

Effective Learning-Based Illuminant Estimation Using Simple Features

Dongliang Cheng¹

Brian Price²

Scott Cohen²

Michael S. Brown¹

¹National University of Singapore

{dcheng, brown}@comp.nus.edu.sg

²Adobe Research

{bprice, scohen}@adobe.com

Abstract

Illumination estimation is the process of determining the chromaticity of the illumination in an imaged scene in order to remove undesirable color casts through white-balancing. While computational color constancy is a well-studied topic in computer vision, it remains challenging due to the ill-posed nature of the problem. One class of techniques relies on low-level statistical information in the image color distribution and works under various assumptions (e.g. Grey-World, White-Patch, etc). These methods have an advantage that they are simple and fast, but often do not perform well. More recent state-of-the-art methods employ learning-based techniques that produce better results, but often rely on complex features and have long evaluation and training times. In this paper, we present a learning-based method based on four simple color features and show how to use this with an ensemble of regression trees to estimate the illumination. We demonstrate that our approach is not only faster than existing learning-based methods in terms of both evaluation and training time, but also gives the best results reported to date on modern color constancy data sets.

1. Introduction and Related Work

An RGB image captured by a camera is a combination of three factors: 1) the scene’s spectral reflectance properties; 2) the spectral illumination of the scene; and 3) the camera RGB sensors spectral sensitivities. Assuming the scene is illuminated uniformly by a single illuminant, the image formation model takes the form:

$$\rho_c(\mathbf{x}) = \int_{\lambda \in \Omega} E(\lambda) R(\lambda, \mathbf{x}) S_c(\lambda) d\lambda \quad c \in \{R, G, B\}, \quad (1)$$

where each channel (Red, Green, Blue) at pixel location \mathbf{x} is an integrated signal resulting from the camera’s sensitivity $S_c(\lambda)$, the spectral scene content $R(\lambda, \mathbf{x})$, and the scene illumination $E(\lambda)$ over the visible spectrum Ω . From Eq. 1 it is clear that the RGB colors are biased by the color of the

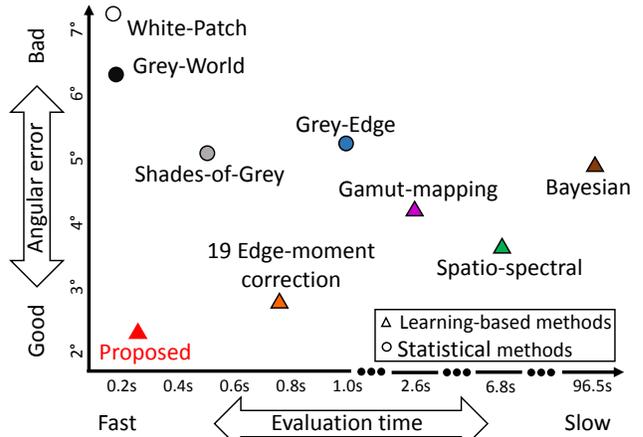


Figure 1: Evaluation time vs. performance of representative illuminant estimation methods. Statistics-based methods are fast but have lower accuracy than learning-based methods. The slow speed of learning-based methods makes them impractical for onboard camera white-balancing. Our proposed learning-based method achieves high accuracy and fast evaluation. (Mean angular error and time statistics for this plot are based results in Table 1 and Table 3). Note time axis is nonlinear.

illumination. When the illumination is not sufficiently white (e.g. daylight), this can cause a notable color cast in the image. One of the key pre-processing steps applied to most images is to remove color casts caused by illumination to improve an image’s aesthetics and to aid in the performance of various color-based computer vision applications.

Scene illumination can be modeled as a direction in the camera’s RGB color space [17]. Based on the estimated illumination, the image colors are transformed such that the scene illumination direction is mapped to lie along the achromatic white-line (i.e. $[0, 0, 0]$ to $[1, 1, 1]$), thus making the illumination ‘white’ and therefore removing the color cast. The challenge of camera-based color constancy lies in estimating the illuminant color ($\rho_R^E, \rho_G^E, \rho_B^E$) defined as:

$$\rho_c^E = \int_{\lambda \in \Omega} E(\lambda) S_c(\lambda) d\lambda \quad c \in \{R, G, B\}. \quad (2)$$

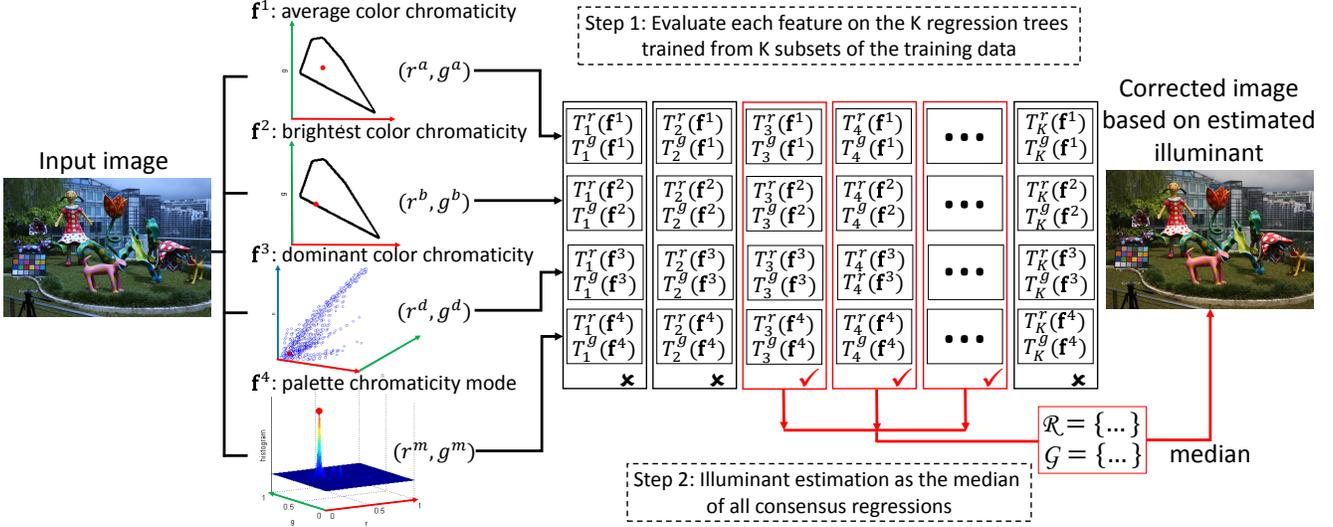


Figure 2: An overview of our proposed learning-based framework for illuminant estimation. Given an input image, we extract four features from the image (Sec. 2.1): 1) the average color chromaticity; 2) the brightest color chromaticity; 3) the dominant color chromaticity; 4) the mode of the color palette. For each feature, a bank of K regression trees is evaluated (Sec. 2.3). Each regression tree outputs a prediction of the illumination. The final result is estimated by combining the results of the regression trees via cross-feature consensus (Sec. 2.4).

Unfortunately, solving for ρ_c^E in Eq. 2 is ill-posed. Even when neglecting the integral and assuming a diagonal correction model [15], there are two unknowns at a pixel, reflectance and the illumination, with only one observation, namely ρ_c from Eq. 1.

Given the difficulty of this problem, there is a large body of literature on illumination estimation. These methods can be broadly classified into two categories: statistics-based methods and learning-based methods. A full literature review on these methods is outside the scope of this paper and only representative papers are discussed here. Readers are referred to an excellent survey [29] for more information.

Statistics-based methods examine the RGB color space to determine values correlated with the scene illumination [7, 9, 19, 36]. These methods include the well known Grey-World and White-Patch methods that make assumptions about the relationship between color statistics and achromatic colors. Other methods rely on the correlation of statistics from spatial derivatives or other frequency information in the image and the scene illumination [4, 5, 13, 26, 30, 39]. Recent works [22, 23] use statistics inspired from the human vision system (e.g. color opponency). Other methods examined scene content looking for physics-based insight to illumination, such as specularities and shadows [14, 32, 37]. Statistics-based methods remain popular because they are efficient to compute, however they do not always give the best performance.

Learning-based methods have shown to be more accurate in illumination estimation. The early gamut-based method [20] learned gamuts for different cameras and used this to constrain the solution space for an input image.

Chromaticity histograms have been used as an input feature for various learning-based methods [10, 18, 21, 33]. This was successfully extended to a full 3D RGB histogram used in a Bayesian framework [24, 34]. Several works incorporate derivative and frequency features into learning-based frameworks [5, 11, 26] to learn the expected distributions of spatio-statistics for different cameras. Recently, a data-driven method using a surface descriptor feature to match image segments was studied in [38]. While these methods give superior results compared to statistics-based methods, they are notably slower due to the complex features used and often have long training times. As a result, these methods are not suitable for applications requiring real-time performance. Figure 1 helps to illustrate this with a plot of various statistics-based and learning-based methods in terms of accuracy versus evaluation time.

Contribution The goal of this paper is to develop a learning-based illumination estimation method with a running-time of statistical methods. Our work is inspired in part by the recent successful method [16] that showed that relatively simple features (color/edge moments) could be used to give good performance in a learning-based framework. In this paper, we simplify the learning-based procedure further to use only four simple features. A key technical contribution of our paper is a method for training an ensemble of decision trees on these simple features that can accurately predict the chromaticity of the illumination. This method achieves our goal by producing the best results to date on a number of illumination data sets with a running-time on par with statistics-based methods.

2. Learning Illumination Estimation with Simple Features

An overview of our method is shown in Figure 2. Given an image, four features are extracted, each of these features is given to a bank of regression trees to generate many illuminant candidates. Results from the multiple regression trees that are in agreement are combined to estimate the illumination. The following subsections details each step of our procedure, including feature extraction, training of the regression trees, and forming the final consensus.

2.1. Image Features

Our approach uses only four features derived directly from the input image color distribution. Similar to prior approaches [10, 18, 21, 33], we use normalized chromaticity, rather than RGB color, as it is intensity invariant. This is useful requirement for illuminant estimation since two images related only by a scale factor (e.g. due to the exposure or light source energy difference) should have the same illuminant estimation. Chromaticity is calculated as:

$$\begin{cases} r = R/(R + G + B) \\ g = G/(R + G + B) \end{cases}, \quad (3)$$

where R , G and B are the camera Red, Green and Blue channel measurements, and r and g are the chromaticity values.

Our four features are as follows: (1) average color chromaticity, (2) brightest color chromaticity, (3) dominant color (mode of RGB histogram) chromaticity and (4) mode of chromaticity of the image color palette. Note that as with other illuminant estimation methods, the standard pre-processing to the input images is applied, namely black offset correction and the removal of saturated pixels.

Average color chromaticity (\mathbf{f}^1) is the chromaticity (r^a, g^a) of the average RGB value (R^a, G^a, B^a) where

$$C^a = \frac{1}{n} \sum_{i=1}^n C_i, \quad C \in \{R, G, B\}, \quad (4)$$

where n is the number of pixels in the image excluding saturated pixels.

Brightest color chromaticity (\mathbf{f}^2) is the chromaticity (r^b, g^b) of the color (R^b, G^b, B^b) of the pixel k which has the largest brightness ($R + G + B$):

$$\begin{aligned} (R^b, G^b, B^b) &= (R_k, G_k, B_k), \\ \text{where } k &= \arg \max_i (R_i + G_i + B_i). \end{aligned} \quad (5)$$

This differs from the maxRGB (i.e. White Patch) method that treats each RGB channels independently.

Dominant color chromaticity (\mathbf{f}^3) is the chromaticity (r^d, g^d) of the average RGB color (R^d, G^d, B^d) of the pixels belonging to a histogram bin which has the largest count

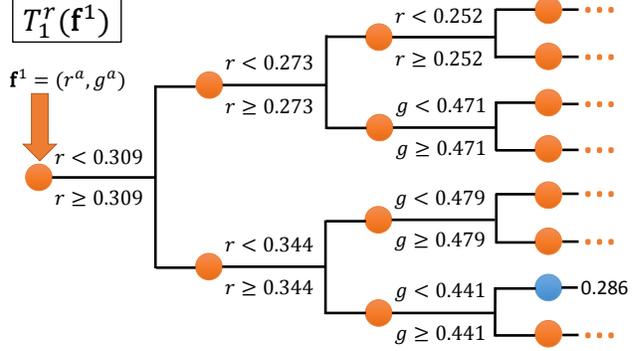


Figure 3: An example of the regression tree $T_1^r(\mathbf{f}^1)$ (the r -chromaticity illuminant prediction on feature 1 with the training subset 1). The orange dots denote non-leaf nodes where a decision is made according to the split rule. The blue dot denotes a leaf node where the final regression value is determined. This figure shows only the first four layers (out of 17) which already contain a leaf (end) node.

(i.e. the mode of the RGB histogram):

$$C^d = \frac{1}{|H_k|} \sum_{j \in H_k} C_j, \quad C \in \{R, G, B\}, \quad (6)$$

where $k = \arg \max_i |H_i|$,

where H_m is the set of pixels in the m^{th} bin of the histogram, $m \in [1, M]$. We used 128 bins per color channel (i.e. $M = 128^3$).

Chromaticity mode of the color palette (\mathbf{f}^4) is the mode of the image color palette in the chromaticity space. We construct the color palette by taking the average value of each bin in the RGB histogram that contains more than a predefined threshold of pixels. In our implementation, a threshold of 200 pixels per bin was used. This results in a palette of approximately 300 colors for a typical image. Each color in the palette is projected onto the normalized chromaticity plane, and an efficient 2D kernel density estimation (KDE) [6] is applied. The mode (r^m, g^m) is the chromaticity with the highest density. This feature is useful because it provides the mode of the chromaticity that is independent of the number of pixels of each color.

2.2. Regression Tree

Our learning-based method is based on variance reduction regression trees [8] that have been shown to be a powerful nonlinear predictive model that are both efficient for training and testing. In particular, for each feature a series of K regression trees is estimated. In our approach, regression trees are estimated in pairs, one for the r and g chromaticity. Thus to obtain an illumination estimate for a feature, we compute two regression trees $r_i^j = T_i^r(\mathbf{f}^j)$ and $g_i^j = T_i^g(\mathbf{f}^j)$, where i is the index of the regression tree,

\mathbf{f}^j (where $j = 1, 2, 3, 4$) represents the feature the regression tree is trained for, and superscripts r and g represent the chromaticity output respectively. For example, $T_1^r(\mathbf{f}^1)$ would mean the r chromaticity for the first regression tree for feature \mathbf{f}^1 . Each of the i trees are trained based on the data sampled more densely to particular region in the chromaticity space. This will be discussed in Section 2.3.

In the training stage, regression trees are obtained using a fast divide and conquer greedy algorithm that recursively partitions the given training data into two smaller subsets to minimize the sum of in-subset variances [8]:

$$\frac{1}{|S_1|} \sum_{p \in S_1} \sum_{q \in S_1} \|\mathbf{f}_p - \mathbf{f}_q\|^2 + \frac{1}{|S_2|} \sum_{p \in S_2} \sum_{q \in S_2} \|\mathbf{f}_p - \mathbf{f}_q\|^2, \quad (7)$$

where $\mathbf{f}_p, \mathbf{f}_q$ are input features of the training data and S_1, S_2 are the resulting split subsets.

After training, the regression tree works in a straightforward manner, where the tree nodes are evaluated starting from the root according to the rule learned by the optimal splitting point in the training stage until reaching a leaf node, where a regression output can be given. Figure 3 shows one real example of the regression tree from our training experiment.

2.3. Sampling for Multiple Trees

As mentioned in the previous section, our approach estimates K pairs of trees per feature. Each of these trees is computed from samples in the training data that are biased to a local region in chromaticity space of the ground truth illuminations. Figure 4 illustrates this sampling procedure where the ground truth illuminant chromaticity for the training images are plotted in the chromaticity space. The plotted illuminant follow the well-known quadratic shape of the Planckian locus of the black body radiance that is commonly used to describe the color temperature of natural and man-made illuminants.

Our method sorts the training data based on its r ground truth chromaticity to capture the relationship of the illuminations along the color temperature. The training data is then divided into K groups which have equal number of training images and overlap 50% with their neighbor groups. For each tree pair, the samples in their local regions are weighted K times more than the other samples in the training data when building the regression tree, thus biasing the result of the regression tree to the local region. We experimented with different number of trees and found that $K = 30$ provided good performance and computation efficiency. More details to this strategy versus alternative strategies are discussed in Section 4.

2.4. Tree Ensemble Consensus

The K trees per feature form an ensemble of decision trees from which the final results need to be estimated. We

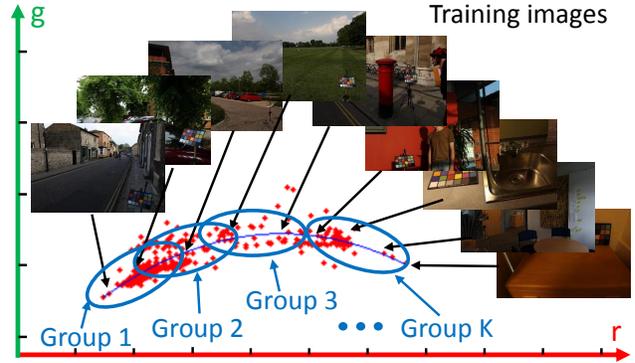


Figure 4: Illustration of sorting the training images and separate them into groups. The red dots indicate the ground truth illuminant rg -chromaticities from Gehler-Shi data set [24, 35] and the curve shows a quadratic fitting to these illuminant chromaticities. A number of training example images from the data set having different illuminant from blueish to reddish are also shown. The whole training data set is separated into K local overlapping groups.

expect that neighboring trees for the same feature will likely give similar results due to the training-data weighting in the tree construction. The power in the ensemble comes when the different features’ trees estimations are in agreement. To find this cross-feature consensus, we examine the output of all $r_i^j = T_i^r(\mathbf{f}^j)$ and $g_i^j = T_i^g(\mathbf{f}^j)$ for each $i \in [1, K]$ trees and $j \in \{1, 2, 3, 4\}$ features. When any three out of four features from trees from the same training data (i.e T_i) give output candidates within a 0.025 2D Euclidean distance to one another, we take all the output from those trees and add to the output sets, \mathcal{R} for r -chromaticity and \mathcal{G} for g -chromaticity (see Figure 2). The final estimated illuminant chromaticity is taken as:

$$\rho_{r,g} = (\text{median}(\mathcal{R}), \text{median}(\mathcal{G})), \quad (8)$$

where $\rho_{r,g}$ is the chromaticity of the estimated illuminant and median finds the median of the set. In the unlikely scenario that none of the K feature trees have any agreement, the result is computed as the median of *all* the trees’ outputs. We found this happens for only 2%-6% of the images for all the data sets tested.

2.5. Training and Testing

The training and testing of the proposed method follows the standard 3-fold cross validation of existing learning-based methods [29] common in the illuminant estimation literature. To do this, the whole data set is randomly divided into three sets and each time two sets are used for training while the remaining image set is used for testing. Parameters used for all experiments were selected based on the 1/3 subset from Gehler-Shi set [24, 35] and then are fixed for all other image sets and cameras. Better performance likely could be obtained with data set specific tuning.

Method		Mean	Median	Trimean	Best-25%	Worst-25%
Statistics-based	Grey-world [9]	6.36	6.28	6.28	2.33	10.58
	White-patch [31]	7.55	5.68	6.35	1.45	16.12
	Shades-of-grey [19]	4.93	4.01	4.23	1.14	10.20
	General Grey-world [39]	4.66	3.48	3.81	1.00	10.09
	1st-order Grey-edge [39]	5.33	4.52	4.73	1.86	10.03
	2nd-order Grey-edge [39]	5.13	4.44	4.62	2.11	9.26
	Bright-and-dark Colors PCA [12]	3.52	2.14	2.47	0.50	8.74
	Local Surface Reflectance [22]	3.31	2.80	2.87	1.14	6.39
Learning-based	Pixel-based Gamut [28]	4.20	2.33	2.91	0.50	10.72
	Edge-based Gamut [28]	6.52	5.04	5.43	1.90	13.58
	Intersection-based Gamut [28]	4.20	2.39	2.93	0.51	10.70
	SVR Regression [21]	8.08	6.73	7.19	3.35	14.89
	Bayesian [24]	4.82	3.46	3.88	1.26	10.49
	Spatio-spectral [11]	3.59	2.96	3.10	0.95	7.61
	CART-based Combination [5]	3.90	2.91	3.21	1.02	8.27
	Natural Image Statistics [26]	4.19	3.13	3.45	1.00	9.22
	Bottom-up+Top-down [40]	3.48	2.47	2.61	0.84	8.01
	Exemplar-based [38]	2.89	2.27	2.42	0.82	5.97
	19-Edge Corrected-moment [16]	2.86	2.04	2.22	0.70	6.34
	Our Proposed	2.42	1.65	1.75	0.38	5.87

Table 1: Performance comparison of our proposed learning-based method against various other methods on the Gehler-Shi data set [24, 35].

3. Experimental Results

We compare our proposed method against a large number of existing white-balance algorithms on three white-balance data sets (Gehler-Shi [24, 35], NUS 8-Camera [12], and SFU Laboratory Image Set [3]). Gehler-Shi and the NUS 8-Camera data sets represent modern white-balance images indicative of real world images and illuminations. The SFU Laboratory data set is an older data set of objects captured in a laboratory under often unusual lighting. It is included here for sake of completeness. For each data set, we give a summary of the performance statistics that is available and always include the best prior-art result known to us.

We use angular error (AE) (Equation. 9) as the error metric to evaluate the methods as it is most widely used in evaluating color constancy algorithms [29] and is correlated to the perceptual Euclidean distance [27]. The angular error $\varepsilon_{\text{angle}}(\mathbf{e}_{\text{est}})$ of the estimated illumination direction \mathbf{e}_{est} from the illumination direction of the ground truth \mathbf{e}_{gt} is computed as follows:

$$\varepsilon_{\text{angle}}(\mathbf{e}_{\text{est}}) = \cos^{-1} \left(\frac{\mathbf{e}_{\text{est}} \cdot \mathbf{e}_{\text{gt}}}{\|\mathbf{e}_{\text{est}}\| \|\mathbf{e}_{\text{gt}}\|} \right) \quad (9)$$

Unlike most of the previous work which examine only the mean and the median of the AEs, we provide a more thorough comparison with additional statistical metrics, including the tri-mean, the mean of the best 25% AEs and the mean of the worst 25% AEs, as done in [29].

3.1. Gehler-Shi Image Set

This data set was originally captured by Gehler *et al.* [24] with Shi and Funt [35] suggesting an updated linearization of the data set. This data set contains 568 images, the largest such data set to date, including a variety of indoor and outdoor scenes with challenging cases. The large number of images is sufficient to allow learning-based methods to work reasonably. In the experiments, a color-checker board has been inserted in the scene and used to compute the ground truth illumination. This color-checker board is masked out when the images are used for training and evaluation.

We compare against 19 previous methods as shown in Table 1. Most of the results from other methods have been evaluated by [24] or collected on the colorconstancy.com website [25] and we directly report them here. In order to have other statistical metrics for the 19-Edge Corrected-moment method other than just mean and median, we implemented the method as described in [16] and achieved a similar performance as reported by the author. Table 1 shows that our proposed method produces state-of-the-art results for all 5 metrics.

3.2. NUS 8-camera Image Set

The NUS 8-camera Image Set [12] is the most recent white-balance data set. It is composed of 1736 images from 8 commercial cameras. Like the Gehler-Shi data set, it provides linear images in RAW format and has a color-chart in each image for ground truth estimation. The data set is composed of more than 210 individual scenes, where each camera has photographed the same scene. This gives the oppor-

	Statistics-based methods								Learning-based methods							
Method	GW [9]	WP [7]	SoG [19]	GGW [39]	GE1 [39]	GE2 [39]	BD [12]	LSR [22]	PG [28]	EG [28]	BF [24]	SS [11]	NIS [26]	CM [16]	CD [1]	Ours
Camera	Mean angular error (degrees °)															
Canon1Ds	5.16	7.99	3.81	3.16	3.45	3.47	2.93	3.43	6.13	6.07	3.58	3.21	4.18	2.94	3.13	2.26
Canon600D	3.89	10.96	3.23	3.24	3.22	3.21	2.81	3.59	14.51	15.36	3.29	2.67	3.43	2.76	2.83	2.43
FujiXM1	4.16	10.20	3.56	3.42	3.13	3.12	3.15	3.31	8.59	7.76	3.98	2.99	4.05	3.23	3.36	2.45
NikonD5200	4.38	11.64	3.45	3.26	3.37	3.47	2.90	3.68	10.14	13.00	3.97	3.15	4.10	3.46	3.19	2.51
OlympEPL6	3.44	9.78	3.16	3.08	3.02	2.84	2.76	3.22	6.52	13.20	3.75	2.86	3.22	2.95	2.57	2.15
LumixGX1	3.82	13.41	3.22	3.12	2.99	2.99	2.96	3.36	6.00	5.78	3.41	2.85	3.70	3.10	2.84	2.36
SamNX2000	3.90	11.97	3.17	3.22	3.09	3.18	2.91	3.84	7.74	8.06	3.98	2.94	3.66	2.74	2.92	2.53
SonyA57	4.59	9.91	3.67	3.20	3.35	3.36	2.93	3.45	5.27	4.40	3.50	3.06	3.45	2.95	2.83	2.18
Camera	Median angular error (degrees °)															
Canon1Ds	4.15	6.19	2.73	2.35	2.48	2.44	2.01	2.51	4.30	4.68	2.80	2.67	3.04	1.98	1.72	1.57
Canon600D	2.88	12.44	2.58	2.28	2.07	2.29	1.89	2.72	14.83	15.92	2.35	2.03	2.46	1.85	1.85	1.62
FujiXM1	3.30	10.59	2.81	2.60	1.99	2.00	2.15	2.48	8.87	8.02	3.20	2.45	2.96	2.11	1.81	1.58
NikonD5200	3.39	11.67	2.56	2.31	2.22	2.19	2.08	2.83	10.32	12.24	3.10	2.26	2.40	2.04	1.94	1.65
OlympEPL6	2.58	9.50	2.42	2.18	2.11	2.18	1.87	2.49	4.39	8.55	2.81	2.24	2.17	1.84	1.46	1.41
LumixGX1	3.06	18.00	2.30	2.23	2.16	2.04	2.02	2.48	4.74	4.85	2.41	2.22	2.28	1.77	1.69	1.61
SamNX2000	3.00	12.99	2.33	2.57	2.23	2.32	2.03	2.90	7.91	6.12	3.00	2.29	2.77	1.85	1.89	1.78
SonyA57	3.46	7.44	2.94	2.56	2.58	2.70	2.33	2.51	4.26	3.30	2.36	2.58	2.88	2.05	1.77	1.48
Camera	Tri-mean error (degrees °)															
Canon1Ds	4.46	6.98	3.06	2.50	2.74	2.70	2.22	2.81	4.81	4.87	2.97	2.79	3.30	2.19	2.08	1.69
Canon600D	3.07	11.40	2.63	2.41	2.36	2.37	2.12	2.95	14.78	15.73	2.40	2.18	2.72	2.12	2.07	1.80
FujiXM1	3.40	10.25	2.93	2.72	2.26	2.27	2.41	2.65	8.64	7.70	3.33	2.55	3.06	2.33	2.20	1.81
NikonD5200	3.59	11.53	2.74	2.49	2.52	2.58	2.19	3.03	10.25	11.75	3.36	2.49	2.77	2.30	2.14	1.82
OlympEPL6	2.73	9.54	2.59	2.35	2.26	2.20	2.05	2.59	4.79	10.88	3.00	2.28	2.42	1.92	1.72	1.55
LumixGX1	3.15	14.98	2.48	2.45	2.25	2.26	2.31	2.78	4.98	5.09	2.58	2.37	2.67	2.00	1.87	1.71
SamNX2000	3.15	12.45	2.45	2.66	2.32	2.41	2.22	3.24	7.70	6.56	3.27	2.44	2.94	2.10	2.05	1.87
SonyA57	3.81	8.78	3.03	2.68	2.76	2.80	2.42	2.70	4.45	3.45	2.57	2.74	2.95	2.16	2.03	1.64
Camera	Error for best 25% images (degrees °)															
Canon1Ds	0.95	1.56	0.66	0.64	0.81	0.86	0.59	1.06	1.05	1.38	0.76	0.88	0.78	0.65	0.59	0.54
Canon600D	0.83	2.03	0.64	0.63	0.73	0.80	0.55	1.17	9.98	11.23	0.69	0.68	0.78	0.65	0.54	0.48
FujiXM1	0.91	1.82	0.87	0.73	0.72	0.70	0.65	0.99	3.44	2.30	0.93	0.81	0.86	0.75	0.56	0.53
NikonD5200	0.92	1.77	0.72	0.63	0.79	0.73	0.56	1.16	4.35	3.92	0.92	0.86	0.74	0.66	0.58	0.52
OlympEPL6	0.85	1.65	0.76	0.72	0.65	0.71	0.55	1.15	1.42	1.55	0.91	0.78	0.76	0.51	0.49	0.43
LumixGX1	0.82	2.25	0.78	0.70	0.56	0.61	0.67	0.82	2.06	1.76	0.68	0.82	0.79	0.64	0.51	0.47
SamNX2000	0.81	2.59	0.78	0.77	0.71	0.74	0.66	1.26	2.65	3.00	0.93	0.75	0.75	0.66	0.55	0.51
SonyA57	1.16	1.44	0.98	0.85	0.79	0.89	0.78	0.98	1.28	0.99	0.78	0.87	0.83	0.59	0.48	0.46
Camera	Error for worst 25% images (degrees °)															
Canon1Ds	11.00	16.75	8.52	7.08	7.69	7.76	6.82	7.30	14.16	13.35	7.95	6.43	9.51	6.93	7.94	5.17
Canon600D	8.53	18.75	7.06	7.58	7.48	7.41	6.50	7.40	18.45	18.66	7.93	5.77	5.76	6.28	7.06	5.63
FujiXM1	9.04	18.26	7.55	7.62	7.32	7.23	7.30	7.06	13.40	13.44	8.82	5.99	9.37	7.66	8.24	5.73
NikonD5200	9.69	21.89	7.69	7.53	8.42	8.21	6.73	7.57	15.93	24.33	8.18	6.90	10.01	8.64	7.80	5.98
OlympEPL6	7.41	18.58	6.78	6.69	6.88	6.47	6.31	6.55	15.42	30.21	8.19	6.14	7.46	7.39	6.43	5.15
LumixGX1	8.45	20.40	7.12	6.86	7.03	6.86	6.66	7.42	12.19	11.38	8.00	5.90	8.74	7.81	6.98	5.65
SamNX2000	8.51	20.23	6.92	6.85	7.00	7.23	6.48	7.98	13.01	16.27	8.62	6.22	8.16	6.27	6.95	5.96
SonyA57	9.85	21.27	7.75	6.68	7.18	7.14	6.13	7.32	11.16	9.83	8.02	6.17	7.18	6.89	7.04	5.03

Table 2: Performance comparison of our proposed learning-based method on the NUS 8-camera data set [12] to Grey-world (GW) [9], White-patch (WP) [7], Shades-of-grey (SoG) [19], General Grey-world (GGW) [2], 1st-order Grey-edge (GE1) [39], 2nd-order Grey-edge (GE2) [39], Bright-and-dark Colors PCA (BD) [12] Local Surface Reflectance Statistics (LSR) [22] Pixels-based Gamut (PG) [28], Edge-based Gamut (EG) [28], Bayesian framework (BF)[24], Spatio-spectral Statistics (SS)[11], Natural Image Statistics (NIS) [26], Corrected-moment method (CM) [16], and Color dog (CD) [1].

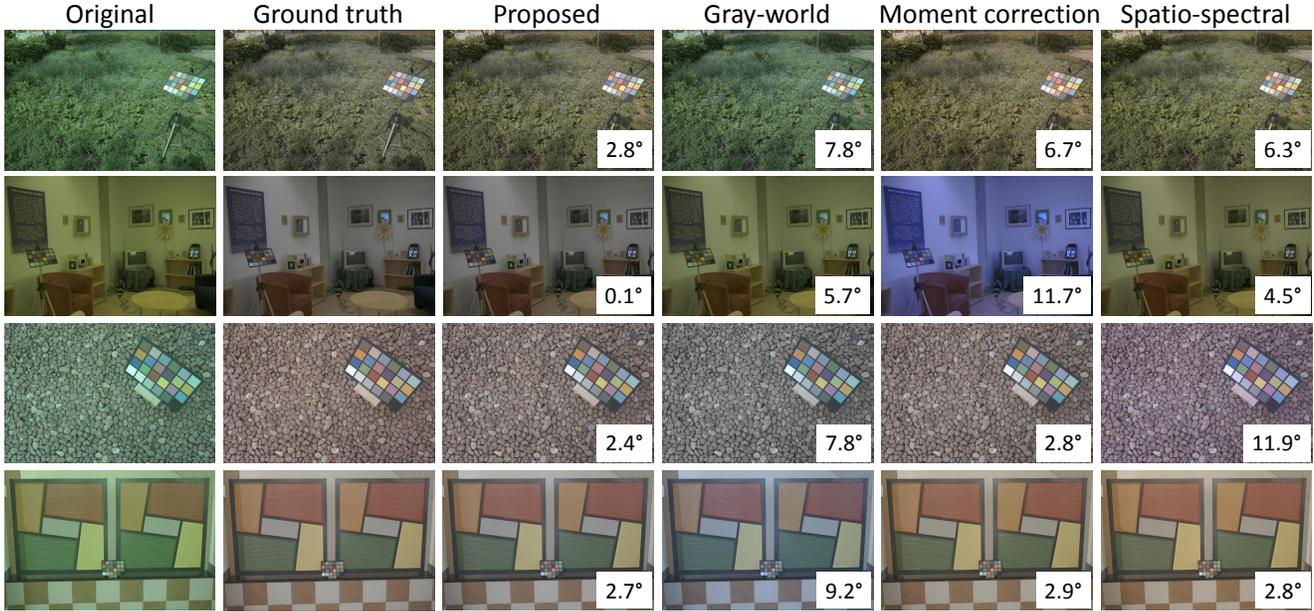


Figure 5: Corrected images using the estimated illuminant from 4 different methods including our proposed one. The angular error is given at the lower right corner of the image. The RAW images have been applied gamma function to boost the contrast for a better visualization.

tunity to evaluate illuminant estimation performance for different camera sensors on the same scene. We report the results on this data set from [12] for 12 methods, and compare with 3 additional recent methods. To compare to Local Surface Reflectance [22], we downloaded the source code from author’s webpage. The 19-Edge Corrected-moment [16] and the Local Surface Reflectance Statistics [22] are reported with the best result achieved with several different parameter settings. The recent Color Dog method is reported with the results from the author. Note that the learning-based methods are trained and tested on each camera subset only. Table 2 lists results on the entire data set for all 8 cameras. From Table 2, we can see that among the multiple methods considered, the proposed algorithm gives the best performance.

3.3. Visual Comparison

Figure 5 shows a visual comparison of results of the proposed method with other algorithms. We can see that for scenes where simple assumptions like the Grey-World assumption are not valid, learning-based methods achieve better results. Compared to other learning-based methods, our proposed method achieves good performance even for extreme cases.

3.4. Timing Comparison

The run-time required to train and test machine learning-based methods is important in determining if a particular method is practical or not. The training and test time were measured on a PC with Intel Xeon 3.5GHz CPU using Matlab 2010. Table 3 reports all the training and testing time for

	Training	Testing
Grey-world [9]	-	0.15 s
White-patch [31]	-	0.16 s
Shades-of-grey [19]	-	0.47 s
General Grey-world [39]	-	0.91 s
1st-order Grey-edge [39]	-	1.05 s
2nd-order Grey-edge [39]	-	1.26 s
Bright-and-dark Colors PCA [12]	-	0.24 s
Local Surface Reflectance [22]	-	0.22 s
Pixel-based Gamut [28]	1345 s	2.65 s
Edge-based Gamut [28]	1986 s	3.64 s
Bayesian [24]	764 s	96.57 s
Spatio-spectral [11]	3159 s	6.88 s
Natural Image Statistics [26]	10749 s	1.49 s
19-Edge Corrected-moment [16]	584 s	0.77 s
Proposed	245 s	0.25 s

Table 3: Training and averaged per-image evaluation times(in seconds) for different methods on Gehler-Shi’s data set [24, 35]. The statistical methods do not require training.

the whole Gehler-Shi data set including image read time. As seen in Table 3, our proposed method is clearly the fastest learning-based method in terms of both training and testing, and requires less than half the run-time of the previous fastest learning-based method [16]. Compared with statistical methods, our proposed method is on par with the fastest methods (e.g. Grey-World and White-Patch).

3.5. SFU Laboratory Object Image Set

For sake of completeness, we also compare results on a commonly used subset of 321 images from the SFU data set [3] that consists of 31 objects viewed under up to 11

Method		Mean	Median	Trimean	Best-25%	Worst-25%
Statistics-based	Grey-world [9]	9.78	7.00	7.60	0.89	23.45
	White-patch [31]	9.09	6.48	7.45	1.84	20.97
	Shades-of-grey [19]	6.39	3.74	4.59	0.59	16.49
	General Grey-world [39]	5.41	3.32	3.78	0.49	13.75
	1st-order Grey-edge [39]	5.58	3.18	3.74	1.05	14.05
	2nd-order Grey-edge [39]	5.19	2.74	3.25	1.10	13.51
	Local Surface Reflectance [22]	5.69	2.43	3.51	0.47	15.84
Learning-based	Pixel-based Gamut [28]	3.70	2.27	2.53	0.46	9.32
	Edge-based Gamut [28]	3.92	2.28	2.70	0.51	9.91
	Intersection-based Gamut [28]	3.62	2.09	2.38	0.50	9.38
	Spatio-spectral [11]	5.63	3.45	4.33	1.23	12.90
	19-Edge Corrected moment [16] (ideal)	2.71	2.25	2.39	0.91	5.26
	19-Edge Corrected moment [16] (CV)	3.22	2.53	2.65	0.91	6.68
	Our Proposed (ideal)	0.25	0.10	0.13	0.00	0.77
	Our Proposed (CV)	3.26	1.75	2.12	0.31	8.90

Table 4: Performance comparison of our proposed learning-based method against other methods on the SFU laboratory data set [3].

different lights in a laboratory setting. The variation of the scene objects, however, is limited and the data set contains many unusual blue and red lights. Another problem with this data set is the images are not camera RAW images and may be affected by onboard camera color manipulation. Because of these issues, statistical methods do not perform well on this data set and it is even difficult for learning-based methods. Thus, instead using a 3-fold cross validation, the Gamut-based [28] and Spatio-spectral [11] methods train using images from all 31 objects using a single light (termed *syl-50MR16Q*) as the target light source. To test our method with this ideal training approach (and to re-evaluate the corrected-moment method [16]), all the images in the data set were used for training. We also note that the corrected-moment method has been evaluated without the extra step to raise the image to the power of 2 as mentioned in the original paper.

Table 4 reports performance from the few methods that have been tested on this data set [25]. Our proposed method gives excellent results for this hard data set when using ideal training (indicated with ideal), which is far better than the second best result from corrected-moment method [16]. As the ideal training allows overfitting to the test set, we also performed the standard 3-fold cross validation for our proposed method and corrected-moment method (indicated with CV). In this case, our proposed method is still the best over three of the error metrics while for the other two metrics, our results are second to the best achieved by the corrected-moment method.

4. Discussion and Concluding Remarks

This work has presented a learning-based method for illumination estimation that uses four 2D features with an ensemble of regression trees. We have demonstrated on three standard data sets that our approach can produce ex-

cellent results with a running-time comparable with statistical methods. Our fast running-time is attributed to our features that are based on simple 2D descriptors computed on the input image’s RGB color distribution. There is no need for convolution, spatial derivatives, distribution moments, or frequency decomposition. In addition, the K tree pairs can be evaluated very quickly given the binary tree structure. Moreover, the training of these trees is reasonably fast.

It is worth noting that we tried a number of alternative designs for our tree ensemble. In particular, we tested our results using a single regression tree trained using all four features described in Section 2.1 combined as a single input feature. This resulted in a 30% worse performance in terms of the average error obtained using the proposed method. We also modified the local weighting scheme described in Section 2.3 to randomly sample the training-data for each K tree, effectively resulting in an ensemble of random forests. This strategy resulted in a 25% worse performance from our proposed implementation. Initially, we also tried constructing a naive random forest. The best result we could obtain using 100 trees was 20% worse than our reported results. Overall, we found our proposed strategy including regression tree consensus and our sampling approach provided gains over alternative designs.

To summarize, this paper has demonstrated a learning-based approach that gives state-of-the-art results with running-time comparable with statistical methods. The larger implication of this work is that learning-based methods can be viable real-time options and suitable for onboard camera processing.

Acknowledgement

This work was supported by Singapore A*STAR PSF grant 11212100 and an Adobe gift award.

References

- [1] N. Banić and S. Lončarić. Color dog: Guiding the global illumination estimation to better accuracy. In *International Conference on Computer Vision Theory and Applications*, 2015. 6
- [2] K. Barnard, L. Martin, A. Coath, and B. Funt. A comparison of computational color constancy algorithms. ii. experiments with image data. *TIP*, 11(9):985–996, 2002. 6
- [3] K. Barnard, L. Martin, B. Funt, and A. Coath. A data set for color research. *Color Research & Application*, 27(3):147–151, 2002. 5, 7, 8
- [4] S. Bianco, G. Ciocca, C. Cusano, and R. Schettini. Improving color constancy using indoor - outdoor image classification. *TIP*, 17(12):2381–2392, 2008. 2
- [5] S. Bianco, G. Ciocca, C. Cusano, and R. Schettini. Automatic color constancy algorithm selection and combination. *Pattern Recognition*, 43(3):695–705, 2010. 2, 5
- [6] Z. Botev, J. Grotowski, D. Kroese, et al. Kernel density estimation via diffusion. *The Annals of Statistics*, 38(5):2916–2957, 2010. 3
- [7] D. H. Brainard and B. A. Wandell. Analysis of the retinex theory of color vision. *JOSA A*, 3(10):1651–1661, 1986. 2, 6
- [8] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and regression trees*. CRC press, 1984. 3, 4
- [9] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of The Franklin Institute*, 310(1):1–26, 1980. 2, 5, 6, 7, 8
- [10] V. C. Cardei, B. Funt, and K. Barnard. Estimating the scene illumination chromaticity by using a neural network. *JOSA A*, 19(12):2374–2386, 2002. 2, 3
- [11] A. Chakrabarti, K. Hirakawa, and T. Zickler. Color constancy with spatio-spectral statistics. *TPAMI*, 34(8):1509–1519, 2012. 2, 5, 6, 7, 8
- [12] 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*, 31(5):1049–1058, 2014. 5, 6, 7
- [13] M. S. Drew and B. V. Funt. Variational approach to inter-reflection in color images. *JOSA A*, 9(8):1255–1265, 1992. 2
- [14] M. S. Drew, H. R. V. Joze, and G. D. Finlayson. Specularity, the zeta-image, and information-theoretic illuminant estimation. In *ECCV, Workshops and Demonstrations*, 2012. 2
- [15] G. Finlayson, S. Hordley, and P. Morovic. Colour constancy using the chromagenic constraint. In *CVPR*, 2005. 2
- [16] G. D. Finlayson. Corrected-moment illuminant estimation. In *ICCV*, 2013. 2, 5, 6, 7, 8
- [17] G. D. Finlayson, M. S. Drew, and B. V. Funt. Color constancy: generalized diagonal transforms suffice. *JOSA A*, 11(11):3011–3019, 1994. 1
- [18] G. D. Finlayson, S. D. Hordley, and P. M. Hubel. Color by correlation: A simple, unifying framework for color constancy. *TPAMI*, 23(11):1209–1221, 2001. 2, 3
- [19] G. D. Finlayson and E. Trezzi. Shades of gray and colour constancy. In *Color and Imaging Conference*, 2004. 2, 5, 6, 7, 8
- [20] D. A. Forsyth. A novel algorithm for color constancy. *IJCV*, 5(1):5–35, 1990. 2
- [21] B. Funt and W. Xiong. Estimating illumination chromaticity via support vector regression. In *Color and Imaging Conference*, 2004. 2, 3, 5
- [22] S. Gao, W. Han, K. Yang, C. Li, and Y. Li. Efficient color constancy with local surface reflectance statistics. In *ECCV*, 2014. 2, 5, 6, 7, 8
- [23] S. Gao, K. Yang, C. Li, and Y. Li. A color constancy model with double-opponency mechanisms. In *ICCV*, 2013. 2
- [24] P. V. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp. Bayesian color constancy revisited. In *CVPR*, 2008. 2, 4, 5, 6, 7
- [25] A. Gijsenij. Color constancy: research website on illuminant estimation. accessed from <http://colorconstancy.com/>. 5, 8
- [26] A. Gijsenij and T. Gevers. Color constancy using natural image statistics and scene semantics. *TPAMI*, 33(4):687–698, 2011. 2, 5, 6, 7
- [27] A. Gijsenij, T. Gevers, and M. P. Lucassen. Perceptual analysis of distance measures for color constancy algorithms. *JOSA A*, 26(10):2243–2256, 2009. 5
- [28] A. Gijsenij, T. Gevers, and J. Van De Weijer. Generalized gamut mapping using image derivative structures for color constancy. *IJCV*, 86(2-3):127–139, 2010. 5, 6, 7, 8
- [29] A. Gijsenij, T. Gevers, and J. van de Weijer. Computational color constancy: Survey and experiments. *TIP*, 20(9):2475–2489, 2011. 2, 4, 5
- [30] A. Gijsenij, T. Gevers, and J. Van De Weijer. Improving color constancy by photometric edge weighting. *TPAMI*, 34(5):918–929, 2012. 2
- [31] E. H. Land and J. J. McCann. Lightness and retinex theory. *JOSA A*, 61(1):1–11, 1971. 5, 7, 8
- [32] H.-C. Lee. Method for computing the scene-illuminant chromaticity from specular highlights. *JOSA A*, 3(10):1694–1699, 1986. 2
- [33] C. Rosenberg, M. Hebert, and S. Thrun. Color constancy using kl-divergence. In *ICCV*, 2001. 2, 3
- [34] C. Rosenberg, A. Ladsariya, and T. Minka. Bayesian color constancy with non-gaussian models. In *NIPS*, 2003. 2
- [35] L. Shi and B. Funt. Re-processed version of the gehler color constancy dataset of 568 images. accessed from <http://www.cs.sfu.ca/~colour/data/>. 4, 5, 7
- [36] L. Shi and B. Funt. Maxrgb reconsidered. *Journal of Imaging Science and Technology*, 56(2):20501–1, 2012. 2
- [37] R. T. Tan, K. Nishino, and K. Ikeuchi. Color constancy through inverse-intensity chromaticity space. *JOSA A*, 21(3):321–334, 2004. 2
- [38] H. Vaezi Joze and M. Drew. Exemplar-based colour constancy and multiple illumination. *TPAMI*, 36(5):860–873, 2014. 2, 5
- [39] J. Van De Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *TIP*, 16(9):2207–2214, 2007. 2, 5, 6, 7, 8
- [40] J. Van De Weijer, C. Schmid, and J. Verbeek. Using high-level visual information for color constancy. In *ICCV*, 2007. 5