SPARSE UNMIXING USING DEEP CONVOLUTIONAL NETWORKS

Behnood Rasti[1], Bikram Koirala[2], and Paul Scheunders[2]

[1] Helmholtz-Zentrum Dresden-Rossendorf (HZDR),Chemnitzer Straße 40, 09599 Freiberg, Germany.
[2] Imec-Visionlab, University of Antwerp (CDE) Universiteitsplein 1, B-2610 Antwerp, Belgium.

ABSTRACT

This paper proposes a sparse unmixing technique using a convolutional neural network (SUnCNN). We reformulate the sparse unmixing problem into an optimization over the parameters of a convolutional network. Relying on a spectral library, the deep network learns in an unsupervised manner a mapping from a fixed input to the sparse abundances. Moreover, SUnCNN fulfills the sum-to-one constraint using a softmax activation layer. We compare SUnCNN with the state-of-the-art using a simulated and a real dataset. The experimental results show that the proposed deep learning-based unmixing method outperforms the others in terms of signal to reconstruction error. Additionally, SUnCNN is visually superior to the competing techniques. SUnCNN was implemented in Python (3.8) using PyTorch as the platform for the deep network and is available online: https://github.com/BehnoodRasti/SUnCNN.

***Index Terms*—** Hyperspectral image, unmixing, convolutional neural network, deep learning, deep prior, endmember extraction

1. INTRODUCTION

Sparse unmixing techniques estimate the fractional abundances of different pure materials by relying on a library of endmembers. As the number of pure materials in the scene is lower than the number of endmembers in the spectral libraries, only a few endmembers are required to reconstruct the mixed hyperspectral pixels. This leads to a sparse abundance matrix. Therefore, sparse regression techniques are exploited to estimate the abundances without having to extract or estimate the endmembers. [1]. As the fractional abundance is an areal percentage and no endmember can have a negative area, yielding the abundance nonnegativity constraint (ANC). Additionally, the observed spectrum should be entirely described by endmembers which leads to the abundance sum-to-one constraint (ASC).

Examples of sparse unmixing methods are sparse unmixing by variable splitting and augmented Lagrangian (SUnSAL), constrained SUnSAL (C-SUnSAL) [1] and collaborative sparse unmixing [2]. Both SUnSAL and C-SUnSAL apply an ℓ_1 penalty on the fractional abundances. SUnSAL utilizes ℓ_2 for the fidelity term while C-SUnSAL assumes a constraint to enforce the data fidelity. Collaborative sparse unmixing is similar to SUnSAL but applies $\ell_{2,1}$ (i.e., the sum of ℓ_2 on the abundances) to promote the sparsity on the abundances. In [3], a spectral prior was added to the sparse regression problem that assumes that some materials are known in the scene.

SUnSAL was improved in [4] by incorporating spatial information through applying a total variation penalty on the abundances (SUnSAL-TV). Some drawbacks of SUnSAL-TV are that it oversmooths boundaries, blurs abundance maps, and is computationally expensive. This was somewhat addressed by developing a technique called local collaborative sparse unmixing (LCSU) [5] and a new spectral–spatial weighted sparse unmixing (S^2WSU) framework [6]. In [7], an efficient two-phase multiobjective sparse unmixing approach was presented to exploit the spatial-contextual information for improving the abundance estimation. The problems of SUnSAL-TV were further tackled by introducing a fast Multiscale Sparse Unmixing Algorithm (MUA) [8] that promotes piecewise homogeneous abundances without compromising sharp discontinuities between neighboring pixels. In (MUA) [8], first, coarse fractional abundances are estimated by grouping pixels into perceptually meaningful regions and performing sparse regression using either a binary partition tree (BPT), MUA_{BPT}, or the simple linear iterative clustering (SLIC), MUA_{SLIC}. Then, sparse regression is used to estimate the abundance while the regularizer is applied to the coarse fractional abundances.

This paper proposes a sparse unmixing method using a convolutional neural network (SUnCNN). Relying on a spectral library, SUnCNN generates the abundances using a deep convolutional encoder-decoder. We reformulate the sparse unmixing problem into an optimization over the deep network's parameters. SUnCNN holds the sum-to-one and nonnegativity constraints using a softmax activation layer. Therefore, the major contributions of this article is three fold: 1) reformulating sparse unmixing into an optimization over a deep network's parameters which leads to the first deep learning-based sparse unmixing technique called SUnCNN; 2) implicitly inducing a deep image prior while holding ASC; and 3)

This research was funded by Alexander von Humboldt foundation and the Research Foundation-Flanders - project G031921N.

incorporating spatial information using the convolutional filters.

2. METHODOLOGY

We use the following mixing model for the observed spectral pixels:

$$\mathbf{Y} = \mathbf{DX} + \mathbf{N}, \tag{1}$$

where $\mathbf{Y}$ and $\mathbf{N}$ 2 $\mathsf{R}^{p \quad n}$ denote the observed HSI and the model error including noise, respectively. $\mathbf{D}$ 2 $\mathsf{R}^{p \quad m}$ is the spectral library containing m endmembers, and $\mathbf{X}$ 2 $\mathsf{R}^{m \quad n}$ contains the unknown fractional abundances ($p \qquad m$). Sparse unmixing estimates the fractional abundances $\mathbf{X}$ using the sparse regressions given by:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \frac{1}{2} jj\mathbf{Y} \quad \mathbf{DX}jj_F^2 + \lambda \sum_{i=1}^{n} jj\mathbf{x}_{(\mathrm{i})}jj_q \quad \text{s.t.} \mathbf{X} \quad 0, \mathbf{1}_m^T\mathbf{X} = \mathbf{1}_n^T, \tag{2}$$

where $0 \quad q < 1$. There is considerable debate on the selection of the sparsity-enforcing penalty and the value for q. In general, such a selection will be highly data dependent and therefore we suggest to use a general prior:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \frac{1}{2} jj\mathbf{Y} \quad \mathbf{DX}jj_F^2 + \lambda R(\mathbf{X}). \tag{3}$$

The general regularizer $R(\mathbf{X})$ can be implicitly applied using a deep network [9, 10]. Therefore, we can reformulated (3) as:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} jj\mathbf{Y} \quad \mathbf{D}f_\theta(\mathbf{Z})jj_F^2 \quad \text{s.t.} \quad \hat{\mathbf{X}} = f_{\hat{\theta}}(\mathbf{Z}), \tag{4}$$

where $\mathbf{Z}$ is the network input, which remains constant during the training, f_θ is the deep network with parameters θ. θ are randomly initialized and updated through the learning process. The deep network f_θ learns a mapping of the input $\mathbf{Z}$ to $\hat{\mathbf{X}}$ such that $\hat{\mathbf{X}} = f_{\hat{\theta}}(\mathbf{Z})$. We use a softmax function in the final layer of the network to enforce both ASC and ANC. The convolutional encoder-decoder used for $f_{\hat{\theta}}$ is shown in Fig. 1. We use 5 convolutional layers (Conv) in the main forward path and one in the skip connection, a downsampling block, and an upsampling block. We use batch norm (BN) to speed up the learning process and to obtain robustness in the hyperparameter selection. Leaky ReLU was selected as the activation function. All the selected hyperparameters for the network are given in Table 1.

3. EXPERIMENTAL RESULTS

In the experiments, SUnCNN is compared with SUnSAL [11], SUnSAL-TV [4], S^2WS [6], MUA_{BPT} and MUA_{SLIC} [8]. For the competing methods, the tuning parameters are set as default values.

Table 1: Hyperparameters used in the experiments for SUnCNN.

Hyperparameters				
	Input Ch.	Ouput Ch.	Filter Size	Stride
Conv1	p	256	3x3	2
Conv2	256	256	3x3	1
Conv3	260	256	3x3	1
Conv4	256	256	1x1	1
Conv5	256	p	1x1	1
ConvSkip	p	4	1x1	1
	Negative Slope			
Leaky ReLU	0.1			
	Scale Factor		Mode	
Upsample	2		Bilinear	
	Type		Learning Rate	
Optimizer	Adam		0.001	

3.1. Simulation Experiment

The synthetic library used for the simulated experiments is composed of 240 spectral signatures from the USGS library. We prune the library in a way that the spectral angle between any two endmembers is larger than 4.44°. The simulated Data Cube (DC1) contains 75 75 pixels and was simulated using a linear mixing model with 5 endmembers. The endmembers were selected from the library and the abundance maps are composed of five rows of square regions uniformly distributed over the spatial dimension. The results are given in terms of signal to reconstruction error (SRE) in dB:

$$SRE(\mathbf{X}, \hat{\mathbf{X}}) = 10 \log_{10} \frac{k\mathbf{X}k_F}{k\mathbf{X} \quad \hat{\mathbf{X}}k_F}, \tag{5}$$

for three levels of additive noise i.e., 20, 30, and 40 dB. Here, the number of iterations is set to 4000, 8000, 16000, for 20 dB, 30 dB, and 40 dB, respectively. The input of SUnCNN is set to the observed data, i.e., $\mathbf{Z} = \mathbf{Y}$,

Table 2 reports the unmixing results obtained from the different techniques applied to DC1. SUnCNN outperforms the other techniques, except in the case of 20 dB noise, where it performed second best. SUnSAL obtained the poorest result. Unlike S^2WSU, which performs better for high SNR, MUA performs better for the low SNRs. One may attribute this to the segmentation-based framework used in MUA which copes with the low SNRs by oversmoothing the abundances. This can be seen in Fig. 2, in which the estimated abundance maps using both segmentation-based techniques are oversmoothed. From Fig. 2, we can observe that SUnCNN successively preserves the geometrical structures in the abundance maps (even for low SNR i.e., 20 dB), which can be attributed to the incorporation of spatial information via the convolutional operators. On the other hand, the competing methods fail to preserve such structures, particularly for lower SNRs. We should

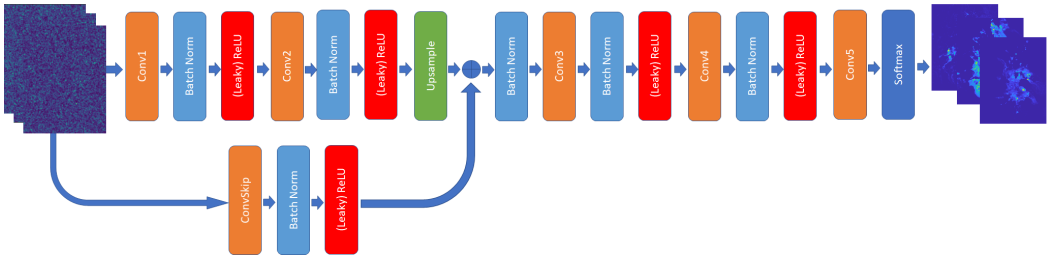

Fig. 1: The architecture of the deep network, f_θ, used in SUnCNN. The network uses a skip connection and different layers are shown using specific colors.

note that S^{2}WSU, MUA$_{\text{BPT}}$, and MUA$_{\text{SLIC}}$ induce artifacts into the abundance maps which is not desirable.

Table 2: The results of different sparse unmixing techniques applied to DC1 in terms of SRE. The best performances are shown in bold.

SNR	SUnSAL	SUnSAL-TV	S^{2}WSU	MUA$_{\text{BPT}}$	MUA$_{\text{SLIC}}$	SUnCNN
20 dB	2.27	4.71	3.85	**6.70**	5.67	5.71
30 dB	4.46	7.22	7.74	9.13	7.87	**10.25**
40 dB	6.89	11.05	14.12	10.72	11.17	**15.20**

3.2. Real Experiment

In the real experiment, we use the Jasper Ridge dataset. This dataset contains 100 100 hyperspectral pixels (see Fig. 3(a)) for the true-color image). Each hyperspectral pixel contains 224 reflection values in the wavelength range [380-2500] nm. Among 224 bands, twenty-six water absorption bands (1–3, 108–112, 154–166, and 220–224) were removed. This dataset has four endmembers i.e., Tree, Water, Soil, and Road. The library $\mathbf{D}\ 2\ \mathbb{R}^{198\ \ 529}$ is composed of 529 spectral pixels obtained from the Jasper Ridge dataset. We use 15000 iterations for the real dataset. Here, the input of SUnCNN is set to noise, i.e., $\mathbf{Z} = \mathbf{N}$.

The results of the sparse unmixing techniques applied to the Jasper Ridge dataset are compared in Table 3 in terms of SRE. SUnCNN outperforms the other techniques for the overall abundance estimation and provides the best abundance estimations for Tree and Water.

Fig. 3 visually compares the abundance maps obtained by different techniques applied on the Jasper Ridge dataset. All the techniques preserve the geometrical structures of the dataset and perform similarly in terms of visual comparison.

4. CONCLUSION

We proposed a sparse unmixing technique using a deep convolutional neural network (SUnCNN). SUnCNN utilizes a convolutional encoder-decoder network to generate the abundance maps. Inspired by deep image prior, the sparsity prior was implicitly enforced using a deep network. The experiments were carried out on a synthetic dataset and the Jasper Ridge dataset. The experimental results show that SUnCNN outperforms the state-of-the-art in terms of SRE. Additionally, visual comparison reveals that, unlike the competing techniques, SUnCNN provides sharp maps without artifacts, even for low SNR values.

Table 3: The results of different sparse unmixing technique applied to the Jasper Ridge dataset in terms of SRE. The best performances are shown in bold.

	SUnSAL	SUnSAL-TV	S2WSU	MUA_BPT	MUA_SLIC	SUnCNN
Tree	8.45	7.06	7.59	6.44	6.57	**9.53**
Water	6.57	7.51	6.64	6.80	6.61	**8.67**
Soil	**7.22**	6.79	6.96	5.79	5.84	6.58
Road	6.59	**6.76**	6.16	4.99	5.14	6.21
Overall	7.20	6.96	6.95	6.28	6.29	**8.01**

5. REFERENCES

[1] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.

[2] M. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Collaborative sparse regression for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 341–354, 2014.

[3] W. Tang, Z. Shi, Y. Wu, and C. Zhang, "Sparse unmixing of hyperspectral data using spectral a priori information," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 2, pp. 770–783, 2015.

[4] M. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 11, pp. 4484–4502, 2012.

[5] S. Zhang, J. Li, K. Liu, C. Deng, L. Liu, and A. Plaza, "Hyperspectral unmixing based on local collaborative

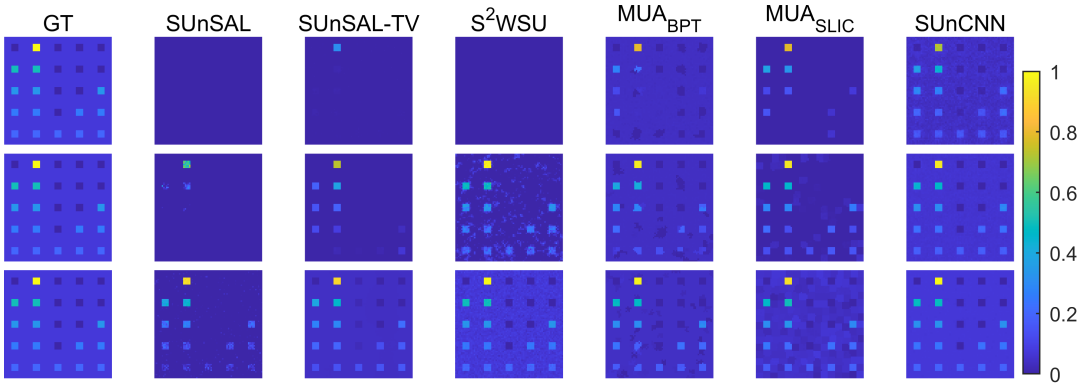

Fig. 2: DC1: The fractional abundance of endmember 2. From top to bottom: SNR of 20, 30, and 40 dB.

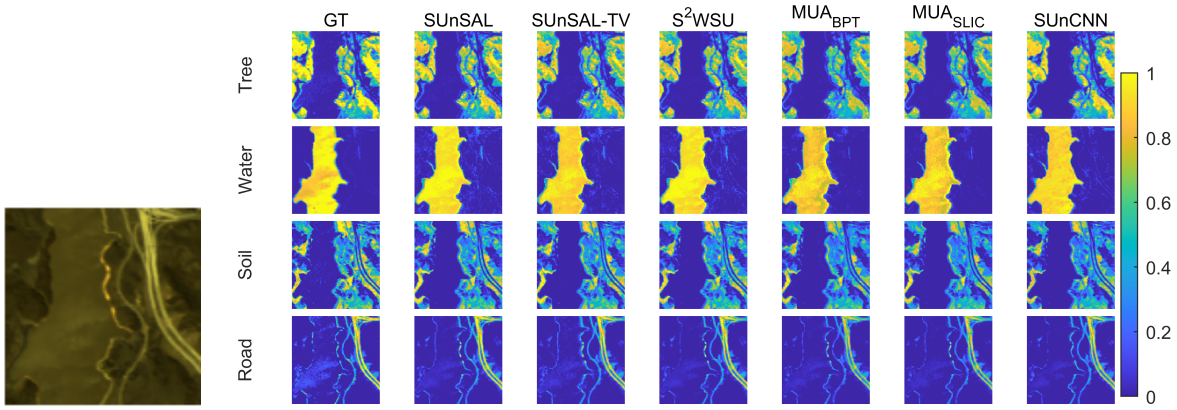

Fig. 3: (a)The Jasper Ridge dataset: the true-color image (Red: 570.14 nm, Green: 532.11 nm, Blue: 427.53 nm) (b)Abundance maps of four endmembers estimated using different sparse unmixing techniques applied to the Jasper Ridge dataset.

sparse regression," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 5, pp. 631–635, 2016.

[6] S. Zhang, J. Li, H. Li, C. Deng, and A. Plaza, "Spectral–spatial weighted sparse regression for hyperspectral image unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 6, pp. 3265–3276, 2018.

[7] X. Jiang, M. Gong, T. Zhan, K. Sheng, and M. Xu, "Efficient two-phase multiobjective sparse unmixing approach for hyperspectral data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 2418–2431, 2021.

[8] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "A fast multiscale spatial regularization for sparse hyperspectral unmixing," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 4, pp. 598–602, 2019.

[9] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," *International Journal of Computer Vision*, vol. 128, no. 7, pp. 1867–1888, Mar 2020.

[10] B. Rasti, B. Koirala, P. Scheunders, and P. Ghamisi, "UnDIP: Hyperspectral unmixing using deep image prior," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–15, 2021.

[11] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," 2010.