End-to-End Illuminant Estimation based on Deep Metric Learning

Bolei Xu [1,2] Jingxin Liu [2,3] Xianxu Hou [2] Bozhi Liu [2] Guoping Qiu [2,4]
[1] College of Computer Science and Technology, Zhejiang University of Technology
[2] College of Information Engineering, Shenzhen University
[3]Histo Pathology Diagnostic Center, Shanghai
[4] School of Computer Science, The University of Nottingham

Abstract

Previous deep learning approaches to color constancy usually directly estimate illuminant value from input image. Such approaches might suffer heavily from being sensitive to the variation of image content. To overcome this problem, we introduce a deep metric learning approach named Illuminant-Guided Triplet Network (IGTN) to color constancy. IGTN generates an Illuminant Consistent and Discriminative Feature (ICDF) for achieving robust and accurate illuminant color estimation. ICDF is composed of semantic and color features based on a learnable color histogram scheme. In the ICDF space, regardless of the similarities of their contents, images taken under the same or similar illuminants are placed close to each other and at the same time images taken under different illuminants are placed far apart. We also adopt an end-to-end training strategy to simultaneously group image features and estimate illuminant value, and thus our approach does not have to classify illuminant in a separate module. We evaluate our method on two public datasets and demonstrate our method outperforms state-of-the-art approaches. Furthermore, we demonstrate that our method is less sensitive to image appearances, and can achieve more robust and consistent results than other methods on a High Dynamic Range dataset.

1. Introduction

Under a different light source, an object will reflect a different color appearance. Color constancy is a feature of the human visual system which ensures that the perceived color of objects remains relatively constant under varying lighting conditions. Computational color constancy tries to develop digital imaging algorithms that mimic such human vision ability. In the literature, many color constancy algorithms have been proposed to achieve this goal including learning-based methods [1, 20, 31, 18, 15] and statistical methods [7, 16, 27, 13, 37].

These traditional approaches usually define a photographed scene image as:

$$I(x) = \int_{\omega} E(x,\lambda)S(x,\lambda)C(\lambda)d\lambda, \tag{1}$$

where $I(x)$ is the image value at the spatial coordinate x, $E(x,\lambda)$ is the color of light source, $S(x,\lambda)$ is the surface spectral reflectance, $C(\lambda)$ is the camera sensor sensitivity function, and ω is the visible spectrum of the wavelength λ. According to the Von Kries coefficient law [6] and the assumption of single light, it could be simplified as [2, 33]:

$$I = E \times S, \tag{2}$$

where each observed RGB pixel in I is the product of the RGB illumination E shared by all pixels and the RGB value S of reflectance under canonical illumination (usually white). The goal of color constancy can thus be defined as estimating E from I.

Recently researchers have applied deep learning [28, 29, 22, 33, 5] to estimate the illuminations. They regard color constancy as a regression problem that aims to learn a mapping function f through a deep learning model to map the observed image content to the illumination value:

$$E = f(I) \tag{3}$$

One problem with these deep learning approaches is that the predicted illuminant is heavily influenced by the scene content. For instance, two different patches x_i and x_j from the same image could have different scene contents. When directly inferring illumination from those two different scene contents, it would lead to different estimation results, i.e., $f(\mathrm{x}_i) \neq f(\mathrm{x}_j)$, since the mapping function f is fixed after training, any changes in the input will directly affect the output. However, with the assumption of single illuminant setting (Equation 3), the estimation should be location independent, which means an observation at any location of the scene should correctly estimate the same illuminant color. On the other hand, most

previous deep learning methods try to directly estimate illuminant from images. We argue that it is beneficial to learn content-insensitive illuminant features first and then estimate illuminant value from these features. These features should be only sensitive to illuminant changes and less influenced by the variation of image contents.

To overcome aforementioned problems, we propose a deep metric learning framework called Illuminant-Guided Triplet Network (IGTN) to first embed input images into an Illuminant Consistent and Discriminative Feature (ICDF) space through a mapping function $h(\cdot)$ and then estimate illuminant value from ICDF by an estimation function $f(\cdot)$. Specifically, the IGTN has three networks with shared weights and thus takes three images as inputs in the training stage, where two images have the same illuminant value, and a third image has a dissimilar illuminant to the previous two images. For each of the input image, we propose a base network to obtain semantic features and a multi-scale learnable color histogram scheme to extract the image color features. Both semantic features and color features are combined to form the ICDF representation. ICDF are further refined by the triplet loss and angular loss to better reflect illuminant information. The whole network can be trained in an end-to-end manner, and thus does not need to cluster images before training the neural network as done in the work of [29].

We test our approach on two public color constancy datasets to show our method's superior performances wtih respect to previous approaches. We also test our method on images of high dynamic range (HDR) scenes taken with different camera parameter settings. We show that our method is capable of more consistently estimating the illuminant color across images taken with different camera parameter settings, thus demonstrating the robustness of our method.

Our contributions are as follows: (1) We provide a new perspective on deep learning based color constancy, where a good feature should be consistent when they are from images with similar illuminants, discriminative when they come from images with different illuminants, and insensitive to the variation of image content. (2) We achieve this by proposing a deep metric learning method termed Illuminant-Guided Triplet Network to generate Illuminant Consistent and Discriminative Features (ICDF) for color constancy. (3) Our proposed method could be trained in an end-to-end manner and thus does not have to do illuminant clustering in a separate module. (4) We evaluate our approach on two public datasets, where our approach demonstrates superior performance with respect to previous methods.

2. Related Work

Recent years have witnessed a large number of work on color constancy. They can be roughly categorized into three branches, (i) statistics-based, (ii) learning-based and (iii) deep learning based.

2.1. Statistics-Based Approaches

Some statistics-based approaches assume the statistics of reflectance in the scene to be achromatic. A number of well-known approaches including Grey-World [7], White-Patch [16, 27] , Shades of Grey [13] and Grey-Edge [37] are based on the assumption of the scene color to be gray.

The advantage of the statistics-based approaches is that they do not require training data and are usually efficient. However, the performance of these methods is not comparable to the learning-based approaches.

2.2. Learning-Based Approaches

The learning-based approaches employ labeled training data to estimate illumination. There are mainly two lines of learning-based methods including combinatorial methods and direct methods.

Combinatorial methods try to optimally combine several statistics-based methods according to the scene contents of the input images. One work [15] trained a neural network to estimate illumination. They binarized the rg-chromaticity as the input. However, as they stated in the paper, such binarization results in a large input layer especially when processing 12-bit raw images. The work of [19] applies low level properties of images to select the best combination of algorithms.

Direct approaches aim to train a learning model and estimate the illumination from the training dataset. The Gamut Mapping methods assume one observes only a limited gamut of colors for a given illuminant [14]. [1, 20] first find the canonical gamut from the training data and then map the gamut of each input image into the canonical gamut. Other learning approaches such as SVR-based algorithm [17], neural networks [35], Bayesian model [31, 18] and the exemplar-based algorithm [24], usually employ hand-crafted features and these learning models are also shallow.

2.3. Deep Learning Based Approaches

With the emergence of deep learning, the deep features [21, 34, 36] are shown to achieve superior performance to the traditional hand-crafted features.

There are several deep learning work trying to solve the color constancy problem. One problem with the deep learning approaches is that the size of dataset is usually small, and it would lead to the over-fitting problem with deep neural network. To overcome this problem, [28]

uses ImageNet dataset to pre-train a CNN whose ground-truths are obtained by the existing method such as Gray-of-shades. Another way to augment the dataset size is to partition the raw image into patches. One pioneer work [4] takes raw image as input and directly predict illumination from a CNN. In their further work [5], they develop a multiple illuminant detector to decide whether to aggregate the local outputs into the single estimate. The author of [22] develop a fully convolutional network architecture that can take any size of input patches. In the work of [33], they propose a selection network to choose an estimate from illumination hypotheses. [30] constructs a recurrent neural network to take a sequence of input image patches to estimate illuminant.

The closest work to ours is probably [29]. Authors regard the color constancy as a classification problem. They try to cluster training data by k-means and to compute illuminants by finding their nearest neighbor in the training dataset. In comparison, our approach does not have to do clustering in a separate module. The deep metric learning framework in our approach can simultaneously group image features and also estimate illuminant value. Also, in the work of [29], they have to manually define the cluster number by applying k-means, which is not required in our approach.

3. Methodology

In this section we describe our proposed method. We aim to design a deep neural network to map image to the ICDF space, and then estimate the illuminant value from ICDF (the whole framework is shown in Figure 1). Our proposed method is composed of two main parts to achieve this goal:

- A Deep Illuminant Network (DIN) based on the AlexNet and learnable histogram scheme to extract semantic features and multi-scale color features from image.
- An Illuminant-Guided Triplet Network (IGTN) consists of three DINs with shared weights to generate ICDF representation.

In the following we detail each of these parts separately.

3.1. Deep Illuminant Network

We propose a Deep Illuminant Network to extract image features $h(\mathrm{x})$ from input image x and it consists of two components: a base network and a learnable histogram network. The base network is constructed to obtain semantic image features and the learnable histogram aims to extract color features. The final image representation is the combination of semantic features and color features.

We choose AlexNet (up to **FC6**) as the base network to extract image features. It is mainly due to two reasons: (1) the size of current color constancy datasets are usually small, thus using very deep network such as ResNet [21] would lead to the over-fitting problem; (2) although very deep networks have powerful discriminative ability, they are usually illuminant-insensitive which is not a suitable property for the illuminant estimation problem. Thus we did not choose those networks with very deep structure in this work.

The original AlexNet is designed for the classification problem, thus it is able to extract semantic features. However, in the illuminant estimation problem, we are also interested in the illuminant color. Therefore, we apply a learnable color histogram scheme to extract color features.

3.2. Learnable Color Histogram

Color feature is one of the most important features to address the illuminant estimation problem. In this work, we extend the work [41] to extract both global and local color histograms to represent image color features.

We choose the learnable color histogram mainly for two reasons: (1) unlike traditional color histogram, the computation of learnable color histogram is differentiable and thus could be trained in an end-to-end manner in the deep learning framework; (2) the computation process of learnable histogram can be represented by existing deep learning layers and thus makes it easy to implement.

In the learnable color histogram, the centers and widths of the bin are learned by the deep neural network. For each pixel in the image, the voting function for it to select a bin is formulated as below:

$$\psi_{k,b}(x_k) = \max\{0, 1 - |x_k - \mu_{k,b}| \times w_{k,b}\}, \qquad (4)$$

where x_k is the value of k-th element in the feature map, b is the index of output bin, $\mu_{k,b}$ is the value of the voted bin center and $w_{k,b}$ is the width of the $b-$th bin.

The nice property of learnable color histogram is that its computation process can be modeled by the existing deep learning layers. In specific, the computation of $|x_k - \mu_{k,b}|$ is the same as to convolving the feature map by a fixed 1×1 unit convolutional kernel with a learnable bias term $-\mu_{k,b}$ and then to compute its absolute value. The calculation of $1 - |x_k - \mu_{k,b}| \times w_{k,b}$ is equivalent to be then convolved by another 1×1 convolutional kernel with learnable weights and fixed bias terms of value 1. The $\max\{0, \cdot\}$ is exactly the same as the ReLU activation function. The output dimension of Equation 4 is $H \times W \times C \times B$, where H, W, C are the number of height, width and channel of input, B is the number of bins of the histogram.

In order to extract global and local color features, we then adopt a spatial pyramid pooling layer based on the learnable color histogram scheme. Formally, we apply three scales of the pooling pyramid by a global average pooling based on the learned color histogram. The strides of three

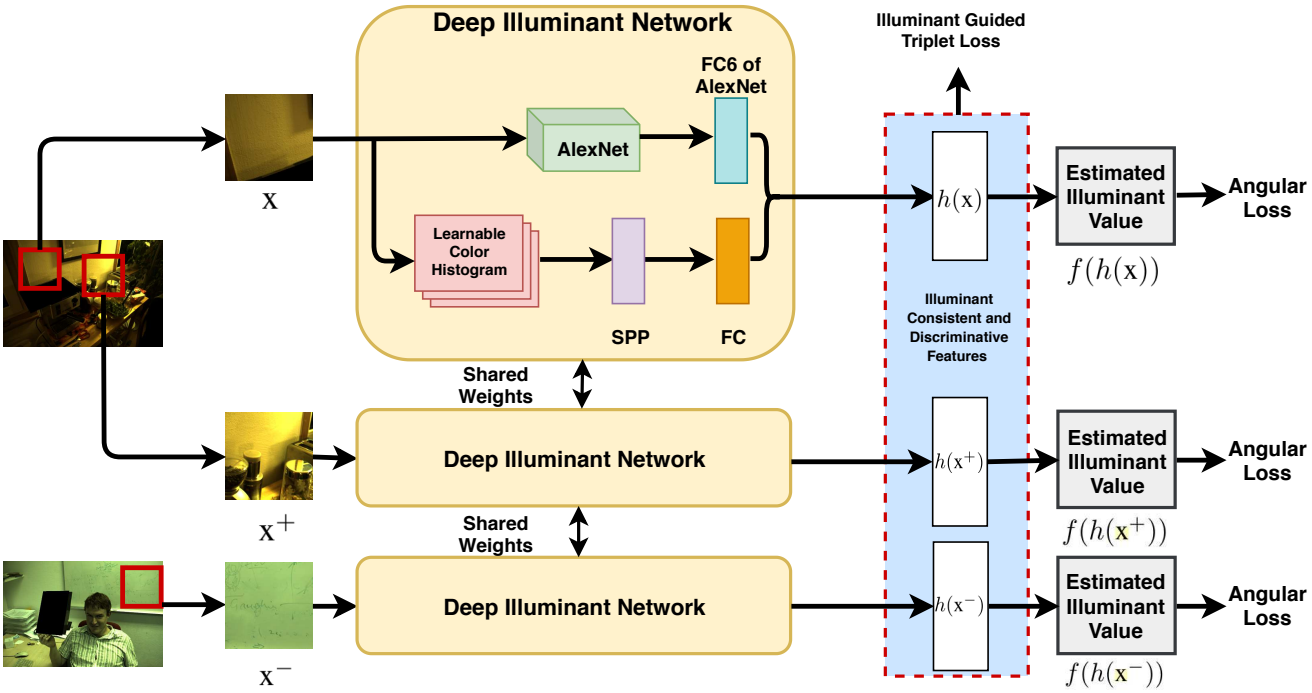

Figure 1. The architecture of Illuminant-Guided Triplet Network (IGTN). The network takes three inputs: two have the same illuminant value and one with different illuminant value. The IGTN maps images to the ICDF space according to their illuminant value. The final illuminant could be inferred from the ICDF representation. Thus, the whole network could be trained in an end-to-end manner. Here "SPP" refers to spatial pyramid pooling layer and "FC" denotes fully-connected layer.

global pooling are set to 1, 2, and 4 respectively. All three pooled color histograms are flatten for concatenation. Then we construct a fully-connected layer with 4,096 neurons to represent the final learnable color histogram features. The learned color histogram features are further combined with the semantic features to form the ICDF representation.

3.3. Illuminant-Guided Triplet Network

The current learned image features $h(\mathrm{x})$ are more related to the image content, while in the illuminant estimation task we expect the features to be more related to the illuminant information.

In order to generate the ICDF representation, we propose a deep metric learning framework called the Illuminant-Guided Triplet Network (IGTN) to achieve the illuminant consistent and discriminative feature (ICDF) embedding. The overall architecture of IGTN is based on the Triplet Network framework which consists of three DINs with shared weights. IGTN takes three inputs $\mathrm{x}, \mathrm{x}^+, \mathrm{x}^-$, where x and x^+ have the same illuminant value, and x^- has different illuminant value from previous two images. IGTN aims to refine the image features $h(\mathrm{x})$ produced by DIN, and to achieve embedding for each of the input images. Then, the IGTN could estimate the illuminant value $\hat{\mathrm{y}}$ from the embedding by an estimation function $f(\cdot)$:

$$\hat{\mathrm{y}} = f(h(\mathrm{x})) \tag{5}$$

In the vanilla triplet network [40, 26], the loss function is formulated as:

$$\mathcal{L}_T = \max(0, \|h(\mathrm{x}_i) - h(\mathrm{x}_i^+)\|_2^2 - \|h(\mathrm{x}_i) - h(\mathrm{x}_i^-)\|_2^2 + \alpha), \tag{6}$$

where α is a constant margin value. This loss function tends to pull images of same class $(\mathrm{x}, \mathrm{x}_+)$ into nearby points in the embedding space, and push images of different classes $(\mathrm{x}, \mathrm{x}_-)$ apart from each other.

While the vanilla triplet loss retains the intra-class and inter-class distances in the classification problems, it could not well-describe the ordinal sample relationship in the regression problem such as illuminant estimation.Thus, a constant penalty α is not enough to disclose the ordinal illuminant difference between image pairs. To overcome this problem, we propose an illuminant guided triplet loss based on the illuminant difference. The margin value of penalty parameter α is defined as:

$$\alpha = \tau D(\mathrm{y}_i, \mathrm{y}_i^-), \tag{7}$$

where τ is a hyperparameter and $D(\mathrm{y}_i, \mathrm{y}_i^-)$ is the angular distance to measure the illuminant difference between

anchor image and the negative image sample. By adopting such learning strategy, we ensure the penalty margin changes w.r.t the illuminant differences between different image pairs. The modified triplet loss can then be formulated as:

$$\mathcal{L}_T = \max(0, \|h(\mathrm{x}_i) - h(\mathrm{x}_i^+)\|_2^2 - \|h(\mathrm{x}_i) - h(\mathrm{x}_i^-)\|_2^2 + \tau D(\mathrm{y}_i, \mathrm{y}_i^-)). \tag{8}$$

3.4. Triplet Sampling

Another issue with triplet network is how to construct the triplet inputs. As we mentioned before, we aim to construct triplet inputs as two image with the same illuminant and one with different illuminant. In the single illuminant task, we assume the illuminant is uniformly distributed within the image. Thus, one safe way to construct two images with the same illuminant is to take two image patches cropping from one image. When looking for the image with different illuminant, we use a threshold η to define the illuminant difference. Here we define two images' illuminant value are different if their angular distance is larger than the threshold:

$$D(\mathrm{y}_i, \mathrm{y}_j) > \eta \tag{9}$$

3.5. End-to-End Optimization

The final illuminant value can be estimated from ICDF to produce normalized r, g value. As illuminant estimation is an ill-posed problem, the learned features have to be supervised by the illuminant labels to achieve more accurate results. We thus further apply an angular error loss to optimize the IGTN. The angular error loss is formulated as:

$$\mathcal{L}_A(\mathrm{x}_i) = \arccos(\frac{f(h(\mathrm{x}_i)) \cdot \mathrm{y}_i}{\|f(h(\mathrm{x}_i))\| \cdot \|\mathrm{y}_i\|}), \tag{10}$$

where $f(h(\mathrm{x}_i))$ is the predicted illuminant value by IGTN and y_i is the ground-truth illuminant value. The total loss is the combination of modified triplet loss and the angular loss for each input image:

$$\mathcal{L}_{total} = \mathcal{L}_T(\mathrm{x}_i, \mathrm{x}_i^+, \mathrm{x}_i^-) + \mathcal{L}_A(\mathrm{x}_i) + \mathcal{L}_A(\mathrm{x}_i^+) + \mathcal{L}_A(\mathrm{x}_i^-). \tag{11}$$

By adopting such training strategy, the whole network could be trained in an end-to-end manner. Also, the IDCF representation are refined to better reflect illuminant information and thus enable to more accurately predict the illuminant value.

4. Experiment

4.1. Settings

Implementation and Training We implemented our networks based on Tensorflow and Keras using four GTX 1080 Ti GPUs. When training the IGTN, we set the learning rate to 1×10^{-4}. The batch size is set to 48. The AlexNet is pre-trained on the ImageNet dataset. We use Adam optimizer [25] to train the network. We use 6 bins for the learnable color histogram, and the initial centers were set to $(0, 0.2, 0.4, 0.6, 0.8, 1)$ and the initial bin widths were set to 0.2. The value of τ in Equation 8 is set to 0.2. We set threshold $\eta = 3$ in Equation 9.

Data Augmentation and Preprocessing The image patches are randomly cropped from raw image with size of 227×227. As the size of color constancy dataset is usually small, we augment the image data by a random angle rotation between -15° and 15° and left-right flipping with a probability of 0.5. We also apply a gamma correction of $\gamma = 1/2.2$ on linear RGB images to be fit with the images in the ImageNet dataset.

Datasets We first evaluate our approach on the reprocessed [32] Color Checker Dataset dataset [18]. It consists of 568 raw images. Another dataset used in the experiment is the NUS 8-camera set [9]. It contains 1,736 images from 8 different cameras and the experiment is done independently on each sub-dataset. Both datasets use a Macbeth Color Checker (MCC) to obtain the ground truth illumination color. When doing experiment, we masked out the MCCs in both training and testing phase. The evaluation is done through a 3-fold cross validation for both datasets. For the NUS dataset, we calculate the performance metric by taking geometric mean over the eight image subsets as done in previous works. We use the angular error metric to evaluate the performance of different methods:

$$err_{angle} = \arccos(\frac{\hat{y_i} \cdot y_i}{\|\hat{y_i}\| \cdot \|y_i\|}) \tag{12}$$

where $\hat{y_i}$ is the estimated illuminant value and y_i is the ground-truth.

4.2. Experiment Results on the Color Constancy Datasets

We present the experimental results in Table 1 and Table 2. In the NUS 8-Camera dataset, it can be seen that our new approach achieves the lowest errors in most evaluation metrics when compared with previous approaches, including the state of art deep learning methods. In the Color Checker dataset, our approach also demonstrates competitive estimation results. We achieve the lowest mean error and worst-25% error and slightly higher errors on the median and best-25% evaluation metrics. These results demonstrate the effectiveness of using the triplet network to extract illuminant consistent and discriminative local features. It should be noticed that FFCC [3] achieves the best performance in Color Checker dataset by utilizing additional "semantics" features from another pre-trained CNN model [39] and the "meta-data"

	Color Checker Dataset			
	Mean	Median	Best-25%	Worst-25%
Gray World [7]	6.36	6.28	2.33	10.58
General Gray World [37]	4.66	3.48	1.00	10.09
White Patch [6]	7.55	5.68	1.45	16.12
Shades-of-Gray [13]	4.93	4.01	1.14	10.20
Spatio-spectral (GenPrior) [8]	3.59	2.96	0.95	7.61
Cheng *et al.* [9]	3.52	2.14	0.50	8.74
NIS [19]	4.19	3.13	1.00	9.22
Corrected-Moment (Edge) [12]	3.12	2.38	0.90	6.46
Corrected-Moment (Color) [12]	2.96	2.15	0.64	6.69
Exemplar [24]	3.10	2.30	-	-
Regression Tree [10]	2.42	1.65	0.38	5.87
CNN [5]	2.36	1.98	-	-
CCC (dist+ext) [2]	1.95	1.22	0.35	4.76
DS-Net (HypNet+SelNet) [33]	1.90	1.12	0.31	4.84
FFCC-4 channels [3]	1.78	0.96	0.29	4.29
FFCC-2 channels, +S [3]	1.67	0.96	0.26	4.23
FFCC-2 channels, +M [3]	1.65	**0.86**	0.24	4.44
FFCC-2 channels, +S +M [3]	1.61	**0.86**	**0.23**	4.27
SqueezeNet-FC4 [22]	1.65	1.18	0.38	3.78
AlexNet-FC4 [22]	1.77	1.11	0.34	4.29
Ours (vanilla triplet loss)	1.73	1.09	0.31	4.25
Ours (no triplet)	1.78	1.13	0.34	4.31
Ours (no learnable histogram)	1.85	1.10	0.31	4.91
Ours (no AlexNet)	2.49	1.70	0.41	6.01
Ours (no SPP; $s = 1$)	1.72	1.08	0.32	4.20
Ours (no SPP; $s = 2$)	1.76	1.09	0.34	4.28
Ours (no SPP; $s = 4$)	1.78	1.11	0.35	4.34
Ours (full)	**1.58**	0.92	0.28	**3.70**

Table 1. Performance of various methods on the Color Checker dataset. For metric values not reported in the literature, their entries are left blank. We denote "S" as the semantic data employed in [3], and denote "M" as the meta-data used in [3].

Method	Mean	Med	Best-25%	Worst-25%
White-Patch [6]	10.62	10.58	1.86	19.45
Edge-based Gamut [1]	8.43	7.05	2.41	16.08
Pixel-based Gamut [1]	7.70	6.71	2.51	14.05
Intersection-based Gamut [1]	7.20	5.96	2.20	13.61
Gray-World [7]	4.14	3.20	0.90	9.00
Bayesian [18]	3.67	2.73	0.82	8.21
NIS [19]	3.71	2.60	0.79	8.47
Shades-of-Gray [13]	3.40	2.57	0.77	7.41
1st-order Gray-Edge [37]	3.20	2.22	0.72	7.36
2nd-order Gray-Edge [37]	3.20	2.26	0.75	7.27
Spatio-spectral (GenPrior) [8]	2.96	2.33	0.80	6.18
Corrected-Moment (Edge) [12]	3.03	2.11	0.68	7.08
Corrected-Moment (Color) [12]	3.05	1.90	0.65	7.41
Cheng *et al.* [9]	2.92	2.04	0.62	6.61
CCC (dist+ext) [2]	2.38	1.48	0.45	5.85
Regression Tree [10]	2.36	1.59	0.49	5.54
DS-Net (HypNet+SelNet) [33]	2.24	1.46	0.48	6.08
AlexNet-FC4 [22]	2.12	1.53	0.48	4.78
FFCC-4 channels [3]	1.99	1.31	**0.35**	4.75
SqueezeNet-FC4 [22]	2.23	1.57	0.47	5.15
Ours (vanilla triplet loss)	2.02	1.36	0.45	4.70
Ours (no triplet)	2.28	1.64	0.51	5.20
Ours (no learnable histogram)	2.15	1.52	0.47	5.28
Ours (no AlexNet)	2.86	1.99	0.59	6.98
Ours (no SPP; $s = 1$)	2.02	1.35	0.45	4.72
Ours (no SPP; $s = 2$)	2.15	1.48	0.60	4.98
Ours (no SPP; $s = 4$)	2.22	1.54	0.45	5.12
Ours (full)	**1.85**	**1.24**	0.36	**4.58**

Table 2. Performance of various methods on the NUS dataset.

of EXIF tags in the Color Checker dataset. In our approach, both of these additional features are not applied to train our deep learning models.

We then conducted a series of ablation experiment to study the importance of each component of our deep metric learning framework. We built four kinds of baseline models:

1. Ours (vanilla triplet loss): We use the vanilla triplet loss [40, 26] instead of Equation 8 to train the network.

2. Ours (no triplet): The triplet network framework is remove and only single Deep Illuminant Network is trained to estimate the illuminant.

3. Ours (no learnable color histogram): The whole learnable color histogram is removed (spatial pyramid pooling is also removed). Only the AlexNet is used to extract image features.

4. Ours (no AlexNet): We removed the base network and only learnable color histogram is served as the feature extractor.

5. Ours (no SPP): The spatial pyramid pooling mechanism is removed (learnable histogram is remained). We only use global average pooling *once*, and different settings of stride ($s = 1, 2, 4$) are evaluated.

The experimental results are also presented in Table 3 and Table 2.

When using the vanilla triplet loss (Vanilla Triplet), we could see that triplet network performs slightly worse than our proposed IGTN. One of the reasons may be that dynamic penalty term in Equation 8 of our version can better model the relationship among samples. This leads to more reasonable feature distribution in the ICDF space, which in turn makes the features easier to be separated by the deep neural network.

When the triplet framework is removed (no triplet), we can see that our method is still able to achieve better performances than most of previous statistical methods and learning methods based on handcrafted features, but it is slightly lower than the state-of-the-art deep learning approaches without the metric learning framework. It demonstrates the importance of triplet network framework to produce ICDF representation, which is able to dramatically improve the prediction accuracy.

When learnable color histogram is not applied (no learnable color histogram), the error also increases dramatically. It is mainly due to the relatively coarse image features extracted by the AlexNet, which are insufficient to represent the color and texture features of input image. We can see that the learnable color histogram scheme is a necessary part of our approach to extract representative image features.

In comparison, the base network demonstrates more contribution to the estimation accuracy. When it is removed (no AlexNet), we could see that the error increases by a large margin. It shows the importance of base network to extract the semantic image features. Those semantic image features including texture features and spatial information

Algorithm	Color Checker		NUS	
	Mean	Med	Mean	Med
Rotation [23]	2.02	1.43	2.38	1.55
Ours	**1.58**	**0.92**	**1.85**	**1.24**

Table 3. Evaluation of the different triplet sampling strategy. [23] rotate the anchor image as the positive image. In our work, we take two cropped patches from one image as the anchor and positive image.

are also the key to the illuminant estimation, which are not included in the learnable color histogram features. The importance of semantic features is also confirmed in the work of [38]. Thus, we can see that both semantic features and learnable color histogram features are both the key components to improve the estimation accuracy.

When spatial pyramid pooling mechanism is removed (no SPP), we show that the error slightly goes up without applying multi-scale learning strategy. It is because the multi-scale feature extraction strategy is able to learn both local and global color features, which could better represent the image color features. When setting the *sole* global average pooling layer with different sizes, we can find that the smaller stride size could lead to lower estimation error. It is due to the smaller stride size is able keep more color histogram information. However, it is still necessary to construct a multi-scale pooling mechanism, since we could obtain both global and local color features from such mechanism.

4.3. Discussion on the Triplet Sampling Strategy

In this work, the way of constructing two images with the same illuminant is to take two different crops from one image. We also evaluate this sampling strategy with one in the work of [23]. In their work, the same class images are selected one image and its rotated image. Following their experiment setting, the rotated image is generated by -10,-5,5,10 degrees respectively.

We present the results in Table 3. It can be seen that our approach has much lower angular than [23]. The main reason is that simply rotating an image would make the triplet network group image features according to the image content instead of the illuminant, since the feature difference between one image and its rotated version is relatively small. It would cause the network more sensitive to the variation of image content, which is not a suitable property for the color constancy problem.

4.4. Threshold Analysis

We then evaluate the influence of threshold η in Equation 9 on the estimation performance. The result is shown in Figure 2. It can be seen that the best performance is achieved when setting $\eta = 3$ on both datasets. When setting η to lower value, the estimation error increases dramatically.

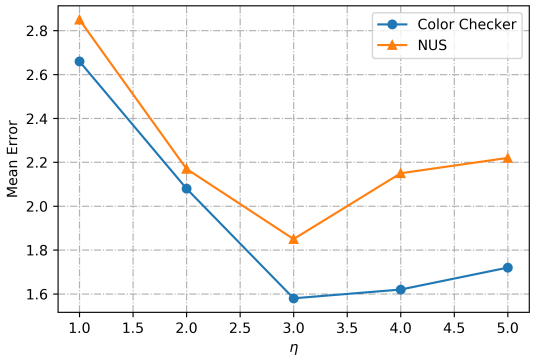

Figure 2. The mean angular errors on two datasets by setting different value of threshold η to determining illuminant differences.

The reason is that the shorter angular distance means closer illuminant value. When defining negative input with closer illuminant value to anchor image, it is difficult for the triplet network to distinguish images by illuminant and thus the network is not able to map image to ICDF space. When setting η to larger value, there is slightly increase on the estimation error. The reason is that more images with a broader range of illuminants would not be considered as having different illuminant, which might leads to the illuminant estimation error.

4.5. Robust Illuminant Estimation

The goal of this experiment is to test how image quality affects the performances of color constancy algorithms. For a given scene with a given illumination, the image acquired by a camera is affected by the camera's parameter settings. For high dynamic range scenes, standard camera often cannot capture the full dynamic range of the scene very well. When using a short exposure time, the image often fail to depict the dark regions very well, and conversely a long exposure will make the bright regions over saturated. In this experiment, we set out to test how the appearance of an image affect illuminant color estimation. For a good algorithm, the estimated result should not be affected by the image quality because it doesn't matter what quality of the image is, the illuminant color of the scene is the same.

The high dynamic range image dataset in [11] contains 97 groups of images. Each group contains a series of images of the same objects taken under the same lighting condition but with different exposure intervals. According to Equation 2, the estimated illuminants should be the same for each image within the same group.

We present four examples in Figure 3 and statistical results in Table 4. We compared our approach with Gray World, White Patch and one state-of-the-art deep learning approach AlexNet-FC4 which is pre-trained on the NUS

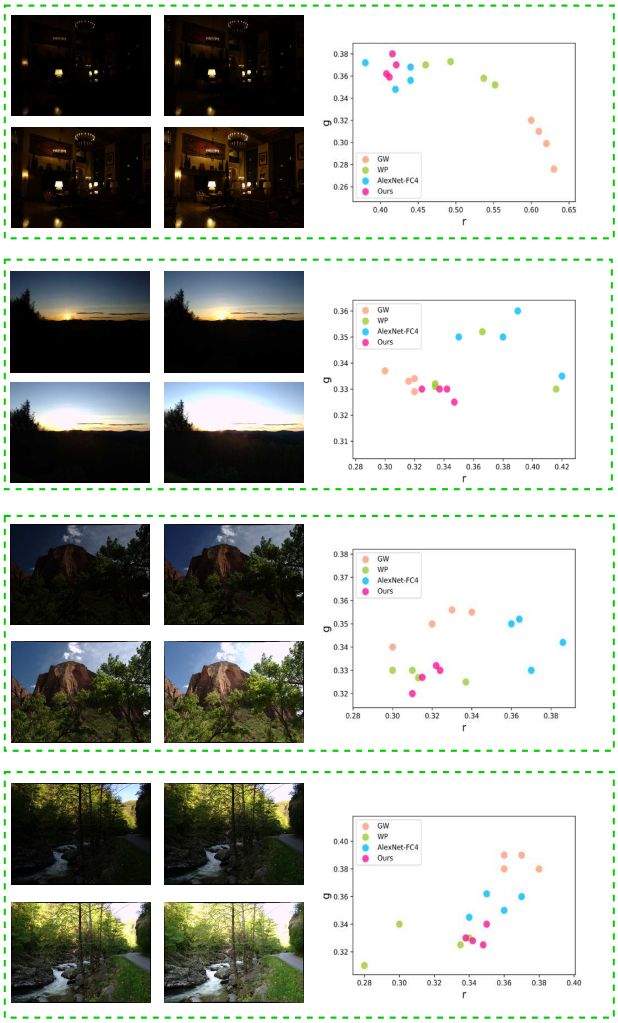

Figure 3. Each scene has 4 images taken with different exposure times. The chromaticity values estimated from each image using different methods are plotted alongside each group of images. It is seen that whilst the estimated illuminant chromatic values of others methods spread widely, our results cluster close together, indicating the more consistent and robust performances of our approach.

dataset [1]. It can be seen from Figure 3 that our approach is able to predict more consistent result when the exposure changes. Statistical analysis also proves the consistency of our approach. We calculate the average variances $(\bar{\sigma_r}, \bar{\sigma_g})$ of r and g channels to measure the spread of the estimated results. The lower value of the average variances means the more consistent estimation results. In Table 4, we can see that our approach is able to predict more consistent illuminant value when the exposure time changes especially when comparing with one deep learning approach (AlexNet-FC4). It is due to the successful usage

[1] https://github.com/yuanming-hu/fc4

	$\bar{\sigma_r}$	$\bar{\sigma_g}$
Gray World [7]	2.72×10^{-4}	8.52×10^{-5}
White Patch [6]	5.86×10^{-4}	7.16×10^{-5}
AlexNet-FC4 [22]	3.48×10^{-4}	2.89×10^{-5}
Triplet Network	3.89×10^{-4}	2.56×10^{-5}
Ours (no triplet)	4.58×10^{-4}	3.90×10^{-5}
Ours (full)	2.62×10^{-4}	2.27×10^{-5}

Table 4. Average variance of r and g channels on a HDR dataset. Lower value means the prediction result is less influenced by the variation of exposure time.

of metric learning strategy to produce ICDF representation, which makes the final illuminant estimation less sensitive to the variation of image content.

5. Concluding Remarks

In this paper, we have introduced a new perspective on color constancy, where a desired color feature should be discriminative and also be content-insensitive. We achieve this by constructing a Illuminant-Guided Triplet Network to learn Illuminant Consistent and Discriminative Feature. In the experiment, our approach is compared with other state-of-the-art methods on the two public datasets, our method is demonstrated to give superior performances. Furthermore, we evaluated the robustness of our method and demonstrated that compared with other methods in the literature, our method achieves more consistent results against variations in camera parameter. In the future work, we will consider to extend our approach to the multi-iiluminant scenario, which should be a more realistic problem in our daily life.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants No. 61902253.

References

[1] K. Barnard. Improvements to gamut mapping colour constancy algorithms. In *European conference on computer vision*, pages 390–403. Springer, 2000.

[2] J. T. Barron. Convolutional color constancy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 379–387, 2015.

[3] J. T. Barron and Y.-T. Tsai. Fast fourier color constancy. In *IEEE Conf. Comput. Vis. Pattern Recognit*, 2017.

[4] S. Bianco, C. Cusano, and R. Schettini. Color constancy using cnns. *arXiv preprint arXiv:1504.04548*, 2015.

[5] S. Bianco, C. Cusano, and R. Schettini. Single and multiple illuminant estimation using convolutional neural networks. *IEEE Transactions on Image Processing*, 26(9):4347–4362, 2017.

[6] D. H. Brainard and B. A. Wandell. Analysis of the retinex theory of color vision. *JOSA A*, 3(10):1651–1661, 1986.

[7] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980.

[8] A. Chakrabarti, K. Hirakawa, and T. Zickler. Color constancy with spatio-spectral statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(8):1509–1519, 2012.

[9] 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.

[10] 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*, pages 1000–1008, 2015.

[11] M. D. Fairchild. The hdr photographic survey. In *Color and Imaging Conference*, volume 2007, pages 233–238. Society for Imaging Science and Technology, 2007.

[12] G. D. Finlayson. Corrected-moment illuminant estimation. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1904–1911. IEEE, 2013.

[13] G. D. Finlayson and E. Trezzi. Shades of gray and colour constancy. In *Color and Imaging Conference*, volume 2004, pages 37–41. Society for Imaging Science and Technology, 2004.

[14] D. A. Forsyth. A novel algorithm for color constancy. *International Journal of Computer Vision*, 5(1):5–35, 1990.

[15] B. Funt, V. Cardei, and K. Barnard. Learning color constancy. In *Color and Imaging Conference*, volume 1996, pages 58–60. Society for Imaging Science and Technology, 1996.

[16] B. Funt and L. Shi. The rehabilitation of maxrgb. In *Color and Imaging Conference*, volume 2010, pages 256–259. Society for Imaging Science and Technology, 2010.

[17] B. Funt and W. Xiong. Estimating illumination chromaticity via support vector regression. In *Color and Imaging Conference*, volume 2004, pages 47–52. Society for Imaging Science and Technology, 2004.

[18] P. V. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp. Bayesian color constancy revisited. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[19] A. Gijsenij and T. Gevers. Color constancy using natural image statistics and scene semantics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):687–698, 2011.

[20] A. Gijsenij, T. Gevers, and J. Van De Weijer. Generalized gamut mapping using image derivative structures for color constancy. *International Journal of Computer Vision*, 86(2-3):127–139, 2010.

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[22] Y. Hu, B. Wang, and S. Lin. Fc 4: Fully convolutional color constancy with confidence-weighted pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4085–4094, 2017.

[23] S. Huang, Y. Xiong, Y. Zhang, and J. Wang. Unsupervised triplet hashing for fast image retrieval. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pages 84–92. ACM, 2017.

[24] H. R. V. Joze and M. S. Drew. Exemplar-based colour constancy. In *BMVC*, pages 1–12, 2012.

[25] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] B. Kumar, G. Carneiro, I. Reid, et al. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5385–5394, 2016.

[27] E. H. Land and J. J. McCann. Lightness and retinex theory. *Josa*, 61(1):1–11, 1971.

[28] Z. Lou, T. Gevers, N. Hu, M. P. Lucassen, et al. Color constancy by deep learning. In *BMVC*, pages 76–1, 2015.

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

[30] Y. Qian, K. Chen, J. Nikkanen, J.-K. Kämäräinen, and J. Matas. Recurrent color constancy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5458–5466, 2017.

[31] C. Rosenberg, A. Ladsariya, and T. Minka. Bayesian color constancy with non-gaussian models. In *Advances in neural information processing systems*, pages 1595–1602, 2004.

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

[33] W. Shi, C. C. Loy, and X. Tang. Deep specialized network for illuminant estimation. In *European Conference on Computer Vision*, pages 371–387. Springer, 2016.

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[35] R. Stanikunas, H. Vaitkevicius, and J. J. Kulikowski. Investigation of color constancy with a neural network. *Neural Networks*, 17(3):327–337, 2004.

[36] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1891–1898, 2014.

[37] J. Van De Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *IEEE Transactions on image processing*, 16(9):2207–2214, 2007.

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

[39] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014.

[40] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. *arXiv preprint arXiv:1505.00687*, 2015.

[41] Z. Wang, H. Li, W. Ouyang, and X. Wang. Learnable histogram: Statistical context features for deep neural networks. In *European Conference on Computer Vision*, pages 246–262. Springer, 2016.