Learning Generic Diffusion Processes for Image Restoration

Peng Qiao[1]
pengqiao@nudt.edu.cn
Yong Dou[1]
yongdou@nudt.edu.cn
Yunjin Chen[2]
chenyunjinnudt@hotmail.com
Wensen Feng[3]
sanmumuren@126.com

[1] Science and Technology on Parallel and Distributed Laboratory
National University of Defense Technology
Changsha, China

[2] ULSee, Inc.
Hangzhou, China

[3] College of Computer Science & Software Engineering
Shenzhen University
Shenzhen, China

Abstract

Image restoration problems are typical ill-posed problems where the regularization term plays an important role. The regularization term learned via generative approaches is easy to transfer to various image restoration, but offers inferior restoration quality compared with that learned via discriminative approaches. On the contrary, the regularization term learned via discriminative approaches are usually trained for a specific image restoration problem, and fail in the problem for which it is not trained. To address this issue, we propose a generic diffusion process (genericDP) to handle multiple Gaussian denoising problems based on the Trainable Non-linear Reaction Diffusion (TNRD) models. Instead of one model, which consists of a diffusion and a reaction term, for one Gaussian denoising problem in TNRD, we enforce multiple TNRD models to share one diffusion term. The trained genericDP model can provide both promising denoising performance and high training efficiency compared with the original TNRD models. We also transfer the trained diffusion term to non-blind deconvolution which is unseen in the training phase. Experiment results show that the trained diffusion term for multiple Gaussian denoising can be transferred to image non-blind deconvolution as an image prior and provide competitive performance.

1 Introduction

Image restoration problems, *e.g.*, image denoising, deconvolution, super-resolution and et. al, have been researched for decades, and are still active research areas. In image restoration problems, we aim to recover the clean image u, given its degraded counterpart f generating by the following procedure,

$$f = Au + v, \tag{1}$$

© 2018. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

where v is the added noise, for Gaussian denoising, v is assumed to be additive zero mean Gaussian noise. A is the degradation operator, *e.g.*, for image denoising, A is identity matrix; for image super-resolution, A is decimating operator; for image deconvolution, A is blur operator.

1.1 Related Works

It is well known that image restoration problems are typical ill-posed problems. Variational approaches are suitable to solve these problems with the proper regularization terms. A typical variational model is given as

$$E(u,f) = \mathcal{R}(u) + \mathcal{D}(u,f), \tag{2}$$

where $\mathcal{R}(u)$ is the regularization term, and $\mathcal{D}(u,f)$ is the data term. Widely used image regularization term models include the most well-known Total Variation (TV) functional [19], Total Generalized Variation (TGV) [2], Expected Patch Log Likelihood (EPLL) [26] and Fields of Experts (FoE) based analysis operator [5, 17].

In recent years, machine learning based approaches have achieved better restoration performance compared with the hand-crafted regularization terms and widely used BM3D [7]. The machine learning based approaches can be divided into two groups, generative approaches and discriminative approaches. Generative approaches, *e.g.*, FoE [17], K-SVD [8] and EPLL [26], aim to learn the probabilistic model of natural images, which is used as the regularization term to recover various degraded images. On the contrary, discriminative approaches aim to learn the inference procedure that minimizes the energy (9) using pairs of degraded and clean images, *e.g.*, Cascade Shrinkage Fields (CSF, [20]), and Trainable Non-linear Reaction Diffusion (TNRD, [6]).

1.2 Our Motivations and Contributions

Taking TNRD as an example, while it offers both high computational efficiency and high restoration quality, it is highly specified for a specific restoration problem and fails in the problem for which it is not trained. If we have to handle Gaussian denoising problems of σ ranging from 1 to 50, we need to train 50 models. In terms of training efficiency, the performance of discriminative approaches is not as good as that of generative approaches. To address this issue, we propose a generic diffusion process model to combine the advantages of both discriminative and generative approaches, i.e., training a model that provides high training efficiency and achieves competitive restoration quality. The contribution of this study is summarized as follows:

(a) Unlike one model for one Gaussian denoising problem in TNRD, we enforce multiple TNRD models handling different noise level to share one diffusion term.

(b) We give the derivations of the gradients for training the proposed model, which is important for training the proposed model. We train the parameters in a supervised manner, and train the models in an end-to-end fashion.

(c) We transfer the trained diffusion term to deal with non-blind image deconvolution which is unseen in the training phase. The resulting optimization problem is optimized via the half quadratic splitting (HQS).

(d) Experiment results show that the genericDP model can achieve almost the same performance compared with the TNRD model trained for a specific σ. In non-blind image deconvolution problem, with this trained diffusion term, it provides competing results.

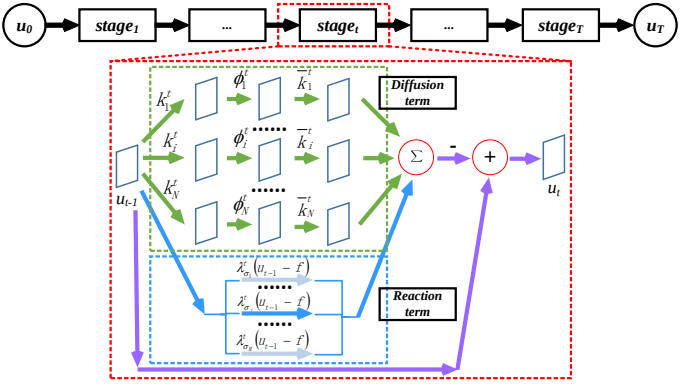


Figure 1: The architecture of the proposed genericDP.

2 Generic Diffusion Process

In this section, we first introduce the proposed generic diffusion process (genericDP), and then we give the gradients of the loss function w.r.t. the parameters.

2.1 Generic Diffusion Process

To handle the inefficiency of discriminative training, we enforce these TNRD models handling different noise level to share one diffusion term, each of which only keeps its own reaction term [1] , formulated as follow,

$$\mathrm{E}(u, f_j) = \sum_{i=1}^{M} \mathbb{1}_i \cdot \frac{\lambda_{\sigma_i}}{2} \|u - f_i\|^2 + \sum_{i=1}^{N_k} \rho_i(k_i * u), \tag{3}$$

where $\mathbb{1}_i$ is an indicator function that is set to 1 when $i = j$, otherwise is set to 0. M is the number of noise levels. Truncating the gradient descent of minimizing Eq. (3) with T steps, we arrive the proposed genericDP model, described as follow,

$$\begin{cases} u_0 = f_j, \quad t = 1, \cdots, T \\ u_t = u_{t-1} - \left(\sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i^t(k_i^t * u_{t-1}) + \lambda_{\sigma_j}^t (u_{t-1} - f_j) \right). \end{cases} \tag{4}$$

The resulting model is shown in Fig. 1. Given an input $u_0 = f_j$ that degraded by Gaussian noise with σ_j, only the data term corresponding to σ_j is used. In this work, we parameterize the local filters and nonlinear functions following [6].

2.2 Training of GenericDP

The parameters of the genericDP in (4) are $\Theta = \{\Theta^t\}_{t=1}^{t=T}$, where $\Theta^t = \{k_i^t, \phi_i^t, \lambda_{\sigma_1}^t, \cdots, \lambda_{\sigma_M}^t\}$. Given these training image pairs of degraded input $f_{j_s}^s$ and ground-truth u_{gt}^s, the training

[1] By omitting the reaction term, the diffusion process is able to handle multiple Gaussian denoising as well. Details are discussed in Section 3.

procedure is formulated as

$$\begin{cases} \Theta^* = \text{argmin}_\Theta \mathcal{L}(\Theta) = \sum_{s=1}^{S} \ell\left(u_T^s, u_{gt}^s\right) \\ \text{s.t.} \begin{cases} u_0^s = f_{j_s}^s, \quad t = 1 \cdots T \\ u_t^s = u_{t-1}^s - \left(\sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i^t(k_i^t * u_{t-1}^s) + \lambda_{\sigma_j}^t (u_{t-1}^s - f_{j_s}^s)\right), \end{cases} \end{cases} \tag{5}$$

where u_T^s is the output of the genericDP in Eq. (4). The inputs are generated using one σ_j among $\sigma_1, \cdots, \sigma_M$. In this paper, the loss function is $\ell(u_T, u_{gt}) = \frac{1}{2}\|u_T - u_{gt}\|^2$.

If not specifically mentioned, we just omit the sample index s to keep the derivation clear. It is easy to extend the following derivation for one training sample to all training samples. Using back-propagation technique [12], the gradients of $\ell(u_T, u_{gt})$ w.r.t. the parameters Θ is

$$\frac{\partial \ell(u_T, u_{gt})}{\partial \Theta_t} = \frac{\partial u_t}{\partial \Theta_t} \cdot \frac{\partial u_{t+1}}{\partial u_t} \cdots \frac{\partial \ell(u_T, u_{gt})}{\partial u_T}. \tag{6}$$

Computing the gradient of $\ell(u_T, u_{gt})$ w.r.t $\lambda_{\sigma_j}^t$. Given Eq. (4), the derivative of u_t w.r.t. $\lambda_{\sigma_j}^t$ is computed as

$$\frac{\partial u_t}{\partial \lambda_{\sigma_j}^t} = -(u_{t-1} - f_j)^\top. \tag{7}$$

Coining $\frac{\partial \ell(u_T, u_{gt})}{\partial u_t} = e$, the derivative of $\ell(u_T, u_{gt})$ w.r.t. λ_{σ_j} is given as

$$\frac{\partial \ell(u_T, u_{gt})}{\partial \lambda_{\sigma_j}^t} = -(u_{t-1} - f_j)^\top e. \tag{8}$$

We only use the samples generated by σ_j to compute the gradients $\ell(u_T, u_{gt})$ w.r.t. λ_{σ_j}, and update the parameter $\lambda_{\sigma_j}^t$ with these gradients.

The derivative of $\ell(u_T, u_{gt})$ w.r.t. k_i^t and ϕ_i^t are similar to that in [6]. Different from the update of parameter $\lambda_{\sigma_j}^t$, all training samples are used to update the parameter k_i^t and ϕ_i^t.

3 Experimental Results

In this section, we first investigate the influence of the parameters, and then compare the trained genericDP models with the state-of-the-art denoising methods. Finally, we transfer the trained diffusion term to the non-blind image deconvolution which is unseen in the training phase, and compare it with the state-of-the-art non-blind deconvolution methods.

3.1 Training Setups

The training dataset is constructed over 1000 natural images collected from the Internet. We randomly cropped 2 regions of size 90×90 from each image, resulting in a total of 2000 training images of size 90×90.

Given pairs of noisy input and ground-truth images, we minimize Eq. (5) to learn the parameters of the genericDP models with commonly used gradient-based L-BFGS [14].

As TNRD model serves as a strong baseline, we initialize the parameters Θ using the TNRD model. We tested the trained models on the 68 images test dataset [18]. We evaluated the denoising performance using PSNR [6] and SSIM [24].

Table 1: Influence of the number of training samples, the range of noise level and with/without reaction term.

	σ			
	5	15	25	50
$\text{TNRD}^8_{7\times7}$	37.77	31.42	28.94	26.01
2k-M=25	37.57	31.38	28.85	-
4k-M=25	37.55	31.41	28.91	-
4k-M=15	37.65	31.40	-	-
4k-M=50	37.38	31.35	28.90	25.99
4k-M=50-w/o	34.19	30.78	28.55	25.68

3.2 Influence of the Parameters

Number of the training samples. In this experiment, the inference stage T was set to 8, the range of the noise level M was set to 25. In [6], 400 images of size 180×180 are used. In terms of the number of total pixels, 1600 images of size 90×90 may be sufficient. Therefore, we use 2000 images of size 90×90 as described above. We also used 4000 images to train the models by doubling each training image in those 2000 images.

We first trained the $\text{TNRD}^8_{7\times7}$ model using the new training samples. The trained TNRD model provided almost the same denoising performance with the models trained using 400 images of size 180×180 in [6]. Therefore, using the new training images to train the genericDP models does not introduce extra image content, which may contribute to the model performance improvement.

As shown in Table 1, given more training images, the overall performance was improved and was very competing with the original TNRD trained for each specific noise level. Therefore, $S = 4000$ is preferred.

Range of the noise levels. In this experiment, the inference stage T was set to 8, the number of the training samples S was set to 4000. We investigated the influence of the range of noise level M, by shortening the range to $M = 15$ and enlarging the range to $M = 50$. In $M = 15$, compared with that in $M = 25$, the performance was improved only in $\sigma = 5$, but very limited . In $M = 50$, compared with that in $M = 25$, the performance in $\sigma = 5$ was reduced by 0.17dB; the performance in other σ were similar to that in $M = 25$. As the time of training these models with different M is almost the same, $M = 50$ is preferred.

Note that, we only trained one genericDP model instead of 50 TNRD models to handle multiple Gaussian denoising with σ ranging of 1 to 50. Therefore, the training time of the genericDP model is almost 50 times faster than that of training 50 TNRD models.

Inference stage. In [6], the TNRD model is considered as a multi-layer network or convolutional network. Meanwhile deeper models, *e.g.*, VGG [22] and ResNet [11], have achieved success in image classification on ImageNet. Therefore, it is worth trying more inference stages in the genericDP model as well.

The number of training images S was set to 4000, the range of noise level M was set to 50. We trained the genericDP model by setting inference stage T to 8, 10 and 16 [2]. As inference stage T increasing, the genericDP model did not provide significantly better denoising performance. Considering the training time and model performance, $T = 8$ is prefered.

[2]There is no available TNRD models for inference stage $T = 10$ and $T = 16$, we trained the genericDP models in greedy and joint training scheme from a plain initialization.

Table 2: Denoising comparison on test images [18].

Method	σ				runtime (s)
	5	15	25	50	
BM3D	37.59 / 0.9635	31.08 / 0.8717	28.57 / 0.8013	25.62 / 0.6864	1.2
$\text{TNRD}^8_{7\times7}$	37.77 / 0.9651	31.42 / 0.8822	28.94 / 0.8162	26.01 / 0.7062	2.2
EPLL	37.57 / 0.9648	31.19 / 0.8820	28.68 / 0.8119	25.68 / 0.6873	145.3
KSVD	37.21 / 0.9621	30.87 / 0.8664	28.29 / 0.7869	25.18 / 0.6551	-
genericDP	37.38 / 0.9620	31.35 / 0.8803	28.90 / 0.8142	25.99 / 0.7030	2.3

3.3 With/without Reaction Term

Intuitively, it makes scene to train the genericDP model without reaction term. The trained genericDP model in such setting, coined as genericDP-wo, offered inferior denoising results compared with the genericDP model trained with reaction term, as shown in the last two rows of Table 1. The genericDP-wo model merely contains a pure diffusion process. Therefore, it tends to oversmooth the texture regions and/or produces artifacts in the homogeneous regions. Therefore, we argue that when training the genericDP model, the reaction term is crucial and is preferred in the following experiments.

3.4 Image Denoising

We trained the genericDP model by setting, the number of training images $S = 4000$, the range of noise level $M = 50$, the inference stage $T = 8$, and with reaction term. We compared the trained genericDP model with BM3D [7], EPLL [26], KSVD [1] and TNRD [6] in Gaussian denoising. The codes of the competing methods were downloaded from the authors' homepage. The comparison noise level were σ =5, 15, 25 and 50. Dealing with one noise level, the genericDP used the corresponding reaction term for that noise level. The comparison results are listed in Table 2. Visual comparison is shown in Fig. 2. Despite the efficient transferring, the generative approaches, *e.g.*, EPLL and KSVD, provide inferior denoising performance, and the inference procedure is quite long [3]. In [16], gating networks are exploited to accelerate the inference process for EPLL-GMM. While inference time goes down significantly, the gated EPLL provides inferior results compared with EPLL-GMM.

Our genericDP model provides competing results with $\text{TNRD}^8_{7\times7}$ trained for a specific noise level, but runs slightly slower than $\text{TNRD}^8_{7\times7}$.

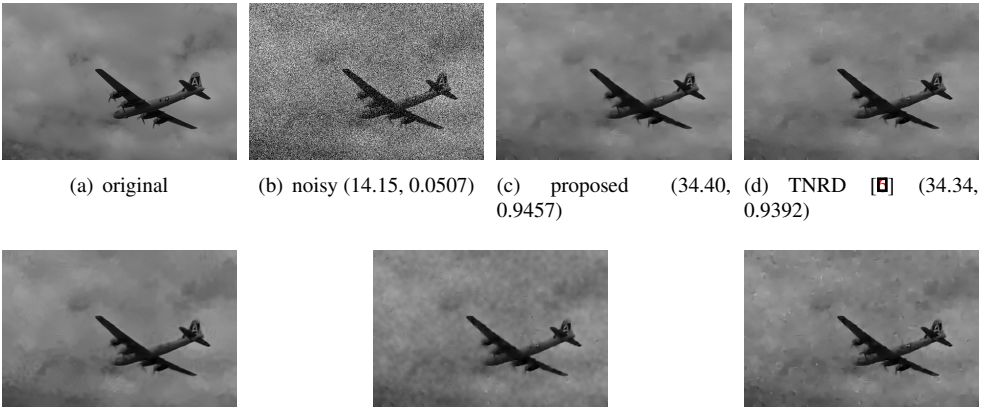
3.5 Non-blind Deconvolution

It is more flexible by decouple the data term and regularization term in Eq. (9) to transfer image prior or prior-like term, *e.g.*, state-of-the-art denoising methods, to other image restoration problems.

$$E(u,f) = \mathcal{R}(u) + \mathcal{D}(u,f), \tag{9}$$

where $\mathcal{R}(u)$ is the regularization term, and $\mathcal{D}(u,f)$ is the data term. In [3, 4, 15, 23], Alternating Direction Method of multiplier (ADMM) is exploited to split the regularization and

[3] The runtime of KSVD varies for different noise level, 761, 117, 51, 18 seconds for σ = 5, 15, 25 and 50 respectively. However, the runtime of KSVD is still larger than that of our genericDP model.



(a) original (b) noisy (14.15, 0.0507) (c) proposed (34.40, 0.9457) (d) TNRD [6] (34.34, 0.9392)

(e) BM3D [9] (33.45, 0.9266) (f) KSVD [1] (31.88, 8940) (g) EPLL [26] (32.51, 0.8868)

Figure 2: Image denoising comparison. From left to right, images are generated by original, noisy, ours genericDP model, TNRD, BM3D, KSVD and EPLL. The original image is added noise level $\sigma = 50$. The numbers in the blankets are PSNR and SSIM values respectively.

data term, while in [9, 10, 20, 25] half quadratic splitting (HQS) is used. In this paper, we focus on HQS approaches.

By introducing a couple of auxiliary variables z, one can decouple the data term and regularization term in Eq. (9), which is reformulated as

$$\mathrm{E}(u,z,f) = \mathcal{R}(z) + \mathcal{D}(Au,f) + \frac{\beta}{2}\|u-z\|^2, \tag{10}$$

where $\frac{\beta}{2}\|u-z\|^2$ is the added quadratic term. β is the penalty parameter, when $\beta \to \infty$, the solution of Eq. (10) goes very close to that of Eq. (9). In HQS setting, Eq. (10) is divided into two sub-problems as

$$\begin{cases} z_{t+1} = \arg\min\limits_{z} \mathcal{R}(z) + \frac{\beta}{2}\|u_t - z\|^2 \\ u_{t+1} = \arg\min\limits_{u} \mathcal{D}(Au,f) + \frac{\beta}{2}\|u - z_{t+1}\|^2. \end{cases} \tag{11}$$

By alternatively minimizing these two sub-problems and increasing β iteratively, we can get the estimation of latent image $\hat{u}$. Sub-problem z_{t+1} can be regarded as a denoising process using image prior $\mathcal{R}(z)$, *e.g.*, FoE, or using state-of-the-art denoising methods, *e.g.*, BM3D or TNRD. Sub-problem u_{t+1} aims to find a solution which satisfies the data term and is close to the z_{t+1}. The commonly used data term is in $\ell 2$ norm, i.e., $\mathcal{D}(Au,f) = \frac{\lambda}{2}\|Au-f\|^2$. With the updated z_{t+1}, u_{t+1} sub-problem has a closed-form solution,

$$u_{t+1} = (\lambda_1 A^\top A + \mathrm{I})^{-1}(\lambda_1 A^\top f + z_{t+1}), \tag{12}$$

where $\lambda_1 = \frac{\lambda}{\beta}$, $\mathrm{I} \in \mathcal{R}^{p\times p}$ is an identity matrix.

The regularization term trained in previous subsection is donated as $\mathcal{R}_{denoise}$. Therefore, the optimization of non-blind deconvolution is reformulated as

$$\begin{cases} z_{t+1} = \arg\min\limits_{z} \mathcal{R}_{denoise}(z) + \frac{\beta}{2}\|u_t - z\|^2 \\ u_{t+1} = \arg\min\limits_{u} \mathcal{D}(Hu,f) + \frac{\beta}{2}\|u - z_{t+1}\|^2, \end{cases} \tag{13}$$

Table 3: Non-blind image deconvolution comparison on test images [13].

Method	σ				runtime (s)
	2.55	5.10	7.65	10.20	
$\text{TNRD}^8_{7\times7}$	28.68 / 0.8597	27.97 / 0.8357	27.41 / 0.8145	26.92 / 0.7939	2.2
EPLL	26.94 / 0.8894	26.09 / 0.8481	25.31 / 0.8102	24.65 / 0.7767	54.4
3×3 FoE	32.46 / 0.9231	30.09 / 0.8745	28.71 / 0.8379	27.74 / 0.8077	2740.5
$\mathcal{R}_{denoise}$	31.39 / 0.9142	29.62 / 0.8766	28.15 / 0.8347	26.97 / 0.7899	1.6

where β is increasing exponentially. Matrix H is the matrix form of the blur kernel h. In this process, λ in $D(Hu, f)$ is set to $\frac{100}{1}$, $\frac{100}{4}$, $\frac{100}{9}$ and $\frac{100}{16}$ for $\sigma = 2.55$, 5.10, 7.65 and 10.20 respectively. β increases exponentially as $\beta = \gamma^i$, where γ is set to 1.8, 1.7, 1.6 and 1.5 for $\sigma = 2.55$, 5.10, 7.65 and 10.20 respectively. The power $i = t - 1$ for each inference stage t respectively. The number of iteration here is simply the number of inference stage T, while the number of iteration is 30 for [25] and more than 100 for [4].

We compare the $\mathcal{R}_{denoise}$ with TNRD [6], EPLL [26] and FoE [21] in non-blind deconvolution. The codes of the competing methods were downloaded from the authors' homepage [4]. The test images are from [13], which are widely used in image deconvolution. In this test dataset, there are eight blur kernels and four images. The blurry images are generated in the following way, firstly applying a blur kernel and then adding zero-mean Gaussian noise with noise level σ. The noise levels are 2.55, 5.10, 7.65 and 10.20.

As illustrated in Table 3, the $\mathcal{R}_{denoise}$ provides competing results with FoE, and better results than EPLL and TNRD. The $\mathcal{R}_{denoise}$ runs slightly faster than TNRD [5], and faster than EPLL and FoE. While FoE method provides good deconvolution results, the inference time is very long. Therefore, it is not scalable to recover large image for FoE. Considering the inference efficiency and deconvolution performance, the $\mathcal{R}_{denoise}$ is very promising. Visual comparison is shown in Fig. 3.

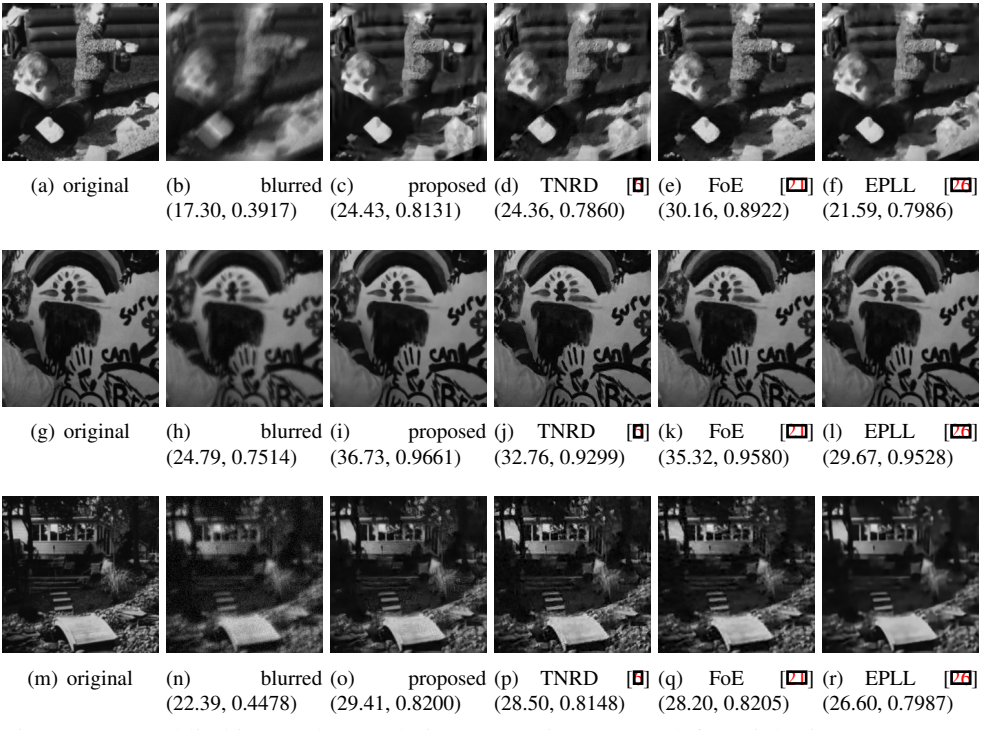
4 Conclusion

Instead of training multiple TNRD models, we enforce these diffusion processes sharing one diffusion term and keeping its own reaction term. As a result, we only need to train one model to handle multiple Gaussian denoising of σ in a range. We derive the gradients of loss function w.r.t. the parameters, and train the model in a supervised and end-to-end manner. The trained genericDP model can offer very competing denoising performance compared with the original TNRD model trained for each specific noise level. Meanwhile, the training efficiency is very impressive compared with TNRD and even generative approaches. We transfer the trained diffusion term to non-blind deconvolution using HQS method. Experiment results show that the trained diffusion term can be used as a generic image prior and work well in image non-blind deconvolution which is unseen during training.

In this work, we train the genericDP model using images only degraded by Gaussian noise in a range. We will use more types of degradation operators A, *e.g.*, image super-

[4] There is no available TNRD codes for non-blind deconvolution, we implement it and train it using greedy training.

[5] Using HQS, we can accelerate the sub-problem u_{t+1} described in Eq. 13 using Fast Fourier Transform (FFT). Note that one needs to careful handle the image boundary conditions.



(a) original (b) blurred (17.30, 0.3917) (c) proposed (24.43, 0.8131) (d) TNRD [6] (24.36, 0.7860) (e) FoE [21] (30.16, 0.8922) (f) EPLL [26] (21.59, 0.7986)

(g) original (h) blurred (24.79, 0.7514) (i) proposed (36.73, 0.9661) (j) TNRD [6] (32.76, 0.9299) (k) FoE [21] (35.32, 0.9580) (l) EPLL [26] (29.67, 0.9528)

(m) original (n) blurred (22.39, 0.4478) (o) proposed (29.41, 0.8200) (p) TNRD [6] (28.50, 0.8148) (q) FoE [21] (28.20, 0.8205) (r) EPLL [26] (26.60, 0.7987)

Figure 3: Non-blind image deconvolution comparison. From left to right, images are generated by original, blurred, ours genericDP model, TNRD, FoE and EPLL. In the first two rows, the original image is blurred by kernel 4 and 5 respectively, then is added noise level $\sigma = 2.55$. In the last row, the original image is blurred by kernel 5, then is added noise level $\sigma = 10.20$. The numbers in the blankets are PSNR and SSIM values respectively.

resolution, image deblocking, to train the genericDP model, hoping that a more generic image prior can be learned. We will also transfer the learned diffusion term to other unseen image restoration problems to validate the generality of the trained diffusion term.

5 Acknowledgements

This work was supported by the National Natural Science Foundation of China under the Grant No.U1435219, No.61732018 and No.61602032.

References

[1] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 54(11):4311, 2006.

[2] Kristian Bredies, Karl Kunisch, and Thomas Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.

[3] Alon Brifman, Yaniv Romano, and Michael Elad. Turning a denoiser into a super-resolver using plug and play priors. In *Image Processing (ICIP), 2016 IEEE International Conference on*, pages 1404–1408. IEEE, 2016.

[4] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2017.

[5] Yunjin Chen, Rene Ranftl, and Thomas Pock. Insights into analysis operator learning: From patch-based sparse models to higher order mrfs. *IEEE Transactions on Image Processing*, 23(3):1060–1072, 2014.

[6] Yunjin Chen, Wei Yu, and Thomas Pock. On learning optimized reaction diffusion processes for effective image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5261–5269, 2015.

[7] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007.

[8] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12): 3736–3745, 2006.

[9] Donald Geman and George Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Transactions on pattern analysis and machine intelligence*, 14 (3):367–383, 1992.

[10] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing*, 4(7):932–946, 1995.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[12] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Understanding and evaluating blind deconvolution algorithms. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1964–1971. IEEE, 2009.

[14] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[15] Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *arXiv preprint arXiv:1611.02862*, 2016.

[16] Dan Rosenbaum and Yair Weiss. The return of the gating network: combining generative models and discriminative training in natural image priors. In *Advances in Neural Information Processing Systems*, pages 2683–2691, 2015.

[17] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 860–867, 2005.

[18] Stefan Roth and Michael J Black. Fields of experts. *International Journal of Computer Vision*, 82(2):205–229, 2009.

[19] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[20] Uwe Schmidt and Stefan Roth. Shrinkage fields for effective image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2774–2781, 2014.

[21] Uwe Schmidt, Kevin Schelten, and Stefan Roth. Bayesian deblurring with integrated noise estimation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2625–2632. IEEE, 2011.

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[23] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 945–948. IEEE, 2013.

[24] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Processing*, 13(4):600–612, 2004.

[25] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. *arXiv preprint arXiv:1704.03264*, 2017.

[26] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, 2011.