- NIS pre-defines the Weibull-based texture descriptor as the visual features of images. However, the best choice of the features is unclear. In contrast, RCC learns the most effective features during the training stage.

Ideally, the ranking list for each image in RCC should be computed using the ground truth illumination. However, it is expensive and time-consuming to obtain the ground truth illumination of an image. In order to measure an accurate illumination value, each image should contain a color checker in the scene [19] [20] [21]. Then the position of the color checker in the image should be marked manually. The illumination value is computed using the color checker by color experts, rather than by untrained annotators. In order to remove the dependence on the ground truth illumination, we further extend RCC to propose a novel ranking-based method without ground truth illumination (RCC_NO). RCC_NO learns the ranking model based on simple partial binary preference annotations provided by any untrained annotator.

The contributions of this paper are summarized as follows:

- It formulates finding the optimal SM method for an image as a label ranking problem, rather than a classification problem. To our best knowledge, no previous methods use label ranking to solve the color constancy problem.
- It proposes a novel ranking model, RCC, based on a specific ranking loss. It introduces a low-rank constraint to control the model complexity and a grouped sparse constraint for feature selection. Thus a reliable prediction model can be learned, even when the number of training images is limited.
- It presents an efficient optimization algorithm for RCC based on the Accelerated Proximal Gradient method (APG) and Alternating Direction Method (ADM) [22].
- It further extends RCC to a novel ranking-based method without ground truth illumination (RCC_NO) that can learn from binary preference annotations supplied by untrained annotators. RCC_NO reduces the requirements of illumination measurements for color constancy and is more practical for imaging devices.
- A large number of experimental results show that RCC is comparable to or even better than most complex deep LM methods, even though it has a much

lower time complexity and requires fewer training images. The proposed RCC can easily be extended to incorporate new features and new SM/shallow LM methods for illumination estimation.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 describes the proposed RCC method in detail. Sections 4 and 5 demonstrate the experimental results and analysis. Section 6 extends RCC to RCC_NO. Section 7 concludes the paper.

2 Related Work

The color signal of the pixel $\mathbf{x}$ in RGB channels , $\mathbf{f}(\mathbf{x}) = [f_R(\mathbf{x}), f_G(\mathbf{x}), f_B(\mathbf{x})]^T$, recorded by a camera for light reflected from a matte surface at corresponding spatial coordinate $\mathbf{x}'$ depends on the surface reflectance $S(\mathbf{x}', \lambda)$, the spectral power distribution $P(\lambda)$ of the light, and the camera's spectral sensitivity functions $\rho(\lambda) = [\rho_R(\lambda), \rho_G(\lambda), \rho_B(\lambda)]^T$ [5]:

$$f_c(\mathbf{x}) = \int\limits_{\omega} P(\lambda)S(\mathbf{x}', \lambda)\rho_c(\lambda)d\lambda, c = \{R, G, B\}, \quad (1)$$

where ω is the visible spectrum interval, and λ indicates wavelength. Assuming that the relative spectral power distribution remains the same throughout the scene, for the case of an ideal 'white' surface reflectance, we obtain the color corresponding to the illumination as $\mathbf{e} \in [R, G, B]^T$ (in RGB channels). Given an image, the value $\mathbf{f}(\mathbf{x})$ of each pixel is known, the color constancy aims to estimate the illumination $\mathbf{e}$.

During the last decades, the scientific community and the imaging industry have witnessed rapid development in illumination estimation [4], [23]. This section briefly reviews both SM and LM methods.

2.1 Statistics-based Method (SM)

The SM methods use the statistical nature of the image's color channels themselves to estimate the illumination without learning stage. The simplest SM method is the MaxRGB algorithm (also called White Patch) [8] that estimates the illumination using the maximum response from the different color channels. The Grey World (GW) algorithm [9], which assumes that the average of the channels represents the illuminant color, is another widely-used SM method. Finlayson et al. [24] integrate GW and MaxRGB into a unified Shades of Grey algorithm (SoG) framework using the Minkowski-norm. Weijer et al. [10] further generalize the SoG to obtain a Grey Edge framework (GE) by including higher-order derivatives and the Minkowski-norm. Gao et al. [25] propose a SM method by modeling the double-opponent cells (DO) of the human visual system. They further explore the human visual system and propose a retinal mechanism inspired method (RM) [26]. Li et al [27] propose to use achromatic surfaces (AS) and Yang et al [28] propose a method using grey pixels (GP) for illumination estimation. Both AS and GP find achromatic pixels in natural scenes for illumination estimation. Cheng and Brown [29] find that large color differences (LCD) are highly correlated with the illumination and select bright and dark pixels for illumination estimation. Tan et al [30] estimate illumination using an inverse-intensity chromaticity space. Bianco and Cusano [7] propose a quasi-unsupervised method (QU) that does not require ground truth illumination information but still needs a training phase.

2.2 Learning-based Method (LM)

This section reviews the LM methods from two categories: shallow LM and deep LM.

2.2.1 *Shallow Learning-based Method*

A classical shallow LM method is Bayesian color constancy (BCC), which is proposed by Brainard and Freeman [12] and extended by Gehler et al. [31]. Traditional shallow neural networks (NN) are also introduced by Cardei et al. [32] for illumination estimation after feeding binarized chromaticity histograms. To improve the generalization of NN, Xiong et al [11] replace the neural networks with Support Vector Regression (SVR). Color by correlation (CbyC) [33] models the relationship between illuminants and chromaticity distributions using a correlation matrix. The illuminant with the highest probability is chosen as the final estimate. Chakrabarti et al. [34] propose spatio-spectral statistics features (SSS) for learning an estimation model using maximum likelihood. Gamut mapping algorithm (GM) [22] is another typical Shallow LM method that finds the relationship between the illumination and the canonical gamut. Gijsenij et al. [35] generalize GM by adding the derivative structures of images. The corrected-moment method (CMM) [36] learns a regression matrix to map the color moments of an image to its illumination. Cheng et al. [37] train regression trees on a set of simple features for illumination estimation (SF). Barron [38] proposes a convolutional color constancy (CCC) method, in which a convolutional filter is used for illumination estimation. He further speeds up the CCC method with a fast Fourier color constancy (FFCC) method [39] in the frequency domain.

Besides low-level visual features, some high-level scene knowledge can be introduced into shallow LM methods. Vazquez-Corral et al. [40] extend CbyC to a category correlation-based method after integrating color categories. Exemplar-based color constancy (ECC) [41] finds the nearest neighbor surfaces for each surface in a test image for illumination estimation. Li et al. [5] propose a multi-cue illumination estimation framework (MC) based on a tree-structured joint sparse representation by combining low-, mid- and high-level features. Faces and skin regions are also used as cues for illumination estimation [42]. Gijsenij et al. [18] use Weibull parameterization for natural image statistics (NIS) to improve performance of SM methods. Bianco et al [43] select the best SM method for each image based on its content-related features using a decision forest (IC). They also introduce indoor and outdoor scene category as a cue for illumination estimation (IO) [44]. Lu et al. [45] use 3D scene geometry to model an image and select the best SM for each region (SG). Weijer et al. [46] propose a high-level visual information-based method (HVI). It firstly applies several SM methods to compute a set of possible illuminants, and then selects the optimal one resulting in the most likely semantic composition of the image (such as green grass and grey road).

2.2.2 Deep Learning-based Method

More recently, deep convolutional neural networks (CNN) have been used for illumination estimation in an end-to-end manner and achieve impressed performance improvements. Bianco et al. [13] pioneer a CNN-based method (CNNM) to predict the scene illumination without hand-crafted features. They [14] extend the network architecture to include both single and multiple illumination estimation. A fully convolutional network architecture (FC^4) [15] is also applied to learn the confidence weights of image patches and estimate illumination by fusing the estimates obtained from the patches. Shi et al. [16] design a deep specialized network architecture (DS-net) with two interacting subnetworks, namely a hypotheses network that generates multiple illuminant hypotheses and a selection network that adaptively picks the most plausible hypotheses. Oh et al. [17] cast the color constancy problem as an illumination classification problem using CNN (C-CNN), in which images are first clustered based on the illumination color. The clusters are then used as ground truth to train the networks. Xu et al [6] introduce an illuminant-guided triplet network (IGTN) to generate a discriminative feature for achieving accurate illuminant estimation. Lo et al [47] propose a contrastive learning-based color constancy method (CLCC), in which deep neural networks learn better illuminant-dependent features via a raw-domain sample augmentation. Moreover, cues from multiple cameras are also explored for color constancy. Abdelhamed et al [48] leverage the availability of two cameras in smart phones for illumination estimation by proposing a two-camera network (TCN). Hernandez-Juarez et al [49] propose a multi-hypothesis-based method (MH), in which a Bayesian framework and an agnostic CNN are used for illumination estimation. Xiao et al [50] design multi-domain networks (MDN) by taking advantage of cross-camera training data to improve the performance of illumination estimation.

3 Ranking-based Color Constancy

This section models illumination estimation as a label ranking problem and proposes a ranking-based color constancy method.

3.1 Formulation

Given an image I_i (i is the index for a set of images) and a set of M candidate SM methods $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$, we set $\mathbf{z}_{i,j} = \psi_j(I_i) \in [R, G, B]^T$ (in RGB channels) i.e. the estimated illumination color for I_i using the j^{th} candidate SM method ψ_j. The performance of each candidate SM method for I_i is evaluated according to the angular error [1], [2]. For an image I_i, the angular error $ang(\mathbf{a}_i, \mathbf{e}_i)$ between the estimated illumination color $\mathbf{a}_i \in [R, G, B]^T$ and the ground truth color $\mathbf{e}_i$ is defined as

$$ang(\mathbf{a}_i, \mathbf{e}_i) = \cos^{-1}\left(\frac{\mathbf{a}_i \bullet \mathbf{e}_i}{\|\mathbf{a}_i\| \, \|\mathbf{e}_i\|}\right) \times \frac{180^\circ}{\pi}, \tag{2}$$

where $\|\bullet\|$ is the modulus of a vector. We rank the candidate SM methods in $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$ for the image I_i according to the angular errors. Consequently, we define a binary relation $\succ$ on the finite set $\boldsymbol{\psi}$ as $\psi_j \succ \psi_k$ if $ang(\psi_j(I_i), \mathbf{e}_i) < ang(\psi_k(I_i), \mathbf{e}_i)$ [51]. The pair $(\boldsymbol{\psi}, \succ)$ is called a partial order set (or poset) and is transitive ($\psi_j \succ \psi_k \wedge \psi_k \succ \psi_p \Rightarrow \psi_j \succ \psi_p$) [51]. Therefore a full ranking list L_i: $\psi_{q_1} \succ \psi_{q_2} \succ ... \succ \psi_{q_M}$ for image I_i is obtained according to $ang(\psi_{q_1}(I_i), \mathbf{e}_i) < ang(\psi_{q_2}(I_i), \mathbf{e}_i) < ... < ang(\psi_{q_M}(I_i), \mathbf{e}_i)$. The ranking list L_i is the performance ranking of the candidate SM methods for the image I_i.

Assume that we are given N training images $I_1, ..., I_N$ along with their corresponding ground truth illumination colors $\mathbf{e}_1, ..., \mathbf{e}_N \in [R, G, B]^T$. Let $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$ be a set of M candidate SM methods. For each training image I_i, $x_i \in R^d$ is its visual feature vector with d dimensions. We can obtain its corresponding SM ranking list L_i based on angular errors. RCC is to learn a function mapping the visual feature to the ranking list. If $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$ is treated as a label set, RCC can be viewed as a solution to the label ranking problem.

According to the formulation, we would like to learn a ranking function that assign a higher score to a SM method ψ_j than to a SM method ψ_k for the image I_i, if $\psi_j \succ \psi_k$. The ranking function is defined as follows. Let $g_j(\bullet)$ be the score prediction function (is defined after (5)) for the SM method ψ_j, let $\delta(\bullet)$ be a logistic loss function (defined as $\delta(z) = \log(1 + e^{-z})$), and let $\varepsilon_{j,k}(x_i)$ measure the error in ranking ψ_j and ψ_k for image I_i. The error $\varepsilon_{j,k}(x_i)$ is defined as follows:

$$\varepsilon_{j,k}(x_i) = \mathbf{1}_{L_i}(\psi_j \succ \psi_k)\delta(g_j(x_i) - g_k(x_i)), \tag{3}$$

where $\mathbf{1}_{L_i}(\psi_j \succ \psi_k)$ is an indicator function defined as:

$$\mathbf{1}_{L_i}(\psi_j \succ \psi_k) = \left\{ \begin{array}{ll} 1, & (\psi_j \succ \psi_k) \in L_i \\ 0, & (\psi_j \succ \psi_k) \notin L_i \end{array} \right. . \tag{4}$$

The underlying meaning of (3) and (4) is that, if $\psi_j \succ \psi_k$, the score $g_j(x_i)$ should also be higher than $g_k(x_i)$. Using ranking error $\varepsilon_{j,k}(x_i)$, we can obtain the ranking error for an image I_i as:

$$\varepsilon(x_i) = \sum_{j,k}^{M} \varepsilon_{j,k}(x_i). \tag{5}$$

If we assume that the score prediction function $\{g_j(\bullet)\}_{j=1}^{M}$ are linear functions and represented as $g_j(x_i) = \mathbf{w}_j^T x_i$. We define $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M] \in R^{d \times M}$ and the overall loss function $J(\mathbf{W})$ of N training images can be computed as:

$$\begin{aligned} J(\mathbf{W}) &= \tfrac{1}{N} \sum_{i=1}^{N} \sum_{j,k}^{M} \varepsilon_{j,k}(x_i) \\ &= \tfrac{1}{N} \sum_{i=1}^{N} \sum_{j,k}^{M} \mathbf{1}_{L_i}(\psi_j \succ \psi_k)\delta(\mathbf{w}_j^T x_i - \mathbf{w}_k^T x_i) \end{aligned} \tag{6}$$

A straightforward approach to search for the matrix $\mathbf{W}$ is to minimize the loss function $J(\mathbf{W})$. This simple approach could lead to the overfitting to the training data when the number of the training images is relatively small. Consequently, an additional regularization term $\Omega(\mathbf{W})$ is introduced to control the model complexity and prevent the overfitting to the training data, so we solve

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \left\{ J(\mathbf{W}) + \Omega(\mathbf{W}) \right\}. \tag{7}$$

The regularization term $\Omega(\mathbf{W})$ in (7) is crucial for the proposed RCC since it determines the interplay of the different SM methods and related features. In this paper, we define $\Omega(\mathbf{W})$ by focusing on two issues:

(1) Limited training samples. In order to make (7) be trained on limited training samples, we effectively take advantage of the correlation among different labels in $\boldsymbol{\psi}$ by assuming that the prediction functions with different $\mathbf{w}_j$ are linearly dependent [52]. Consequentially, $\mathbf{W}$ is a low rank matrix that is always represented as a relaxed form with nuclear norm $\|\|_*$, as $\|\mathbf{W}\|_*$ [53].

(2) Feature selection. Since it is still unclear which visual features have high correlation to SM methods [18], [43], [44], [45], we introduce a feature selection constraint based on a grouped sparsity constraint with $\ell_{2,1}$-norm on $\mathbf{W}$(as $\|\mathbf{W}\|_{2,1} = \sum\limits_{j=1}^{d} \sqrt{\sum\limits_{k=1}^{M} (\mathbf{W}_{j,k})^2}$), which can learn to select effective features automatically from a large number of visual features. The $\ell_{2,1}$-norm regularization encourages multiple score functions to share similar sparsity patterns so as to implement feature selection [54].

According to the above analysis, the regularization term $\Omega(\mathbf{W})$ can be written as $\Omega(\mathbf{W})=\lambda\|\mathbf{W}\|_* + \gamma\|\mathbf{W}\|_{2,1}$ with regularization coefficients λ and γ. Eq (7) can be rewritten as:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \left\{ J(\mathbf{W}) + \lambda\|\mathbf{W}\|_* + \gamma\|\mathbf{W}\|_{2,1} \right\}. \tag{8}$$

3.2 Optimization

To optimize (8), we decompose the objective function $F(\mathbf{W})$ into two parts $J(\mathbf{W})$ and $\Omega(\mathbf{W})$, i.e. $F(\mathbf{W}) = J(\mathbf{W}) + \Omega(\mathbf{W})$, where $\Omega(\mathbf{W})=\lambda\|\mathbf{W}\|_* + \gamma\|\mathbf{W}\|_{2,1}$. Based on this decomposition, the Accelerated Proximal Gradient (APG) algorithm [55], [56] can be used to solve (8). The APG has a convergence rate of $O(1/t^2)$ (t denotes the iteration count) with 'optimal' first-order gradients. The optimization of (8) using APG is given in Algorithm 1.

Algorithm 1 Optimization algorithm for (8)

```
Initialize: $H_0 > 0$, $\xi > 0$, $\mathbf{W}_{(0)} \in R^{d \times M}$, $\mathbf{V}_{(0)} = \mathbf{W}_{(0)}$, $\alpha_0 = 1$, and$t = 0$
Repeat
    1) Set $H = H_t$;
    2) While $F(r_H(\mathbf{V}_{(t)})) > R_H(r_H(\mathbf{V}_{(t)}), \mathbf{V}_{(t)})$
        $H = \xi H$;
    3) Set $H_{t+1} = H$;
    4) Compute
        $\mathbf{W}_{(t+1)} = \arg\min_{\mathbf{W}} R_{H_{t+1}}(\mathbf{W}, \mathbf{V}_{(t)})$;
        $\alpha_{t+1} = \frac{2}{t+3}$;
        $\delta_{t+1} = \mathbf{W}_{(t+1)} - \mathbf{W}_{(t)}$;
        $\mathbf{V}_{(t+1)} = \mathbf{W}_{(t+1)} + \frac{1-\alpha_t}{\alpha_t}\alpha_{t+1}\delta_{t+1}$;
    5) Set $t = t + 1$;
Until convergence of $\mathbf{W}_{(t)}$
Output $\mathbf{W}_{(t)}$
```

The generalized gradient update step in Algorithm 1 is defined as:

$$\begin{aligned} R_H(\mathbf{W}, \mathbf{W}_{(t)}) =& J(\mathbf{W}_{(t)}) + < \mathbf{W} - \mathbf{W}_{(t)}, \nabla J(\mathbf{W}_{(t)}) > \\ &+ \tfrac{H}{2} \left\| \mathbf{W} - \mathbf{W}_{(t)} \right\|_F^2 + \Omega(\mathbf{W}), \end{aligned} \tag{9}$$

$$r_H(\mathbf{W}_{(t)}) = \arg\min_{\mathbf{W}} R_H(\mathbf{W}, \mathbf{W}_{(t)}), \tag{10}$$

where $\mathbf{W}_{(t)}$ is the solution of $\mathbf{W}$ at the t^{th} iteration, $< \mathbf{A}, \mathbf{C} >= tr(\mathbf{A}^T\mathbf{C})$ denotes the matrix inner product, and $\nabla J(\mathbf{W}_{(t)})$ is the sub-differential of $J(\mathbf{W})$ at $\mathbf{W}_{(t)}$. After rewriting (10), we obtain

$$\begin{aligned} r_H(\mathbf{W}_{(t)}) =& \operatorname*{argmin}_{\mathbf{W}} \Big(J(\mathbf{W}_{(t)}) + < \mathbf{W} - \mathbf{W}_{(t)}, \nabla J(\mathbf{W}_{(t)}) > \\ & + \tfrac{H}{2} < \mathbf{W} - \mathbf{W}_{(t)}, \mathbf{W} - \mathbf{W}_{(t)} > + \Omega(\mathbf{W}) \Big) \\ =& \operatorname*{argmin}_{\mathbf{W}} \Big(< \mathbf{W}, \nabla J(\mathbf{W}_{(t)}) > + \tfrac{H}{2} < \mathbf{W}, \mathbf{W} > \\ & - H < \mathbf{W}, \mathbf{W}_{(t)} > + \Omega(\mathbf{W}) \Big) \\ =& \operatorname*{argmin}_{\mathbf{W}} \Big(\Big[< \mathbf{W}, \mathbf{W} > - 2 < \mathbf{W}, \mathbf{W}_{(t)} - \tfrac{1}{H} \nabla J(\mathbf{W}_{(t)}) > \\ & + < \mathbf{W}_{(t)} - \tfrac{1}{H} \nabla J(\mathbf{W}_{(t)}), \mathbf{W}_{(t)} - \tfrac{1}{H} \nabla J(\mathbf{W}_{(t)}) > \Big] + \tfrac{2}{H} \Omega(\mathbf{W}) \Big) \\ =& \operatorname*{argmin}_{\mathbf{W}} \Big(\left\| \mathbf{W} - (\mathbf{W}_{(t)} - \tfrac{1}{H} \nabla J(\mathbf{W}_{(t)})) \right\|_F^2 + \tfrac{2}{H} \Omega(\mathbf{W}) \Big) \end{aligned} \tag{11}$$

The $\nabla J(\mathbf{W}_{(t)})$ can be computed as $\nabla J(\mathbf{W}_{(t)}) = \frac{1}{N} \sum\limits_{i=1}^{N} \sum\limits_{j,k}^{M} \mathbf{1}_{L_i}(\psi_j \succ \psi_k) \nabla\delta(\mathbf{w}_j^T x_i - \mathbf{w}_k^T x_i) x_i (\mathbf{i}_j^M - \mathbf{i}_k^M)$, where $\mathbf{i}_j^M$ is a vector of M dimensions with all the elements zero except for the j^{th} element, which is 1. For the sake of simplicity, we denote $\mathbf{B} = \big(\mathbf{W}_{(t)} - \frac{1}{H} \nabla L(\mathbf{W}_{(t)})\big)$, $\tilde{\lambda} = \frac{2\lambda}{H}$, $\tilde{\gamma} = \frac{2\gamma}{H}$ and rewrite (11) as

$$r_H(\mathbf{W}_{(t)}) = \arg\min_{\mathbf{W}} \left(\|\mathbf{W} - \mathbf{B}\|_F^2 + \tilde{\lambda}\|\mathbf{W}\|_* + \tilde{\gamma}\|\mathbf{W}\|_{2,1} \right). \tag{12}$$

To solve (12), we first introduce an auxiliary variable $\mathbf{U}$ to make the objective function separable, i.e., (12) becomes

$$\begin{aligned} r_H(\mathbf{W}_{(t)}) = \arg\min_{\mathbf{W}} & \left(\|\mathbf{W} - \mathbf{B}\|_F^2 + \tilde{\lambda}\|\mathbf{W}\|_* + \tilde{\gamma}\|\mathbf{U}\|_{2,1} \right) \\ & s.t. \quad \mathbf{U} = \mathbf{W} \ . \end{aligned} \tag{13}$$

Considering the balance between efficiency and accuracy in practice, the convex problem defined in (13) can be solved with the Alternating Direction Method (ADM) [57], which minimizes the following augmented Lagrangian function ϑ:

$$\begin{aligned} \vartheta(\mathbf{W}, \mathbf{U}, \mathbf{Y}, \mu) =& \|\mathbf{W} - \mathbf{B}\|_F^2 + \tilde{\lambda}\|\mathbf{W}\|_* + \tilde{\gamma}\|\mathbf{U}\|_{2,1} \\ & + \mathrm{Tr}(\mathbf{Y}^T(\mathbf{W} - \mathbf{U})) + \tfrac{\mu}{2} \left\| \mathbf{W} - \mathbf{U} \right\|_F^2 , \end{aligned} \tag{14}$$

where $\mathbf{Y}$ is the Lagrangian multiplier and $\mu > 0$ controls the penalty for violating the linear constraints. We can obtain $\mathbf{W}$ by minimizing (14) with ADM [57]. We outline the optimization procedure in Algorithm 2 and provide the details for each iteration.

Updating $\mathbf{W}$. When $\mathbf{U}^k$ is fixed, the update $\mathbf{W}^{k+1}$ at the $(k+1)^{th}$ iteration is obtained by solving the following problem:

$$\begin{aligned} \mathbf{W}^{k+1} &= \underset{\mathbf{W}}{\arg\min}\, \vartheta(\mathbf{W}, \mathbf{U}^k, \mathbf{Y}^k, \mu^k) \\ &= \underset{\mathbf{W}}{\arg\min}\, \|\mathbf{W}-\mathbf{B}\|_F^2 + \tilde{\lambda}\|\mathbf{W}\|_* \\ &\quad + \mathrm{Tr}((\mathbf{Y}^k)^T(\mathbf{W}-\mathbf{U}^k)) + \tfrac{\mu^k}{2}\left\|\mathbf{W}-\mathbf{U}^k\right\|_F^2 \\ &= \underset{\mathbf{W}}{\arg\min}\, \tau\|\mathbf{W}\|_* + \tfrac{1}{2}\left\|\mathbf{W}-\mathbf{D}\right\|_F^2 \end{aligned} . \tag{15}$$

where $\tau = \tilde{\lambda}/(\mu^k+2)$ and $\mathbf{D} = (2\mathbf{B}+\mu^k\mathbf{U}^k-\mathbf{Y}^k)/(\mu^k+2)$. The solution to (15) can be derived as [58]:

$$\mathbf{W}^{k+1} = \mathbf{P}T_\tau[\mathbf{\Sigma}]\mathbf{Q}^T, where(\mathbf{P}, \mathbf{\Sigma}, \mathbf{V}^T) = SVD(\mathbf{D}), \tag{16}$$

where $\mathbf{\Sigma}$ is the matrix of the singular values of $\mathbf{D}$, and the operator $T_\tau[\bullet]$ is the singular value thresholding (SVT) [58] defined by element-wise τ thresholding of $\mathbf{\Sigma}$, i.e. $T_\tau[\mathbf{\Sigma}] = diag([t_\tau[\sigma_1], t_\tau[\sigma_2], ..., t_\tau[\sigma_r]])$ for $rank(\mathbf{\Sigma}) = r$, and each $t_\tau[\sigma_i]$ is determined as

$$t_\tau[\sigma_i] = \begin{cases} \sigma_i - \tau, & \sigma_i > \tau \\ 0, & otherwise \end{cases} \tag{17}$$

Updating $\mathbf{U}$. When $\mathbf{W}^{k+1}$ is fixed, the update of $\mathbf{U}^{k+1}$ at the $(k+1)^{th}$ iteration is obtained by solving the following problem:

$$\begin{aligned} \mathbf{U}^{k+1} &= \underset{\mathbf{U}}{\arg\min}\, \vartheta(\mathbf{W}^{k+1}, \mathbf{U}, \mathbf{Y}^k, \mu^k) \\ &= \underset{\mathbf{U}}{\arg\min}\, \tilde{\gamma}\|\mathbf{U}\|_{2,1} + \mathrm{Tr}((\mathbf{Y}^k)^T(\mathbf{W}^{k+1}-\mathbf{U}^k)) \\ &\quad + \tfrac{\mu^k}{2}\left\|\mathbf{W}^{k+1}-\mathbf{U}\right\|_F^2 \\ &= \underset{\mathbf{U}}{\arg\min}\, \eta\|\mathbf{U}\|_{2,1} + \tfrac{1}{2}\left\|\mathbf{U}-\mathbf{E}\right\|_F^2 \end{aligned} , \tag{18}$$

where $\eta = \tilde{\gamma}/\mu^k$ and $\mathbf{E} = \mathbf{W}^{k+1} + \mathbf{Y}^k/\mu^k$. The solution to (18) can be derived as [59]:

$$\mathbf{U}_i^{k+1} = \begin{cases} \mathbf{0}, & \|\mathbf{E}_i\| \le \eta \\ (1 - \frac{\eta}{\|\mathbf{E}_i\|})\mathbf{E}_i, & \|\mathbf{E}_i\| > \eta \end{cases} . \tag{19}$$

where $\mathbf{U}_i^{k+1}$ and $\mathbf{E}_i$ are the i^{th} row of $\mathbf{U}^{k+1}$ and $\mathbf{E}$, respectively.

Algorithm 2 Optimization algorithm for (14)

```
Input: $\mathbf{B} = \left(\mathbf{W}_{(t)} - \frac{1}{H}\nabla L(\mathbf{W}_{(t)})\right)$, $\tilde{\lambda} = \frac{2\lambda}{H}$, $\tilde{\gamma} = \frac{2\gamma}{H}$
Initialize: $\mathbf{W}^0 = \mathbf{U}^0 = \mathbf{0}$, $\mathbf{Y}^0 = \mathbf{0}$, $\mu^0 = 0.1$, $\rho = 1.1$, $k = 0$
While not converged do
    $\mathbf{W}^{k+1} = \underset{\mathbf{W}}{\arg\min}\, \vartheta(\mathbf{W}, \mathbf{U}^k, \mathbf{Y}^k, \mu^k)$
    $\mathbf{U}^{k+1} = \underset{\mathbf{W}}{\arg\min}\, \vartheta(\mathbf{W}^{k+1}, \mathbf{U}, \mathbf{Y}^k, \mu^k)$
    $\mathbf{Y}^{k+1} = \mathbf{Y}^k + \mu^k(\mathbf{W}^{k+1} - \mathbf{U}^{k+1})$
    $\mu^{k+1} = \rho\mu^k$
    $k = k + 1$
End While
Output $\mathbf{W}^k$
```

3.3 Illumination Estimation

Given the learned matrix $\mathbf{W}$ and a test image I_t with its feature vector $x_t \in R^d$, we compute the score for each SM candidate method ψ_j by the prediction function $g_j(x_t) = \mathbf{w}_j^T x_t$. Then the SM methods are re-ordered in the descending order of the predicted scores as $\psi_{q_1} \succ \psi_{q_2} \succ ... \succ \psi_{q_M}$ such that $g_{q_1}(x_t) > g_{q_2}(x_t) > ... > g_{q_M}(x_t)$. Based on the ranking list, we can estimate the illumination value of the test image I_t using two strategies.

(1) Select the optimal SM method. According to the predicted SM methods ranking list, the optimal method is selected as the final estimation method for the test image. The estimated illumination is computed as:

$$\mathbf{a}_t = \mathbf{z}_{t,q_1} = \psi_{q_1}(I_t). \tag{20}$$

(2) Combine the top k SM methods (method denoted as RCC_C). Another strategy is combining the illumination values estimated by the top k SM methods using simple weighted average. In order to simplify the weight selection, we use the top 3 SM method to compute the final estimate as:

$$\mathbf{a}_t = \alpha\hat{\mathbf{z}}_{t,q_1} + \beta\hat{\mathbf{z}}_{t,q_2} + (1-\alpha-\beta)\hat{\mathbf{z}}_{t,q_3} \tag{21}$$

where $\hat{\mathbf{z}}_{t,q_k} \in [r, g, b]^T$ ($r = R/(R+G+B), g = G/(R+G+B), g = 1-r-b$) is normalized 3D chromaticity values of the estimate $\mathbf{z}_{t,q_k} = \psi_{q_k}(I_t)$ and $\alpha, \beta \in [0, 1]$. The optimal values of α, β are determined by a simple exhaustive search method on a training set.

3.4 Feature Extraction

Feature ($x_i \in R^d$) extraction is an important step for the proposed RCC. Following Li [23], we design $x_i \in R^d$ including two kinds of features, namely a low-level-initial-estimate feature and a high-level-scene-content feature.

(1) Low-Level-Initial-Estimate Feature

For the low-level feature, we use a feature vector encoding the initial illumination estimates obtained from the candidate SM methods $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$. These estimates are low-level statistics of the colors. They can be viewed as the moments of color distribution [36]. In addition, the distribution of these initial estimates can reflect the relationship among different candidate methods in the illumination chromaticity space. For an image I, the estimate in RGB color space using the SM method ψ_j is $\psi_j(I) \in (R, G, B)^T$. We transform $\psi_j(I)$ from RGB space to the r-g chromaticity space $\psi_j^{rg}(I) \in (r, g)^T$ based on $r = R/(R+G+B), g = G/(R+G+B)$, then concatenate these estimated illumination chromaticities as the low-level-initial-estimate feature vector, shown as $[\psi_1^{rg}(I), \psi_2^{rg}(I), ..., \psi_M^{rg}(I)]^T \in R^{2\times M}$.

(2) High-Level-Scene-Content Feature

High-level information about the scene content is used as a cue for illumination estimation in [5], [18]. Images of the same scene category (e.g., indoor versus outdoor) tend to be taken under similar lighting conditions, so knowledge of the scene category can be exploited for illumination estimation [44], [45], [46]. Li's evaluations [23] show that two kinds of features, Weibull feature [60] and content-related feature [43], have high correlations to illumination estimation.

The Weibull feature is proposed by Geusebroek and Smeulders [60] as a texture descriptor. It correlates well with the image scene types (such as indoor/outdoor). They [60] find that the distribution of edge responses in an image can be modeled by fitting them with a two-parameter integrated Weibull distribution, as

$$wb(z) = \zeta \exp\left(-\frac{1}{\varphi}\left|\frac{z}{\omega}\right|^{\varphi}\right), \tag{22}$$

where z denotes the edge responses from a single channel to the Gaussian derivative filter, ζ is a normalization constant, $\omega > 0$ is the scale parameter of the distribution representing the image contrast, and $\varphi > 0$ is the shape parameter determining the grain size. The same as the work [18], the Weibull parameters $< \varphi, \omega >$ for each channel in the 3-d opponent color space $(O_1, O_2, O_3)^1$ [61] are combined to form a 6-d feature vector. For obtaining more details, we divide an image into right-half, left-half, top-half, and bottom-half sub-images and compute the Weibull features of each sub-image respectively resulting in a 24-d feature vector.

Bianco et al [43] divide the content-related feature for illumination estimation into two groups: general-purpose features and problem-dependent features. The general-purpose features include a color histogram (27 dimensions), an edge direction histogram (18 dimensions), an edge strengths histogram (5 dimensions), statistics on the wavelet coefficients (20 dimensions), and color moments (6 dimensions). The problem-dependent features include the number of different colors (1 dimensions), the clipped color components (8 dimensions), and the cast indexes (2 dimensions). For each image I_i, we can concatenate these values into an 87-d feature vector.

4 Experiments

This section evaluates the proposed RCC on two widely-used real-world image sets and gives ablation analysis on experimental results.

4.1 Experimental Setting

(1) SM candidate set construction

The SM candidate set $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$ is key for RCC. To construct the SM candidate set $\boldsymbol{\psi}$, the general Grey Edge framework [10] is used to generate different SM methods systematically with different parameter settings. The Grey Edge framework integrates derivatives of different orders and scales, as well as the Minkowski-norm, as

$$\left(\int\left|\frac{\partial^m \mathbf{f}^\sigma(\mathbf{x})}{\partial \mathbf{x}^m}\right|^p d\mathbf{x}\right)^{1/p} = \kappa \mathbf{e}^{m,p,\sigma}, \tag{23}$$

where $\mathbf{f}^\sigma(\mathbf{x}) = \mathbf{f}(\mathbf{x}) \otimes G^\sigma$ denotes convolution of the image $\mathbf{f}(\mathbf{x}) = [f_R(\mathbf{x}), f_G(\mathbf{x}), f_B(\mathbf{x})]^T$ (RGB color values at spatial coordinate $\mathbf{x}$) with a Gaussian filter G^σ of standard deviation σ, p is the Minkowski-norm value, κ is a scaling, and $\mathbf{e}^{m,p,\sigma}$ is the resulting illumination estimate.

The Grey Edge framework can generate different SM methods with different selections of m, p, σ. The parameter selection for constructing $\boldsymbol{\psi}$ should follow two principles: diversity and representativeness. **(1) Diversity.** In order to generate enough diverse SM methods for constructing $\boldsymbol{\psi}$, we set $m = 0, 1, 2$ to cover 0-, 1- and 2-order derivative image structures. For each m, the ranges of p and σ are set as $p = \{1, 4, 7, ..., 31\}$ and $\sigma = \{1, 4, 7, ..., 31\}$, deriving a SM pool with 363 different SM methods. **(2) Representativeness.** Since too many SM methods in $\boldsymbol{\psi}$ will result in the difficulty of training the RCC model, it is not reasonable to use all the 363 SM methods to compose the candidate set $\boldsymbol{\psi}$. Considering the diversity and representativeness of $\boldsymbol{\psi}$, the center of the parameter space $< p, \sigma > (< p, \sigma > = < 16, 16 >$) and four proximal centers of four subspaces (i.e. $< p, \sigma > = \{< 7, 7 >, < 7, 25 >, < 25, 7 >, < 25, 25 >\}$), as the red points shown in Fig.2, are selected for each m, which generates 15 representative SM methods to compose the set $\boldsymbol{\psi}$. Although this fixed parameter setting scheme is simple, it makes RCC more efficient and practical for real ISPs.

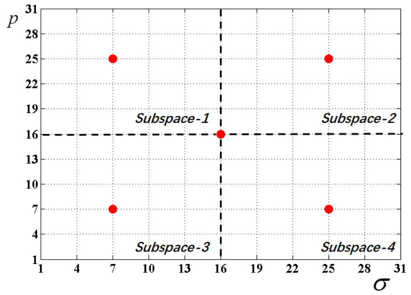


Fig. 2. Red points are the five representative parameter groups in $< p, \sigma >$.

(2) Error measurement

The angular error $ang(\mathbf{a}_i, \mathbf{e}_i)$ defined in (2) is used for evaluating illumination estimation methods. For an image set, the mean, median, trimean [62], best-25% (B-25%) and worst-25% (W-25%) errors are used to measure the performance of each method on this set. The worst-25% (or best-25%) indicates the mean angular error of the largest (or smallest) 25% of the angular errors on the test images [34].

The proposed RCC is comprehensively compared with nearly 30 prevailing illumination estimation methods from both SM and LM categories. For the SM category, we use GW [9], MaxRGB [8], SoG [24], GE (widely-used parameter settings $m, p, \sigma = \{(0, 13, 2), (1, 1, 6), (2, 15)\}$ as $\text{GE}^{0,13,2}$, $\text{GE}^{1,1,6}$, $\text{GE}^{2,1,5}$) [10], RM [26], GP [28], LCD [29], QU [7]. For the shallow LM category, we use BCC [12], NN [32], SVR [11] , SSS [34] , GM [22], DGM [35], CMM [36], SF [37], CCC [38], FFCC [39], ECC [41], MC [5], NIS [18], IC [43], IO [44], SG [45], HVI [46]. In addition, most recent deep LM methods, including CNNM [13], FC^4 [15], DS-net [16], C-CNN [17], IGTN [6], GLCC [47] are also used for the evaluation. Some deep LM methods, such as TCN [48], MH [49] and MDN [50], are not included in the experiments because they need images of the same scene from multiple cameras. We have implemented the proposed RCC method in Matlab. The optimal parameters $\lambda \in \{0.0001, 0.001, 0.1, 1\}$ and $\gamma \in \{0.0001, 0.001, 0.1, 1\}$ are selected through 3-fold cross validation on the training set in each of the following experiments.

$^1 O_1 = \frac{R-G}{\sqrt{2}}, O_2 = \frac{R+G-2B}{\sqrt{6}}, O_3 = \frac{R+B+G}{\sqrt{3}}$

TABLE 1
Performance comparison on the Gehler-Shi image set.

Category	Subcategory	Method	Mean	Median	Trimean	B-25%	W-25%
SM	/	DN [23]	9.24	4.80	6.68	1.43	24.0
		GW [9]	4.72	3.63	3.93	0.80	10.5
		SoG [24]	6.35	4.48	5.20	0.51	15.0
		MaxRGB [8]	10.2	9.15	9.48	1.45	20.5
		$\text{GE}^{0,13,2}$ [10]	6.28	3.90	4.76	0.50	15.8
		$\text{GE}^{1,1,6}$ [10]	4.15	3.28	3.54	0.95	8.73
		$\text{GE}^{2,1,5}$ [10]	4.19	3.35	3.62	1.05	8.59
		RM [26]	4.8	2.7	–	–	–
		GP [28]	4.6	3.1	–	–	–
		LCD [29]	3.52	2.14	2.47	0.50	8.74
		QU [7]	3.46	2.23	–	–	–
LM	Shallow	NN [32]	5.13	3.77	4.06	1.12	11.5
		SVR [11]	4.08	3.23	3.35	0.72	9.03
		SSS [34]	3.96	3.24	3.46	1.52	7.61
		GM [22]	5.96	3.98	4.53	0.83	14.3
		BCC [31]	4.82	3.46	3.88	1.26	10.49
		NIS [18]	4.40	3.18	3.49	0.54	10.3
		IC [42]	3.87	2.83	3.07	0.36	9.29
		IO [44]	5.18	3.55	4.00	0.65	12.4
		SG [45]	4.49	3.09	3.45	0.56	10.6
		HVI [46]	4.31	3.06	3.38	0.80	9.86
		MC [5]	3.25	2.20	2.55	0.30	8.13
		SF [37]	2.45	1.65	1.75	0.38	5.87
		FFCC# [39]	2.91	1.98	2.25	0.65	6.86
		ECC [41]	2.89	2.27	2.42	0.82	5.07
	Deep	CNNM [13]	2.63	1.89	–	–	–
		FC^4 [15]	1.65	1.18	1.27	0.38	3.78
		C-CNN [17]	2.16	1.47	1.61	0.37	5.12
		DS-net [16]	1.90	1.12	1.33	0.31	4.84
		IGTN [6]	1.58	0.92	–	0.28	3.70
		GLCC [47]	1.44	0.92	1.04	0.27	3.48
Our	Ranking	RCC	2.53	1.30	1.54	0.14	6.91
		RCC_C	2.37	1.20	1.42	0.22	6.66

4.2 Results on Gehler-Shi Set

The Gehler-Shi image set originally provided by Gehler et al. [31] contains 568 images taken with two cameras (Canon 5D and Canon1D). Shi et al. [19] further reprocess the raw data in this set and create linear (gamma=1) almost-raw 12-bit Portable Network Graphics (PNG) format images. To reduce the correlations between neighboring images in the set, uncorrelated threefold cross validation provided by Li et al. [23] is used instead of the random threefold cross validation in this set. The 'Do Nothing' (DN) method in this experiment uses the white points provided by Shi [5] for the images obtained from the Canon 5D and Canon1D cameras as $[1/1.6976, 1/0.92971/1.0237]$ and $[1/2.2245581/0.9286621/1.164364]$. The experimental results are shown in Table 1. The FFCC in Table 1 (denoted as FFCC#) uses uncorrelated threefold cross validation [23], rather than the random 3-fold cross validation in [39], by removing the high correlations among images.

As illustrated in Table 1, both RCC and RCC_C outperform all the SM methods and shallow LM methods in terms of median, trimean errors. Especially, compared with NIS, the mean, median, trimean errors of RCC are reduced by more than 40%, which indicates that the ranking strategy is more effective than the exclusively classification. Compared with the deep LM methods, according to the most important median and trimean errors [63], the proposed RCC_C outperforms both CNNM and C-CNN, and achieves comparable performance to both FC^4 and DS-net methods. It is worth noting that most of the deep LM methods in Table 1 use random cross-validation settings which inevitably result in a performance gain due to the high correlations in this image set. According to Table 1, both RCC and RCC_C achieve much lower angular errors using the proposed ranking model and SM methods, which also indicates that it's possible to obtain comparable performance to (even better performance than) deep LM methods using an effective ranking model with several simple SM methods. In addition, RCC needs much a lower computational cost and much less training samples than deep LM methods. The fact that RCC_C achieves better performance than RCC indicates that the simple linear fusion of top 3 SM methods can effectively improve the stableness and accuracy of the RCC model.

4.3 Results on NUS Set

The NUS set is composed of 1736 high-quality images produced by Cheng et al. [29]. These images are taken using 8 commercial cameras (Canon 1DS Mark III, Canon 600D, Fujifilm XM1, Nikon D5200, Olympus EPL6, Panasonic GX1, Samsung NX 2000, Sony 57). For each camera, over 200 images including both indoor and outdoor scenes are captured. The same as in the Gehler-Shi set, a color checker is used to measure the ground truth illumination for each image. In this experiment, for each camera, LM methods are trained and tested separately using a 3-fold cross validation. Then the average performance of 8 cameras are reported in Table 2.

TABLE 2
Performance comparison on the NUS image set.

Category	Subcategory	Method	Mean	Median	Trimean	B-25%	W-25%
SM	/	DN [23]	19.58	19.57	19.48	9.91	22.71
		GW [9]	4.14	3.20	3.39	0.90	9.00
		SoG [24]	3.40	2.57	2.73	0.77	7.41
		MaxRGB [8]	10.62	10.58	10.19	1.86	19.45
		$\mathrm{GE}^{0,13,2}$ [10]	4.04	2.45	2.78	0.37	10.49
		$\mathrm{GE}^{1,1,6}$ [10]	6.83	5.45	5.82	3.47	12.52
		$\mathrm{GE}^{2,1,5}$ [10]	9.99	9.40	9.55	7.05	13.83
		QU [7]	3.00	2.25	–	–	–
LM	Shallow	CMM [36]	3.05	1.90	2.13	0.65	7.41
		GM [22]	7.70	6.71	6.90	2.51	14.05
		DGM [35]	8.43	7.05	7.37	2.41	16.08
		SVR [11]	2.91	2.18	2.33	0.68	6.39
		BCC [31]	3.67	2.73	2.91	0.82	8.21
		NIS [18]	3.71	2.60	2.84	0.79	8.47
		IC [42]	7.20	5.96	6.28	2.20	13.61
		SSS [34]	3.11	2.49	2.60	2.84	0.79
		SSS(GP) [34]	2.96	2.33	2.47	0.80	6.18
		SF [37]	2.92	2.04	2.24	0.62	6.61
		CCC [38]	2.38	1.48	1.69	0.45	5.85
		FFCC [39]	1.99	1.31	1.43	0.35	4.75
	Deep	FC^4 [15]	2.23	1.57	1.72	0.47	5.15
		C-CNN [17]	2.41	2.15	–	–	4.16
		DS-net [16]	2.24	1.46	1.68	0.48	6.08
		IGTN [6]	1.85	1.24	–	0.36	4.58
		GLCC [47]	1.84	1.31	1.42	0.41	4.20
Our	Ranking	RCC	2.70	1.81	2.03	0.50	6.38
		RCC_C	2.62	1.43	1.76	0.52	6.06

According to Table 2, the proposed RCC and RCC_C achieve much better performance than all the SM methods. RCC_C outperforms all the shallow LM methods except FFCC. The CCC and FFCC treat the illumination estimation as an illumination classification problem on a certain image set. Since the illumination set of an image set in the two methods is a finite set, these illumination classification-based methods easily overfit to the image set. In addition, as a general framework, the proposed RCC can also include CCC and FFCC into the candidate set to improve the performance, which is discussed in Section 5.6. Compared with the deep LM methods, the proposed RCC_C outperforms C-CNN, FC^4 and DS-net, and achieves comparable performance to GLCC, but its model is much smaller and more efficient than deep CNN models.

4.4 Visual Comparison

Besides quantitative analysis above, this section gives some visual comparisons among different methods. Five images (2 indoor images and 3 outdoor images) from the image sets are selected out as examples. For each image, we correct the image colors under canonical light sources using the diagonal transformation [64] according to the estimated illumination by 9 methods, including GW [9], MaxRGB [8], GE ($\mathrm{GE}^{0,13,2}$, $\mathrm{GE}^{1,1,6}$, and $\mathrm{GE}^{2,1,5}$) [10], SVR [11], CNNM [13], RCC and RCC_C. The corrected results are shown in Fig. 3, in which the ideal results using ground truth (GT) are also given for comparison. According to Fig. 3, we can find that: (1) Some of the 9 competitors achieve better results for indoor images, while some obtain better results for outdoor ones. In contrast, RCC and RCC_C always outperform all the other methods on both indoor and outdoor images, which further proves their stableness. (2) The last two outdoor images contain few reflectance surfaces, which makes it more difficult to estimate illumination accurately. Consequently, most methods have not obtained reasonable results, but both RCC and RCC_C still achieve much lower angular errors and good corrected results. It benefits from the underlying ranking strategy that adaptively selects optimal SM methods for each image so as to avoid most unreasonable estimates.

5 Analysis and Discussion

This section gives out more analysis of the experimental results. In addition, the performance of the proposed RCC with different settings is also analyzed.

5.1 Performance with Optimal SM Method

The immediate question for RCC is whether the ranking strategy is reasonable and what its performance upper bound is. To answer these questions, we consider the situation in which the k^{th} best SM method is always chosen for each input image. Fig. 4 illustrates the median angular error for different k. From the curves in Fig. 4, we obtain the following observations:

- If each image is always assigned the best (top 1) SM method, the median errors on both the sets are 0.44 and 1.06 degrees, which are much lower than the errors of the SOTA methods including those deep LM methods. It implies that it is effective to use a ranking scheme on the SM methods to achieve good performance for color constancy.
- Even though each image is assigned the top 4 or top 5 SM method, the performance is still comparable to that of most deep LM methods. It further indicates

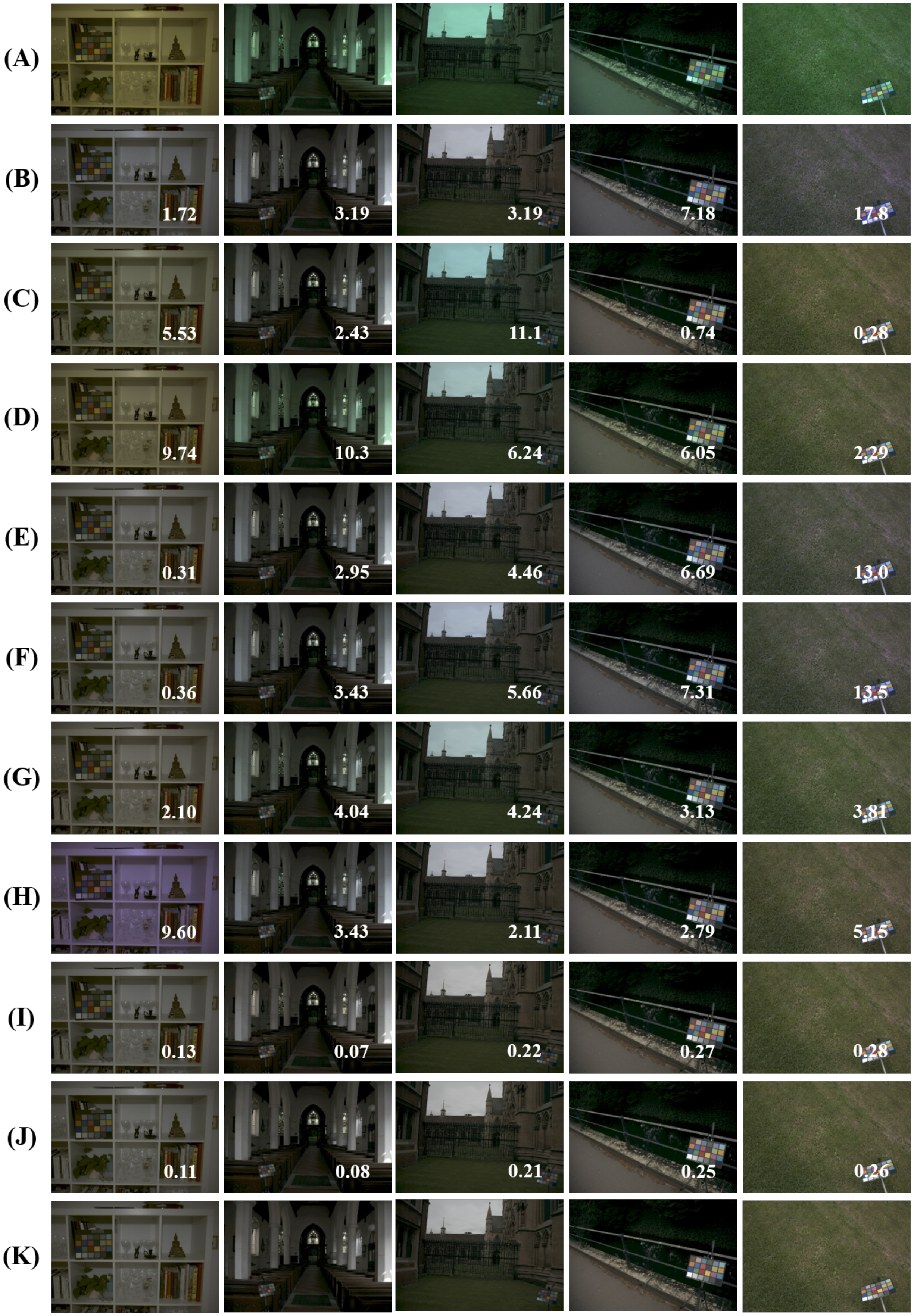


Fig. 3. Examples of images that are corrected under canonical light sources according to the estimated illumination by corresponding methods: (A) original image, (B) GW, (C) MaxRGB, (D) $\mathrm{GE}^{0,13,2}$, (E) $\mathrm{GE}^{1,1,6}$, (F)$\mathrm{GE}^{2,1,5}$, (G) SVR, (H)CNNM, (I) RCC, (J)RCC_C, (K) GT. The corresponding angular errors are shown in the lower right corner of the images.

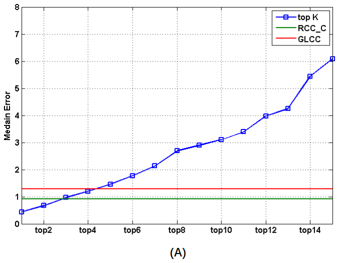


Fig. 4. Performance changes with the top-k SM method assignment: (A) on the Gehler-Shi set, (B) on the NUS set.

that the proposed ranking strategy is more reasonable than exclusively selecting the best SM method for an image.

- Compared with the ideal situation that each image is assigned the top 1 SM, the current RCC still has a performance room to improve. A more elaborated ranking model for RCC will be studied in the future.

5.2 Feature Selection

Feature selection is important for the proposed RCC. It is not known which features of an image are best for the selection of SM methods. The ℓ_{21}-norm constraints embedded in the proposed RCC ensure that effective features can be found automatically. The feature selection in RCC is based on grouped sparsity with ℓ_{21}-norm on $\mathbf{W}$, in which the ℓ_2-norm of each row of $\mathbf{W}$ is the weight of the corresponding feature. If the ℓ_2-norm of a row is equal or approximate to 0, the corresponding feature will be non-significant in the model. We visualize the matrix $\mathbf{W}$ of RCC for the two image sets, as illustrated in Fig. 5. We can find:

- The feature selection in RCC discards some useless features so as to improve the accuracy and generalization of the ranking model.
- The visualizations on both the image sets (including 8 subsets of the NUS set) in Fig. 5 show that the distributions of ℓ_2-norm of entries in $\mathbf{W}$ on different image sets are highly similar. It shows that the low-level-initial-estimate features and the majority of high-level Weibull features have much higher correlations to the SM method selection. Because the low-level-initial-estimate features represent the relations among estimates using the SM candidates and the Weibull features pay more attention to the textures of an image.
- Based on the feature selection, we can add more new visual features into RCC in the future to explore more correlated features for color constancy.

In addition, we compare the performance of RCC and RCC_C with and without feature selection as shown in Table 3. It also shows the effectiveness of feature selection that reduces the median and trimean errors by more than 0.4 degrees on each set. The feature selection not only reduces the effects of useless features, but it also reduces the model size so as to improve the efficiency of training and testing.

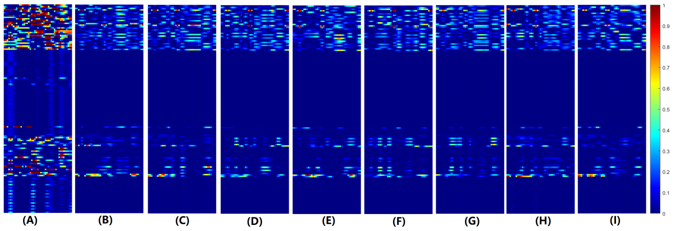


Fig. 5. The visualization of the $\mathbf{W}$ in RCC. (A) Gehler-Shi set, (B)-(I) 8 subsets of the NUS set.

TABLE 3
Performance comparison between RCC/RCC_C with and without feature selection.

Data set	Methods	Feature Selection	Mean	Median	Trimean
Gehler-Shi	RCC	✓	**2.53**	**1.30**	**1.54**
		✗	2.87	1.77	1.99
	RCC_C	✓	**2.37**	**1.20**	**1.42**
		✗	2.52	1.62	1.83
NUS	RCC	✓	**2.70**	**1.81**	**2.03**
		✗	3.86	2.43	2.71
	RCC_C	✓	**2.62**	**1.43**	**1.76**
		✗	4.10	2.15	2.69

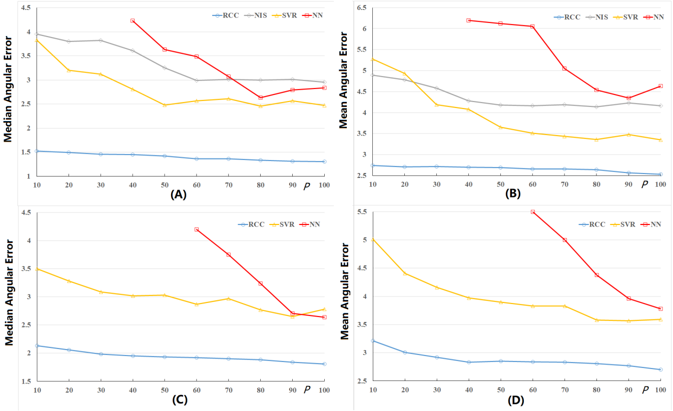


Fig. 6. Experimental results with limited training samples: (A)(B) median and mean errors change as the functions of p on the Gehler-Shi set; (C)(D) median and mean errors change as the functions of p the on NUS set.

5.3 Performance with Limited Training Samples

One of the advantages of RCC is that it can be trained with a limited number of labelled samples, which benefits from taking advantage of the correlations among different score functions with a low rank constraint as well as the feature selection. This section describes the experiments in which the number of training images is gradually decreased.

The cross validation settings on both the image sets are the same as those in Section 4. In each cross validation, the test set is fixed, but $p\%$ samples are randomly selected from the training set to comprise a new training set. The new training set is used to train the SVR, NN, NIS, and RCC methods. The procedure is repeated 3 times and the average performance is the final result for each method. Fig. 6 shows the median and mean errors with different values of p on two image sets. We find that:

- As the number of training samples is reduced, the median and mean errors of SVR, NN and NIS are increased rapidly on both the sets. The median errors of SVR, NN and NIS are increased by 55.1%, 48.9%, and 33.9% as p is decreased on the Gehler-Shi set. The median errors of SVR and NN are increased by 25.9% and 59.1% on the NUS set. In contrast, the median errors of RCC on both the sets are increased by only 16.9% and 17.6%, which are much lower than those of SVR, NN and NIS. Even when using only 10% training samples, the median errors of RCC still can achieve 1.52 and 2.13 degrees, which are much lower than most LM methods using 100% of the training samples, as shown in Tables 1 and 2. These results on both the sets verify the robustness of the proposed RCC when only a few training samples are used.
- Since the deep LM methods [13], [15], [16], [17] cannot be well trained on so small training sets, we use a shallow neural network architecture in [27] instead. The NN method has much fewer parameters than deep neural networks, but even so, it cannot produce a reasonable model when $p < 40$ and $p < 50$ on the two sets. It shows that the neural networks-based methods, either deep or shallow architecture, cannot be well trained if the number of training samples is small.
- The good robustness of RCC to only a few training samples is a result of the lower model complexity and of specific constraints embedded in the objective function. The model parameter size of RCC is much smaller than the model size of deep LM methods as well as many shallow LM methods, as discussed in detail in Section 5.5. In addition, both the low rank $\|\mathbf{W}\|_*$ and grouped sparsity $\|\mathbf{W}\|_{21}$ constraints in (7) take advantage of the correlation among prediction functions so as to improve the learning ability with a few training samples.

5.4 Comparisons among Ranking, Classification and Regression

RCC ranks the candidate SM methods for each image. This is in contrast with NIS, which uses a classifier to select the most appropriate SM for each image. In this experiment, the classification strategy and the ranking strategy are compared, using a Support Vector Machine as a classifier (CSVM) with the same SM candidate set to select an optimal SM for each image. In addition, experiments are also carried out to assess the performance of direct regression using the outputs of the SM methods. Based on the outputs of all the SM candidates, i.e. the low level feature $[\psi_1^{rg}(I), \psi_2^{rg}(I), ..., \psi_M^{rg}(I)]^T$, SVR estimates the illuminate value using regression (CSVR). For both the SVM and SVR, the Radial Basis Function (RBF) and linear kernels as well as corresponding parameters are selected by cross-validation in a training set for each image set. The performance of NIS, CSVM, CSVR, RCC, and RCC_C are shown in Table 4. From the results, we obtain the following observations:

TABLE 4
Performance comparison among ranking, classification and regression.

Data set	Method	Mean	Median	Trimean
Gehler-Shi	RCC	2.53	1.30	1.54
	RCC_C	**2.37**	**1.20**	**1.42**
	NIS [18]	4.40	3.18	3.49
	CSVM	2.99	1.83	1.96
	CSVR	2.67	2.06	2.20
NUS	RCC	2.70	1.81	2.03
	RCC_C	**2.62**	**1.43**	**1.76**
	NIS [18]	3.71	2.60	2.84
	CSVM	2.96	2.08	2.35
	CSVR	3.20	1.97	2.25

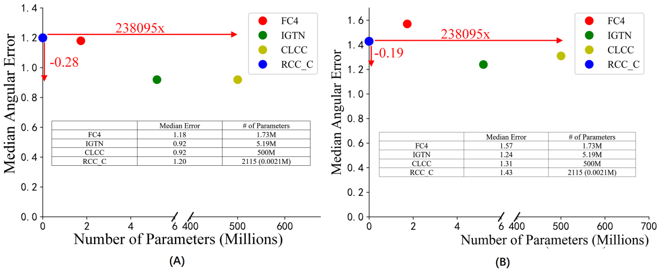
- Both RCC and RCC_C outperform CSVM on both the image sets, thus verifying the superiority of ranking compared with classification. CSVM classifies each image using 15 classes corresponding to 15 SM methods. Therefore, it has to face three potential difficulties: effective image features, limited training set, and imbalanced training samples. Fortunately, the proposed RCC method effectively avoids these issues well. The feature selection embedded in RCC selects effective ones from a number of image features during the training procedure. The low rank constraint in RCC takes advantage of the correlation among different prediction functions to improve its robustness to maintain performance if there are only a few training samples. Furthermore, the ranking does not assign only one label to each image but assigns different scores for the labels, which reduces the effect of the any imbalances in the numbers of the training samples. Consequently, RCC achieves much better performance than these classification-based methods.
- The direct regression method, CSVR, has comparable performance to CSVM, but the performance is much lower than that of both RCC and RCC_C. Since each SM method always depends on a specific assumption and it is difficult for an image to satisfy all the assumptions of 15 candidate SM methods. Some SM methods inevitably gives out inaccurate estimates that result in inaccurate final estimate for CSVR. In contrast, RCC_C first ranks the candidate SM methods and selects the top 3 SM methods to compute the final illumination using the weighted average. It effectively discards the inaccurate estimates to obtain more stable and accurate estimation than the direct regression method.

5.5 Comparison with Deep LM Methods

In this section, we compare RCC with recent deep LM methods in terms of accuracy, complexity and efficiency.

(1) Accuracy and Complexity

Fig. 7 illustrates the median angular errors and parameter numbers of deep LM methods and RCC. Although both IGTN and GLCC achieve slightly lower median errors than RCC_C, their parameter sizes reach to 5.19M and 500M that are nearly 2,454 and 238,095 times more than that of RCC (only $(30 + 24 + 87) \times 15 = 2115$ parameters). Even the lightest CNN-based method FC^4 still has 1.73M parameters, nearly 1000 times more than RCC, but it only



	Median Error	# of Parameters
FC4	1.18	1.73M
IGTN	0.92	5.19M
CLCC	0.92	500M
RCC_C	1.20	2115 (0.0021M)

(A)

	Median Error	# of Parameters
FC4	1.57	1.73M
IGTN	1.24	5.19M
CLCC	1.31	500M
RCC_C	1.43	2115 (0.0021M)

(B)

Fig. 7. Model complexity versus median angular error: (A) Gehler-Shi set, (B) NUS set

TABLE 5
Computation time (including feature extraction) in seconds for FC^4 and RCC.

	Training	Test (per image)	CPU	GPU
FC^4	99,000 (761times)	0.23(2.6times)	✓	✓
	309,600 (2,381times)	0.75 (8.3times)	✓	✗
RCC	130	0.09	✓	✗

achieves comparable median error to RCC_C on the Gehler-Shi set and a larger median error than RCC_C on the NUS set. So large scale of parameters in the deep LM methods need high-performance computing and storage resources. Conversely, RCC with only 2115 parameters can be trained and deployed efficiently on low-power terminal devices.

(2) Efficiency

To give an idea of the computational costs of the proposed RCC, we compare it to the lightest CNN-based method FC^4. Table 5 lists the training and test times for both RCC and FC^4 on the Gehler-Shi set using one cross-validation procedure with 379 training images and 179 test images. RCC is run using a PC with Intel Core i7-7700 at 3.6GHz and 8 GB RAM. The FC^4 is trained and tested by using a NVIDIA GeForce RTX 3090 GPU. To extend the comparisons, we also test FC^4 using the same CUP as RCC without a GPU. Since RCC needs to extract features, the feature extraction time per image in RCC is also given in Table 6. From Tables 5 and 6, we find that:

- Even though FC^4 is a light CNN-based method, it still takes 110s/epoch and 344s/epoch with and without GPU, respectively. The code provided by [15], [65] suggests 900 epochs for training a reasonable FC^4 model. Consequently, the training time for FC^4 with and without GPU is 99,000s (more than 1 day) and 309,600s (more than 3.5 days); the test time for FC^4 in the two situations is 0.23s and 0.75s. The training and test times (including feature extraction) of RCC are only 130s and 0.09s. If only using CPU, FC^4 should spend 2,381 and 8.3 times the training time and test time of RCC. It indicates that the complex deep LM methods cannot be directly deployed and run on a low-resource real-time device.
- The test time per image for RCC without feature extraction is only 3×10^{-6}s. By adding the running time of feature extraction, the inference time per image for RCC requires only 0.09s.

TABLE 6
Feature extraction time in seconds per image.

Low-level Feature	High-level Feature
0.06	0.03

TABLE 7
Performance of RCC with different candidate sets.

Data set	Methods	Mean	Median	Trimean
Gehler-Shi	RCC#	**2.46**	**1.27**	**1.48**
	RCC	2.53	1.30	1.54
	RCC_C#	**2.35**	**1.13**	**1.41**
	RCC_C	2.37	1.20	1.42
NUS	RCC#	**2.65**	**1.78**	**1.96**
	RCC	2.70	1.81	2.03
	RCC_C#	**2.48**	**1.28**	**1.64**
	RCC_C	2.66	1.58	1.91

5.6 Performance with Shallow LM Candidates

The candidate set $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$ in RCC is composed of SM methods, which are simple and have no training stage. In fact, some efficient shallow LM methods can also be included in $\boldsymbol{\psi}$. This experiment adds three effective shallow LM methods, including SVR, SF, and FFCC, into the candidate set $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$. The shallow LM methods are trained using the training set in each cross-validation. Table 7 shows the results of RCC with new candidate set $\boldsymbol{\psi}$ (denoted as RCC# and RCC_C#), using the same experimental settings on the two image sets. For comparison, results of RCC and RCC_C are also listed in Table 7. Both RCC# and RCC_C# obtain performance improvements on the two sets, compared with RCC and RCC_C. This implies that including better candidate methods into $\boldsymbol{\psi}$ can improve the performance of RCC. On the Gehler-Shi set, the median error of RCC_C# even achieves 1.13 degrees, which is slightly lower than the median error of RCC_C and outperforms all the methods listed in Table 1 except IGTN and GLCC. As a general framework, the proposed RCC method can be easily extended to incorporate new SM and LM methods that might be proposed in the future.

5.7 Performance across Image Sets

Generalization is important for a color constancy method. It is expensive and time-consuming if we always have to re-train a model from the ground up for any new imaging system. This experiment tests the generalization of RCC using images from different cameras for training and testing. We first use the NUS set as a training set and the Gehler-Shi set as a test set, then swap them. The performance of RCC and RCC_C methods crossing image sets is listed in Table 8. For comparison, the representative LM method SVR [11] is also tested crossing the image sets as a baseline. From Table 8, we obtain the following observations:

- The proposed RCC has much better generalization than SVR. When using the NUS as a training set and the Gehler-Shi as a test set, RCC and RCC_C still achieve 1.77 and 1.68 degrees median errors, while the median error of SVR is 7.66 degrees. When using the Gehler-Shi as the training set and the NUS as the test set, the median errors of RCC and RCC_C are

TABLE 8
Performance of different methods across image sets

Training set	Test Set	Methods	Mean	Median	Trimean
NUS	Gehler-Shi	RCC	3.08	1.77	2.11
		RCC_C	**2.99**	**1.68**	**2.00**
		SVR [11]	8.87	7.66	7.80
Gehler-Shi	NUS	RCC	3.83	2.40	2.77
		RCC_C	**3.29**	**2.15**	**2.49**
		SVR [11]	8.08	6.75	6.90

2.40 and 2.15 degrees, which are much lower than SVR's median error 6.75 degrees. The results show that SVR cannot learns a reasonable illumination prediction model due to the data distribution difference between the two image sets. However, RCC and RCC_C still perform well when crossing from one image set to the other. According to Tables 1, 2, and 8, on both the image sets, excepting SF, CCC and FFCC, RCC_C with crossing the image sets still outperforms all the SM and shallow LM methods that are trained and tested on the same image set.

- The performance reductions of RCC are much lower than those of SVR. On comparing the performance of RCC and RCC_C trained and tested within the same set in Tables 1 and 2, their median errors using crossing sets settings only increase by 0.44 and 0.46 degrees on the Gehler-Shi set. The two errors also only increase by 0.59 and 0.72 degrees on the NUS set. In contrast, the median errors of SVR using different settings increase by 4.43 and 4.57 degrees on both the sets. These increases are nearly 10 times those of RCC and RCC_C. The good generalization embedded in the proposed RCC method makes it easy to transfer from one ISP to another with few, even no, additional training samples.
- The underlying reasons for the good generalization of RCC can be summarized as two aspects: (1) Most LM methods, such as SVR, NN, BCC, directly predict illumination using visual features. In the crossing sets setting, the feature distribution gap between different sets inevitably results in a large performance reduction. Different from direct illumination prediction, RCC selects the SM method from the candidate set and applies the selected SM to test images, which effectively reduces the sensitivity to feature distribution differences. (2) The images in the two sets are taken using different ISPs. Nearly all the LM methods use color features that are highly correlated to ISPs, while most of the high-level visual features used in RCC are relatively stable across ISPs, which further improves the generalizations and robustness of RCC.

6 Extended RCC without Ground Truth Illumination

In the training stage of RCC, the ranking list for each training sample is computed based on its ground truth illumination. However, it is costly and time-consuming to measure the ground truth illumination of each image accurately. In order to reduce the dependence on the ground truth, we further extend the proposed RCC to a new ranking model without ground truth illumination (RCC_NO). The RCC_NO learns a ranking model using simple binary preference annotations provided by untrained annotators rather than the full ranking list based on ground truth illumination.

6.1 Formulation of RCC_NO

RCC_NO does not have a full ranking list for each image because of the lack of ground truth. It is formulated as: Given N training images $I_1, ..., I_N$ with corresponding feature vectors $x_1, ..., x_N$ and a set of M candidate SM methods $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$, each image is corrected as M images under canonical illumination using the estimated illumination by the M SM methods in $\boldsymbol{\psi}$. Then Q untrained annotators are asked to select at most O ($O < M$) images that have better white balancing according to their personal visual perceptions, as shown in Fig. 8. Consequently, we obtain a binary label matrix $\mathbf{A} \in \{0,1\}^{N\times Q\times M}$ instead of the SM ranking list L_i in RCC. The element $\mathbf{A}_{i,q,k}$ represents the tag of the q^{th} annotator on the k^{th} SM method for the i^{th} training image (1: the annotator prefers this SM method for this image, 0: otherwise). Considering Q annotators, the error $\varepsilon_{j,k}(x_i)$ in (3), which measures the error in ranking ψ_j and ψ_k for image I_i, is rewritten as:

$$\varepsilon_{j,k}(x_i) = \frac{1}{Q}\sum_{q=1}^{Q} \mathbf{1}_{\mathbf{A}_i}(\mathbf{A}_{i,q,j} \neq \mathbf{A}_{i,q,k})\delta(g_j(x_i) - g_k(x_i)), \tag{24}$$

where $\mathbf{1}_{\mathbf{A}_i}(\mathbf{A}_{i,q,j} \neq \mathbf{A}_{i,q,k})$ is an indicator function defined as:

$$\mathbf{1}_{\mathbf{A}_i}(\mathbf{A}_{i,q,j} \neq \mathbf{A}_{i,q,k}) = \begin{cases} 1, & \mathbf{A}_{i,q,j} \neq \mathbf{A}_{i,q,k} \\ 0, & \mathbf{A}_{i,q,j} = \mathbf{A}_{i,q,k} \end{cases}. \tag{25}$$

The overall loss function $J(\mathbf{W})$ in (5) of N training images labeled by Q annotators can be rewritten as:

$$\begin{aligned} J(\mathbf{W}) = \tfrac{1}{NQ}\sum_{i=1}^{N}\sum_{q=1}^{Q}\sum_{j,k}^{M} & \mathbf{1}_{\mathbf{A}_i}(\mathbf{A}_{i,q,j} \neq \mathbf{A}_{i,q,k}) \\ & \cdot\delta((\mathbf{A}_{i,q,j} - \mathbf{A}_{i,q,k})(\mathbf{w}_j^T x_i - \mathbf{w}_k^T x_i)). \end{aligned} \tag{26}$$

We substitute the new loss function $J(\mathbf{W})$ into (8) and optimize it using Algorithm 1 in Section 3.2.

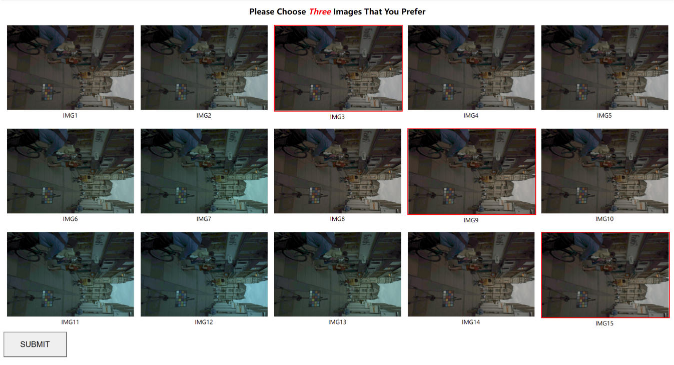
Compared with RCC, RCC_NO has the following advantages:

- RCC_NO does not need accurate illumination values of training samples. It is much easier and more efficient to collect more annotated training samples.
- RCC_NO does not require color checker in the training images for illumination measurement. A training set can be constructed by taking images of different scenes or by collecting images from the internet.
- It is more reasonable to evaluate a SM method using human visual perception rather than the mathematical angular error [62].

6.2 Training of RCC_NO

The training procedure of RCC_NO can be divided in two steps: (1) training sample annotation, and (2) model training.

(1) Training sample annotation. For RCC_NO, we keep the same candidate set $\boldsymbol{\psi}$ as that in Section 4.1. Three



Please Choose *Three* Images That You Prefer

Fig. 8. An annotation system for RCC_NO.

students in our lab without color constancy background knowledge in our lab are asked to annotate images in the two image sets. Fig. 8 illustrates the annotation system of RCC_NO. For each image, we adjust the image colors under canonical light sources according to the estimated illumination by each of the 15 candidate SM methods with the diagonal transformation [64]. The three images that have the best white balancing are selected out by an annotator according to his personal perception. The labels of the corresponding SM methods are set as 1, otherwise 0.

(2) Model training. After sample annotation, we can obtain the binary label matrix $\mathbf{A} \in \{0,1\}^{N\times Q\times M}$ for N training images $I_1, ..., I_N$ with corresponding feature vectors $x_1, ..., x_N$ and the SM method set $\boldsymbol{\psi} = \{\psi_1, \psi_2, ..., \psi_M\}$. Then we optimize $\mathbf{W}$ in (8) for the RCC_NO model using Algorithm 1 by substituting the new loss function $J(\mathbf{W})$ in (26) into (8).

6.3 Results of RCC_NO

We also test RCC_NO on both the Gehler-Shi and the NUS image sets, and compare it with RCC and other leading methods. The combining strategy in (21) is also used in RCC_NO and denoted as RCC_NO_C. The experimental settings are the same as those in Section 4. The results are given in Table 9. According to the comparison in Table 9, we find that:

- Both RCC_NO and RCC_NO_C outperform all the SM and most of the shallow LM methods on the two image sets.This indicates that good performance can be obtained using an effective ranking model without ground truth illumination. Consequently, it makes collecting training samples for an ISP much easier.
- The performance of RCC_NO and RCC_NO_C is lower than those of RCC and RCC_C on the two image sets. This implies that the binary annotation does lose some useful relationship information. The inconsistency between the human annotation and raking list based on ground truth illumination also pulls the performance of RCC_NO down.
- According to the differences between RCC and RCC_NO, the human white balancing preferences are slightly different from mathematical angular errors. Therefore, RCC_NO can be easily extended to the personalized color constancy in the future.

TABLE 9
Performance comparison between RCC_NO and RCC on two image sets.

Data set	Methods	Mean	Median	Trimean
Gehler-Shi	RCC_NO	2.94	1.61	1.88
	RCC	**2.53**	**1.30**	**1.54**
	RCC_NO_C	2.51	1.31	1.52
	RCC_C	**2.37**	**1.20**	**1.42**
NUS	RCC_NO	4.38	2.88	3.16
	RCC	**2.70**	**1.81**	**2.03**
	RCC_NO_C	4.04	2.04	2.64
	RCC_C	**2.62**	**1.43**	**1.76**

7 Conclusions

This paper presents a novel ranking-based color constancy method that achieves comparable performance to deep LM methods with much lower model complexity. The main idea is that, if we can select the optimal method from a set of SM methods for each image, we can achieve significant performance improvements. A novel ranking-based color constancy method is proposed to take advantages of label ranking scheme to predict the order of the candidate SM methods for each image. In order to reduce dependence on a large number of training samples and improve the efficiency, we introduce the low-rank constraint to control the model complexity and the group sparse constraint for feature selection. For furthermore removing the requirements of illumination measurements for color constancy, RCC is extended to a novel ranking-based method without ground truth illumination (RCC_NO). A large number of experimental results show that the simple RCC achieves comparable (even better) performance to most complex deep LM methods with much lower computational costs and fewer training samples. The work in this paper indicates that designing an effective and practical ranking strategy using simple SM methods can achieve better color constancy performance.

Acknowledgments

This work is partly supported by the National Key R&D Plan (No. 2020AAA0106800), the Natural Science Foundation of China (Nos.62122086, U1936204). Bing Li is also supported by Youth Innovation Promotion Association, CAS.

References

[1] K. Van De Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1582–1596, 2009.

[2] K. Barnard, V. Cardei, and B. Funt, "A comparison of computational color constancy algorithms. i: Methodology and experiments with synthesized data," *IEEE transactions on Image Processing*, vol. 11, no. 9, pp. 972–984, 2002.

[3] K. Barnard, L. Martin, A. Coath, and B. Funt, "A comparison of computational color constancy algorithms. ii. experiments with image data," *IEEE transactions on Image Processing*, vol. 11, no. 9, pp. 985–996, 2002.

[4] A. Gijsenij, T. Gevers, and J. Van De Weijer, "Computational color constancy: Survey and experiments," *IEEE transactions on image processing*, vol. 20, no. 9, pp. 2475–2489, 2011.

[5] B. Li, W. Xiong, W. Hu, B. Funt, and J. Xing, "Multi-cue illumination estimation via a tree-structured group joint sparse representation," *International Journal of Computer Vision*, vol. 117, no. 1, pp. 21–47, 2016.

[6] B. Xu, J. Liu, X. Hou, B. Liu, and G. Qiu, "End-to-end illuminant estimation based on deep metric learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3616–3625.

[7] S. Bianco and C. Cusano, "Quasi-unsupervised color constancy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 212–12 221.

[8] E. H. Land, "The retinex theory of color vision," *Scientific american*, vol. 237, no. 6, pp. 108–129, 1977.

[9] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin institute*, vol. 310, no. 1, pp. 1–26, 1980.

[10] J. Van De Weijer, T. Gevers, and A. Gijsenij, "Edge based color constancy," *IEEE Transactions on image processing*, vol. 16, no. 9, pp. 2207–2214, 2007.

[11] W. Xiong and B. Funt, "Estimating illumination chromaticity via support vector regression," *Journal of Imaging Science and Technology*, vol. 50, no. 4, pp. 341–348, 2006.

[12] D. H. Brainard and W. T. Freeman, "Bayesian color constancy," *JOSA A*, vol. 14, no. 7, pp. 1393–1411, 1997.

[13] S. Bianco, C. Cusano, and R. Schettini, "Color constancy using cnns," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 81–89.

[14] S. Bianco, C. Cusano, and R. a. Schettini, "Single and multiple illuminant estimation using convolutional neural networks," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4347–4362, 2017.

[15] Y. Hu, B. Wang, and S. Lin, "Fc4: Fully convolutional color constancy with confidence-weighted pooling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4085–4094.

[16] W. Shi, C. C. Loy, and X. Tang, "Deep specialized network for illuminant estimation," in *European conference on computer vision*. Springer, 2016, pp. 371–387.

[17] S. W. Oh and S. J. Kim, "Approaching the computational color constancy as a classification problem through deep learning," *Pattern Recognition*, vol. 61, pp. 405–416, 2017.

[18] A. Gijsenij and T. Gevers, "Color constancy using natural image statistics and scene semantics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 687–698, 2010.

[19] L. Shi, "Re-processed version of the gehler color constancy dataset of 568 images," *http://www. cs. sfu. ca/˜ color/data/*, 2000.

[20] F. Ciurea and B. Funt, "A large image database for color constancy research," in *Color and Imaging Conference*, vol. 2003, no. 1. Society for Imaging Science and Technology, 2003, pp. 160–164.

[21] "Color constancy," Website, 2022, http://colorconstancy.com/.

[22] D. A. Forsyth, "A novel algorithm for color constancy," *International Journal of Computer Vision*, vol. 5, no. 1, pp. 5–35, 1990.

[23] B. Li, W. Xiong, W. Hu, and B. Funt, "Evaluating combinational illumination estimation methods on real-world images," *IEEE Transactions on Image Processing*, vol. 23, no. 3, pp. 1194–1209, 2013.

[24] G. D. Finlayson and E. Trezzi, "Shades of gray and colour constancy," in *Color and Imaging Conference*, vol. 2004, no. 1. Society for Imaging Science and Technology, 2004, pp. 37–41.

[25] S.-B. Gao, K.-F. Yang, C.-Y. Li, and Y.-J. Li, "Color constancy using double-opponency," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 10, pp. 1973–1985, 2015.

[26] X.-S. Zhang, S.-B. Gao, R.-X. Li, X.-Y. Du, C.-Y. Li, and Y.-J. Li, "A retinal mechanism inspired color constancy model," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1219–1232, 2016.

[27] B. Li, D. Xu, W. Xiong, and S. Feng, "Color constancy using achromatic surface," *Color Research & Application*, vol. 35, no. 4, pp. 304–312, 2010.

[28] K.-F. Yang, S.-B. Gao, and Y.-J. Li, "Efficient illuminant estimation for color constancy using grey pixels," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2254–2263.

[29] D. Cheng, D. K. Prasad, and M. S. Brown, "Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution," *JOSA A*, vol. 31, no. 5, pp. 1049–1058, 2014.

[30] R. T. Tan, K. Nishino, and K. Ikeuchi, "Color constancy through inverse-intensity chromaticity space," *JOSA A*, vol. 21, no. 3, pp. 321–334, 2004.

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

[32] V. C. Cardei, B. Funt, and K. Barnard, "Estimating the scene illumination chromaticity by using a neural network," *JOSA a*, vol. 19, no. 12, pp. 2374–2386, 2002.

[33] G. D. Finlayson, S. D. Hordley, and P. M. Hubel, "Color by correlation: A simple, unifying framework for color constancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1209–1221, 2001.

[34] A. Chakrabarti, K. Hirakawa, and T. Zickler, "Color constancy with spatio-spectral statistics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1509–1519, 2011.

[35] A. Gijsenij, T. Gevers, and J. Van De Weijer, "Generalized gamut mapping using image derivative structures for color constancy," *International Journal of Computer Vision*, vol. 86, no. 2, pp. 127–139, 2010.

[36] G. D. Finlayson, "Corrected-moment illuminant estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1904–1911.

[37] D. Cheng, B. Price, S. Cohen, and M. S. Brown, "Effective learning-based illuminant estimation using simple features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1000–1008.

[38] J. T. Barron, "Convolutional color constancy," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 379–387.

[39] J. T. Barron and Y.-T. Tsai, "Fast fourier color constancy," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 886–894.

[40] J. Vazquez-Corral, M. Vanrell, R. Baldrich, and F. Tous, "Color constancy by category correlation," *IEEE Transactions on image processing*, vol. 21, no. 4, pp. 1997–2007, 2011.

[41] H. R. V. Joze and M. S. Drew, "Exemplar-based color constancy and multiple illumination," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 860–873, 2013.

[42] S. Bianco and R. Schettini, "Color constancy using faces," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 65–72.

[43] S. Bianco, G. Ciocca, C. Cusano, and R. a. Schettini, "Automatic color constancy algorithm selection and combination," *Pattern recognition*, vol. 43, no. 3, pp. 695–705, 2010.

[44] S. Bianco, G. Ciocca, C. Cusano, and R. Schettini, "Improving color constancy using indoor outdoor image classification," *IEEE Transactions on image processing*, vol. 17, no. 12, pp. 2381–2392, 2008.

[45] R. Lu, A. Gijsenij, T. Gevers, V. Nedović, D. Xu, and J.-M. Geusebroek, "Color constancy using 3d scene geometry," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 1749–1756.

[46] J. Van De Weijer, C. Schmid, and J. Verbeek, "Using high-level visual information for color constancy," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[47] Y.-C. Lo, C.-C. Chang, H.-C. Chiu, Y.-H. Huang, C.-P. Chen, Y.-L. Chang, and K. Jou, "Clcc: Contrastive learning for color constancy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8053–8063.

[48] A. Abdelhamed, A. Punnappurath, and M. S. Brown, "Leveraging the availability of two cameras for illuminant estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6637–6646.

[49] D. Hernandez-Juarez, S. Parisot, B. Busam, A. Leonardis, G. Slabaugh, and S. McDonagh, "A multi-hypothesis approach to color constancy," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2270–2280.

[50] J. Xiao, S. Gu, and L. Zhang, "Multi-domain learning for accurate and few-shot color constancy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3258–3267.

[51] S. Vembu and T. Gärtner, "Label ranking algorithms: A survey," in *Preference learning*. Springer, 2010, pp. 45–64.

[52] G. Zhu, S. Yan, and Y. Ma, "Image tag refinement towards low-rank, content-tag prior and error sparsity," in *Proceedings of the 18th ACM international conference on Multimedia*, 2010, pp. 461–470.

[53] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE transactions*

on pattern analysis and machine intelligence, vol. 35, no. 1, pp. 171–184, 2012.

[54] X.-T. Yuan, X. Liu, and S. Yan, "Visual classification with multitask joint sparse representation," *IEEE Transactions on Image Processing*, vol. 21, no. 10, pp. 4349–4360, 2012.

[55] Y. Nesterov, "Gradient methods for minimizing composite functions," *Mathematical programming*, vol. 140, no. 1, pp. 125–161, 2013.

[56] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *submitted to SIAM Journal on Optimization*, vol. 2, no. 3, 2008.

[57] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," *Advances in neural information processing systems*, vol. 24, 2011.

[58] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[59] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient l2, 1-norm minimization," *arXiv preprint arXiv:1205.2631*, 2012.

[60] J.-M. Geusebroek and A. W. Smeulders, "A six-stimulus theory for stochastic texture," *International Journal of Computer Vision*, vol. 62, no. 1, pp. 7–16, 2005.

[61] D. L. Ruderman, T. W. Cronin, and C.-C. Chiao, "Statistics of cone responses to natural images: implications for visual coding," *JOSA A*, vol. 15, no. 8, pp. 2036–2045, 1998.

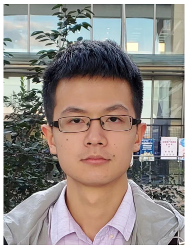
[62] A. Gijsenij, T. Gevers, and M. P. Lucassen, "Perceptual analysis of distance measures for color constancy algorithms," *JOSA A*, vol. 26, no. 10, pp. 2243–2256, 2009.

[63] S. D. Hordley and G. D. Finlayson, "Re-evaluating colour constancy algorithms," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 1. IEEE, 2004, pp. 76–79.

[64] J. Von Kries, "Influence of adaptation on the effects produced by luminous stimuli," *handbuch der Physiologie des Menschen*, vol. 3, pp. 109–282, 1905.

[65] Y. Hu, "The homepage of fc4," Website, 2022, https://github.com/yuanming-hu/fc4.

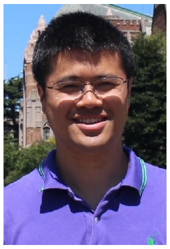
Bing Li received the Ph.D. degree from the Department of Computer Science and Engineering, Beijing Jiaotong University, Beijing, China, in 2009. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences, Beijing. His current research interests include video understanding, color constancy, visual saliency, multi-instance learning, and Web content security.



Haina Qin received the bachelor's degree in software engineering from the Dalian University of Technology in 2016. He is currently pursuing the Ph.D. degree in pattern recognition and intelligent system with the Institute of Automation, Chinese Academy of Sciences. His research interests include color constancy and image processing.

Weihua Xiong received the Ph.D. degree from the Department of Computer Science, Simon Fraser University, Canada, in 2007. His research interests include color science, computer vision, color image processing, and stereo vision.

Yangxi Li received the Ph.D. degree from Peking University. He is currently a Senior Engineer with the National Computer Network Emergency Response Technical Team/Coordination Center of China. His research interests lie primarily in multimedia search, information retrieval, and computer vision.



Songhe Feng received the Ph.D. degree in the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, P.R. China, in 2009. He is currently a Full Professor in the School of Computer and Information Technology, Beijing Jiaotong University. His research interests include computer vision and machine learning.

Weiming Hu received the Ph.D. degree from the department of computer science and engineering, Zhejiang University in 1998. From April 1998 to March 2000, he was a postdoctoral research fellow with the Institute of Computer Science and Technology, Peking University. Now he is a professor in the Institute of Automation, Chinese Academy of Sciences. His research interests are in visual motion analysis, recognition of web objectionable information, and network intrusion detection.

Stephen Maybank received a BA in Mathematics from King's College Cambridge in 1976 and a PhD in computer science from Birkbeck college, University of London in 1988. Now he is a professor emeritus in the Department of Computer Science and Information Systems, Birkbeck College. His research interests include the geometry of multiple images, camera calibration, visual surveillance etc.