COMPUTATIONAL HYPERSPECTRAL IMAGING WITH DIFFRACTIVE OPTICS AND DEEP RESIDUAL NETWORK

Ayoung Kim, Ugur Akpinar, Erdem Sahin, Atanas Gotchev
Faculty of Information Technology and Communication Sciences,
Tampere University, Finland

***Abstract*—Hyperspectral imaging critically serves for various fields such as remote sensing, biomedical and agriculture. Its potential can be exploited to a greater extent when combined with deep learning methods, which improve the reconstructed hyperspectral image quality and reduce the processing time. In this paper, we propose a novel snapshot hyperspectral imaging system using optimized diffractive optical element and color filter along with the residual dense network. We evaluate our method through simulations considering the effects of each optical element and noise. Simulation results demonstrate high-quality hyperspectral image reconstruction capabilities through the proposed computational hyperspectral camera.**

***Index Terms*—Hyper-spectral Imaging, Computational Imaging, Deep Learning, Residual Learning**

1. INTRODUCTION

Hyperspectral imaging (HSI) has attracted much attention over the last few decades because it has shown tremendous improvements in image-related applications from remote sensing to various applications [1]. A hyperspectral image is in the form of data cube consisting of a two-dimensional spatial domain and a one-dimensional spectral domain. Since the sensors in conventional cameras are two-dimensional, additional processes or optical elements are required to capture a hyperspectral data cube.

The traditional approach is the scanning method, which scans along one axis, e.g., line scanning, point scanning, or area scanning. Accumulating measures can restore them to cube form, but it takes a long time to acquire. Snapshot imaging methods instead can obtain an entire data cube within a single detector integration period via an additional optical element [2]. Such methods rely on acquiring a multiplexed image of the data cube, and solving an ill-posed reconstruction problem adopting techniques from inverse imaging to obtain the original hyperspectral data.

Many state-of-the-art methods utilize coded aperture snapshot spectral imaging (CASSI) [3] as a camera model, along with various reconstruction methods, from iterative optimization to deep learning methods. We group the existing methods in three different categories. The first group of methods adopt a combination of CASSI and an iterative optimization method [4]–[6]. Iterative methods are theoretical approaches that can alleviate ill-posedness. They are very flexible methods, but they require exhaustive parameter tuning and heavy computations. The second group of methods use a combination of CASSI and a deep learning method [7], [8]. Deep learning approaches present significant improvement in spatial-spectral image quality and computation time compared to the iterative optimization methods. The last group of methods additionally propose to replace the CASSI system with a snapshot camera based on novel optical elements such as diffractive optical elements (DOEs) [9], [10]. DOE offers a significantly more compact solution as it can replace several optical elements of CASSI including the coded aperture, relay optics, and dispersive elements, enabling arbitrary modifications to the system point spread function (PSF). Moreover, the design of the DOE can be incorporated into the deep learning framework via a fully differentiable simulation model leading to an optimal solution, a paradigm which has been utilized extensively in recent works, see e.g. [11].

In this paper, we propose an end-to-end computational hyperspectral imaging system using DOE-based snapshot imaging followed by residual dense network (RDN) [12]. The suggested camera model includes a refractive lens, a DOE, and a color filter on top of the sensor, where the latter two are jointly learnt with the RDN based on existing HS image datasets. The hybrid refractive-diffractive optics relieves the learning task in terms of DOE, which enables larger imaging apertures, yet it remains to have compact form-factor.

The rest of the paper is organized as follows. In Section 2, we explain the proposed method in more detail. In Section 3, we evaluate our method through simulation results. Lastly, the conclusions are in Section 4.

2. PROPOSED METHOD

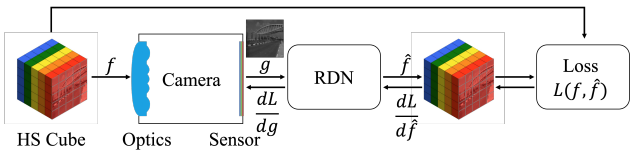

Figure 1. Schematic diagram of the proposed method.

The overview of the proposed method is illustrated in Fig. 1. It consists of a differentiable, wave optics-based simulation module to mimic the sensor image formation in the presence of HS data, as well as an inverse imaging stage to reconstruct the HS cube back from the sensor measurements. The camera model includes a refractive lens, a DOE, and a color filter on top of the sensor. The reconstruction model is based on the residual dense network

(RDN) [13]. In the forward pass of each training instance, the input HS data, f, is first converted to the sensor image, g, via the camera model, which is then given into RDN to output the predicted HS data, $\hat{f}$. The network output is compared with the input HS data via a loss function. The network, the color filter and the DOE parameters are then updated by using backpropagation starting from the loss and applying gradient descent. Thus, we jointly optimize the parameters of the camera, specifically the DOE and the sensor color filter, and the reconstruction network in an end-to-end manner. Once the training is finalized, the camera simulation model is replaced with the actual, physical implementation of the HS camera. The reconstruction network then takes as an input the sensor image obtained by the camera to solve for the HSI of the scene. In the following, we present each stage in details.

2.1. Camera Model

The output sensor image $g(x, y)$ of the camera model is the 2D convolution of the HS data cube with the wavelength-dependent PSF, $p(x, y, \lambda)$, followed by the spectral modification via a color filter, $\kappa(x, y, \lambda)$, i.e.

$$g(x,y) = \iiint f(\eta,\xi,\lambda)p(x-\eta,y-\xi,\lambda)d\eta d\xi\kappa(x,y,\lambda)d\lambda, \tag{1}$$

where $f(x, y, \lambda)$ is the input hyperspectral image projected onto the camera sensor plane. In practice we have the discrete model:

$$\boldsymbol{g} = \sum_{\lambda}(\boldsymbol{f}_{\lambda} *_{[x,y]} \boldsymbol{p}_{\lambda}) \odot \boldsymbol{\kappa}_{\lambda} + \boldsymbol{e}, \tag{2}$$

where $*_{[x,y]}$ is the discrete 2D convolution operator over (x, y), $\odot$ is the Hadamard (element-wise) multiplication, $\boldsymbol{e}$ is the noise, and the bold symbols indicate matrix format. From here, we discuss $p(\xi, \eta, \lambda)$ and $\kappa(x, y, \lambda)$ through the camera model description based on the paraxial wave optics as illustrated in Fig. 2.

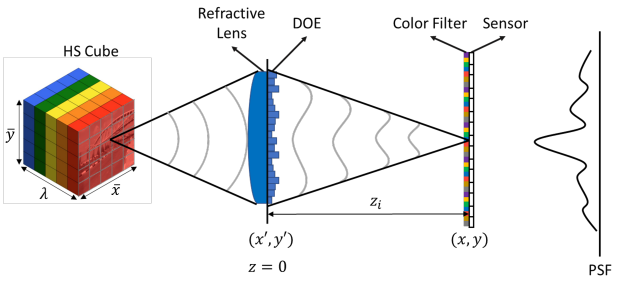

Figure 2. The proposed camera model.

Let us start with a monochromatic point source with wavelength λ, which is propagated to the lens plane (x', y'). Under the paraxial approximation, the wavefield just before the refractive lens is found as [14]

$$U_{\lambda}^{-}(x', y', z = 0) = \frac{1}{i\lambda z_f} \exp\left[\frac{ik}{2z_f}(x'^2 + y'^2)\right], \tag{3}$$

where $k = 2\pi/\lambda$ is the wave number, and z_f is the distance between the point source to the lens plane. After the refractive lens and the DOE, the wavefield is modified as

$$U_{\lambda}^{+}(x', y', z = 0) = U_{\lambda}^{-}(x', y', z_f)A(x', y') \exp\{i\phi_{\lambda}^{d}(x', y')\} \exp\{i\phi_{\lambda}^{l}(x', y')\}, \tag{4}$$

where $A(x', y')$ is the circular aperture function of the refractive lens and DOE, and $\phi_{\lambda}^{d}(x', y')$ and $\phi_{\lambda}^{l}(x', y')$ are the phase shift introduced by the DOE and the refractive lens at the wavelength λ, respectively. We consider the distance between the refractive lens and DOE as negligible.

The wavefield $U_{\lambda}^{+}(x', y', z_f)$ continues the propagation over a distance z_i to the sensor from the lens plane. The wavefield on the sensor plane is then found using Fresnel diffraction integral as

$$U_{\lambda}(x, y, z_i) = \frac{\exp(ikz_i)}{i\lambda z_i} \iint U_{\lambda}^{+}(x', y', 0) \exp\left[\frac{ik}{2z_i}\{(x - x')^2 + (y - y')^2\}\right] dx'dy'. \tag{5}$$

Finally, the incoherent PSF, $p(x, y, \lambda)$, is the intensity of $U_{\lambda}(x, y, z_i)$. After some algebraic manipulations to (5), the PSF can be expressed as

$$p(x, y, \lambda) \propto \left| \mathcal{F}\left\{ A(x', y') \exp(i\phi_{\lambda}^{d}(x', y')) \exp\left[\frac{ik}{2}\left(\frac{1}{z_f} + \frac{1}{z_i} - \frac{1}{f_{\lambda}}\right)(x'^2 + y'^2)\right]\right\}\right|^2, \tag{6}$$

where we use the thin lens model for $\phi_{\lambda}^{l}(x', y')$, and $\mathcal{F}$ denotes the Fourier transform. Note that this is a simplified expression after substituting $U_{\lambda}^{+}(x', y', 0)$ with (3) and (4) and omitting the constants. Assuming that the scene is located at infinity, i.e. $z_f = \infty$, we set $z_i = f_{\lambda_0}$, where λ_0 is the nominal (optimization) wavelength. If we further employ a refractive lens free from the chromatic aberration, i.e. $f_{\lambda} = f_{\lambda_0}$ for each λ, the second exponential term of (6) disappears, leaving only the wavelength-dependent phase shift caused by DOE inside the Fourier transform.

The phase delay by the DOE, $\phi_{\lambda}^{d}(x', y')$, is a learning parameter inside the training network. For a transparent diffractive element, the wavelength dependent phase delay is related to the height profile, $h(x', y')$, as

$$\phi_{\lambda}^{d}(x', y') = k\Delta n_{\lambda} h(x', y'), \tag{7}$$

where Δn_{λ} indicates the refractive index difference between air and the DOE material in wavelength, λ. Using such identity, we optimize for a nominal wavelength λ_0, from which the phase delay at an arbitrary wavelength λ is calculated as

$$\phi_{\lambda}^{d}(x', y') = \phi_{\lambda_0}^{d}(x', y')\frac{\lambda_0 \Delta n_{\lambda}}{\lambda \Delta n_{\lambda_0}}. \tag{8}$$

In addition to the DOE, we also employ a color filter that is placed on top of the sensor to further modify the hyperspectral datacube. The spectral response of the filter at each pixel location, $\kappa(x, y, \lambda)$, is set as an optimization parameter. In practice, we approximate the continuous spectral response function by optimizing for a discrete

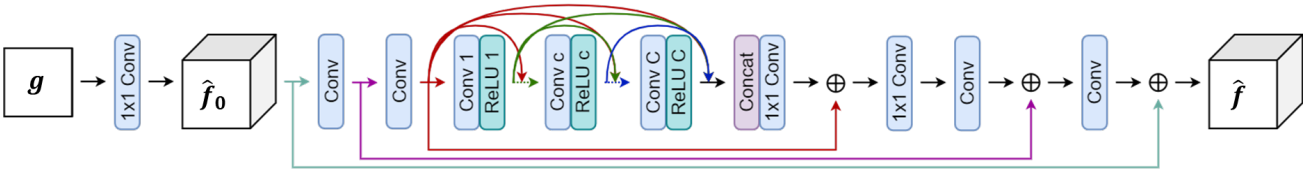

Figure 3. The utilized reconstruction network.

set of wavelength values. In addition, the color filter is assumed to be repeating with a period m, i.e.,

$$\kappa(x, y, \lambda) = \kappa(x + m\Delta x, y + m\Delta, y, \lambda). \quad (9)$$

The period size is another hyperparameter determined by the experimental approach described in Section 3. The design of the color filter involves the following steps: First, the color filter values within a period are initialized to uniformly distributed random numbers. Next, the values are updated with the sensor spectral function in Eq. 9. At each instance of the training, the color filter is regulated by rectifying linear unit (ReLU) to ensure a non-negative response. Lastly, it is normalized over the spectral dimension so that $\sum_\lambda \kappa(x, y, \lambda) = 1$.

2.2. Reconstruction Network

Reconstruction network recovers the hyperspectral image $\hat{\boldsymbol{f}} \in \mathbb{R}^{N_x \times N_y \times N_\lambda}$ from the sensor image $\boldsymbol{g} \in \mathbb{R}^{N_x \times N_y}$. The utilized reconstruction network is residual dense net (RDN) as illustrated in Fig. 3. It consists of four parts, two 2D convolutional layers of 3×3 spatial size; four dense connected layers with skip connections, i.e., $C = 4$, where each sub-block contains a convolutional layer of 3×3 spatial size with ReLU activation function; a concatenation layer for combining features at each layer followed by a 1×1 convolution; and after an additional 1×1 convolutional layer, two 3×3 convolutional layers with summations. The arrow symbols in Fig. 3 indicate skip connections. Prior to RDN, we first perform a so-called tentative reconstruction by applying a 1×1 convolution onto the sensor image $\boldsymbol{g}$. The output $\hat{\boldsymbol{f}}_0 \in \mathbb{R}^{N_x \times N_y \times N_\lambda}$ is then refined through RDN to get the final prediction $\hat{\boldsymbol{f}}$.

The residual learning scheme is widely used in image processing applications: image restoration [15]–[17], image classification [18], image denoising [13], [19], and super-resolution [20]. The skip-connection mechanism of residual learning allows for better feature extraction between layers and long-term memory effects through connections between the first and last layers [21]. The advantage of residual learning is to improve performance and computational efficiency while increasing network depth [22]. Moreover, as a deep learning network, the proposed network can utilize the benefits of data-driven priors that significantly improve reconstruction quality through joint feature extraction at spatial and spectral domains [1].

We use sparsity-inducing loss function, l_1 loss, expressed as

$$\|\boldsymbol{f} - \hat{\boldsymbol{f}}\|_1 = \sum_{x,y,\lambda} |\boldsymbol{f} - \hat{\boldsymbol{f}}|, \quad (10)$$

where $\hat{\boldsymbol{f}}$ is the reconstructed hyperspectral image. It has been demonstrated previously that l_1 loss outperforms the standard l_2 loss significantly in several image processing problems, especially in terms of reconstructing sharp image details [23].

3. SIMULATION RESULTS

We conduct simulations demonstrating the reconstruction results, which are further compared with state of the art methods, specifically, DeSCI [4] and Choi et al. [7]. These methods are selected to be representative methods from iterative optimization method and a deep learning methods, respectively. Those comparison methods are implemented through the source codes provided by the authors. Note that comparsion methods use the DD-CASSI [24] camera model. We also check the effect of each optimized optical element to analyze our method in detail. Lastly, we evaluate how well the proposed method works in the phase and sensor noise cases.

In the training stage, we use 1890 cropped image patches from the KAIST dataset: 256×256 spatial size and 31 spectral channels covering the wavelength range, [420 nm 720 nm], with a 10 nm interval. We train the network without phase and sensor noise. We set the learning rate to 0.0001 and the number of epochs to 100.

We assume a refractive lens with the effective focal length 35 mm at the specification wavelength 587.6 nm. The radius of curvature of lens is 16 mm, and the central thickness of lens is 2 mm. The wavelength-dependent refractive indices is based on silica material. We set the aperture radius as 5 mm, and DOE sampling rate as 30 µm. The color filter period is determined to be 9×9 by an experimental approach. We try the period size with three intervals from 3×3 to 15×15, while the sensor resolution is fixed as 6 µm, and 9×9 size shows the best result.

Figure 4. The test images from ICVL dataset used in the simulations.

For testing, we use ten different ICVL dataset images [25] in Fig. 4 and calculated the average PSNR and SSIM

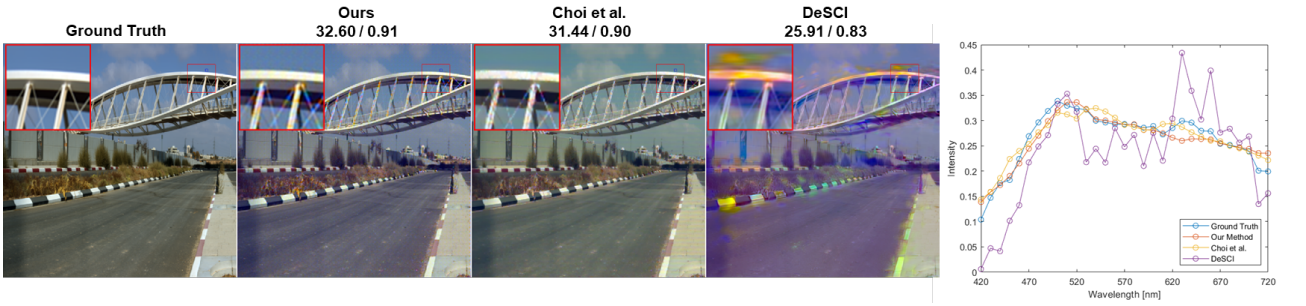

Figure 5. Comparison with the state-of-the-art methods.

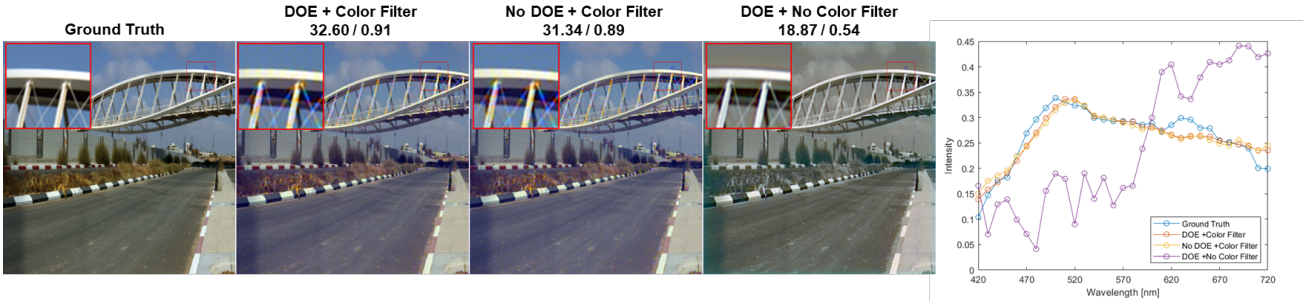

Figure 6. The effects of DOE and color filter.

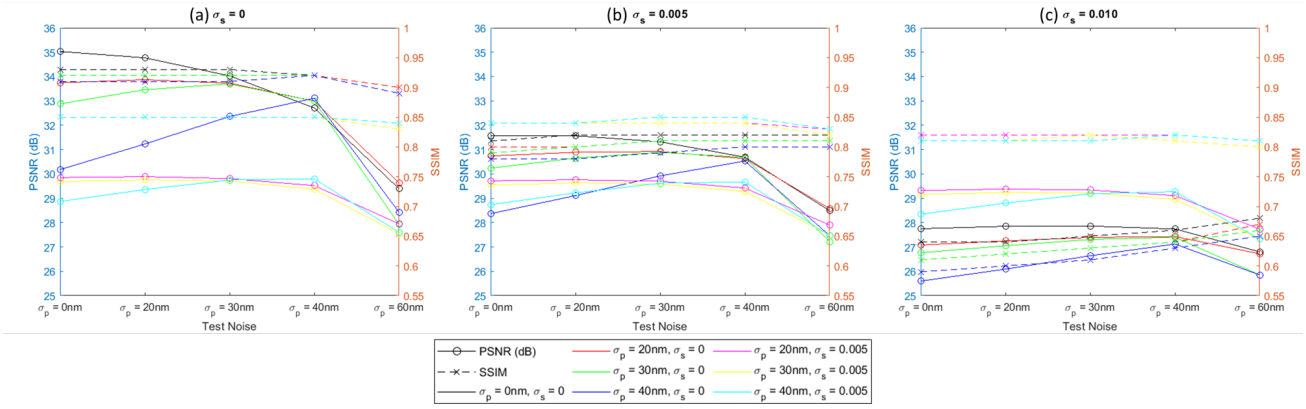

Figure 7. Average PSNR (dB) and SSIM of 10 ICVL test scenes at noise cases.

values. All test images use a spatial size of 512×512 and 31 spectral channels, which correspond to the wavelengths used in the training.

3.1. Comparison with the state-of-the-art methods

We compare our results with widely recognized state-of-the-art methods that are based on iterative optimization [4] and deep learning [7]. The results are provided in in Table 1 and Fig. 5. The PSNR and SSIM values of each test scene are in Table 1. In most of the images, our method shows the best result, and in a few cases, Choi et al [7] presents slightly better PSNR where the difference is less than 0.5dB. Scene 3 shows an exceptional result among these 10 ICVL images in favor of [4], which is possibly because of the low light illumination condition, as illustrated in the third figure of the upper row of Fig. 4. Fig. 5 visually illustrates an example result for scene 7.

TABLE 1. PSNR (dB) and SSIM of 10 ICVL test scenes reconstructed by different methods.

Method	Our Method	Choi et al. [7]	DeSCI [4]
Scene 1	**34.27 / 0.92**	32.39 / 0.88	26.90 / 0.79
Scene 2	**39.03 / 0.96**	37.59 / 0.95	28.96 / 0.86
Scene 3	32.99 / 0.97	33.43 / 0.98	**38.35 / 0.98**
Scene 4	**33.00 / 0.91**	30.44 / 0.89	25.24 / 0.81
Scene 5	**36.85 / 0.96**	33.46 / 0.92	26.51 / 0.83
Scene 6	**34.87 / 0.92**	30.76 / 0.90	26.15 / 0.83
Scene 7	**32.60 / 0.91**	31.44 / 0.90	25.91 / 0.83
Scene 8	**36.08 / 0.94**	34.00 / 0.93	28.72 / 0.84
Scene 9	33.29 / 0.90	**33.51 / 0.90**	30.65 / 0.89
Scene 10	**37.29 / 0.94**	35.96 / 0.94	29.52 / 0.84
Average	**35.03 / 0.93**	33.30 / 0.92	28.69 / 0.85

The cropped figures show the spatial accuracy in detail. The spectral accuracy can be found in the right-most plot representing intensity values of the specific pixels marked

with blue boxes. PSNR and SSIM values are calculated between hyperspectral images, while example results are presented as converted RGB images by weighting spectral channels of the hyperspectral images. As evident also from the PSNR and SSIM values, our method demonstrates superior performance compared to the state of the art in terms of reconstructed spatial and spectral information. In particular, it provides images with higher spatial details (less blur) and fewer artifacts, such as color shading (with respect of DeSCI [4]) and chromatic aberration (with respect of Choi et al. [7]). In general, the deep learning methods, i.e., the proposed method and Choi et al. [7], demonstrate good spatial-spectral resolution compared to the iterative one, DeSCI [4].

3.2. The effects of DOE and color filter

In this part, to evaluate the relative contributions of DOE and color filter to reconstruction results, we train different end-to-end systems by excluding each of the elements. That is, in the case there is no DOE we set the DOE height values to 0, and in the case there is no color filter we set all the corresponding pixels of the color filter to 1.Table 2 presents the average PSNR and SSIM values of 10 ICVL test scenes in 3 different combinations with presence or absence of the DOE and the color filter. Also, Fig. 6 illustrates how each element actually affects the reconstructed image. The right side of Fig. 6 compares the spectral values in the blue box on the left side of the figure of each combination. It is clear that the optimized color filter has more impact than the optimized DOE. Nevertheless, the contribution of DOE is non-negligible.

TABLE 2. Average PSNR (dB) and SSIM of 10 ICVL test scenes for cameras with different combinations of optical elements.

Camera	with DOE	without DOE
with Color Filter	35.03 / 0.93	33.94 / 0.92
without Color Filter	20.70 / 0.58	-

3.3. The effects of phase and sensor noises

The effects of possible errors in DOE fabrication (phase noise) and sensor image noise are quantified through noise modelling and analysis in the training and testing stages, respectively. Phase noise is added to the phase mask of the DOE, and sensor noise is added to the sensor image. Both noises are random Gaussian noises with standard deviations of σ_p for phase noise and σ_s for sensor noise, respectively. Fig. 7 presents the average PSNR (solid lines) and SSIM (dashed lines) values for 10 ICVL test images at various test noise settings. Each color corresponds different networks trained under different noise levels.

The noise-free training case shows noticeably better performance compared to other noise training cases in low noise conditions, where the phase noise and sensor noise are no more than $\sigma_p = 30\,\mathrm{nm}$ and $\sigma_s = 0.005$, respectively. However, in high noise environments, e.g., $\sigma_p = 60\,\mathrm{nm}$ and $\sigma_s = 0.010$, training with noise can demonstrate more robust results compared to noise-free training case.

4. CONCLUSION

We propose a snapshot computational hyperspectral imaging system that incorporates end-to-end learnt diffractive optics and color filter array in the camera and deep residual network in the postprocessing stage. Our method is demonstrated to advance the recent existing methods in terms of reconstructed spatio-spectral image quality.

As a future work, we plan to further improve our hyperspectral camera through utilizing other modern post-processing networks and considering different camera architectures. In addition, we target to demonstrate the efficacy of our approach via real system with fabricated DOE and color filter array.

REFERENCES

[1] A. Signoroni, M. Savardi, A. Baronio, and S. Benini, "Deep learning meets hyperspectral image analysis: A multidisciplinary review," *Journal of imaging*, vol. 5, no. 5, p. 52, May 08, 2019.

[2] N. Hagen and M. W. Kudenov, "Review of snapshot spectral imaging technologies," *Optical Engineering*, vol. 52, p. 090901, 9 2013.

[3] G. R. Arce, D. J. Brady, L. Carin, H. Arguello, and D. S. Kittle, "Compressive coded aperture spectral imaging: An introduction," pp. 105–115, Jan 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6678264

[4] Y. Liu, X. Yuan, J. Suo, D. J. Brady, and Q. Dai, "Rank minimization for snapshot compressive imaging," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 12, pp. 2990–3006, Dec 01, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8481592

[5] J. M. Bioucas-Dias and M. A. T. Figueiredo, "A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE transactions on image processing*, vol. 16, no. 12, pp. 2992–3004, Dec 2007. [Online]. Available: https://ieeexplore.ieee.org/document/4358846

[6] S. Wright, R. Nowak, and M. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, 7 2009.

[7] I. Choi, D. Jeon, G. Nam, D. Gutierrez, and M. Kim, "High-quality hyperspectral reconstruction using a spectral prior," *ACM transactions on graphics*, vol. 36, no. 6, pp. 1–13, Nov 20, 2017. [Online]. Available: http://dl.acm.org/citation.cfm?id=3130810

[8] X. Miao, X. Yuan, Y. Pu, and V. Athitsos, "lambda-net: Reconstruct hyperspectral images from a snapshot measurement." IEEE, 2019, pp. 4058–4068.

[9] D. Jeon, S.-H. Baek, S. Yi, Q. Fu, X. Dun, W. Heidrich, and M. Kim, "Compact snapshot hyperspectral imaging with diffracted rotation," *ACM transactions on graphics*, vol. 38, no. 4, pp. 1–13, Jul 12, 2019. [Online]. Available: http://dl.acm.org/citation.cfm?id=3322946

[10] H. Arguello, S. Pinilla, Y. Peng, H. Ikoma, J. Bacca, and G. Wetzstein, "Shift-variant color-coded diffractive spectral imaging system," *Optica*, vol. 8, p. 1424, 11 2021.

[11] U. Akpinar, E. Sahin, M. Meem, R. Menon, and A. Gotchev, "Learning wavefront coding for extended depth of field imaging," *IEEE transactions on image processing*, vol. 30, pp. 3307–3320, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9362315

[12] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2472–2481.

[13] G. Singh, A. Mittal, and N. Aggarwal, "Resdnn: deep residual learning for natural image denoising," *IET Image Processing*, vol. 14, no. 11, pp. 2425–2434, 2020.

[14] J. W. Goodman, *Introduction to Fourier Optics*. Roberts and Company Publishers, 2005.

[15] J. Jiao, W.-C. Tu, S. He, and R. W. H. Lau, "Formresnet: Formatted residual learning for image restoration." The Institute of Electrical and Electronics Engineers, Inc. (IEEE), 2017, p. 1034.

[16] Z. Guo, Y. Sun, M. Jian, and X. Zhang, "Deep residual network with sparse feedback for image restoration," *Applied Sciences (Switzerland)*, vol. 8, 11 2018.

[17] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral-spatial residual network for hyperspectral image classification: A 3-d deep learning framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, pp. 847–858, 2 2018.

[18] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral-spatial feature learning via deep residual conv-deconv network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, pp. 391–406, 1 2018.

[19] C. Shan, X. Guo, and J. Ou, "Residual learning of deep convolutional neural networks for image denoising," *Journal of Intelligent and Fuzzy Systems*, vol. 37, pp. 2809–2818, 2019.

[20] C. Guo, C. Li, J. Guo, R. Cong, H. Fu, and P. Han, "Hierarchical features driven residual learning for depth map super-resolution," *IEEE Transactions on Image Processing*, vol. 28, pp. 2545–2557, 5 2019.

[21] H. Son and S. Lee, "Fast non-blind deconvolution via regularized residual networks with long/short skip-connections." Institute of Electrical and Electronics Engineers Inc., 6 2017.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision – ECCV 2016*, ser. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 630–645.

[23] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on computational imaging*, vol. 3, no. 1, pp. 47–57, 2016.

[24] M. E. Gehm, R. John, D. J. Brady, R. M. Willett, and T. J. Schulz, "Single-shot compressive spectral imaging with a dual-disperser architecture," *Optics express*, vol. 15, no. 21, pp. 14 013–14 027, Oct 17, 2007. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/19550674

[25] B. Arad and O. Ben-Shahar, "Sparse recovery of hyperspectral signal from natural rgb images," in *European Conference on Computer Vision*. Springer, 2016, pp. 19–34.