2024 14th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS) | 979-8-3315-1313-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/WHISPERS65427.2024.10876552

ACTIVE LEARNING FOR HYPERSPECTRAL IMAGES WITH REGULARIZED WASSERSTEIN DICTIONARY LEARNING

Scott Fullenbaum, James M. Murphy

Tufts University, Department of Mathematics
Medford, MA 02155, USA

ABSTRACT

We present a novel active learning method for hyperspectral images based on representation learning in Wasserstein space. We perform regularized Wasserstein dictionary learning in the space of hyperspectral pixels, then leverage the learned barycentric coefficients to embed the high-dimensional spectra into a low-dimensional space. Sampling in the low-dimensional space leads to high-quality labels that propagate accurately to the remaining pixels in the data. Our method achieves a high level of accuracy with very few training labels, suggesting its utility for hyperspectral image classification in the active labeling setting.

Index Terms— representation learning, Wasserstein space, active learning, semisupervised learning

1. INTRODUCTION

Hyperspectral images (HSI) capture reflectances over a wide range of wavelengths, which allows the gathering of useful information via aerial and spaceborne instruments. However, collecting labeled HSI data is expensive and time-intensive, which motivates the development of semisupervised and unsupervised learning methods that can label images with little to no labeled training data. The difficulty of learning representations in these contexts in which training data is not plentiful is compounded by the high-dimensional nature of HSI; typical sensors record reflectances over hundreds of spectral bands.

In this paper, we approach the problem of labeling hyperspectral pixels through an active learning approach based on representation learning in Wasserstein space. Each pixel is represented as a probability measure and pixels are compared using entropic Wasserstein distances [1]. Our approach captures the geometry of each pixel by performing dictionary learning of spectra in Wasserstein space. We then carry out an active labeling algorithm combining the learned coefficients with the (unnormalized) mass of each pixel. Promising results on the Salinas A HSI are shown, indicating the viability of our approach to active learning.

The remainder of this paper is organized as follows. In Section 2, we overview semisupervised and active learning methods before providing background on data analysis in Wasserstein space. In Section 3, we discuss our approach to cluster-based active learning via Wasserstein dictionary learning for HSI. Section 4 showcases our approach on the Salinas A HSI [2], and then we discuss directions for future work in Section 5.

2. BACKGROUND

Background on Semisupervised and Active Learning: Given HSI pixels $\{\mu_i\}_{i=1}^n \subset \mathbb{R}^D$ with D corresponding to the number of spectral bands in the image, we aim to learn the labels $\{y_i\}_{i=1}^n$ corresponding to the material class in the scene the pixels belongs to. Semisupervised learning assumes that a small number of labeled pairs (μ_j, y_j) are known in addition to the full unlabeled dataset $\{\mu_i\}_{i=1}^n$. Deep learning [3, 4], graph-based approaches [5], and SVM-based methods [6] have demonstrated their use for semisupervised labeling of hyperspectral images.

Active learning is a particular form of semisupervised learning that assumes no labeled pairs (μ_j, y_j) are known initially. Instead, the active learning algorithm leverages the unlabeled data to determine a list of points—selected by an algorithm-specific criteria—to query for labels. By doing so, the algorithm can use a small number of labeled points to generate meaningful insights. Nonlinear diffusion [7, 8], Bayesian methodologies [9, 10], neural networks [11], and SVMs [12] have all seen use for active learning of HSI.

Background on Representation Learning in Wasserstein Space: We propose to learn a representation of each pixel in Wasserstein space and leverage the learned representations for both querying labels in active learning and propagating the queried labels to the remainder of the data set. To do so, we utilize nonlinear dictionary learning in Wasserstein space [13, 14]. Let $\Delta^D := \{(x_1, \dots, x_D) \in \mathbb{R}^D \mid \sum_{k=1}^D x_k = 1, x_k \geq 0, \ \forall k\}$ be the discrete D-dimensional probability simplex. Given $\mu, \nu \in \Delta^D$, define $\Pi(\mu, \nu) \subset \Delta^{D \times D}$ as the set of *couplings* between μ and ν, the collection of $\pi \in \mathbb{R}_{\geq 0}^{D \times D}$ such that $\forall j, \ \sum_{i=1}^D \pi_{ij} = \nu_j$, and $\forall i, \ \sum_{j=1}^D \pi_{ij} = \mu_i$. Suppose μ, ν are associated to a common set of spectral wavelengths $\{b_k\}_{k=1}^D \subset \mathbb{R}$; this will be the case for HSI spectra

from a common scene. For $p \geq 1$, the *entropic p-Wasserstein distance* [15] between μ and ν is

$$W^p_{p,\epsilon}(\mu,\nu) := \min_{\pi\in\Pi(\mu,\nu)} \sum_{k=1}^{D}\sum_{\ell=1}^{D}(\pi_{k\ell}|b_k - b_\ell|^p + \epsilon\pi_{k\ell}\log(\pi_{k\ell})) \tag{1}$$

where $\epsilon > 0$ is a regularization parameter. The solution π^* to (1) couples μ to ν efficiently and smoothly; for a detailed discussion of entropic optimal transport, we refer to [1, 16]. For a set of m distributions $\{\nu_j\}_{j=1}^m \subset \Delta^D$ and a vector of weights $w \in \Delta^m$, we define the *entropic p-Wasserstein barycenter* [17, 18, 19] to be

$$\text{Bary}(\{\nu_j\}_{j=1}^m; w) := \arg\min_{\mu\in\Delta^D} \sum_{j=1}^{m} w_j W^p_{p,\epsilon}(\mu,\nu_j). \tag{2}$$

Given observed probability distributions $\{\mu_i\}_{i=1}^n$ (in our context HSI pixels after normalization so that they lie in Δ^D), we can learn meaningful representations useful for semisupervised learning by finding probability distributions $\{\nu_j\}_{j=1}^m$ and weights $\{w_i\}_{i=1}^n \subset \Delta^m$ such that each data point μ_i is close to an entropic barycenter with reference measures $\{\nu_j\}_{j=1}^m$ and weights w_i; this may be thought of as extracting coefficients via unmixing in Wasserstein space. Specifically, we solve the following regularized Wasserstein dictionary learning (WDL) problem [13, 14, 20]:

$$\begin{aligned}(\{\nu_j^*\}_{j=1}^m, \{w_i^*\}_{i=1}^n) \\ = \arg\min_{\{\nu_j\}_{j=1}^m, \{w_i\}_{i=1}^n} \sum_{i=1}^{n} W^p_{p,\epsilon}(\text{Bary}(\{\nu_j\}_{j=1}^m; w_i), \mu_i) \\ + \rho\sum_{i=1}^{n}\sum_{j=1}^{m}[w_i]_j W^p_{p,\epsilon}(\mu_i,\nu_j),\end{aligned} \tag{3}$$

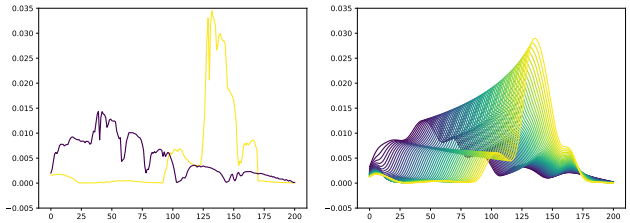
where $\rho > 0$ is a regularization parameter that balances the two terms. This non-convex optimization problem aims to reconstruct each observation as an entropic barycenter (first term), subject to a locality regularizer that promotes representing using nearby atoms (second term). This program can be approximately optimized using first-order methods that jointly optimize the atoms and weights; for details, we refer the reader to [13, 14].

In Figure 1, we compare the effectiveness of non-negative matrix factorization (NMF) [21] to WDL in representing the underlying geometry of the space. Two spectra from the Salinas A HSI (see Section 4) are used as the underlying atoms. We use these spectra to create entropic Wasserstein barycenters with interpolating weights from $(1-t, t) \in \Delta^2$ for $t \in \{0, 0.02, ...0.98, 1\}$ with $\epsilon = 0.001$. The WDL parameters of note for this example are $\epsilon = 0.001$ and $\rho = 0$.

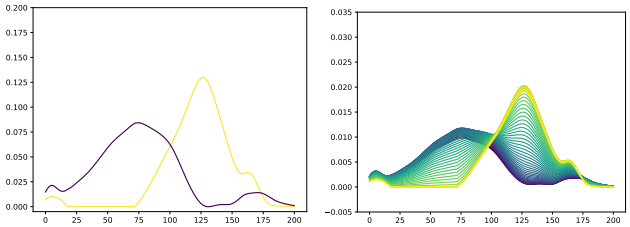
3. SEMISUPERVISED LEARNING ALGORITHM

Our approach has two steps: first, given HSI pixels $\{\mu_i\}_{i=1}^n$, learn coefficients $\{w_i\}_{i=1}^n$ via approximately solving (3); second, use the weights in combination with information about the unnormalized spectra to determine which points to query to label the rest of the pixels. Since Algorithm 1 requires the input data to be probability measures, we must normalize μ_i by its ℓ^1 norm. However, with this step, we lose information about the *scale* of the spectra. See how in Figure 2, the spectra reaching higher measured reflectances appear closer to the rest of the spectra when normalized. If two pixels μ_i and μ_j have similar learned coefficients w_i and w_j, then the normalized spectral signatures are similar, which potentially indicates the same groundtruth class. By incorporating the scale of the spectra into our approach, we prevent the issue with normalization presented in Figure 2.

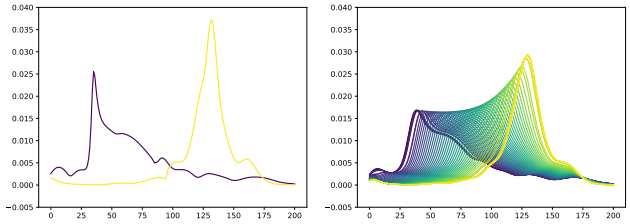
Concretely, our approach has two steps: first, learn weights via approximately solving (3), second use the weights in combination with information about the unnormalized spectra to determine which points to query to label the rest of the pixels. The key idea for approximately solving (3) is to use automatic differentiation to iteratively update the weights $\{w_i\}_{i=1}^n$ and atoms $\{\nu_j\}_{j=1}^m$; we note that the training data



(a) HSI spectra (left) and Wasserstein barycenters (right)



(b) NMF components (left) and reconstructions (right)



(c) Learned WDL atoms (left) and reconstructions (right)

Fig. 1. Unmixing result for a mixture created from HSI spectra. The second row contains the learned NMF components and reconstructions, and the third row contains the learned WDL atoms and reconstructions. NMF reconstructions fail to capture the smooth transformation from one atom to the other, while WDL interpolates meaningfully between the atoms.

$\{\mu_i\}_{i=1}^n$ is fixed in the learning process. The basic procedure is to: (i) initialize the weights $\{w_i\}_{i=1}^n$ uniformly at random from Δ^m and the atoms via Wasserstein K-means over the training data (ii) iteratively update via automatic differentiation on $\mathcal{G}$. This is outlined in Algorithm 1, where $\mathcal{G}$ is the right hand side of (3).

Algorithm 1: Geometric Wasserstein Dictionary Learning

1: **Input:** HSI spectra: $\{\mu_i\}_{i=1}^n \subset \mathbb{R}^D$; Wasserstein parameter: p; entropic regularization: ϵ; locality regularization: ρ; number iterations: L; number of atoms: m.
2: Normalize each pixel μ_i to lie in Δ^D.
3: Initialize variables $\boldsymbol{\alpha}^{(0)} \in \mathbb{R}^{m\times N}$, $\boldsymbol{\beta}^{(0)} \in \mathbb{R}^{n\times m}$.
4: **for** $k \leftarrow 1, \ldots, L$ **do**
5: $\quad \{\nu_j^{(k)}\}_{j=1}^m \leftarrow \sigma(\boldsymbol{\alpha}^{(\mathbf{0})}), \{w_i^{(k)}\}_{i=1}^n \leftarrow \sigma(\boldsymbol{\beta}^{(\mathbf{0})})$.
6: $\quad$ Compute the objective function `loss` $\leftarrow \mathcal{G}(\{\nu_j^{(k)}\}_{j=1}^m, \{w_i^{(k)}\}_{i=1}^n, \{\mu_i\}_{i=1}^n)$.
7: $\quad$ Compute the gradients with automatic differentiation: `loss.backward()`.
8: $\quad$ Update $\boldsymbol{\alpha}^{(k)}, \boldsymbol{\beta}^{(k)}$.
9: **end for**
10: **Output:** Learned atoms: $\{\nu_j\}_{j=1}^m \leftarrow \sigma(\boldsymbol{\alpha}^{(k)})$; learned weights: $\{w_i\}_{i=1}^n \leftarrow \sigma(\boldsymbol{\beta}^{(k)})$.

Since $\{\mu_i\}_{i=1}^n$ is constrained to be a probability measure, we use the softmax function σ to extend GeoWDL to data that cannot fit that form. For a vector $(x_1, x_2, ...x_n)$ the softmax is defined as:

$$\sigma(x_1, \ldots, x_n) := \left(\frac{\exp(x_1)}{\sum_{i=1}^n \exp(x_i)}, \ldots, \frac{\exp(x_n)}{\sum_{i=1}^n \exp(x_i)} \right).$$

We define the mass of μ_i as $\varphi_i := \|\mu_i\|_1$, before normalization to carry out Algorithm 1. We then define new weights $\{\tilde{w}_i\}_{i=1}^n \subset \mathbb{R}^{m+1}$ which are $\tilde{w}_i := (w_i, \varphi_i)$. We then renormalize $\tilde{w}_i$ by ℓ^1 norm to avoid numerical issues. Since $\varphi_i \gg \sum_i^m w_i$, no normalization often results in the non-mass terms being irrelevant, with the learned embedding resembling one similar to TSNE embeddings of already low-dimensional data.

From here, we use t-stochastic neighbors embedding (TSNE) to map $\{\tilde{w}_i\}_{i=1}^n$ from $\mathbb{R}^{m+1}$ to $\mathbb{R}^2$. TSNE is a widely used nonlinear dimensionality reduction technique for visualizing high dimensional data capable of also representing clusters present in the higher dimensional space [22]. After utilizing TSNE, the embedding is clustered using K-means with a greater number of clusters than known or anticipated to exist in the data. For a given cluster, we query for the labels of the N closest labeled points (η_i, y_i) to the centroid, which is a total of NK points. Each pixel in that cluster is labeled with the most common label among the queried points. The overall active labeling algorithm is detailed in Algorithm 2.

We will refer to steps 10 and 11 of Algorithm 2 as spatial relabeling and inpainting, respectively. Spatial relabeling changes the labels of pixels that are likely to be mislabeled by the active learning process. Given a pixel (η_i, y_i) where y_i is learned, spatial relabeling first finds the 10 ℓ^1 labeled spatial nearest neighbors of η_i. If among these neighbors, the number of pixels that are labeled with y_i is less than or equal to 1, y_i is relabeled to the most frequent label among its 10 nearest neighbors. Both steps are based on the idea that spatially, pixels with common labels are grouped together. This may not hold for all HSIs, but is reasonable in the context of Salinas A and farming related imagery.

Algorithm 2: Wasserstein Dictionary Active Learning

1: **Input:** Number of pixels: P; WDL weights: $\{w_i\}_{i=1}^P$; number of clusters: K; queries per cluster: N.
2: Append mass of $\{\mu_i\}_{i=1}^P$ to make new coefficients $\{\tilde{w}_i\}_{i=1}^P \subset \mathbb{R}^{m+1}$
3: Normalize $\{\tilde{w}_i\}_{i=1}^P$ to lie in Δ^{m+1} and get a 2d TSNE embedding of coefficients.
4: On the embedding, run K-means with K clusters.
5: **for** $i \leftarrow 1, \ldots, K$ **do**
6: $\quad$ Find the N closest labeled points within the embedding to the centroid of cluster i.
7: $\quad$ Determine the most frequent label, y among those points.
8: $\quad$ All pixels in cluster i are labeled with y
9: **end for**
10: Relabel any pixel that has at most 1 of its 10 nearest ℓ^1 neighbors with the same label to the most common label among them.
11: Label all remaining unlabeled pixels by the most common label among its 10 nearest ℓ^1 neighbors.
12: **Output:** Learned labels $\{y_i\}_{i=1}^n$

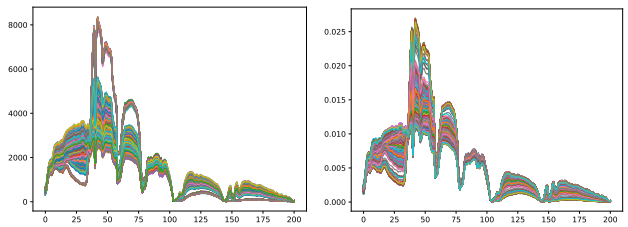


Fig. 2. *Left:* Spectra before normalization. *Right:* Spectra after normalization.

4. EXPERIMENTAL RESULTS

The Salinas A hyperspectral image was captured in 1998 by the AVIRIS sensor. The scene is an agricultural region in Sali-

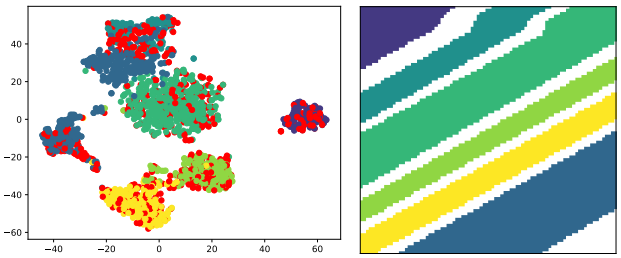


Fig. 3. *Left:* TSNE embedding pre K-means of Algorithm 2, where colors correspond to ground truth label. Red pixels are unlabelled. *Right:* Salinas A ground truth; white pixels are unlabelled.

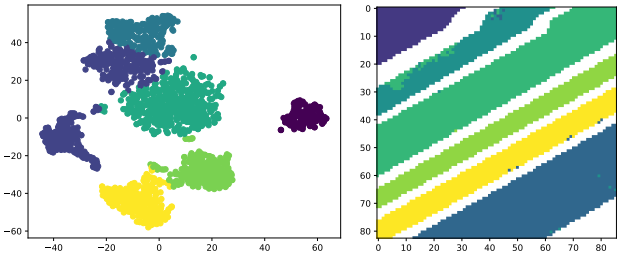


Fig. 4. *Left:* Learned labels with OA of 93%, *Right:* Post inpainting accuracy of 99%. $K = 8$, $N = 1$

nas Valley, CA, USA. It consists of 224 spectral bands ranging from 380-2500 nm. The 6 classes in the image are: broccoli greens; corn green weeds; and romaine lettuce at 4 different growth times, namely 4, 5, 6, and 7 weeks. The full image is 83×86; see Figure 2 for the spectra of a random sample of Salinas A pixels, and see Figure 3 for the spatial groundtruth of Salinas A. To carry out our experiment, we randomly sampled 2000 pixels from the full image, then ran Algorithm 1 to learn representations in Wasserstein space. We use $m = 32$, $L = 400$, $\epsilon = 0.08$, and $\rho = 0.0001$. For further discussion on WDL parameters and clustering results, we refer the reader to [23, 24]. We then run Algorithm 2 to learn labels, and use overall accuracy (OA) as a metric, namely the number of correctly labeled pixels divided by the total number of labeled pixels. Since the sample data consists of labeled and unlabeled points, OA only considers initially labeled pixels in its calculations. We also calculate an OA for before and after spatial inpainting/relabeling. We only carry out spatial inpainting on pixels with ground truth labels, but it can be easily extended to the whole image. The results we show are for $K = 8$ and $N = 1$, but code for experiments across a wide variety of parameters are public[1]. Experimentally, spatial relabeling never caused a decrease in accuracy. The increase is generally around $4 - 5\%$, though varies significantly. However, spatial relabeling performs better generally at higher initial OAs.

[1] https://github.com/fullenbs/WDL-active-learning

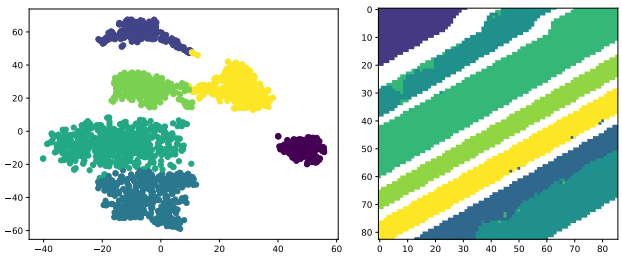


Fig. 5. *Left:* Learned embedding OA of 82%. *Right:* Post-relabeling/inpainting accuracy of 85%. $K = 8$, $N = 1$

Due to the non-convex cost function in TSNE, embeddings vary across different runs [22], which ultimately impacts the results of Algorithm 2. Figure 5 is another example embedding resulting from the same data as in Figure 3. Over a run of 20 trials and excluding spatial relabeling/inpainting, at $K = 8$ and $N = 1$, Algorithm 2 achieved an average accuracy of 85% with a standard deviation of 5%; the variance in results is due to the use of TSNE. As K increases, average accuracy increases and the standard deviation decreases. At $K = 20$ and $N = 1$, the average accuracy was 89.7% with a standard deviation of 1.6%. When including spatial relabeling and inpainting, OA for $K = 8$ and $N = 1$ goes to an average of 93.1% with a variance of 6.8%. For $K = 20$ and $N = 1$, OA average accuracy rises to 96.3% with a standard deviation of 2.8%.

5. CONCLUSION AND FUTURE WORK

This paper demonstrates a new semisupervised approach for HSI clustering based on representations of pixels in Wasserstein space. These coefficients provide a useful representation of hyperspectral images in a lower dimensional space which we utilize in our clustering scheme.

The non-convex nature of TSNE has an outsized impact on OA of our results. Developing a robust methodology that utilizes TSNE or another technique such as UMAP that minimizes variance across different runs of Algorithm 2 is an area of ongoing work.

Acknowledgments: The authors acknowledge partial support from NSF DMS 2309519 and NSF DMS 2318894.

6. REFERENCES

[1] G. Peyré and M. Cuturi, "Computational optimal transport: With applications to data science," *Foundations and Trends® in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.

[2] "Hyperspectral remote sensing scenes," https://www.ehu.eus/ccwintco/index.php/

Hyperspectral_Remote_Sensing_Scenes, Accessed: 2024-4-11.

[3] 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, 2017.

[4] B. Liu, X. Yu, P. Zhang, X. Tan, A. Yu, and Z. Xue, "A semi-supervised convolutional neural network for hyperspectral image classification," *Remote Sensing Letters*, vol. 8, no. 9, pp. 839–848, 2017.

[5] L. Ma, A. Ma, C. Ju, and X. Li, "Graph-based semi-supervised learning for spectral-spatial hyperspectral image classification," *Pattern Recognition Letters*, vol. 83, pp. 133–142, 2016.

[6] M. Chi and L. Bruzzone, "Semisupervised classification of hyperspectral images by svms optimized in the primal," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 6, pp. 1870–1880, 2007.

[7] J.M. Murphy and M. Maggioni, "Iterative active learning with diffusion geometry for hyperspectral images," in *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2018, pp. 1–5.

[8] M. Maggioni and J.M. Murphy, "Learning by active nonlinear diffusion," *Foundations of Data Science*, vol. 1, no. 3, pp. 271–291, 2019.

[9] J.M. Haut, M.E. Paoletti, J. Plaza, J. Li, and A. Plaza, "Active learning with convolutional neural networks for hyperspectral image classification using a new bayesian approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6440–6461, 2018.

[10] J. Li, J.M. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentation using a new bayesian approach with active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10, pp. 3947–3960, 2011.

[11] X. Cao, J. Yao, Z. Xu, and D. Meng, "Hyperspectral image classification with convolutional neural network and active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 7, pp. 4604–4616, 2020.

[12] B. Demir, C. Persello, and L. Bruzzone, "Batch-mode active-learning methods for the interactive classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 3, pp. 1014–1031, 2010.

[13] M.A. Schmitz, M. Heitz, N. Bonneel, F. Ngole, D. Coeurjolly, M. Cuturi, G. Peyré, and J.-L. Starck, "Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning," *SIAM Journal on Imaging Sciences*, vol. 11, no. 1, pp. 643–678, 2018.

[14] M. Mueller, S. Aeron, J.M. Murphy, and A. Tasissa, "Geometrically regularized Wasserstein dictionary learning," in *Topological, Algebraic and Geometric Learning Workshops*, 2023, pp. 384–403.

[15] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in neural information processing systems*, vol. 26, 2013.

[16] M. Nutz, "Introduction to entropic optimal transport," *Lecture notes, Columbia University*, 2021.

[17] M. Agueh and G. Carlier, "Barycenters in the Wasserstein space," *SIAM Journal on Mathematical Analysis*, vol. 43, no. 2, pp. 904–924, 2011.

[18] J. Solomon, F. De Goes, G. Peyré, M. Cuturi, A. Butscher, A. Nguyen, T. Du, and L. Guibas, "Convolutional wasserstein distances: Efficient optimal transportation on geometric domains," *ACM Transactions on Graphics*, vol. 34, no. 4, pp. 1–11, 2015.

[19] M. Werenski, R. Jiang, A. Tasissa, S. Aeron, and J.M. Murphy, "Measure estimation in the barycentric coding model," in *International Conference on Machine Learning*, 2022, pp. 23781–23803.

[20] M. Mueller, *Local Sparse Representations: Connections With the Delaunay Triangulation and Dictionary Learning in Wasserstein Space*, Ph.D. thesis, 2024.

[21] M.W. Berry, M. Browne, A.N. Langville, V.P. Pauca, and R.J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[22] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.

[23] S. Fullenbaum, M. Mueller, A. Tasissa, and J.M. Murphy, "Hyperspectral image clustering via learned representation in Wasserstein space," in *IEEE International Geosciences and Remote Sensing Symposium*, 2024, pp. 8791–8796.

[24] S. Fullenbaum, M. Mueller, A. Tasissa, and J.M. Murphy, "Nonlinear unmixing of hyperspectral images via regularized Wasserstein dictionary learning," in *IEEE International Geosciences and Remote Sensing Symposium*, 2024, pp. 8289–8294.