2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428158

ROBUST IMAGE DENOISING WITH TEXTURE-AWARE NEURAL NETWORK

Bo Fu[1,2], *Liyan Wang*[2], *Zhongxuan Luo*[1,*]

[1]School of Software Technology, Dalian University of Technology, Dalian, China
[2]School of Computer and Information Technology, Liaoning Normal University, Dalian, China
fubo@lnnu.edu.cn, wangliyanlee@163.com, zxluo@dlut.edu.cn.

ABSTRACT

Image denoising is a well-studied yet still hot research topic in the image processing community. Recently, image denoising with deep neural networks has achieved superior performance, however they can not recover tiny details from noisy images. Motivated by this problem, we propose a Texture-Aware Neural Network named TANet, which is composed of main network part with attention mechanism, residual structure and Texture-Aware Modular. Proposed Texture-Aware Modular owns dual paths, denoised image from main denoising network and clean image are input different path respectively. From Texture-Aware Modular, we get two sets intermediate codes and calculate corresponding perceptual loss. This perceptual loss is designed to generate auxiliary supervision for tiny detail recovery from mixed residual details and noise set. Extensive experimental results demonstrate that the proposed TANet is on a par with the state-of-the-art denoising methods.

我只能说写得很烂

***Index Terms*—** Image denoising, Texture-Aware Neural Network, Perceptual loss, Tiny details

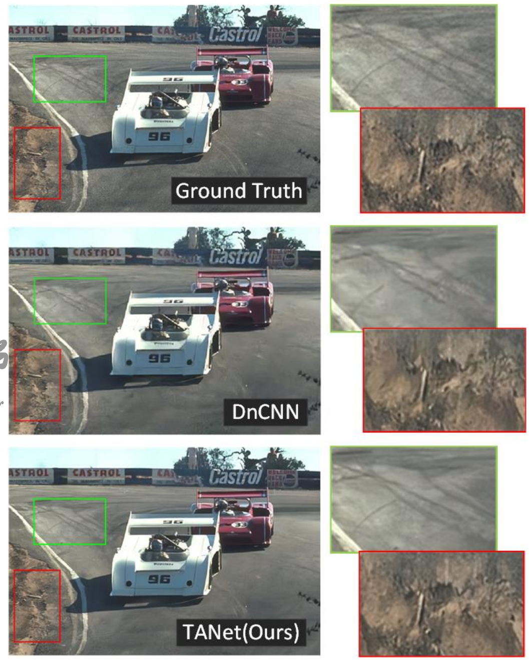

Fig. 1. A comparison of denoised results with level of 25 between the TANet and DnCNN [12]. Our TANet method can recover more details compared with DnCNN method, visually closer to Ground Truth.

1. INTRODUCTION

Image denoising [1] is a well-studied yet still hot research topic in the image processing community, since it is a significant pre-processing step before applying high-level vision tasks. Its goal is to recover clean image from corrupted image y with various additive noises. The most common scenario of noise is the additive white Gaussian (AWG) noise, supposing that n is drawn from a noise Gaussian distribution with mean 0 and variance σ^2, i.e. $n \sim N(0, \sigma^2)$.

Over a period of time, many single image based methods have been proposed for handling image denoising with the AWG noise [2, 3, 4, 5, 6, 7, 8, 9]. For example, the representative methods include non-local means (NLM)[1], block-matching 3D filtering (BM3D)[6] and weighted nuclear norm minimization (WNNM)[7], cascade of shrinkage fields (CSF)[10], trainable nonlinear reaction diffusion (TNRD)[11] etc. These traditional methods recover clean image by mainly exploiting the global or non-local similar information from the current single noisy image, or using prior knowledge to smooth the noisy image and recover the details of potentially clean image. These traditional methods work well with lower noise levels. However, existing single image based methods suffer from following drawbacks that limit their applicability:(1) most methods need to be optimized in the test phase; (2) the parameters in the model need to be set manually; (3) too few training data leads to poor universality of the model.

哪来的train?

With the continuously development of deep learning technology, its strong self-learning ability is applied in high-level computer vision [13, 14, 15, 16] and low-level computer vision tasks such as image restoration and image segmentation[17, 18]. In 2012, Burger et al. introduced multi-layers neural network to resolve denoising task[11]

*Corresponding author: zxluo@dlut.edu.cn

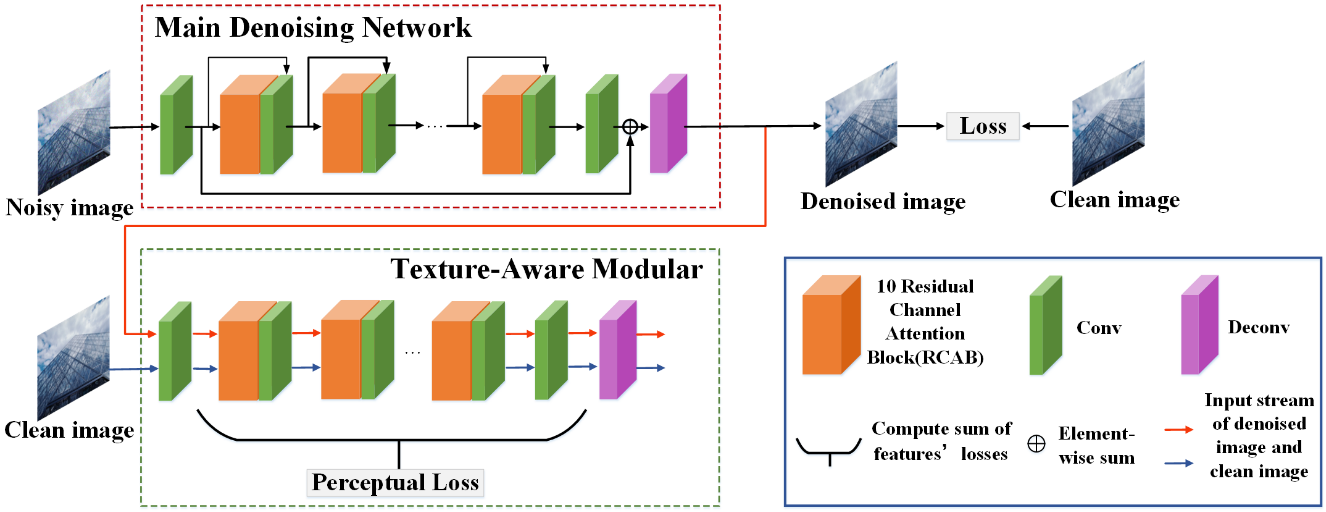

Fig. 2. The network architecture of the Texture-Aware Neural Network(TANet).

with superior performance than BM3D[6] which owned the state of the art performance at that time. Mao et al. adopted a multiple CNN architecture to resolve restoration task including image denoising and super-resolution[19]. Another interesting attempt is DnCNN[12] which focus on predicting noise image instead of directly estimating the repaired image. Lefkimmiatis introduced non-local similarity into CNN denoiser [20]. For resolving blind denoising task, FFDNet[21] can use a same model to adapt to denoising tasks with different noise levels. CBDNet[22] can effectively estimate the noise map in different levels and remove noise from noisy image.

In general, the above methods own strong denoising ability which are capable of smoothing noise meanwhile recovering image salient features. However, on handling more complex noisy images which own rich and mixed texture content, existing denoising methods still need to be improved. In response to this kind of problem, we propose a Texture-Aware Neural Network named TANet which owns main network part with attention mechanism, residual structure and Texture-Aware Modular. Proposed Texture-Aware Modular owns dual paths, denoised image from main network and clean image are input different path respectively. From Texture-Aware Modular, we get two sets intermediate codes and corresponding calculate perceptual loss. Then, this perceptual loss is designed to generate auxiliary supervision that facilitates recovery tiny details from mixed residual details and noise set. In figure 1, we show that the TANet can effectively recovery more details from complex and irregular texture region of noisy image.

module design

2. RELATED WORKS

In the previous section, there are many deep learning denoising methods improve effects by increasing the number of network layers and increasing the network width, and modifying the loss function. Among of these methods, some of methods are focus on feature extraction and texture recovery. In this section, we review this kind of methods which closely to our work.

真的吗 我不信

Due deep learning is essentially a process of selecting generative features, feature extraction and representation are very important in the whole process of image denoising. However traditional deep learning methods are difficult to control and choose internal features, it may lead to some atypical features being submerged in the process of feature selection. Motivated by this problem, some researchers tried to control the process of features extraction. For example, Liu et al. introduced wavelet tools to generate deep learning features [23]. Mildenhall et al. converted non-linear features into linearity by a set of kernels facilitate digging details information [24]. Channel attention mechanism is also proposed to enhance internal features of deep learning, Zhang et al. proposed RCAN [17] and introduced its networks framework to image super-resolution task. These methods improve the detail recovery of denoising network from different aspects, but they usually cannot solve the problem that residual noise and residual details are difficult to distinguish. Different from these methods, we proposed CNN based method constrained by Texture-Aware Modular which supervising detail texture recovery from network internal features perspective.

3. PROPOSED METHOD

In section 3.1, we briefly review denoising mechanism of deep neural networks and further introduce our denoising framework for image denoising. In section 3.2, we introduce our Texture-Aware Neural Network and its corresponding loss function.

废话

3.1. Revisit denoising mechanism of deep neural networks

Here, we firstly define some universal terminology necessary to any formal description of denoising problem. We denote a clean image as x , and x is usually corrupted by many additive noises n to generate corrupted image named y, described as $y = x + n$. Any image denoising task is basically to learn a generating function F to apply to corrupted image, yields higher visual quality image denotes y' reducing efficiently the influence of the noise. As far as we know, the basic idea of denoising neural network is adopt neural networks to solve a regress problem where input corrupted image and output denoised image [17, 18, 21] or predicted noise [12, 22]. And this regress problem can be further divided into three sub-problems including (1)How to construct train set? (2) How to design network neural structure and its target function? (3)How to test a new unseen image? For an end to end denoising neural network structure, its corresponding training set can be given by two independent set X and set Y, clean image x and noisy image y is come from set X and set Y, respectively. Given this paired training set of clean and noisy images $\{(x, y)\}$, denoising task is equivalent to a multi-output regression problem. Then, we can introduce a neural network model to predict the denoised image. It can be achieved by minimizing a target function as following:

$$\min_{\theta} \sum_{i=1}^{M} L(F(y|\theta), x) + \lambda R(\theta) \tag{1}$$

where $L(\cdot)$ denotes the loss function, $F(\cdot|\theta)$denotes a neural network with a set of parameters θ; $R(\theta)$ is the regularization term with respect to θ and $\lambda > 0$ is the regularization parameter. In test step, the well trained model $F(\cdot|\theta^*)$ can be used to generate denoised images from any unseen corrupted image.

3.2. Proposed Texture-Aware Neural Network

The essence of denoising neural network repairing is to learn a function to suppress the noise in the noise image and restore corrupted pixels to its original information as much as possible. We denote the difference between the repaired image and clean image as ϵ. Although most of the current denoising neural networks have been able to learn salient features or noise features, these networks generally cannot let ϵ small enough to satisfy requirements for tiny detail recovery. We believe that the main reason for this limitation is that most denoising neural networks are insensitive to the difference between tiny inconspicuous features and noise. Thus, these denoising neural networks cannot learn the difference between the residual noise and the normal inconspicuous tiny details.

大伙都懂啊

To alleviate this problem, we propose a novel image denoising framework. The overall structure of TANet is shown

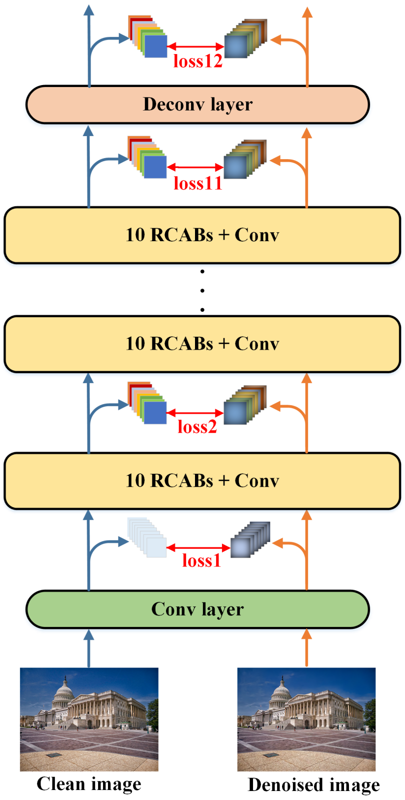

Fig. 3. The architecture of Texture-Aware Modular.

in Figure 2. In TANet, there are two parts, main denoising network part and additional Texture-Aware Modular. We specially design the Texture-Aware Modular to distinguish between residual noise and the normal inconspicuous tiny details shown in Figure 3, thought to be regularizers. The Texture-Aware Modular consists of two paths, referred to as denoised path and proposal path. Both of them share the same CNN network architecture. The denoised path is used to fit $\{(y', x)\}$; and the proposal path is used to fit $\{(x, x)\}$. The Texture-Aware network which is trained between two clean image datasets is supposed to find the subtle difference. Thus, we use the denoised image and corresponding clean image as input of Texture-Aware network to dig the tiny detail as much as possible. So in Texture-Aware Modular, we can get two intermediate feature codes from the corresponding hidden layer of proposal path and denoised path respectively. We think that the features of the network middle layer have a certain similarity in the feature space. What we are trying to find this internal similarity is for retaining as much important information in the input data (such as weak texture information), supplementing and assisting the training of the main network. Formally, consider any architecture of Texture-Aware Modular $f(\cdot)$ with L hidden layers. Let $\phi = \phi_1, \phi_2, \ldots, \phi_L$ denote the parameters of Texture-Aware Modular. Besides, we define $\phi_{1 \to j} = \phi_1, \phi_2, \ldots, \phi_L$ to denote the parameters before the j-th hidden layer. In the learning paradigm of whole

训练过程?

合理

Table 1. The PSNR and SSIM scores of our method and some denoising methods for Grayscale images (Red numbers denote the best scores while blue numbers denote the second best scores)

Datasets	σ	BM3D	IRCNN	DnCNN	FFDNet	RCAN(denoising)	TANet
Set12	15	32.37/0.8952	32.76/0.9006	32.85/0.9025	32.74/0.9024	33.02/0.9059	33.08/0.9068
	25	29.97/0.8505	30.37/0.8598	30.43/0.8617	30.42/0.9631	30.58/0.8660	30.73/0.8686
	50	26.72/0.7676	27.12/0.7804	27.17/0.7828	27.30/0.7899	27.49/0.7962	27.57/0.7976
BSD68	15	31.08/0.8722	31.74/0.8907	31.64/0.8902	31.64/0.8882	31.83/0.8938	31.88/0.8945
	25	28.57/0.8017	29.23/0.8279	29.19/0.8288	29.15/0.8248	29.29/0.8316	29.41/0.8352
	50	25.62/0.6869	26.24/0.7189	26.29/0.7239	26.19/0.7169	26.23/0.6813	26.43/0.7292
Urban100	15	32.34/0.9220	32.48/0.9244	32.67/0.9250	32.42/0.9273	33.08/0.9277	33.16/0.9293
	25	29.70/0.8777	29.81/0.8840	29.96/0.8793	29.92/0.8887	30.32/0.8905	30.63/0.8946
	50	25.94/0.7791	26.23/0.7928	26.27/0.7869	26.51/0.8075	27.03/0.8123	27.15/0.8168

Table 2. The PSNR and SSIM scores of our method and some denoising methods for Color images (Red numbers denote the best scores while blue numbers denote the second best scores)

Datasets	σ	CBM3D	IRCNN	DnCNN	FFDNet	RCAN(denoising)	TANet
Set14	15	-	33.23/0.9009	32.39/0.8894	33.18/0.9013	33.08/0.8978	33.12/0.8978
	25	-	30.91/0.8599	30.45/0.8515	30.98/0.8608	30.91/0.8580	31.02/0.8603
	50	-	27.84/0.7817	27.65/0.7739	28.05/0.7851	28.09/0.7867	28.16/0.7893
CBSD68	15	33.52/0.9215	33.90/0.9290	33.88/0.9290	33.87/0.9285	34.21/0.9331	34.29/0.9339
	25	30.71/0.8672	31.24/0.8830	31.22/0.8821	31.18/0.8824	31.49/0.8885	31.61/0.8905
	50	27.38/0.7626	27.95/0.7896	27.97/0.7887	27.88/0.7898	28.29/0.8021	28.32/0.8037
Urban100	15	-	33.78/0.9402	32.98/0.9314	33.83/0.9418	33.88/0.9386	33.96/0.9382
	25	-	31.20/0.9088	30.81/0.9015	31.40/0.9120	31.61/0.9098	31.77/0.9112
	50	-	27.70/0.8396	27.59/0.8331	28.05/0.8476	28.65/0.8534	28.72/0.8582

TANet, Texture-Aware Modular need be trained firstly, formulated as follows:

$$\min_{\phi} \sum_{i=1}^{M} L(F(x|\phi), x) + \lambda R(\phi) \tag{2}$$

Given the well trained ϕ^*, we design the perceptual losses to generate auxiliary supervision, formulated as follows:

→用F(y|θ)会怎么样?

$$\min_{\theta} \sum_{i=1}^{M} L(F(y|\theta), x) + \sum_{j \in \Omega} L_p(f_j(y'|\phi^*), f_j(x|\phi^*)) + \lambda R(\theta) \tag{3}$$

The first item is to calculate the loss function between clean image and output. The second is to calculate the loss function between the feature x and y' of the middle layer output, when both the clean image and the denoised image pass through the Texture-Aware network at the same time, and the third is regularization term. Where Ω denotes a layer index set, storing the selected hidden layers associated with the perceptual loss; L_p denotes the perceptual loss of the j-th hidden layer; and $f_j(\cdot)$ denotes the shallower base network before the j-th hidden layer.

4. EXPERIMENTS

4.1. Experimental Setting

Dataset For gray images, we use 432 paired images from BSD500 dataset as the training set, and the rest of 68 paired images are verified and tested. For color images, we used CBSD500 as data set and the same amount of training and testing is used as the grayscale. For the noisy image training set, the simulated Gaussian noise is added to the original data set.
Baseline algorithm We use five existing image denoiesd algorithms as baseline, including BM3D[6], IRCNN[18], DnCNN[12], FFDNet[21], RCAN(denoising)[17].
Evaluation Metric The peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) are adopted to measure the objective performance of our algorithm.

4.2. Parameter Setting and Training Details

Our network structure consists of two parts, which are a main denoising network and an texture-aware neural network, respectively. They have the same structure, which all con-

?

tain about 100 residual channel attention blocks (RCABs)[17] with the channel attention mechanism, 12 alone convolution layers and a deconvolution layer. In addtion, we use the long and short skip connections throughout the whole network. On the parameter setting, the training pictures is cut into the patches with size of 48×48. The batch size is set as 16, the learning rate is set as le-4, and the ADAM is adopted as optimizer of our model.

如何监督?

During training stage, Firstly,we only train 20 epochs for the Texture-Aware Modular, which uses GT as input to implement an end-to-end network training, so it is easy to converge. Secondly, 500 epochs for main denoising network imported the trained Texture-Aware Modular, and we calculate the loss function of every 10 RCABs output features to assist the training of the main denoising network. Finally, we choose same training dataset and iteration times for comparison methods such as RCAN(denoising) and our method. Our network framework is based on the PyTorch platform, and all experiments are implemented on a workstation with NVIDIA Titan X GPU, and Core (TM) i7-7700K CPU.

4.3. Performance Comparison

Experiments are carried out on several common public test datasets, including Set12, Set14, BSD68, Urban100. The PSNR scores and SSIM scores of our TANet and some denoised methods(Grayscale) are compared in Table 1. The PSNR scores and SSIM scores of our TANet and some denoised methods(Color) are compared in Table 2.

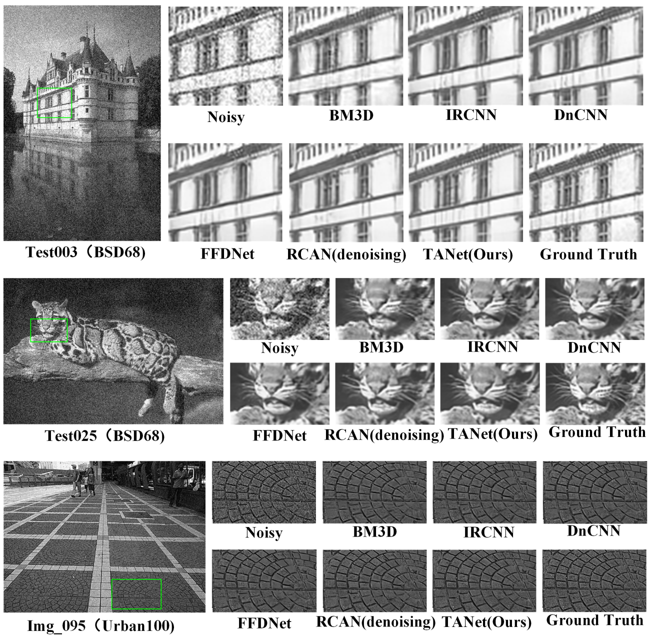

Fig. 4. Denoising results of the gray images with noise level 25.

It can be seen that our algorithm obtain superiority in most test set either the PSNR or the SSIM index. It proves that our method can remove the noise and repair the details better. At the same time, gray visual comparisons with level of 25 on the images from different test sets are also provided in Figure 4, and the color visual comparisons are provided in Figure 5.

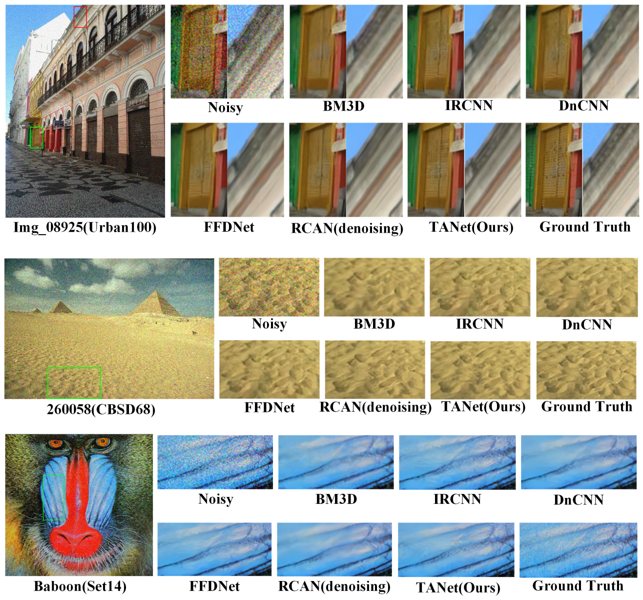

Fig. 5. Denoising results of the color images with noise level 25.

Because the basic network framework of the algorithm comes from RCAN[17], we separately train RCAN for denoising task in order to compare the effectiveness of Texture-Aware Modular in TANet. It can be seen that our method has significantly improved both in PSNR and visual effect than situation with no Texture-Aware Modular.

5. CONCLUSION

Recently, existing deep learning denoising networks usually can remove noise effectively meanwhile recovery tiny detail texture good enough. Motivated by this problem, we propose a Texture-Aware Neural Network which owns attention mechanism, residual structure and Texture-Aware Modular, named TANet. Proposed Texture-Aware Modular can generate auxiliary supervision by calclulating perceptual loss. In terms of revory details, our TANet shows superior performance. A series of experiments demonstrate that the proposed TANet is favorably against state-of-the-art algorithms.

?

6. ACKNOWLEDGEMENT

This work is supported by China Postdoctoral Science Foundation No.2019M651123 and Science and Technology Innovation Found (Youth Science and Technology Star) of Dalian, China No. 2018RQ65.

7. REFERENCES

[1] P. Chatterjee and P. Milanfar, "Is denoising dead?," *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 895–911, May. 2010. 1

[2] Chang, S. Grace, and Bin Yu, "Adaptive wavelet thresholding for image denoising and compression.," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1532, May. 2000. 1

[3] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Oct. 2005, pp. 60–65, IEEE Computer Society. 1

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, May. 2007. 1

[5] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *2009 IEEE 12th International Conference on Computer Vision*. Oct. 2009, pp. 2272–2279, IEEE Computer Society. 1

[6] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian, "Image restoration by sparse 3d transform-domain collaborative filtering," in *Image Processing: Algorithms and Systems VI, San Jose, California,USA, January 28-29, 2008*. Jun. 2008, vol. 6812 of *SPIE Proceedings*, p. 681207, SPIE. 1, 1, 4.1

[7] M. Lebrun, A. Buades, and J. M. Morel, "A nonlocal bayesian image denoising algorithm," *Siam Journal on Imaging ences*, vol. 6, no. 3, pp. 1665–1688, Jul. 2013. 1

[8] S. Gu, L. Zhang, W. M. Zuo, and X. C. Feng, "Weighted nuclear norm minimization with application to image denoising," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014, pp. 2862–2869, IEEE Computer Society. 1

[9] F. Chen, L. Zhang, and H. Yu, "External patch prior guided internal clustering for image denoising," in *IEEE International Conference on Computer Vision*. 2015, pp. 603–611, IEEE Computer Society. 1

[10] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014, pp. 2774–2781, IEEE Computer Society. 1

[11] Y. J. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, Nov. 2017. 1, 1

[12] K. Zhang, W. M. Zuo, Y. J. Chen, D. Y. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, Oct. 2016. 1, 1, 3.1, 4.1

[13] L. Wu, Y. Wang, and L. Shao, "Cycle-consistent deep generative hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1602–1612, May. 2019. 1

[14] J. J. Peng, Y. Wang, H. B. Wang, Z. Zhang, X. P. Fu, and M. Wang, "Unsupervised vehicle re-identification with progressive adaptation," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, Christian Bessiere, Ed. 2020, pp. 913–919, ijcai.org. 1

[15] Y. Wang, "Survey on deep multi-modal data analytics:collaboration, rivalry and fusion," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. abs/2006.08159, Jun. 2020. 1

[16] L. Wu, Y. Wang, H. Z. Yin, M. Wang, and L. Shao, "Few-shot deep adversarial learning for video-based person re-identification," *IEEE Trans. Image Process.*, vol. 29, no. 1, pp. 1233–1245, May. 2020. 1

[17] Y. L. Zhang, K. P. Li, K. Li, L. C. Wang, B. N. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," vol. 11211, pp. 294–310, 2018. 1, 2, 3.1, 4.1, 4.2, 4.3

[18] K. Zhang, W. M. Zuo, S. H. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2808–2817, IEEE Computer Society. 1, 3.1, 4.1

[19] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d?," in *IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 2392–2399, IEEE Computer Society. 1

[20] S. Lefkimmiatis, "Non-local color image denoising with convolutional neural networks," pp. 5882–5891, 2017. 1

[21] K. Zhang, W. M. Zuo, and L. Zhang, "Ffdnet: Toward a fast and flexible solution for cnn based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, May. 2018. 1, 3.1, 4.1

[22] S. Guo, Z. F. Yan, K. Zhang, W. M. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," pp. 1712–1722, 2019. 1, 3.1

[23] P. J. Liu, H. Z. Zhang, K. Zhang, L. Lin, and W. M. Zuo, "Multi-level wavelet-cnn for image restoration," in *2018 IEEE Conference on Computer Vision and Pattern RecognitionWorkshops, CVPR Workshops 2018, Salt Lake City, UT, USA, June 18-22,2018*. 2018, pp. 773–782, IEEE Computer Society. 2

[24] B. Mildenhall, J. T. Barron, J. W. Chen, D. Sharlet, and R. Carroll, "Burst denoising with kernel prediction networks," pp. 2502–2510, 2018. 2