A New Backbone for Hyperspectral Image Construction and Improvement based on Mask Mixture Training and Energy Normalization

1st Yukai Song
Department of Electrical Engineering
Columbia University
New York, USA
ys3493@columbia.edu

2nd Zixuan Yan
Department of Electrical Engineering
Columbia University
New York, USA
zy2501@columbia.edu

3rd Jiawei Lu
Department of Electrical Engineering
name of organization (of Aff.)
New York, USA
jl5999@columbia.edu

***Abstract*—The study of 3D hyperspectral image (HSI) reconstruction refers to the inverse process of snapshot compressive imaging when the optical system, e.g., the coded aperture snapshot spectral imaging (CASSI) system, captures the 3D spatial-spectral signal and encodes it to a 2D measurement. Although numerous neural networks have been built for end-to-end reconstruction, those previous works have a hard time balancing among performance, efficiency (training and inference time), and feasibility (the ability to restore high-resolution HSI on limited GPU memory). In this work, we try to solve this challenge by creatively proposing SSI-ResU-Net, Spatial/Spectral Invariant Residual U-Net. The SSI-ResU-Net makes several modifications compared to U-Net: scale/spectral-invariant learning, and nested residual learning. Benefiting from these updates, the SSI-ResU-Net can achieve a trade-off between performance, efficiency, and feasibility. Apart from that, mask mixture training and energy normalization are integrated into the process of generating measurements to increase the ability to generalize for SSI-ResU-Net. The dataset of this work can be found at https://www1.cs.columbia.edu/CAVE/databases/multispectral/.**

***Index Terms*—hyperspectral image reconstruction, coded aperture snapshot spectral imaging, snapshot compressive imaging, Spatial/Spectral Invariant Residual U-Net**

I. Introduction

Hyperspectral imaging defines as multi-channel imaging in which each channel stores information for a scene [11]. Hyperspectral imaging apples widely in medical image processing [12], remote sensing [13], and object detection [14], [15], which proves the importance of hyperspectral imaging.

Hyperspectral imaging can be compressed by snapshot compressive imaging systems and transformed into one 2D measurement. One of the most popular snapshot compressive imaging systems is the coded aperture snapshot spectral imaging (CASSI) system, which is the system that will discuss in detail in this paper.

A large number of algorithms to reconstruct the 2D measurements generated by CASSI have been proposed [3]. The output of those reconstruction algorithms is hyperspectral imaging that should be as similar to original hyperspectral imaging before compressing as possible. End-to-end deep neural networks have been proved to be an effective method to recover the original hyperspectral image and we utilize a deep neural network named SSI-ResU-net in this project.

A. Motivation of This Project

Although deep neural networks have achieved great success in HSI reconstruction, there are still many challenges brought by using deep neural networks. Firstly, due to the lack of a large hyperspectral imaging dataset, it is easy for complex deep learning models to become overfitting. Therefore, we need to simplify the deep neural networks to avoid overfitting. Secondly, U-Net [1] achieves a great result in hyperspectral imaging reconstruction but it is specially designed for biomedical image processing. Thirdly, other existing deep learning models, like TSA-Net [3], can achieve a great performance too. However, long training time (need several weeks to train TSA-Net) and inference time are needed for those models due to their extensive parameters. Next, multiple channels of higher resolution hyperspectral images can become a large burden in limited computational resources, which limits the deployments of deep neural networks under real conditions. Last but not least, the original SSI-ResU-net lacks generalizing ability when applied to the new model. We wanted to improve its robustness to help it can work better in the real industry

B. Contributions of This Project

To alleviate the above challenges, we firstly reimplemented a modified version of U-Net, SSI-ResU-Net [5]. The result of reimplementation, which we named as specific training, got a close result compared with the original paper, indicating the success of our work.

In the original paper, the SSI-ResU-Net achieves a state-of-the-art result in hyperspectral imaging through nested residual learning, spatial/spectral invariance, and computation minimization (SSI-ResU-Net uses 2.82 percent parameters of TSA-Net with only less than 2 days of training). However, this great result is only applicable if we use the same mask as the mask used in the training, meaning that the original SSI-ResU-Net lacks robustness. To tackle this issue, we implemented mask mixture training and energy normalization [16]. Those

two changes have been proved to successfully improve the robustness of the SSI-ResU-Net. Besides, a combination of those two modifications can further improve the generalizing ability of SSI-ResU-Net.

II. Literature Review

The basic idea of snapshot compressive imaging (SCI) is to build a compressive imaging system where multiple frames are mapped into one single measurement. One representative application is hyperspectral compressive imaging, which is mapping a hyperspectral image with hundreds of spectrum channels to a compressed image representation with only one channel. In this manner, a compressed representation will include the information in the high-dimensional signal, reducing the complexity of storing and transmitting these high-dimensional images. Also, a high-performance algorithm is needed to recover the desired data. As a novel implementation of SCI, CASSI uses a coded aperture and a prism to implement the spectral modulation and achieve wonderful results in compression ratio and reconstruction performance.

Inspired by the success that deep learning achieves in other image translation problems, researchers have been using deep learning to reconstruct hyperspectral images from CASSI representations [2], [3], [6]–[10], tending to directly learn a complete mapping function from measurements (always packaged with masks) to original HSIs. At the same time, some researchers manage to introduce CNN models into conventional optimization algorithms, leading to more lightweight and interpretable methods. Among all reasonable models, U-Net [1], a CNN originating from biomedical image segmentation, has been deemed as a reconstructive backbone and widely used. For example, the λ-net [2] is a dual-stage generative model which employs a U-Net as its main model structure. The TSA-Net [3], which combined spatial-spectral self-attention with U-Net led to excellent results on both simulation and real data. Recent Gaussian Scale Mixture Prior-based (GSM-based) baseline [4] employs two U-Net for different parts: a lightweight U-Net for approximating the regularization parameters, another lightweight U-Net for estimating the local-mean of GSM prior. The core idea behind the success of U-Net is that it combines low-resolution and high-resolution feature maps via multiple concatenation paths and thus perfectly matched the big challenges with medical images.

However, for HSI reconstruction, U-Net only achieves sub-optimal performance when being solely referred to as a baseline. Researchers thus generally wrap U-Net into larger models and make efforts out of U-net, devoting less attention inside. At the same time, the performance of neural networks is sensitive to minor adjustments, there existing a "variant" of U-Net that enables a significant performance boost especially on HSI reconstruction is a reasonable assumption. Thus, we build our project on top of a recently proposed variant of U-Net, which is called SSI-ResU-net [5]. This variant first substitutes improper partitions with reconstruction-oriented components to acquire better reconstructive performance and then further cut off inefficient modules and take actions to reduce FLOPs in the model to ultimately improve the efficiency and the feasibility of the proposed model. We will give a detailed illustration of this model in section III-B.

III. Methodology

A. Mathematical Model of CASSI

The hyperspectral image can be expressed as a 3D spectral cube $F \in R^{N_x \times N_y \times N_\lambda}$. N_x, N_y, and N_λ denote the height of the image, the width of the image, and the number of wavelengths respectively. M^* is a pre-defined physical mask that is used for computing signal Modulation. 2D measurement, essentially a compressed frame of hyperspectral image, can be represented as $Y \in R^{N_x \times (N_y + N_\lambda - 1)}$ while the noise of the 2D measurement is represented as $G \in R^{N_x \times (N_y + N_\lambda - 1)}$.

To calculate the measurement Y, we need firstly to shift signal frames and masks of various wavelengths like the following formula:

$$M(u, v, n_\lambda) = M^*(x, y + d(\lambda_n - \lambda_c)), \tag{1}$$

$$F_1(u, v, n_\lambda) = F(x, y + d(\lambda_n - \lambda_c)) \tag{2}$$

Based on shifted $M \in R^{N_x \times N_y \times N_\lambda}$ and $F_1 \in R^{N_x \times N_y \times N_\lambda}$, the measurement Y can be expressed as:

$$Y = \sum_{n_\lambda}^{N_\lambda} F_1(:, :, n_\lambda) \odot M(:, :, n_\lambda) + G \tag{3}$$

$\odot$ means the Hadamard product.

In an attempt to accelerate the computation in the computer, we firstly vectorized the aboved expressions by assigning $y = vector(Y) \in R^n$, $g = vector(G) \in R^n$ and $f_1^(n_\lambda) = vector(F_1(:, :, n_\lambda))$ to be the vectorization of Y, G and $F_1(:, :, n_\lambda)$, in which vector(.) concatenates all columns of a matrix and $n = N_x(N_y + N_\lambda - 1)$. Then, $f_1^(1), f_1^(2), ..., f_1^(n_\lambda, ..., f_1^(N_\lambda)$ can be concatenated to form $f = vector([f_1^(1), f_1^(2), ..., f_1^(n_\lambda, ..., f_1^(N_\lambda])$. Next, we defined the sensing matrix as

$$\phi = [D_1, D_2, ..., D_{n_\lambda}, ..., D_{N_\lambda}] \tag{4}$$

$D_{n_\lambda} = Diag(vector(M(:, :, n_\lambda)))$ is a diagonal matrix expanded by $vector(M(:, :, n_\lambda$.

With the above preparation, we can finally expression vectorization of matrices Y as:

$$y = \phi f + g \tag{5}$$

After obtaining the measurement y and ϕ depending upon predesign of the camera, we can use the deep learning neural network to find f to recover the compressed measurement.

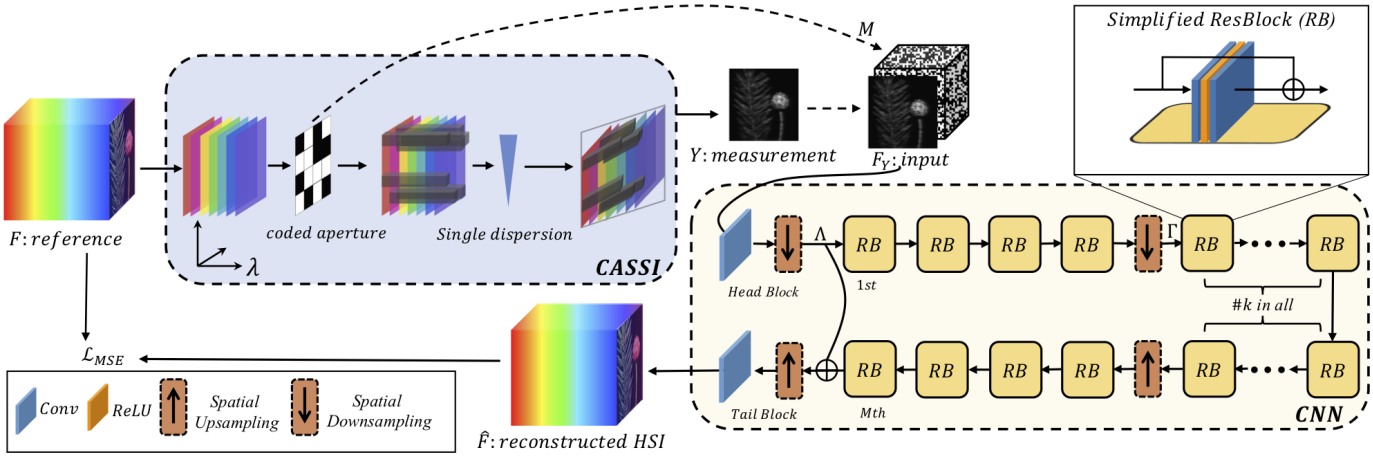

Fig. 1. Flow chart of snapshot compressive imaging (by CASSI) and the inverse process. Our reconstruction net is shown in the bottom right yellow box with all Spatial Upsampling and Spatial Downsampling units, SSI-ResU-Net (v3) in [5]. We adapted this version becuase it requires the least computation resource.

B. Network Structure

After the compression discussed above, we will have $Y \in R^{N_x \times (N_y + N_\lambda - 1)}$ as the 2-D compressed representation of the hyperspectral image and $M \in R^{N_x \times (N_y + N_\lambda - 1) \times N_\lambda}$ as shifted mask. These two information are combined to construct the model input $F_Y \in R^{N_x \times N_y \times N_\lambda}$ by defining

$$\boldsymbol{F}_Y [:, :, n_\lambda] := shift\left(\boldsymbol{M}_{n_\lambda} \odot \boldsymbol{Y}\right) \tag{6}$$

Here $\odot$ is Hadamard product, and $shift$ means the inverse process of the shifting of channels that are defined in (1). With this input, the proposed model aims to learn the mapping function as

$$f_{\text{res}}(\cdot) : \boldsymbol{F}_Y \rightarrow \widehat{\boldsymbol{F}} \tag{7}$$

Here $\widehat{\boldsymbol{F}}$ denotes the expected reconstruction result. Essentially, this mapping function is trying to approximate Φ^{-1} expressed in a vector space.

Overall, this SSI-ResU-Net shares the same symmetric U-shape with U-net but contains a huge difference. This network can be divided into three parts: 1) head block; 2) main body with residual connection and 3) tail block as shown in fig [ref]. The details of this network will be discussed below.

1) Nested Residual Network: The main body of this model is a stack of M identical Residual Blocks (RB) which originates from ResNet[ref]. Each RB contains a CONV-ReLU-CONV structure with identity connection, which can be formulized by $H_m(x) = CONV(ReLU(CONV(x))) + x$. We name this model as nested residual network because there are identity connection not only inside, but also outside RB. There is an extra identity connection from the start to the end of the body block. Therefore, the whole reconstruction model can be defined as

$$f_{\text{res}}\left(\boldsymbol{F}_Y\right) = f_{\text{tail}}\left(f_{body}\left(f_{\text{head}}\left(\boldsymbol{F}_Y\right)\right) + f_{\text{head}}\left(\boldsymbol{F}_Y\right)\right) \tag{8}$$

$$f_{body} = H_M\left(H_{M-1}\left(\ldots H_2\left(H_1\left(f_0\right)\right)\ldots\right)\right) \tag{9}$$

Pure residual learning releases the burden of the mapping function by reformulating the learning objective and combining feature maps with different resolutions. This nested residual learning will make resolutions more diverse, thus leading to better feature extraction.

2) Spatial/Spectral Invariance: In the original U-Net, one underlying reason for sub-optimal performance is its Maxpooling operations, which lead to a lossy compression regarding spatial information. It will discard pixels with lower intensities and also miss location information of pixels of max values. Therefore, this network turns to creating the mapping function (DCNN model) in a spatial invariant fashion.

To approximate the 3D signal in a rank minimization manner, previous works attempted to reduce the spectral channels in the model, which lead to a lossy compression and is avoided here. The original U-Net tends to enlarge the spectral dimensions when reducing the spatial size and do vice versa, which works well for the input image with a single or few channels. However, it is improper for the reconstructive inputs that are already packaged with respective large channels (i.e., in this work we use 28 channels). Therefore, in the proposed model, we expand the spectral channel from $N_\lambda = 28$ to $N_\gamma = 64$ in the head part and keep it the same through the body part(spectral invariant). The underlying intention of this augmentation is to increase the spectral-wise redundancy of intermediate embeddings.

3) Training and Testing: The model is trained to minimize a mean squared error (MSE) between the ground truth and output. MSE loss is defined as following

$$\mathcal{L}_{MSE}(\Theta) = \frac{1}{N} \sum_{n=1}^{N} \left\| \widehat{\boldsymbol{F}}_n - \boldsymbol{F}_n \right\|^2 \tag{10}$$

Because of the time and computational resources limitation, we only do experiments on simulation data instead of both simulation and real-world data in the original paper. For simulation experiments, synthetic hyperspectral images are

put through the CASSI system for reconstructing-aimed input initialization. Therefore, the 3D cube naturally becomes the ground truth. Training and testing data are separately abstracted from different datasets. The result of our reproduction on simulation data is similar to the original work produced by them.

C. Improvements Other than Original Paper

1) Mask Mixture: Mask Mixture Training is a method that we choose a stochastic mask from the dataset when training our SSI-ResU-Net.

The reference paper assumes that we know the ground truth mask in the coded aperture snapshot spectral imaging (CASSI) system. They used the same mask in the progress of training and testing to obtain a good result. However, this assumption is too idealistic in reality, and it is very likely that we do not know what the ground truth value of the mask is within the CASSI system. If we use only one mask to train the SSI-ResU-Net, the results will obviously become relatively poor when we perform tests on the image generated by another mask.

We propose the Mask Mixture Training method to solve this problem. It breaks into the following 3 steps.

- Build a mask dataset M contains N masks from different CASSI system, i.e. initialize $M = \{m_1, m_2, \cdots, m_N\}$.
- Choose a stochastic mask m_i from dataset M, i.e. randomly select $m_i \in M$.
- Train the SSI-ResU-Net with mask m_i.

The experiment shows that the Mask Mixture Training method will improve the robustness of training results, narrowing the gap among highest PSNR and lowest PSNR in experiments.

2) Energy Normalization: Energy normalization is a technique used in video snapshot compressive imaging [16]. This modification can gather energy from every pixel of a video, bringing more motionless information of video frames, which proves to be very successful.

Motivated by this idea, we made our mind to bring energy normalization in our hyperspectral snapshot compressive imaging. The first step of energy normalization is to sum all shifted masks into one energy normalization matrix.

$$M^{'} = \sum_{n_\lambda=1}^{N_\lambda} M(:,:,n_\lambda) \tag{11}$$

Each element in $M^{'}$ describes how many channels of $F_1(u,v,n_\lambda)$ are integrated into the 2D measurement Y. After that , we normalize the 2D measurement Y by $M^{'}$ to acquire the energy normalization measurement Y_{en} as:

$$Y_{en} = Y \oslash M^{'} \tag{12}$$

$\oslash$ represents element-wise division. It is obvious that Y_{en} can have more information about different channels compared with Y. From another perspective, Y_{en} can be treated as an average of weighted $(M(:,:,n_\lambda))$ summation of images in different channel $(F_1(u,v,n_\lambda))$, containing more information in different channels.

After we gained the energy normalization measurement Y_{en}, we took the concatenation as our input:

$$E = [Y_{en}, Y_{en} \odot M(:,:,1), ..., Y_{en} \odot M(:,:,N_\lambda)]_3 \tag{13}$$

$[\cdot]_3$ means the concatenation along the 3^{rd} dimension. It should be noticed that compared with the original input $F_Y \epsilon R^{N_x \times N_y \times N_\lambda}$, the dimension of new input E is $R^{N_x \times N_y \times (N_\lambda + 1)}$.

IV. Experiments

In this part, what we mainly focused on is to make a comparison among SSI-ResU-Net specific mask training, SSI-ResU-Net mask mixture training, and SSI-ResU-Net energy normalization training based on PSNR and SSIM.

A. Experimental Settings

The hyperspectral image dataset that we used is as same as the one used in [3], which ranged from 450nm to 650nm. The experiment was conducted in the simulation data.

CAVE [16] and KAIST [17] synthetic datasets were applied in our experiment. In terms of the training set, 205 $1024 \times 1024 \times 28$ image instances were created from 30 $256 \times 256 \times 28$ images in the CAVE dataset through randomly concatenating. Data augmentation strategies like rotation and re-scaling were used to make the training set more robust. In the end, the training set is composed of 205 $256 \times 256 \times 28$ images after randomly concatenating, re-scaling, and rotating. Ten $256 \times 256 \times 28$ images from the KAIST dataset constitute the testing set, which will be used to test the training model's performance.

We compared the performance of SSI-ResU-Net specific mask training, SSI-ResU-Net mask mixture training, and SSI-ResU-Net energy normalization training based on PSNR and SSIM by two commonly-used criteria, Peak Signal-to-Noise Racial (PSNR) and Structural Similarity (SSIM). The PSNR can be calculated by:

$$PSNR_{ch} = 10log_{10}(\frac{MAX_I^2}{MSE_{ch}}) \tag{14}$$

$PSNR_{ch}$ represents the channel-wise PSNR. After computing PSNR of every channel, we take the average. MAX_I^2 is the maximum pixel value in ground truth image I while MSE_{ch} is the mean square error of each channel.

4 different pre-defined physical masks are used in this experiment for SSI-ResU-Net specific mask training, SSI-ResU-Net mask mixture training, and SSI-ResU-Net energy normalization training. It is important to notice that we did not get the energy normalization mask by summing up these 4 masks. Instead, we firstly shifted them to 28 channels in the way $M(u,v,n_\lambda) = M^*(x, y+d(\lambda_n - \lambda_c))$ and then added the 28 channel masks to obtain the energy normalization mask.

Based on Tensorflow, we built the SSI-ResU-Net and adapted Adam as our optimizer. 16 simplified ResBlocks were put in the main body. Initially, the learning rate was set as 4×10^{-4} and the learning rate halved every 50 epochs. The batch size was set to 4 for our experiment and we trained 200 epochs with Nvidia Tesla T4 GPU for around 15 hours.

B. Experiments on Synthetic Data

Due to the limit of time and computation resources, we only did the experiment on the synthetic data. There are four masks provided by the authors of the original paper and mask 1 is what they used in their original work. Specific training on mask 1 and test on mask 1 corresponds to their original works, which can examine whether our reimplementation is successful or not. The testing results can be found through the following tables:

TABLE I
SPECIFIC TRAINING ON MASK 1

Train	*Test*	*PSNR*	*SSIM*
Train mask 1	Test mask 1	**31.20**	**0.890**
Train mask 1	Test mask 2	28.36	0.823
Train mask 1	Test mask 3	28.06	0.821
Train mask 1	Test mask 4	27.77	0.815

TABLE II
ENERGY NORMALIZATION TRAINING ON MASK 1

Train	*Test*	*PSNR*	*SSIM*
Train mask 1	Test mask 1	**30.55**	**0.878**
Train mask 1	Test mask 2	28.50	0.824
Train mask 1	Test mask 3	28.48	0.826
Train mask 1	Test mask 4	28.59	0.826

TABLE III
MASK MIXTURE TRAINING

Train	*Test*	*PSNR*	*SSIM*
Train mixture	Test mask 1	30.44	0.880
Train mixture	Test mask 2	30.44	0.879
Train mixture	Test mask 3	29.91	0.873
Train mixture	Test mask 4	**31.11**	**0.886**

The overstriking number and underling number represent the largest number and the smallest number in the column respectively. The authors of the original paper only completed the specific training on mask 1 and tested on mask 1, which is the first row of the table, specific training on mask 1. What they got was PSNR 31.36dB while our PSNR is 31.20dB after our reproduction. It only has a small gap with the original work, proving the success of our reimplementation work.

From the three tables above, We can easily conclude that mask specific training can achieve the largest PSNR and SSIM in the test that used the mask for training while it has a large drop of 3.42 dB in PSNR and 0.075 in SSIM compared with the lowest PSNR and the lowest SSIM respectively; energy normalization has a lower PSNR when used the mask for training to test but it can experience a lower drop in PSNR (2.07 dB) and SSIM (0.052) compared with the lowest PSNR and the lowest SSIM without using other masks during training; mask mix training has the strongest robustness because it adapts the information of other masks during training.

Inspired by the better performance of mask mix training and energy normalization in terms of generalization, we decided to add another experiment to test if the combination of mask mix training and energy normalization can further improve the robustness of our model.

TABLE IV
ENERGY NORMALIZATION AND MASK MIXTURE TRAINING

Train	*Test*	*PSNR*	*SSIM*
Train mask 1	Test mask 1	30.14	0.867
Train mask 1	Test mask 2	**30.48**	**0.873**
Train mask 1	Test mask 3	30.20	0.871
Train mask 1	Test mask 4	30.08	0.866

The result demonstrates that the training model by both energy normalization and mask mixture has only a drop of 0.4 dB in PSNR and 0.007 in SSIM when compared the highest and the lowest PSNR and SSIM respectively, the smallest drop among 4 training models!

Since in the real world, we usually cannot obtain the same mask that we use in training to test the trained model, it is meaningful to adapt energy normalization and mask mixture together during training to increase the generalizing ability of the trained model.

In an attempt to help readers better understand the whole process, we visualized the ground truth, energy normalization reconstructed grayscale images, specific training reconstructed grayscale images, mixed training reconstructed grayscale images, and energy normalization and mixed training reconstructed grayscale images of scene 1 in wavelength=450nm and wavelength=650nm, shown in the following two figures.

Fig. 2. Comparison of Ground Truth, Specific, Energy Normaliztion, Mixture, Energy Normalization and Mixture at wavelength=450nm

Fig. 3. Comparison of Ground Truth, Specific, Energy Normaliztion, Mixture, Energy Normalization and Mixture at wavelength=650nm

V. CONCLUSION

We reviewed the reference paper and explain the simple yet highly efficient method specially designed for hyperspectral imaging reconstruction in detail. We built the SSI-ResU-Net (v3, needing the least computation resource), then trained it, and finally reproduced a closed result compared to the original work. What's more, we proposed two methods, Mask Mixture Training and Energy Normalization, to improve the robustness of our model when a different mask that is not used during the training, is applied. Last but not least, we combine those

two modifications and their cooperation can further increase the generalizing ability of our trained SSI-ResU-Net.

REFERENCES

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[2] X. Miao, X. Yuan, Y. Pu, and V. Athitsos, "l-net: Reconstruct hyperspectral images from a snapshot measurement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4059–4069.

[3] Z. Meng, J. Ma, and X. Yuan, "End-to-end low cost compressive spectral imaging with spatial-spectral self-attention," in *European Conference on Computer Vision*. Springer, 2020, pp. 187–204.

[4] T. Huang, W. Dong, X. Yuan, J. Wu, and G. Shi, "Deep gaussian scale mixture prior for spectral compressive imaging," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 216–16 225.

[5] J. Wang, Y. Zhang, X. Yuan, Y. Fu, and Z. Tao, "A new backbone for hyperspectral image reconstruction," *arXiv preprint arXiv:2108.07739*, 2021.

[6] Z. Meng, M. Qiao, J. Ma, Z. Yu, K. Xu, and X. Yuan, "Snapshot multispectral endomicroscopy," *Optics Letters*, vol. 45, no. 14, pp. 3897–3900, 2020.

[7] L. Wang, C. Sun, Y. Fu, M. H. Kim, and H. Huang, "Hyperspectral image reconstruction using a deep spatial-spectral prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8032–8041.

[8] L. Wang, C. Sun, M. Zhang, Y. Fu, and H. Huang, "Dnu: deep non-local unrolling for computational spectral imaging," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1661–1671.

[9] L. Wang, T. Zhang, Y. Fu, and H. Huang, "Hyperreconnet: Joint coded aperture optimization and image reconstruction for compressive hyperspectral imaging," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2257–2270, 2018.

[10] S. Zheng, Y. Liu, Z. Meng, M. Qiao, Z. Tong, X. Yang, S. Han, and X. Yuan, "Deep plug-and-play priors for spectral snapshot compressive imaging," *Photonics Research*, vol. 9, no. 2, pp. B18–B29, 2021.

[11] P. Benediktsson, J. A. Boardman, J. W. Brazile, J. Bruzzone, L. Camps-Valls, G. , and G. Trianni ." Recent advances in techniques for hyperspectral image processing," in *Remote sensing of environment* , 113, pp. 110-122.

[12] G. Lu, and B. Fei." Medical hyperspectral imaging: a review," in *Journal of biomedical optics* , 2014, 19(1), 010901.

[13] M. Borengasser, W.S. Hungate, and R. Watkins." Hyperspectral remote sensing: principles and applications," in *CRC press* , 2007.

[14] M.H. Kim, T.A. Harvey, D.S. Rushmeier, H. Rushmeier, J. Dorsey, R.O. Prum, and D.J. Brady." 3D imaging spectroscopy for measuring hyperspectral patterns on solid objects," in *ACM Transactions on Graphics (TOG)* , 2012, 31(4), pp.1-11.

[15] Y. Xu, Z. Wu, Z. Li, J. Plaza, A., and Z. Wei." Anomaly detection in hyperspectral images based on low-rank and sparse representation," in *IEEE Transactions on Geoscience and Remote Sensing* , 2015, 54(4) pp. 1990-2000.

[16] Z. Cheng, R. Lu, Z. Wang, H. Zhang, B. Meng, and X. Yuan." BIRNAT: Bidirectional recurrent neural networks with adversarial training for video snapshot compressive imaging," in *European Conference on Computer Vision* , 2015, pp. 258-275.

[17] I. Choi, M.H. Kim, D. Gutierrez, D.S. Jeon, and G. Nam." High-quality hyperspectral reconstruction using a spectral prior ," in *Technical report*, 2017. 8.

CONTRIBUTION OF EACH MEMBER

In this work, all three members of the team have done their best. Specifically, Yukai Song analyzed the mathematical model CASSI as well as designed the code to realize the model in the python code and wrote the code for energy normalization training and testing; Zixuan Yan was in charge of constructing the SSI-ResU-Net and reimplemented the training and testing in the original paper; Jiawei Lu worked for mask mixture training and testing. Besides, Yukai Song and Jiawei Lu combined their work to finish the code for the training and testing for mask mixture and energy normalization training. The following table can show more details in perspectives of code and report.

TABLE V
CONTRIBUTION OF EACH MEMBER

	Code	Project Report
Yukai Song	1. Fulfill Mathematical Model-of CASSI 2. Energy Normalization 3. Combine Mask Mixture and Energy Normalization	Abstract, I. Introduction III.A. Mathematical Model of CASSI III.C(2)Energy Normalization IV.Experiments
Zixuan Yan	1. Construct the SSI-ResU-Net 2. Reimplement the training and testing	II. Literature Review III.B. Network Structure
Jiawei Lu	1. Mask Mixture 2. Combine Mask Mixture and Energy Normalization	III.C (1) Mask Mixture IV. Conclusion