Semi-supervised Deep Convolutional Transform Learning for Hyperspectral Image Classification

Shikha Singh, Angshul Majumdar, Emilie Chouzenoux, Giovanni Chierchia

Semi-supervised Deep Convolutional Transform Learning for Hyperspectral Image Classification

Shikha Singh
Dept. of ECE, IIIT-Delhi
shikhas@iiitd.ac.in

Angshul Majumdar
Dept. of ECE, IIIT-Delhi
angshul@iiitd.ac.in

Emilie Chouzenoux
CVN, Inria, Univ. Paris Saclay
emilie.chouzenoux@inria.fr

Giovanni Chierchia
ESIEE, Paris
giovanni.chierchia@esiee.fr

***Abstract*— This work addresses the problem of hyperspectral image classification when the number of labeled samples is very small (few shot learning). Our work is based on the recently proposed framework of convolutional transform learning. In this work, we propose a semi-supervised version of deep convolutional transform learning. We compare with four recent studies which are tailored for solving the few-shot learning problem in hyperspectral classification. Results show that our method can improve over the state-of-the-art.**

Keywords*— *hyperspectral classification, supervised learning, deep learning.

I. INTRODUCTION

In hyperspectral image classification, the usual protocol is to label some samples manually after acquiring the image; this forms the training set. The labeled training data is fed into some classifier (after feature extraction if needed) in order to generate the labels for the remaining unlabeled testing samples. Ideally one would like to manually label as few samples as possible since this is a laborious time-consuming process that needs expertise.

Deep learning is known to be data hungry. Consequently, initial studies on deep learning [1, 2] used more than 80% of the data as a training set; in practice this is infeasible. Over time, deep learning techniques like [3, 4] have been able to reduce the data requirement, but even with the new age of deep neural networks (DNN), the number of labeled samples required is usually higher than shallow learning techniques like [5, 6].

Given that DNNs require copious amounts of labeled data for training, the only approach is to augment the number of labeled samples. The standard approach to address this issue had been employing generative models [7, 8] in the recent past and meta-learning [9, 10] in current years. Classical techniques like Gaussian mixture models have also been used for labeled data augmentation [11]. A thorough review of different data augmentation techniques can be found in [12].

Over the years various deep learning architectures have been employed for the task of hyperspectral image classification. Initial studies used stacked autoencoders [1] and deep belief networks [2] for image classification. Later on, convolutional neural networks (CNN) showed improvement in results [13, 14]. Later more complex CNNs like ResNet [15] and DenseNet [16] were employed for the said problem. Capsule networks had also been used for hyperspectral classification with varying degrees of success [17, 18]. Even though not a natural choice recurrent neural networks too have been used for hyperspectral image classification [19]. In the limited scope of this letter, it is not possible to review all deep learning techniques. The interested reader can peruse survey papers [20, 21]. The said papers are from two years back and do not include the latest deep learning techniques for hyperspectral image classification based on attention networks [22] and graph CNNs [23].

The unique and interesting aspect about hyperspectral image classification is that the sum of training (labeled) and testing (unlabeled) samples are usually fixed, as discussed at the beginning of this section. This allows for semi-supervised learning of the test features [24] while training the DNN. This work follows the same paradigm.

Our work is based on the framework of convolutional transform learning (CTL) [25, 26]. CTL overcomes the major shortcomings of CNN since it can learn convolutional filters in an unsupervised fashion. A recent study proposed a supervised version of shallow convolutional transform learning [27]. In this work, we propose a semi-supervised formulation of convolutional transform learning, where the features for both the labeled and the unlabeled data are learnt during training of the in-built classifier. During testing, the features corresponding to the unlabeled training data are input to the learnt classifier.

Since the CTL is a relatively new framework, we will discuss it in depth in Section II; in the recent past, the said framework has been used successfully in other hyperspectral imaging applications [28, 29]. This is necessary for understanding our formulation discussed in Section III. The experimental evaluation is detailed in Section IV. The conclusion is discussed in Section V.

II. BACKGROUND

In convolutional transform learning (CTL) a set of filters are learnt such that when operated on the data they produce the corresponding representations. Formally this is expressed as

$$t_m * s^{(k)} = x_m^{(k)},\ \forall\ m \in \{1...M\} \text{ and } \forall k \in \{1...K\} \quad (1)$$

Here $s^{(k)}$ denotes the k^{th} sample, t_m the m^{th} convolutional filter and $x_m^{(k)}$ the representation of the k^{th} sample after applying the m^{th} convolutional filter. There are a total of K samples and M filters. The symbol '$*$'denotes a convolutional operation with zero padding.

During training, the filters and the representations are learnt by solving the following optimization problem.

$$\min_{(t_m)_m,\left(x_m^{(k)}\right)_{m,k}} \sum_{k=1}^{K}\sum_{m=1}^{M}\left(\left\|t_m * s^{(k)} - x_m^{(k)}\right\|_2^2 + \psi\left(x_m^{(k)}\right)\right) + \mu\|T\|_F^2 - \lambda \log\det(T) \quad (2)$$

Here ψ is a regularization on the representation and (μ, λ) is positive. T is defined as $T = [t_1 | ... | t_m]$. The log determinant term prevents the trivial solution $t_m = 0, x_m^{(k)} = 0$ and also promotes diversity among the learnt filters; the other penalty $\|T\|_F^2$ prevents degenerate solutions where $t_m \to \infty, x_m^{(k)} \to \infty$. The penalty on T is borrowed from transform learning [30].

One can see how it is possible to learnt convolutional filters from training data in an unsupervised fashion. CNNs do not have the said penalty on the learnt filters and hence can end up at the trivial solution. Furthermore, without the penalty on T, there is no guarantee that CNNs will learn unique filters.

In matrix-vector form, (2) can be expressed as follows

$$\min_{T,X} \|T\bullet S - X\|_F^2 + \Psi(X)) + \mu\|T\|_F^2 - \lambda \log\det(T) \quad (3)$$

where $S = \left[s^{(1)} | ... | s^{(K)}\right]^T$, $X = \left[x_1^{(k)} | ... | x_M^{(k)}\right]_{1\le k\le K}$, $T\bullet S = \begin{bmatrix} t_1 * s^{(1)} & ... & t_M * s^{(1)} \\ ... & ... & \\ t_1 * s^{(K)} & ... & t_M * s^{(K)} \end{bmatrix}$ and Ψ amounts to applying the penalty term ψ column-wise on matrix X and summing.

The unsupervised formulation for CTL was proposed in [27]. Later the label consistency term from [31, 32] was added to this formulation to make it supervised [29]. This led to the following formulation.

$$\min_{T,X} \|T\bullet S - X\|_F^2 + \Psi(X)) + \mu\|T\|_F^2 - \lambda \log\det(T) + \eta\|Q - MX\|_F^2 \quad (4)$$

In (4), the last term is the supervision term. Matrix Q consists of one-hot encoded class labels, M is a linear map that projects the representation X to the corresponding class labels. Here η is a positive constant that controls the relative importance of the label consistency term. Note that in [29] the problem was multi-label classification and hence this simple label consistency term is sensible.

In [28] a deeper extension of the unsupervised formulation was proposed leading to deep convolutional transform learning (DCTL). As the name suggests, in the said formulation, multiple convolutional filters were being applied to the samples one after the other to generate the representation. This is expressed as follows,

$$\min_{T_1,T_2,T_3X} \left\|T_3\bullet\left(T_2\bullet\left(T_1\bullet S\right)\right) - X\right\|_F^2 + \Psi(X)) + \mu\sum_{i=1}^{3}\left\{\|T_i\|_F^2 - \lambda \log\det(T_i)\right\} \quad (5)$$

Hereabove, T_i refers to the i^{th} layer of convolutional filters. In each layer, the T_i is formed by stacking the filters as columns of a matrix. The formulation (5) is shown for 3 layers but can be extended further.

III. PROPOSED FORMULATION

Let us re-iterate the problem of hyperspectral image classification. Once the hyperspectral imagery is captured the task is to label each position / pixel. Let us assume that there are N such positions. The usual protocol is to label a subset of these N (say n) positions manually and train a classifier to label the remaining N-n positions. Whatever the value of n, the total number of positions to be labeled (be it manual or automatic) is always N.

In this work, we propose a semi-supervised formulation where the n labeled samples (corresponding to n positions) will be used for training a classifier. During training, all the representation / features for all the N samples will be generated. At testing, the N-n unlabeled samples will be input to the learnt classifier for labeling.

We add binary cross-entropy (J_{BCE}) loss for supervision instead of the label consistency term to the deep convolutional transform learning formulation. Note that this loss only pertains to the labeled data; for the unlabeled portion of the data only deep convolutional transform learning formulation is applicable. Our semi-supervised formulation can be expressed as,

$$\min_{T_1,T_2,T_3X_L,X_U} \left\|T_3\bullet\left(T_2\bullet\left(T_1\bullet[S_L | S_U]\right)\right) - [X_L | X_U]\right\|_F^2 + \Psi([X_L | X_U]) + \mu\sum_{i=1}^{3}\left\{\|T_i\|_F^2 - \lambda \log\det(T_i)\right\} + \eta J_{BCE}\left(\sigma(MX_L), Q\right) \quad (6)$$

Equivalently –

$$\min_{T_1,T_2,T_3X_L,X_U} \mu\sum_{i=1}^{3}\left\{\|T_i\|_F^2 - \lambda \log\det(T_i)\right\} + \underbrace{\left\|T_3\bullet\left(T_2\bullet\left(T_1\bullet S_L\right)\right) - X_L\right\|_F^2 + \Psi(X_L) + \eta J_{BCE}\left(\sigma(MX_L), Q\right)}_{\text{Supervised}} + \underbrace{\left\|T_3\bullet\left(T_2\bullet\left(T_1\bullet S_U\right)\right) - X_U\right\|_F^2 + \Psi(X_U)}_{\text{Unsupervised}}$$

The subscript '*L*' corresponds to labeled and '*U*' corresponds to unlabelled. The BCE loss is only being defined on the labeled data. In (6) σ refers to the sigmoid function and Q (as defined before) refers to the one-hot encoded label vectors. In a more comprehensible notation, our semi-supervised formulation (6) can be expressed as the cost of 'supervised + unsupervised' terms as follows:

Here we have shown the formulation for 3 layers. One can see that extending it to more layers is straightforward. The problem (6) is solved using any standard gradient descent algorithm. Here, we have used Adam [33]. The activation function used is scaled exponential linear unit (SELU). The schematic diagram for 2-layers semi-supervised deep convolutional transform learning is shown in Fig. 1.

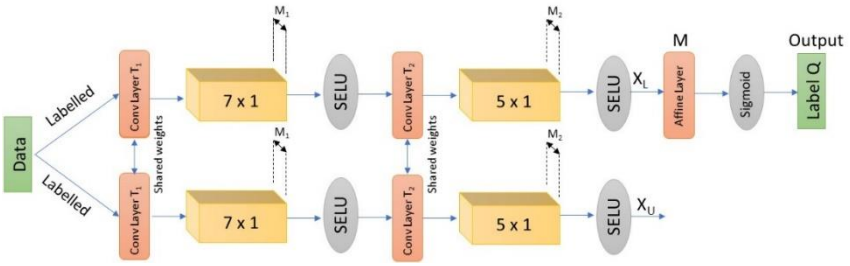

Fig. 1. Schematic diagram of the proposed approach

IV. Experimental Results

We evaluate our proposed technique on two benchmark datasets – Indian Pines[1] and Pavia University[2]. The standard pre-processing steps are performed on these datasets prior to classification.

For both datasets, an extremely challenging experimental protocol from [34] is followed. The number of labeled samples to be considered is 5 per class; the rest are unlabeled – the goal is to label the unlabeled samples. All the experiments were carried out 100 times, using random splits of labeled and unlabeled samples. The arithmetic means of these 100 sets of experiments are reported in this work.

We compare with recent techniques that have been specifically tailored for limited training samples. Deep few-shot learning (DFSL) [3] is based on the meta-learning approach. Dual-path Siamese network (DPSN) [35] uses a Siamese network for classification; to augment labeled data it uses a combination of a generative adversarial network. Dual graph convolutional network (DGCN) [25] uses a graph convolutional network for classification followed by label propagation. The final work that we compare is Intraclass Similarity Structure Representation (ISSR) [36]. In [36] this work a novel label propagation approach is proposed to increase the number of training samples; the augmented data serves as input to a standard deep CNN for classification.

Our proposed network does not consider spatial information; only the spectral information for every pixel is input to the network as 1D data. Our proposed work requires the specification of three parameters. We have used the values $\mu=\lambda=.1$ and $\eta=.5$. As for the size of the filters we have used 7x1, 5x1, 3x1; we also tried with four layers but the results deteriorated owing to overfitting and hence we are not showing the results.

TABLE I Comparison with State-of-the-art Techniques

Dataset	Metric	ISSR	DPSN	DGCN	DFSL	Proposed
Pavia	OA	80.09	85.02	88.91	86.24	**89.79**
	AA	80.26	85.69	89.48	87.41	**90.08**
	Kappa	.78	.82	.85	.84	**0.86**
Indian Pines	OA	70.36	75.32	77.38	75.85	**81.02**
	AA	70.95	75.97	77.91	76.30	**84.32**
	Kappa	.69	.74	.77	.75	**0.80**

The comparative results in terms of overall accuracy (OA), average accuracy (AA), and Kappa coefficient are shown in Table I. One can see that our method performs the best in terms of all the metrics. Among the benchmarks DGCN performs the best; in fact, it is better than our four-layer architecture. The meta-learning based approach DFSL also performs relatively well but not as good as DGCN. DPSN combines several aspects of deep learning and data augmentation but performs worse than DGCN. The performance of ISSR is the worst; it has a separate data augmentation stage and a classification stage; such a piecemeal approach does not perform at par with the other end-to-end architectures. For visual clarity we show the classification results as images in Fig. 2. It corroborates the numerical findings.

[1] https://paperswithcode.com/dataset/indian-pines

[2] https://paperswithcode.com/dataset/pavia-university

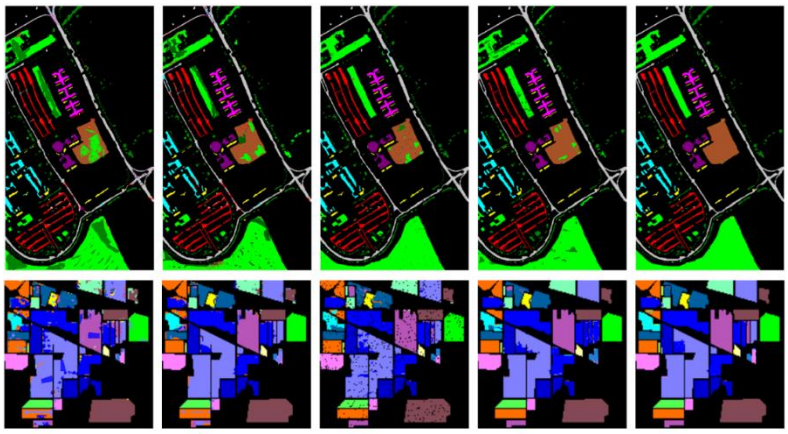

Fig. 2. Top - Pavia University; Bottom – Indian Pines. Pictorial view of classification. Left to Right – ISSR, DPSN, DGCN and Proposed

A. *Ablation Studies*

In the first set of experiments, we analyze the different aspects of our proposed solution. We will see how the results change with depth. We will also see in each depth, how the results vary between semi-supervised and supervised formulations. The former is the proposition of this study and has been discussed in detail. The supervised formulation means that the network is only trained on the labeled samples; the unlabeled samples are not used during training. After training, the learnt filters are used for extracting the features from the input data and subsequent classification.

TABLE II RESULTS ON PAVIA UNIVERSITY

	Metric	1 layer	2 layers	3 layers	4 layers
Semi-supervised	OA	88.17	89.06	**89.79**	88.23
	AA	88.45	89.73	**90.08**	89.14
	Kappa	.83	.84	**0.86**	.84
Supervised	OA	82.16	81.02	78.67	72.38
	AA	81.29	80.64	78.11	71.60
	Kappa	.78	.76	0.75	.70

TABLE III RESULTS ON INDIAN PINES

	Metric	1 layer	2 layers	3 layers	4 layers
Semi-supervised	OA	79.84	80.19	**81.02**	78.68
	AA	83.72	84.01	**84.32**	82.11
	Kappa	0.77	0.79	**0.80**	0.76
Supervised	OA	74.12	72.02	70.13	65.03
	AA	74.96	72.42	70.87	65.62
	Kappa	0.72	0.70	0.69	0.64

From Tables II and III we see that for the semi-supervised formulation the results improve from layers one to three and then dips when one more layer is added. This is probably due to over-fitting. The supervised formulation always yields considerably worse results than the semi-supervised one. This is because the volume of data used for training the network is very less leading to over-fitting. Furthermore, for the supervised formulation, the results keep on deteriorating with depth. This too is likely due to over-fitting. With limited data, learning more parameters hampers the generalization ability of the network.

Finally, we show the empirical convergence plots for our proposed algorithm. Owing to limitations in space we only show it for the three-layer architecture. For other layers, the trend is similar. From Fig. 3, one can see that the algorithm convergences in 50 iterations.

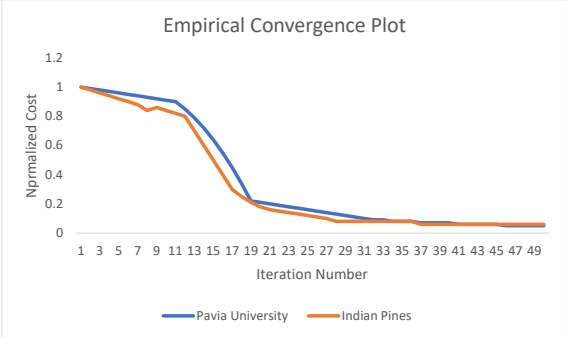

Fig. 3. Cost vs Iteration. Empirical Convergence Plot for 3 Layer Architecture.

V. CONCLUSION

This work proposes a new approach for hyperspectral image classification when the number of labeled samples are limited. The work is based on convolutional transform learning. A semi-supervised architecture for deep convolutional transform learning is proposed. Comparison with state-of-the-art techniques on few shot learning has shown that our proposed method can improve over rest.

We believe there is further scope for improvement. The current work only accounts for spectral information while ignoring spatial information. There can be two possibilities for incorporating spatial information. First, we can consider 2D convolutions on patches / superpixels. Second, we can keep the 1D convolutional filters as it is but incorporate the spatial information via graph regularization.

In terms of methodology, we believe that the framework of deep convolutional transform learning can improve by incorporating metric learning. Recent studies like [38] have shown how it can be incorporated into the deep dictionary learning framework; we believe that similar improvements can be achieved in the deep convolutional transform formulation as well.

ACKNOWLEDGMENT

E.C. and G.C. acknowledge support from the Agence Nationale de la Recherche of France under MAJIC (ANR-17-CE40-0004-01) project.

REFERENCES

[1] Y. Chen, Z. Lin, X. Zhao, G. Wang and Y. Gu, "Deep Learning-Based Classification of Hyperspectral Data," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 7, no. 6, pp. 2094-2107, 2014.

[2] Y. Chen, X. Zhao and X. Jia, "Spectral–Spatial Classification of Hyperspectral Data Based on Deep Belief Network," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8, no. 6, pp. 2381-2392, 2015.

[3] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan and R. Wang, "Deep Few-Shot Learning for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 4, pp. 2290-2304, 2019.

[4] K. Gao, W. Guo, X. Yu, B. Liu, A. Yu and X. Wei, "Deep Induction Network for Small Samples Classification of Hyperspectral Images," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 3462-3477, 2020.

[5] A. C. Karaca, "Domain Transform Filter and Spatial-Aware Collaborative Representation for Hyperspectral Image Classification Using Few Labeled Samples," IEEE Geoscience and Remote Sensing Letters (preprint).

[6] Y. Xu, Q. Du, W. Li and N. H. Younan, "Efficient Probabilistic Collaborative Representation-Based Classifier for Hyperspectral Image Classification," IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 11, pp. 1746-1750, 2019

[7] L. Zhu, Y. Chen, P. Ghamisi and J. A. Benediktsson, "Generative Adversarial Networks for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 9, pp. 5046-5063, 2018.

[8] X. Wang, K. Tan, Q. Du, Y. Chen and P. Du, "Caps-TripleGAN: GAN-Assisted CapsNet for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 9, pp. 7232-7245, 2019.

[9] K. Gao, B. Liu, B., X. Yu, P. Zhang, X. Tan and Y. Sun, "Small sample classification of hyperspectral image using model-agnostic meta-learning algorithm and convolutional neural network," International Journal of Remote Sensing, vol. 42, no. 8, pp. 3090-3122, 2021.

[10] K. Gao, W. Guo, X. Yu, B. Liu, A. Yu and X. Wei, "Deep Induction Network for Small Samples Classification of Hyperspectral Images," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 3462-3477, 2020.

[11] A. Davari, E. Aptoula, B. Yanikoglu, A. Maier and C. Riess, "GMM-Based Synthetic Samples for Classification of Hyperspectral Images With Limited Training Data," IEEE Geoscience and Remote Sensing Letters, vol. 15, no. 6, pp. 942-946, 2018

[12] J. Nalepa, M. Myller and M. Kawulok, "Hyperspectral data augmentation," arXiv preprint arXiv:1903.05580, 2019.

[13] K. Makantasis, K. Karantzalos, A. Doulamis and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," IEEE IGARSS, pp. 4959-4962, 2015.

[14] P. Jia, M. Zhang, W. Yu, F. Shen and Y. Shen, "Convolutional neural network based classification for hyperspectral data," IEEE IGARSS, pp. 5075-5078, 2016.

[15] Z. Zhong, J. Li, L. Ma, H. Jiang and H. Zhao, "Deep residual networks for hyperspectral image classification," IEEE IGARSS, pp. 1824-1827, 2017.

[16] Y. Bai, Q. Zhang, Z. Lu and Y. Zhang, "SSDC-DenseNet: A Cost-Effective End-to-End Spectral-Spatial Dual-Channel Dense Network for Hyperspectral Image Classification," IEEE Access, vol. 7, pp. 84876-84889, 2019.

[17] M. E. Paoletti et al., "Capsule Networks for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 4, pp. 2145-2160, 2019.

[18] J. Yin, S. Li, H. Zhu and X. Luo, "Hyperspectral Image Classification Using CapsNet With Well-Initialized Shallow Layers," IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 7, pp. 1095-1099, 2019.

[19] L. Mou, P. Ghamisi and X. X. Zhu, "Deep Recurrent Neural Networks for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 7, pp. 3639-3655, 2017

[20] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi and J. A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 9, pp. 6690-6709, 2019.

[21] N. Audebert, B. Le Saux and S. Lefevre, "Deep Learning for Classification of Hyperspectral Data: A Comparative Review," IEEE Geoscience and Remote Sensing Magazine, vol. 7, no. 2, pp. 159-173, 2019.

[22] M. Zhu, L. Jiao, F. Liu, S. Yang and J. Wang, "Residual Spectral–Spatial Attention Network for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 59, no. 1, pp. 449-462, 2021.

[23] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza and J. Chanussot, "Graph Convolutional Networks for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing (preprint).

[24] H. Wu and S. Prasad, "Semi-Supervised Deep Learning Using Pseudo Labels for Hyperspectral Image Classification," IEEE Transactions on Image Processing, vol. 27, no. 3, pp. 1259-1270, 2018.

[25] J. Maggu, E. Chouzenoux, G. Chierchia and A. Majumdar, "Convolutional Transform Learning", ICONIP, pp. 162-174, 2018.

[26] J. Maggu, E. Chouzenoux, G. Chierchia and A. Majumdar, "Deep Convolutional Transform Learning", ICONIP, pp. 300-307, 2020.

[27] S. Singh, J. Maggu, A. Majumdar, E. Chouzenoux and G. Chierchia, "Multi-Label Consistent Convolutional Transform Learning: Application to Non-Intrusive Load Monitoring," IEEE ICASSP, pp. 4287-4291, 2020.

[28] A. Goel and A. Majumdar, "K-Means Embedded Deep Transform Learning for Hyperspectral Band Selection," IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022.

[29] A. Goel and A. Majumdar, "Sparse Subspace Clustering Friendly Deep Dictionary Learning for Hyperspectral Image Classification," IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022

[30] S. Ravishankar, B. Wen and Y. Bresler, "Online Sparsifying Transform Learning—Part I: Algorithms," IEEE Journal of Selected Topics in Signal Processing, vol. 9, no. 4, pp. 625-636, 2015.

[31] Z. Jiang, Z. Lin and L. S. Davis, "Label Consistent K-SVD: Learning a Discriminative Dictionary for Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 11, pp. 2651-2664, 2013.

[32] J. Maggu, H. K. Aggarwal and A. Majumdar, "Label-Consistent Transform Learning for Hyperspectral Image Classification," IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 9, pp. 1502-1506, 2019.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," ICLR 2014.

[34] S. Jia, X. Deng, J. Zhu, M. Xu, J. Zhou and X. Jia, "Collaborative Representation-Based Multiscale Superpixel Fusion for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 10, pp. 7770-7784, 2019.

[35] L. Huang and Y. Chen, "Dual-Path Siamese CNN for Hyperspectral Image Classification With Limited Training Samples," IEEE Geoscience and Remote Sensing Letters, vol. 18, no. 3, pp. 518-522, 2021.

[36] W. Wei, L. Zhang, Y. Li, C. Wang and Y. Zhang, "Intraclass Similarity Structure Representation-Based Hyperspectral Imagery Classification With Few Samples," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 1045-1054, 2020.

[37] W. Tang, E. Chouzenoux, J. -C. Pesquet and H. Krim, "Deep Transform and Metric Learning Networks," IEEE ICASSP, pp. 2735-2739, 2021.