

Appendix for “USB-Net: Unfolding Split Bregman Method With Multi-Phase Feature Integration for Compressive Imaging”

Zhen Guo and Hongping Gan, *Member, IEEE*

APPENDIX

A. Sampling and Initialization Process for Different Compressive Imaging Tasks

Natural Image Compressive Sensing (ICS). Assuming an input natural image $\mathbf{x} \in \mathbb{R}^{H \times W}$ with height and width of H and W , respectively. The image \mathbf{x} is initially partitioned into non-overlapping blocks of size $B \times B$. If the width or height of \mathbf{x} is not exactly divisible by B , zero-padding is applied to ensure consistent block sizes. These blocks are then flattened into vectors, and a sampling matrix $\Phi \in \mathbb{R}^{M \times N}$ ($M \ll N$) is applied to obtain measurements $\mathbf{y} \in \mathbb{R}^M$. Typical choices for Φ include Gaussian or Bernoulli random matrices. Let $\mathcal{F}_B(\cdot) : \mathbb{R}^{W \times H} \rightarrow \mathbb{R}^{B^2}$ denote the partitioning and flattening function, and τ represent the sampling ratio, with $M = \lfloor N \times \tau \rfloor = \lfloor B^2 \times \tau \rfloor$. The sampling process of ICS can be represented as:

$$\mathbf{y} = \Phi \mathcal{F}_B(\mathbf{x}). \quad (1)$$

For ICS, the measurements \mathbf{y} are processed using the transpose of the sampling matrix $\Phi^T \in \mathbb{R}^{N \times M}$ to reconstruct the image blocks. Subsequently, the function $\tilde{\mathcal{F}}_B(\cdot) : \mathbb{R}^{B^2} \rightarrow \mathbb{R}^{W \times H}$ recovers the image blocks and stitches them together to form the initial estimate $\mathbf{x}^{(0)}$. The initialization process of ICS is expressed as:

$$\mathbf{x}^{(0)} = \tilde{\mathcal{F}}_B(\Phi^T \mathbf{y}). \quad (2)$$

Compressive Sensing Magnetic Resonance Imaging (CS-MRI). CS-MRI takes the specific form of a partial Fourier transform matrix as the sampling matrix Φ , *i.e.*, $\Phi = \mathbf{SF}$, where \mathbf{S} represents a sub-sampling mask, and \mathbf{F} corresponds to the Discrete Fourier Transform (DFT). The choice of \mathbf{S} can vary, including options like 1D Cartesian masks, Pseudo Radial masks and others. Moreover, the size of \mathbf{S} is the same as the size of the input image \mathbf{x} , and τ represents the proportion between the number of measurement points M and the total number of pixels N in \mathbf{x} , *i.e.*, $\tau = \frac{M}{N}$. The sampling process of CS-MRI is mathematically expressed as:

$$\mathbf{y} = \Phi \mathbf{x} = \mathbf{SFx}. \quad (3)$$

This work was supported in part by the National Natural Science Foundation of China under Grants 62471395 and 62101455, in part by the Basic Research Programs of Taicang under Grant TC2024JC17.

Zhen Guo and Hongping Gan (Corresponding author) are with the School of Software, Northwestern Polytechnical University (NPU), Taicang 215400, China; They are also with Yangtze River Delta Research Institute of NPU, Taicang 215400, China (e-mail: guozhen2022@mail.nwpu.edu.cn; ganhongping@nwpu.edu.cn).

In CS-MRI, the initialization module commences by applying the inverse DFT, denoted as $\tilde{\mathbf{F}}$, to the acquired measurements \mathbf{y} , yielding the initial estimate $\mathbf{x}^{(0)}$. Thus, the initialization of USB-Net for CS-MRI can be summarized as:

$$\mathbf{x}^{(0)} = \tilde{\mathbf{F}} \mathbf{y}. \quad (4)$$

Snapshot Compressive Sensing (SCI). In the realm of SCI, a 3D hyperspectral image (HSI) $\mathbf{X} \in \mathbb{R}^{H \times W \times N_\zeta}$ is considered, where W , H , and N_ζ denote its height, width, and number of wavelengths, respectively. Initially, the captured HSI undergoes modulation via a pre-defined coded aperture $\Phi_\zeta \in \mathbb{R}^{H \times W}$, yielding the modulated HSI \mathbf{X}' :

$$\mathbf{X}'(:,:,n_\zeta) = \mathbf{X}(:,:,n_\zeta) \odot \Phi_\zeta, \quad (5)$$

where \odot signifies element-wise multiplication, and $n_\zeta \in [1, \dots, N_\zeta]$ indexes the spectral channel. Subsequently, by laterally shifting \mathbf{X}' through a disperser, the resulting measurement $\mathbf{X}'' \in \mathbb{R}^{H \times (W+d(N_\zeta-1))}$ undergoes shear and tilt, with d representing the step of spatial shifting. The dispersion process is formulated as:

$$\mathbf{X}''(u, v, n_\zeta) = \mathbf{X}'(x, y + d(\zeta_n - \zeta_c), n_\zeta), \quad (6)$$

where ζ_c denotes the reference wavelength, ζ_n denotes the wavelength of the n_ζ -th spectral channel, (u, v) represents the coordinate system on the detector array, and $d(\zeta_n - \zeta_c)$ represents the spatial shifting offset of the n_ζ -th channel on \mathbf{X}'' . Consequently, the 2D compressed measurement $\mathbf{Y} \in \mathbb{R}^{H \times (W+d(N_\zeta-1))}$ is obtained as follows:

$$\mathbf{Y} = \sum_{n_\zeta=1}^{N_\zeta} \mathbf{X}''(:,:,n_\zeta) + \mathbf{E}, \quad (7)$$

where \mathbf{E} represents the random image noise generated by the photon sensing detector.

Regarding the initialization phase, the dispersion of CASSI is reversed by shifting back the measurement \mathbf{Y} to derive $\mathbf{X}'^{(0)} \in \mathbb{R}^{H \times W \times N_\zeta}$:

$$\mathbf{X}'^{(0)} = \mathbf{Y}(x, y - d(\zeta_n - \zeta_c)). \quad (8)$$

Then, the concatenation of $\mathbf{X}'^{(0)}$ and the 3D mask $\Phi \in \mathbb{R}^{H \times W \times N_\zeta}$ is fed into a convolutional layer with a kernel size of 1×1 to produce the initial estimate $\mathbf{X}^{(0)} \in \mathbb{R}^{H \times W \times N_\zeta}$:

$$\mathbf{X}^{(0)} = \text{Conv}_1(\text{Concat}(\mathbf{X}'^{(0)}, \Phi)). \quad (9)$$

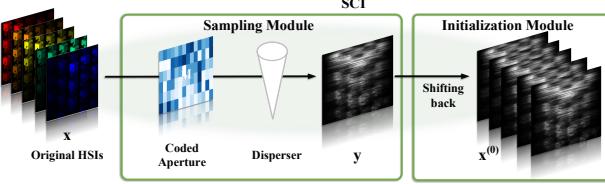


Fig. 1. Sampling and initialization process of SCI.

B. Additional Experiments on ICS

1) *Experimental Settings:* We use a dataset of 41,000 images (40,000 images are used for training, and 1,000 images are reserved for validation) randomly selected from the COCO2017 unlabeled images dataset [1] as train data. During the training process, we first randomly crop the image to 96×96 size and use data augmentation techniques such as scaling and rotating to expand the training dataset. As for USB-Net, Φ is initialized as a Gaussian random matrix; the basic channel number C is set to 32; the number of reconstruction phases n is 8; regularization parameters $\{\lambda^{(k)}\}_{k=1}^n$ are initialized to 1×10^{-3} ; steps sizes $\{\alpha^{(k)}\}_{k=1}^n$ are initialized to 1×10^{-3} and penalty terms $\{\mu^{(k)}\}_{k=1}^n$ are initialized to 1. We adopt the AdamW optimizer with a weight decay of 0.05 and a momentum of 0.9. The training procedure utilizes a batch size of 16 and a block size of $B = 32$. Initially, USB-Net is trained at a sampling ratio of 0.25 for 800 epochs. The learning rate is initially increased linearly from 0 to 1×10^{-4} over 10 epochs for warm-up and is then reduced to 1×10^{-6} over the next 790 training epochs using a cosine annealing scheduler. The other sampling ratios of USB-Net are fine-tuned for 400 epochs using the same scheduler based on the well-trained model at the sampling ratio of 0.25. The evaluation metrics used are peak signal-to-noise ratio (PSNR) and structural similarity (SSIM); and all contrast methods are trained and tested based on their corresponding works. The best and second best results in the tables are highlighted in Red and Blue, respectively.

2) *Additional Comparisons:* First, we compare our USB-Net with up to twenty-one DL-based ICS methods on the Set11 dataset. These contrast methods include model-based methods (ReconNet [2], SCSNet [3], CSNet⁺ [4], DPA-Net [5], MAC-Net [6], BCS-Net [7], NL-CSNet [8], AutoBCS [9], CSformer [10], TCS-Net [11], CSRN [12] and MTC-CSNet [13]) and unfolding-based methods (ISTA-Net⁺ [14], OPINE-Net⁺ [15], AMP-Net [16], COAST [17], CASNet [18], DGUNet⁺ [19], FSOINet [20], TransCS [21], DPC-DUN [22], MAPUN [23], OCTUF [24], LTwIST [25], SODAS-Net [26], UFC-Net [27], NesTD-Net [28]).

As shown in Tab. I, USB-Net achieves superior or competitive performance compared to other methods in terms of reconstructed image quality, as indicated by the higher values of PSNR and SSIM, across all seven sampling ratios. Notably, at the sampling ratio of 0.04, USB-Net displays remarkable improvements. Specifically, there are percentage gains of approximately 0.43 dB (1.60%), 0.81 dB (3.06%), and 1.80 dB (7.07%) in comparison to DGUNet⁺, OCTUF, and TCS-Net, respectively. Moreover, USB-Net showcases consistent enhancements in average SSIM, with percentage

gains of approximately 0.0105 (1.28%), 0.0209 (2.57%), and 0.0472 (6.00%). Besides, the results of high sampling ratios (0.30, 0.40, 0.50) are summarized in Tab. II, demonstrating that USB-Net remains the best in the high sampling ratios as well.

In addition, Fig. 2 presents visual comparisons of the reconstructed image details using various methods at a sampling ratio of 0.10. It is evident that the reconstructed images produced by USB-Net exhibit an absence of block artifacts, which is a common issue observed in ISTA-Net⁺, CSNet⁺, TransCS and TCS-Net. On the other hand, methods such as CASNet, DGUNet⁺ and OCTUF are susceptible to blurring and the omission of fine image details, leading to poor visual perception and a degradation in image quality. In contrast, USB-Net's reconstructed images demonstrate superior preservation of image details, resulting in a better visual sensation and overall image quality. These observations strongly indicate that USB-Net possesses an excellent ability to efficiently extract, preserve, and reconstruct images with high fidelity.

Furthermore, as shown in Fig. 3, we include visualizations of the feature maps at different stages of USB-Net, as compared to ISTA-Net⁺ [14], AMP-Net [16], FSOINet [20], CASNet [18], DGUNet⁺ [19] and OCTUF [24], under sampling ratios of 0.04 and 0.10. Our results demonstrate that, irrespective of the sampling ratio, the initial feature maps of USB-Net contain richer and more accurate information compared to those of other unfolding methods. Furthermore, as the iteration progresses, the detail-rich regions within the feature maps become increasingly pronounced, indicating the effectiveness of USB-Net in feature extraction and fusion. Consequently, USB-Net yields reconstructed images with richer and more precise details, even under low sampling ratios. These findings underscore USB-Net's superior ability to extract and fuse relevant information effectively, enabling the capture of intricate details and the generation of accurate reconstructed images, particularly under challenging conditions such as low sampling ratios.

3) *Recovery Performance Under Gaussian Noise:* In order to evaluate the performance of USB-Net under different Gaussian noise levels, we introduce Gaussian noise to the Set11 dataset and evaluate the model's performance on these noisy images. We explore four distinct levels of Gaussian noise variance, denoted as $\delta \in \{0.001, 0.002, 0.004, 0.008\}$. The results, as shown in Tab. III, reveal that USB-Net consistently achieves the highest PSNR and SSIM scores across all four levels of Gaussian noise. DGUNet⁺ predominantly yields sub-optimal outcomes at sampling ratios of 0.01, 0.04, and 0.10. At the sampling ratio of 0.25, OCTUF displays sub-optimal PSNR performance, whereas CASNet presents sub-optimal SSIM results. Furthermore, we present visual comparisons of reconstructed images using various methods across diverse Gaussian noise levels (with variances $\sigma \in \{0.002, 0.004\}$) at a sampling ratio of 0.04, illustrated in Fig. 4. Notably, the images reconstructed by USB-Net exhibit enhanced details and reduced distortion when compared to other ICS methods.

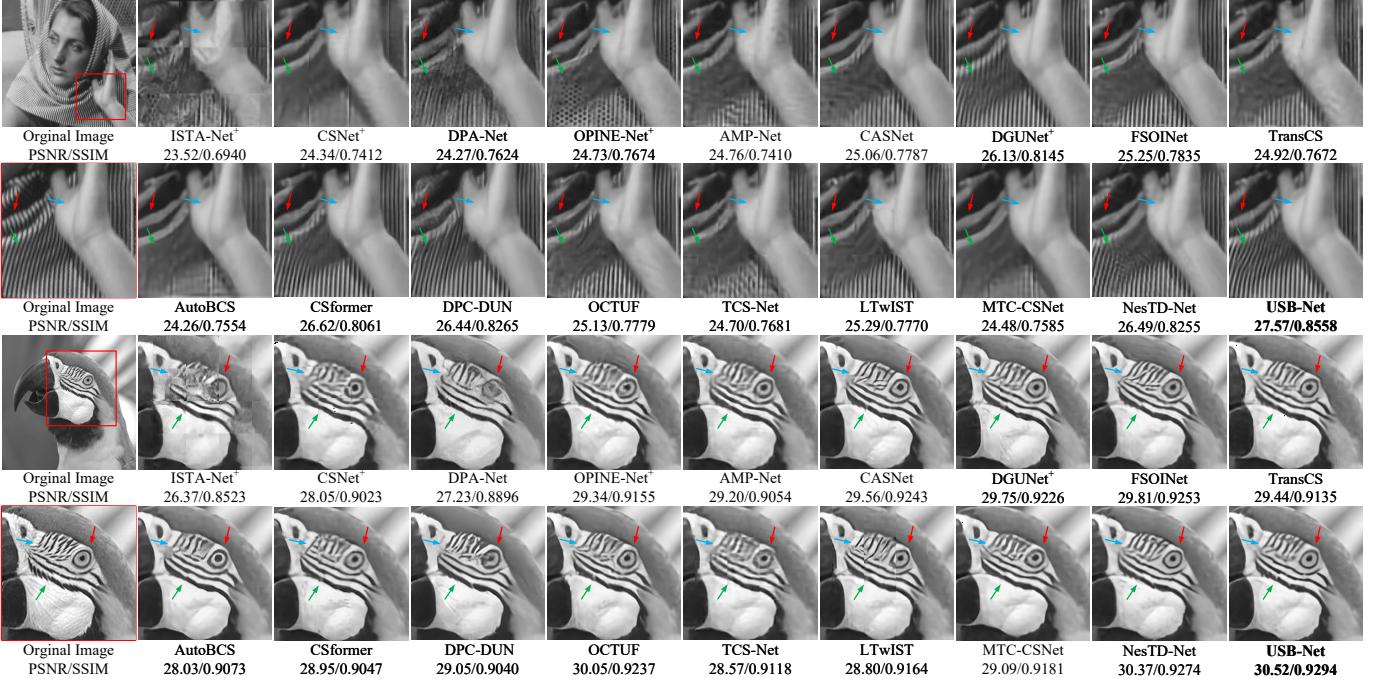


Fig. 2. Visual comparisons of PSNR (dB)/SSIM scores of grayscale images reconstructed by various methods at a sampling ratio of 0.10. Local images are zoomed in for better comparison and analysis. Our proposed USB-Net demonstrates better visualization and excellent preservation of image details.

TABLE I

AVERAGE PSNR (dB) AND SSIM SCORES COMPARISONS AMONG VARIOUS DL-BASED ICS METHODS ON GRAYSCALE DATASET SET11 WITH SEVEN DIFFERENT SAMPLING RATIOS $\tau \in \{0.01, 0.04, 0.10, 0.25, 0.30, 0.40, 0.50\}$. HYPHEN ‘-’ REPRESENTS THAT THE ORIGINAL WORK NEITHER DISCLOSES THE RECOVERY RESULTS NOR PROVIDES THE TRAINING CODE UNDER THESE CONDITIONS.

Dataset	Methods	0.01		0.04		0.10		0.25		0.30		0.40		0.50		Avg.	
		PSNR	SSIM														
Set11	ReconNet (CVPR 2016)	17.43	0.4017	20.93	0.5897	24.38	0.7301	28.44	0.8531	29.09	0.8693	30.60	0.9020	32.25	0.9177	26.16	0.7519
	ISTA-Net ⁺ (CVPR 2018)	17.45	0.4131	21.56	0.6240	26.49	0.8036	32.44	0.9237	33.70	0.9382	36.02	0.9579	38.07	0.9706	29.39	0.8044
	SCSNet (CVPR 2019)	21.04	0.5562	24.29	0.7589	28.52	0.8616	33.43	0.9373	34.64	0.9511	36.92	0.9666	39.01	0.9769	31.12	0.8584
	CSNet ⁺ (TIP 2020)	20.69	0.5238	24.54	0.7445	28.12	0.8664	32.20	0.9337	33.70	0.9495	36.41	0.9677	38.28	0.9771	30.56	0.8518
	DPA-Net (TIP 2020)	18.05	0.5011	23.50	0.7205	26.99	0.8354	31.74	0.9238	33.35	0.9425	35.04	0.9565	36.73	0.9670	29.34	0.8353
	MAC-Net (ECCV 2020)	18.26	0.4003	24.22	0.6982	27.68	0.8182	32.91	0.9244	33.96	0.9372	35.94	0.9560	37.67	0.9668	30.09	0.8144
	OPINE-Net ⁺ (J-STSP 2020)	20.02	0.5362	25.52	0.7879	29.81	0.8904	34.81	0.9514	36.00	0.9600	38.31	0.9724	40.18	0.9800	32.09	0.8683
	AMP-Net (TIP 2021)	20.20	0.5581	25.26	0.7722	29.40	0.8779	34.63	0.9481	36.03	0.9586	38.28	0.9715	40.34	0.9807	32.02	0.8667
	BCS-Net (TMM 2021)	20.81	0.5427	24.90	0.7531	29.36	0.8650	34.20	0.9408	35.40	0.9527	36.52	0.9640	39.58	0.9734	31.54	0.8560
	COAST (TIP 2021)	12.40	0.2637	23.55	0.7158	28.70	0.8609	33.96	0.9405	35.09	0.9504	37.09	0.9645	38.92	0.9743	29.96	0.8100
	NL-CSNet (TMM 2021)	21.96	0.6005	26.26	0.8108	30.05	0.8995	-	-	35.68	0.9606	-	-	-	-	-	-
	CASNet (TIP 2022)	21.76	0.6019	26.25	0.8118	30.29	0.9005	35.65	0.9592	36.90	0.9663	39.03	0.9760	40.93	0.9827	32.97	0.8855
	DGUNet ⁺ (CVPR 2022)	22.15	0.6114	26.83	0.8230	30.93	0.9088	36.18	0.9616	36.72	0.9661	38.99	0.9766	41.24	0.9837	33.29	0.8902
	FSOINet (ICASSP 2022)	21.73	0.5937	26.37	0.8119	30.44	0.9018	35.80	0.9595	37.00	0.9665	39.14	0.9764	41.08	0.9832	33.08	0.8847
	TransCS (TIP 2022)	20.15	0.5066	25.41	0.7883	29.54	0.8877	35.06	0.9548	35.62	0.9588	38.46	0.9737	40.49	0.9815	32.10	0.8645
	AutoBCS (TCYB 2023)	19.63	0.5605	24.73	0.7871	28.44	0.8827	33.56	0.9481	34.48	0.9549	36.51	0.9680	36.88	0.9732	30.60	0.8678
	CSformer (TIP 2023)	21.86	0.6071	26.41	0.8058	30.09	0.8925	34.99	0.9534	-	-	-	-	40.23	0.9802	-	-
	DPC-DUN (TIP 2023)	18.03	0.4601	24.38	0.7498	29.42	0.8801	34.75	0.9483	35.88	0.9570	37.98	0.9694	39.84	0.9778	31.47	0.8489
	MAPUN (IJCV 2023)	-	-	-	-	30.19	0.9014	35.73	0.9602	37.08	0.9676	39.22	0.9775	40.98	0.9834	-	-
	OCTUF (CVPR 2023)	21.75	0.5934	26.45	0.8126	30.70	0.9030	36.10	0.9604	37.21	0.9673	39.41	0.9773	41.34	0.9838	33.28	0.8854
	TCS-Net (TCI 2023)	21.09	0.5505	25.46	0.7863	29.04	0.8834	33.94	0.9508	34.34	0.9541	35.27	0.9601	37.36	0.9723	30.93	0.8654
	CSRNet (TETCI 2023)	21.40	0.5771	26.57	0.8076	29.00	0.8733	32.55	0.9293	35.21	0.9556	37.49	0.9699	39.60	0.9790	31.69	0.8703
	SODAS-Net (TIM 2023)	-	-	-	-	28.89	0.8669	34.24	0.944	35.55	0.9543	37.74	0.968	39.60	0.9769	-	-
	MTC-CSNet (TCYB 2024)	21.38	0.5664	24.45	0.7813	29.40	0.8905	34.17	0.9506	-	-	-	-	-	-	-	-
	LTwIST (TCSTV 2024)	20.98	0.5469	25.62	0.7867	29.84	0.8930	35.00	0.9546	36.22	0.9627	38.31	0.9730	40.18	0.9802	32.31	0.8710
	NesTD-Net (TIP 2024)	21.40	0.5891	26.73	0.8238	30.91	0.9099	36.27	0.9622	37.17	0.9677	39.47	0.9778	41.55	0.9847	33.36	0.8879
	UFC-Net (CVPR 2024)	21.24	0.5607	25.92	0.7943	30.15	0.8960	35.42	0.9567	36.58	0.9641	38.68	0.9746	40.66	0.9818	32.66	0.8755
USB-Net (Our Method)		22.29	0.6168	27.26	0.8335	31.31	0.9149	36.42	0.9632	37.65	0.9699	39.64	0.9785	41.47	0.9843	33.72	0.8944

C. Additional Experiments on CS-MRI

1) **Experimental Settings:** In our CS-MRI experiments, we explore two types of sub-sampling masks: Pseudo Radial masks and 1D Cartesian masks. In the context of the Pseudo Radial masks, our training and test data is sourced from the Brain dataset [29] and the Cardiac dataset [30], which consist of 100 and 3000 training MRI images, as well as 50 and

300 test MRI images, respectively. All images share size of 256×256 . For the 1D Cartesian masks, we utilize training data from the single-coil knee dataset available within the fastMRI dataset [31]. Our approach involves extracting the central 5 slices from the h5 files of the training data and cropping them to size of 320×320 , amassing a total of 4865 slices from the single-coil knee dataset for training purposes.

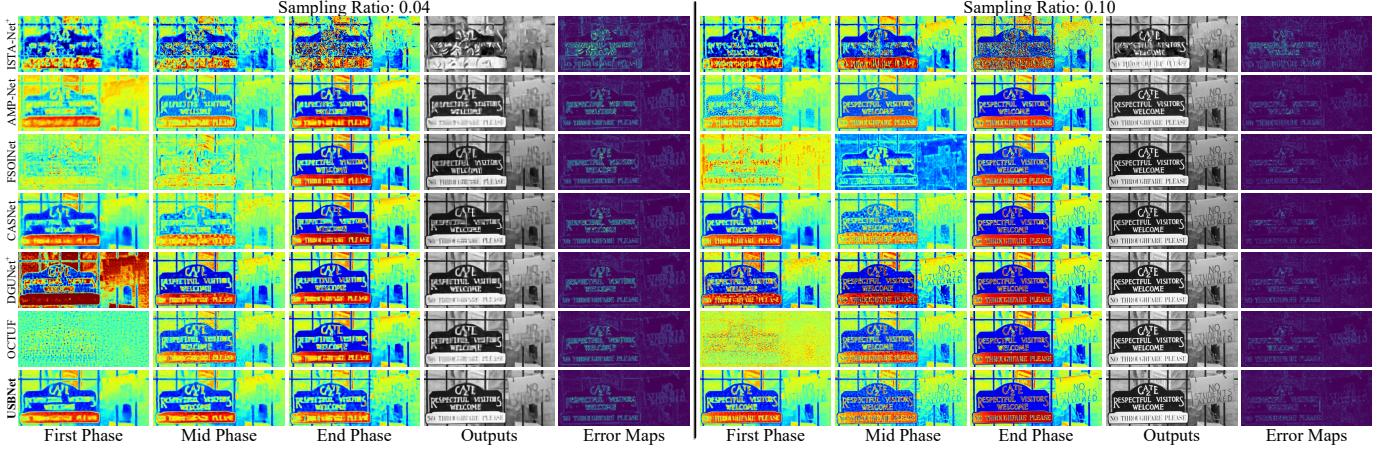


Fig. 3. Feature maps visualization of different phases in various methods at sampling ratios of 0.04 and 0.10.

TABLE II

AVERAGE PSNR (dB) AND SSIM SCORES COMPARISONS AMONG VARIOUS METHODS ON GRayscale DATASETS URBAN100, McM18, SET14 AND GENERAL100 WITH THREE DIFFERENT SAMPLING RATIOS $\tau \in \{0.30, 0.40, 0.50\}$.

Datasets	Sampling Ratios	CASNet (TIP 2022)		DGUNet ⁺ (CVPR 2022)		FSOINet (ICASSP 2022)		OCTUF (CVPR 2023)		USB-Net (Our Method)	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Urban100	0.30	33.35	0.9509	33.16	0.9510	33.84	0.9540	34.21	0.9555	34.79	0.9601
	0.40	35.46	0.9666	35.24	0.9666	35.93	0.9688	36.25	0.9669	36.72	0.9723
	0.50	37.46	0.9773	37.65	0.9785	37.80	0.9777	38.29	0.9797	38.45	0.9804
	Avg.	35.42	0.9650	35.35	0.9654	35.86	0.9668	36.25	0.9674	36.66	0.9710
McM18	0.30	39.07	0.9733	39.01	0.9737	39.19	0.9739	39.27	0.9740	39.54	0.9753
	0.40	41.42	0.9828	41.46	0.9835	41.56	0.9835	41.65	0.9836	41.92	0.9843
	0.50	43.60	0.9888	43.62	0.9888	43.74	0.9893	43.91	0.9895	44.13	0.9899
	Avg.	41.36	0.9816	41.36	0.9820	41.50	0.9822	41.61	0.9824	41.86	0.9832
Set14	0.30	35.12	0.9432	34.69	0.9412	35.18	0.9433	35.29	0.9436	35.58	0.9461
	0.40	37.23	0.9603	36.78	0.9590	37.27	0.9606	37.42	0.9608	37.60	0.9621
	0.50	39.23	0.9715	38.83	0.9709	39.13	0.9713	39.40	0.9720	39.43	0.9726
	Avg.	37.19	0.9583	36.77	0.9570	37.20	0.9584	37.37	0.9588	37.54	0.9603
General100	0.30	39.32	0.9730	38.87	0.9724	39.40	0.9735	39.54	0.9740	39.70	0.9751
	0.40	41.56	0.9827	41.07	0.9821	41.62	0.9831	41.76	0.9833	42.00	0.9841
	0.50	43.74	0.9887	43.26	0.9884	43.69	0.9884	43.95	0.9892	44.12	0.9896
	Avg.	41.54	0.9815	41.07	0.9810	41.57	0.9818	41.75	0.9822	41.94	0.9826

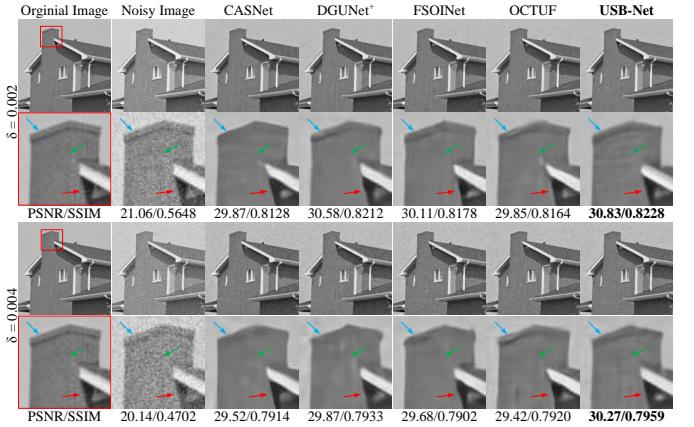


Fig. 4. Visual comparisons of reconstructed images among different methods under varied Gaussian noise levels (variance $\sigma \in \{0.002, 0.004\}$) at a sampling ratio of 0.04.

We utilize evaluation data from the single-coil knee dataset for test, subjecting it to the same preprocessing steps as the training data. This results in a total of 995 test slices. During the training phase, the initial learning rate is set to 5×10^{-5} ; the batch size is set to 1, and we fine-tune the model for 100 epochs based on the well-trained ICS model at the sampling ratio of 0.25. The remaining settings follow those mentioned in Appendix B.1.

2) *Additional Comparisons:* Tab. IV shows the results of USB-Net and other CS-MRI methods (including RDN [32], ISTA-Net⁺ [14], CDDN [33], ADMM-CSNet [29], HiTDUN [34], PUERT [35], MAPUN [23] and LTwIST [25]) on the Cardiac dataset [30] using Pseudo Radial masks.

Moreover, Fig. 5 and Fig. 6 provides visual comparisons of the reconstructed MRI images produced by USB-Net and other DL-based CS-MRI methods. It becomes evident that methods such as ISTA-Net⁺, ADMM-CSNet, PUERT and HiTDUN tend to introduce blurring and lose image details, resulting in poor visual perception, *i.e.*, pronounced detail degradation and distortion in reconstructed images at low sampling ratios. In contrast, USB-Net's reconstructed images stand out for their exceptional preservation of image details, resulting in

TABLE III

AVERAGE PSNR (dB) AND SSIM SCORES COMPARISONS AMONG VARIOUS ICS METHODS ON GRayscale DATASET SET11 WITH DIFFERENT GAUSSIAN NOISE OF MULTI-LEVELS VARIANCES $\delta \in \{0.001, 0.002, 0.004, 0.008\}$ AT SAMPLING RATIOS $\tau \in \{0.01, 0.04, 0.10, 0.25\}$.

δ	Sampling Ratios	CASNet (TIP 2022)		DGUNet ⁺ (CVPR 2022)		FSOINet (ICASSP 2022)		OCTUF (CVPR 2023)		USB-Net (Our Method)	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
0.001	0.01	21.71	0.6016	22.13	0.6095	21.70	0.5922	21.73	0.5917	22.22	0.6136
	0.04	26.09	0.8030	26.59	0.8119	26.13	0.8002	26.19	0.8017	26.95	0.8203
	0.10	29.34	0.8683	29.82	0.8741	29.41	0.8679	29.67	0.8697	30.15	0.8793
	0.25	32.02	0.8806	32.07	0.8800	31.92	0.8786	32.13	0.8806	32.30	0.8821
	Avg.	27.29	0.7884	27.65	0.7939	27.29	0.7847	27.43	0.7859	27.90	0.7988
0.002	0.01	21.71	0.6016	22.09	0.6080	21.67	0.5901	21.69	0.5902	22.19	0.6111
	0.04	25.90	0.7924	26.35	0.8009	25.92	0.7889	26.02	0.7921	26.71	0.8097
	0.10	28.71	0.8410	29.09	0.8456	28.70	0.8394	29.01	0.8427	29.38	0.8486
	0.25	30.26	0.8235	30.21	0.8220	30.15	0.8207	30.33	0.8232	30.44	0.8237
	Avg.	26.65	0.7644	26.94	0.7691	26.61	0.7598	26.76	0.7621	27.18	0.7733
0.004	0.01	21.66	0.5982	22.02	0.6046	21.61	0.5866	21.66	0.5877	22.05	0.6066
	0.04	25.61	0.7751	25.97	0.7801	25.62	0.7699	25.74	0.7741	26.31	0.7881
	0.10	27.78	0.7952	28.05	0.7982	27.73	0.7924	28.00	0.7957	28.31	0.8016
	0.25	28.09	0.7419	27.95	0.7360	27.92	0.7372	28.16	0.7412	28.25	0.7417
	Avg.	25.79	0.7276	26.00	0.7297	25.72	0.7216	25.89	0.7247	26.23	0.7345
0.008	0.01	21.59	0.5932	21.97	0.5992	21.54	0.5819	21.59	0.5817	21.99	0.5996
	0.04	25.08	0.7448	25.46	0.7490	25.10	0.7375	25.21	0.7431	25.77	0.7570
	0.10	26.41	0.7264	26.60	0.7267	26.37	0.7211	26.61	0.7254	26.88	0.7302
	0.25	25.62	0.6408	25.44	0.6342	25.46	0.6355	25.67	0.6400	25.75	0.6410
	Avg.	24.68	0.6763	24.87	0.6773	24.62	0.6690	24.77	0.6725	25.10	0.6819

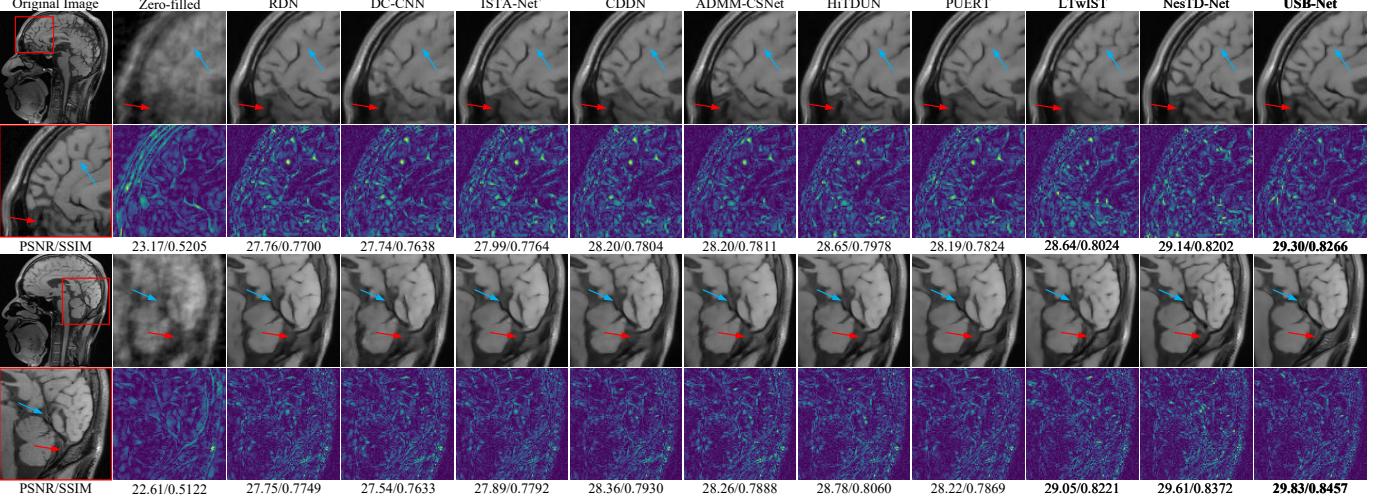


Fig. 5. Visual comparisons of PSNR (dB)/SSIM scores of MRI images reconstructed by various CS-MRI methods on the Brain dataset at a sampling ratio of 0.10. Local images are zoomed in, and corresponding error maps are presented for better comparison and analysis. Our proposed USB-Net demonstrates better visualization and excellent preservation of MRI image details.

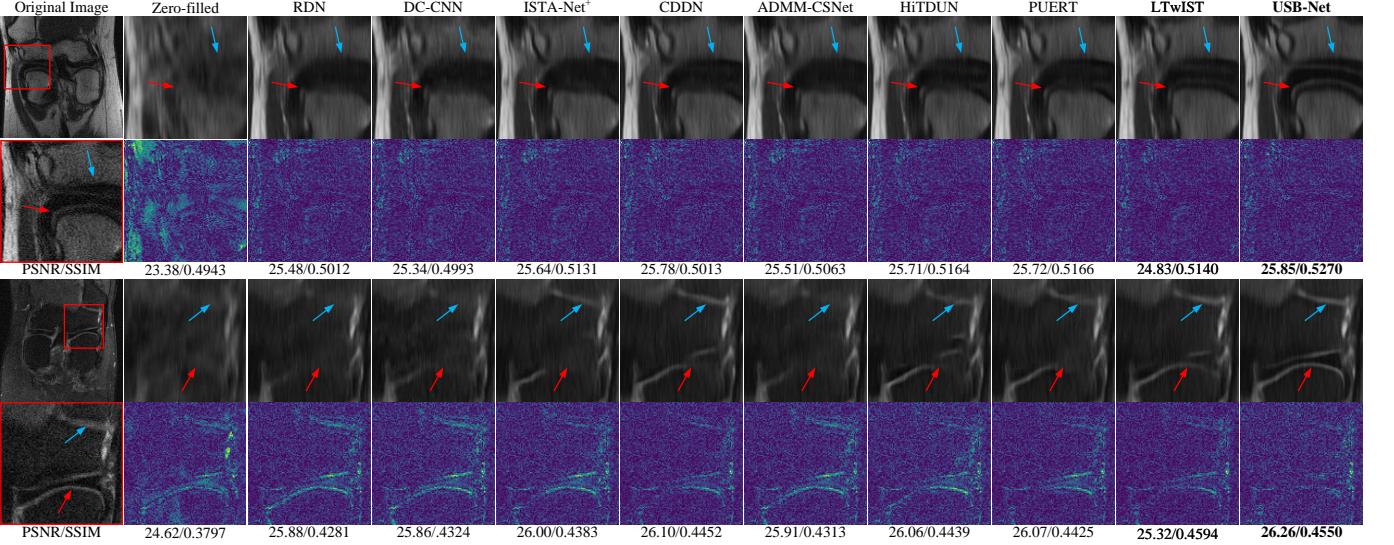


Fig. 6. Visual comparisons of PSNR (dB)/SSIM scores of magnetic resonance images reconstructed by various CS-MRI methods on the fastMRI dataset at a sampling ratio of 0.05. Local images are zoomed in, and corresponding error maps are presented for better comparison and analysis. Our proposed USB-Net demonstrates better visualization and excellent preservation of MRI image details.

significantly improved visual quality and overall perception. This visual evidence underscores USB-Net’s remarkable ability to efficiently extract, retain, and reconstruct images with exceptional fidelity in the context of CS-MRI tasks.

3) *Experiments on Complex-Value MRI Data:* To enhance USB-Net’s capability in handling complex-valued data, we introduce USB-Net-Complex. Specifically, the Depthwise Attention Blocks are adapted to serve as encoders and decoders for the real and imaginary components of the complex values. These components are then concatenated along the channel dimension, ensuring consistency throughout the iterative reconstruction process. This design maintains USB-Net’s original architectural strengths while extending its functionality to effectively process complex-valued MRI data. As illustrated in Tab. V, USB-Net-Complex demonstrates robust performance

across multiple datasets, underscoring its ability to reconstruct complex-valued MRI data with high fidelity.

D. Additional Experiments on SCI

1) *Experimental Settings:* Our experimentation encompasses both simulated and real datasets. Following established methodologies [36]–[44], we select 28 wavelengths ranging from 450nm to 650nm, employing spectral interpolation manipulation to derive the HSI. For simulation experiments, the CAVE dataset [45], containing 32 hyperspectral images (HSIs) with a spatial size of 512×512, serves as the training set, while the test set comprises 10 scenes from KAIST dataset [46]. In experiments involving real data, we introduce 11-bit shot noise into measurements during training to replicate noise disturbances typical in real-world imaging scenes. The real

TABLE IV
AVERAGE PSNR (dB) AND SSIM SCORES COMPARISONS AMONG
VARIOUS CS-MRI METHODS ON CARDIAC DATASET AT THREE
DIFFERENT SAMPLING RATIOS $\tau \in \{0.05, 0.10, 0.30\}$ USING PSEUDO
RADIAL MASKS.

Methods	Cardiac Dataset							
	0.05		0.10		0.30		Avg.	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Zero-filled	20.72	0.4753	24.43	0.5455	31.57	0.7574	25.57	0.5927
RDN (AAAI 2018)	27.77	0.8014	31.82	0.8863	40.77	0.9764	33.45	0.8880
ISTA-Net ⁺ (CVPR 2018)	28.52	0.8190	32.77	0.9049	41.64	0.9805	34.31	0.9015
CDDN (NeurIPS 2019)	28.62	0.8207	32.70	0.9038	41.56	0.9802	34.29	0.9016
ADMM-CSNet (TPAMI 2020)	28.21	0.7974	32.45	0.8890	41.42	0.9792	34.03	0.8885
HITDUN (J-STSP 2022)	29.32	0.8508	33.43	0.9192	42.81	0.9824	35.19	0.9175
PUERT (J-STSP 2022)	28.79	0.8156	33.50	0.9107	43.11	0.9832	35.13	0.9032
MAPUN (IJCV 2023)	28.75	0.8459	33.42	0.9242	41.76	0.9824	34.64	0.9175
LTwiST (TCSVT 2024)	27.93	0.7985	31.65	0.8815	37.26	0.9530	32.28	0.8777
USB-Net (Our Method)	30.64	0.8716	34.35	0.9242	43.56	0.9844	36.18	0.9267

TABLE V
AVERAGE PSNR (dB) AND SSIM SCORES OF USB-NET-COMPLEX ON
DIFFERENT DATASETS AT A SAMPLING RATIO OF 0.10.

Dataset	Brain dataset	fastMRI dataset
PSNR	35.73	28.86
SSIM	0.9222	0.6474

CASSI system [38], featuring 5 real scenes, is employed for testing. As for USB-Net, we change the basic channel number from 1 to 28 to fit in the number of wavelengths. Throughout the training phase, the initial learning rate is set to 1×10^{-4} with a batch size of 2, and training is conducted for 300 epochs. The remaining settings align with those detailed in Appendix B1.

2) *Additional Comparisons*: Visual comparisons in Fig. 7 on Scene 2 with 4 spectral channels demonstrate USB-Net's superior preservation of accurate details and fewer artifacts compared to other methods across different spectral channels. Besides, the spectral density curves in the bottom left of Fig. 7, corresponding to the red box areas in the RGB image, underscore the highest correlation and coincidence of USB-Net's spectral curves with the reference curves, emphasizing the advantages of our proposed USB-Net in HSI reconstruction.

Furthermore, we conduct comparisons with other methods on the real HSI data. Fig. 8 showcases visual comparisons of different methods on Scene 4 and Scene 5 with 2 spectral channels of real HSI data. Notably, our USB-Net achieves more accurate HSI reconstruction with fewer artifacts, less blurring, and distortion, demonstrating increased robustness to noise in real scenes compared to other methods.

E. Comparisons with Other Advanced Modules

Recently, many deep unfolding networks incorporate Transformer or Mamba architectures, reflecting the growing trend of leveraging advanced structures to improve performance. However, our proposed USB-Net, based on the unfolding Split Bregman framework, primarily adopts convolutional modules due to their computational efficiency and simplicity. Importantly, USB-Net's modular design allows seamless integration of Transformer and Mamba-based components, enabling the development of various USB-Net variants. Below, we provide a more detailed analysis.

1) *Transformer-based Methods*: Several recent works have utilized Transformers in deep unfolding frameworks or deep learning models for compressive sensing, such as TransCS [21], CSformer [10] and OCTUF [24]. TransCS is built on the ISTA-Net framework [14], this method employs Transformer Encoders and Decoders as sparse-inducing components within the unfolding ISTA framework. CSformer is a purely Transformer-based model combining Transformer Blocks with CNN modules to map measurements to images. Besides, OCTUF enhances the unfolding ISTA framework by leveraging a Cross-Attention mechanism to capture inter-iteration dependencies effectively. These Transformer-based methods leverage the powerful global modeling capabilities of Transformers to achieve high performance in image recovery. However, Transformer-based methods always suffer from the high complexity (e.g. computationally expensive and memory-intensive) due to computation of attention mechanism.

2) *Mamba-based Methods*: Mamba [47] has garnered increasing attention due to its ability to model state-space dynamics. Some notable Mamba-based approaches. VmambaSCI [48] combines Mamba's State Space Module (SSM) with the unfolding ISTA framework to reconstruct images. DHM [49] integrates Mamba's SSM with U-Net structures for enhanced image reconstruction. These approaches exploit Mamba's ability to efficiently handle long-range dependencies with linear complexity. However, Mamba-based methods typically demand more data and a long time for training.

3) *Distinctive Features and Advantages of USB-Net*: The differences between USB-Net and these advance architectures lie in several aspects:

- **Unfolding Split Bregman Framework**: USB-Net is grounded in the unfolding Split Bregman algorithm that can be applied for a wide range of Compressive Imaging tasks. The advantages of this approach, such as reduced information loss and improved performance, are demonstrated in our ablation studies.
- **Simplicity and Efficiency of Convolutional Structure**: USB-Net utilizes a straightforward architecture based on convolutional modules, specifically the Depthwise Attention Block and Iteration Fusion Module, to harmonize and integrate insights across iterative reconstruction phases. This design not only maintains simplicity and efficiency at the structural level but also effectively highlights the strengths of the unfolding Split Bregman framework in iterative reconstruction.
- **Extensibility**: The key elements of the unfolding Split Bregman algorithm in USB-Net, represented by $\mathcal{C}^{(k)}$ and $\tilde{\mathcal{C}}^{(k)}$ in Eq. (26), can be easily replaced or enhanced with more advanced components, such as Mamba's State Space Modules or Transformer Blocks in Fig. 9. This flexibility allows USB-Net to evolve into various enhanced variants.

Therefore, the benefits of USB-Net lie in its novel unfolding framework, simple and efficient architecture, and high extensibility. We also view integrating Transformer or Mamba Block into USB-Net without increasing too much complexity as a promising directions for future research.

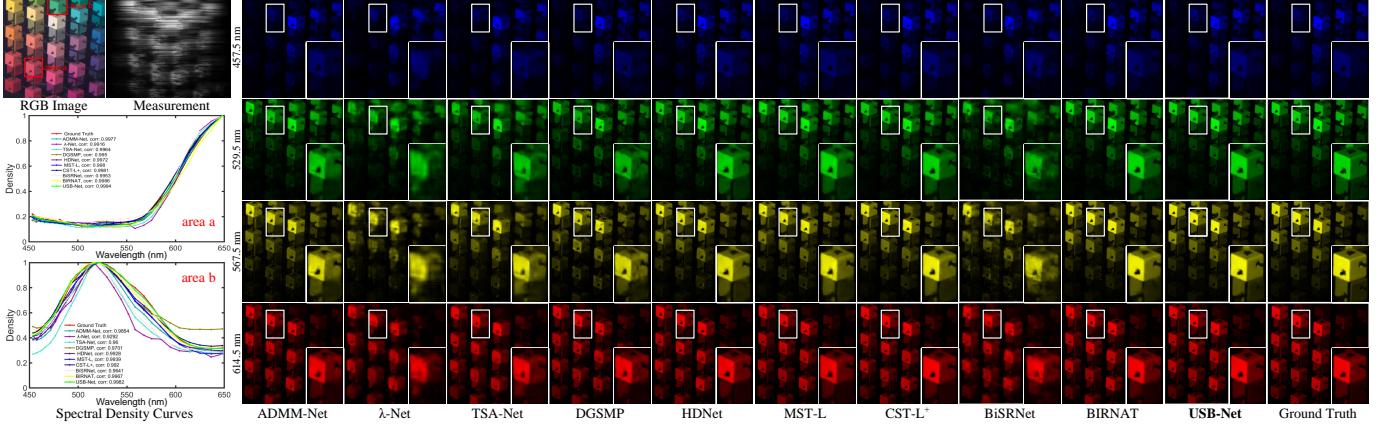


Fig. 7. Simulation HSI reconstruction comparisons of our USB-Net and different methods on Scene 2 with 4 (out of 28) spectral channels. Local images are zoomed in for better comparison and analysis.

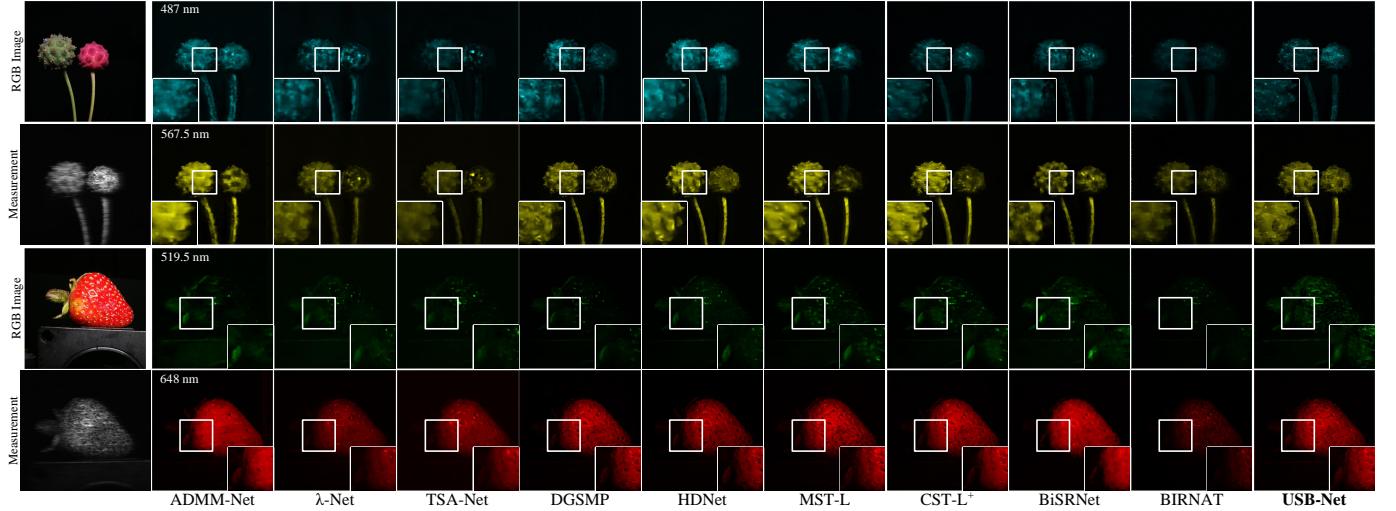


Fig. 8. Real HSI reconstruction comparisons of our USB-Net and different methods on Scene 4 and Scene 5 with 2 (out of 28) spectral channels. Local images are zoomed in for better comparison and analysis.

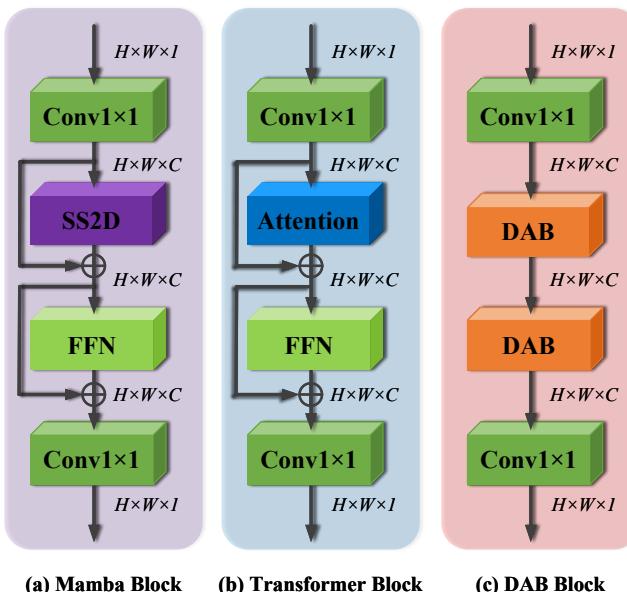


Fig. 9. Other modules that can be integrated to our proposed USB-Net.

REFERENCES

- [1] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.
- [2] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, “ReconNet: Non-iterative reconstruction of images from compressively sensed measurements,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 449–458.
- [3] W. Shi, F. Jiang, S. Liu, and D. Zhao, “Scalable convolutional neural network for image compressed sensing,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 12 290–12 299.
- [4] ———, “Image compressed sensing using convolutional neural network,” *IEEE Trans. Image Process.*, vol. 29, pp. 375–388, 2020.
- [5] Y. Sun, J. Chen, Q. Liu, B. Liu, and G. Guo, “Dual-path attention network for compressed sensing image reconstruction,” *IEEE Trans. Image Process.*, vol. 29, pp. 9482–9495, 2020.
- [6] J. Chen, Y. Sun, Q. Liu, and R. Huang, “Learning memory augmented cascading network for compressed sensing of images,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 513–529.
- [7] S. Zhou, Y. He, Y. Liu, C. Li, and J. Zhang, “Multi-channel deep networks for block-based image compressive sensing,” *IEEE Trans. Multimedia*, vol. 23, pp. 2627–2640, 2021.

- [8] W. Cui, S. Liu, F. Jiang, and D. Zhao, "Image compressed sensing using non-local neural network," *IEEE Trans. Multimedia*, vol. 25, pp. 816–830, Dec. 2021.
- [9] H. Gan, Y. Gao, C. Liu, H. Chen, T. Zhang, and F. Liu, "AutoBCS: Block-based image compressive sensing with data-driven acquisition and noniterative reconstruction," *IEEE Trans. Cybern.*, vol. 53, no. 4, pp. 2558–2571, Apr. 2023.
- [10] D. Ye, Z. Ni, H. Wang, J. Zhang, S. Wang, and S. Kwong, "CSformer: Bridging convolution and transformer for compressive sensing," *IEEE Trans. Image Process.*, vol. 32, pp. 2827–2842, 2023.
- [11] H. Gan, M. Shen, Y. Hua, C. Ma, and T. Zhang, "From patch to pixel: A transformer-based hierarchical framework for compressive image sensing," *IEEE Trans. Comput. Imaging*, vol. 9, pp. 133–146, 2023.
- [12] Y. Zhou, Y. Chen, X. Zhang, P. Lai, L. Huang, and J. Jiang, "A lightweight recurrent learning network for sustainable compressed sensing," *IEEE Trans. Emerg. Topics Comput. Intell.*, pp. 1–14, 2023.
- [13] M. Shen, H. Gan, C. Ma, C. Ning, H. Li, and F. Liu, "MTC-CSNet: Marrying transformer and convolution for image compressed sensing," *IEEE Trans. Cybern.*, pp. 1–13, 2024.
- [14] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1828–1837.
- [15] J. Zhang, C. Zhao, and W. Gao, "Optimization-inspired compact deep compressive sensing," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 4, pp. 765–774, May 2020.
- [16] Z. Zhang, Y. Liu, J. Liu, F. Wen, and C. Zhu, "AMP-Net: Denoising-based deep unfolding for compressive image sensing," *IEEE Trans. Image Process.*, vol. 30, pp. 1487–1500, 2021.
- [17] D. You, J. Zhang, J. Xie, B. Chen, and S. Ma, "COAST: Controllable arbitrary-sampling network for compressive sensing," *IEEE Trans. Image Process.*, vol. 30, pp. 6066–6080, 2021.
- [18] B. Chen and J. Zhang, "Content-aware scalable deep compressed sensing," *IEEE Trans. Image Process.*, vol. 31, pp. 5412–5426, 2022.
- [19] C. Mou, Q. Wang, and J. Zhang, "Deep generalized unfolding networks for image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17378–17389.
- [20] W. Chen, C. Yang, and X. Yang, "FSOINET: Feature-space optimization-inspired network for image compressive sensing," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2022, pp. 2460–2464.
- [21] M. Shen, H. Gan, C. Ning, Y. Hua, and T. Zhang, "TransCS: A transformer-based hybrid architecture for image compressed sensing," *IEEE Trans. Image Process.*, vol. 31, pp. 6991–7005, 2022.
- [22] J. Song, B. Chen, and J. Zhang, "Dynamic path-controllable deep unfolding network for compressive sensing," *IEEE Trans. Image Process.*, vol. 32, pp. 2202–2214, 2023.
- [23] ———, "Deep memory-augmented proximal unrolling network for compressive sensing," *Int. J. Comput. Vis.*, vol. 131, no. 6, pp. 1477–1496, Jun. 2023.
- [24] J. Song, C. Mou, S. Wang, S. Ma, and J. Zhang, "Optimization-inspired cross-attention transformer for compressive sensing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Apr. 2023.
- [25] H. Gan, X. Wang, L. He, and J. Liu, "Learned two-step iterative shrinkage thresholding algorithm for deep compressive sensing," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, 2023.
- [26] J. Song and J. Zhang, "SODAS-Net: Side-information-aided deep adaptive shrinkage network for compressive sensing," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–12, 2023.
- [27] X. Wang and H. Gan, "UFC-Net: Unrolling fixed-point continuous network for deep compressive sensing," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2024, pp. 25149–25159.
- [28] H. Gan, Z. Guo, and F. Liu, "NesTD-Net: Deep nester-optimized unfolding network with dual-path deblocking structure for image compressive sensing," *IEEE Trans. Image Process.*, vol. 33, pp. 1923–1937, 2024.
- [29] Y. Yang, J. Sun, H. Li, and Z. Xu, "ADMM-CSNet: A deep learning approach for image compressive sensing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 3, pp. 521–538, Mar. 2020.
- [30] A. Andreopoulos and J. K. Tsotsos, "Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI," *Med. Image Anal.*, vol. 12, no. 3, pp. 335–357, Jun. 2008.
- [31] F. Knoll, J. Zbontar, A. Sriram, M. J. Muckley, M. Bruno, A. Defazio, M. Parente, K. J. Geras, J. Katsnelson, H. Chandarana, Z. Zhang, M. Drozdzal, A. Romero, M. Rabbat, P. Vincent, J. Pinkerton, D. Wang, N. Yakubova, E. Owens, C. L. Zitnick, M. P. Recht, D. K. Sodickson, and Y. W. Lui, "fastMRI: A publicly available raw k -space and dicom dataset of knee images for accelerated MR image reconstruction using machine learning," *Radiology: Artif. Intell.*, vol. 2, no. 1, p. e190007, Jan. 2020.
- [32] L. Sun, Z. Fan, Y. Huang, X. Ding, and J. Paisley, "Compressed sensing MRI using a recursive dilated network," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 32, no. 1, Apr. 2018.
- [33] H. Zheng, F. Fang, and G. Zhang, "Cascaded dilated dense network with two-step data consistency for MRI reconstruction," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, 2019.
- [34] J. Zhang, Z. Zhang, J. Xie, and Y. Zhang, "High-throughput deep unfolding network for compressive sensing MRI," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 4, pp. 750–761, Jun. 2022.
- [35] J. Xie, J. Zhang, Y. Zhang, and X. Ji, "PUERT: Probabilistic undersampling and explicable reconstruction network for CS-MRI," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 4, pp. 737–749, Jun. 2022.
- [36] J. Ma, X.-Y. Liu, Z. Shou, and X. Yuan, "Deep tensor admim-net for snapshot compressive imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 10223–10232.
- [37] X. Miao, X. Yuan, Y. Pu, and V. Athitsos, "Lambda-Net: Reconstruct hyperspectral images from a snapshot measurement," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4058–4068.
- [38] Z. Meng, J. Ma, and X. Yuan, "End-to-end low cost compressive spectral imaging with spatial-spectral self-attention," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, ser. Lecture Notes in Computer Science, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., 2020, pp. 187–204.
- [39] T. Huang, W. Dong, X. Yuan, J. Wu, and G. Shi, "Deep gaussian scale mixture prior for spectral compressive imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 16216–16225.
- [40] X. Hu, Y. Cai, J. Lin, H. Wang, X. Yuan, Y. Zhang, R. Timofte, and L. Van Gool, "HDNet: High-resolution dual-domain learning for spectral compressive imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 17542–17551.
- [41] Y. Cai, J. Lin, X. Hu, H. Wang, X. Yuan, Y. Zhang, R. Timofte, and L. Van Gool, "Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 17502–17511.
- [42] ———, "Coarse-to-Fine Sparse Transformer for Hyperspectral Image Reconstruction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, ser. Lecture Notes in Computer Science, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., 2022, pp. 686–704.
- [43] Y. Cai, Y. Zheng, J. Lin, X. Yuan, Y. Zhang, and H. Wang, "Binarized spectral compressive imaging," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Oct. 2023, pp. 1–12.
- [44] Z. Cheng, B. Chen, R. Lu, Z. Wang, H. Zhang, Z. Meng, and X. Yuan, "Recurrent neural networks for snapshot compressive imaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 2264–2281, Feb. 2023.
- [45] J.-I. Park, M.-H. Lee, M. D. Grossberg, and S. K. Nayar, "Multispectral imaging using multiplexed illumination," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.
- [46] I. Choi, M. H. Kim, D. Gutierrez, D. S. Jeon, and G. Nam, "High-quality hyperspectral reconstruction using a spectral prior," in *ACM Trans. Graph.*, no. ART-2017-104309, 2017.
- [47] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, and Y. Liu, "Vmamba: Visual state space model," May 2024.
- [48] M. Zhang, L. Li, W. Shi, J. Guo, Y. Li, and X. Gao, "Vmambasci: Dynamic deep unfolding network with mamba for compressive spectral imaging," in *Proceedings of the 32nd ACM International Conference on Multimedia*, ser. MM '24. Association for Computing Machinery, Oct. 2024, pp. 6549–6558.
- [49] J. Dong, H. Yin, H. Li, W. Li, Y. Zhang, S. Khan, and F. S. Khan, "Dual hyperspectral mamba for efficient spectral compressive imaging," Jun. 2024.