

FSOINET: FEATURE-SPACE OPTIMIZATION-INSPIRED NETWORK FOR IMAGE COMPRESSIVE SENSING

Wenjun Chen, Chunling Yang*, Xin Yang

School of Electronic and Information Engineering, South China University of Technology, China
(eecwjjun@mail.scut.edu.cn; eeclyang@scut.edu.cn; eexyang@mail.scut.edu.cn)

ABSTRACT

In recent years, deep learning-based image compressive sensing (ICS) methods have achieved brilliant success. Many optimization-inspired networks have been proposed to bring the insights of optimization algorithms into the network structure design and have achieved excellent reconstruction quality with low computational complexity. But they keep the information flow in pixel space as traditional algorithms by updating and transferring the image in pixel space, which does not fully use the information in the image features. In this paper, we propose the idea of achieving information flow phase by phase in feature space and design a Feature-Space Optimization-Inspired Network (dubbed FSOINET) to implement it by mapping both steps of proximal gradient descent algorithm from pixel space to feature space. Moreover, the sampling matrix is learned end-to-end with other network parameters. Experiments show that the proposed FSOINET outperforms the existing state-of-the-art methods by a large margin both quantitatively and qualitatively. The source code is available on <https://github.com/cwjjun/FSOINET>.

Index Terms— Image compressive sensing, deep learning, convolutional neural network, image reconstruction

1. INTRODUCTION

Compressive sensing (CS) [1], which demonstrates that a signal can be reconstructed by fewer measurements than the Nyquist sampling theorem when it has sparsity in a proper transform domain, has gained increasing attention for several years. As an inverse problem, CS aim to reconstruct the original signal $\mathbf{x} \in \mathbb{R}^N$ from its CS measurements $\mathbf{y} \in \mathbb{R}^M$ ($M \ll N$) gained with a linear projection $\mathbf{y} = \Phi\mathbf{x}$, where $\Phi \in \mathbb{R}^{M \times N}$. The inverse problem is ill-posed and can be converted into the following optimization problem:

$$\min_{\mathbf{x}} \frac{1}{2} \|\Phi\mathbf{x} - \mathbf{y}\|_2^2 + \lambda\psi(\mathbf{x}), \quad (1)$$

where $\psi(\mathbf{x})$ comes from prior knowledge and λ is a regularization parameter. Iterative optimization algorithms usually

use proximal gradient descent to solve this problem, which can be depicted as two update steps:

$$\mathbf{r}^{(k)} = \mathbf{x}^{(k-1)} - \rho\Phi^T(\Phi\mathbf{x}^{(k-1)} - \mathbf{y}), \quad (2)$$

$$\mathbf{x}^{(k)} = \text{prox}_{\lambda, \psi}(\mathbf{r}^{(k)}), \quad (3)$$

where $\text{prox}_{\lambda, \psi}(\mathbf{r}) = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{r}\|_2^2 + \lambda\psi(\mathbf{x})$, k denotes the iteration index and ρ is the step size.

It's well known that natural image has lots of redundant information. In consequence, the study of how to use CS theory in image compression has become more significant. Because of the high resolution of the image, Block-based Compressive Sensing (BCS) [2] was proposed to sample the image more efficiently by sampling the image block by block. Recently, Convolution Neural Networks (CNN) have been successfully used in many computer vision tasks. First CNN-based ICS method [3] provides improved performance in terms of reconstruction quality and speed than traditional iterative optimization-based CS methods. Subsequent CNN-based ICS methods could be divided into two categories, one is vanilla neural networks and the other is optimization-inspired neural networks. The former [4–7] build neural networks as a black box to learn the non-linear mapping from the CS measurements to the reconstructed image. The latter [8–13] unfold iterative algorithms to neural networks comprising fixed numbers of phases, and each phase corresponds to an iteration in traditional algorithms.

Although optimization-inspired neural networks usually get better reconstruction quality by using measurements in each reconstruction phase, they still keep the information flow in pixel space between phases as traditional algorithms. Each phase supplements information in pixel space according to Eq.2 and only transfers pixel-space image denoised by the network, which suffers from two limitations. First, the simple process of correcting information in pixel space from measurements can not fully exploit the information contained in measurements. Second, the structure which transfers information in pixel space doesn't make full use of the information contained in the image features.

To overcome these drawbacks, we build a novel ICS network structure unfolding iterative optimization algorithms in feature space, dubbed Feature-Space Optimization-Inspired

Supported by the Key Program of Natural Science Foundation of Guangdong Province (2017A030311028) and the Natural Science Foundation of Guangdong Province (2019A1515011949). * means corresponding author

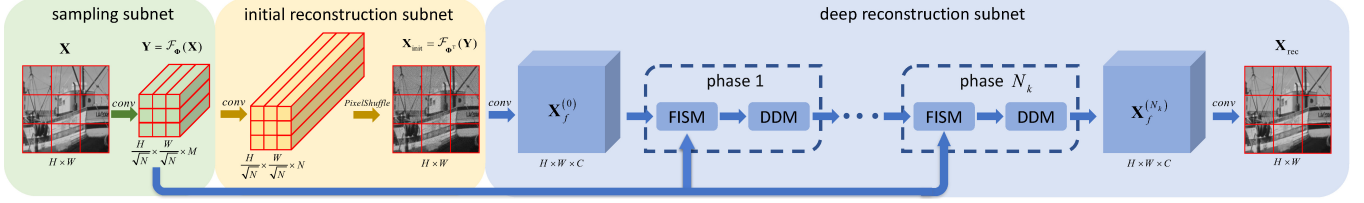


Fig. 1. The architecture of the FSOINet.

Network (FSOINet) for ICS, which processes and transfers image features phase by phase to make full use of features extracted from the image. In each phase, we use measurements to supplement the information in feature space and then denoise the image feature by building a Feature-space Information Supplementing Module (FSIM) and a Dual-scale Denoising Module (DDM).

The main contributions of this paper are summarized as follows: 1) We propose the idea of implementing the information flow phase by phase in feature space for ICS. 2) A FSIM and a DDM are designed to implement update steps in feature space for supplementing the information from measurements and dual-scale denoising, respectively. 3) A novel deep unfolding network named FSOINet is proposed, which contains a sampling subnet, an initial reconstruction subnet, and a deep reconstruction subnet. 4) Experiment shows our framework outperforms other state-of-the-art ICS methods significantly in terms of both reconstruction quality and visual quality.

2. THE PROPOSED FSOINET

Inspired by the iterative optimization algorithm, as shown in Fig.1, we built FSOINet comprising three subnets: a sampling subnet, an initial reconstruction subnet, and a deep reconstruction subnet. The first two subnets respectively realize the linear sampling and initial reconstruction between the pixel domain and the measurement domain. The last subnet completes the non-linear reconstruction of the image by processing image features, which contains N_k phases and each of them corresponds to an iteration in the iterative process.

2.1. Sampling and Initial Reconstruction

In the BCS [2] strategy, an image $\mathbf{X} \in \mathbb{R}^{H \times W}$ is divided into $\frac{H}{\sqrt{N}} \times \frac{W}{\sqrt{N}}$ non-overlapping image blocks with size of $\sqrt{N} \times \sqrt{N}$, then each of them is reshaped into a vector $\mathbf{x} \in \mathbb{R}^N$. When the CS sampling rate is r , the same sampling matrix $\Phi \in \mathbb{R}^{M \times N}$ is used to sample each \mathbf{x} , where $M = \lceil r \times N \rceil$. In addition, Cui et al. [14] proposes to use a learnable sampling matrix instead of fixed random Gaussian Matrix to get CS measurements, and have achieved gratifying results. In this paper, we use a sampling sub-network which is denoted as $\mathcal{F}_\Phi(\cdot)$ to sample the image and use convolution without bias to simulate the block sampling process described

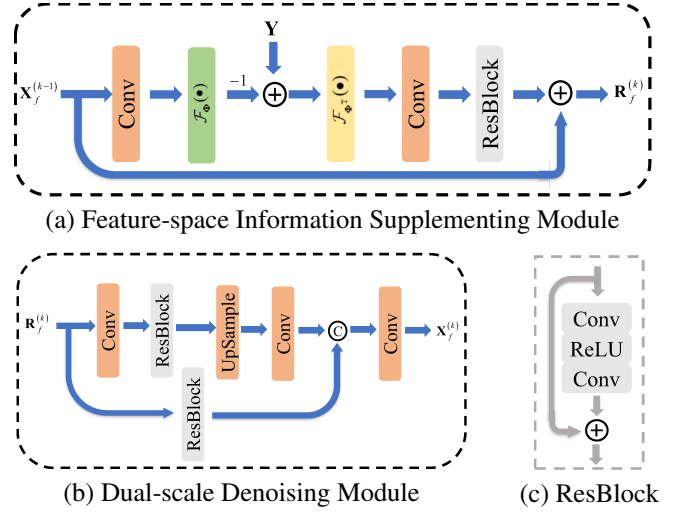


Fig. 2. The architecture of the FSIM and DDM.

above. To be specific, we set Φ as learnable network parameters, reshape each row of Φ into a convolution kernel of size $1 \times \sqrt{N} \times \sqrt{N}$ to obtain W_Φ . Therefore, the sampling process can be described by convolution with a stride of \sqrt{N} as:

$$\mathbf{Y} = \mathcal{F}_\Phi(\mathbf{X}) = W_\Phi * \mathbf{X}. \quad (4)$$

To obtain a reasonable initial estimate of each image block from CS measurements and not introduce more learnable parameters in the meanwhile, we use Φ^T to complete the initial reconstruction. Like the sampling process, each row of Φ^T is reshaped into a convolution kernel of size $M \times 1 \times 1$ to obtain W_Φ^T , then PixelShuffle is used to obtain the initial reconstruction image, which can be illustrated as:

$$\mathbf{X}_{\text{init}} = \mathcal{F}_{\Phi^T}(\mathbf{Y}) = \text{PixelShuffle}(W_\Phi^T * \mathbf{Y}). \quad (5)$$

2.2. Deep Reconstruction

In previous optimization-inspired ICS networks, each phase implements supplementing information in pixel space according to Eq.2 and sends the denoised image to the next phase, although the denoising is finished in feature space. This process makes the information flow in pixel space phase by phase,

Table 1. Average PSNR and SSIM comparisons of different ICS methods on Set11, BSD68 and BSD100.

Datasets	methods	cs ratio r				
		0.01	0.05	0.1	0.3	0.5
Set11	CSNet ⁺	21.02/0.5566	25.86/0.7846	28.34/0.8508	34.30/0.9490	38.52/0.9749
	SCSNet	21.04/0.5562	25.85/0.7839	28.52/0.8616	34.64/0.9511	39.01/0.9769
	SPLNet	21.22/0.5552	26.59/0.8177	29.49/0.8874	35.79/0.9603	40.27/0.9815
	AMP-Net	20.20/0.5425	26.17/0.8128	29.40/0.8876	36.03/0.9623	40.34/0.9821
	OPINE-Net ⁺	20.02/0.5362	26.36/0.8186	29.81/0.8904	36.04/0.9600	40.19/0.9800
	FSOINet	21.73/0.5937	27.36/0.8415	30.44/0.9018	37.00/0.9665	41.08/0.9832
BSD68	CSNet ⁺	21.71/0.5249	25.04/0.6845	26.89/0.7756	31.66/0.9152	35.42/0.9614
	SCSNet	21.88/0.5250	24.98/0.6843	27.13/0.7785	31.76/0.9173	35.67/0.9640
	SPLNet	22.33/0.5242	25.87/0.7198	27.85/0.8094	32.77/0.9303	36.86/0.9708
	AMP-Net	22.28/0.5315	25.77/0.7204	27.85/0.8113	32.84/0.9321	36.82/0.9715
	OPINE-Net ⁺	21.88/0.5162	25.66/0.7136	27.81/0.8040	32.50/0.9236	36.32/0.9658
	FSOINet	22.75/0.5418	26.21/0.7324	28.27/0.8187	33.29/0.9348	37.34/0.9727
Urban100	CSNet ⁺	19.27/0.4812	22.63/0.6792	24.64/0.7741	29.90/0.9162	33.55/0.9572
	SCSNet	19.28/0.4798	22.63/0.6774	24.93/0.7827	30.12/0.9193	33.92/0.9601
	SPLNet	19.55/0.4873	23.55/0.7301	26.19/0.8290	32.11/0.9405	36.41/0.9737
	AMP-Net	19.62/0.4969	23.45/0.7290	26.04/0.8283	32.19/0.9418	36.33/0.9737
	OPINE-Net ⁺	19.38/0.4872	23.70/0.7363	26.61/0.8362	32.58/0.9414	36.62/0.9727
	FSOINet	19.87/0.5223	24.57/0.7750	27.53/0.8627	33.84/0.9540	37.80/0.9777

which limits the use of the robust feature representation ability of CNN. In this paper, we propose the idea of implementing the information flow phase by phase in feature space, including supplementing the information from measurements and dual-scale denoising all in the feature space, to fully utilize the feature representation ability of CNN and retain more details when the information flows to the next phase. The deep reconstruction subnet is shown in Fig.1.

Information supplement in feature space: Consistent with the optimization algorithm, we also use measurements throughout the entire reconstruction process to guarantee the solution complies with the degradation process $\mathbf{y} = \Phi\mathbf{x}$. And different from previous optimization-inspired ICS methods, we propose the idea of supplementing information in feature space instead of that in pixel space from measurements. To implement this idea, we build a Feature-space Information Supplementing Module (FSIM) to map the gradient of the fidelity term $\frac{1}{2}\|\Phi\mathbf{x} - \mathbf{y}\|_2^2$ in Eq.1 to the feature space and fuse it with the image features, which is shown in Fig.2(a). It is worth noting that $\mathcal{F}_\Phi(\cdot)$ and $\mathcal{F}_\Phi^T(\cdot)$ in Fig.2(a) are block operations, which will introduce blocking artifacts. Here, ResBlock operates on the entire image, which can suppress blocking artifacts.

Dual-scale denoising in feature space: Benefiting from the data-driven characteristics of deep learning, the inherent prior knowledge of the image can be learned from the training set to denoise the image. In this paper, we propose a dual-scale structure network to improve denoising efficiency. And we design a neural network module called Dual-scale Denoising Module (DDM) to implement it, as shown in Fig.2(b). To maintain not too many parameters, only two ResBlocks are used to denoise the high-resolution features

and low-resolution features respectively in DDM. Then, the denoised features are fused to obtain the output features, which will be sent into the next phase. Finally, we implement the information flow in feature space phase by phase. The high-resolution features maintain the same resolution as the original image, while the low-resolution features are subsampled from the high-resolution features with stride-2 Conv and have twice the number of channels.

2.3. Loss Function

For an original image \mathbf{X} , the proposed model first obtains CS measurements \mathbf{Y} by sampling it and then predicts the reconstructed image \mathbf{X}_{rec} . We optimize our FSOINet end-to-end through the following loss function:

$$\mathcal{L} = \mathcal{L}_{mse}(\mathbf{X}_{\text{rec}}, \mathbf{X}) + \gamma \mathcal{L}_{orth}(\Phi), \quad (6)$$

where \mathcal{L}_{mse} is the Mean Squared Error (MSE) of the original image \mathbf{X} and the reconstructed image \mathbf{X}_{rec} . In addition, \mathcal{L}_{orth} is the orthogonal constraint of the sampling matrix in [12]: $\mathcal{L}_{orth}(\Phi) = \frac{1}{M^2} \|\Phi\Phi^T - \mathbf{I}\|_F^2$, where \mathbf{I} represents the identity matrix. The regularization parameter γ in Eq.6 is set to 0.01 in our experiments.

3. EXPERIMENT RESULT

3.1. Dataset and Implementation Details

For training, following [5], we use 400 images from the training set and test set of the BSDS500 dataset [15]. The training images are cropped to 89600 96×96 pixel sub-images with data augmentation. For the network parameters, the block

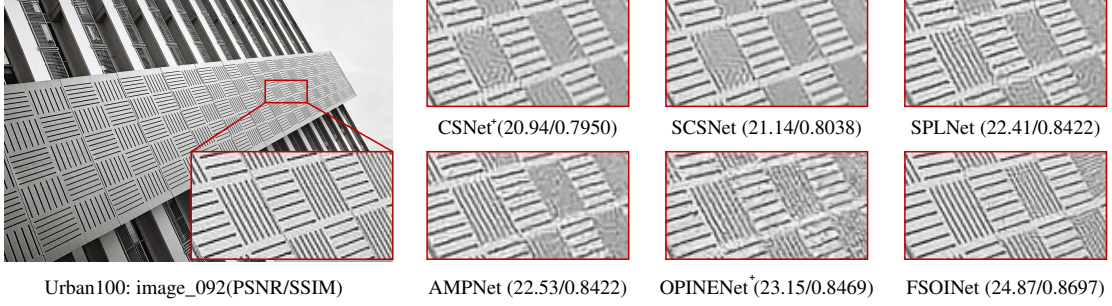


Fig. 3. Visual comparison of various ICS methods on one sample from Urban100 at the CS sampling ratio of 0.3.

Table 2. Model size and average computational time of different ICS methods on BSD68 ($r = 0.5$).

Model	SPLNet	AMPNet	OPINENet ⁺	FSOINet
#Para	1.39M	1.53M	1.10M	1.06M
Time	0.0059s	0.0466s	0.0109s	0.0239s

size \sqrt{N} is 32, the channel number C is 16, the phase number N_k is 16 and the batch size is 32. The size of the unspecified convolution kernel is 3×3 . We use Adam [16] optimizer to train the network with the initial learning rate of 2×10^{-4} , which is decreased to 5×10^{-5} through 100 epochs using the cosine annealing strategy [17] and the warm-up epochs are 3. All the experiments are implemented in PyTorch with a Core i5-6500 CPU Intel and a GTX2080ti GPU. For testing, we utilize three widely-used benchmark datasets, including Set11 [3], BSDS68 [18] and Urban100 [19]. Color images are processed in the YCbCr space and evaluated on the Y channel. Two common-used image assessment criteria, Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM), are adopted to evaluate the reconstruction results.

3.2. Comparisons with State-of-the-Arts

Considering that the sampling matrix obtained by training is often better than the fixed Gaussian random matrix. In order to make fair comparisons, we compare our FSOINet with five state-of-the-art ICS methods whose sampling matrices are learnable too, including CSNet [5], SCSNet [6], SPLNet [10], AMPNet [11], and OPINE-Net [12]. Extensive experiments have proved the advantages of FSOINet in terms of quality and visualized results.

Table 1 and Fig.3 clearly show that our FSOINet outperforms all the other competing methods at all the CS sampling rates. It is worth noting that the three optimization-inspired methods, SPLNet, OPINENet, and AMPNet, are significantly better than vanilla neural networks CSNet and SCSNet in terms of reconstruction quality. But our FSOINet got a more compelling reconstruction quality and reconstruct sharper edges and clearer background at all the sampling rates.

Table 2 provides a comparison of model sizes and calcu-

Table 3. Analysis of FSIM in FSOINet. The experiment are evaluated on Set11 ($r = 0.1$).

Model	VNet	OINet	FSOINet
PSNR/SSIM	29.33/0.8895	29.90/0.8942	30.44/0.9018
Parameters	536257	536080	536257

lation times for different methods at a CS ratio of 0.5. Our model has the fewest parameters and medium calculation speed while achieving the best reconstruction results.

3.3. Ablation Study

This section mainly analyzes the effectiveness of using FSIM to replace the gradient descent in the optimization algorithm. We retrain our network without FSIM, then the network is similar to vanilla neural networks, which only use the measurements during the initial reconstruction, dubbed VNet. At the same time, we change the FSIM to the gradient descent in the pixel domain and retrain our network. At this time, the network which we named OINet is more similar to the previous optimization-inspired neural networks. For a fair comparison, we move the convolution block used in FSIM to DDM to maintain similar model parameters.

As shown in Table 3, the PSNR/SSIM of OINet is 0.54dB/0.0076 lower than that of FSOINet at the sampling ratio of 0.1, which reflects the effectiveness of mapping the gradient information to the feature space. While VNet's PSNR/SSIM is 0.57db/0.0047 lower than that of OINet, which indicates the insights of the optimization algorithm could help the neural network structure design.

4. CONCLUSION

In this paper, we propose a novel ICS network structure dubbed FSOINet. The optimization algorithm is unfolded in the feature space through FSIM and DDM so that the information flow is kept phase by phase in the feature space. The experimental results show our model has achieved a significant improvement in performance compared with other state-of-the-art methods.

5. REFERENCES

- [1] David L Donoho, “Compressed sensing,” *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [2] Lu Gan, “Block compressed sensing of natural images,” in *2007 15th International conference on digital signal processing*. IEEE, 2007, pp. 403–406.
- [3] Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok, “Reconnet: Non-iterative reconstruction of images from compressively sensed measurements,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 449–458.
- [4] Kai Xu, Zhikang Zhang, and Fengbo Ren, “Lapran: A scalable laplacian pyramid reconstructive adversarial network for flexible compressive sensing reconstruction,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 485–500.
- [5] Wuzhen Shi, Feng Jiang, Shaohui Liu, and Debin Zhao, “Image compressed sensing using convolutional neural network,” *IEEE Transactions on Image Processing*, vol. 29, pp. 375–388, 2019.
- [6] Wuzhen Shi, Feng Jiang, Shaohui Liu, and Debin Zhao, “Scalable convolutional neural network for image compressed sensing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12290–12299.
- [7] Jiwei Chen, Yubao Sun, Qingshan Liu, and Rui Huang, “Learning memory augmented cascading network for compressed sensing of images,” in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*. Springer, 2020, pp. 513–529.
- [8] Jian Sun, Huibin Li, Zongben Xu, et al., “Deep admm-net for compressive sensing mri,” *Advances in neural information processing systems*, vol. 29, 2016.
- [9] Jian Zhang and Bernard Ghanem, “Ista-net: Interpretable optimization-inspired deep network for image compressive sensing,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1828–1837.
- [10] Hanqi Pei, Chunling Yang, and Yan Cao, “Deep smoothed projected landweber network for block-based image compressive sensing,” in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 2870–2874.
- [11] Zhonghao Zhang, Yipeng Liu, Jiani Liu, Fei Wen, and Ce Zhu, “Amp-net: Denoising-based deep unfolding for compressive image sensing,” *IEEE Transactions on Image Processing*, vol. 30, pp. 1487–1500, 2020.
- [12] Jian Zhang, Chen Zhao, and Wen Gao, “Optimization-inspired compact deep compressive sensing,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 4, pp. 765–774, 2020.
- [13] Di You, Jian Zhang, Jingfen Xie, Bin Chen, and Siwei Ma, “Coast: Controllable arbitrary-sampling network for compressive sensing,” *IEEE Transactions on Image Processing*, vol. 30, pp. 6066–6080, 2021.
- [14] Wenxue Cui, Feng Jiang, Xinwei Gao, Wen Tao, and Debin Zhao, “Deep neural network based sparse measurement matrix for image compressed sensing,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 3883–3887.
- [15] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik, “Contour detection and hierarchical image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 5, pp. 898–916, 2010.
- [16] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [17] Ilya Loshchilov and Frank Hutter, “Sgdr: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.
- [18] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. IEEE, 2001, vol. 2, pp. 416–423.
- [19] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja, “Single image super-resolution from transformed self-exemplars,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5197–5206.