Year 5 Final Project Reports 2016-2017

VDI

Center for Visual & Decision Informatics

A National Science Foundation University Cooperative Research Center

Platform Invariant Low-level Image Processing

Contents

Personnel

Principal Investigators:

Professor Moncef Gabbouj, Tampere University of Technology

Professor Serkan Kiranyaz, co-PI

Graduate Students:

Çağlar Aytekin, PhD student at Tampere University of Technology

Executive Summary/Abstract

Objectives:

- To research novel Neural Network structures that outperform the state-of-the-art in low-level image processing problems.
- To collect a database in order to enable evaluation of platform invariance.
- To test the platform invariance of the developed Neural Network based image processing algorithms.
- To develop methods to achieve the platform invariance.

Methods:

- A novel Convolutional Neural Network-based illuminant estimation algorithm was proposed. The key elements behind the novelty of the method are: 1) the method includes a multi-resolution database population technique 2) Thanks to 1, a very deep CNN could be trained from scratch, 3) a progressive training methodology was proposed that trains the CNN patch-wise first for a good initialization of its parameters and mean-wise to simulate real-testing conditions.
- A database was collected that involves images taken by three different cameras in both lab environment and in the field. All the scenes have been captured by three different cameras and their white points were noted separately. For lab images, there exists different illuminations for the same scene. There is a separate field image set (Field 2) that involves images taken only by a single camera.
- A 3 step evaluation strategy was proposed to assess
    - 1) camera invariance: A network is trained in one camera, validated on another and tested on a third camera. This only evaluates the camera invariance since all the scenes in training, validation and test sets are the same.
    - 2) camera and scene invariance: A network is trained on two different cameras, validated on a separate one and tested on field2 images which are taken by the camera used in validation. This evaluates both camera and scene invariance since the scenes that test set include are entirely different.

    - 3) camera and scene Invariance from single camera: A network is trained and validated on partitions of field2 set and tested on others. This constitutes the most challenging scenario and evaluates if both invariances can be achieved from using only one camera in training and validation where no information, assumptions can be made in test scenes.
- A linear transformation based color-conversion strategy was adopted to achieve camera invariance. The algorithm is based on a least squares estimation of the color conversion matrix that is used to transform RGB values of a scene taken by a camera to the RGB vales of the same scene taken by another camera.

Results:

- The proposed Convolutional Neural Network based method performed the state-of-the-art in well-known color constancy datasets.
- Another CNN based method and our method was evaluated with the above-mentioned evaluation strategies. It was observed that our method outperforms the baseline CNN method in scene invariance thanks to its deeper structure, multi-resolution population strategies and two-stage training procedure. However, both methods as is could not achieve camera invariance.
- The linear color conversion approach was applied for each CNN-based method and it was observed that camera invariance could be achieved for both methods.
- Our proposed CNN-based method combined with color conversion approach yields top performances and can achieve both scene and camera invariances.

Conclusions and Future Work:

- Database population in multi-resolution helps improving the CNN-based methods' performance since the more uncorrelated data leads to better generalization in CNN's. Moreover, multiresolution helps representing the same scene in different scales where the information contained is richer in higher scales.
- Deeper CNN's also enable better generalization. We could train deeper CNN's thanks to database population.
- CNN's used as is cannot handle camera invariance implicitly unless some information about the cameras are not provided beforehand.
- A simple least-squares based color conversion method –obtained via the camera spectral sensitivities- empirically proved to achieve camera invariance to a successful extent.

Related Work and Differences of the Proposed Methods

INTEL-TUT Database:

One of the earliest datasets for color constancy (CC) was provided in [1]. To capture the images in this dataset, a Sony DXC-930 3CCD camera was used. The dataset includes the spectral response of the camera, provided in steps of 4 nm within the interval 380-780 nm. In addition, a total of 1995 surface reflectance coefficients covering a variety of surfaces, and illuminant spectra, with 11 lab illuminants and 81 real-world illuminations, are also included in the dataset. The dataset also contains additional images capturing 50 scenes, with minimal specularities, non-negligible dielectric specularities, and metallic specularities, along with fluorescent scenes. Each scene was captured under 11 lab illuminants and after removing images that include calibration deficiencies, 529 images were provided. For CC techniques that assume diffuse reflectance, the usable part of this dataset includes only 30 scenes out of the 50 provided. Although this dataset shares illuminant spectra from real-world illuminations, the captured images do not include outdoor images and are restricted to the ones lit under lab illuminants.

The hyperspectral images provided in [1] contain only lab images. In [2], hyperspectral images of real-world images were collected by a progressive-scanning monochrome digital camera Pulnix TM-1010 within the range of 400-720 nm sampled at 10 nm intervals. 30 scenes were captured from rural and urban areas. Together with the images of the scenes, light reference images were also captured by the same apparatus but pointed at a uniformly reflecting flat surface. Noise and transmission effects were later corrected and finally, spectral reflectance of each pixel was extracted via normalizing the corrected image with a white standard.

A large dataset was provided in [3] that contains 11000 images of indoor/outdoor scenes shot by a Sony VX-2000 digital video camera. The scenes were captured in two locations to include variations in geography and weather conditions. The scene illumination was measured by a smooth gray sphere. The shortcoming of this dataset is that it includes low-resolution images that were subject to correction. Moreover, it was claimed in [9] that since the images were frames extracted from video sequences, only ~600 of them are truly uncorrelated.

A wide variety of images was collected in the dataset provided by [9], which contains 246 indoor and 322 outdoor images. The scenes were captured by Canon 1D and Canon 5D DSLR cameras. Although this dataset includes 2 different cameras, the same scenes were not captured by both cameras, moreover, only 86 images were taken by Canon 1D. The images contain clipped pixels, are nonlinear and demosaiced. To overcome these problems, a reprocessed version of this dataset is provided in [4], which includes almost RAW linear images with the least amount of processing.

Another dataset with 105 scenes was captured in [5] with a Nikon D700 camera. Up to 9 images of different exposures (with 1 exposure value difference between consecutive images) were captured using the camera's auto-bracketing. The raw images are then processed in two different ways. First, almost-raw base images were created, one image per exposure. Next, a set of high dynamic range (HDR) images were created from the base images. To measure the illumination ground-truth, 4 color-checkers were placed in the scene with varying angles and the median of measurements were used. The images with a high-level variation in color-checker measurements were discarded as the dataset targeted uniformly illuminated scenes.

In contrast, the dataset in [6] was specifically designed to contain multiple illumination in a scene. The dataset contains 9 outdoor and 59 lab images taken by a Sigma SD10 camera with Foveon X3 sensor. The ground-truths for outdoor images were collected with the several gray balls. Indoor images were constructed by illuminating 7 different objects using a combination of two different halogen lamps under 4 color filters. Misaligned images of the same scenes were discarded. Ground truth for lab images were collected by gray cards.

Another dataset for multiple illumination was provided in [7] containing 20 real-world and 58 lab images taken by a Sigma SD10 camera with Foveon X3 sensor. Lab images were constructed by 6 different pairs of illuminations of 10 scenes. Again, misaligned images were removed. The real-world images were carefully selected to capture scenes with 1 direct and 1 ambient light. Ground-truths were collected by a complicated procedure explained in [7].

A dataset for exploiting videos for CC was provided in [8]. The dataset includes both single-illuminant and double-illuminant videos. A Panasonic HD-TM 700 was used to capture videos. Single illuminant dataset includes 3 outdoor and 6 indoor scenes. The videos were recorded by moving the camera and the ground truth is collected via a gray card present in each video by averaging the measurements within the video. In the double-illuminant dataset, the scenes were lit by two sources and two gray cards were present in the scene where each gray card was lit by a single illuminant.

All of the above datasets are collected by a single camera with the exception of [9]. However, in the latter, the cameras, namely Canon 1D and Canon 5D are quite similar and do not capture the same scene. Besides, there is a large disproportion in the images taken by Canon 1D -86 images- and Canon 5D -482 images. Hence, none of the above datasets is suitable for camera-invariant CC research. For supervised approaches, one might consider training and testing on different datasets for camera invariance. However, in this setup, the impact from scene and illumination variation cannot be distinguished from the impact of camera response variation. Therefore, it would not be possible to know if the camera invariance problem was in fact addressed at all. A database consisting of aligned images of the same scene taken by various cameras is needed to allow testing different hypotheses about achieving camera invariance. The only dataset that provides such images is the NUS dataset presented in [10]. The dataset includes over 1600 indoor and outdoor images taken by 9 different cameras. Each scene was captured by each camera with slight misalignments. Ground truth was measured via color-checker for each image.

The proposed Intel-TUT database [28] is similar to NUS dataset in the sense that it also provides images of the same scene that are captured by different cameras. Moreover, Intel-TUT database has the following novelties:

- The spectral sensitivities of the cameras are provided;
- The spectral power distributions of the light sources used in lab images are provided;
- In the lab images, the same scene was captured under different illuminants.
- Unlike other datasets, a mobile camera is used in our camera set. This adds value to the dataset since mobile cameras are nowadays the most frequently used type of cameras.
- Images taken by mobile camera come with and without color shading correction, enabling further studies on the impact of residual color shading.
- We provide a field test-set which is collected by one of the cameras for a better assessment of CC methods.
- We share per-image color conversion matrices, which can be used as ground truth for light source spectrum estimation.
- Finally, in our database, no color-checkers (unless intentionally placed), gray balls or tripod legs (which could serve as a ground truth clue) are visible.

Color Constancy Methods:

Color constancy is a unique feature of the human visual system, which enables robust perception of an object's true color under changing illumination. Computational color constancy (CC) aims to simulate this feature via computational models. The common approach is first to estimate the color of the illuminating source, and then to discount for it.

Color constancy methods can be categorized as unsupervised and supervised ones according to the way that they estimate the illuminant chromaticity. Several unsupervised methods rely on some assumptions on the scene reflectance statistics. Among these methods, White Patch (WP) [11] assumes that there is a perfectly reflecting object in the scene; Gray World (GW) [12] assumes that the average reflectance in a scene is gray; while Shades of Gray (SoG) [13] assumes that the average chromaticity in the scene is gray when raised to the power of p. This method experimentally searches for the optimal value of p. Gray Edge (GE) [14] algorithm on the other hand assumes that the average reflectance derivatives are gray. Such assumptions were shown to be derived from a single formulation with different parameters. Another group of unsupervised CC algorithms makes assumptions on the physical properties of objects. These methods focus particularly on specular objects in the scene, i.e. objects acting as mirror-like reflectors. In [16], a histogram based decomposition of reflected light into specular and illumination components was investigated. Lee [15] achieves this decomposition in CIE color space and using these results, the works in [17] and [18] show that specular components can be effectively exploited to estimate the color of the light source. Although providing a somewhat general approach to CC, the performance of unsupervised methods is somewhat limited due to the underlying assumptions.

Supervised approaches to color constancy either aim to learn a combination of unsupervised CC methods or learn the light source chromaticity directly. Combination-based methods exploit information such as semantic content of the scene [19] or scene type, e.g. indoor/outdoor scene [20]. Nevertheless, these methods are still dependent on the assumptions of unsupervised methods, as they only learn a combination of unsupervised methods.

Direct supervised methods are free from the assumptions of the unsupervised methods and learns a direct mapping from the input image to the illumination chromaticity. This is achieved by exploiting well-known machine learning algorithms such as neural networks [21], support vector regression [22] and Bayesian framework [23],[24]. Recently, due to their success in other fields, convolutional neural networks are also applied to CC and have shown promising results [25]-[26]. In [26], a three-stage learning is employed. First, a deep network was trained for object classification task. Second, this pre-trained network was fine-tuned to regress to an unsupervised CC method's output. Finally, the network parameters obtained by the second training were fine-tuned to regress to ground truth chromaticities. The main drawback of this study is the sub-optimal training procedure. The adaptation of the first deep network, dedicated to object classification, to CC task is suboptimal as the regression is made to the results of an unsupervised method. Moreover, the third network is trained only with thousands of images whereas the first network for classification task is trained with millions. In [25], a more effective use of deep networks is made via directly learning the CC problem through an 8-layer CNN that is trained on 32x32 patches. Considering the high-resolution datasets as in [27], this approach enables a significant population of the dataset from of the order of thousands to millions, making it more suitable for deep learning purposes.

In [25], the network learned per-patch from this populated dataset is later fine-tuned per-image such that the knowledge of the global estimation from local patches is incorporated in the training. Despite the fine-tuning procedure, the initial training on 32x32 patches indirectly affects the final network's performance due to the possible limited scene contents in such small patches for accurate illumination estimation.

In this study, we propose a CNN-based approach to CC by addressing the shortcomings of the previous methods. First, we create an image-pyramid with progressively downscaling the original image at several scales. Then, local patches were extracted from all resolutions, which were then used as training samples. Such an approach eliminates the effect of limited content in small patches to some extent.

Moreover, the dataset is populated in greater numbers, which allows training a very deep network in an end-to-end manner. As expected, exploiting such a deep network helps to achieve better generalization accuracy.

Methods

A. Deep Color Constancy

Dataset Population

A limiting issue about application of deep learning to color constancy (CC) problem is the small size of available datasets. For example, a widely-used database, Gehler-Shi [27], consists of only 568 images. However, if one assumes that there is only one illuminating source in a scene, an image can be cropped to many local patches all of which have the same ground truth, i.e. the chromaticity of the single illuminating source. Therefore, the small number of high-resolution images can be populated to a much greater number of local patches that can be exploited in deep learning based approaches [25]. The problem with this approach is the fact that local patches may sometimes correspond to regions that only cover very limited contextual information. Therefore, one should be aware of the tradeoff between data population and loss of useful context information when selecting the size of local patches. We propose handling this tradeoff as follows. We keep the original size of the images in datasets and form an image pyramid of 5 images by progressively scaling the original image 5 times with 0.75 scale. From each image in the pyramid, we extract 64x64 patches and form a database out of these patches. This way, we make use of a wide variety of scales and especially in the higher scales, the 64x64 local patches are not expected to correspond to regions with limited context as they actually corresponding to much larger regions in the original image. As expected, the patches in higher scales cover a large variety of context, hence possess a richer color distribution. The above dataset population results in a lot of patches which is suitable to train very deep architectures.

Preprocessing

The images in the Gehler-Shi dataset are taken by two cameras, one with black level 0 and another with 129. This black level is extracted from the images taken by the second camera. As it is a common procedure in deep learning, the 12 bit images are normalized such that the maximum possible intensity is 1. A global histogram stretching is also applied to each image in order to gain robustness to illumination intensity. The stretching is only applied to the illumination channel in HSV color space. Note that the above preprocessing steps do not distort the chromaticity of the pixels, hence do not degrade the performance of illuminant chromaticity estimation.

Network Architecture

The convolutional neural network exploited consists of three blocks of convolution, activation and pooling layers. The convolution filters are selected to be of size 5x5, 5x5 and 4x4 in the first, second and third layers, respectively. In all layers, the number of convolutional filters are selected as 32. The activation functions are rectified linear units, which are commonly used in deep learning. All pooling operations are performed by selecting the maximum in a 2x2 patch with stride of two. Finally, a fully connected layer is employed in the end with 256 hidden neurons. The network consists of 6 weighted layers and has approximately 160000 parameters.

Loss and Error Functions

Training is conducted via backpropagating the Euclidean difference (the loss function) between the ground truth illumination chromaticity $\boldsymbol{\rho_{GT}}$ and the estimated one by CNN, $\boldsymbol{\rho_e}$. The error function on the other hand is the commonly used recovery angular error ($\boldsymbol{RAE}$) calculated as in Eq. (1). The Euclidean loss is a good approximation of the $\boldsymbol{RAE}$. In fact, given that $\boldsymbol{\rho_{GT}}$ and $\boldsymbol{\rho_e}$ are $\boldsymbol{l_2}$ normalized, minimizing the Euclidean loss exactly corresponds to minimizing $\boldsymbol{RAE}$.

$$\boldsymbol{RAE(\rho_{GT}, \rho_e) = \cos^{-1}\left(\frac{\rho_{GT} \cdot \rho_e}{\|\rho_{GT}\|\|\rho_e\|}\right)} \qquad (1)$$

Three-stage Training

In training, 3-fold cross validation is utilized based on the folds suggested by the dataset. We employ a three-stage training strategy. The first training is applied directly on the 64x64 image patches with batch size 256. Due to the large number of patches available, this training helps achieving effective training of CNN.

The second training is applied on the 568 samples (original images) where the estimated illumination chromaticity is determined by taking the prediction averages of all 64x64 patches corresponding to the image at hand. This training is utilized via fine-tuning the parameters of the CNN obtained by the first training.

The third training is similar to the second with the only difference that the median of the predictions from the local patches is taken as the final estimate. The third training fine-tunes the CNN parameters obtained by the second training.

During testing, the model obtained by the third training is used to estimate the illumination of each 64x64 patch corresponding to an image and the final estimate is the median of these local estimates.

B. Least Squares Color Conversion

Let RGB_i be a matrix of RGB values of camera i for various reflectances under white light.

$$\widetilde{RGB}_1 = RGB_2\, xCCM_{1,2} \qquad (2)$$

In Eq. 2, $CCM_{1,2}$ is the color conversion matrix representing a linear conversion of camera 2's colors to Camera 1. $\widetilde{RGB}_i$ is the estimated colors of the image taken by camera i. The least square estimation of $CCM_{1,2}$ corresponds to minimizing the error:

$$\min \sum \left(RGB_1 - \widetilde{RGB}_1\right)^T \left(RGB_1 - \widetilde{RGB}_1\right) \qquad (3)$$

The minimization in Eq. 3 has a closed form solution for the color conversion matrix as follows.

$$CCM_{1,2}=RGB_2{}^{\dagger}xRGB_1 \qquad (4)$$

The matrix RGB_i is obtained by multiplying the camera spectral sensivity of camera i and some well-known object surfance reflectances.

Experimental Results

A. Performances in INTEL-TUT Database

Table 1 illustrates the mean RAE of baseline unsupervised methods WP [11], GW [12], SoG [13] and GE [14]. As one may observe, even in unsupervised methods, there is a varying performance across the different cameras.

Table 1 Mean RAE of Baseline Unsupervised Methods.

	Canon	Nikon	Mobile
WP	5.2786	4.9145	3.9149
GW	4.9718	5.7111	4.4011
SoG	3.8278	4.2692	3.6671
GE	3.6767	3.5690	4.3821

Tables 2-4 shows the RAE of Bianco's Method [25] under Evaluation Strategy 1. Every fold indicates the errors in Training, Validation and Test sets respectively. The camera that each set corresponds to in each fold is represented by the first letters of the cameras. Table 2-4 corresponds to experiments with high resolution images, low resolution images and low resolution images with color conversion (CCM) respectively.

Table 5 shows the RAE of our method [29] for low resolution images with color conversion under Evaluation Strategy 1. It is clearly observed that color conversion strategy can clearly achieve camera invariance to a notable extent. Moreover, the errors obtained with our method outperforms the ones obtained by [25] with a large gap.

Table 2. (Bianco's Method, Evaluation 1: Camera Invariance) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of training, validation and test errors for each fold –High Resolution Images: Canon (C), Nikon (N), Mobile (M). Epochs of selected models are indicated next to folds.

	Fold 1			Fold 2			Fold 3			Fold 4			Fold 5			Fold 6		
	N	M	C	M	N	C	N	C	M	M	C	N	C	N	M	C	M	N
$\overline{RAE}$	3.902	3.426	5.246	3.08	4.399	5.353	4.811	4.683	3.978	4.206	4.528	5.357	4.586	5.188	4.56	5.215	4.725	5.064
$\widetilde{RAE}$	3.151	2.778	4.803	2.477	3.488	4.836	3.737	4.135	3.61	3.541	3.729	4.097	3.799	4.587	4.268	4.602	4.044	4.552
RAE_M	17.888	15.68	16.035	14.178	16.451	14.782	20.924	15.549	18.282	16.248	14.589	17.022	15.348	19.344	16.128	16.34	14.771	17.867

Table 3. (Bianco's Method, Evaluation 1: Camera Invariance) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of training, validation and test errors for each fold -1080p Images: Canon (C), Nikon (N), Mobile (M). Epochs of selected models are indicated next to folds.

	Fold 1			Fold 2			Fold 3			Fold 4			Fold 5			Fold 6		
	N	M	C	M	N	C	N	C	M	M	C	N	C	N	M	C	M	N
$\overline{RAE}$	3.96	3.283	5.437	3.218	4.29	5.729	5.355	4.399	5.046	4.746	4.408	5.583	5.18	4.385	3.69	5.18	3.69	4.385
$\widetilde{RAE}$	3.259	2.669	4.892	2.477	3.408	5.093	4.863	3.762	4.688	4.322	3.574	4.432	4.449	3.533	2.998	4.449	2.998	3.533
RAE_M	17.684	17.356	14.847	14.183	17.265	15.404	18.07	16.208	16.872	14.725	14.877	17.221	15.444	17.266	16.312	15.444	16.312	17.266

Table 4. (Bianco's Method, Evaluation 1: Camera Invariance with CCM) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of training, validation and test errors for each fold -1080p Images: Canon (C), Nikon (N), Mobile (M). Epochs of selected models are indicated next to folds.

	Fold 1			Fold 2			Fold 3			Fold 4			Fold 5			Fold 6		
	N	M	C	M	N	C	N	C	M	M	C	N	C	N	M	C	M	N
$\overline{\boldsymbol{RAE}}$	3.682	3.893	3.674	3.029	3.153	3.104	3.682	3.674	3.893	3.029	3.104	3.153	3.296	3.386	3.495	3.272	3.506	3.332
$\widetilde{\boldsymbol{RAE}}$	2.99	3.259	2.77	2.343	2.47	2.207	2.99	2.77	3.259	2.343	2.207	2.47	2.532	2.73	2.799	2.379	2.776	2.64
$\boldsymbol{RAE_M}$	17.599	16.25	17.095	14.024	14.254	14.113	17.599	17.095	16.25	14.024	14.113	14.254	14.806	15.89	13.803	14.103	13.399	15.22

Table 5. (Our Method, Evaluation 1: Camera Invariance with CCM) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of training, validation and test errors for each fold -1080p Images: Canon (C), Nikon (N), Mobile (M). Epochs of selected models are indicated next to folds.

	Fold 1			Fold 2			Fold 3			Fold 4			Fold 5			Fold 6		
	N	M	C	M	N	C	N	C	M	M	C	N	C	N	M	C	M	N
$\overline{RAE}$	1.787	1.9	1.888	1.517	1.935	1.959	1.786	1.864	1.99	1.681	1.947	2.021	1.595	1.822	1.776	1.509	1.829	1.782
$\widetilde{RAE}$	1.438	1.344	1.327	1.069	1.412	1.477	1.408	1.253	1.361	1.238	1.418	1.508	1.253	1.405	1.399	1.106	1.44	1.455
RAE_M	8.766	14.515	14.722	11.17	11.468	10.602	9.262	13.287	13.889	10.723	10.446	11.153	11.073	11.174	11.11	10.55	10.782	10.669

Table 6-7 shows the RAE of Bianco's Method under evaluation strategy 2, without and with CCM respectively. Table 8 shows the RAE of our method under evaluation strategy 2 with CCM. It is clearly observed that thanks to CCM, the camera invariance is achieved. Scene invariance is better achieved with our method than Bianco's, given the significantly less error in test set.

Table 6. (Bianco's Method, Evaluation 2: Camera and Scene Invariance) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of Training, Validation and Test Set.

	$\overline{RAE}$	$\widetilde{RAE}$	RAE_M
Training (Nikon+Mobile)	5.426	4.452	20.324
Validation (Canon)	4.160	3.396	15.116
Canon 2nd Field	6.281	6.408	24.492

Table 7. (Bianco's Method, Evaluation 2: Camera and Scene Invariance with CCM) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of Training, Validation and Test Set.

	$\overline{RAE}$	$\widetilde{RAE}$	RAE_M
Training (Nikon+Mobile)	3.366	2.705	14.723
Validation (Canon)	3.407	2.58	14.864
Canon 2nd Field	4.676	4.169	23.542

Table 8 (Our Method, Evaluation 2: Camera and Scene Invariance with CCM) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of Training, Validation and Test Set.

	$\overline{RAE}$	$\widetilde{RAE}$	RAE_M
Training (Nikon+Mobile)	1.685	1.259	11.369
Validation (Canon)	1.715	1.27	11.567
Canon 2nd Field	3.939	3.63	18.738

Table 9-10 shows the RAE of Bianco's and our method respectively under evaluation strategy 3. As one can observe we significantly outperform Bianco's method under this evaluation and can achieve camera invariance at a notable level and scene invariance at a satisfactory level in this challenging evaluation.

Table 9 (Bianco's Method, Evaluation 3: Camera and Scene Invariance) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of Training, Validation and Test Set.

	$\overline{RAE}$	$\widetilde{RAE}$	RAE_M
TRAIN	2.716	1.877	21.554
VAL	3.350	2.756	21.452
CANON-FIELD	4.496	3.752	14.357
CANON-ALL	6.503	6.325	20.807
NIKON-FIELD	6.122	5.157	14.911
NIKON-ALL	9.317	7.829	25.617
MOBILE-FIELD	7.188	5.571	20.841
MOBILE-ALL	9.882	8.471	23.115

Table 10 (Our Method, Evaluation 3: Camera and Scene Invariance) Mean, median and maximum recovery angle errors ($\overline{RAE}, \widetilde{RAE}, RAE_M$) of Training, Validation and Test Set.

	$\overline{RAE}$	$\widetilde{RAE}$	RAE_M
TRAIN	1.2877	0.89285	8.697
VAL	2.6013	1.84505	18.2525
CANON-FIELD	3.7458	2.7378	13.58235
CANON-ALL	4.704	3.689	22.8215
NIKON-FIELD	3.3337	2.355	12.04105
NIKON-ALL	4.715	4.1237	21.5262
MOBILE-FIELD	3.7281	2.9039	13.41245
MOBILE-ALL	5.0082	4.1834	21.8893

Functionality of Innovation(s)

There are several innovations in the project. First, our CNN-based method proposes a database population technique that both drastically increases the number of image patches and handles the content loss with a multiresolution approach. Second, the database is an innovation itself, since it provides the tools for evaluating the camera invariance that is desired to be achieved. The first innovation enables state-of-the-art performance since -thanks to many images with enough content information- we can train a very deep network end-to-end. It is well-known that deeper networks achieve better generalization in general. The second innovation resulted in a tool to evaluate camera invariance and we have used this dataset to evaluate our method's as well as another CNN based method's camera invariance. We have observed that both methods with no additional processing cannot achieve camera invariance. Then, we have adopted a linear color conversion approach as a preprocessing method on images and then thanks to the evaluation strategies we have empirically proven that this approach achieves camera invariance for both CNN-based color constancy algorithms. Moreover, with another strategy (camera and scene invariance tests) we also have shown that our method –thanks to its own innovations-achieve much better results than state-of-the-art.

Conclusions and Recommendations

Database population in multi-resolution helps improving the CNN-based methods' performance since the more uncorrelated data leads to better generalization in CNN's. Moreover, multiresolution helps representing the same scene in different scales where the information

contained is richer in higher scales. Deeper CNN’s also enable better generalization. We could train deeper CNN’s thanks to database population. CNN’s used as is cannot handle camera invariance implicitly unless some information about the cameras are not provided beforehand.
A simple least-squares based color conversion method –obtained via the camera spectral sensitivities- empirically proved to achieve camera invariance to a successful extent. The only work that remains is to populate the dataset in order to achieve better generalization error following the well-known fact that more uncorrelated training images results into the better generalization error in CNN-based approaches.

Impact and Uses/Benefits

A supervised camera invariant color constancy algorithm is immediately applicable to camera industry. With such method, the dependency on unsupervised methods’ somewhat general but inferior performance can be eliminated. Moreover, the need of training a machine for each specific camera is also unnecessary thanks to color-conversion approach. With this method, one can only train a color constancy algorithm for one camera and immediately apply this CNN for other methods. The only need is the color conversion preprocessing step which loads almost no computational complexity on top of the color constancy algorithm. Although since this method is also supervised, one needs the camera spectral sensitivity of the cameras used. This information is very easy to be obtained compared to the very heavy task of collecting a new database for each camera and training a network for each camera.

List of References

[1] K. Barnard, L. Martin, B. Funt and A. Coath, “A Data Set for Color Research,” Color Research & Application, vol. 27, no. 3, pp. 147-151, 2002.

[2] S. M. Nascimento, F. P. Ferreira and D. H. Foster, “Statistics of Spatial Cone-Excitation Ratios in Natural Scenes,” JOSA A, vol. 19, no. 8, pp. 1484-1490, 2002.

[3] F. Ciurea and B. Funt, “A Large Image Dataset for Color Constancy Research,” in Proc. CIC, 2003, pp. 160-164.

[4] 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/

[5] B. Funt, “HDR Dataset,” accessed from http://www.cs.sfu.ca/~colour/data/

[6] A. Gijsenij, R. Lu and T. Gevers, “Color Constancy for Multiple Light Sources,” IEEE Trans Image Process., vol. 21, no. 2, pp. 697-707, 2012.

[7] S. Beigpour, C. Riess, J. van de Weijer, E. Angelopoulou, “Multi-illuminant Estimation with Conditional Random Fields,” IEEE Trans Image Process vol. 23, no.1, pp. 83-96, 2014.

[8] V. Prinet, D. Lischinski, M. Werman, “Illuminant Chromaticity from Image Sequences,” in Proc. ICCV, 2013, pp. 3320-3327.

[9] P. V. Gehler, C. Rother, A. Blake, T. Minka and T. Sharp, “Bayesian Color Constancy Revisited,” in Proc. CVPR, 2008, pp. 1-8.

[10] 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.

[11] E. H. Land and J. J. McCann, “Lightness and Retinex Theory,” JOSA, vol. 61, no. 1, pp. 1-11, 1971.

[12] G. Buchsbaum, “A spatial Processor Model for Object Color Perception,” vol. 310, no. 1, pp. 1-26, 1980.

[13] G. D. Finlayson and E. Trezzi, “Shades of Gray and Colour Constancy,” in Proc. CIC, 2004, pp. 37-41.

[14] J. Van de Weijer, T. Gevers, A. Gijsenij, “Edge-based Color Constancy,” IEEE Trans. Image Process., vol. 16, no. 9, pp 2207-2214, 2007.

[15] H. C. Lee, “Method for Computing the Scene-Illuminant Chromaticity from Specular Highlights,” J. Opt. Soc. Am. A, vol. 3, pp. 1694-1699, 1986.

[16] G. J. Klinker, S. A. Shafer and T. Kanade, “The Measurement of Highlights in Color Images,” Int. J. Comput. Vision, vol. 2, pp. 7-32, 1988.

[17] G. D. Finlayson and G. Schaefer, “Solving for Color Constancy using a Constrained Dichromatic Reflection Model,” Int. J. Comput. Vision, vol. 42, no. 3, pp. 127-144, 2001.

[18] R. T. Tan and K. Ikeuchi, “Separating Reflection Components of Textured Surfaces Using a Single Image,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 2, pp. 178- 193, 2005.

[19] J. Van de Weijer, C. Schmid and J. Verbeek, “Using High-level Visual Information for Color Constancy,” in Proc. IEEE ICCV, 2007, pp. 1-8.

[20] S. Bianco, G. Ciocca, C. Cusano and R. Schettini, “Improving Color Constancy Using Indoor-Outdoor Image Classification,” IEEE Trans. Image Process., vol. 17, no. 12, pp. 2381-2392, 2008.

[21] V. C. Cardei, B. Funt and K. Barnard, “Estimating the Scene Illumination Chromaticity by Using a Neural Network,” J. Opt. Soc. Am., vol. 19, no. 12, 2002.

[22] B. Funt, W. Xiong, “Estimating Illumination Chromaticity via Support Vector Regression,” in Proc. CIC, 2004, pp. 47-52.

[23] C. Rosenberg, A. Ladsariya and T. Minka, “Bayesian Color Constancy with non-Gaussian Models,” in Proc. NIPS, 2003.

[24] P. V. Gehler, C. Rother, A. Blake, T. Minka and T. Sharp, “Bayesian Color Constancy Revisited,” in Proc. CVPR, 2008, pp. 1-8.

[25] S. Bianco, C. Cusano, R. Schettini, “Color Constancy Using CNNs,” in Proc. IEEE CVPRW, 2015, pp. 81-89.

[26] Z. Lou, T. Gevers, N. Hu and M. Lucassen, “Color Constancy by Deep Learning,” in Proc. BMVC, 2015.

[27] 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/

[28] C. Aytekin, J. Nikkanen and Moncef Gabbouj, “INTEL-TUT Dataset for Camera Invariant Color Constancy Research,” arXiv:1703.09778 . (Submitted to IEEE TIP).

[29] C. Aytekin, J. Nikkanen and Moncef Gabbouj, “Deep Multi-Resolution Color Constancy” to be presented at IEEE ICIP 2017.