FormNet: Formatted Learning for Image Restoration

Jianbo Jiao, *Member, IEEE,* Wei-Chih Tu, Ding Liu, *Member, IEEE,* Shengfeng He, *Member, IEEE,* Rynson W.H. Lau, *Senior Member, IEEE,* and Thomas S. Huang, *Life Fellow, IEEE*

***Abstract*—In this paper, we propose a deep CNN to tackle the image restoration problem by learning formatted information. Previous deep learning based methods directly learn the mapping from corrupted images to clean images, and may suffer from the gradient exploding/vanishing problems of deep neural networks. We propose to address the image restoration problem by learning the structured details and recovering the latent clean image together, from the shared information between the corrupted image and the latent image. In addition, instead of learning the pure difference (corruption), we propose to add a *residual formatting layer* and an adversarial block to format the information to structured one, which allows the network to converge faster and boosts the performance. Furthermore, we propose a cross-level loss net to ensure both pixel-level accuracy and semantic-level visual quality. Evaluations on public datasets show that the proposed method performs favorably against existing approaches quantitatively and qualitatively.**

***Index Terms*—Image restoration, format, residual, GAN, CNN.**

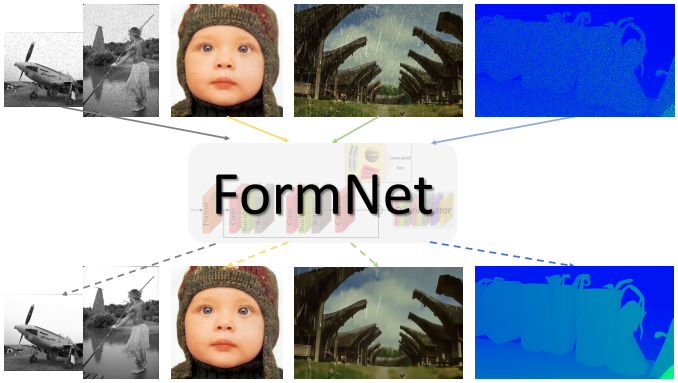


Fig. 1. Illustration of the proposed FormNet on image restoration tasks such as denoising, super-resolution, rain removal, and depth enhancement.

I. Introduction

A lot of image processing algorithms/applications assume the input images to be clean and of high-resolution. However, in practice, these images may suffer from corruption, *e.g.*, noise, or low resolution due to the limitation of digital imaging. The image restoration task aims to handle this problem and recover the latent clean image, including image denoising, super-resolution, artifact removal, *etc*. In general, a corrupted image I_C can be modeled as the latent clean image I added by a certain type of corruption C. Image restoration aims to recover clean image I by separating it from corruption C. Hence, if C can be accurately estimated, I can then be well recovered. Notwithstanding the demonstrated success, most of the traditional image restoration methods are task-specific and cannot be easily adapted to different tasks.

In recent years, deep convolutional neural networks (CNNs) have become very popular in solving many high-level vision problems. There are also some emerging works applying CNNs to low-level vision tasks like image denoising [1], [2], by directly learning the mapping function from a noisy image to its clean version. However, learning such a dense mapping is prone to the gradient vanishing/exploding problems of deep CNNs [3], [4]. Besides, most existing CNN-based methods train the networks based on the pixel-level ℓ_2 norm (or Mean Square Error, MSE) objective, which can easily result in blur artifacts in the final inference.

To resolve the above problems, we start with modeling the image restoration problem as learning the residual, in which the corruption is considered as "residual information" between the clean image and its corrupted version. However, learning the residual solely usually results in remaining artifacts. We observe that the clean image and the corrupted image share similar information in most homogeneous regions, but differ more in highly-structured (*e.g.*, textured) regions. Since both the structured regions and corruptions are high-frequency signals in most cases, directly learning the high-frequency residual is similar to approximating a low-pass filter, and the highly-structured details in the latent image are also filtered out (see Section III-B). Thus, we further propose to extend the baseline network to learn the formatted residual information. To this end, we add a *residual formatting layer* to format the residual to sparsely distributed and more structured information, which is favored by deep residual learning [4]. The highly structured details can then be reconstructed in the following layers.

We further introduce a cross-level loss net to reduce the artifacts caused by the conventional pixel-level ℓ_2 norm. Two gradient layers are added to model the loss in the gradient domain. Besides, high-level similarity measured in the feature domain is taken into consideration, which helps improve the visual quality of the final result. Adversarial learning is also incorporated to format the distribution. We refer to the final framework as *FormNet*. In this paper, we present two

Jianbo Jiao is with the Department of Engineering Science, University of Oxford (e-mail: jianbo@robots.ox.ac.uk).

Ding Liu is with Bytedance Inc., Mountain View, CA, 61801 USA (e-mail: liuding@bytedance.com).

Wei-Chih Tu is with the Graduate Institute of Electronics Engineering, National Taiwan University (e-mail: wctu@media.ee.ntu.edu.tw).

Shengfeng He is with the School of Computer Science and Engineering, South China University of Technology (e-mail: hesfe@scut.edu.cn).

Rynson W.H. Lau is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: Rynson.Lau@cityu.edu.hk).

Thomas S. Huang are with the Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana–Champaign, Urbana, IL, 61801 USA (e-mail: t-huang1@illinois.edu).

S. He and R. W.H. Lau are the corresponding authors.

variants of FormNet, FormResNet and FormGAN, where their objective functions differs. Fig. 1 shows example applications of the FormNet. Extensive evaluation on public datasets shows that the proposed framework performs favorably against the state-of-the-art denoising and other image restoration methods.

A preliminary version of this work was presented in [5]. In comparison to our earlier work, this journal version provides additional analysis on the proposed model, further tuning, and more experimental results. In addition, we further formulate the model into a generative adversarial framework. Experimental results show that the new model reinforces the perceptual visual quality of the image restoration task. Code is publicly available online[1].

The main contributions of this work include:

1) We design a new deep neural network to learn the formatted residual information to reconstruct the structural details for image restoration.
2) We propose a cross-level loss net that supervises the network based on both pixel-level and high-level similarities, resulting in better visual quality compared to traditional MSE-based loss.
3) We incorporate the adversarial learning into the proposed model to set a discriminative objective for a better perceptual quality.
4) Experimental evaluation on the Set14[6], BSD[7] and Kodak[2] datasets shows that the proposed approach performs favorably against state-of-the-art methods. In addition, the proposed method works well across different noise levels and noise types in a single model, and is shown to be able to handle other image restoration tasks.

The remainder of the paper is organized as follows: Section II presents the related work to our method. Section III introduces the proposed method in details. Detailed analysis on the network properties is performed in Section IV, followed by extensive experimental evaluations in Section V. Finally, the paper is concluded in Section VI.

II. Related Work

A. Image Restoration

Image restoration is a widely studied problem in computer vision and remains an active area. Extensive studies have been conducted to solve the problem in the past decades. We refer the readers to a survey [8] on image restoration for more details. Generally, these methods can be categorized into single image based methods and multiple image based methods. Single image based methods like BM3D [9] utilizes non-local information from the corrupted image itself to remove artifact like noise. As image restoration is an ill-posed problem, image priors learned from external dataset (multiple images) are also widely used [10], [11], [12], [13], [14], [15] to reconstruct the latent clean image. Usually the above methods focus on a specific kind of corruptions and the result image tend to be over-smoothed.

[1] https://bitbucket.org/JianboJiao/formnet/
[2] http://r0k.us/graphics/kodak/index.html

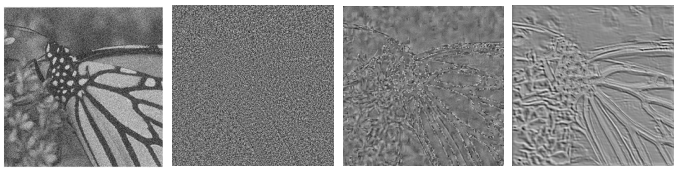
B. Deep Learning for Image Restoration

In recent years, CNN has been applied to multiple low-level vision problems, including image filters [16], [17], [18], super-resolution [19], [20], denoising [2], [21], deconvolution [22], [23], image compression [24], stereo matching [25], optical flow [26], among others. Xie *et al.* [21] combine sparse coding and deep networks to handle problems like complex pattern removing in image inpainting and denoising. Burger *et al.* [2] learn a plain multi-layer perceptron based on a large dataset for denoising, and obtain competitive results to BM3D. Since then, other multi-layer models [27], [28], [29], [30] are also proposed for image restoration.

Although promising result has been achieved, most of these methods focus on learning the dense mapping from observed image to the target one directly, while for many image restoration problems such mapping is close to an identical mapping, which is difficult to learn and prone to the gradients vanishing/exploding problems [3], [4], [20]. The recent proposed residual learning scheme [4] aims to solve such problems in deep neural networks and achieves superior performance on various high-level problems like classification, detection, segmentation, *etc*. For low-level problems, residual learning has also shown its effectiveness in single image super-resolution [20], in which a very deep network is learned efficiently with the help of residual learning. A recent work [31] adopts the same global residual structure [20] for image denoising and achieves promising results. Instead of using residual structure, Zhang et al. [32] propose to train a plain CNN but taking the noise-level map as an additional input. Dong et al. [33] propose a new architecture that iteratively combines multiple denoiser modules and back-projection modules. Liu et al. [34] integrate the wavelet transform to a CNN for a better tradeoff between receptive field size and computational efficiency. Besides CNN-based approaches, other learning-based architectures are also explored for image denoising, *e.g.*, reinforcement learning [35] and recurrent network [36]. More recent work also propose to study the learning-based blind denoising problem by synthesizing realistic noisy-clean training pairs and including real noisy images to the CNN training [37], [38]. Unlike previous learning-based approaches that either stack several blocks or directly learn the difference, we propose a simple architecture by introducing a residual formatting layer to model stochastic residual information into a more structured one. It can handle different noise types and noise levels in a single network and generalizes well to other image restoration tasks. To our knowledge, this is the first approach to tackle multiple noise types and noise levels in a single model.

C. Objective Function

As CNN based methods are data-driven, an objective/loss function is necessary to constrain the training process. Usually the objective is to minimize a ℓ_2 norm (or MSE) loss $L = \|T - I\|^2$ which is used to measure the difference between the network inference I and the target label T. For regression problems like image restoration, such kind of ℓ_2 norm has been widely used in the literature [16], [17],



(a) Noisy image (b) Inference by DiffResNet (c) Remained residual (d) Inference by FormResNet

Fig. 2. Different residual information. (a) is a noisy image corrupted by Gaussian noise with $\sigma = 25$; (b) is the inference output of DiffResNet, which includes both the noise and high-structured regions; (c) is the difference between the ground-truth noise and (b); (d) shows the inference from FormResNet. High-structured details (c) are also removed when doing denoising (subtract (b) from (a)) by DiffResNet while (d), after the formatting layer, well recovers the structured details.

[20], [31]. However, the ℓ_2 norm is prone to result in over-smoothed artifact. While most deep learning methods focus on handcrafting the network structure, little attention has been paid on the design of loss function. In [39], Gatys *et al.* use the feature maps extracted from a basic CNN to model the loss function for image style transfer. As pixel-wise accuracy fails to capture the perceptual similarity for the application of style transfer, the feature map based objective function leads to a good visual quality. The recent popular Generative Adversarial Network (GAN) [40], [41] directly uses a CNN which named discriminator to supervise the training process of the front generator network. Such ingenious structure not only supervises the generator training but also improves the objective part (discriminator) simultaneously. However, the pixel-wise accuracy is not guaranteed and the training of such GANs is unstable. In this paper, we propose a new stable cross-level loss net that integrates pixel-level and semantic-level similarities, so that both pixel-wise accuracy and high-level visual quality are guaranteed.

III. Proposed Method

In many image restoration tasks, the observed image is similar to the target latent one. Taking denoising as an example, the "difference" between noisy image and clean image is the pure noise itself. We observe that most homogeneous regions in the corrupted image and clean image share similar low-frequency information, while the highly-structured (high-frequency) regions between them are relatively different. Due to the inherently different properties in these two regions, learning the difference map only cannot well reconstruct the high-frequency regions, which is illustrated in Fig. 2(c). As a result, we bias the learning process to structured regions, while the homogeneous regions are mainly handled by a formatting layer. In this way, the residual after formatting layer refers to the structure or fine details of the image (Fig. 2(d)). As a reference, we first construct a baseline network to learn the difference between the corrupted image and the target image. Details are given in Section III-A. We introduce the proposed formatted residual learning in Section III-B. Then we describe the adversarial and cross-level loss in Section III-C and Section III-D, respectively.

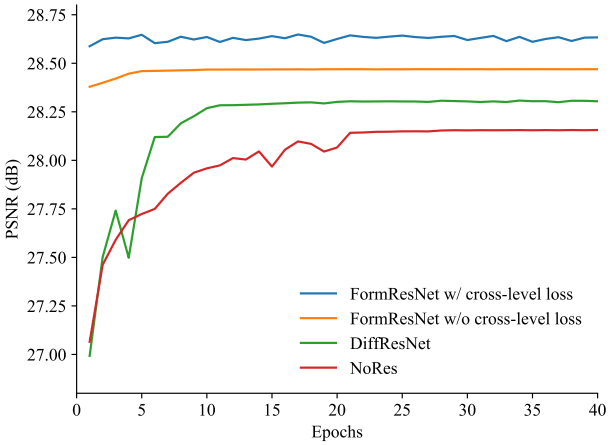


Fig. 3. Performances for residual learning. We compare FormResNet with and without the proposed cross-level loss net, DiffResNet, and NoRes (training without residual learning).

A. Learning the Difference

Conventional CNN-based methods usually learn the mapping from corrupted image to clean image directly [19], [22], [2]. Whereas during the training of deep neural networks, all the image details require to be preserved through many layers. This is prone to the gradient vanishing and exploding problems [3], [4], [20]. Thus we first present the approach of learning the residual mapping $\hat{C} = f(I_C)$ in which only sparse residual information needs to be learned. We name this network as DiffResNet which consists of fully convolutional layers and a skip connection from the network input to the inference. A similar structure is proposed in [31]. A group of convolutional (*conv.*) layers with rectified linear unit (ReLU) are used in the DiffResNet (more details please refer to Sec. V-A). The input of the network is the corrupted image, and the inference is the residual, *i.e.*, corruption. The inference is subtracted from the corrupted input to form the loss function as $\frac{1}{2}\|I - (I_C - f(I_C))\|^2$. By minimizing this objective over the training set $\{I_C^{(i)}, I^{(i)}\}_{i=1}^N$ the parameters of the model can be learned.

B. Learning the Formatted Residual

Due to learning the residual instead of dense mapping, the above DiffResNet architecture is shown to achieve better performance and converge faster than previous "direct learning" (Fig. 3). Such DiffResNet can be considered as approximating a low-pass filter. The advantage of low-pass filter is that the high-frequency artifact (*e.g.*, noise) can be filtered, whereas the drawback is also the "low-pass" property. Besides the artifact, other high-frequency information (structures, edges, *etc*.) is also filtered out. Thus the latent highly-structured regions are difficult to be recovered, as shown in Fig. 2(c). This is because the high-frequency structured regions present inherently different properties to the homogeneous regions.

As a result, we propose a new framework, *FormResNet*, to handle this problem, and the network architecture is shown in Fig. 4. Specifically, we add a *residual formatting layer* (orange part in Fig. 4) to transform the residual into more structured representation. This *format* operation can be also considered

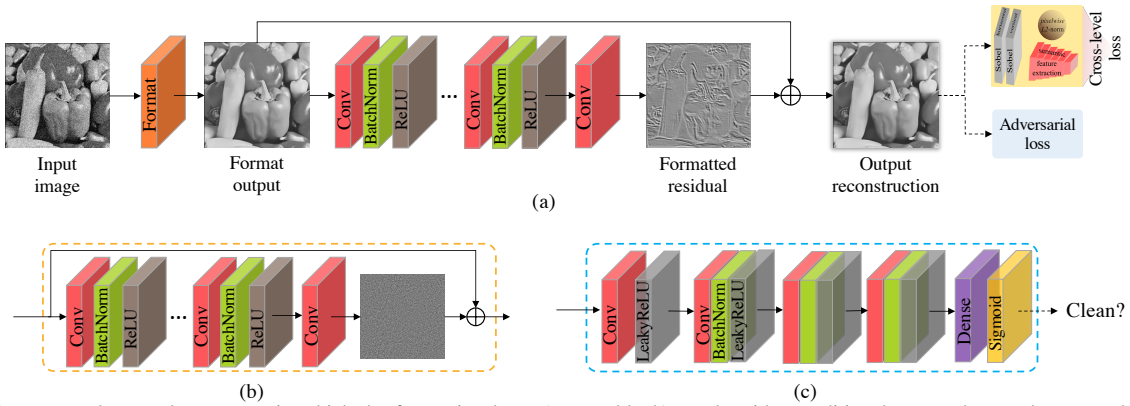


Fig. 4. (a) The proposed network structure, in which the formatting layer (orange block) can be either traditional approaches or deep neural networks. (b) shows an example of formatting layer as CNNs. The cross-level loss incorporates pixel-wise ℓ_2 norm, gradient consistency, and semantic high-level features. (c) Adversarial learning (blue block) is included to achieve a perceptually better visual reconstruction. $\oplus$ denotes pixel-wise addition.

as a pre-processing of the corrupted images. Different from the original scenario, where DiffResNet learns the residual, *i.e.*, the pure difference between the corrupted images and the latent images, now the residual formatting layer allows our model to learn the modified residual, which appears to be spatially more structured, between the pre-processed images and the latent images. That is, the pre-processing step "formats" the residual.

The residual formatting layer is a non-linear operator. It can be constructed by either conventional methods (*e.g.*, BM3D) or parametric models (*e.g.*, neural networks as shown in Fig. 4 (b)). If conventional methods are adopted, the layer is treated as a fixed function and will not be trained together with the following network weights. If the operator is implemented using neural networks, it can be jointly trained with the following layers. Through this formatting layer, the residual map lies more on the image details, instead of random distributed noise. As shown in Fig. 2(d) and Fig. 4, the formatted residual is much sparser than the previous random one, with most regions closer to zero and residual lies in highly-structured regions. The rest part of the network is similar to DiffResNet with several weight layers. The proposed formatting layer well removes high-frequency corruption in homogeneous regions, while the structured regions are left to the remaining part of the network. In this way, the framework takes advantage from both low-pass filter and high-pass filter. When taking neural network as the formatting layer, the FormResNet can be represented in a recursive fashion: $y[k] = x[k] + y[k-1]$ where $y[k]$ is the output of the k^{th} formatting layer and $x[k]$ is the learned formatted residual. Fig. 4 shows the structure when $k = 2$. In this fully convolutional version, the formatting layer is jointly trained with other layers end-to-end.

C. Format the Distribution by Adversarial Learning

In addition to the formatting in the pixel-wise spatial domain, we also propose to format the distribution of the model inference to be similar to that of natural clean images. Adversarial learning [40] is included to achieve the goal. It

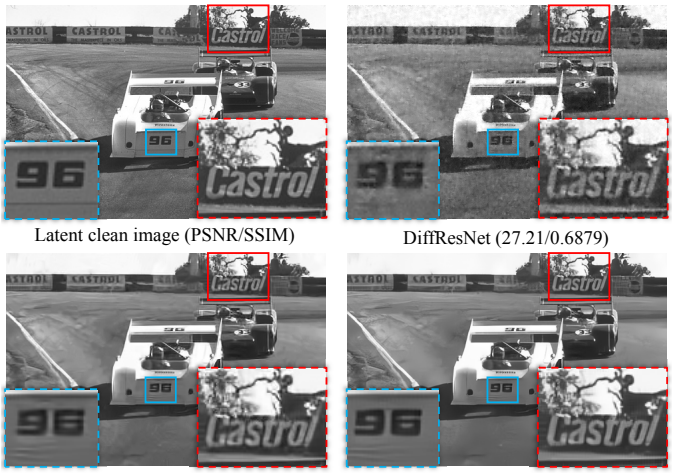


Fig. 5. Visual comparison between DiffResNet and FormResNet. The experiment setups are the same as in Fig. 3.

generally solves a minimax problem defined as:

$$\min_G \max_D \mathbb{E}_{I\sim p_{clean}(I)}[\log D(I)] + \mathbb{E}_{I_C\sim p_G(I_C)}[\log(1 - D(G(I_C)))], \tag{1}$$

where D is a discriminator and G is a generator. By taking the above equation as an objective during the network training, the inference of the FormResNet (here the generator G in Eq. 1) is judged by the discriminator D so that the G network is encouraged to predict more perceptually superior results. Specifically, we add a discriminator network (blue part in Fig. 4) at the end of the FormResNet to discriminate the recovered output and the latent clean images. The new architecture with adversarial learning is termed as *FormGAN*. LeakyReLU is used with $\alpha = 0.2$ following [42]. The filter kernel size increases from 64 to 512 with 4 convolution layers. The final feature maps are followed by a dense layer and a sigmoid function to achieve the classification probability.

D. Cross-level Loss Net

Computers process images in a "pixel-to-pixel" manner, while we humans see more semantic information. In most

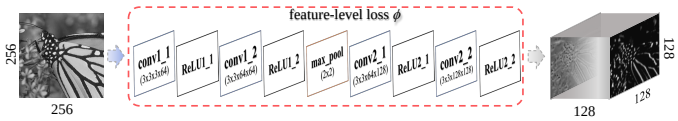


Fig. 6. Illustration on the network structure used for feature-level loss.

CNN-based methods, when measuring the quality of image, a pixel-wise similarity (*e.g.*, ℓ_1 norm, MSE) is adopted as the loss function. Whereas in practice we not only count on pixel-wise performance, but care more about the visual quality in many situations. In addition, only using MSE loss can usually get blurry images, as shown in Fig. 5. Thus in this paper we also consider high-level visual information for the loss description, and propose a cross-level loss function that combines both the pixel-level information and high-level semantic features, to supervise the FormResNet training.

Let x be the corrupted image and y the latent clean image and $F(\cdot)$ as the formatting function in the residual formatting layer. Then the pixel-level loss can be defined as:

$$L_{pix} = \frac{1}{2N}\sum_{i=1}^{N}\left\|r - R(x^i)\right\|^2, \tag{2}$$

where $r = y - F(x)$ is the ground-truth residual image, $R(x^i)$ is the estimated residual by the network, and N is the number of training pairs.

For high-level loss, we first leverage the feature map extracted from a stack of convolutional layers ϕ, which is part of a pre-trained network used for high-level vision. These convolutional layers are concatenated to the end of our FormResNet. The feature-level loss part is inspired by [39], [43] which optimize a style transfer problem by minimizing the difference between feature maps. An illustration of the utilized feature extraction network structure is shown in Fig. 6. As ϕ is only used to extract feature maps for loss computation, all the parameters in ϕ are fixed instead of simultaneously learning with the main body as in [39]. Denote ϕ_l as the feature map after the l-th ReLU layer of ϕ, and the dimension of ϕ_l as $W_l \times H_l \times C_l$ where W, H, C are the width, height, and number of channels respectively. Then the feature-level loss is defined as:

$$L_{ft} = \frac{1}{2N}\sum_{i=1}^{N}\frac{1}{W_l^i H_l^i C_l^i}\left\|\phi_l(y^i) - \phi_l(\hat{y}^i)\right\|^2, \tag{3}$$

where $\hat{y} = F(x) + \hat{r}$ is the recovered image. In addition, information in gradient domain is also leveraged as a high-level loss term:

$$L_{gd} = \frac{1}{2N}\sum_{i=1}^{N}\left|\nabla_h(y^i) - \nabla_h(\hat{y}^i)\right| + \left|\nabla_v(y^i) - \nabla_v(\hat{y}^i)\right|, \tag{4}$$

where ∇_h and ∇_v indicate the horizontal and vertical gradients. The gradient loss term is achieved by two Sobel layers concatenated to the end of FormResNet. By combining the above pixel-level and high-level loss components together, we get the final cross-level loss net:

$$L_c = (1 - \alpha - \beta)\cdot L_{pix} + \alpha \cdot L_{ft} + \beta \cdot L_{gd}, \tag{5}$$

where α, β are balancing weights for the corresponding components.

IV. Network Properties

In this section, we study the properties of the proposed network, including the effectiveness of formatted residual learning, format layer analysis, loss components, network depth and the extension to learn multiple corruptions in a single model.

A. Formatted Residual Learning

Residual learning is suitable for image restoration as in many restoration problems the corrupted image and its corresponding latent image are highly correlated. However, the difference between the corrupted and latent images varies for different problems. It is not that easy to directly apply the same structure (*e.g.*, [20]) to different tasks. As a result, we show the effect of FormResNet compared to learning the difference (DiffResNet). In addition, the influence of the cross-level loss net is also included for the comparison.

In this experiment, image denoising is taken as an example. We use 10 layers (each layer consists of *conv.*, BN, and ReLU except the first and last layer) for the study on BSD100 (Section V-B) and the corrupted noise is an additive Gaussian noise with zero mean and standard deviation of 25. A conventional method (BM3D [9]) is used as the formatting layer in FormResNet (other methods like EPLL [12], WNNM [6] can also be taken, BM3D is just used for simplicity when considering the accuracy and efficiency). In this case, the formatting layer is fixed and only the following layers are updated during the network training. The VGG-16 net [44] pre-trained for classification is utilized as the function ϕ, and $l = 4$ (feature map after $ReLU2\_2$). The performance curve is shown in Fig. 3. We can see that by using residual learning the network converges faster than without residual (NoRes) learning. After adding the residual formatting layer, the network converges in fewer iterations and results in higher performance. When replacing the MSE loss with the proposed cross-level loss, the performance boosts further. The added formatting layer together with the cross-level loss function are powerful for residual learning. A visual comparison is shown in Fig. 5, in which FormResNet shows a better visual quality compared to others, and more details are recovered by the cross-level loss.

B. Format Layer Configuration

We also evaluate different conventional methods as the formatting layer. Here we choose the median filter (*medfilt2* in Matlab), NLM [45], and BM3D [9] for example. Experiment settings are the same as in Fig. 3. The comparison curves are shown in Fig. 7. For NLM, the patch size is fixed to 7×7 while the search window radius is set to 3 (NLM3) and 10 (NLM10). From the curves we can observe that different formatting layers show different performance and all surpass their conventional-versions (without CNN). Specially, we can see that after incorporating NLM as the formatting layer,

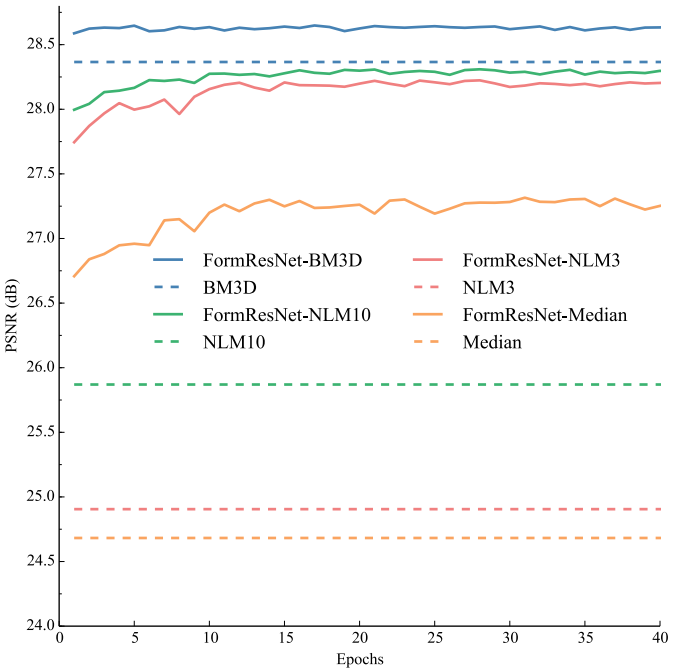


Fig. 7. Performance curve for different formatting layer. Formatting methods Median filter, NLM, and BM3D with their corresponding non-CNN performances are shown for comparison.

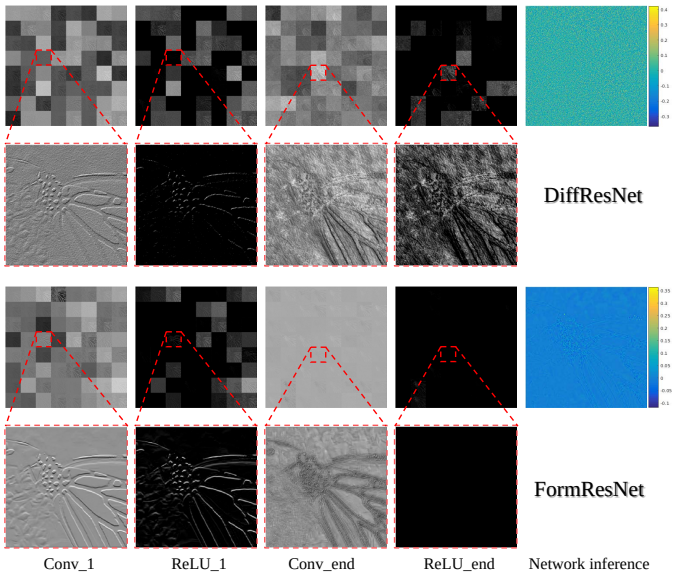


Fig. 8. Analysis on the feature maps in the hidden layers of DiffResNet and FormResNet. The first and third rows show the feature maps from DiffResNet and FormResNet, respectively. The second and fourth rows are the magnifications of the selected regions in the feature maps.

their performances are improved by a large margin (red and green curves), approaching the BM3D. In comparison to their original gap (∼3dB), the "formatted" versions are with much smaller gap (<0.5dB). Compared to FormResNet-BM3D and FormResNet-NLM, the FormResNet-Median has a relatively lower performance. This mainly due to the low performance of the initial median filter and its "median" scheme (while NLM and BM3D reference to neighbor similarity), which influence the following learning process. This study reveals the effectiveness of the proposed formatted learning approach.

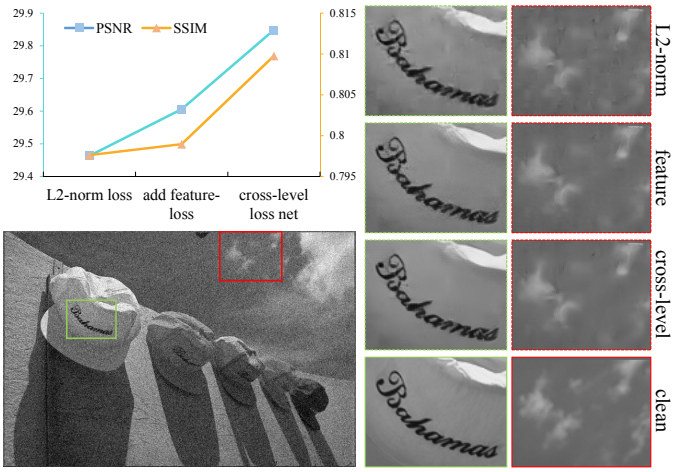


Fig. 9. Effectiveness of different loss terms. Quantitative performance (left corner) and qualitative results (right) are shown for comparison.

C. Feature Map Analysis

We next analyze the feature maps extracted from the hidden layers of the proposed FormResNet and DiffResNet, in the context of image denoising. For instance, Fig. 8 shows the feature maps from the first *conv.* and ReLU layers (_1) and the last *conv.* and ReLU layers (_end). The first two rows are feature maps from DiffResNet, while the rest two rows are those from the FormResNet. The first four columns represent different hidden layers, and the last column is the network inference. We can see that after the activation layer (ReLU), the feature maps become sparser and closer to zero. Such sparse information is easier to learn than the dense nearly-identity mapping [4]. When comparing the feature maps between DiffResNet and FormResNet, we can see that the feature maps of FormResNet focus more on image details and structures, while those of DiffResNet are corrupted with noise and arbitrary patterns (see the enlarged regions in the 2nd and 4th rows). The feature maps of FormResNet are also much sparser than those of DiffResNet, focusing more on fine details. In addition, the inferences of the networks also reveal this property – the output of DiffResNet contains high-frequency random noise while the output of FormResNet contains structured details.

D. Loss Components

In order to evaluate the effectiveness of the proposed cross-level loss net, in this study we compare the performance of different loss terms in the loss net. The FormResNet with $k = 1$ *i.e.*, DiffResNet is taken as the testbed, after which different loss terms are concatenated. Three parts of ℓ_2 norm (MSE) loss, ℓ_2 norm added feature-loss, and the final cross-level loss are concatenated respectively. The restoration task of image denoising is used for the evaluation with added Gaussian noise ($sigma = 25$) on Kodak dataset. The comparison result is shown in Fig. 9. From the quantitative result we can see that with each added loss term, the performance boosts continually. We speculate this is due to the local convexity and smoothness properties of different measurements: ℓ_2-only may has many local minima that prevents a global (or better local) minimum,

TABLE I
PERFORMANCES FOR DIFFERENT CORRUPTIONS, INCLUDING GAUSSIAN, SALT&PEPPER, SPECKLE, AND POISSON NOISE (UNSEEN IN THE TRAINING SET) ON THE KODAK DATASET. AVERAGE PSNR/SSIM VALUES ARE REPORTED FOR QUANTITATIVE EVALUATION. DNCNN-B* INDICATES THE FINETUNED RESULT OF DNCNN-B [31].

Methods	Gaussian15	Gaussian25	Gaussian45	Salt&Pepper	Speckle	Poisson	Average
medfilt2	27.73/0.6987	25.40/0.8745	21.73/0.3515	30.08/0.8682	23.98/0.5184	30.60/0.8745	26.59/0.6976
DnCNN-B*	31.93/0.8624	29.81/0.8037	27.64/0.7297	28.57/0.7876	28.81/0.8087	33.59/0.9019	30.06/0.8156
FormResNet-m	**32.61/0.8842**	**30.34/0.8301**	**27.82/0.7489**	**43.76/0.9945**	**31.01/0.8667**	**38.80/0.9682**	**34.06/0.8821**

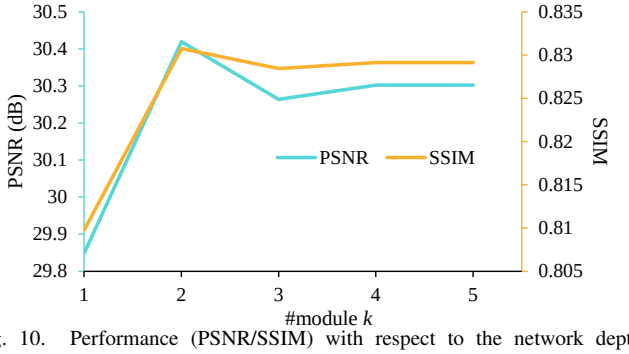


Fig. 10. Performance (PSNR/SSIM) with respect to the network depth (module number).

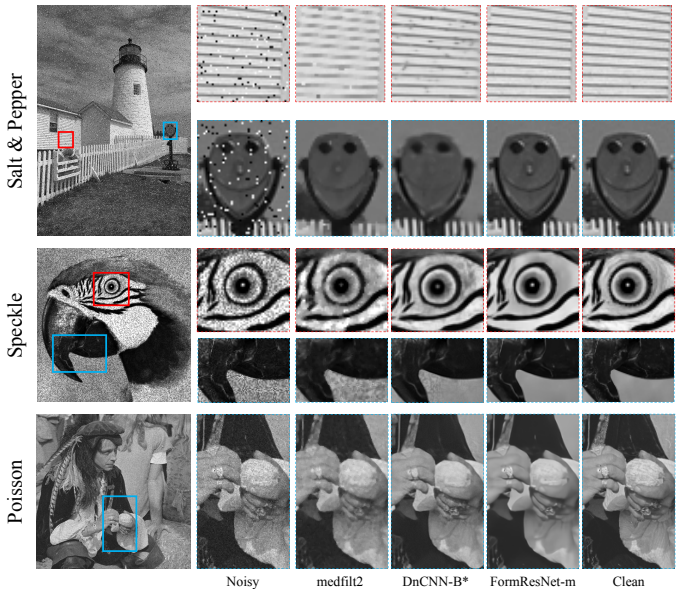


Fig. 11. Visual results on multiple corruptions. *kodim19*, *Parrot*, and *Man* are corrupted with Salt&Pepper, Speckle, and Poisson noise, respectively.

while for the combination with ℓ_1-gradient and feature-level constrains perceptually plausible solutions may lead to a much better minimum. In the qualitative comparison (right side of Fig. 9), the blur artifact caused by ℓ_2 norm is noticeable on the sky region, and for the characters on the hat our cross-level loss recovers more details.

E. Network Depth

Here we study the network performance with respect to the depth of the network. Taking the formatting layer as a 10-layer convolutional module, the performance of different number of modules (k) are evaluated as shown in Fig. 10 (experimental settings are the same as in Section IV-D). We can see that the network performance boosts when using the formatting layer and almost converges after the second module, *i.e.*, the structure shown in Fig. 4. It suggests that the performance not always increases with the depth when the network achieves its capacity.

F. Multiple Corruptions in a Single Model

Usually for CNN-based methods, each kind of corruptions (*e.g.*, noise level or noise type) corresponds to a single model, which is not flexible for real applications. With the proposed formatting layer, different kinds of corruptions can be formatted to an analogous representation and jointly learned for the corresponding residual maps.

In this study, we consider a general blind denoising problem. The training data consists of different noise levels and types: the Gaussian noise with different noise levels, salt&pepper noise and speckle noise. The recursive version of FormResNet with $k = 2$ is trained on the above multi-type data as a single model for all noise types. We denote this network as FormResNet-m. Due to the various noise types, median filter (*medfilt2* in Matlab with default parameters) is used as a baseline method for comparison since many applications use median filter as preprocessing. The state-of-the-art CNN-based denoising method DnCNN [31] is also included for the comparison. As the DnCNN has a blind Gaussian denoising version (DnCNN-B), we finetune their model on our multi-type training data for a fair comparison. Experiment is performed on Kodak dataset for example and the result is shown in Table I. We can see that by training a single model, image corrupted with different noise levels/types can be improved to a large extent. We also test the Poisson noise which is unseen in the training data. Our FormResNet-m also performs well for Poisson noise. Example visual results are shown in Fig. 11.

G. Relation to Regularization

Previous non-CNN optimization-based approaches design their objective functions to promote some empirically observed priors (*e.g.*, sparsity prior [46], color statistics prior [47]). Our FormResNet can also be explained as a regularization operation for image restoration. If we consider the residual formatting layer as the main body, the following residual learning part can be considered as a data-driven regularizer used to identify and fix the unnatural structures. Different from empirically designed regularization priors, our approach automatically learns the natural image prior from data, which is more flexible and likely to handle complicated structures.

TABLE II
COMPARISON RESULTS ON SET14, BSD100, AND KODAK. WE COMPARE DIFFERENT METHODS ON THE AVERAGE PSNR/SSIM VALUES. THE BEST PERFORMANCE IS SHOWN IN RED AND THE SECOND BEST IS SHOWN IN BLUE. THE PROPOSED FORMRESNET CONSISTENTLY SURPASSES OTHER METHODS ON EACH DATASET.

	σ	BM3D	EPLL	WNNM	MLP	DnCNN-S	FFDNet	RLRestore	MWCNN	FormResNet	FormGAN
Set14	15	32.31/0.8959	32.03/0.8952	32.62/0.8981	-	32.75/0.9034	32.64/0.9034	31.18/0.8654	33.03/0.9092	32.77/0.9036	32.65/0.9054
	25	29.79/0.8471	29.48/0.8436	30.02/0.8506	29.70/0.8455	30.22/0.8584	30.20/0.8598	28.77/0.8123	30.51/0.8669	30.30/0.8599	30.12/0.8592
	45	26.55/0.7663	26.33/0.7556	26.76/0.7693	26.60/0.7603	26.91/0.7747	27.06/0.7826	22.87/0.5413	-	27.38/0.7873	26.95/0.7772
	75	23.41/0.6766	22.80/0.6443	23.03/0.6622	23.88/0.6829	23.18/0.6637	23.26/0.6697	16.23/0.2639	-	24.89/0.7093	23.35/0.6661
BSD100	15	30.79/0.8641	30.92/0.8763	31.01/0.8684	-	31.39/0.8831	31.29/0.8826	30.11/0.8368	31.37/0.8819	31.51/0.8848	31.08/0.8712
	25	28.14/0.7842	28.29/0.7979	28.31/0.7893	28.46/0.7960	28.71/0.8106	28.67/0.8111	27.44/0.7488	28.80/0.8157	28.98/0.8153	28.46/0.7903
	45	25.16/0.6680	25.28/0.6743	25.31/0.6707	25.61/0.6656	25.56/0.6883	25.61/0.6930	22.67/0.5406	-	26.34/0.7114	25.37/0.6541
	75	22.56/0.5703	22.20/0.5481	22.23/0.5446	23.25/0.5771	22.29/0.5515	22.31/0.5541	16.98/0.2917	-	24.31/0.6154	22.10/0.5183
Kodak	15	32.19/0.8738	32.12/0.8792	32.45/0.8770	-	32.76/0.8883	32.67/0.8889	31.54/0.8562	32.74/0.8868	32.87/0.8890	32.54/0.8749
	25	29.69/0.8112	29.57/0.8134	29.90/0.8145	29.84/0.8142	30.19/0.8300	30.18/0.8319	28.67/0.7700	30.36/0.8381	30.42/0.8307	30.02/0.8069
	45	26.76/0.7207	26.60/0.7148	26.95/0.7220	26.95/0.7129	27.05/0.7329	27.16/0.7391	23.33/0.5396	-	27.76/0.7457	26.83/0.7009
	75	23.95/0.6413	23.36/0.6126	23.70/0.6295	24.51/0.6461	23.57/0.6273	23.65/0.6330	16.84/0.2690	-	25.58/0.6702	23.18/0.5942

Fig. 12. Commonly used test images (Set14). Top-left to bottom-right: *C.man, House, Peppers, Starfish, Monarch, Airplane, Parrot, Lena, Barbara, Boat, Man, Couple, Montage, Bridge*.

V. EXPERIMENTS AND EVALUATIONS

In this section, we show the detail setups for the proposed network and the performance on several image restoration applications. The description of training details is presented at first. Then we compare the proposed method with several state-of-the-art image restoration methods.

A. Implementation Details

The proposed network is implemented by the *PyTorch* framework on a server equipped with an Nvidia Tesla K40 GPU card and an Intel Core i7-4790 CPU.

We use the fully convolutional FormResNet and FormGAN (training with discriminator) in our experiment (*i.e.*, with and without the blue block in Fig. 4). The final network depth is set to 20, which corresponds to the receptive field of the training patch size. For the aforementioned CNNs, the input layer is a *conv.* layer with 64 filters of size $3 \times 3 \times c$ ($c = 1$ for gray-scale image and $c = 3$ for color image) followed by a ReLU, while the following layers except the last layer are of the same type consist of 64 filters of size $3 \times 3 \times 64$ and followed by ReLUs. The last layer which is used for reconstruction, is a single (or 3 for color image) filter of size $3 \times 3 \times 64$. In order to avoid the resolution reduction problem [17], [19] and predict a dense output with the same size as the input, we pad zero values to the input before each *conv.* layer and it turns out to work well. In addition, we find the batch normalization (BN) [48] is beneficial to the convergence speed and we simply add a BN layer between each of the *conv.* and ReLU layer. The network is trained by using the stochastic gradient descent (SGD) [49] optimizer on mini-batch with weight decay set as 10^{-4} and momentum as 0.9. The mini-batch size is 128 for denoising and 64 for other applications. For all experiments, the number of training iterations is under 40 epochs (some converge in 10 epochs). The learning rate decreases gradually and is initialized to 0.1. The balancing weights for the cross-level loss net are set to $\alpha = \beta = 0.3$. The weights for the network are initialized according to the method proposed in [50], which is shown to be better than random initialization when using non-linear ReLU as the activation function.

B. Quantitative Performance Evaluation

Image denoising is a fundamental problem for many computer vision problems. Theoretically, synthetic training data can be infinitely generated. Whereas in this paper, the training set is generated from a small dataset covers 400 natural images: the BSD500 [7] (*train* and *test* subsets). For testing, we use three datasets: 14 commonly used benchmark images (Set14) [9], [45], [6], [51] as shown in Fig. 12, BSD100 (the *val* subset of BSD500), and the Kodak Lossless Image Suite[3]. In this experiment, only gray-scale images are shown for example (for color images, we can simply adjust the number of input channel to 3). Training images are added Gaussian noise or other types of noises in corresponding experiments. Data augmentation including flipping and rotation are used on the training set. For testing, we use the whole image as input without cropping.

We show both quantitative (Table II) and example qualitative (Fig. 13) performance for the proposed method and present a comparison with other state-of-the-art image denoising methods including: BM3D [9], EPLL [12], WNNM [6], MLP [2], DnCNN [31], FFDNet [32], RLRestore [35], and MWCNN [34]. The implementation of these methods are all from the authors' codes. Metrics of PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural SIMilarity) are calculated for the evaluation. Following [29], the noisy images are quantized to range [0-255] for realistic evaluation. From the result we can see that the proposed approach performs favorably against the state-of-the-art methods and recovers more details and structure, especially on high noise levels. Note that the FFDNet and MWCNN are trained on a much larger dataset (~5,500 images) compared to ours (400 images).

[3] http://r0k.us/graphics/kodak/index.html

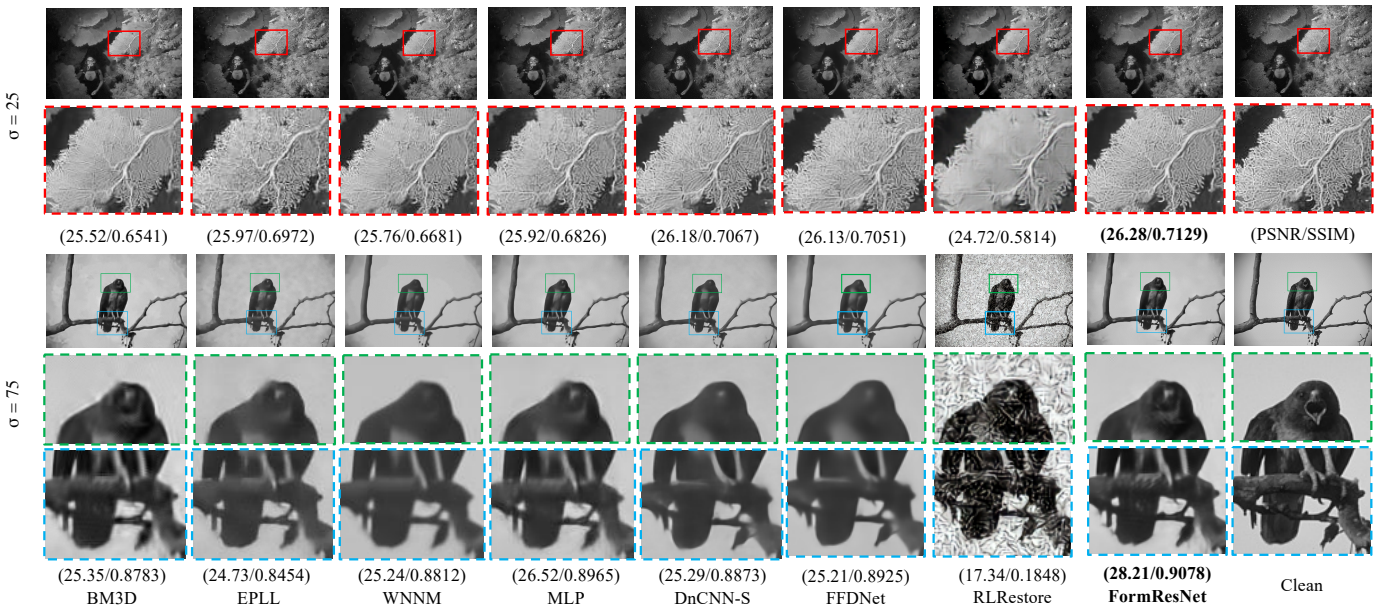


Fig. 13. Qualitative results on *156065* and *42049* from BSD100 with Gaussian noise level 25 and 75, respectively. The proposed FormReNet recovers sharp contours and more details, compared to other methods.

C. Perceptual Performance Evaluation

In order to quantify the visual performance perceptually and evaluate the adversarial learning, we perform a mean opinion score (MOS) test. Following [52], we use rational scores from 1 to 5, indicating the image quality from *bad* to *excellent*. 22 Raters are asked to assign a score to each observation. We use the datasets of Set14 and Kodak for the evaluation. Two noise levels of $\sigma = 45, 75$ are included for the evaluation (two separate sub-test). The methods of BM3D [9], EPLL [12], WNNM [6], MLP [2], DnCNN [31], FFDNet [32], RLRestore [35], FormResNet, and the FormGAN are performed on each image to produce the observations for the raters (unseen to the method names) to score. For reference, the ratings on the ground-truth clean images are also presented (GT). We follow the rater calibration process as in [52]. The experimental performance of perceptual evaluation is shown in Table III. The results show that the proposed FormGAN presents the best visual performance, with the GT acting as an upper-bound ($\rightarrow 5$). Compared to the results in Table II, although the FormGAN not shows a high performance for conventional metrics (PSNR/SSIM), the perceptual quality evaluated by MOS validates its effectiveness. Without the adversarial part, the proposed FormResNet has a lower MOS score than the FormGAN but still demonstrates a better perceptual quality than the other methods.

D. Running Time

Table IV compares the computation time of different methods. Image sizes of 256×256 and 512×512 are included, with Gaussian noise level 25. Computation time on GPU is shown if available. Overall, our running time is comparable to BM3D on CPU. However, our method on GPU is fast, and comparable to the state-of-the-art DnCNN and FFDNet.

TABLE III
PERFORMANCE OF THE MOS TEST ON SET14 AND KODAK.

$\sigma = 45$	BM3D	EPLL	WNNM	MLP	DnCNN-B
Set14	2.56	2.63	2.96	2.81	3.30
Kodak	2.64	2.40	3.16	2.91	3.29
$\sigma = 45$	FFDNet	RLRestore	FormResNet	FormGAN	GT
Set14	3.33	1.48	3.59	3.63	**4.63**
Kodak	3.58	1.29	3.67	3.84	**4.73**
$\sigma = 75$	BM3D	EPLL	WNNM	MLP	DnCNN-B
Set14	2.50	2.40	2.63	2.70	2.90
Kodak	2.50	2.14	2.68	2.68	2.50
$\sigma = 75$	FFDNet	RLRestore	FormResNet	FormGAN	GT
Set14	3.10	1.20	3.37	3.47	**4.93**
Kodak	2.70	1.20	3.40	3.44	**4.94**

TABLE IV
COMPARISON ON COMPUTATION TIME IN SECONDS. TIME ON CPU/GPU (IF AVAILABLE) IS REPORTED.

Size	BM3D	EPLL	WNNM	MLP	DnCNN-B	FFDNet	RLRestore	MWCNN	FormResNet
256^2	0.54	30.67	146.42	2.37	1.05/**0.01**	0.83/**0.01**	0.38/0.37	1.80/0.06	1.11/**0.01**
512^2	2.24	124.54	599.16	6.56	4.60/**0.04**	3.11/**0.02**	0.50/0.52	6.41/0.09	4.62/**0.05**

E. More Applications to Image Restoration

Single Image Super-resolution. Our proposed network can also be applied to single image super-resolution. We use 91 images from [53] as our training set, which is smaller than the final training set (291 images) of [20]. Multiple scaling factors of 2, 3, 4 are trained together for the network. Evaluation is performed on Set5 [54] and the results are shown in Table V. The results of state-of-the-art VDSR [20] training on 91/291 images and the basic bicubic interpolation are included for comparison. From the table we can see that, even training with much fewer images, our FormResNet performs better than VDSR training on the same 91 image set. Our method even performs better than the VDSR training on 291 images. In Fig. 14, example qualitative performance of our model with

Fig. 14. Qualitative performance for single image super-resolution. Sampling scales of $\times 3$ (left) and $\times 4$ (right) are shown for example.

TABLE V
PERFORMANCE COMPARISON ON SINGLE IMAGE SUPER-RESOLUTION ON SET5. NUMBERS IN THE TABLE ARE PSNR VALUES.

Scale	Bicubic	VDSR-91	VDSR-291	FormResNet-91
$\times 2$	33.66	37.06	37.53	**37.55**
$\times 3$	30.39	33.27	33.66	**33.75**
$\times 4$	28.42	30.95	31.35	**31.40**

comparison to prior works on super-resolution are presented.

Single Image Rain Removal. We further apply our method to the problem of rain removal as an illustration to artifact removal. As there is no large public rain dataset, we use the 12 rain image dataset from [55] for our evaluation. Training is performed on randomly selected 10 images from the 12 rain images and the rest 2 images are used for evaluation. Similar to the denoising application, image patches are extracted with data augmentation for the training process. For comparison, a single image rain removal method DSC [56], and DnCNN-B finetuned on the 10 training images are used here. Results are shown in Fig. 15, in which the top rows show comparison on synthetic rainy images, while the bottom rows show performances on real rainy scenes. We can see that most of the rain artifact on the input image is removed by our method. Far fewer rain streaks can be observed compared to other methods.

Single Image Blind Deblurring. The proposed method can also be applied to the blind image deblurring problem [13]. Here we use the challenging GoPro dataset from [57] and follow the same train/test split. The experimental settings are kept the same as in the above experiments except the patch size which is modified to 128 (following prior works) and the batch size to 32 to cope with the GPU memory. Example results are shown in Fig. 16, with comparison to Liu *et al.* [13]. It can be observed that most motion blurs are removed by our method with sharper reconstructions.

Other Applications. The powerful capacity of our network can also benefits other image restoration applications like natural image inpainting and depth map enhancement, as shown in Fig. 17 and Fig. 18, respectively. For natural image inpainting, 30% and 50% of the total number of pixels are randomly removed from the original image, while for depth image enhancement both a downsample (with scale=2, 3) and pixel removal (with 20%, 50%) are performed on the clean sharp depth map. The training set for inpainting is the BSD400 [7], while 344 random selected images from [58] are used to train the depth enhancement application. The

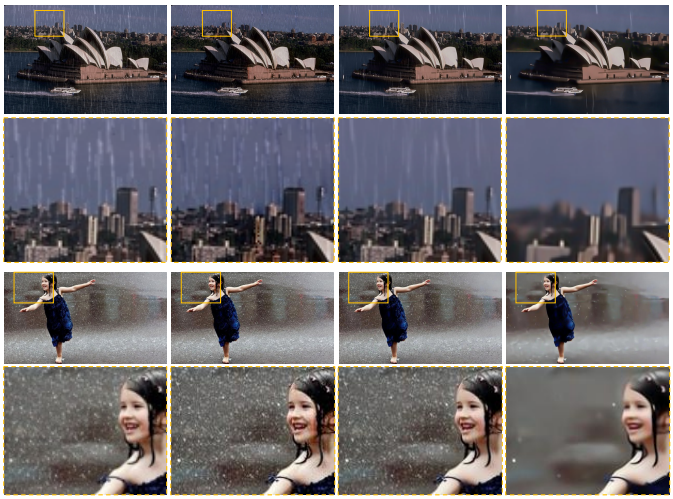


Fig. 15. Comparison result on rain removal for both synthetic data (top two rows) and real rainy scene (bottom two rows). Fewer rain streaks can be observed on the result of our method, compared to those of DSC [56] and DnCNN-B [31].

Fig. 16. Comparison result on image deblurring. Fewer motion blur and artifacts can be observed on the result of the proposed method, compared to those of Liu *et al.* [13] (zoom-in for a better comparison).

performance for these additional applications again validates the effectiveness of the proposed approach.

VI. CONCLUSION

In this paper, we have presented a formatted learning framework for image restoration. A residual formatting layer is proposed to format the residual information to structured details. The proposed cross-level loss net contributes to high visual quality by leveraging semantic-level similarity. An additional adversarial learning block is included to further boost the perceptual quality. Evaluations on multiple public datasets show that the proposed FormNet (FormResNet and FormGAN) performs favorably against existing image restoration methods, while being very efficient. FormNet is also able to handle different corruptions (noise types and noise levels) in a single model. By applying different operations to the residual formatting layer, we believe the proposed framework can be easily extended to more other low-level vision problems.

Fig. 17. Application on natural image inpainting. Top: 30% pixels are randomly removed from the image; Bottom: 50% pixels are randomly removed from the image.

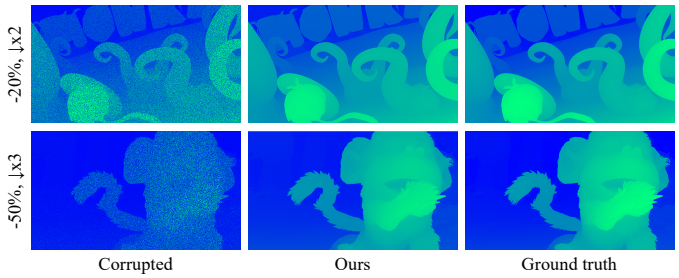


Fig. 18. Application on depth map enhancement. Top: 20% random pixel removal and $\times 2$ downsampling are applied to the depth map; Bottom: 50% random pixel removal and $\times 3$ downsampling are applied to the depth map.

Acknowledgment

This work was partially supported by the EPSRC Project Seebibyte (EP/M013774/1), the Hong Kong PhD Fellowship Scheme from the Research Grants Council of Hong Kong, the National Natural Science Foundation of China (No. 61972129, No. 61972162, and No. 61702194), the Guangzhou Key Industrial Technology Research fund (No. 201802010036), the CCF-Tencent Open Research fund (CCF-Tencent RAGR20190112), and the NVIDIA Corporation with the donation of GPU card.

References

[1] V. Jain and S. Seung, "Natural image denoising with convolutional networks," 2009. 1

[2] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d?" in *CVPR*, 2012. 1, 2, 3, 8, 9

[3] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994. 1, 2, 3

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 1, 2, 3, 6

[5] J. Jiao, W. C. Tu, S. He, and R. W. H. Lau, "Formresnet: Formatted residual learning for image restoration," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017. 2

[6] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *CVPR*, 2014. 2, 5, 8, 9

[7] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE TPAMI*, vol. 33, no. 5, pp. 898–916, 2011. 2, 8, 10

[8] L. Zhang and W. Zuo, "Image restoration: From sparse and low-rank priors to deep priors [lecture notes]," *IEEE Signal Processing Magazine*, vol. 34, no. 5, pp. 172–179, 2017. 2

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE TIP*, vol. 16, no. 8, pp. 2080–2095, 2007. 2, 5, 8, 9

[10] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE TIP*, vol. 15, no. 12, pp. 3736–3745, 2006. 2

[11] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *CVPR*, 2005. 2

[12] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *ICCV*, 2011. 2, 5, 8, 9

[13] G. Liu, S. Chang, and Y. Ma, "Blind image deblurring using spectral properties of convolution operators," *IEEE TIP*, vol. 23, no. 12, pp. 5047–5056, 2014. 2, 10

[14] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE TIP*, vol. 20, no. 7, pp. 1838–1857, 2011. 2

[15] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE TPAMI*, vol. 32, no. 6, pp. 1127–1133, 2010. 2

[16] L. Xu, J. S. Ren, Q. Yan, R. Liao, and J. Jia, "Deep edge-aware filters," in *ICML*, 2015. 2, 3

[17] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *ECCV*, 2016. 2, 3, 8

[18] S. Liu, J. Pan, and M.-H. Yang, "Learning recursive filters for low-level vision via a hybrid neural network," in *ECCV*, 2016. 2

[19] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *ECCV*, 2014. 2, 3, 8

[20] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *CVPR*, 2016. 2, 3, 5, 9

[21] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," 2012. 2

[22] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," 2014. 2, 3

[23] T. Tirer and R. Giryes, "Image restoration by iterative denoising and backward projections," *IEEE TIP*, vol. 28, no. 3, pp. 1220–1234, March 2019. 2

[24] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, "Learning convolutional networks for content-weighted image compression," in *CVPR*, 2018. 2

[25] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, pp. 1–32, 2016. 2

[26] A. Dosovitskiy, P. Fischery, E. Ilg, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox *et al.*, "Flownet: Learning optical flow with convolutional networks," in *ICCV*, 2015. 2

[27] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *CVPR*, 2014. 2

[28] Y. Chen, W. Yu, and T. Pock, "On learning optimized reaction diffusion processes for effective image restoration," in *CVPR*, 2015. 2

[29] R. Vemulapalli, O. Tuzel, and M.-Y. Liu, "Deep gaussian conditional random field network: A model-based deep network for discriminative denoising," in *CVPR*, 2016. 2, 8

[30] D. Liu, B. Wen, X. Liu, Z. Wang, and T. S. Huang, "When image denoising meets high-level vision tasks: A deep learning approach," *arXiv preprint arXiv:1706.04284*, 2017. 2

[31] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE TIP*, vol. PP, no. 99, pp. 1–1, 2017. 2, 3, 7, 8, 9, 10

[32] K. Zhang, W. Zuo, and L. Zhang, "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE TIP*, vol. 27, no. 9, pp. 4608–4622, 2018. 2, 8, 9

[33] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *IEEE TPAMI*, vol. 41, no. 10, pp. 2305–2318, 2018. 2

[34] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *CVPR Workshops*, 2018. 2, 8

[35] K. Yu, C. Dong, L. Lin, and C. Change Loy, "Crafting a toolchain for image restoration by deep reinforcement learning," in *CVPR*, 2018. 2, 8, 9

[36] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, "Non-local recurrent network for image restoration," in *NeurIPS*, 2018. 2

[37] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *CVPR*, 2019. 2

[38] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *CVPR*, 2018. 2

[39] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *CVPR*, 2016. 3, 5

[40] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014. 3, 4

[41] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015. 3

[42] ——, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *ICLR*, 2016. 4

[43] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV*, 2016. 5

[44] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015. 5

[45] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *CVPR*, 2005. 5, 8

[46] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *ICCV*, 2009. 7

[47] S. Ono and I. Yamada, "A convex regularizer for reducing color artifact in color image recovery," in *CVPR*, 2013. 7

[48] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015. 8

[49] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951. 8

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *ICCV*, 2015. 8

[51] H. Liu, R. Xiong, J. Zhang, and W. Gao, "Image denoising via adaptive soft-thresholding based on non-local samples," in *CVPR*, 2015. 8

[52] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network." in *CVPR*, 2017. 9

[53] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE TIP*, vol. 19, no. 11, pp. 2861–2873, 2010. 9

[54] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *BMVC*, 2012. 9

[55] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *CVPR*, 2016. 10

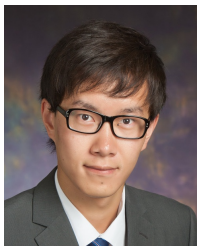
[56] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *ICCV*, 2015. 10

[57] S. Nah, T. Hyun Kim, and K. Mu Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *CVPR*, 2017. 10

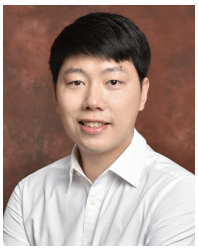
[58] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *CVPR*, 2016. 10

Jianbo Jiao is a Postdoctoral Researcher in the Department of Engineering Science at the University of Oxford. He obtained his Ph.D. degree in Computer Science from City University of Hong Kong in 2018. He was the recipient of the Hong Kong PhD Fellowship. He was a visiting scholar with the Beckman Institute at the University of Illinois at Urbana-Champaign from 2017 to 2018. His research interests include computer vision and machine learning.

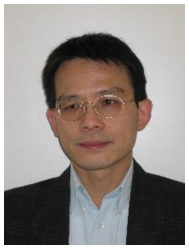
Wei-Chih Tu is a Ph.D. candidate with the Graduate Institute of Electronics Engineering at National Taiwan University, Taipei, Taiwan. He received the B.S. degree in Electrical Engineering from the National Taiwan University in 2012. His research interests include computer vision, multimedia, and deep learning.



Ding Liu received the Ph.D. degree from the University of Illinois at Urbana-Champaign, USA, in 2018. He is a Research Scientist in Bytedance Inc., Mountain View, CA, USA. His research experience encompasses low-level vision problems, including image/video restoration and enhancement. He has broad research interests in the area of computer vision, image processing and deep learning.



Shengfeng He is an Associate Professor in the School of Computer Science and Engineering, South China University of Technology. He was a Research Fellow at City University of Hong Kong. He obtained his B.Sc. degree and M.Sc. degree from Macau University of Science and Technology, and the Ph.D degree from City University of Hong Kong. His research interests include computer vision, image processing, computer graphics, and deep learning.



Rynson W.H. Lau received his Ph.D. degree from University of Cambridge. He was on the faculty of Durham University and is now with City University of Hong Kong. Rynson serves on the Editorial Board of Computer Graphics Forum, and Computer Animation and Virtual Worlds. He has served as the Guest Editor of a number of journal special issues, including ACM Trans. on Internet Technology, IEEE Trans. on Multimedia, IEEE Trans. on Visualization and Computer Graphics, and IEEE Computer Graphics & Applications. He has also served in the committee of a number of conferences, including Program Co-chair of ACM VRST 2004, ACM MTDL 2009, IEEE U-Media 2010, and Conference Co-chair of CASA 2005, ACM VRST 2005, ACM MDI 2009, ACM VRST 2014. Rynson's research interests include computer graphics and computer vision.

Thomas S. Huang (F'01) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., and the M.S. and Sc.D. degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge. He was on the Faculty of the Department of Electrical Engineering at MIT from 1963 to 1973; and on the Faculty of the School of Electrical Engineering and Director of its Laboratory for Information and Signal Processing at Purdue University from 1973 to 1980. In 1980, he joined the University of Illinois at Urbana-Champaign, where he is currently a William L. Everitt Distinguished Professor of Electrical and Computer Engineering, and a Research Professor with the Coordinated Science Laboratory, and also with the Beckman Institute for Advanced Science, as the Technology and Co-Chair of the Institute's major research theme Human Computer Intelligent Interaction. His professional interests lie in the broad area of information technology, especially the transmission and processing of multidimensional signals. He has published 21 books, and over 600 papers in network theory, digital filtering, image processing, and computer vision. Dr. Huang is a Member of the National Academy of Engineering; a Member of the Academia Sinica, Republic of China; a Foreign Member of the Chinese Academies of Engineering and Sciences; and a Fellow of the International Association of Pattern Recognition, and the Optical Society of America. Among his many honors and awards: the Honda Lifetime Achievement Award, the IEEE Jack Kilby Signal Processing Medal, and the King-Sun Fu Prize of the International Association for Pattern Recognition.