arXiv:1801.06302v1 [cs.CV] 19 Jan 2018

Fully Point-wise Convolutional Neural Network for Modeling Statistical Regularities in Natural Images

Jing Zhang[1], Yang Cao[2], Yang Wang[2], Zheng-Jun Zha[2], Chenglin Wen[1] and Chang Wen Chen[3]
[1]Hangzhou Dianzi University, Hangzhou, China
[2]University of Science and Technology of China, Hefei, China
[3]State University of New York at Buffalo, Buffalo, U.S.A.

Abstract

Modeling statistical regularities is the problem of representing the pixel distributions in natural images, and usually applied to solve the ill-posed image processing problems. In this paper, we present an extremely efficient CNN architecture for modeling statistical regularities. Our method is based on the observation that, by random sampling the pixels in natural images, we can obtain a set of pixel ensembles in which the pixel value is independent identically distributed. This leads to the idea of using $1 * 1$ *(point-wise) convolution kernel instead of* $k * k$ *convolution kernel to learn the feature representation efficiently. Accordingly, we design a novel architecture with fully point-wise convolutions to greatly reduce the model complexity while maintaining the representation ability. Experiments on three applications: color constancy, image dehazing and underwater image enhancement demonstrate the superior performance of our proposed network over the existing architectures, i.e., using 1/10∼1/100 network parameters and computational cost over the state-of-the-art networks while achieving comparable accuracy. Codes and models will be made publicly available*[1].

1. Introduction

Modeling statistical regularities is very important for natural image processing because of its impact on solving the ill-posed problem. Due to the complex, diverse and high-dimensional distributions of the pixels, it is a challenging task to discover and model the intrinsic regularities in natural images.

One feasible solution is to assume the statistical regularities, which utilizes the prior knowledge about the types of distributions that exist, and thus to design specialized algorithms for the specific tasks. For example, many color constancy algorithms work by assuming some regularities in the colors of natural objects viewed under a white light, e.g., gray world [9], gray edge [42], and shades of gray [19]. Besides, by assuming that the surface shading and scene transmission are locally uncorrelated, most of single image dehazing methods are proposed based on various image priors, e.g., color attenuation [46], dark channel [23, 24], and haze line [5].

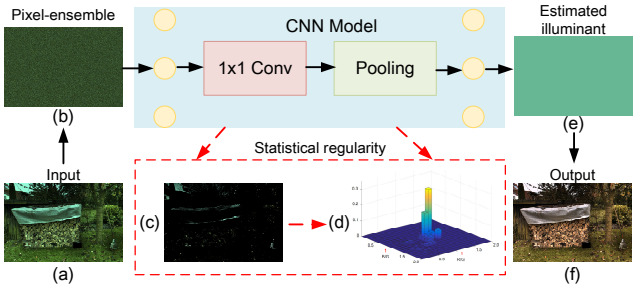

Figure 1. An exemplar illustration of the proposed method for color constancy. (a) An illuminated image. (b)Network input: pixel-ensemble randomly sampled from (a). (c) Re-projections of the most activated pixels after first point-wise convolution layer and pooling layer onto (a). (d) The histogram of corresponding strong activated pixels in intrinsic image. (e) Illuminant prediction. (f) The restored intrinsic image using predicted Illuminant.

Another practicable approach is to learn the statistical regularities, which formulates regression models of how the pixels are distributed, and adapts the model parameters to fit the input image. Thus, the adapted model parameters indeed reveal the statistics of the pixels, while the internal representations of models reflect the individual pixel patterns. This approach makes minimal assumptions on the pixel distributions and thus can result in more general representations [31].

Recently, deep learning has made much success in natural image processing problems like image denoising [10, 43], super-resolution [16] and the most relevant ones with statistical regularities, e.g. color constancy [39] and image dehazing [11]. The hierarchical neural network representa-

[1] https://chaimi2013.github.io/Research/FPC.html

tions of deep learning can make the regression task much simpler, and learn the sub-models of regularities and their corresponding active regions jointly. Therefore, building convolutional neural networks (CNNs) with better representation ability become popular in the development of major image processing methods.

Naturally, a deeper and larger CNN will have better modeling capacity. The most powerful CNNs for image processing tasks usually have dozens of layers and hundreds of channels [39, 28, 44], thus resulting in millions of network parameters. This leads to a high computational cost, which limits its wide range of applications. The goal of this paper is to pursue the best representation accuracy with an extremely efficient CNN structure.

We find that, if we sample the pixels from the original image randomly to generate new pixel ensembles, it is easier for the human eyes to focus on the degraded cue and detect it. Inspired by this observation, we first obtain a set of pixel ensembles by randomly sampling from the input natural image, in which the pixel value is independent identically distributed (i.i.d.). Then, we show that the statistical regularity of the original image is, i) well preserved in the obtained pixel ensembles, and ii) able to be represented by using $1 * 1$ (point-wise) convolution kernels instead of $k * k$ convolution kernels. Accordingly, we build a novel architecture with fully point-wise convolution units to reduce the network complexity while maintaining representation accuracy. Compared with popular structures used in image processing applications, our proposed architecture is lightweight, compact and resisting to overfitting.

A typical example of applying our proposed method on color constancy problem is shown in Fig. 1. The statistics of illuminant color are well preserved in the generated pixel ensemble, and easier to be represented by our proposed fully point-wise CNN. As revealed in Fig. 1(c) and (d), the statistical property is modeled as a regularity in the color distribution of the most activated pixels. Then the regularity leads to an efficient estimate of illuminant in determining a global color constancy result.

We evaluate our proposed method on the three challenging applications: color constancy, image dehazing and underwater image enhancement. Various contrastive experiments show the effectiveness of our proposed networks. Generally, the networks with our designed architecture only need 1/10$\sim$1/100 parameters and computational cost over the state-of-the-art networks while maintaining comparable accuracy.

2. Related Work

2.1. Modeling statistical regularities

Modeling statistical regularities is an important topic in natural image processing. A comprehensive review about literatures in this topic is beyond the scope of this paper. Here we choose the typical image enhancement applications that are most relevant with statistical regularities, i.e., color constancy and image dehazing, and present a brief review about these researches. Most image enhancement methods are based on imaging models, and usually described as the problems of inferring intermediate variables with physical sense, e.g., illumination color and haze transmission, and then removing them from the input images. Since the problems are ill-posed, they are often solved by modeling the statistical regularities of these intermediate variables. In general, the modeling methods can be divided into two categories: assuming some distributions of the pixels or learning some distributions of pixels from training data. One thing that these assumption or learning based models have in common is that they can all be formulated as the following regression problem: inferring a common variable for a set of candidate pixels (we call it as pixel-ensemble in this paper).

Specifically, for color constancy problem, since the illumination color is usually assumed to be global and consistent, the pixel-ensemble indeed includes all image pixels. Nearly all algorithms for this task work by assuming some distributions in the colors of the pixel-ensemble. For example, the gray world algorithm assumes that the illuminant color is the average color of all image pixels. Since then, various methods are proposed to generalize this idea by exploiting gradient information or generalized norms [2, 42], modeling the distribution of color histograms [18], or implicitly reasoning about the moments of colors using PCA [13]. Recently, some work further propose to learn the representation of the statistical regularity in the CNN framework [6, 39, 28].

For image dehazing problem, the pixel-ensemble can be considered as a local image patch. Based the local constant/smoothness assumption, various methods have been proposed to learn or estimate a transmission value for each local patch [23, 41, 11]. For example, He et al. propose a powerful dark channel prior to directly estimate haze transmission and then remove the haze from the input image [23]. Tang et al. investigate four types of haze-relevant features with Random Forests to estimate the transmission [41]. Cai et al. apply a deep CNN framework to regress the transmission from the learning features [11]. Recently, Li et al. present an all-in-one CNN to estimate a transformed variable and consequently recover the dehazed image [33].

In this paper, we propose to generate the pixel ensembles by randomly sampling from the input images. This strategy not only preserves the statistical regularity of the original image, but also makes it easier to be detected. Moreover, for a natural image, the generated pixel ensemble tends to be independent identically distributed. It allows us to use an extremely efficient point-wise CNN to learn the representa-

tion of inherent statistical regularity.

2.2. Efficient architecture design

Recently, the research on efficient architecture design for CNN draws a lot of attention because of the increasing needs of running CNN on embedded devices [27, 45]. For example, Global Convolutional Network [35] reduces the network parameters by decomposing a large convolutional kernel of $k * k$ into two small convolutional kernels of $1 * k$ and $k * 1$. Point-wise convolution, which is introduced in Inception [40] and Resnet [25] as a bottleneck layer, has shown the performance to improve the efficiency. ShuffleNet introduces point-wise group convolution into the bottleneck layer to reduce the parameters and uses a shuffle operation to enable the information exchange across multiple group convolution channels.

Different from the previous work, our proposed architecture is fully point-wise convolutional, and no larger kernels (e.g $3 * 3$ kernels) or group convolutions is used. Besides, there is no need for additional operation after the point-wise convolution. Since it is designed according to the statistical property of the input pixel ensembles which is randomly sampled from original image, the proposed architecture can extract and represent the statistical regularities more efficiently.

3. Proposed Method

3.1. Observation

The key idea of our proposed method is based on the observation of the degraded images. We find that it is easier for the human eyes to detect the degraded cues from the new images generated by randomly sampling from the original images. Some typical examples are shown in Fig. 2. We randomly sample the pixels from the input image patches to produce the new patches. As can be seen, the degraded cues, e.g. illuminate color and haze transmission, are easier to be detected, and the degraded degree can be effectively distinguished.

The main reason for this is that the random sampling strategy actually destroys the spatial correlation between pixels, but retains their statistical properties in the original image patches. This observation leads to the idea of developing an extremely efficient statistical regularity modeling method.

3.2. Motivation

Based on the above observation, we first present to generate a pixel ensemble from the input image by randomly sampling. Therefore, the pixels in the generated ensemble are spatially uncorrelated and independent identically distributed. In the next, we give a presentation that *this i.i.d. property enables the statistical regularity in the pixel-ensemble can be represented by using CNN with* $1*1$ *(point-wise) convolution kernels.*

First, we begin with the illustration by some notations. The input pixel-ensemble X is denoted as:

$$X = \{I(m,n,c), (m,n) \in \Lambda, c = 0, ...C-1\}, \quad (1)$$

where $I(m,n,c)$ is a pixel randomly sampled from the original image, C is the number of image channels and Λ is the pixel index set. We have the mean value of X on channel c as follows:

$$\mu_c = \frac{1}{\Lambda} \sum_{(m,n)\in\Lambda} I(m,n,c). \quad (2)$$

If Λ_s is a subset of Λ, we can get that the mean value of pixels in Λ_s is approximated to μ_c according to the i.i.d. property. Mathematically,

$$\mu_c \approx \frac{1}{|\Lambda_s|} \sum_{(m,n)\in\Lambda_s} I(m,n,c). \quad (3)$$

For the exemplar network architecture in Fig. 3, the input of shape $(2k-1)*(2k-1)$ is first convolved by a $k \times k \times C$ kernel, and then pooled by averaging. Without loss of generality, we assume the stride in convolution layer is 1. The output can be calculated as follows:

$$\begin{aligned} output &= \frac{1}{k^2} \sum_{p=0}^{p=k^2-1} \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} \left(I_p(m,n,c) \times K(m,n,c)\right) \\ &= \frac{1}{k^2} \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} \sum_{p=0}^{p=k^2-1} \left(I_p(m,n,c) \times K(m,n,c)\right) \\ &= \frac{1}{k^2} \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} \left(\left(\sum_{p=0}^{p=k^2-1} I_p(m,n,c) \right) \times K(m,n,c) \right). \end{aligned} \quad (4)$$

Here I_p is the p-th patch with the size of $k * k$ in the input, and $I_p(m,n,c)$ is the pixel located at (m,n) in the c-th channel. $\sum_{p=0}^{p=k^2-1} I_p(m,n,c)$ is the sum of the sampled k^2 pixels and approximated as $k^2\mu_c$ according to Eq.(3). Therefore, we have:

$$\begin{aligned} output &= \frac{1}{k^2} \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} \left(\left(k^2\mu_c\right) \times K(m,n,c) \right) \\ &= \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} \left(\mu_c \times K(m,n,c)\right) \\ &= \sum_{c=0}^{c=C-1} \mu_c \sum_{m=0}^{m=k-1} \sum_{n=0}^{n=k-1} K(m,n,c) \\ &= \sum_{c=0}^{c=C-1} \mu_c \times K_c \end{aligned}, \quad (5)$$

where K_c is the sum of kernel weights for c-th channel. Substitute Eq.(2) into the above equation, we have:

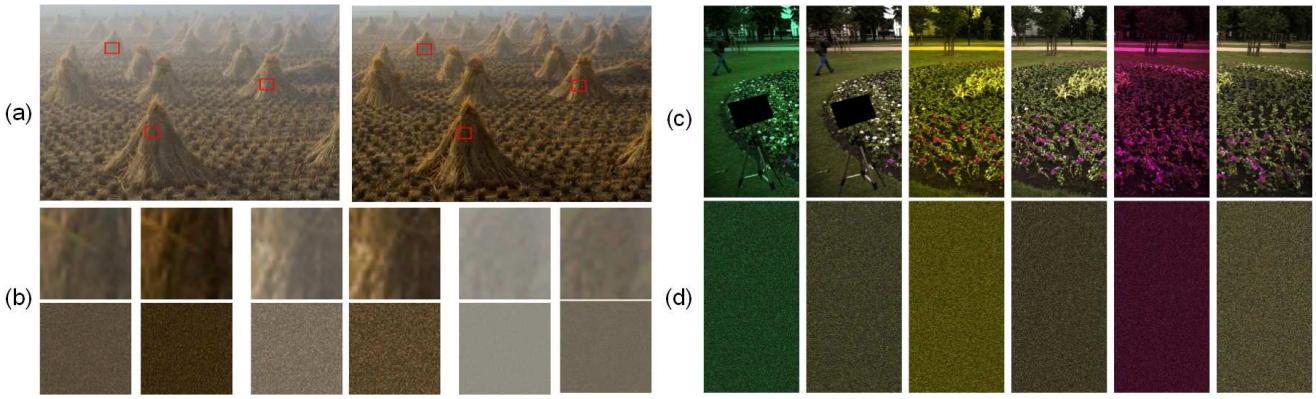

Figure 2. Illustration of the observation. (a) A Hazy image and haze-free image obtained by [11]. (b) Patches and the generated pixel-ensembles. (c) Images under different illuminant and intrinsic images. (d) The generated pixel-ensembles of (c).

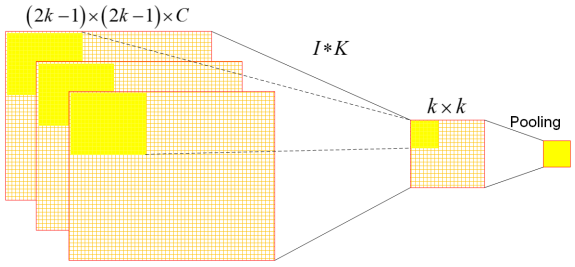

Figure 3. The exemplar network structure.

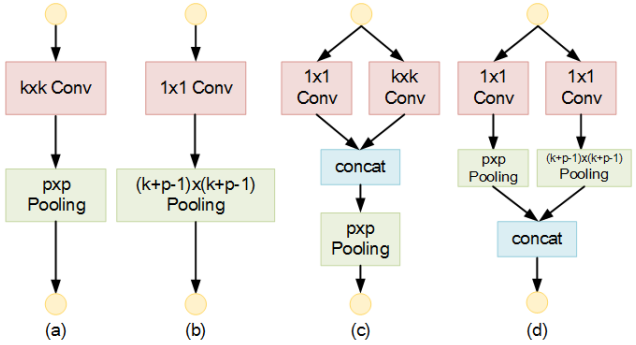

Figure 4. The proposed point-wise convolution unit.

$$output = \frac{1}{(2k-1)^2} \sum_{c=0}^{c=C-1} \sum_{m=0}^{m=2k-1} \sum_{n=0}^{n=2k-1} I(m,n,c) \times K_c. \quad (6)$$

As can be seen, given the pixel-ensembles with i.i.d. property, the large $k * k$ convolution kernel is equivalent to $1 * 1$ convolution kernel. Accordingly, we can design a fully point-wise CNN to model the statistical regularity in the natural images. This architecture requires very few network parameters and is extremely computational efficient. Meanwhile, it should be pointed out that the network is easier to learn the statistical regularity of pixels since the spatial correlation between pixels has been eliminated.

3.3. Point-wise Convolution Unit

Based on the above description, we propose two novel point-wise convolution units, which can be used to specially design a fully point-wise CNN for modeling statistical regularity. We begin with a typical network structure as shown in Fig. 4(a), where a convolution layer with $k*k$ convolution kernels is followed by a pooling layer. According to Eq.(6), since the input is pixel-ensembles that has the i.i.d. property, the $k * k$ convolution can be replaced with the point-wise convolution. To retain the size of receptive field, we enlarge the pooling size from $p*p$ to $(k+p-1)*(k+p-1)$, as shown in Fig. 4(b).

In addition, we present a simple parallel structure including a point-wise convolution layer and a $3 * 3$ convolution layer to extract the multi-scale features [40, 11, 37]. The extracted features are then concatenated and pooled together, as shown in Fig. 4(c). Note that the order of the concatenation layer and pooling layer is interchangeable. Similarly, the $k * k$ convolution can be replaced with the point-wise convolution, which results in two parallel point-wise convolution layers, as shown in Fig. 4(d). According to Eq.(6), we use two pooling layers with different pooling sizes and concatenate the pooled features together.

Taking advantage of the point-wise convolution kernel, our proposed units have less parameters and can be computed efficiently. For example, given the input size $c * h * w$ and the output channels m, the unit in Fig. 4(a) requires cmk^2 parameters, while our unit requires only cm parameters. In addition, the point-wise convolution is indeed a scalar-multiplication and an add-operation. Therefore, its implementations are more efficiently than the ones with large kernels. As for the computational complexity, since

it relies on the explicit network architecture and parameter configuration, we will demonstrate the computational efficiency of the proposed method in Sect.4.

4. Applications

To evaluate the effectiveness of the proposed method, we employ it on three typical image enhancement applications, i.e., color constancy, image dehazing and underwater image enhancement. The three applications correspond to some common problems in many image processing tasks, such as retinex [32], HDR compression [17], low-light enhancement [22] and image defocus [12]. These problems include, I) modeling the statistical regularity in the whole image, II) modeling the statistical regularity in the local patch, and III) a more complicated case that models multiple statistical regularities. For each application, we perform contrastive experiments on benchmarks against the state-of-the-art methods. All the experiments are conducted on a workstation with Nvidia GeForce GTX Titan X (Maxwell) GPUs, and the proposed networks are implemented in Caffe [29].

4.1. Color constancy

An image captured under color illumination can be modeled as follows:

$$I_c = E_c \times R_c, c \in \{R, G, B\}\,, \tag{7}$$

where E_c is the illuminant color and R_c is the RGB value of reflectance under canonical (often white) illumination. Thus the color constancy problem can be formulated as estimating the illuminant color E_c and recovering the reflectance R_c by given an input image I_c .

The evaluation of our color constancy method is performed on two benchmark datasets, i.e., the reprocessed [38] Color Checker Dataset [20] and the NUS 8-Camera Dataset [13]. As in [28, 4], we evaluate the proposed method using a three-fold cross-validation. Several standard metrics are presented based on the angular error between the estimated illuminant (E_{est}) and the ground truth (E_{gt}):

$$\varepsilon = \arccos\left(\frac{E_{est} \cdot E_{gt}}{\|E_{est}\|\,\|E_{gt}\|}\right). \tag{8}$$

The settings of hyper-parameters during training are the same as [6]. If not specified, 128 test patches are used for testing, and median pooling is applied to the network outputs to obtain the estimated illuminant color.

4.1.1 Ablation experiments

Here we present ablation experiments and evaluate the performance of our proposed structure on modeling statistical regularities. Referring to [6], we present our base network (BaseNet) as shown in Table 1. Compared with [6], the BaseNet has a fully CNN structure and three separated prediction sub-nets for RGB channels, respectively. Then, we design a novel fully point-wise CNN (FPC) according to the proposed units in Sect.3.3. The architectures and their numbers of parameters as well as computational complexity are shown in Table 1.

We perform the ablation experiments on Color Checker Dataset and the results are listed in Table 2. As in [6] and [39], we first use the local patches as the input of the BaseNet and achieve a mean angle error of 2.64, which is better than 3.07 in [6]. Then we use the pixel-ensemble gen-

Table 1. Network architectures for color constancy.

Network	Type	Input Size	Num	Filter	Pad	stride
BaseNet	Conv1-1x1	3x32x32	240	1x1	0	1
	Conv1-3x3	3x32x32	240	3x3	1	1
	Concat1	480x32x32	-	-	-	-
	Maxpool1	480x32x32	-	8x8	0	8
	Conv2-(RGB)	480x4x4	40	4x4	0	4
	Conv3-(RGB)	40x1x1	1	1x1	0	1
	Params			$9.29\text{x}10^5$		
	Complexity[1] (Conv layers)			$8.29\text{x}10^6$		
FPC	Conv1-1	3x32x32	240	1x1	0	1
	Maxpool1-1	240x32x32	-	8x8	0	8
	Conv1-2	3x32x32	240	1x1	0	1
	Maxpool1-2	240x32x32	-	10x10	1	8
	Concat1	480x4x4	-	-	-	-
	Conv2-(RGB)	480x4x4	80	1x1	0	1
	Maxpool2	480x4x4	-	4x4	0	4
	Conv3-(RGB)	80x1x1	1	1x1	0	1
	Params			$1.17\text{x}10^5$		
	Complexity (Conv layers)			$3.32\text{x}10^6$		

Table 2. Results of different networks on Color Checker Dataset [38]. LP: local patch. PE: pixel-ensemble. PE+Aug: pixel-ensemble with augmented data.

Method	Mean	Med.	Tri.	Best 25%	Worst 25%	95% Quant.
BaseNet (LP)	2.64	1.98	2.10	0.61	5.84	7.22
BaseNet (PE)	2.40	1.63	1.86	0.56	5.40	6.75
FPC (PE)	2.22	1.51	1.69	0.45	5.12	6.85
FPC (PE+Aug)	2.06	1.46	1.60	0.46	4.66	5.94

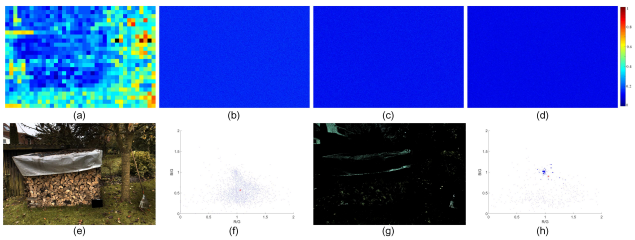

Figure 5. Error maps of different networks. (a)-(b) Error maps of BaseNet tested with local patches and randomly sampled pixel-ensembles. (c)-(d) Error maps of the proposed FPC tested with randomly sampled pixel-ensembles, and with augmented data. (e) The intrinsic image. (f) The distribution of all pixel values in intrinsic image. (g) Re-projections of most activated pixels after the first pooling layer onto the input image. (h) The distribution of the corresponding most activated pixel values in intrinsic image. Dot size represents the accumulated activation strength.

[1]Evaluated with FLOPs, i.e. the number of floating-point multiplication-adds.

erated by randomly sampling from the whole image as the input of the BaseNet, and the mean angle error is reduced to 2.40. This verifies the correctness of our observation in Sect.3.2. In the next, we use the same pixel-ensemble as the input for our proposed fully point-wise CNN, the mean angle error is reduced to 2.22, with only 12.5% network parameters and 40% computational cost of BaseNet. This demonstrates that our proposed structure models the global statistics more efficiently and accurately taking advantage of the i.i.d. property of the generated pixel-ensemble. After using the edge pixels as the augmented data, the mean angle error of FPC is further reduced to 2.06, which is close to the best result achieved by single network structures. As can be seen, the results in terms of Worst-25% metric are significantly improved, which implies that the proposed method achieves a better modeling performance on the hard samples.

To further demonstrate the modeling capacity of each structure, we present the corresponding error maps of different networks in Fig. 5. As can be seen, the modeling errors at each patch are diverse when using original local patches as input. As a contrast, the modeling errors are relatively uniform and small when using the generated pixel-ensembles as input. This implies that the statistical properties in the pixel-ensembles are more stable and easier to be represented.

In addition, for visual inspection of the learned statistics by our proposed FPC, we extract the most activated pixels after the first pooling layer and re-project them onto the input image, as shown in Fig. 5(g). The most activated pixels include the bright pixels, white pixels and the distinct edge pixels. We plot the R/G and B/G values of these pixels on the intrinsic image. As can be seen in Fig. 5(h), these pixels concentrate around (1,1). This reveals that our proposed FPC implicitly learns the regularity that the reflectance of objects corresponding to the most activated pixels tends to be gray. Consequently, this regularity leads to an easy solution for estimating the illuminant color. More results are shown in supplement.

4.1.2 Comparisons with state-of-the-art methods

Here we compare the proposed FPC with previous methods on the NUS 8-Camera Dataset. We report each error metric for each method across all cameras, as done in previous work. Results are summarized in Table 3. Some visualizations results can be found in supplement. For most metrics, FPC achieves comparable results with the state-of-the-art methods, e.g. FC4 [28] and FFCC [4]. This implies that our proposed structure well captures the statistical regularity in the color space, and can serve as an alternative CNN based solution that requires much less network parameters for color constancy.

Moreover, we also compare the computational efficiency with the previous methods. The results in terms of running time are listed in Table 3. Our proposed structure is found to be 10x $\sim$ 100x faster than previous CNN based methods [6, 39, 28]. The proposed FPC processes a single image in 2.7ms with an unoptimized GPU implementation on the Matlab platform, compared to 208ms for [6, 7], 3s for [39], and 25ms for [28]. The advantage on running time of our proposed FPC is due to the fully point-wise convolutional structure, which can be implemented efficiently. Besides, as shown in Fig. 5, the modeling errors for the whole pixel-ensemble are consistent. This implies that we can use a small number of pixel-ensembles instead of all the pixels for testing. This strategy will further boost the computational efficiency of our proposed FPC.

Table 3. Results on NUS 8-Camera Dataset [13].

Method	Params	Time	Mean	Med.	Tri.	Best 25%	Worst 25%	95% Quant.
White-Patch [8]	-	0.16	10.62	10.58	10.49	1.86	19.45	8.43
Edge-based Gamut [1]	-	3.6	4.40	3.30	3.45	0.99	9.83	-
Gray-World [9]	-	0.15	4.14	3.20	3.39	0.90	9.00	3.25
Bayesian [20]	-	97	3.67	2.73	2.91	0.82	8.21	2.88
Natural Image Statistics [21]	-	1.5	3.71	2.60	2.84	0.79	8.47	2.83
Shades-of-Gray [19]	-	0.47	3.40	2.57	2.73	0.77	7.41	2.67
General Gray-World [2]	-	0.91	3.21	2.38	2.53	0.71	7.10	2.49
Bright Pixels [30]	-	-	3.17	2.41	2.55	0.69	7.02	2.48
1st-order Gray-Edge [42]	-	1.1	3.20	2.22	2.43	0.72	7.36	2.46
Cheng et al. [13]	-	0.24	2.92	2.04	2.24	0.62	6.61	2.23
CCC(dist+ext) [3]	-	0.52	2.38	1.48	1.69	0.45	5.85	1.74
Regression Tree [14]	-	0.25	2.36	1.59	1.74	0.49	5.54	1.78
CNN [7]	0.154M	0.208	-	1.73	-	-	-	-
DS-Net(HypNet+SelNet) [39]	4.23M	3.0	2.24	1.46	1.68	0.48	5.28	1.69
AlexNet-FC4 [28]	2.48M	0.025	2.12	1.53	1.67	0.48	4.78	1.66
SqueezeNet-FC4 [28]	2.12M	0.025	2.23	1.57	1.72	0.47	5.15	1.71
FFCC - full,4 channels [4]	-	0.07	1.99	1.31	1.43	0.35	4.75	1.44
FFCC - thumb,2 channels [4]	-	0.0011	2.06	1.39	1.43	0.35	4.75	1.44
Proposed Method	0.117M	0.0027	2.17	1.57	1.66	0.51	4.88	1.70

4.2. Image dehazing

The formation of a hazy image can be described as follows:

$$I_c = J_c t + A_c (1 - t), c \in \{R, G, B\}, \quad (9)$$

where J_c is the target clear image, t is transmission, A_c is atmosphere light. Thus the image dehazing problem can be formulated as estimating haze transmission t at each pixel position and recovering the clear image J_c by given an input image I_c.

We build the synthesized hazy image dataset as in [11]. We first obtain 30,000 haze-free patches by randomly sampling from the images collected from Internet. Then, we generate a total of 300, 000 synthetic hazy image patches according to Eq.(9), and split the synthetic patches into two non-overlapped subsets for training and testing. Mean Square Error (MSE) is applied to evaluate the errors between predicted transmissions and ground truth. The settings of hyper-parameters during training are the same as [11]. More details can be found in supplement.

4.2.1 Ablation experiments

Here we present ablation experiments to design the network for image dehazing. We refer dehazenet [11] as our base-

Table 4. Network architectures for dehazing.

Network	Type	Input Size	Num	Filter	Pad	stride
FPC	Conv1	3x16x16	16	1x1	0	1
	Maxout	16x16x16	-	4x1	-	-
	Maxpooling	4x16x16	-	2x2	0	2
	Conv2	4x8x8	48	1x1	0	1
	Maxpool	48x8x8	-	8x8	0	8
	Conv3	48x1x1	1	1x1	0	1
	Params	288				
	Complexity (Conv layers)	$2.46\text{x}10^4$				
DehazeNet	Params	8240				
	Complexity (Conv layers)	$9.39\text{x}10^5$				

line and present a novel fully point-wise CNN (FPC) according to the proposed units in Sect. 3.3. In addition, we insert a pooing layer before the multi-scale feature layers with stride 2 to reduce the computational complexity. Since max pooling is more efficient in preserving useful features than average pooling in practice, we use max pooling as our default setting. The architectures and their numbers of parameters as well as computational complexity are shown in Table 4.

To verify the modeling capacity of the proposed FPC, we first train it using 200, 000 patches and test it on the left 100, 000 patches. Table 5 shows the MSE between ground truth transmission and predictions of different methods. As for the proposed FPC, it achieves the best performance due to its superior modeling capacity of local statistical regularities. Compared with the state-of-the-art (Dehazenet), our proposed FPC only uses 3.5% network parameters and 2.62% computational cost.

Then, we test the effectiveness of the proposed method on complete synthesized images. We synthesize hazy images by using stereo images from Middlebury Stereo Datasets [26] referring [11]. The PSNR and SSIM results of different methods are summarized in Table 6. As can be seen, the proposed FPC achieves better results than DCP and Dehazenet for both PSNR and SSIM.

Table 5. MSE of predicted transmission on test set of different methods.

Methods	DCP[23]	DehazeNet[11]	FPC
MSE(x10^{-2})	2.41	1.20	1.17

Table 6. PSNR and SSIM of different methods on complete synthetic hazy images.

Methods	DCP[23]	DehazeNet[11]	FPC
PSNR	20.16	20.29	21.17
SSIM	0.8611	0.8680	0.8733

4.2.2 Comparisons with state-of-the-art methods

To evaluate the performance of the proposed method on outdoor real hazy images, we compare it with state-of-the-art methods including DCP [23], dehazeNet [11] and AODNet [33]. Codes and Models of these methods are provided by the authors. Figure 6 shows the visual inspection results

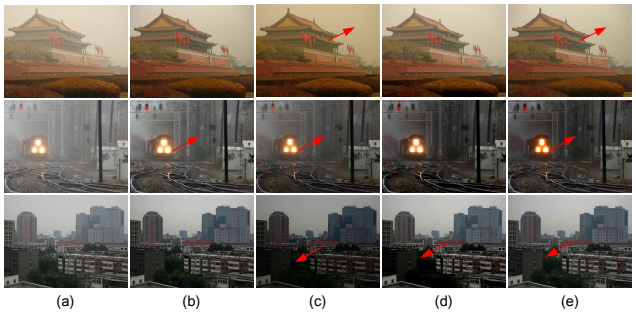

Figure 6. Dehazing results. (a) Input hazy images. (b) Results of DCP [23]. (c) Results of AODNet [33]. (d) Results of DehazeNet [11]. (e) Results of the proposed FPC.

on some challenging natural hazy images. As can be seen, our proposed FPC achieves the most competitive visual results among all. This demonstrates that our proposed FPC has superior capacity of modeling the statistics in the local patches, even for the challenging cases, such as the sky region, illumination variant regions and textureless regions.

To compare the running time of different methods, we test them on the hazy images with a size of $640*480$. All the networks are tested on Matlab platform with GPU acceleration. The proposed FPC processes a single image in 3ms, while DehazeNet takes 466ms and AODNet takes 4.3ms. The proposed FPC is the fastest due to its light-weight and fully point-wise convolutional structure.

4.3. Underwater image enhancement

Following the previous methods [15], we adopt the underwater imaging model which describes light attenuation and scattering effect relative to light path distance and wavelength simultaneously:

$$I_c = J_c \eta_c t + A_c \eta_c \left(1 - t\right), c \in \{R, G, B\}, \qquad (10)$$

where η_c is color cast, A_c is airlight, and J_c is the target clear image. Therefore, the goal of underwater image enhancement is to recover J_c after estimating η_c and t. It usually consists of two sequential steps: color correction and dehazing. Data preparation is according to Sect. 4.1 and Sect. 4.2. It should be noted that since the underwater image is usually green or blue, we generate color casts by randomly sample H value from $[120, 240]$ in HSV color space. More details and results can be found in supplement.

4.3.1 Ablation experiments

Since its resemblance with previous two problems, we propose a novel network architecture for underwater image enhancement. It consists of two cascaded sub-networks: color correction network and dehazing network, which is same as the proposed networks in Sect. 4.1 and Sect. 4.2.

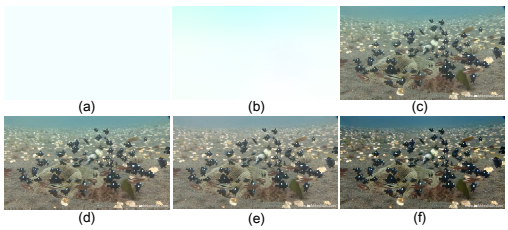

Figure 7. Color cast estimation results. (a) The estimated global color cast. (b) The estimated local color cast. (c) The original underwater image. (d) Color correction result by using (a). (e) Color correction result by using (b). (f) Final enhancement result by dehazing on (e).

In previous work [15, 34, 36], the color cast in the underwater image is estimated as global constant. However, in practical underwater environment, the color cast is spatial variant with the changes of distance. As shown in Fig. 7(a), the estimated global color cast is mainly affected by the near scene objects and tends to be whitish. It results in poor color correction results on the distant regions. To overcome this problem, we use pixel-ensembles randomly sampled from local patches as network input and estimate the color cast locally. As can be seen from Fig. 7(e) and (f), the proposed method can better remove color cast, especially in the distant regions, and leads to a visual pleasing enhancement result.

4.3.2 Comparisons with state-of-the-art methods

To evaluate the performance of the proposed method, we compare it with state-of-the-art methods [15, 34, 36] on several test images commonly used in literatures. Results are shown in Fig. 8 and Fig. 9. As can be seen, results of [15, 34, 36] improve image contrast but exhibit color distortion. As a contrast, our method achieves visual pleasing results since it estimate color cast and transmission accurately. It convinces that the proposed cascaded fully point-wise CNN can model multiple statistical regularities and handle the complicated cases like underwater image enhancement.

In addition, we show that the proposed method is computational efficient due to its fully CNN structure with less parameters. For a $640 * 480$ image, the proposed method processes it in less 25ms on Matlab platform with GPU acceleration. By integrating the illuminant estimating and haze removal phases together, it is promising to develop an end-to-end modeling method for underwater image enhancement task and achieves faster and better results.

5. Conclusion

According to our analysis on the degraded images, we find that the statistical properties are easier to be detected in the pixel-ensembles generated by randomly sampling from the original images. Following this observation, we propose a fully point-wise CNN for modeling the statistical regularities in natural images. The evaluation on three applications demonstrates that our proposed structures achieve the superior efficiency over the existing architectures while maintaining comparable accuracy.

The limitation of our proposed method is that it only captures the statistical distribution while misses the spatial structures in the pixel space. One feasible solution for this is to design a multi-branch network and take our proposed structures as one of them. Another solution is to connect our proposed structures to the network that extracts structural features, which directly models the statistical properties in the feature space. Due to the lightweight and compact properties, our proposed architecture is promising to work well in the two scenarios. We will leave these researches as our future work.

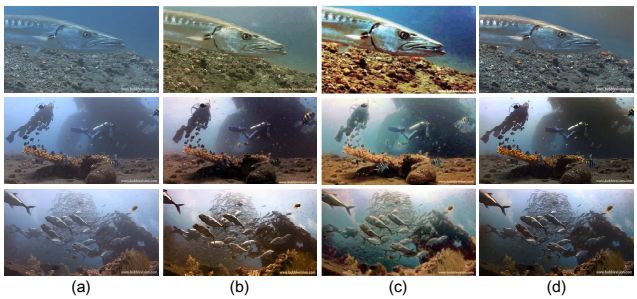

Figure 8. Underwater enhancement results. (a) Input underwater images. (b) Results of [15]. (c) Results of [34]. (d) Results of the proposed FPC.

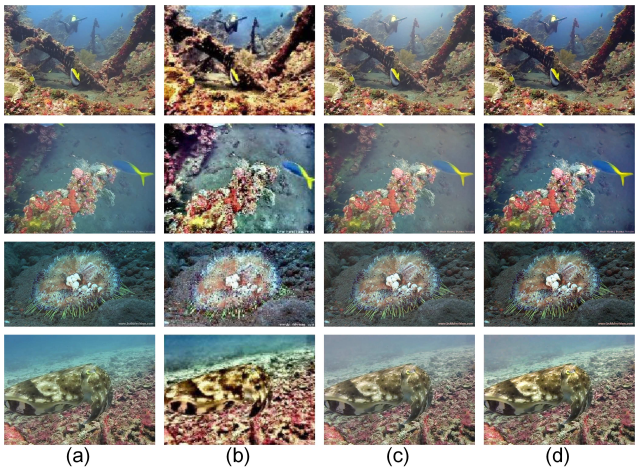

Figure 9. Underwater enhancement results. (a) Input underwater images. (b) Results of [36]. (c)-(d) Color correction results and final enhancement results of the proposed method.

References

[1] K. Barnard. Improvements to gamut mapping colour constancy algorithms. *Computer Vision-ECCV 2000*, pages 390–403, 2000. 6

[2] K. Barnard, V. Cardei, and B. Funt. A comparison of computational color constancy algorithms. i: Methodology and experiments with synthesized data. *IEEE transactions on Image Processing*, 11(9):972–984, 2002. 2, 6

[3] J. T. Barron. Convolutional color constancy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 379–387, 2015. 6

[4] J. T. Barron and Y.-T. Tsai. Fast fourier color constancy. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 5, 6

[5] D. Berman, S. Avidan, et al. Non-local image dehazing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1674–1682, 2016. 1

[6] S. Bianco, C. Cusano, and R. Schettini. Color constancy using cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 81–89, 2015. 2, 5, 6

[7] S. Bianco, C. Cusano, and R. Schettini. Single and multiple illuminant estimation using convolutional neural networks. *IEEE Transactions on Image Processing*, 2017. 6

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

[9] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980. 1, 6

[10] H. C. Burger, C. J. Schuler, and S. Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2392–2399. IEEE, 2012. 1

[11] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao. Dehazenet: An end-to-end system for single image haze removal. *IEEE Transactions on Image Processing*, 25(11):5187–5198, 2016. 1, 2, 4, 6, 7

[12] Y. Cao, S. Fang, and Z. Wang. Digital multi-focusing from a single photograph taken with an uncalibrated conventional camera. *IEEE Transactions on image processing*, 22(9):3703–3714, 2013. 5

[13] 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. 2, 5, 6

[14] 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. 6

[15] J. Y. Chiang and Y.-C. Chen. Underwater image enhancement by wavelength compensation and dehazing. *IEEE Transactions on Image Processing*, 21(4):1756–1769, 2012. 7, 8

[16] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016. 1

[17] R. Fattal, D. Lischinski, and M. Werman. Gradient domain high dynamic range compression. In *ACM Transactions on Graphics (TOG)*, volume 21, pages 249–256. ACM, 2002. 5

[18] G. D. Finlayson, S. D. Hordley, and P. M. Hubel. Color by correlation: A simple, unifying framework for color constancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1209–1221, 2001. 2

[19] 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. 1, 6

[20] 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. 5, 6

[21] 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. 6

[22] X. Guo, Y. Li, and H. Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on Image Processing*, 26(2):982–993, 2017. 5

[23] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2009. 1, 2, 7

[24] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2011. 1

[25] 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. 3

[26] H. Hirschmuller and D. Scharstein. Evaluation of cost functions for stereo matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 7

[27] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[28] Y. Hu, B. Wang, and S. Lin. Fc4: Fully convolutional color constancy with confidence-weighted pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4085–4094, 2017. 2, 5, 6

[29] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. 5

[30] H. R. V. Joze, M. S. Drew, G. D. Finlayson, and P. A. T. Rey. The role of bright pixels in illumination estimation. In *Color and Imaging Conference*, volume 2012, pages 41–46. Society for Imaging Science and Technology, 2012. 6

[31] Y. Karklin and M. S. Lewicki. A hierarchical bayesian model for learning nonlinear statistical regularities in nonstationary natural signals. *Neural computation*, 17(2):397–423, 2005. 1

[32] E. H. Land and J. J. McCann. Lightness and retinex theory. *Josa*, 61(1):1–11, 1971. 5

[33] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng. An all-in-one network for dehazing and beyond. *arXiv preprint arXiv:1707.06543*, 2017. 2, 7

[34] C.-Y. Li, J.-C. Guo, R.-M. Cong, Y.-W. Pang, and B. Wang. Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. *IEEE Transactions on Image Processing*, 25(12):5664–5677, 2016. 7, 8

[35] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters–improve semantic segmentation by global convolutional network. *arXiv preprint arXiv:1703.02719*, 2017. 3

[36] Y.-T. Peng and P. C. Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Transactions on Image Processing*, 26(4):1579–1594, 2017. 7, 8

[37] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang. Single image dehazing via multi-scale convolutional neural networks. In *European Conference on Computer Vision*, pages 154–169. Springer, 2016. 4

[38] L. Shi. Re-processed version of the gehler color constancy dataset of 568 images. *http://www. cs. sfu. ca/~ color/data/*, 2000. 5

[39] 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. 1, 2, 5, 6

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 3, 4

[41] K. Tang, J. Yang, and J. Wang. Investigating haze-relevant features in a learning framework for image dehazing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2995–3000, 2014. 2

[42] J. Van De Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *IEEE Transactions on image processing*, 16(9):2207–2214, 2007. 1, 2, 6

[43] J. Xie, L. Xu, and E. Chen. Image denoising and inpainting with deep neural networks. In *Advances in Neural Information Processing Systems*, pages 341–349, 2012. 1

[44] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 2017. 2

[45] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *arXiv preprint arXiv:1707.01083*, 2017. 3

[46] Q. Zhu, J. Mai, and L. Shao. Single image dehazing using color attenuation prior. In *BMVC*, 2014. 1