

# Extreme Channel Prior Embedded Network for Dynamic Scene Deblurring

Jianrui Cai, Wangmeng Zuo, and Lei Zhang, *Fellow, IEEE*

**Abstract**—Recent years have witnessed the significant progress on convolutional neural networks (CNNs) in dynamic scene deblurring. While most of the CNN models are generally learned by the reconstruction loss defined on training data, incorporating suitable image priors as well as regularization terms into the network architecture could boost the deblurring performance. In this work, we propose an Extreme Channel Prior embedded Network (ECPeNet) to plug the extreme channel priors (*i.e.*, priors on dark and bright channels) into a network architecture for effective dynamic scene deblurring. A novel trainable extreme channel prior embedded layer (ECPeL) is developed to aggregate both extreme channel and blurry image representations, and sparse regularization is introduced to regularize the ECPeNet model learning. Furthermore, we present an effective multi-scale network architecture, namely image full scale exploitation (IFSE), which works in both coarse-to-fine and fine-to-coarse manners for better exploiting information flow across scales. Experimental results on GoPro and Köhler datasets show that our proposed ECPeNet performs favorably against state-of-the-art deep image deblurring methods in terms of both quantitative metrics and visual quality.

**Index Terms**—Dynamic scene deblurring, Convolutional neural network, Extreme channel prior, Multi-scale strategy

## I. INTRODUCTION

REPRODUCING a high quality image faithful to the scene is an essential goal of digital photography. The real images, however, are often blurred during image acquisition due to the effect of many factors such as camera shake, object motion, and out-of-focus [1]. The resulting blurry images will not only degrade the perceptual quality of photos but also degenerate the performance of many image analytic and understanding models [2]. Blind image deblurring, which has been studied extensively in low level vision for decades of years [3], plays an essential role in improving the visual quality of real-world blurry images.

In general, the purpose of blind image deblurring is to recover the latent sharp image  $\mathbf{y}$  from its blurry observation:

$$\mathbf{x} = \mathbf{k} \otimes \mathbf{y} + \mathbf{n} \quad (1)$$

where  $\mathbf{k}$  is an unknown blur kernel (*i.e.*, uniform or non-uniform),  $\mathbf{n}$  is an additive white Gaussian noise and  $\otimes$  de-

This work is supported by Hong Kong RGC RIF grant (R5001-18).

J. Cai is with Department of Computing, The Hong Kong Polytechnic University, Kowloon Hong Kong (e-mail: csjcai@comp.polyu.edu.hk).

W. Zuo is with School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, and also with Peng Cheng Laboratory, Shenzhen, China (e-mail: wmuo@hit.edu.cn).

L. Zhang is with Department of Computing, The Hong Kong Polytechnic University, Kowloon Hong Kong, and also with DAMO Academy, Alibaba Group (e-mail: cslzhang@comp.polyu.edu.hk).

notes the convolution operator. This inverse problem, however, is severely ill-posed and requires extra information on latent image  $\mathbf{y}$  to constrain the solution space. Thus, there are mainly two categories of approaches for utilizing prior knowledge, *i.e.*, optimization-based and deep learning based deblurring methods. Optimization-based approaches explicitly model prior knowledge to regularize the solution space of blur kernel [4], [5], [6], [7], [8] and latent image [9], [10], [11], [12], [13] via an optimization framework. In contrast, deep learning based methods [1], [2], [14], [15] implicitly learn a direct mapping function (*e.g.*, convolutional neural network, CNN) from degraded image to latent clean image.

For blind image deblurring problem, optimization-based and deep learning methods respectively have their merits and limitations. Optimization-based methods are flexible in incorporating versatile priors or regularizations [4], [5], [8], [11], [12] tailored for blind deblurring, but suffer from the time-consuming optimization procedure and over-simplified assumptions on blur kernels (*e.g.*, spatially invariant and uniform). Moreover, conventional image priors (*e.g.*, total variation [4]) are limited in blind deblurring and prone to the ordinary solution of delta kernel. Stronger priors, *e.g.*,  $\ell_0$ -norm [16] and normalized sparsity [5], are then suggested for blur kernel estimation. On the other hand, deep learning methods [1], [2], [14], [15] benefiting from end-to-end training and joint optimization enjoy fast speed and flexibility in handling spatially variant blur in the dynamic scene. However, deep models may be limited in capturing specific priors for blind deblurring. As for dynamic scene deblurring, existing dataset [1] is of a relatively small scale, hindering the performance of learned model.

Taking the merits and drawbacks of optimization-based and deep learning based methods into account, one interesting question is whether we can exploit prior model to constrain both the network architecture and learning losses for improved dynamic scene deblurring performance. We make a good attempt to address this challenging problem. In particular, based on the effectiveness of image prior in blind deblurring, we propose an Extreme Channel Prior embedded Network (ECPeNet) to help the restoration of latent clean image. The critical component of ECPeNet is a novel trainable extreme channel prior embedded layer (ECPeL), which can aggregate extreme channel and blurry image representations to leverage their respective advantages. By enforcing sparsity on both dark and bright channels of feature maps, we can regularize the solution space of CNN during training, thereby incorporating extreme channel priors into ECPeNet.

Moreover, existing deep dynamic scene deblurring models

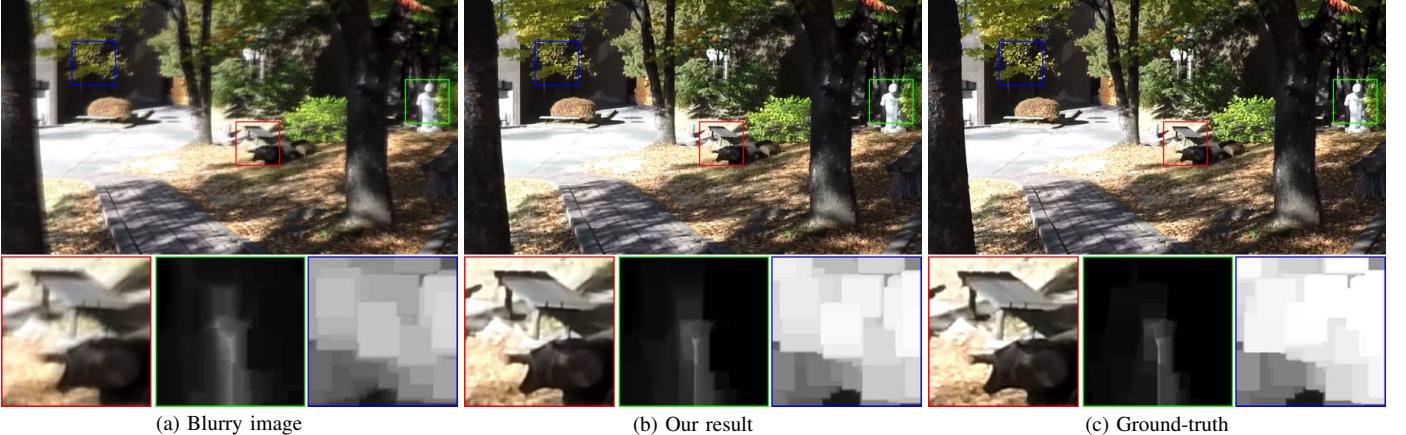


Fig. 1: Deblurring result on a GoPro image [1]. (a) The blurry image; (b) our result; and (c) the Ground-truth image. **Red box:** zoom-in view of the original local patch. **Green box:** zoom-in view of the dark channel of its corresponding local patch. **Blue box:** zoom-in view of the bright channel of its corresponding local patch.

[1], [14] usually adopt the multi-scale network architecture to exploit coarse and middle level information for finer scale image deblurring. However, these deep multi-scale network architectures only consider the coarse-to-fine information flow. That is, blind deblurring is first performed on the small scale, and then deblurring results (or latent representations) are combined with feature representations on a larger scale for further refinement. Unfortunately, such a multi-scale network architecture could not fully exploit the information flow across scales. In this work, we show that feature representations of larger scales actually can also benefit the dynamic scene deblurring on a smaller scale. To this end, we present a more effective multi-scale network architecture that works in both coarse-to-fine and fine-to-coarse manners. By training the deblurring network in this new multi-scale network structure, which we call image full scale exploitation (IFSE), we can better exploit the information flow across scales.

Experimental results on GoPro [1] and Köhler [17] datasets demonstrate that our ECPeNet outperforms state-of-the-art deep dynamic scene deblurring methods. As shown in Figure 1, one can see that the dark and bright channels of our deblurring result are sparser than the blurry image. The contribution of this paper is two-fold:

- We propose a novel trainable structural layer, namely extreme channel prior embedded layer (**ECPeL**), which can aggregate data information and prior knowledge (*i.e.*, priors on dark and bright channels) to leverage their merits but avoid limitations. ECPeL provides an effective way to plug prior knowledge (*i.e.*, statistical properties) into a deep deblurring network in an end-to-end manner.
- We introduce a new multi-scale network structure, namely image full scale exploitation (**IFSE**), which works in both coarse-to-fine and fine-to-coarse manners to fully exploit different resolution images for maximizing the information flow.

The remainder of this paper is organized as follows. Section II briefly reviews the relevant works of optimization based deblurring methods and deep learning based deblurring methods.

Section III presents the proposed ECPeNet for aggregating both data information and extreme channel prior knowledge. Section IV presents the experimental results, and Section V provides some concluding remarks.

## II. RELATED WORK

In this section, we briefly review the recent optimization-based and deep learning based image deblurring methods.

### A. Optimization-based Deblurring Methods

The optimization-based methods aim to develop effective image priors to favor clean images over the blurry one. Representative priors include sparse gradients [9], [18], [19], [20], hyper-Laplacian prior [21], normalized sparsity prior [5],  $\ell_0$ -norm prior [16], patch recurrence prior [22] and discriminatively learned prior [8], [13]. By taking advantages of the aforementioned priors, existing optimization-based methods could deliver competitive results on generic natural images. These approaches, however, cannot be generalized well to handle domain specific images. Thus, specific priors are introduced for specific images, *e.g.*, light streak prior [23] for low light images, and a combination of intensity and gradient prior [24] for text images. Recently, Pan *et al.* [11] developed the dark channel prior (DCP) [25] to enforce sparsity on the dark channel of latent image and achieved promising result on both generic and specific images. With the success of [11], Yan *et al.* [12] further introduced a bright channel prior (BCP) to solve the corner case image, which contains a large amount of bright pixels. By plugging the extreme channel prior (a combination of BCP and DCP) into the deblurring model, Yan *et al.* achieved state-of-the-art results on various scenarios.

Although the optimization based algorithms have demonstrated their effectiveness in image deblurring, the simplified assumptions on the blur model and time-consuming parameter-tuning process are two lethal problems to hinder their performance in real-world cases. In this work, we utilize a realistic GoPro dataset [1] to end-to-end train a new multi-scale network for latent sharp image restoration.

### B. Deep Learning based Deblurring Methods

Deep learning based methods focus on exploiting external training data to learn a mapping function in accordance with the degradation process [26], [27]. The powerful end-to-end training paradigm and the non-linear modeling capability make CNNs a promising approach to image deblurring. Early CNN-based deblurring methods aim to mimic conventional deblurring frameworks for the estimation of both latent image and blur kernel. Prior works [28], [29] first use a network to predict the non-uniform blur kernel and then utilize a non-blind deblurring method [30] to restore images. In [31], Schuler *et al.* introduced a two-stages network to simulate iterative optimization. In [32], Chakrabarti *et al.* utilized a network to predict frequency coefficients of blur kernels. However, these methods may fail when the estimated kernel is inaccurate [33]. Therefore, more recent approaches prefer to train kernel estimation-free networks to restore latent images directly. Specifically, Nah *et al.* [1] proposed a multi-scale CNN to progressively recover the latent image. Tao *et al.* [14] introduced a scale-recurrent network equipped with a ConvLSTM layer [34] to further ensure information flow between different resolution images. Kupyn *et al.* [2] adopted the Wasserstein GAN [35], [36] as an objective function to restore the texture details of latent image. Zhang *et al.* [15] employed spatially variant recurrent neural networks (RNNs) to reduce the computational cost.

While deep learning based deblurring methods have reported impressive results, the limited number of training data and the disappreciation of prior knowledge are two main factors hampering the performance improvement. To mitigate these issues, in this paper, we introduce the extreme channel prior into CNN to regularize the solution space of clear images.

## III. EXTREME CHANNEL PRIOR EMBEDDED NETWORK

In this section, we introduce an extreme channel prior embedded network (ECPeNet) for dynamic scene deblurring. To begin with, we first present the motivation of our proposed ECPeNet. Then, we describe in detail the network architecture, followed by the definitions of our proposed image full scale exploitation (**IFSE**), extreme channel prior embedded layer (**ECPeL**) and the objective function. Finally, we discuss the differences between deep learning based dynamic scene deblurring networks and traditional extreme channel prior optimization based deblurring methods.

### A. Motivation

Given a single blurry image  $\mathbf{x}_i$ , existing CNN-based methods aim at learning a mapping function  $F_\Theta$  to generate an estimation of latent sharp image  $\hat{\mathbf{y}}_i$ , which is required to approximate the ground-truth  $\mathbf{y}_i$ . This procedure can be formulated as:

$$\begin{aligned} \hat{\Theta} = \operatorname{argmin}_{\Theta} & \sum_i \ell(\hat{\mathbf{y}}_i, \mathbf{y}_i) \\ \text{s.t. } & \hat{\mathbf{y}}_i = F_\Theta(\mathbf{x}_i) \end{aligned} \quad (2)$$

where  $(\mathbf{x}_i, \mathbf{y}_i)$  refer to the  $i$ -th image pair in the training dataset and  $\Theta$  is the parameter of mapping function.

However, such a formulation is limited in capturing image priors specified to blind deblurring, which is generally very different from those for non-blind restoration. Moreover, the existing training image pairs are insufficient to learn an effective mapping function  $F_\Theta$ . Therefore, incorporating suitable image priors as well as regularization terms into the network architecture is essential to further improve the deblurring performance. According to the observations by [11], [12], the dark channel of an image would be less dark after the blurring process, while the bright channel of an image would be no longer bright. The reason is that a dark/bright pixel will be averaged with its neighboring high/low intensity pixels during the blurring process, as shown in Figure 2. Based on the above observations, we find that enforcing sparsity on the extreme channel representations favors clear images over blurred ones. To ensure that the extreme channel prior can be end-to-end trained with the network, the gradient of each component should be calculated for back-propagation. To this end, we propose a novel trainable ECPeL that aggregates both blurry image and extreme channel representations to enhance deblurring performance. Besides, we utilize  $\ell_1$ -norm as the sparse regularization term since it is differentiable. Thus, we can rewrite Eqn. (2) as:

$$\begin{aligned} \hat{\Theta} = \operatorname{argmin}_{\Theta} & \sum_i \ell(\hat{\mathbf{y}}_i, \mathbf{y}_i) \\ \text{s.t. } & \hat{\mathbf{y}}_i = F_\Theta(\mathbf{x}_i | \Lambda, \Omega) \end{aligned} \quad (3)$$

where  $\Lambda$  and  $\Omega$  are the extreme channel representations under the constraint of dark and bright channels priors. By this way, we embed the image priors into the mapping function  $F_\Theta$  to generate higher quality latent sharp image.

### B. Architecture

Most of the traditional deblurring methods and the CNN-based dynamic scene deblurring networks [1], [14] prefer to utilize the multi-scale structure where the coarse-to-fine strategy is used for handling blur kernel. However, we found that a larger scale feature representations can also benefit the dynamic scene deblurring at a smaller scale. A multi-scale network architecture that works in both coarse-to-fine and fine-to-coarse manners could help the network fully exploit the information flow across scales.

The overall architecture of our proposed ECPeNet is illustrated in Figure 3. It contains three sub-networks respectively for three scales, and each of them consists of three major components: (i) input and output; (ii) encoder and decoder; (iii) feature mapper. Note that instead of utilizing the Rectifier Linear Units (ReLU) [37], we take the parametric ReLU (PReLU) [38] as activation function since it can improve the modeling capability with negligible extra computational cost. Unless denoted otherwise, all the convolution filters are set to  $3 \times 3$ , instead of  $5 \times 5$  as utilized by most of the other dynamic scene deblurring networks (e.g., [1] and [14]). Although a filter of size  $5 \times 5$  has more parameters than two filters of size  $3 \times 3$ , the one utilizing  $3 \times 3$  filters is more efficient and additional nonlinearity can be inserted between them [39]. Besides, the stride size for all convolution layers is set to 1 and the number of feature maps in each layer is set to 64, except for the last



Fig. 2: The extreme channels of the blurry and ground-truth images.

layer and the proposed ECPeL, where the number of feature maps is set to 3 and  $\{3, 64, 3\}$ , respectively. The details of each component are described as follows.

**Input and Output.** An effective multi-scale network architecture is utilized in this work to restore the latent sharp image from a coarser scale to finer scales. Consider an image pair  $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{H \times W \times C}$ , where  $H \times W$  is the pixel resolution and  $C$  is the number of channels, which is set to 3. We first use bicubic interpolation to progressively downsample the image pair with the ratio of  $\frac{1}{2}$ , and generate 3 scales of image pairs with the resolution of  $\{H \times W, \frac{H}{2} \times \frac{W}{2}, \frac{H}{4} \times \frac{W}{4}\}$ . We then take those blurred images at each scale as inputs to produce their corresponding sharp images. The sharp one at the original resolution is considered as the final output.

**Encoder and Decoder.** Each scale of the encoder consists of 4 convolution layers, the proposed ECPeL and a shuffle operation with factor  $\frac{1}{2}$ . As for the decoders, they basically mirror the architecture of the encoders, except that the factor of shuffle operation [40] is set to 2. The encoder and decoder networks are mainly designed for three purposes. (i) They progressively transform images on different scales to extract shallow features (in encoders) and transform them back to the resolution of inputs (in decoders). (ii) They work in both coarse-to-fine and fine-to-coarse manners for better exploiting information flow across scales. More details of the proposed multi-scale structure are described in Subsection III-C. (iii) They integrate the extreme channel prior into the network via ECPeL for regularizing the model learning. More details of the proposed ECPeL are presented in Subsection III-D.

**Feature Mapper.** The feature mapper module, which aims to refine the shallow features progressively, is an essential part of latent image restoration. One critical factor for reducing blur artifacts is the size of receptive field. To enlarge the receptive field, we (i) stack a set of convolutional layers to achieve a larger depth network and (ii) utilize the shuffle operations for

downsampling and upsampling features. Considering that a deeper neural network is more difficult to converge, we adopt the residual blocks to speed-up the training procedure [41], [26]. Besides, the feature mapper module utilizes the long skip connection and short skip connection to make full use of hierarchical features in all convolutional layers. A similar skip connection strategy has been utilized in a very recent work [42]. As illustrated in Figure 3, the feature mapper contains 16 residual-in-residual blocks (RIRBlock), and each of them has 4 residual blocks (ResBlock). The ResBlock consists of 2 convolution layers and a PReLU activation function. Since we utilize the filter of size  $3 \times 3$ , the total number of parameters of our ECPeNet is almost the same as previous methods. Note that all the weights across different scales of feature mapper sub-modules are shared.

### C. Image Full Scale Exploitation

Considering that large blur kernels are difficult to be directly estimated, previous blind deblurring methods [43], [1], [14] utilize the multi-scale strategy to progressively predict the blur kernel and the latent image. Generally, methods are first performed at the smallest scale of the blurry image to estimate its corresponding coarsest scale kernel and latent sharp image, and then the coarsest deblurring result is combined with the larger scale blurry input for further refinement. However, such a coarse-to-fine strategy could not fully exploit the information flow across scales because we find that: (i) the information of finer scale blurry image representation is beneficial for the estimation of coarser scale latent sharp image; and (ii) the combination in shallow feature domain can yield a better result than image (RGB) domain. Unlike previous methods [1], [43], [14] that directly concatenate the upsampled coarser scale latent sharp image with the finer scale blurry image, we propose a new multi-scale structure, namely image full scale exploitation (IFSE), which works in both coarse-to-

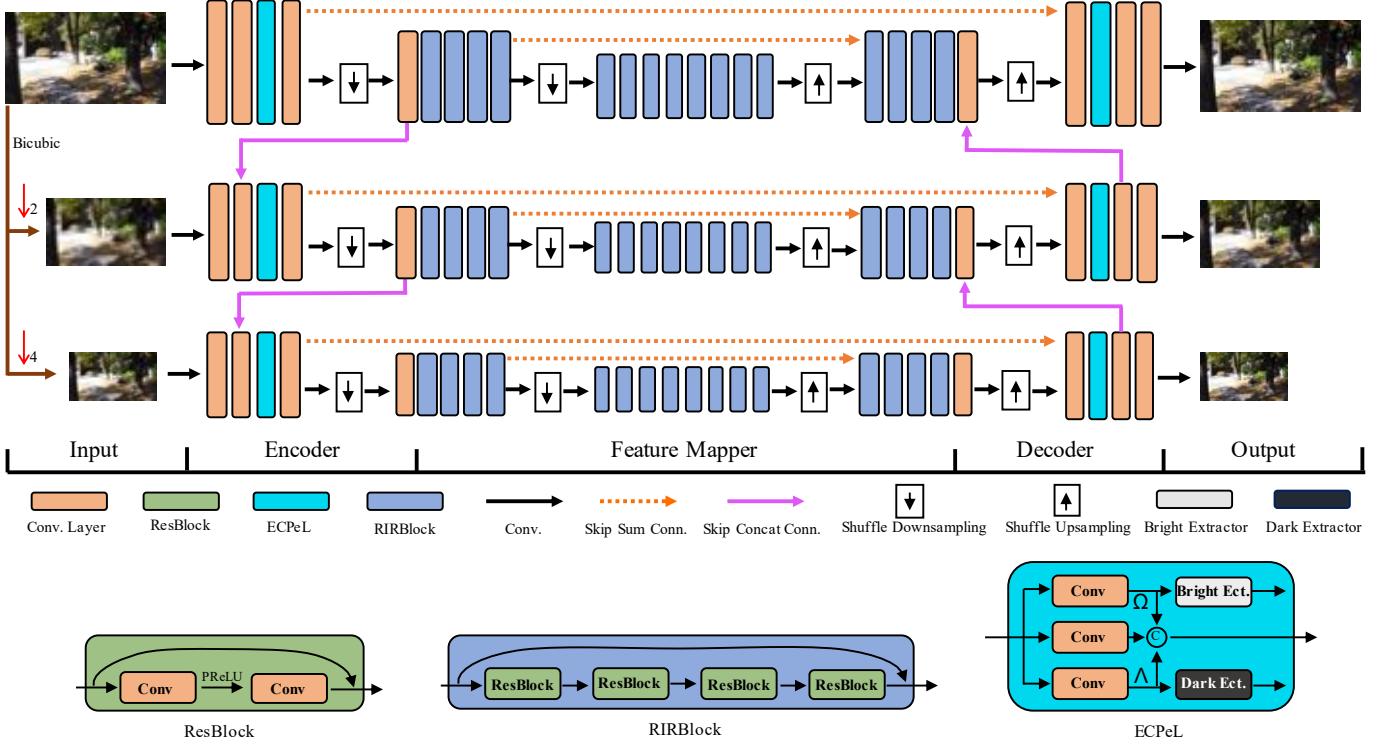


Fig. 3: Illustration of our proposed ECPeNet architecture.

fine and fine-to-coarse manners to ensure the information flow across different scale images and expand the receptive field. Specifically, in the fine-to-coarse phase, the encoder first downsamples the finest scale latent image representations by shuffling the features with factor  $\frac{1}{2}$  to ensure the same resolution on different scales of features (namely, a features of size  $m \times n \times c$  is shuffled to  $\frac{m}{2} \times \frac{n}{2} \times 4c$ ). Then the encoder further concatenates the downsampled features with coarser scale features for restoring the coarser scale sharp image. On the contrary, in the coarse-to-fine phase, the decoder upsamples the restored coarser scale features by shuffling the features of size  $\frac{m}{2} \times \frac{n}{2} \times 4c$  back to  $m \times n \times c$  and concatenates them with finer scale features to estimate the finer scale sharp image. Consequently, by training the network in both coarse-to-fine and fine-to-coarse manners, our proposed ECPeNet can fully exploit different scale images to maximize the information flow between them, resulting in better performance. In Section IV-C, we will conduct an ablation study to verify its effectiveness.

#### D. Extreme Channel Prior Embedded Layer

Since the extreme channel of a sharp image is sparser than a blurry image, extreme channel prior and sparse constraints favor clear images over blurred images. Therefore, we propose an ECPeL to aggregate blurry image representation and extreme channel representation together to regularize the solution space of CNN. Specifically, it first learns 3 mapping functions  $\mathcal{M}_\theta$ ,  $\mathcal{M}_{[\alpha|D]}$  and  $\mathcal{M}_{[\beta|B]}$  to transform the feature map  $f^{l-1}$  from previous layer into 3 new feature maps, including a deeper layer transformed feature  $f^l$ , a dark channel prior

constrained feature  $\Lambda$ , and a bright channel prior constrained feature  $\Omega$ . It then adopts a concatenation operation to concatenate those 3 feature maps for the integration of blurry image representation and extreme channel representation. Formally, the proposed ECPeL can be expressed as:

$$\begin{aligned} [\Lambda, f^l, \Omega] &= ECPeL(f^{l-1}) \\ f^l &= \mathcal{M}_\theta(f^{l-1}) \\ \Lambda &= \mathcal{M}_{[\alpha|D]}(f^{l-1}) \\ \Omega &= \mathcal{M}_{[\beta|B]}(f^{l-1}) \end{aligned} \quad (4)$$

where  $[\Lambda, f^l, \Omega]$  denotes the concatenation of these feature maps, and the subscripts  $[\alpha|D]$  and  $[\beta|B]$  denote that the parameters  $\alpha$  and  $\beta$  are optimized under the dark and bright channel prior constraint. To add extreme channel prior constraint into a network, the ECPeL utilizes (i) extractors to extract both dark and bright channel of features, and (ii) the  $\ell_1$ -regularization term to enforce sparsity in training.

The extractor  $D(\cdot)$  is designed to extract the dark channel of  $\Lambda$  via computing its minimum values in a local patch. Its **forward** function can be written as follows:

$$\begin{aligned} D(\Lambda)_{[h,w]} &= \Lambda_{[\mathcal{I}_D[h,w]]} \\ \mathcal{I}_D[h,w] &= \operatorname{argmin}_{i^* \in \Psi_{[h,w,c]}} \Lambda_{[i^*]} \end{aligned} \quad (5)$$

The extractor  $B(\cdot)$  aims to extract the bright channel of  $\Omega$  by calculating its maximum values in a local patch. Its **forward** function can be formulated as:

$$\begin{aligned} B(\Omega)_{[h,w]} &= \Omega_{[\mathcal{I}_B[h,w]]} \\ \mathcal{I}_B[h,w] &= \operatorname{argmax}_{i^* \in \Psi_{[h,w,c]}} \Omega_{[i^*]} \end{aligned} \quad (6)$$

where  $\Psi_{[h,w,c]}$  is the index set of inputs in a sub-window centered at pixel location  $[h, w, c]$ ,  $\mathcal{I}_{\mathcal{D}_{[h,w]}}$  and  $\mathcal{I}_{\mathcal{B}_{[h,w]}}$  are the masks that record the indices of the minimum and maximum values in a local patch, respectively. The patch sizes for each scale are set to  $\{31 \times 31, 19 \times 19, 11 \times 11\}$ . A single element  $\Lambda_{[h,w,c]}$  or  $\Omega_{[h,w,c]}$  of the input may be assigned to different outputs of  $D(\Lambda)_{[h,w]}$  or  $B(\Omega)_{[h,w]}$ .

The **backward** function of extractors computes partial derivative of the loss function with respect to input variables  $\Lambda_i$  and  $\Omega_i$  as follows:

$$\begin{aligned}\frac{\partial L}{\partial \Lambda_i} &= \sum_h \sum_w \sum_c 1\{i = \mathcal{I}_{\mathcal{D}_{[h,w]}}\} \frac{\partial L}{\partial D(\Lambda)_{[h,w]}} \\ \frac{\partial L}{\partial \Omega_i} &= \sum_h \sum_w \sum_c 1\{i = \mathcal{I}_{\mathcal{B}_{[h,w]}}\} \frac{\partial L}{\partial B(\Omega)_{[h,w]}}\end{aligned}\quad (7)$$

where  $i$  refers to the pixel location  $[h, w, c]$ . In other words, the partial derivatives  $\frac{\partial L}{\partial D(\Lambda)_{[h,w]}}$  and  $\frac{\partial L}{\partial B(\Omega)_{[h,w]}}$  are accumulated if  $i$  is the argmin and argmax selected for  $D(\Lambda)_{[h,w]}$  and  $B(\Omega)_{[h,w]}$ , respectively. In back-propagation, the partial derivatives  $\frac{\partial L}{\partial D(\Lambda)_{[h,w]}}$  and  $\frac{\partial L}{\partial B(\Omega)_{[h,w]}}$  are already calculated by the **backward** function of the loss layer.

With the proposed ECPeL, we can extract the dark and bright channels of shallow features (*i.e.*,  $D(\Lambda)$  and  $B(\Omega)$ ), which can be further enforced to be sparse via the objective function. By integrating the constrained features  $\Lambda$  and  $\Omega$  into the network, the proposed ECPeNet can achieve a better performance while using the same training dataset. The ablation study in Section IV-C is conducted for the evaluation.

### E. Loss Function

We utilize the  $\ell_1$ -norm of the reconstruction error as loss function for each scale. More specifically, we can rewrite Eqn. (3) as:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^3 \|\mathbf{y}_i^j - F_\Theta(\mathbf{x}_i^j | \Lambda^j, \Omega^j)\|_1 \quad (8)$$

where  $N$  is the total number of training pairs  $(\mathbf{x}, \mathbf{y})$  and  $j$  is the number of scales, which is set to 3 in this paper. Symbol  $(\cdot)^j$  refers to the image and feature on the  $j$ -th scale.

As described above, the sparsity regularization term is more beneficial for restoring a sharp image than a blurred one. To this end, we introduce an  $\ell_1$ -regularization term to enforce sparsity on both dark and bright channels of shallow features. The objective function can be given by:

$$\begin{aligned}\mathcal{L} &= \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^3 \|\mathbf{y}_i^j - F_\Theta(\mathbf{x}_i^j | \Lambda^j, \Omega^j)\|_1 \\ &\quad + \lambda \|D(\Lambda^j)\|_1 + \omega \|1 - B(\Omega^j)\|_1\end{aligned}\quad (9)$$

where  $\lambda$  and  $\omega$  are the trade-off parameters.  $D(\cdot)$  and  $B(\cdot)$  are the extractors to extract the dark channel and bright channel of features, respectively. With the **forwards** and **backwards** functions, the dark and bright channel extractors can be jointly optimized with the network in an end-to-end manner.

### F. Discussions

In this subsection, we discuss the differences between our ECPeNet and the previous methods, including traditional ECP based deblurring methods and deep deblurring networks.

**Difference to traditional ECP based deblurring methods.** In [11], [12], it has been proven that extreme channel prior and sparse constraint favors sharp images over blurry images. Inspired by [11], [12], our proposed ECPeNet utilizes the extreme channel prior and sparse constraints to regularize the network model learning. Although our ECPeNet and [11], [12] have something in common, there are three major differences. First, [11] and [12] are optimization-based deblurring methods, while our ECPeNet is a deep learning-based deblurring method. Second, [11], [12] focus on uniform blind deblurring, while our network aims at dynamic scene deblurring. Third, the ECP in [11] and [12] is used in the image domain, while our ECPeNet imposes ECP and sparse constraint in the feature domain, which is a non-trivial task.

**Difference to previous CNN-based deblurring methods.** Previous dynamic scene deblurring models [1], [14] are generally learned by the reconstruction loss defined on training data. Moreover, their multi-scale architectures only work in a coarse-to-fine manner. However, we found that: (i) incorporating suitable image priors (*i.e.*, ECP) and regularization term (*i.e.*,  $\ell_1$ -norm) into the network architecture could further improve the deblurring performance; and (ii) networks trained in either coarse-to-fine or fine-to-coarse manner could not fully exploit the information of different resolution images. Based on the above observations, we propose the ECPeNet for dynamic scene deblurring. The main differences between our ECPeNet and previous networks lie in the following aspects: (i) we impose the ECP and sparse constraint into CNN, which can regularize the solution space of CNN; and (ii) a new multi-scale structure, which works in both coarse-to-fine and fine-to-coarse manners, is introduced to better exploit image information across scales.

## IV. EXPERIMENTAL RESULTS

In this section, we provide experimental results to show the advantage of our proposed ECPeNet. We first present the experimental settings, including training and testing datasets, as well as parameter settings. We then compare ECPeNet with state-of-the-art dynamic scene deblurring methods. Finally, we conduct the ablation studies to verify the effectiveness of our proposed ECPeL and the IFSE strategy.

### A. Experiment Settings

We implement our framework by using the Caffe toolbox [44], and train the model on a PC equipped with an Intel Core i7-7820X CPU, 128G RAM and a single Nvidia Quadro GV100 GPU. Code, trained models and the deblurring results on the GoPro and Köhler datasets will be made publicly available<sup>1</sup>.

**Datasets.** We train our proposed ECPeNet on the GoPro training dataset [1], which contains 22 sequences with 2,103

<sup>1</sup><https://github.com/csjaicai/ECPeNet>

blurred/clear image pairs. Once the model is trained, we test it on the standard GoPro testing dataset [1] and Köhler [17] dataset. It is worth pointing out that, following previous methods [1], [14], we use the linear subset of the GoPro dataset to train the network and test the model. The GoPro testing dataset consists of 11 sequences with 1,111 image pairs, and the Köhler dataset has 4 latent images and 12 blur kernels. Note that, to simulate the realistic blurring process, the GoPro dataset generates blurred images through averaging adjacent short-exposure frames captured by a high-speed video camera, and the Köhler dataset replays the recorded 6D real camera motion trajectory to synthesize blurred images.

**Parameter Settings.** We crop the GoPro training dataset (linear subset) into  $256 \times 256 \times 3$  patches and make use of these patches to train the ECPeNet. The mini-batch size in all the experiments is set to 10, and the trade-off parameters  $\lambda$  and  $\omega$  are set to 0.1 and 0.2, respectively. For the model training, we utilize Xavier [45] to initialize all the trainable variables. The Adam solver [46] with the default parameters ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and  $\epsilon = 10^{-8}$ ) was adopted to optimize the network parameters. We fix the learning rate as  $10^{-4}$  and train the network with  $600K$  iterations, which takes about 140 hours. Additionally, we randomly rotate and/or flip the image patches for data augmentation. The 1% additive Gaussian noise is also randomly added to the blurred images for robust learning.

### B. Comparisons with State-of-the-art Methods

In this subsection, both quantitative and qualitative evaluations are conducted to verify the proposed ECPeNet on the benchmark datasets.

**Quantitative Evaluations.** We followed the same experimental setting as the prior arts [1], [15], [14], and compared ECPeNet with previous state-of-the-art deblurring methods [47], [28], [1], [2], [15], [14] in a quantitative way. A traditional dynamic scene deblurring method [47] is also used as one of the competitors. The source codes and trained models of the aforementioned methods are publicly available on the authors' websites, except for [47] and [15] whose results have been reported in previous works [1] and [15], respectively. Additionally, we utilize the same training dataset to retrain the network provided by [15] for its evaluation on the Köhler dataset. The average PSNR, SSIM, and MSSIM indices for different deblurring methods on GoPro testing and Köhler datasets are shown in Table I. One can see that on the GoPro testing dataset, the proposed ECPeNet significantly outperforms both the conventional non-uniform deblurring method [47] and these recently developed CNN based methods [28], [1], [2], [15], [14]. Even compared to the previous state-of-the-art method [14], the proposed ECPeNet still has **0.84** dB improvement. While on the Köhler dataset, all these deep learning based dynamic scene networks (including our proposed ECPeNet) trained on the GoPro dataset cannot achieve satisfactory results. This is mainly because the GoPro training dataset itself is not comprehensive enough to cover the different types of kernels. Moreover, the kernel distribution and the kernel size between GoPro and Köhler datasets are very

TABLE I: Average PSNR (dB), SSIM, MSSIM indices and runtimes for different methods on the benchmark datasets (running time is measured for an image with size  $1280 \times 720 \times 3$ ).

Method	GoPro		Köhler		Time
	PSNR	SSIM	PSNR	MSSIM	
Kim [47]	23.64	0.824	24.68	0.794	1 hr
Sun [28]	24.64	0.843	25.22	0.774	20 min
Nah [1]	29.08	0.914	26.48	0.808	2.87s
Tao [14]	30.26	0.934	26.75	0.837	0.62s
Kupyn [2]	28.70	0.858	26.10	0.816	0.59s
Zhang [15]	29.19	0.931	25.71	0.800	0.76s
Proposed	31.10	0.945	26.79	0.839	0.65s

different. Thus, these dynamic scene deblurring networks have comparative performance on the Köhler dataset. However, one can still notice that our method has certain advantage over competing methods.

Meanwhile, the running time by different methods for processing an image of resolution  $1280 \times 720 \times 3$  is also listed in Table I. One can notice that it takes a lot of time for a conventional method to restore an image because of the time-consuming iterative inference and the CPU implementation. While for these end-to-end training networks, they can achieve much faster speed to process an image on GPU. Considering that these dynamic scene deblurring networks are implemented by different deep learning platforms, the minor difference between them can be neglected within the margin of error.

**Qualitative Evaluations.** We compare the visual quality of restored images by our proposed ECPeNet and these recently developed CNN based dynamic scene deblurring networks, including Nah [1], Tao [14], Kupyn [2], and Zhang [15]. Figure 4 shows several blurred images from the GoPro [1] testing dataset and their corresponding deblurring results produced by the above methods. One can see that although these recently developed CNNs could remove the overall motion blur artifacts, the results restored by them are not perceptually pleasing enough because of the blurred edges and noticeable artifacts. For example, in the fourth column, all these previous CNN based deblurring networks could not recover the text information (see the red box zoom-in region). While in the second column, noticeable artifacts appear around the license plate number. By contrast, benefiting from the extreme channel prior constraint, our method can deliver more visually pleasing results with much fewer artifacts and sharper edges.

To further demonstrate the robustness of our method, the visual comparison results on images from the Köhler [17] dataset are also provided in Figure 5. Again, it can be seen that artifacts and blurred edges in the zoom-in areas (see characters 'B', '70', and '15') are noticeable for these previous CNN based methods. Although results recovered by Kupyn [2] and Tao [15] are sharper than other methods, distortion still exists. Compared with these methods, the ECPeNet can restore image sharpness and naturalness more effectively.

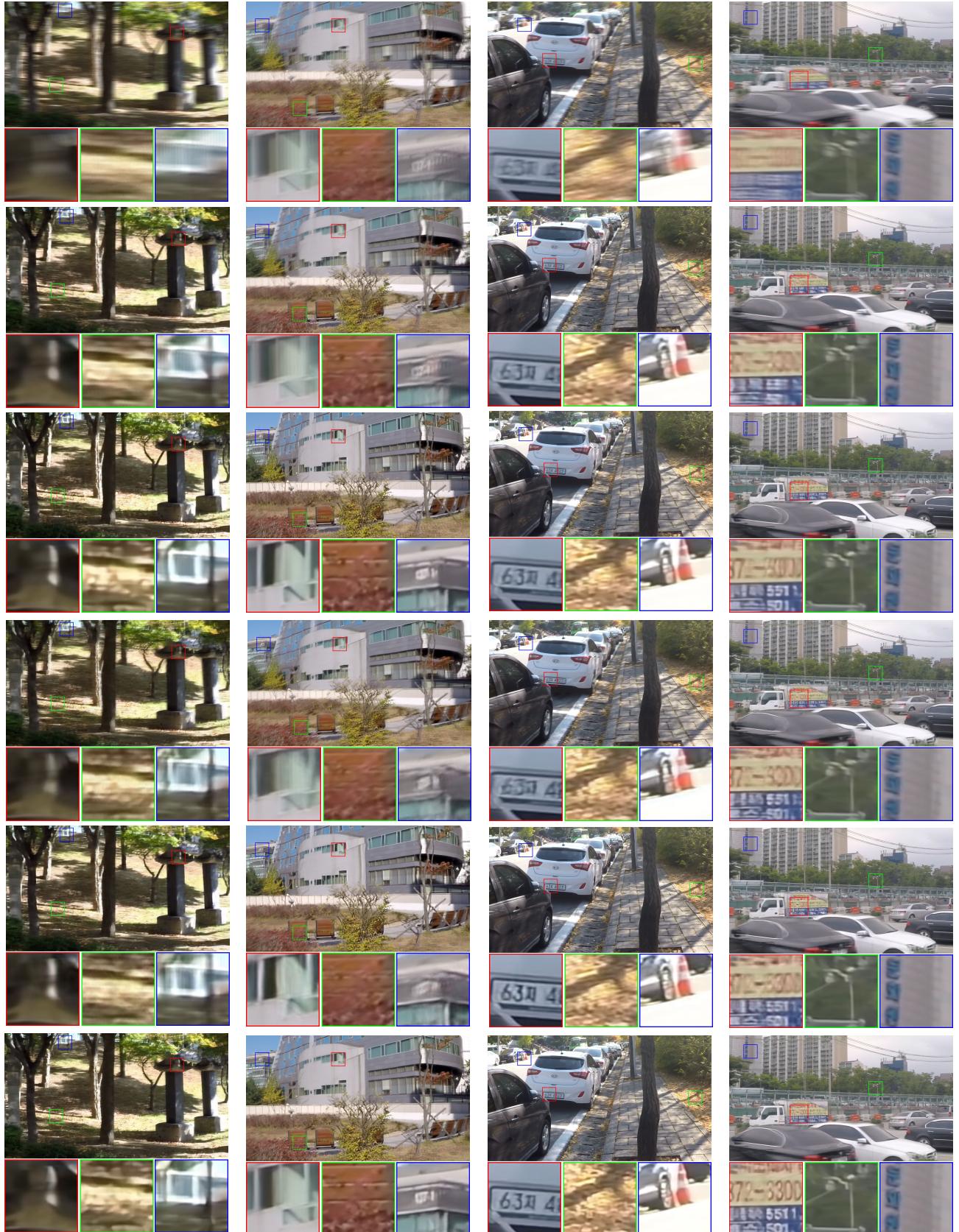


Fig. 4: Deblurring results on the GoPro dataset [1] by different methods. From top row to bottom row, we show inputs, results of Nah *et al.* [1], Tao *et al.* [14], Kupyn *et al.* [2], Zhang *et al.* [15], and our results.

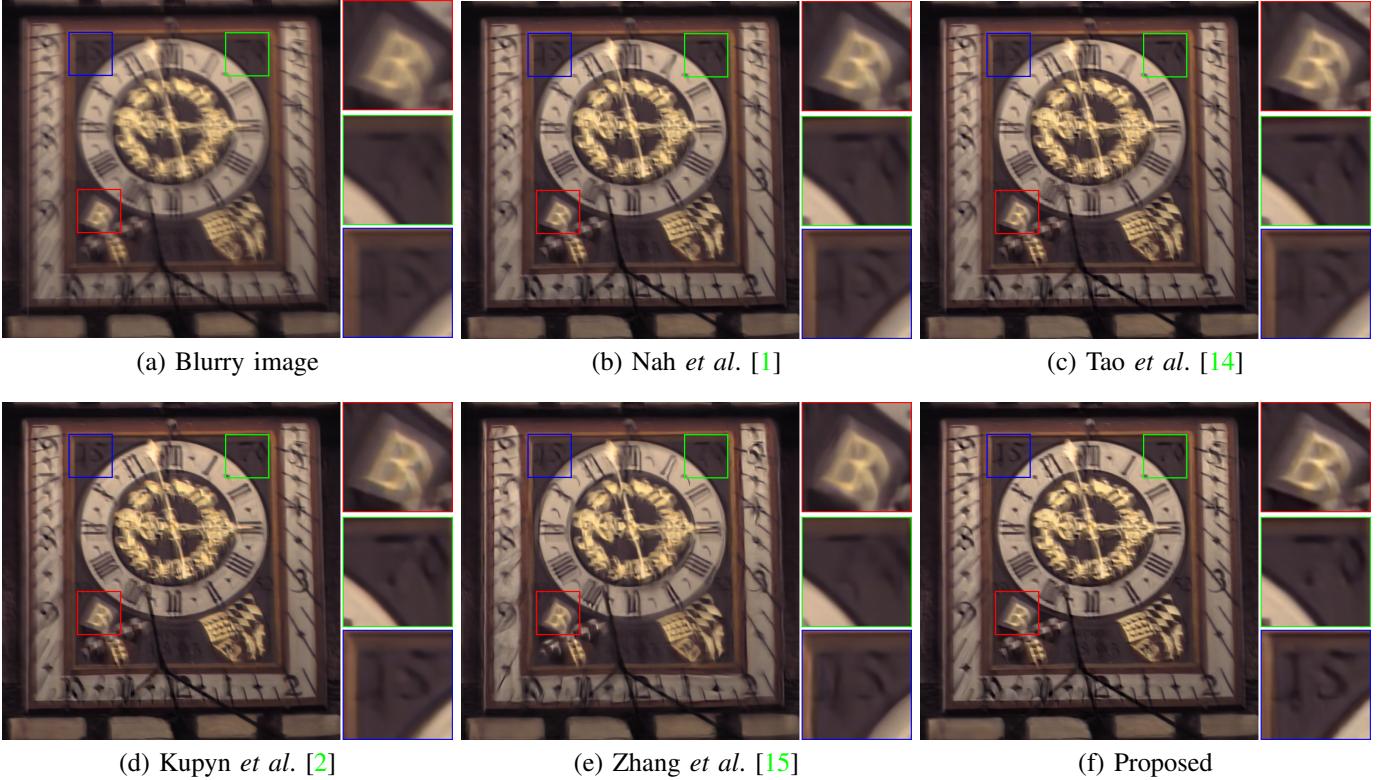


Fig. 5: Deblurring results on the Köhler dataset [17] by different methods.

### C. Ablation Study

It is generally agreed that a larger scale training dataset which covers various image contents and blur models will bring benefit to train a robust deep network. The type of scenes and number of images in the current GoPro dataset, however, are barely sufficient to train an efficient network. Rather than enlarging the training dataset, we propose to integrate the extreme channel prior into CNN and exploit different scales of images for the performance improvement. In order to verify the effectiveness of the ECPeL and IFSE structure, we conduct ablation studies to compare our proposed ECPeNet with several baseline networks. Besides, the quantitative evaluation results of our ECPeNet with different scale levels are also provided.

**Evaluation of ECPeL.** As shown in Figure 3, the proposed ECPeL works in feature domain and is applied to both front and back layers. The reason of our network design is two-fold. (i) In the training phase, we have the ground-truth in the final layer (image domain) to supervise the model learning. Therefore, simply imposing ECP on the final output images cannot achieve significant improvement. (ii) The hidden layers (feature domain) aim to convert the blurry image to sharp image step by step. Considering that the property of ECP also exhibits in hidden layer features, enforcing ECP and sparse constraints in hidden layers can make the learning of early and deeper layers more effective.

We compare our ECPeNet with 6 baseline networks, including **w/o ECP**, **ECP-final**, **ECP-front**, **ECP-front 2**, **ECP-back**, and **ECP-back 2**. Here ‘**w/o ECP**’ refers to the

model without using ECPeL; ‘**ECP-final**’ refers to the model applying ECP and sparse constraint only to the final output image; ‘**ECP-front**’ and ‘**ECP-front 2**’ refer to the model employing one and two ECPeL in the front parts of the network, respectively; and ‘**ECP-back**’ and ‘**ECP-back 2**’ refer to the model using one and two ECPeL in the back parts of the network, respectively. All these 6 baseline networks share the same backbone as our ECPeNet, except some of them take an additional convolutional layer to replace the ECPeL. It is worth pointing out that all these 6 baseline networks use the proposed IFSE structure.

Table II shows the deblurring results. It can be seen that the one without ECPeL performs much worse than the ECPeNet in terms of PSNR (30.834 dB *v.s.* 31.102 dB). If we simply add ECP and sparse constraints on the final output image, as done in previous ECP based methods [11], [12], we can only get about 0.06 dB gain (see model **ECP-final** in Table II). In contrast, placing the ECPeL in the front or back part of the network could benefit dynamic scene deblurring (see models **ECP-front** and **ECP-back** in Table II). However, adding more than one ECPeL in the front or back part of the network could not further boost the performance (see models **ECP-front 2** and **ECP-back 2** in Table II). This is mainly because the ECPeL is introduced to guide the learning of hidden layers. Simply replacing ECPeL in the same part of the network is not effective enough to learn the front or deeper layers. Thus, we place ECPeL in both front and back layers and it achieves the best results.

To further verify the effectiveness of the proposed ECPeL,

Fig. 6: One feature map before and after the dark channel extractor  $\mathbf{D}$  with and without ECP and sparse constraints.

TABLE II: Ablation study on the extreme channel prior embedded layer (ECPeL). The average PSNR (dB) on GoPro testing dataset is shown.

Method	PSNR
w/o ECP	30.834
ECP-final	30.891
ECP-front	30.913
ECP-front 2	30.917
ECP-back	30.952
ECP-back 2	30.955
ECPeNet	<b>31.102</b>

TABLE III: Ablation study on the image full scale exploitation (IFSE). The average PSNR (dB) on GoPro testing dataset is shown.

IFSE	$\times$	$\checkmark$
PSNR	30.96	<b>31.10</b>

we visualize the feature maps before and after the dark channel extractor  $\mathbf{D}$  with and without ECP and sparse constraints. In Figure 6, we randomly visualize one feature map before the extractor  $\mathbf{D}$  (with the  $\text{abs}(\cdot)$  operator). We see that the one with ECP and sparse constraints has sharper edge and its dark channel is sparser.

**Evaluation of IFSE.** To demonstrate the advantages of our IFSE strategy, we compare ECPeNet with a baseline multi-scale architecture which only works in the coarse-to-fine manner. Note that for a fair comparison, the baseline method shares the same backbone as our ECPeNet and uses the same number of scales. Table III verifies our strategy. One can see that the one adopting the proposed IFSE structure (both coarse-to-fine and fine-to-coarse manners) can have 0.14 dB improvement in terms of PSNR index. These comparisons firmly indicate the proposed IFSE structure benefits performance improvement.

**Number of Scales.** We show the quantitative evaluation results of our method with different scale level  $S$  in terms of PSNR over the test data. In Table IV, one can notice that the multi-scale architecture can bring better results for dynamic scene deblurring. Our proposed ECPeNet with  $S = 3$  produces the best results for GoPro testing dataset. While for the Köhler dataset, the average PSNR index can be further improved with the increase of scale level  $S$ . In this work, we take the scale level  $S = 3$  for both GoPro and Köhler dataset.

TABLE IV: Ablation study on the number of scales. The average PSNR (dB) index is listed for different scale level  $S$  on the benchmark datasets.

	$S = 1$	$S = 2$	$S = 3$	$S = 4$
GoPro	30.67	30.81	<b>31.10</b>	30.92
Köhler	25.53	26.28	26.79	<b>26.94</b>

## V. CONCLUSION

In this work, we presented a simple yet effective Extreme Channel Prior embedded Network (ECPeNet) with a novel trainable extreme channel prior embedded layer (ECPeL), which aims to integrate extreme (*i.e.*, dark and bright) channel priors into a deep CNN for dynamic scene deblurring. By extracting the extreme channels of shallow features and enforcing sparsity on them, ECPeNet can regularize the solution space of the network. Additionally, ECPeNet works in both coarse-to-fine and fine-to-coarse manners to exploit information of blurred images at different resolutions to maximize information flow across scales. Benefiting from the extreme channel prior constraint and the effective multi-scale network architecture, the developed ECPeNet outperforms previous dynamic scene deblurring networks by a large margin. Quantitative evaluations on the challenging GoPro dataset showed that the proposed ECPeNet had at least 0.84 dB PSNR gains over the existing state-of-the-arts.

## ACKNOWLEDGMENT

We gratefully acknowledge the support from NVIDIA Corporation for providing us the GPU used in this research.

## REFERENCES

- [1] S. Nah, T. Hyun Kim, and K. Mu Lee, “Deep multi-scale convolutional neural network for dynamic scene deblurring,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3883–3891. [1](#), [2](#), [3](#), [4](#), [6](#), [7](#), [8](#), [9](#)
- [2] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, “Deblurgan: Blind motion deblurring using conditional adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8183–8192. [1](#), [3](#), [7](#), [8](#), [9](#)
- [3] L. B. Lucy, “An iterative technique for the rectification of observed distributions,” *The astronomical journal*, vol. 79, p. 745, 1974. [1](#)
- [4] T. F. Chan and C.-K. Wong, “Total variation blind deconvolution,” *IEEE transactions on image processing*, vol. 7, no. 3, pp. 370–375, 1998. [1](#)
- [5] D. Krishnan, T. Tay, and R. Fergus, “Blind deconvolution using a normalized sparsity measure,” in *CVPR 2011*. IEEE, 2011, pp. 233–240. [1](#), [2](#)
- [6] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, “Deblurring face images with exemplars,” in *European conference on computer vision*. Springer, 2014, pp. 47–62. [1](#)

- [7] L. Sun, S. Cho, J. Wang, and J. Hays, "Edge-based blur kernel estimation using patch priors," in *IEEE international conference on computational photography (ICCP)*. IEEE, 2013, pp. 1–8. [1](#)
- [8] W. Zuo, D. Ren, S. Gu, L. Lin, and L. Zhang, "Discriminative learning of iteration-wise priors for blind deconvolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3232–3240. [1, 2](#)
- [9] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," in *ACM transactions on graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 787–794. [1, 2](#)
- [10] Y. Bahat, N. Efrat, and M. Irani, "Non-uniform blind deblurring by reblurring," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3286–3294. [1](#)
- [11] J. Pan, D. Sun, H. Pfister, and M.-H. Yang, "Blind image deblurring using dark channel prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1628–1636. [1, 2, 3, 6, 9](#)
- [12] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao, "Image deblurring via extreme channels prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4003–4011. [1, 2, 3, 6, 9](#)
- [13] L. Li, J. Pan, W.-S. Lai, C. Gao, N. Sang, and M.-H. Yang, "Learning a discriminative prior for blind image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6616–6625. [1, 2](#)
- [14] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8174–8182. [1, 2, 3, 4, 6, 7, 8, 9](#)
- [15] J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M.-H. Yang, "Dynamic scene deblurring using spatially variant recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2521–2529. [1, 3, 7, 8, 9](#)
- [16] L. Xu, S. Zheng, and J. Jia, "Unnatural 10 sparse representation for natural image deblurring," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1107–1114. [1, 2](#)
- [17] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling, "Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database," in *European conference on computer vision*. Springer, 2012, pp. 27–40. [2, 7, 9](#)
- [18] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," in *Acm transactions on graphics (TOG)*, vol. 27, no. 3. ACM, 2008, p. 73. [2](#)
- [19] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1964–1971. [2](#)
- [20] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in *European conference on computer vision*. Springer, 2010, pp. 157–170. [2](#)
- [21] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," in *Advances in neural information processing systems*, 2009, pp. 1033–1041. [2](#)
- [22] T. Michaeli and M. Irani, "Blind deblurring using internal patch recurrence," in *European Conference on Computer Vision*. Springer, 2014, pp. 783–798. [2](#)
- [23] Z. Hu, S. Cho, J. Wang, and M.-H. Yang, "Deblurring low-light images with light streaks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3382–3389. [2](#)
- [24] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "L<sub>0</sub>-regularized intensity and gradient prior for deblurring text images and beyond," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 2, pp. 342–355, 2017. [2](#)
- [25] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011. [2](#)
- [26] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654. [3, 4](#)
- [27] J. Cai, S. Gu, and L. Zhang, "Learning a deep single image contrast enhancer from multi-exposure images," *IEEE transactions on image processing*, vol. 27, no. 4, pp. 2049–2062, 2018. [3](#)
- [28] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 769–777. [3, 7](#)
- [29] D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. Van Den Hengel, and Q. Shi, "From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2319–2328. [3](#)
- [30] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 479–486. [3](#)
- [31] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," *IEEE transactions on pattern analysis and machine intelligence*, no. 7, pp. 1439–1451, 2016. [3](#)
- [32] A. Chakrabarti, "A neural approach to blind motion deblurring," in *European conference on computer vision*. Springer, 2016, pp. 221–235. [3](#)
- [33] D. Ren, W. Zuo, D. Zhang, J. Xu, and L. Zhang, "Partial deconvolution with inaccurate blur kernel," *IEEE transactions on image processing*, vol. 27, no. 1, pp. 511–524, 2018. [3](#)
- [34] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, 2015, pp. 802–810. [3](#)
- [35] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680. [3](#)
- [36] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *stat*, vol. 1050, p. 26, 2017. [3](#)
- [37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105. [3](#)
- [38] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034. [3](#)
- [39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. [3](#)
- [40] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883. [4](#)
- [41] 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*, 2016, pp. 770–778. [4](#)
- [42] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 286–301. [4](#)
- [43] S. Cho and S. Lee, "Fast motion deblurring," in *ACM Transactions on graphics (TOG)*, vol. 28, no. 5. ACM, 2009, p. 145. [4](#)
- [44] 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*. ACM, 2014, pp. 675–678. [6](#)
- [45] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256. [7](#)
- [46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [7](#)
- [47] T. Hyun Kim and K. Mu Lee, "Segmentation-free dynamic scene deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2766–2773. [7](#)