PID Controller-Guided Attention Neural Network Learning for Fast and Effective Real Photographs Denoising

Ruijun Ma, Bob Zhang, *Senior Member, IEEE*, Yicong Zhou, *Senior Member, IEEE*, Zhengming Li, and Fangyuan Lei

***Abstract*— Real photograph denoising is extremely challenging in low-level computer vision since the noise is sophisticated and cannot be fully modeled by explicit distributions. Although deep-learning techniques have been actively explored for this issue and achieved convincing results, most of the networks may cause vanishing or exploding gradients, and usually entail more time and memory to obtain a remarkable performance. This article overcomes these challenges and presents a novel network, namely, PID controller guide attention neural network (PAN-Net), taking advantage of both the proportional-integral-derivative (PID) controller and attention neural network for real photograph denoising. First, a PID-attention network (PID-AN) is built to learn and exploit discriminative image features. Meanwhile, we devise a dynamic learning scheme by linking the neural network and control action, which significantly improves the robustness and adaptability of PID-AN. Second, we explore both the residual structure and share-source skip connections to stack the PID-ANs. Such a framework provides a flexible way to feature residual learning, enabling us to facilitate the network training and boost the denoising performance. Extensive experiments show that our PAN-Net achieves superior denoising results against the state-of-the-art in terms of image quality and efficiency.**

***Index Terms*— Attention neural network, image denoising, proportional-integral-derivative (PID) controller, real photograph.**

Manuscript received April 30, 2020; revised August 27, 2020 and December 3, 2020; accepted December 25, 2020. Date of publication January 15, 2021; date of current version July 7, 2022. This work was supported in part by the University of Macau under Grant MYRG2019-00006-FST, in part by the Youth Innovation Project, Department of Education, Guangdong Province, under Grant 2020KQNCX040, in part by the National Natural Science Foundation of China under Grant 61702117, in part by the Science and Technology Program of Guangzhou under Grant 201804010355, and in part by the Special Projects for Key Fields in Higher Education of Guangdong under Grant 2020ZDZX3077. *(Corresponding author: Bob Zhang.)*

Ruijun Ma is with the PAMI Research Group, Department of Computer and Information Science, University of Macau, Taipa, Macau, and also with the Guangdong Industrial Training Center, Guangdong Polytechnic Normal University, Guangzhou 510665, China (e-mail: yb97442@um.edu.mo).

Bob Zhang is with the PAMI Research Group, Department of Computer and Information Science, University of Macau, Taipa, Macau (e-mail: bobzhang@um.edu.mo).

Yicong Zhou is with the Faculty of Science and Technology, University of Macau, Taipa, Macau (e-mail: yicongzhou@um.edu.mo).

Zhengming Li is with the Guangdong Industrial Training Center, Guangdong Polytechnic Normal University, Guangzhou 510665, China (e-mail: gslzm@gpnu.edu.cn).

Fangyuan Lei is with the School of Electronics and Information, Guangdong Polytechnic Normal University, Guangzhou 510665, China (e-mail: leify@gpnu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2020.3048031.

Digital Object Identifier 10.1109/TNNLS.2020.3048031

I. Introduction

BENEFITING from the great progress in computer vision, image denoising has achieved noticeable improvements in recent years. In general, the purpose of image denoising is to recover a clean image from its noisy observation, which is an essential preprocessing step in image analysis systems. In the past decades, a vast amount of promising approaches [1]–[19] have been proposed and many efforts have been devoted to additive white Gaussian noise (AWGN) removal. Despite the great successes made, most of these approaches may not perform well on noise in real photographs. According to [20]–[23], the image noise corrupted in a CCD or CMOS camera system has multiple sources, such as dark current noise, amplifier noise, and so on. The noise type, in general, is non-Gaussian and inhomogeneous and can easily be influenced by different camera devices and in-camera processing pipelines. These make the real noise much more sophisticated and different from AWGN. Therefore, the performance of many AWGN denoising algorithms may be limited when applied to real noisy photographs.

To address the above-mentioned drawbacks, some algorithms [24]–[27] utilize Gaussian or mixture of Gaussians (MoG) distributions to estimate the noise on the real images. In fact, the real noise is spatially variant, signal-dependent and can be much more complex. Thus, using such explicit distributions may still be inflexible enough to well estimate the underlying noise model. Besides the above-mentioned modeling-based methods, there are several attempts [28]–[31] to cope with real noisy images by learning image prior models from the sparse coding framework. These methods, however, cannot capture the full characteristics of the realistic noise property because the learned priors are generally defined explicitly.

In recent years, with the renaissance of the convolutional neural network (CNN), substantial progress has been achieved in this research area. The CNN-based methods [32]–[39], such as Real Image Denoising Network (RIDNet) [35] and variational denoising network (VDN) [37], can leverage the advantage of a deep learning framework to effectively accumulate knowledge from large data sets. As such, they can break through the limitations of the aforementioned methods and achieve more impressive results. However, it should be pointed out that most of these methods, especially those with deep architecture, can suffer from two major drawbacks. First, some of these deep networks are easily affected by

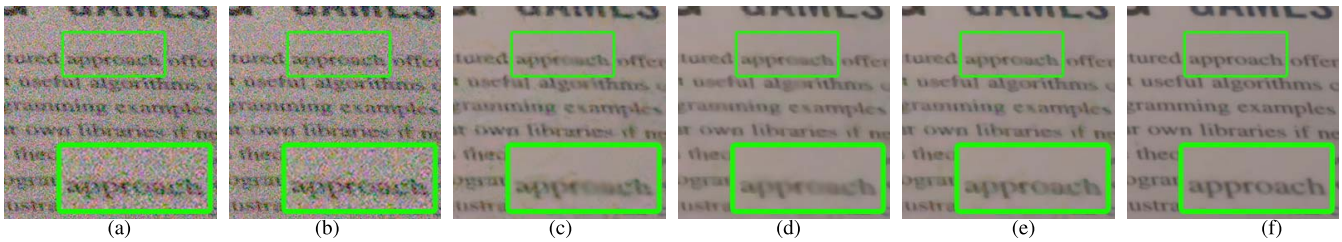


Fig. 1. Denoising results of different algorithms on real noisy photograph from SIDD. Our method is able to recover visually more pleasant and artifact-free output. Please zoom in for a better view. (a) Noisy input. (b) DnCNN-B [14]. (c) CBDNet [36]. (d) RIDNet [35]. (e) VDN [37]. (f) Ours.

the vanishing or exploding gradients problem. Second, these denoising algorithms are able to achieve a satisfactory denoising quality but entail heavy computational loads and memory consumption.

For image denoising problems, a noisy image $\boldsymbol{y}$ can be expressed as: $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{v}$, where $\boldsymbol{v}$ is the noisy observation and $\boldsymbol{x}$ is the latent clean image. Here, $\boldsymbol{v}$ can be viewed as the deviation between $\boldsymbol{y}$ and $\boldsymbol{x}$. In the control domain, a proportional-integral-derivative (PID) controller is able to use the error (i.e., deviation) to update the system output toward the desired value. When a feedback control mechanism is introduced, the control procedure will be more fast and robust [40]–[42]. Here, one can observe that the *error* in the PID controller shares the same spirit as the *deviation* used in the noisy image. This motivates us to utilize a PID controller to the field of image denoising. According to the previous analysis, a discriminative deep learning based method has good effects on image denoising even though there are still some deficiencies. Therefore, we take a further step and make a combination of the CNN model and PID control technology for the challenging real photography denoising task.

In this article, we propose a novel denoiser, namely, the PID controller guide attention neural network (PAN-Net). The basic idea is to make the concatenation of the PID controller and the attention model become a dynamic control system. The control goal is to achieve an efficient and effective boost to the ability of network discriminative representation learning, thus separating the noise from the image content. Specifically, we propose a PID-attention network (PID-AN), which consists of a PID controller and one attention model. In PID-AN, the attention model aims to capture the noise-free image features, while the PID controller exploits the error to guide the attention model for better feature representation learning. Our PID-AN is based on a feedback system, which delivers the benefits of adaptive and robust learning. To boost the network performance, we stack several PID-ANs together with share-source skip connections (SSCs). Compared with the aforementioned methods, our PAN-Net enjoys several pleasant properties. First, by building a modular learning framework, we are able to avoid the occurrence of gradient vanishing or exploding problems. Second, we convert the PID technology into neural network learning, which helps us to improve network performance while encouraging efficiency and robustness. Third, the adaptively strong learning ability of PID-AN allows us to cope with various and sophisticated real-world noise soundly and feasibly. Some visual comparisons can be seen in Fig. 1.

To sum up, the major contributions of this article are threefold.

1) We propose a novel paradigm for real photograph denoising by introducing the PID controller and attention neural network. To the best of our knowledge, we are the first to explore the potential of the PID technology for the image denoising task.
2) We link the control action and neural network learning to devise an adaptive dynamic network. This scheme is simple yet effective at strengthening the discriminative learning ability. Besides this, such an integrated scheme can also ensure a robust and flexible denoising performance.
3) We evaluate our method on both synthetic and real noisy images. The results show that the proposed PAN-Net produces perceptually appealing denoising results against the state-of-the-art. Meanwhile, it is worthy to note that our PAN-Net is highly efficient and has an encouraging performance to eliminate the sophisticated noise while well-preserving fine-scale image details.

II. Related Work

A. Image Denoising

As a classic topic, image denoising has been extensively studied in recent years. In this section, we mainly focus our discussion on the representative denoising methods.

1) AWGN Image Denoising: Most of denoising methods [1]–[19] are developed for AWGN removal. Early approaches [1]–[8] exploited the sparsity, or the self-similarity property to model image priors and performed well on AWGN denoising. Discriminative image prior [11]–[16] also attracts much attention in recent years. In general, these methods solved the denoising problem by first learning priors or mapping functions from paired training data and then applying the learned models to remove the noisy observation. For instance, Zhang *et al.* [14] developed 17-layer denoising CNN (DnCNN) for blind Gaussian denoising by combining residual learning [43] and batch normalization (BN) [44]. Recently, Zhang *et al.* [16] incorporated residual learning and dense connection structure and proposed a residual dense network (RDN) for image restoration. It achieved a significant improvement in AWGN removal compared with the previous methods.

Despite the considerable results of the above-mentioned methods, most of these approaches are specific trained for AWAG removal, making them much less effective for the sophisticated real noise.

2) Real Photograph Denoising: Regarding the real photograph denoising problems, various models have been developed in recent years. Research on this issue can be categorized into three major groups, i.e., model-based methods, sparse methods, and deep network learning-based methods.

Many model-based methods [24]–[27] follow the idea that the real noise can be modeled by Gaussian or MoG distributions and remove the noise via the estimated noise model. For instance, Zhu *et al*. [27] estimated the noise with MoG distribution and proposed to build a "Dependent Dirichlet Process Tree" to cope with the real-world noisy images. Nonetheless, the real noise is signal dependent and much more complex, making it fairly difficult to be well estimated by explicit distributions.

For the second group, most of the sparse-based models [28]–[31] focus on integrating sparse coding and dictionary learning technologies to address the denoising problem of real noise. After solving a sparse linear framework, the clean signal can be recovered from the noisy input. Although these approaches have obtained competitive denoising quality, their learned image priors mostly rely on human knowledge. As a consequence, the capacity of characterizing the complex image textures and structures can be reduced, which may limit a performance boost.

The third group of approaches [32]–[39] based on deep network learning have recently become a new research trend and received much attention. Recently, a convolutional blind denoising network (CBDNet) [36] was proposed to estimate the noise via building a more realistic noise model according to the in-camera process pipeline. Later, RIDNet [35] established the attention mechanism to exploit and learn prominent image features, enforcing more satisfying results. Yue *et al*. [37] developed a variational denoising network (VDN), which employed variational inference technology to estimate the noise distribution within a Bayesian framework. Current CNN-based discriminative learning methods have achieved good performance, however, such deep networks might be faced with performance saturation or involve a complex training procedure. Conversely, by leveraging a dynamic controller and neural network, our proposed PAN-Net can overcome the aforementioned drawbacks in a robust and feasible manner. Comprehensive empirical results demonstrate its superior denoising performance on popular benchmarks.

B. PID Controller

As popular solutions to practical control systems, a PID controller has drawn substantial attention in the automatic control area for decades. Owing to its functionality, simplicity, and reliability, the PID controller has prompted many applications, such as autonomous vehicles [40], [41], industrial robots [42], variable-frequency power [45], to list a few. The objective of a PID controller can offer a robust and fast way to approximate the system output to the desired value. Mathematically, a PID controller exploits the error $e(t)$ between the reference point and actual output, before employing proportional, integral, and derivative actions of $e(t)$ to form a correction $u(t)$ for system control. The implemental procedure can be explained as

$$u(t) = K_p e(t) + K_i \int_0^t e(t)\,dt + K_d \frac{de(t)}{dt} \tag{1}$$

where K_p, K_i, and K_d represent the proportional, integral, and derivative coefficients of the PID controller, respectively.

Inspired by the fact that the PID controller can provide robust and adaptive performances to various systems, we propose to integrate the PID controller and attention neural network to form a learning system for the challenging real photograph denoising task. Recently, there is another method [46] that involves PID control technique and neural network learning as well. However, our work differs from [46] in three aspects.

First, the motivation is different. The method of [46] proposed a new PID optimizer, which was designed for accelerating the learning process of deep networks. Its motivation was based on the fact that the error in PID control and the gradient in deep learning optimization shared the same spirit. In contrast, for our proposed method, the error in PID control corresponded to the noise in the noisy image. We utilized this property and devised a novel denoiser for noise removal.

Second, the differences in model design lead to contrasting network capacities. The method of [46] followed a deep neural network optimization-based framework. Network parameters were updated via the present, past, and future information of the gradient. On the contrary, we are dedicated to building a dynamic control system, where the PID controller was deployed to guide the attention learning on a feedback network. Such a mechanism allowed us to enhance the discriminative ability of feature learning.

Third, compared with [46], our network has significant differences in terms of control action. In [46], the control action was based on the backward propagation of the updating weights. While in our work, the control action was implemented with residual feature learning. We emphasize that the proposed PID-AN is not only an attention-based neural network but a controller whose optimization provides an efficient and robust network performance.

To the best of our knowledge, to date, the study on PID controller for image denoising is still underexplored. PAN-Net is the first model along this research line. As we will see later, the incorporation of a PID controller and attention neural network has beneficial effects on achieving a higher efficiency and significantly better denoising performance.

III. Approach

In this section, we begin by introducing our network architecture and loss function. Then, we describe in detail the share-source residual module (SSRM) and PID-AN. The comprehensive implementation of the PID controller guided attention learning is introduced subsequently.

A. Network Architecture

The general architecture of our PAN-Net is shown in Fig. 2. It contains three parts: 1) a shallow feature extraction module;

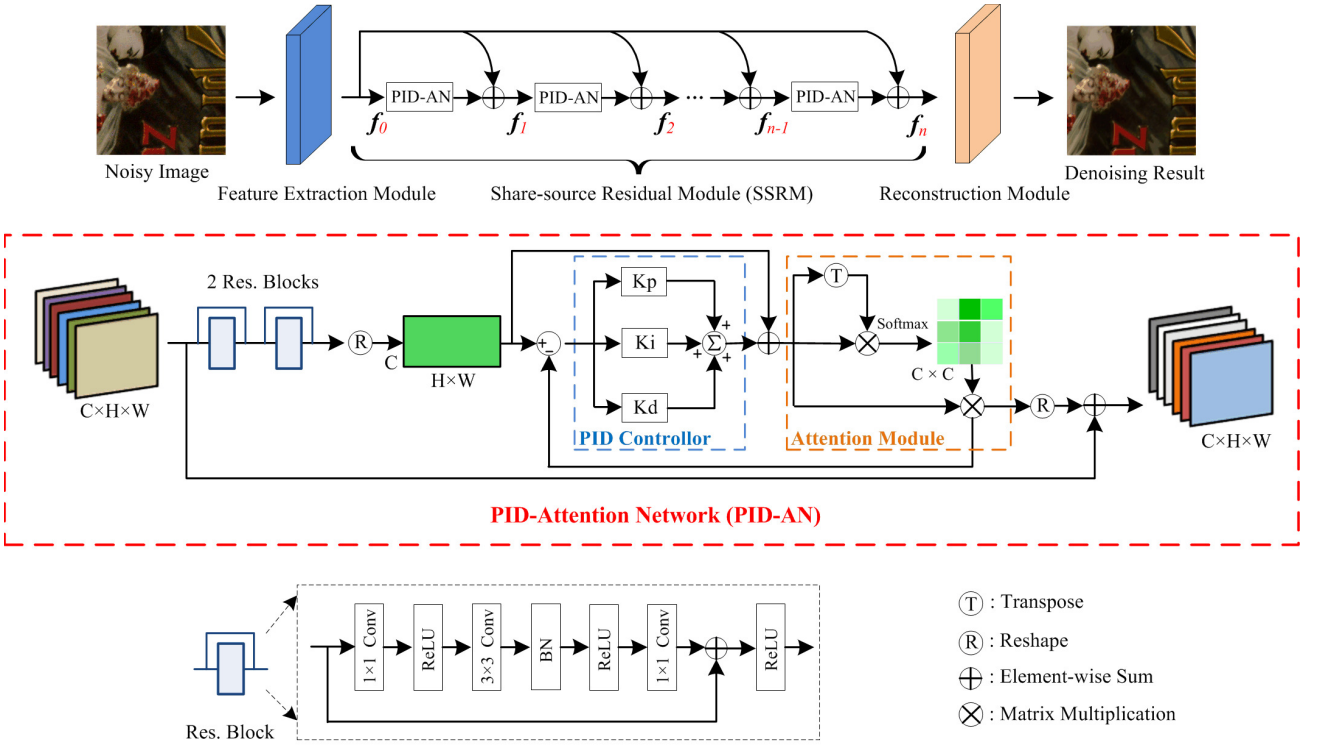


Fig. 2. Architecture of the proposed PAN-Net.

2) SSRM; and 3) reconstruction module. Given a noisy image $\boldsymbol{I}_N$, the shallow feature extraction module is first adopted to obtain the initial features $\boldsymbol{f}_0$

$$\boldsymbol{f}_0 = \boldsymbol{F}_{\text{SF}}(\boldsymbol{I}_N) \tag{2}$$

where $\boldsymbol{F}_{\text{SF}}$ denotes the convolution operation. Our shallow feature extraction module is simply composed of one convolutional layer with BN [44] and rectified linear units (ReLUs). Then, SSRM takes $\boldsymbol{f}_0$ as input and produces a deep learning feature $\boldsymbol{f}_n$

$$\boldsymbol{f}_n = \boldsymbol{F}_{\text{SSRM}}(\boldsymbol{f}_0) \tag{3}$$

where $\boldsymbol{F}_{\text{SSRM}}$ stands for the function of SSRM. As shown in Fig. 2, our SSRM consists of several PID-ANs, which are stacked together with SSCs. Such a design allows abundant contextual information from $\boldsymbol{f}_0$ to be passed on to each PID-AN, thus helping to strengthen feature propagation during training. Afterward, we leverage the reconstruction module, which contains one convolutional layer to transfer the feature maps of $\boldsymbol{f}_n$ to the domain of noise-free image $\boldsymbol{I}_C$

$$\boldsymbol{I}_C = \boldsymbol{F}_{\text{RE}}(\boldsymbol{f}_n) = \boldsymbol{F}_{\text{PAN-Net}}(\boldsymbol{I}_N) \tag{4}$$

where $\boldsymbol{F}_{\text{RE}}$ and $\boldsymbol{F}_{\text{PAN-Net}}$ represent the reconstruction part and proposed PAN-Net, respectively.

B. Objective Function

In training the proposed PAN-Net, our goal is to make the denoising results similar to the corresponding ground truth clean images. To this end, we employ $L1$ norm regularization to optimize the network parameters. Given a training set with K pairs of images $\{\boldsymbol{I}_N^i, \boldsymbol{I}_G^i\}_{i=1}^K$, where $\boldsymbol{I}_N^i$ is the ith noisy image and $\boldsymbol{I}_G^i$ is the ground truth counterpart, the implementation of the optimization process takes the form

$$\mathcal{L}_1 = \frac{1}{K}\sum_{i=1}^{K}\left\|\boldsymbol{F}_{\text{PAN-Net}}\left(\boldsymbol{I}_N^i\right) - \boldsymbol{I}_G^i\right\|_1. \tag{5}$$

However, directly minimizing the $L1$ distance tends to deliver smooth and blurry edges and textures. This stems from the fact that this simple loss is based on a pixel-by-pixel operation and encouraged to predict the mean of the possible solutions to avoid heavy penalties. We remedy this limitation by adding a perceptual loss, which aims at enforcing a small distance between the features of the network output and the corresponding ground truth image. Specifically, we adopt the last two layers of the VGG-19 network [47] to calculate our perceptual loss

$$\mathcal{L}_{\text{per}} = \frac{1}{2K}\sum_{j=1}^{2}\sum_{i=1}^{K}\left\|\boldsymbol{F}_{vgg19}\left(\boldsymbol{I}_C^i\right)^j - \boldsymbol{F}_{vgg19}(\boldsymbol{I}_G^i)^j\right\| \tag{6}$$

where j denotes the last jth convolutional layer. Since the VGG-19 network is well pretrained on ImageNet [48], it can capture meaningful image features and maintain perceptual similarity for fine details preservation.

Our final objective function is a weighted sum of $L1$ loss and perceptual loss

$$\mathcal{L} = \lambda_1\mathcal{L}_1 + \lambda_2\mathcal{L}_{\text{per}} \tag{7}$$

where λ_1 and λ_2 are tradeoff parameters to balance the effects of these two terms.

C. Share-Source Residual Module

In this section, we introduce the SSRM whose objective is to capture the full clean image features in a robust and efficient manner. As shown in Fig. 2, our SSRM contains several PID-ANs, each of which consists of two residual blocks, followed by a PID controller and an attention module (AM) to obtain better feature representations. Furthermore, we stack repeated PID-ANs with SSCs to ease the flow of information, providing a flexible way to feature residual learning and allowing the stable training of our network model.

The key component of SSRM is the PID-AN, which inherits advantages of the PID controller and attention learning mechanism to remove the noise characteristics with high efficiency. To improve the network performance, many researches are dedicated to increasing the network depth and filter size. However, most of these very deep CNNs, in general, can hardly get rid of the complex optimization procedure. In addition, as to the growth of depth, the network performance is liable to saturation even degeneration owing to the vanishing or exploding gradients problem. That is to say, simply stacking repeated PID-ANs may be inflexible enough to efficiently handle practical image denoising tasks. Motivated by the success of [49], we employ SSCs in the cascaded PID-ANs to form a residual-based deeper network such that the tradeoff between network depth and image denoising can be well guaranteed. In our SSRM, the kth PID-AN can be represented as follows:

$$\boldsymbol{f}_k = \boldsymbol{F}_{\text{PID-AN}}(\boldsymbol{f}_{k-1}) + \boldsymbol{f}_0 \tag{8}$$

where $\boldsymbol{F}_{\text{PID-AN}}$ denotes the function of the PID-AN. $\boldsymbol{f}_{k-1}$ and $\boldsymbol{f}_k$ are the input and output of the kth PID-AN, respectively.

In Section III-D, we will elaborate on the implementation process of our major subnetwork: PID-AN.

D. PID-Attention Network

The proposed PID-AN plays a critical role in robustly learning and exploiting discriminant feature representations. Its implementation process follows a four-stage framework. To begin with, two residual blocks are deployed to extract the image features, which are then fed into the AM to generate an attention feature map. Next, we adopt a combination of proportional, integral, and derivative operations to compute the error (i.e., difference) between the input and output of the AM. Third, we generate a fused feature by concatenating the initial input of the AM and the output of the PID controller, and finally, the fused output is passed through the AM. We continue the above-mentioned second step to further facilitate attention learning and boost discriminant feature representations in the feedback control system. An overview of the PID-AN can be seen in the second row of Fig. 2.

1) Attention Module: The AM is proposed to learn the semantic long-range dependencies in channel dimension and enhance the feature representation capability. Specifically, let $\boldsymbol{f}_k$ represent the input of the kth PID-AN. We first learn convolutional features by feeding $\boldsymbol{f}_k$ into two residual blocks. The implemental procedure can be formulated as

$$\boldsymbol{\Upsilon} = \boldsymbol{F}_{\text{Res}}(\boldsymbol{F}_{\text{Res}}(\boldsymbol{f}_k)) \tag{9}$$

where $\boldsymbol{\Upsilon}$ denotes the output features of the last residual block. $\boldsymbol{F}_{\text{Res}}$ is the function of residual block. As shown in the third row of Fig. 2, each residual block is a shallow CNN containing three convolutional layers with ReLU activations. Meanwhile, we utilize batch normalization in the second layer for easy optimization [44].

Assume that $\boldsymbol{\Upsilon} = [\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_c]$ is a $C \times H \times W$ dimensional feature tensor, where C is the number of channels of size $H \times W$. We reshape $\boldsymbol{\Upsilon}$ to a feature matrix $\boldsymbol{X} \in \mathbb{R}^{C \times N}$, aiming at decomposing the feature dimension. Note that in $\boldsymbol{X}$, $N = H \times W$ denotes the number of pixels. We then multiply $\boldsymbol{X}$ and the transpose of $\boldsymbol{X}$, and adopt a sigmoid activation to produce the interchannel relationship $\boldsymbol{A} \in \mathbb{R}^{C \times C}$:

$$a_{j,i} = \frac{\exp(x_i \cdot x_j)}{\sum_{i=1}^{C} \exp(x_i \cdot x_j)} \tag{10}$$

where $a_{j,i}$ indicates the degree to which the ith channel's impact on the jth channel. Meanwhile, the higher value implies the greater interdependencies of features between them. We continue to make a matrix multiplication operation between $\boldsymbol{A}$ and $\boldsymbol{X}$ to obtain the attention map $\boldsymbol{O}$, which can be explained as

$$o_j = \xi \sum_{i=1}^{C} (a_{j,i} x_i) \tag{11}$$

where ξ is a scale parameter and gradually learns to assign more weight from 0 [50]. Note that $\boldsymbol{O}$ is the same size as $\boldsymbol{X}$. As the training process goes on, the proposed AM can better emphasize the inherent feature interdependencies among all channel maps via accurate feature disparities. Moreover, benefiting from the long-range correlations of channelwise features, we can further improve the discriminative learning ability and strengthen feature representations.

2) PID-Controller Guided Attention Learning: As we described in Section III-D.1, the AM can be trained for a more powerful feature expression. However, due to the complexity of real noise, its feature extraction performance may degenerate. That is to say, employing the AM alone may be faced with trouble in thoroughly distinguishing the features between image contents and noise. Consequently, the denoising results would either remain noise or introduce artifacts. Even if a deeper and stronger attention network can be well constructed, it may suffer from the difficulty of optimization and lead to heavy computational loads. A practical image denoiser is expected to perform well in balancing the tradeoff between performance and efficiency. Owing to the robust, functional, and efficient capabilities of the PID controller, an alternative solution is proposed—integrating the inherent advantages of a PID controller and attention learning for this challenging problem. We believe that with a strong and improved learning mechanism, a plausible denoising performance can be achieved.

The ability of the PID controller is to eliminate the error between the actual output and the desired value via a dynamically closed-loop correcting system. Therefore, when performing image denoising, it is quite natural for us to take the "noise" as "error," and leverage the PID control system to

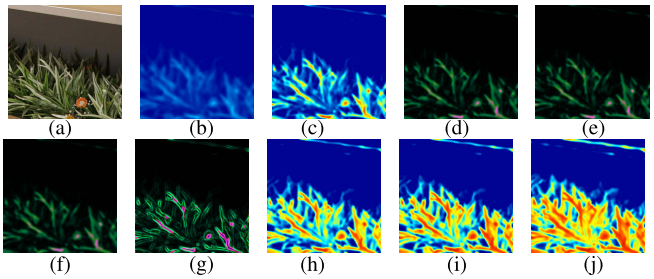


Fig. 3. Visualization of feature maps during the implementation of PID-AN. In PID-AN, we first apply two residual blocks to obtain the initial features (b). After three iterations, the residual feature (d) is fed into the PID controller to perform an integration of (e) proportional, (f) integral, and (g) derivative operations. One can see that with the increasing of time steps, the PID control can make beneficial guidance for the AM to focus on more and more discriminative contents. (a) Input. (b) Initial feature. (c) $t = 3$. (d) Residual feature. (e) P. (f) I. (g) D. (h) $t = 5$. (i) $t = 8$. (j) $t = 11$.

remove the noisy observation. Such a scheme is reasonable but difficult to implement because, with pairs of training data, it is easier to learn the image contents than the complex noise. According to the previous analysis, the AM can capture some noise-free image features, but not completely. During the network training, what we can easily get is the input and output features of the AM. A constructive idea comes to mind: taking the difference between the input and output of the AM as "system deviation," before making the PID controller utilize such deviation information to guide the AM for better discriminative learning. We call this PID-AN. Now, we will describe in detail how the PID-AN works.

According to the above-mentioned analysis, the first step for our PID-AN is to set an error between the desired setting point and actual output. Therefore, let the initial input of AM $\boldsymbol{X}$ be the desired value, and the output of AM $\boldsymbol{O}$ be the actual output. This means at iteration t, the error $e(t)$ takes the following form:

$$e(t) = \boldsymbol{X} - \boldsymbol{O}_t. \tag{12}$$

From an image point of view, $e(t)$ can be interpreted as the tth residual feature map, which mainly contains the features of some image contents $\boldsymbol{c}_t$ and noise $\boldsymbol{n}$. In other words, 12 can be rewritten as

$$e(t) = \boldsymbol{c}_t + \boldsymbol{n}. \tag{13}$$

Note that the purpose of our PID-AN is to exploit $e(t)$ to make the AM capable enough to fully characterize noisy-free image features. Therefore, our control goal is to make $e(t)$ gradually settle to $\boldsymbol{n}$ rather than 0, i.e., setting $\Delta e(t)$ to 0. Then, we perform proportional, integral, and derivative operations on $e(t)$, or say, the residual feature map, to calculate the correction term of $u(t)$

$$u(t) = K_p(\boldsymbol{c}_t + \boldsymbol{n}) + \frac{K_i}{3}\int_{t-2}^{t}(\boldsymbol{c}_t + \boldsymbol{n})\,dt + K_d\frac{d(\boldsymbol{c}_t + \boldsymbol{n})}{dt}. \tag{14}$$

Here, we emphasize that 14 does not simply just calculate the errors, but updates the residual feature map in a beneficial way. To better understand this implementation process, we illustrate some visualization images in Fig. 3(e)–(g). To be specific, one can observe that the proportional term $K_p(\boldsymbol{c}_t + \boldsymbol{n})$ can highlight the residual features. For the integral term, $(K_i/3)\int_{t-2}^{t}(\boldsymbol{c}_t + \boldsymbol{n})\,dt$ makes use of the averaging effect of three residual features to reduce the influence of noise [26]. The derivative term $K_d(d(\boldsymbol{c}_t + \boldsymbol{n})/dt)$ can be viewed as the image gradient that reflects the edge and texture information. The above-mentioned three terms are combined to obtain $u(t)$. Afterward, we concatenated $u(t)$ with $\boldsymbol{X}$ and feed them into the AM, whose output is utilized as the feedback signal of the PID controller. We continue the control procedures within the closed-loop system. As shown in Fig. 3, one can see that the AM can capture more and more prominent image characteristics with the increasing of each time step. When there is little improvement for AM [i.e., $\Delta e(t) = 0$], we stop the control process and append the output of AM to $\boldsymbol{f}_k$ with a global residual learning strategy to obtain the final output of PID-AN: $\boldsymbol{E} \in \mathbb{R}^{C\times H\times W}$

$$E_j = o_j + f_k^j. \tag{15}$$

Our PID-AN completely takes both efficiency and performance into account. It is known that mining enough meaningful features irrespective of noise is crucial to achieve a good denoising performance. Rather than constructing a deeper network to enhance the learning ability, our PID-AN introduces the PID controller to guide the AM for better feature learning. Such a control system can well boost feature representations effectively without a major investment in network parameters. Besides this, it also allows a positive acceleration in network training and testing. In addition to efficiency, our PID-AN is armed with a closed-loop negative feedback. That means the attention neural network works in a controlled manner. On the one hand, since the AM is consolidated by a PID controller, the complexity of training a blind denoiser can be reduced. On the other hand, it exhibits a robust response to complex real noise, providing a flexible and convenient way for applications in practice.

IV. Experiment

In this section, we first introduce the experimental settings, including the training and testing data sets, along with the implementation details of the parameters and network training in the proposed model. Then, we will give detailed analyses of the ablation study. Finally, we conduct three experiments to verify the effectiveness of our proposed PAN-Net: 1) the evaluation on AWGN removal; 2) the evaluation on real image denoising; and 3) the evaluation on network efficiency.

A. Experimental Settings

1) Data Sets: As for synthetic noisy image denoising, we first collected a large source of clean images from three data sets, namely, Waterloo Exploration database [51], DIV2K [52], and MIT-Adobe FiveK [53]. Then, we added AWGN in the [0, 75]-noise range to generate corresponding noisy images. As such, we can totally obtain 40 000 image pairs for network training. For fair comparisons, all the competitive methods were retrained on the same training set. To evaluate the denoising performance, we utilized five standard benchmark data sets: Set12 [14], BSD68 [54],

Algorithm 1 Learning Procedure of the Proposed PAN-Net

Input: Noisy image $\boldsymbol{I}_N$

```
for T = 1 to T_Train do
    Extract the initial features f_0;
    Obtain the attention feature map O_t;
    Set the residual feature map e(t) via Eq. 12;
    while Δe(t) ≠ 0 do
        The PID controller calculated the correction term of u(t) via Eq. 14;
        Update the residual feature map;
    end while
end for
```

Output: The final denoising image $\boldsymbol{I}_C$.

CBSD68 [55], Kodak24 [48], and McMaster [56], which are widely used for denoising test sets.

As for real noisy image denoising, we chose the Smartphone Image Denoising DATA set (SIDD) [57] to learn and evaluate our PAN-Net. Specifically, the noisy images in SIDD contained ten static scenes. Each scene was captured by five smartphone cameras under different lighting conditions and camera settings. It collected 30 000 real noisy images, with which 24 000 image pairs could be used for network training and 1280 images pairs were for validation purposes. Due to the varied network architectures, all of the compared methods were trained on the original data set by the authors. Meanwhile, we adopted the data sets from PolyU [58], Nam [26], and Darmstadt Noise Data set (DND) [59] to further evaluate the denoising performance.

2) Implementation Details: The learning procedure of the proposed network consists of three stages, as shown in Algorithm 1. First, we adopted the shallow feature extraction module to extract the initial features $\boldsymbol{f}_0$, followed with an AM to obtain the feature map $\boldsymbol{O}_t$. Second, we deployed the PID controller to guide the AM for better discriminative feature learning. The $\mathcal{L}_1$ and $\mathcal{L}_{\text{per}}$ were incorporated to fine-tune the network in the last step. Note that our PID-AN contains feedback control mechanism and backpropagation. After expanding the feedback network, the PID-AN can be represented as a feed-forward network with recurrent residual structure, with which the feedback control system can not only be compatible with the back propagation during network training but also contribute to faster and more stable convergence. To effectively train our PAN-Net, we used an Adam optimizer [60] with the hyperparameters $\beta_1 = 0.7$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We set the initial learning rate as 10^{-3} and reduced it to 5×10^{-4} with 40 epochs. When the training error showed little variation, we employed another learning rate of 10^{-6} with 50 epochs to further fine-tune the proposed model.

In our work, the convolutional kernel size in both feature extraction module and reconstruction module was 3×3. In the SSRM structure, we set four PID-ANs to extract meaningful image features and found that increasing the number of PID-ANs delivered a slightly better performance. However, this entailed heavy computational costs. The parameters of the PID controller includes the gain coefficients of K_p, K_i, and K_d. To combine the neighbor pixels during the implementation of control action, we set the three tunable parameters, each represented as a 3×1 matrix. In our case, these three gain coefficients were adjusted based upon the literature of [61]. To be specific, K_p was empirically set as $[1.24, 1.24, 1.24]^T$, while K_i and K_d were set as $[1.8, 1.8, 1.8]^T$ and $[1.12, 1.12, 1.12]^T$, respectively. As for the weights of the objective loss function, we reported the PSNR/SSIM results on the testing data sets by setting different values of $\{\lambda_1, \lambda_2\}$. We empirically found that when $\lambda_1 = 10^{-3}$ and $\lambda_2 = 3 \times 10^{-3}$, the proposed PAN-Net achieved the best quantitative results. We fixed all parameters throughout the experiments, which were implemented on PyTorch [62] environment with a CUDA of 10.0 and cuDNN of 7.5, running on a PC with an Intel Core i7-8565u CPU, RAM of 16 GB, and an NVIDIA Titan X GPU.

TABLE I

ABLATION EXPERIMENTS IN TERM OF THE LOSS FUNCTION, PID CONTROLLER, AND SSC ON THE NAM DATA SET

$\mathcal{L}_1$		✓	✓	✓	✓
$\mathcal{L}_{per}$	✓		✓	✓	✓
PID controller	✓	✓		✓	✓
SSC	✓	✓	✓		✓
PSNR (dB)	36.29	34.93	37.09	39.01	**39.72**
SSIM	0.873	0.844	0.899	0.958	**0.986**

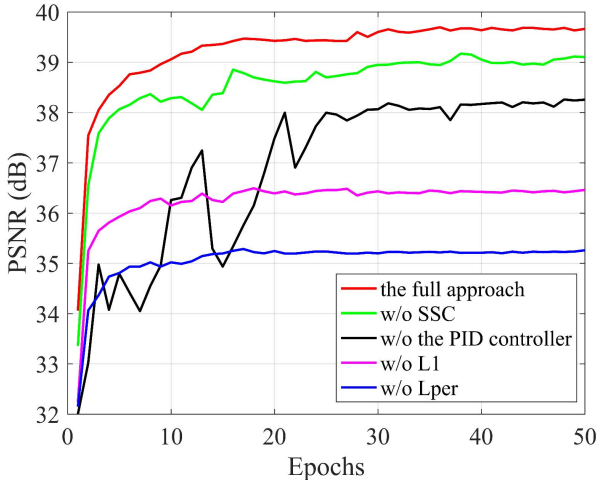


Fig. 4. Convergence analysis on the proposed PAN-Net and its variants.

B. Ablation Study

In this section, we performed ablation studies to verify the contributions brought by the loss function, PID controller, and SSC. Experimental results are shown in Table I and Figs. 4 and 5.

1) Influence of the Loss Function: As discussed in Section III, we used a combination of $\mathcal{L}_1$ and $\mathcal{L}_{\text{per}}$ to train our proposed model. Specifically, $\mathcal{L}_1$ attempts to remove the noisy observation and $\mathcal{L}_{\text{per}}$ has the objective of maintaining better image details. In Fig. 4, the lack of $\mathcal{L}_1$ and $\mathcal{L}_{\text{per}}$ made the performance drop during the training. In Fig. 5(b) and (c),

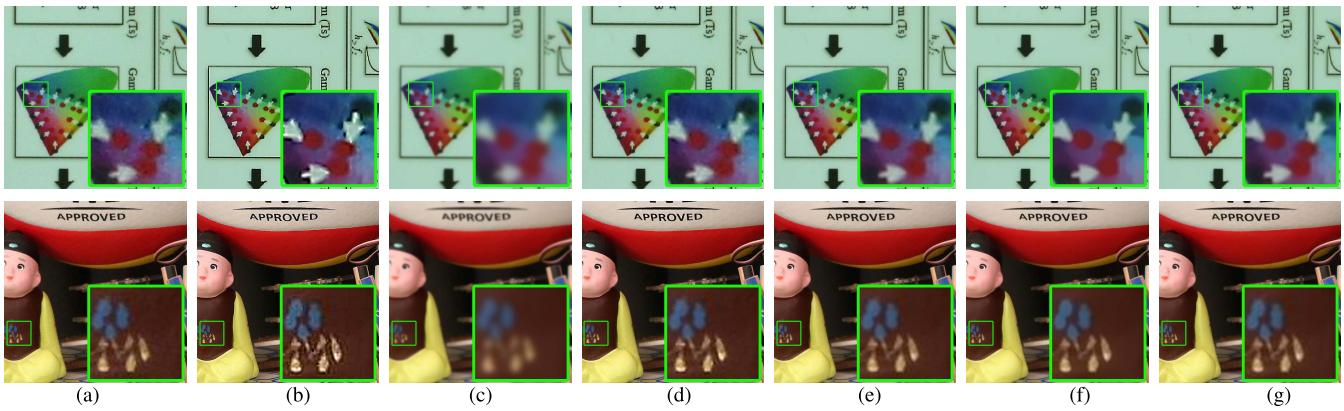


Fig. 5. Denoising results under different variants. Please zoom in for a better view. (a) Noisy input. (b) w/o $\mathcal{L}_1$. (c) w/o $\mathcal{L}_{\text{per}}$. (d) w/o PID. (e) w/o SSC. (f) Ours. (g) GT.

TABLE II
AVERAGE PSNR (dB) RESULTS OF DIFFERENT METHODS ON THE BSD68 DATA SET WITH NOISE LEVELS OF 15, 25, AND 50

Method	BM3D	WNNM	EPLL	MLP	CSF	TNRD	DnCNN	IRCNN	FFDNet	BRDNet	RIDNet	RDN	Ours
σ=15	31.07	31.37	31.26	-	31.32	31.49	31.81	31.69	31.71	31.85	31.88	31.92	**32.57**
σ=25	28.57	28.83	28.75	29.01	28.80	29.02	29.30	29.23	29.26	29.34	29.42	29.49	**30.12**
σ=50	25.62	25.87	25.68	26.05	-	25.99	26.28	26.24	26.33	26.40	26.45	26.44	**26.89**

the denoising results without $\mathcal{L}_{\text{per}}$ tended to smooth out the image details, while the denosing results without $\mathcal{L}_1$ were perceptually unsatisfying and likely to produce oversharp edges. In Table I, we can observe that with these two losses, the denoising performance improved to 39.72 dB on PSNR and 0.986 on SSIM.

2) Influence of the PID Controller: The PID controller plays a crucial role in guiding the AM for better feature representation learning. In Fig. 4, one can see that the PID controller can facilitate the network training with faster and initially more stable convergence. In addition, we found that without using the PID controller, the denoising results would maintain more noise, or give rise to artifacts, as shown in Fig. 5(d). Since training a CNN model for real noise denoising could inevitably increase the complexity of the network design and might lead to performance saturation, we decreased such drawbacks by introducing a PID controller. As shown in Table I, we can see that the PID controller significantly improved the PSNR/SSIM performance, viz., a 2.63-dB improvement on PSNR and a 0.087 improvement on SSIM.

3) Influence of the Share-Source Skip Connection: Our SSRM is made up of several stacked PIN-ANs through SSC. The SSC is able to ease the flow of low-frequency information across the PID-ANs, encouraging feature residual learning and facilitating the network training. To verify its effectiveness, we trained the proposed model with and without the SSC and reported the corresponding performance. In Fig. 4, we can observe that the SSC was desirable for stabilizing the convergence of training the model. It can be seen from Fig. 5(e) and (f) that with the help of SSC, our PAN-Net obtained plausible image details and the noise can be eliminated completely. Moreover, Table I shows the SSC made a positive influence to improve the performance by almost 0.71 dB on PSNR and 0.028 on SSIM.

C. Evaluation on AWGN Removal

In this subsection, we evaluated the flexibility of our PAN-Net in the task of AWGN removal, since AWGN is one of the widely studied noises. The representative and state-of-the-art comparison approaches include three model-based methods (i.e., BM3D [2], WNNM [8], and CBM3D [10]) and ten learning-based methods (i.e., EPLL [3], MLP [4], CSF [5], TNRD [63], DnCNN [14], IRCNN [15], FFDNet [32], Batch-renormalization Denoising Network (BRDNet) [33], RIDNet [35], and RDN [16]).

We first compared our model with the other denoising algorithms on two data sets, i.e., BSD68 [54] and Set12 [14]. These two data sets are widely used for gray-noisy image denoising. We set three noise levels ($\sigma = 15, 25, 50$) for evaluation and listed the average PSNR results in Tables II and III. Both tables exhibit that the proposed PAN-Net performs impressively against the model-based and discriminative learning-based methods. Specifically, in Table II, we achieved much better PSNR values than the competitive algorithms across all noise levels and outperformed the second best method (RDN) by an average of 0.58 dB. In Table III, one can see that our PAN-Net performed favorably than other approaches. For example, we achieved the best quantitative measure on 9 out of the 12 images when $\sigma = 15$, and 11 out of the 12 images when $\sigma = 25$ as well as $\sigma = 50$, showing that PAN-Net was more robust and effective in handling a wide range of noise levels.

Besides gray-noisy images, we further performed color-image denoising. In this task, we set five different noise levels from 15 to 75 and tested the competitive denoising algorithms

TABLE III
AVERAGE PSNR (dB) RESULTS FOR DIFFERENT METHODS ON SET12 WITH NOISE LEVELS OF 15, 25, AND 50

Images	C.man	Hourse	Peppers	Starfish	Monarch	Airplane	Parrot	Lena	Barbara	Boat	Man	Couple	Average
Noise level	σ=15												
BM3D	31.92	34.93	32.69	31.14	31.85	31.07	31.37	34.26	33.10	32.13	31.92	32.10	32.37
WNNM	32.17	35.13	32.99	31.82	32.71	31.39	31.62	34.27	33.60	32.27	32.11	32.17	32.70
EPLL	31.88	34.18	32.67	31.18	32.10	31.23	31.42	33.96	31.39	31.93	32.06	31.99	32.17
CSF	31.95	34.42	32.86	31.55	32.37	31.38	31.37	34.11	31.94	32.05	32.15	32.03	32.33
TNRD	32.23	34.59	33.04	31.78	32.62	31.50	31.68	34.29	32.15	32.14	32.26	32.15	32.53
DnCNN	32.68	35.01	33.35	32.25	33.12	31.75	31.87	34.65	32.68	32.44	32.49	32.51	32.93
IRCNN	32.59	34.95	33.37	32.08	32.88	31.77	31.87	34.56	32.46	32.39	32.44	32.47	32.82
FFDNet	32.47	35.09	33.15	32.09	32.83	31.65	31.84	34.65	32.59	32.41	32.43	32.50	32.81
BRDNet	32.82	35.28	33.47	32.24	**33.35**	31.86	32.00	34.75	32.92	32.55	32.51	32.62	33.03
RDN	32.82	35.31	33.45	33.36	33.27	31.88	**32.02**	**34.77**	32.94	32.54	32.50	32.77	33.08
Ours	**32.97**	**35.31**	**33.58**	**32.41**	33.33	**31.94**	31.98	34.73	**33.09**	**32.62**	**32.67**	**32.96**	**33.14**
Noise level	σ=25												
BM3D	29.45	32.85	30.16	28.56	29.25	28.42	28.93	32.07	30.71	29.90	29.61	29.71	29.97
WNNM	29.64	33.22	30.42	29.03	29.84	28.69	29.15	32.24	31.24	30.03	29.76	29.82	30.26
EPLL	29.30	32.11	30.20	28.51	29.41	28.65	28.96	31.77	28.63	29.77	29.68	29.53	29.75
MLP	29.63	32.60	30.35	28.85	29.62	28.86	29.26	32.29	29.58	29.05	29.88	29.77	30.08
CSF	29.51	32.44	30.37	28.81	29.64	28.75	28.96	31.82	29.05	29.81	29.75	29.56	29.86
TNRD	29.77	32.56	30.61	29.05	29.88	28.91	29.24	32.05	29.44	29.96	29.91	29.75	30.15
DnCNN	30.22	33.10	30.91	29.46	30.33	29.19	29.47	32.47	30.05	30.27	30.18	30.17	30.48
IRCNN	30.14	33.11	30.93	29.33	30.14	29.20	29.52	32.48	29.96	30.22	30.10	30.13	30.42
FFDNet	30.13	33.35	30.85	29.38	30.21	29.55	29.48	32.57	29.03	30.31	30.13	30.22	30.49
BRDNet	31.42	33.41	31.06	29.45	30.48	29.21	29.57	32.65	30.36	30.35	30.16	30.29	30.66
RDN	31.45	33.57	31.12	29.54	30.69	29.60	29.66	**32.68**	30.36	30.44	30.25	30.37	30.69
Ours	**31.56**	**33.68**	**31.19**	**29.63**	**30.79**	**29.64**	**29.70**	32.63	**30.61**	**30.57**	**30.38**	**30.49**	**30.90**
Noise level	σ=50												
BM3D	26.13	29.69	26.68	25.04	25.82	25.10	25.90	29.05	27.22	26.78	26.81	26.46	26.72
WNNM	26.45	30.33	26.95	25.44	26.32	25.42	26.14	29.25	27.79	26.97	26.94	26.64	27.05
EPLL	26.10	29.12	26.83	25.15	25.94	25.33	25.96	28.70	24.83	26.75	26.79	26.33	26.48
MLP	26.39	29.64	26.70	25.44	26.30	25.57	26.14	29.36	25.27	27.05	27.09	26.69	26.78
TNRD	26.63	29.51	27.10	25.44	26.35	25.64	26.18	28.93	25.71	26.97	27.01	26.53	26.83
DnCNN	27.08	30.09	27.33	25.76	26.81	25.92	26.51	29.43	26.27	27.24	27.26	26.93	27.22
IRCNN	26.91	29.99	27.35	25.61	26.65	25.91	26.58	29.45	26.26	27.22	27.22	26.90	27.16
FFDNet	27.06	30.47	27.46	25.79	26.91	25.94	26.60	29.72	26.52	27.35	27.30	27.11	27.35
BRDNet	27.47	30.55	27.67	25.79	27.00	25.96	26.69	29.77	26.85	27.38	27.27	27.19	27.46
RDN	27.56	30.60	27.77	25.83	27.06	26.08	26.77	29.85	26.96	27.46	**27.33**	27.25	27.51
Ours	**27.61**	**30.65**	**27.81**	**25.86**	**27.11**	**26.14**	**26.85**	**29.98**	**27.03**	**27.52**	27.29	**27.31**	**27.58**

on three data sets (i.e., CBSD68, Kodak24, and McMaster). These results are shown in Table IV. Overall, our method performed effectively and was able to obtain more notable image quality scores than the competing methods. Meanwhile, PAN-Net outperformed all the competitors when $\sigma \leq 50$ and was still superior with the increase of the noise level. Fig. 6 shows some visual comparisons of the different methods. Here, we selected a noisy image from CBSD68 and then set a large noise intensity $\sigma = 50$ for comparison. It can be seen that CBM3D oversmoothed much of the image details when removing noise. For the methods of TNRD, and DnCNN, they cannot recover clearly the details and blur the structures in some regions due to the large noise intensity. Compared with the state-of-the-art deep learning-based methods (i.e., FFDNet, BRDNet, RIDNet, and RDN), it is clear that our PAN-Net was able to reproduce more fine-scale details and sharper edges, achieving better perceptual visual quality.

D. Evaluation on Real Photographs Denoising

In this section, we proposed experiments on real photographs denoising to further show the practicality of our PAN-Net. As stated by [20]–[23], the noise in real photographs comes from multiple different sources and is much more complex than AWGN. Besides this, the noise can be sophisticated and signal dependent, making it difficult to be described by explicit distributions. Consequently, many prior works lack flexibility when coping with real-image noise. Therefore, evaluation of real photographs denoising

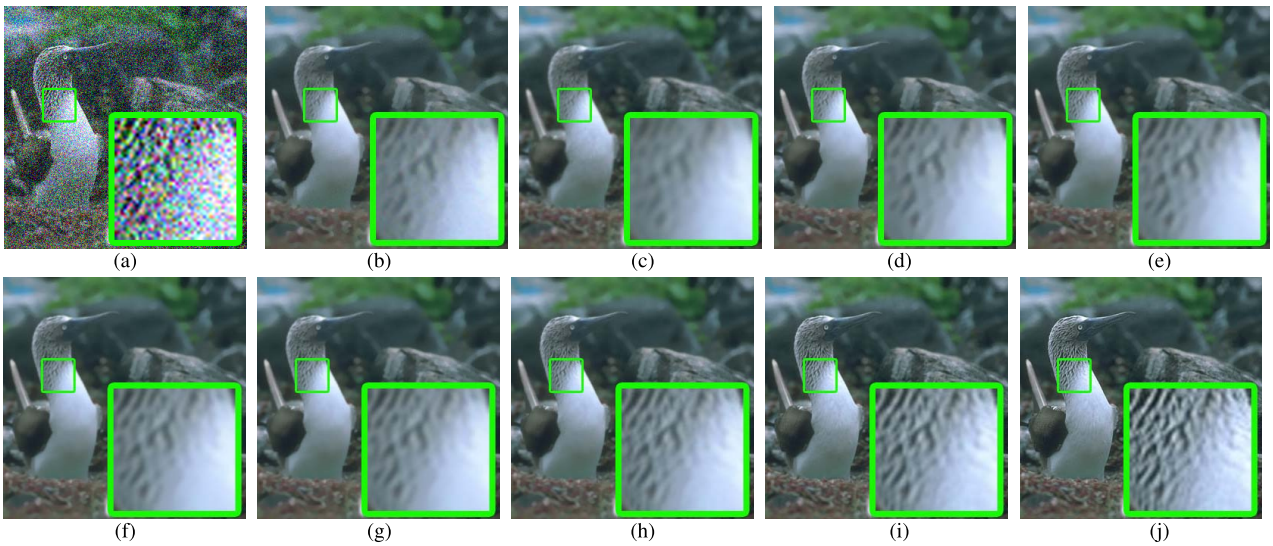


Fig. 6. Visual comparisons between PAN-Net and its competitors in the evaluation of color-noisy image denoising. The test image was cropped from CBSD68 with a large noise intensity of $\sigma = 50$. We reported the PSNR values for each denoising result. Please zoom in for a better view. (a) Noisy input: 14.96 dB. (b) CBM3D [10]: 26.33 dB. (c) TDNR [63]: 26.73 dB. (d) DnCNN [14]: 27.10 dB. (e) FFDNet [32]: 27.32 dB. (f) BRDNet [33]: 27.44 dB. (g) RIDNet [35]: 27.43 dB. (h) RDN [16]: 27.56 dB. (i) Ours: **27.74 dB**. (j) GT.

TABLE IV

AVERAGE PSNR (dB) RESULTS OF DIFFERENT METHODS ON THE CBSD68, KODAK24, AND MCMASTER DATA SETS WITH NOISE LEVELS OF 15, 25, 35, 50, AND 75

Dataset	Methods	σ=15	σ=25	σ=35	σ=50	σ=75
CBSD68	CBM3D	33.52	30.71	28.89	27.38	25.74
	FFDNet	33.90	31.26	29.64	27.97	26.25
	DnCNN	33.95	31.25	29.63	27.92	24.50
	IRCNN	33.90	31.22	29.55	27.85	-
	BRDNet	34.15	31.45	29.77	28.17	26.43
	RDN	34.19	31.53	29.80	28.22	26.55
	Ours	**34.45**	**31.56**	**29.84**	**28.27**	**26.58**
Kodak24	CBM3D	34.28	31.68	29.90	28.46	26.82
	FFDNet	34.69	32.16	30.61	29.00	27.27
	DnCNN	34.55	32.07	30.52	28.86	25.03
	IRCNN	34.60	32.06	30.48	28.81	-
	BRDNet	34.89	32.44	30.83	29.22	27.40
	RDN	35.14	32.65	30.95	29.31	**27.53**
	Ours	**35.41**	**32.89**	**31.06**	**29.37**	27.49
MCMaster	CBM3D	34.06	31.66	29.92	28.51	26.79
	FFDNet	34.71	32.37	30.84	29.20	27.33
	DnCNN	33.46	31.55	30.19	28.61	25.11
	IRCNN	34.62	32.20	30.65	28.91	-
	BRDNet	35.10	32.77	31.15	29.52	27.72
	RDN	35.08	32.78	31.26	29.60	27.76
	Ours	**35.61**	**33.08**	**31.49**	**29.67**	**27.79**

is essential and reveals significant importance to real-world applications.

We designed comparison experiments against the following state-of-the-art algorithms: DnCNN-B [14], RDN [16], FFDNet+ [32], Noise Clinic (NC) [25], Neat Image (NI) [64], Trilateral Weighted Sparse Coding (TWSC) [30], MCWNNM [28], CBDNet [36], BRDNet [33], RIDNet [35], and VDN [37]. The testing images contained four real photographs benchmark data sets, including PolyU [58], Nam [26], SIDD [57], and DND [59].

TABLE V

QUANTITATIVE RESULTS (IN PSNR (dB)/SSIM) FOR POLYU AND NAM

Method	PolyU	Nam
DnCNN-B	34.68 / 0.874	34.95 / 0.885
NI	35.91 / 0.921	36.61 / 0.926
NC	36.84 / 0.936	37.69 / 0.952
MCWNNM	37.72 / 0.945	37.84 / 0.956
RDN	37.94 / 0.946	38.16 / 0.956
FFDNet+	38.17 / 0.951	38.81 / 0.957
TWSC	38.68 / 0.958	38.96 / 0.962
BRDNet	38.32 / 0.954	38.88 / 0.960
CBDNet	38.74 / 0.961	39.08 / 0.969
RIDNet	38.86 / 0.962	39.20 / 0.973
VDN	39.04 / 0.965	39.68 / 0.976
Ours	**40.68 / 0.980**	**40.71 / 0.988**

1) Results on PolyU and Nam: PolyU contains noisy images of 40 different scenes, while Nam contains images of 11 static scenes. Each scene was shot 500 times with the same camera. By averaging these 500 shots, the ground-truth noise-free image can be obtained. Since the "ground truth" images of PolyU and Nam have been publicly released, we performed both quantitative and qualitative evaluations on these two data sets.

Table V exhibits the quantitative evaluation results of the compared methods. It can be seen that our PAN-Net obtained a much higher PSNR and SSIM than the other denoising methods. For example, in PolyU, the denoising results of our method had a 1.64-, 1.82-, and 1.94-dB improvement over VDN, RIDNet, and CBDNet, respectively. In Nam, we can see again that the proposed model achieved comparable image

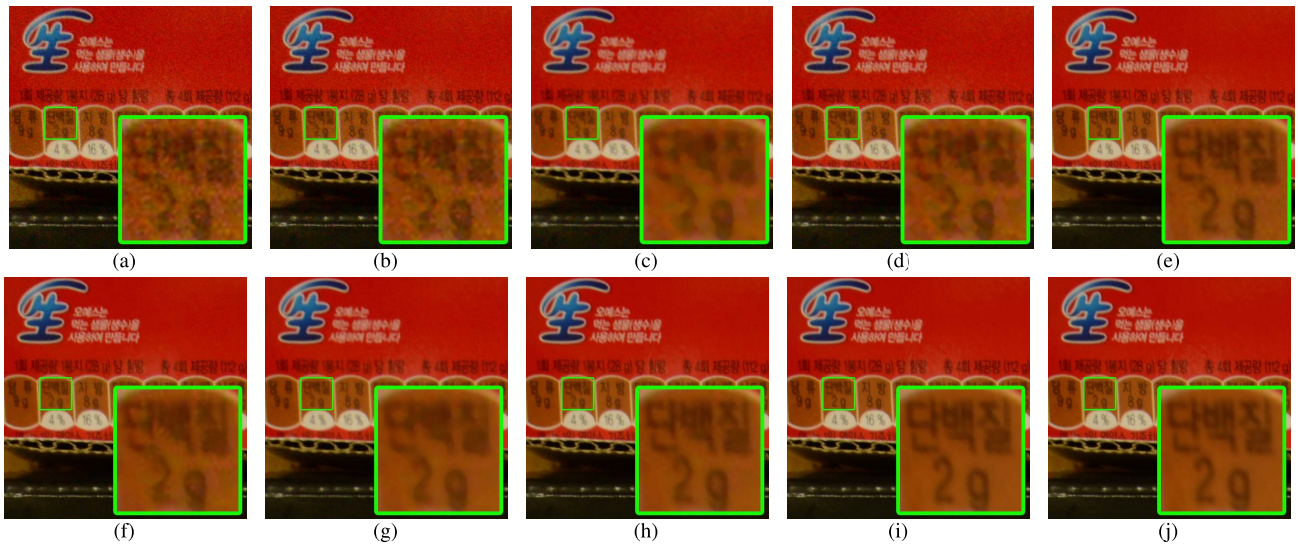


Fig. 7. Visual comparisons between PAN-Net and its competitors on the Nam testing data sets. We reported the PSNR values for each denoising result. Please zoom in for a better view. (a) Noisy input: 26.25 dB. (b) DnCNN-B [14]: 28.15 dB. (c) RDN [16]: 29.07 dB. (d) FFDNet+ [32]: 30.39 dB. (e) TWSC [30]: 34.91 dB. (f) CBDNet [36]: 33.94 dB. (g) RIDNet [35]: 35.88 dB. (h) VDN [37]: 36.17 dB. (i) Ours: **38.02 dB**. (j) GT.

TABLE VI
QUANTITATIVE RESULTS (IN PSNR (dB)/SSIM) FOR SIDD AND DND

Method	SIDD	DND
DnCNN-B	23.66 / 0.583	32.43 / 0.790
FFDNet+	- / -	37.61 / 0.942
CBDNet	33.28 / 0.868	38.06 / 0.942
RIDNet	38.71 / 0.914	39.26 / **0.953**
VDN	39.23 / 0.955	39.38 / 0.952
Ours	**39.76** / **0.960**	**39.44** / 0.952

quality scores, where the improvements of PAN-Net over VDN, RIDNet, and CBDNet were 1.03, 1.51, and 1.63 dB, respectively. Fig. 7 is used to show visual comparisons. It can be observed that DnCNN-B, RDN, and FFDNet+ failed to remove the complete noise, while TWSC, CBDNet, RIDNet, and VDN can better eliminate the complex noise, but tended to generate artifacts in some regions. In comparison, our PAN-Net was able to eliminate the complex noise without introducing artifacts, delivering better visual quality.

2) Results on SIDD and DND: DND [59] was obtained from four consumer cameras and included 50 pairs of images. However, the "ground truth" noise-free images of DND are still not available online. For SIDD and DND, one can obtain the average PSNR and SSIM values by submitting the denoised results to the official benchmark website. The quantitative comparisons between PAN-Net and the competitors are presented in Table VI. Note that we only reported the methods whose objective is for real photograph denoising and the results are accessible on the website.

It is evident that PAN-Net performs favorably among the competing methods in most cases. In particular, the proposed method achieved the best result on the SIDD benchmark and produced a PSNR gain of 0.53 dB on VDN and 1.05 dB on RIDNet. Besides this, it is easy to see that we surpassed CBDNet and DnCNN-B by a significant margin. As for the DND benchmark, we performed slightly better than RIDNet and VDN and outperformed CBDNet and DnCNN-B by about 1.3 dB at least.

Figs. 8 and 9 show the denoised images of PAN-Net and those of the other competing methods. As one can see in Fig. 8, CBDNet failed to remove the complex real noise and generated noticeable artifacts and blotchy textures; on the other hand, RIDNet performed denoising by sacrificing fine-scale image details. Compared with the recent best method VDN, we can find that our method obtained better visual quality, where the proposed PAN-Net did a good job in eliminating the complex noise while preserving the textural and structural information without generating artifacts. In Fig. 9, again we can see that our PAN-Net was sufficient to describe the full image information and achieved the best overall visual quality among the competing algorithms.

E. Computational Evaluation

Computational evaluation is another important metric in evaluating the denoising performance. We, therefore, compared the model parameters, floating-point operations (FLOPs), and running time with seven popular denoising methods. These three metrics were computed on the PolyU and Nam testing images with 512×512 pixel. For a fair comparison, all experiments were implemented on the same machine with an NVIDIA Titan X GPU. For each testing image, we performed the evaluation ten times and calculated the average testing value. Results are shown in Table VII.

From Table VII, one can see that FFDNet+ had the fewest number of parameters and computational expense. Note that DnCNN-B had more FLOPs but used much fewer parameters than CBDNet. This is reasonable since that DnCNN-B mainly learned the feature maps on the full-resolution domain, while CBDNet operated on a downscaled space. Owing to parallel residual blocks, RDN required a large number of network

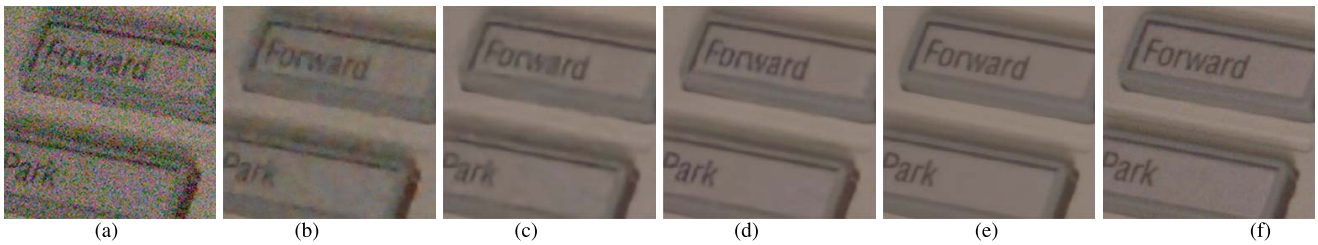


Fig. 8. Visual comparisons between PAN-Net and its competitors on the SIDD benchmark. We reported the PSNR values for each denoising result. Please zoom in for a better view. (a) Input: 18.25 dB. (b) CBDNet [36]: 28.84 dB. (c) RIDNet [35]: 35.57 dB. (d) VDN [37]: 36.39 dB. (e) Ours: **37.28 dB**. (f) GT.

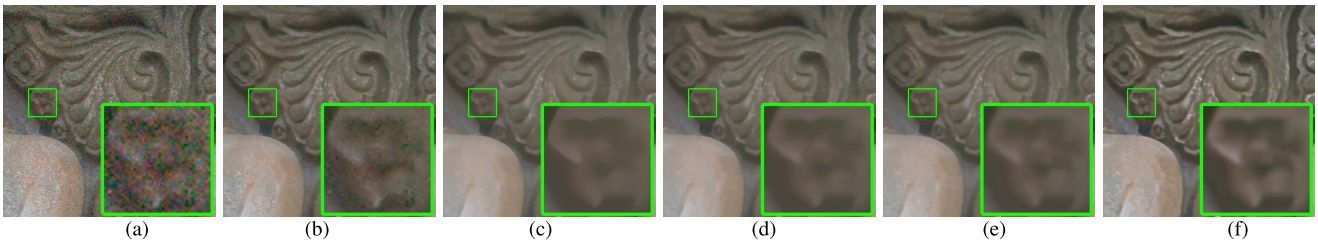


Fig. 9. Visual comparisons between PAN-Net and its competitors. The test image was cropped from DND benchmark. Note that the ground-truth noise-free image of this noisy input has not been released yet. Thus, we adopted the visual comparison for evaluation. Please zoom in for a better view. (a) Input. (b) FFDNet+ [32]. (c) CBDNet [36]. (d) RIDNet [35]. (e) VDN [37]. (f) Ours.

TABLE VII

COMPLEXITY EVALUATIONS OF DIFFERENT DENOISING METHODS. NOTE THAT THESE METRICS WERE COMPUTED ON THE POLYU AND NAM TESTING IMAGES WITH SIZE OF 512×512

Methods	DnCNN-B	FFDNet+	RDN	BRDNet	CBDNet	RIDNet	VDN	Ours(w/o PID)	Ours
Params (10^6)	0.56	**0.49**	21.97	1.11	4.34	1.50	4.36	1.18	1.18
Flops (10^9)	146.94	**107.29**	717.82	283.87	144.36	392.53	158.49	154.08	126.53
Speed (s)	0.064	**0.028**	1.56	0.211	0.429	0.214	0.497	0.226	0.142
PSNR (dB)	34.63	37.55	37.91	38.37	38.78	38.74	39.16	38.75	**40.33**
SSIM	0.865	0.924	0.937	0.943	0.946	0.953	0.961	0.940	**0.979**

parameters and FLOPs, resulting in heavy computational loads. Compared with the recent state-of-the-art methods (i.e., CBDNet, RIDNet, VDN), our PAN-Net was cost effective and achieved the optimal computational efficiency with significantly fewer parameters and FLOPs.

In addition to model parameters and FLOPs, we further provided the execution time of processing a 512×512 image for the various approaches. One can see that FFDNet+ obtained the best efficiency and was about two times faster than the second-best method (DnDNN-B). However, these two methods were not flexible enough to handle the complex noise. Compared with the other competitive methods, the proposed PAN-Net spent around 0.142 s in processing a color image. To summarize, taking both denoising performance and efficiency into account, our PAN-Net can be considered effective and efficient for real photograph denoising tasks, exhibiting distinct advantages over other popular and state-of-the-art methods.

Finally, we studied the impact of a PID controller on the network complexity. From the penultimate column of Table VII, we have the following observations. First, our PAN-Net in the absence of the PID controller nearly retained its network parameters similar to before. Here, we can observe that the FLOPs increased by about 27.55 G. Second, we find that armed with the PID controller, the proposed PAN-Net produced a more notable PSNR gain of about 1.58 dB. These results supported the advantage of PID control technology. Since the real noise is sophisticated and spatially variant, using the attention mechanism only to achieve higher denoising performance does not hold for a resource-efficient manner. Analysis of the data shows that the PID controller was effective to noticeably boost the network computation efficiency and robustness.

V. CONCLUSION

In this article, we made an attempt for real photograph denoising by integrating the advantages of both the PID controller and neural attention network. In particular, we developed stacked PID-ANs that can provide a more comprehensive understanding of feature representations. An integrated learning scheme is built up to enhance the robustness and flexibility of the PID-ANs. In addition, the incorporation of both the residual structure and SSC to the cascaded PID-ANs brought the feature residual learning strategy for our method, which, as a result, eased the network training and improved the denoising performance. To obtain a more pleasing visual

quality, we combined $\mathcal{L}_1$ and $\mathcal{L}_{\text{per}}$ to train the proposed model. Our PAN-Net is efficient and can better cope with the sophisticated noise while preserving fine-scale texture and structural information. The comparisons on both synthetic and real noisy images demonstrate the superiority of our PAN-Net over state-of-the-art denoising algorithms.

REFERENCES

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

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

[3] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 479–486.

[4] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2392–2399.

[5] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2774–2781.

[6] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 60–65.

[7] M. Lebrun, A. Buades, and J. M. Morel, "A nonlocal Bayesian image denoising algorithm," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1665–1688, Jan. 2013.

[8] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2862–2869.

[9] P. Zhong and R. Wang, "Jointly learning the hybrid CRF and MLR model for simultaneous denoising and classification of hyperspectral imagery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 7, pp. 1319–1334, Jul. 2014.

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2007, pp. 313–316.

[11] K. W. Jorgensen and L. K. Hansen, "Model selection for Gaussian kernel PCA denoising," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 1, pp. 163–168, Jan. 2012.

[12] J.-K. Im, D. W. Apley, and G. C. Runger, "Tangent hyperplane kernel principal component analysis for denoising," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 4, pp. 644–656, Apr. 2012.

[13] H. Duan and X. Wang, "Echo state networks with orthogonal pigeon-inspired optimization for image restoration," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2413–2425, Nov. 2016.

[14] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[15] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2808–2817.

[16] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jan. 21, 2020, doi: 10.1109/TPAMI.2020.2968521.

[17] S. Karatsiolis and C. N. Schizas, "Conditional generative denoising autoencoder," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 4117–4129, Oct. 2019.

[18] P. Wei, Y. Ke, and C. K. Goh, "Feature analysis of marginalized stacked denoising autoenconder for unsupervised domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1321–1334, May 2019.

[19] A. Creswell and A. A. Bharath, "Denoising adversarial autoencoders," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 968–984, Apr. 2019.

[20] G. E. Healey and R. Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 3, pp. 267–276, Mar. 1994.

[21] Y. Tsin, V. Ramesh, and T. Kanade, "Statistical calibration of CDD imaging process," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 480–487.

[22] S. J. Kim, H. T. Lin, Z. Lu, S. Süsstrunk, S. Lin, and M. S. Brown, "A new in-camera imaging model for color computer vision and its application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2289–2302, Dec. 2012.

[23] H. C. Karaimer and M. S. Brown, "A software platform for manipulating the camera imaging pipeline," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 429–444.

[24] T. Rabie, "Robust estimation approach for blind denoising," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1755–1765, Nov. 2005.

[25] M. Lebrun, M. Colom, and J.-M. Morel, "Multiscale image blind denoising," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3149–3161, Oct. 2015.

[26] S. Nam, Y. Hwang, Y. Matsushita, and S. J. Kim, "A holistic approach to cross-channel image noise modeling and its application to image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1683–1691.

[27] F. Zhu, G. Chen, J. Hao, and P.-A. Heng, "Blind image denoising via dependent Dirichlet process tree," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1518–1531, Aug. 2017.

[28] J. Xu, L. Zhang, D. Zhang, and X. Feng, "Multi-channel weighted nuclear norm minimization for real color image denoising," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1105–1113.

[29] J. Xu, L. Zhang, and D. Zhang, "External prior guided internal prior learning for real-world noisy image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2996–3010, Jun. 2018.

[30] J. Xu, L. Zhang, and D. Zhang, "A trilateral weighted sparse coding scheme for real-world image denoising," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 21–38.

[31] A. Majumdar, "Blind denoising autoencoder," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 312–317, Jan. 2019.

[32] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[33] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep CNN with batch renormalization," *Neural Netw.*, vol. 121, pp. 461–473, Jan. 2020.

[34] C. Tian, Y. Xu, Z. Li, W. Zuo, L. Fei, and H. Liu, "Attention-guided CNN for image denoising," *Neural Netw.*, vol. 124, pp. 117–129, Apr. 2020.

[35] S. Anwar and N. Barnes, "Real image denoising with feature attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3155–3164.

[36] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1712–1722.

[37] Z. Yus, H. Yong, Q. Zhao, D. Meng, S. Ozair, and L. Zhang, "Variational denoising network: Toward blind noise modeling and removal," in *Proc. IEEE Conf. Neural Inf. Process. Syst. (NIPS)*, 2019, pp. 1690–1701.

[38] R. Ma, H. Hu, S. Xing, and Z. Li, "Efficient and fast real-world noisy image denoising by combining pyramid neural network and two-pathway unscented Kalman filter," *IEEE Trans. Image Process.*, vol. 29, pp. 3927–3940, 2020.

[39] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3155–3164.

[40] G. Fiengo, D. G. Lui, A. Petrillo, S. Santini, and M. Tufo, "Distributed robust PID control for leader tracking in uncertain connected ground vehicles with V2V communication delay," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 3, pp. 1153–1165, Jun. 2019.

[41] S. Gao, Y. Hou, H. Dong, Y. Yue, and S. Li, "Global nested PID control of strict-feedback nonlinear systems with prescribed output and virtual tracking performance," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 2, pp. 325–329, Feb. 2020.

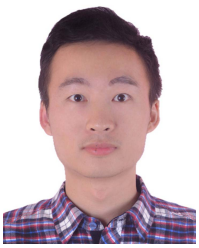
[42] Y. Pan, X. Li, and H. Yu, "Efficient PID tracking control of robotic manipulators driven by compliant actuators," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 2, pp. 915–922, Mar. 2019.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[44] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2015, pp. 448–456.

[45] A. Singh *et al.*, "A digital low-dropout regulator with autotuned PID compensator and dynamic gain control for improved transient performance under process variations and aging," *IEEE Trans. Power Electron.*, vol. 35, no. 3, pp. 3242–3253, Mar. 2020.

[46] W. An, H. Wang, Q. Sun, J. Xu, Q. Dai, and L. Zhang, "A PID controller approach for stochastic optimization of deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8522–8531.
[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[48] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[49] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11057–11066.
[50] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 97, Jun. 2019, pp. 7354–7363.
[51] K. Ma *et al.*, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 1004–1016, Feb. 2017.
[52] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 126–135.
[53] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, "Learning photographic global tonal adjustment with a database of input / output image pairs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 97–104.
[54] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 860–867.
[55] S. Roth and M. J. Black, "Fields of experts," *Int. J. Comput. Vis.*, vol. 82, no. 2, pp. 205–229, Apr. 2009.
[56] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *J. Electron. Imag.*, vol. 20, no. 2, pp. 1–15, 2011.
[57] A. Abdelhamed, S. Lin, and M. S. Brown, "A high-quality denoising dataset for smartphone cameras," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1692–1700.
[58] J. Xu, H. Li, Z. Liang, D. Zhang, and L. Zhang, "Real-world noisy image denoising: A new benchmark," 2018, *arXiv:1804.02603*. [Online]. Available: http://arxiv.org/abs/1804.02603
[59] T. Plotz and S. Roth, "Benchmarking denoising algorithms with real photographs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2750–2759.
[60] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[61] W. L. Torres, I. B. Q. Araujo, J. B. M. Filho, and A. G. Costa, "Mathematical modeling and PID controller parameter tuning in a didactic thermal plant," *IEEE Latin Amer. Trans.*, vol. 15, no. 7, pp. 1250–1256, 2017.
[62] A. Paszke *et al.*, "Automatic differentiation in Pytorch," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 1–4.
[63] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, Jun. 2017.
[64] Neatlab ABSoft. *Neat Image*. Accessed: Feb. 18, 2020. [Online]. Available: https://ni.neatvideo.com/home



Ruijun Ma received the B.S. degree from the Guangdong University of Technology, Guangzhou, China, in 2014, and the M.S. degree from Sun Yat-sen University, Guangzhou, in 2018. He is currently pursuing the Ph.D. degree with the Faculty of Science and Technology, University of Macau, Taipa, Macau. He is currently with Guangdong Polytechnic Normal University, Guangzhou. His current research interests include image denoising, meta-learning, and deep learning-based models.

Bob Zhang (Senior Member, IEEE) received the B.A. degree in computer science from York University, Toronto, ON, Canada, in 2006, the M.A.Sc. degree in information systems security from Concordia University, Montreal, QC, Canada, in 2007, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2011.

He was with the Center for Pattern Recognition and Machine Intelligence, University of Waterloo, and a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA. He is currently an Associate Professor with the Department of Computer and Information Science, University of Macau, Taipa, Macau. His research interests focus on biometrics, pattern recognition, and image processing.

Dr. Zhang is a Technical Committee Member of the IEEE Systems, Man, and Cybernetics Society. He is an Associate Editor of *IET Computer Vision*.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree from Hunan University, Changsha, China, in 1992, and the M.S. and Ph.D. degrees from Tufts University, Medford, MA, USA, in 2008 and 2010, respectively, all in electrical engineering.

He joined as an Assistant Professor with the Department of Computer and Information Science, University of Macau, Taipa, Macau, in 2011, where he is currently an Associate Professor and the Director of the Vision and Image Processing Laboratory. His research interests include image processing, computer vision, machine learning, and multimedia security.

Dr. Zhou is a fellow of the Society of Photo-Optical Instrumentation Engineers. He received the Third Price of the Macao Natural Science Award as a sole winner in 2020 and a co-recipient in 2014. He received the Best Editor Award in recognition of his editorial contribution to *Journal of Visual Communication and Image Representation* in 2020. Since 2015, he has been the leading Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He serves as an Associate Editor for the IEEE TRANSACTIONS ON NEUTRAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and four other journals. He was recognized as the highly cited researcher in Web of Science in 2020.

Zhengming Li received the M.S. degree from Southwest University, Chongqing, China, in 2007, and the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2017.

He is currently an Associate Professor with Guangdong Polytechnic Normal University, Guangzhou, China. His current research interests include pattern recognition and biometrics.

Fangyuan Lei received the B.S. and Ph.D. degrees from Northwestern Polytechnic University, Xi'an, China, in 1998 and 2004, respectively.

He is currently an Associate Professor with the School of Electronic and Information, Guangdong Polytechnic Normal University, Guangzhou, China. His research interests include graph convolutional network, image processing, and sensor network.