Spectral–Spatial Diffusion Geometry for Hyperspectral Image Clustering

James M. Murphy and Mauro Maggioni

***Abstract*— An unsupervised learning algorithm to cluster hyperspectral image (HSI) data that leverages spatially regularized random walks is proposed. Markov diffusions are defined on the space of HSI spectra with transitions constrained to near spatial neighbors. The explicit incorporation of spatial regularity into the diffusion construction leads to smoother random processes that are more adapted for unsupervised machine learning than those based on spectra alone. The regularized diffusion process is subsequently used to embed the high-dimensional HSI into a lower-dimensional space through diffusion distances. Cluster modes are computed using kernel density estimation and diffusion distances, and all other points are labeled according to these modes. The proposed method has low computational complexity and performs competitively against state-of-the-art HSI clustering algorithms on real data. In particular, the proposed spatial regularization confers both theoretical and empirical advantages over nonregularized methods.**

***Index Terms*— Graph theory, harmonic analysis, hyperspectral imaging, machine learning, unsupervised learning.**

I. Introduction

As THE volume of data captured by remote sensors grows unabated, the human capacity for providing labeled training data sets is strained. In order to take advantage of the deluge of unlabeled remote sensing data, new methods that are *unsupervised*—requiring no training data—are necessary.

This letter makes two contributions to the unsupervised analysis of high-dimensional remotely sensed hyperspectral images (HSIs). First, *spectral–spatial diffusion geometry* is proposed for remote sensing images. This allows for high-dimensional data to be analyzed in a manner that respects not only intrinsic pixel geometry in the data but also the spatial regularity in the 2-D image structure of the pixels. Second, we propose a new algorithm for efficient unsupervised clustering of HSI called spatially regularized diffusion learning (SRDL). This method integrates spectral–spatial diffusion geometry into the recently proposed diffusion learning (DL) algorithm, which has achieved competitive performance versus benchmark and state-of-the-art unsupervised HSI clustering algorithms [1], [2].

Manuscript received March 21, 2019; revised August 5, 2019; accepted September 18, 2019. Date of publication October 14, 2019; date of current version June 24, 2020. This work was supported by AFOSR under Grant FA9550-17-1-0280, Grant NSF-IIS-1546392, Grant NSF-ATD-1737984, Grant NSF-DMS 1912737, and Grant NSF-DMS 1924513. *(Corresponding author: James M. Murphy.)*

J. M. Murphy is with the Department of Mathematics, Tufts University, Medford, MA 02155 USA (e-mail: jm.murphy@tufts.edu).

M. Maggioni is with the Department of Mathematics, Johns Hopkins University, Baltimore, MD 21218 USA, and also with the Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: mauromaggionijhu@icloud.com).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2019.2943001

The remainder of this letter is organized as follows. Background on HSI clustering and diffusion geometry is presented in Section II. The proposed algorithm is described and evaluated in Sections III and IV, respectively. Conclusions and discussion are presented in Section V.

II. Background

A. Background on HSI Clustering

Unsupervised clustering of HSI data $X = \{x_i\}_{i=1}^n \subset \mathbb{R}^D$—understood as a point cloud—consists in providing labels $\{\hat{y}_i\}_{i=1}^n$ to each data point without access to labeled training data. The total number of pixels in the HSI is n and the data dimensionality is D, corresponding to the number of spectral bands. The data dimension D is large for HSI, rendering standard clustering methods such as K-means clustering, Gaussian mixture models (GMMs) [3], and density-based methods [4], [5] ill-suited for HSI, due to the "curse of dimensionality" [3]. Moreover, linear dimension reduction methods such as principal component analysis (PCA) [3], independent component analysis (ICA) [6], and random projections (RPs) [7] are of limited use for HSI clustering, because they may wash out geometric information that discriminates between distinct clusters.

Fortunately, clusters in HSI typically exhibit intrinsically low-dimensional structure, which a range of methods have been proposed to capture, including matrix factorizations [8], sparse subspace learning [9], [10], and manifold learning methods based on graph Laplacians [11]–[14]. Manifold learning methods have also been applied to the related unsupervised problems of anomaly and target detection [15]–[19]. However, methods based on matrix factorizations and subspace learning may struggle to learn clusters that lack subspace structure (i.e., exhibit nonlinear structure), while manifold learning methods tend to be overly sensitive to outliers and may not account for spatial structure in the HSI. It is thus critical to develop clustering methods that not only exploit the manifold structure of the HSI but also its spatial regularity.

B. Background on Diffusion Geometry

This letter proposes the *diffusion geometry* [20], [21] of an HSI as a method to infer clusters. Diffusion geometry is captured through *diffusion distances*, which are driven by time-dependent Markov processes on the underlying data. The *diffusion distance at time t* between $x_i, x_j \in X$, denoted $d_t(x_i, x_j)$, is a notion of distance driven by latent, low-dimensional geometry in the point cloud X. Intuitively, points with many short paths in X connecting them will be close in diffusion distance. Unlike Euclidean distance, diffusion distances are data-dependent and account for the global structure of X when making comparisons between points. Moreover, unlike shortest-path distances, diffusion distances are robust to noise and outliers [21], [22].

Diffusion distances are computed via a weighted, undirected graph $\mathcal{G}$ with vertices $X = \{x_i\}_{i=1}^n$ and edges stored in the weight matrix $\mathbf{W}_{ij} = \exp(-\|x_i - x_j\|_2^2/\epsilon^2)$ if $x_i \in NN(x_j; k)$, $\mathbf{W}_{ij} = 0$ otherwise for some scaling parameter ϵ, and with $NN(x_i; k)$, the set of k-nearest neighbors of x_i in X measured in the ℓ^2 metric. Typically, ϵ is chosen adaptively [23], and $k \ll n$ is chosen so that $\mathbf{W}$ is sparse; we set $k = 100$ in all experiments. Let $\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}$ be a Markov diffusion matrix defined on X, where $\mathbf{D}$ is the diagonal degree matrix with $\mathbf{D}_{ii} = \sum_{\ell=1}^n \mathbf{W}_{i\ell}$. Assume $\mathbf{P}$ is irreducible and aperiodic. The *diffusion distance at time t* is

$$d_t(x_i, x_j) = \sqrt{\sum_{\ell=1}^{n} \left(\mathbf{P}_{i\ell}^t - \mathbf{P}_{j\ell}^t\right)^2 / \pi(\ell)} \tag{1}$$

where π satisfies $\pi\mathbf{P} = \pi$. The computation of $d_t(x_i, x_j)$ involves comparing the transition probabilities of x_i and x_j after t time steps, so $d_t(x_i, x_j)$ is small if x_i, x_j have similar probabilistic behavior according to $\mathbf{P}^t$. The parameter t is the time scale of the diffusion process on X: small values of t correspond to small amounts of diffusion, which may prevent the exploration of macroscopic geometry of X. On the other hand, large t may ruin the fine geometry of X through homogenization. In Section IV, $t = 30$; see [2] and [22] for empirical and theoretical analyses of t, respectively.

Under mild assumptions [21], $\mathbf{P}$ has (right) eigenvectors $\{\psi_i\}_{i=1}^n$ with real eigenvalues $\{\lambda_i\}_{i=1}^n$ ordered such that $1 = \lambda_1 > |\lambda_2| \geq \cdots \geq |\lambda_n|$. This yields

$$d_t(x_i, x_j) = \sqrt{\sum_{\ell=1}^{n} \lambda_\ell^{2t} (\psi_\ell(x_i) - \psi_\ell(x_j))^2}. \tag{2}$$

Note that (2) may be truncated at some $2 \leq m \ll n$ while retaining good accuracy. This reduces computational complexity, since only the first $m = O(1)$ eigenpairs need to be computed. In our experiments, m was set to be the "elbow" value [2] of the plot of the eigenvalues $\{\lambda_i\}_{i=1}^n$.

C. Background on Spatial Regularity in HSI

If X has a structure beyond its D-dimensional spectral coordinates, this may be incorporated into the diffusion maps construction by modifying the underlying transition matrix $\mathbf{P}$. In the case of HSI, each point is not only a high-dimensional spectrum but also a pixel arranged in an image. In particular, many HSIs enjoy *spatial regularity*, in the sense that points in a particular cluster are likely to have their nearest spatial neighbors in the same cluster. This is particularly true for natural and agricultural scenes, or for scenes with high spatial resolution. Spatial regularity is powerful a priori information that can be accounted for in the construction of statistical and machine learning algorithms. Indeed, $\mathbf{P}$ encodes pixelwise similarities to infer latent structure in the data; it is thus natural to incorporate meaningful spatial structure into $\mathbf{P}$.

In this letter, we extend the recently proposed DL unsupervised clustering framework [2] by directly incorporating spatial information into the underlying diffusion matrix $\mathbf{P}$. Incorporating spatial information into machine learning for remote sensing data has been helpful in supervised [24] and unsupervised contexts [9], [10], [25]. HSI clustering methods benefit from spatial regularization not only in smoothing out speckling errors but also in separating classes that overlap spectrally, which is a common problem for HSI and in general for high-dimensional data corrupted by noise.

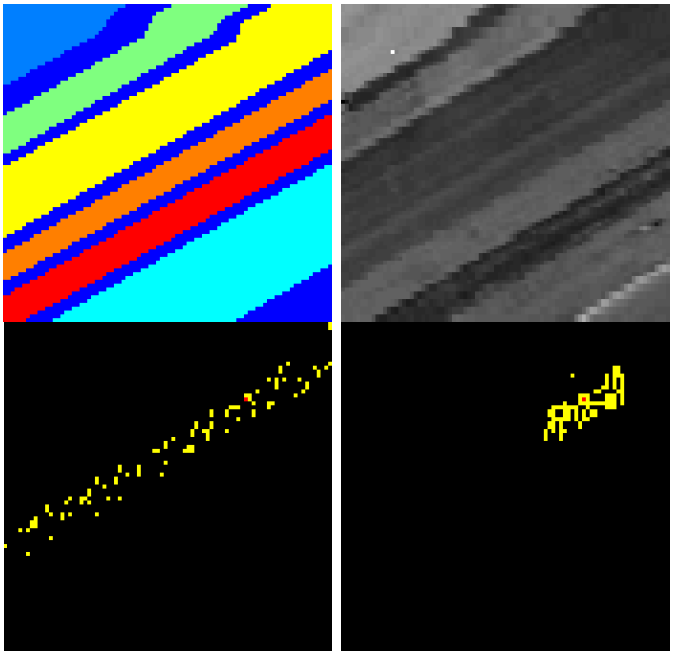


Fig. 1. The Salinas A data set was collected over Salinas Valley, CA, USA. It has spatial dimensions 86×83 and consists of 224 spectral bands with spatial resolution 3.7 m/pixel. In order to differentiate between certain pixels that have the same value, Gaussian noise with variance $= 10^{-4}$ was added during preprocessing of the HSI. (Top left) ground truth (GT), showing six classes. (Top right) Sum across all spectral bands. (Bottom left) 100 nearest neighbors (yellow) of a pixel (red) without spatial regularization. (Bottom right) 100 nearest neighbors (yellow) of a pixel (red) with spatial regularization. The spatial regularization forces the random walk $\mathbf{P}$ to converge to mesoscopic equilibria on distinct classes more rapidly, which improves the discriminatory power of diffusion distances for unsupervised learning [22].

III. Spatially Regularized Diffusion Learning

Intuitively, SRDL computes cluster *modes*, which are well-separated in both the spectral and spatial domains. These modes are representative of the distinct clusters in the data, and all other pixels are labeled according to these modes.

The proposed algorithm first constructs a Markov diffusion matrix, $\mathbf{P}$, under the constraint that pixels may only be connected to other pixels that are within some spatial radius R; see Fig. 1 and Algorithm 1. Mathematically, the incorporation of spatial proximity accelerates the mixing of the Markov transition matrix $\mathbf{P}$ within a spatial cluster and inhibits between-cluster mixing, which improves the clustering properties of diffusion distances [22]. The eigenpairs of $\mathbf{P}$ with largest eigenvalues in modulus are computed, so that diffusion distances are simply Euclidean distances in the new coordinate system $x \mapsto (\lambda_1^t \psi_1(x), \ldots, \lambda_m^t \psi_m(x))$ as in (2).

Algorithm 1 Spectral–Spatial Diffusion Maps

1 *Input*: X, R.
2 Connect each $x \in X$ to its $k = 100$ nearest neighbors within spatial radius R, call them y, with weight $\exp(-\|x - y\|_2^2/\epsilon^2)$, with ϵ set adaptively [23].
3 Let $\mathbf{D}$ be the diagonal degree matrix and $\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}$.
4 Compute the m (right) eigenpairs of $\mathbf{P}$ with largest (in modulus) eigenvalues, $\{(\lambda_i, \psi_i)\}_{i=1}^m$.
5 *Output:* $\{(\lambda_i, \psi_i)\}_{i=1}^m$.

The proposed algorithm first learns cluster modes as the maximizers of $\mathscr{D}_t(x) = f(x)\delta_t(x)$, where $f(x)$ is a kernel density estimator (KDE) [3] and $\delta_t(x)$ is the diffusion distance of a point to its nearest neighbor of higher f value (unless x is the global density maximizer, in which case $\delta_t(x) = \max_{y \in X} d_t(x, y)$). The mode detection algorithm is summarized in Algorithm 2; see [2] for details.

Algorithm 2 Spectral–Spatial Mode Estimation

Input: X, R, K.
Compute spectral-spatial diffusion distances using Algorithm 1 and (2).
For each $x_i \in X$, compute a KDE $f(x_i)$.
For each $x_i \in X$, compute $\delta_t(x_i)$.
Set $\{z_i\}_{i=1}^K$ to be the K maximizers of $\mathscr{D}_t(x_i) = f(x_i)\delta_t(x_i)$.
Output: $\{f(x_i)\}_{i=1}^n, \{z_i\}_{i=1}^K$.

From the modes, the rest of the data is labeled in a two-phase process. Points are labeled iteratively—from highest f value to lowest f value—to have the same label as their nearest spectral neighbor of higher f value that has already been labeled, unless it is the case that such a labeling violates smoothness of the spatial labels. If spatial smoothness of a point x_i is violated, x_i is not labeled in the first phase. In the second phase, an unlabeled point is given the same label as the most common label among its spatial nearest neighbors; we call this the *consensus spatial label*. The spectral–spatial labeling scheme is summarized in Algorithm 3, and its crucial parameters and the role of spatial consensus labels are discussed at length in [2].

The proposed method—described in Algorithm 3—is called *spatially regularized diffusion learning* (SRDL).

Algorithm 3 Spatially Regularized Diffusion Learning (SRDL)

Input: X, R, K.
Compute $\{(\lambda_i, \psi_i)\}_{i=1}^m$ using Algorithm 1.
Compute $\{f(x_i)\}_{i=1}^n, \{z_i\}_{i=1}^K$ using Algorithm 2.
For $i = 1, \ldots, K$, give z_i the label i.
Labeling Phase 1: In order of decreasing f value, give each x the label of its d_t-nearest spectral neighbor of higher density, unless the spatial consensus label exists and differs, in which case the point is not labeled.
Labeling Phase 2: In order of decreasing f value, give each unlabeled x its consensus spatial label, if it exists, otherwise the same label as its d_t-nearest spectral neighbor of higher density.
Output: Learned labels $\{\hat{y}_i\}_{i=1}^n$.

IV. Empirical Analysis of SRDL Algorithm

The SRDL algorithm is evaluated on the publicly available[1] Kennedy Space Center, Indian Pines, and Salinas A data sets. The Kennedy Space Center and Indian Pines data sets are cropped to subsets, due to well-documented challenges of unsupervised learning for data with a large number of classes [14]. Since it contains only six classes, the full Salinas A data set was used. While the proposed SRDL algorithm automatically estimates the number of clusters based on the decay of $\mathscr{D}_t$, the number of class labels in the GT images were used as a parameter K for all clustering algorithms to make a fair comparison with methods that cannot reliably estimate the number of clusters. Experiments are performed on the entire data set, including points without GT labels; only pixels with GT labels are used for quantitative evaluation. Three metrics are used for quantitative comparison of a clustering with the GT. Overall accuracy (OA) is the ratio of correctly labeled pixels to the total number of pixels. Average accuracy (AA) is the average of the OA of each class, which equalizes the significance of small and large classes. *Cohen's κ-statistic* (κ) measures agreement across two labeling in a manner robust to a random chance.

[1] http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

A. Benchmark and State-of-the-Art Comparison Methods

We consider 13 methods of HSI clustering for comparison. The benchmark methods are: *K-means* [3] applied to the raw data X; PCA followed by K-means; ICA [6] followed by K-means; Gaussian random projections (RPs) followed by K-means [7]; spectral clustering (SC) [26]; Gaussian mixture models (GMMs) [3] with parameters determined by expectation maximization; and density-based spatial clustering of applications with noise (DBSCAN) [4]. All dimension reduction methods project into a number of dimensions equal to the number of GT classes K.

Several state-of-the-art HSI clustering methods are also considered: hierarchical clustering with nonnegative matrix factorization (HCNMF) [8]; sparse manifold clustering and embedding (SMCE) [12]; a Merriman–Bence–Osher (MBO) [27] method for minimizing a Mumford–Shah (MS) functional on a graph [13] denoted *MBOMS*; the density peaks clustering (DPC) algorithm [5]; and two variants of the recently proposed DL algorithm, in which the labeling process considers only spectral information (DL) or both spectral and spatial information (DLSS) [2].

Among these methods, DL and DLSS bear the closest resemblance to SRDL. These two methods and SRDL differ critically in how the underlying geometry for clustering is learned. In DL and DLSS, $\mathbf{P}$ considers the HSI only as a spectral point cloud. SRDL regularizes the construction of $\mathbf{P}$ by incorporating spatial information into the nearest neighbors construction. The proposed SRDL method also bears similarity to SC, SMCE, and MBOMS since all these methods use data-driven graphs. The DPC algorithm uses a mode detection scheme similar to SRDL, but with neither diffusion geometry nor spatial information.

B. Experimental Results

The *Kennedy Space Center (K.S.C.)* data set used for experiments is shown in Fig. 2 along with GT. Clustering results appear in Table I; for reasons of space, visual results are not shown. SRDL was run with $R = 20$. SRDL gives the best results, and in particular outperforms the spatially unregularized DLSS.

The *Indian Pines (I.P.)* data used for our experiments appears in Fig. 3. Visual results are shown in Fig. 4 and quantitative results are provided in Table I. SRDL was run

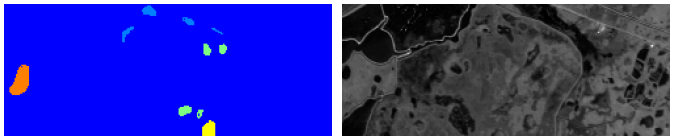


Fig. 2. The Kennedy Space Center data set was collected in FL, USA. It consists of 176 spectral bands at 18 m/pixel spatial resolution. To make the data suitable for unsupervised learning, a 250 × 100 subset of the full data set is used for experiments. (Left) GT, showing four classes. (Right) Projection onto first principal component.

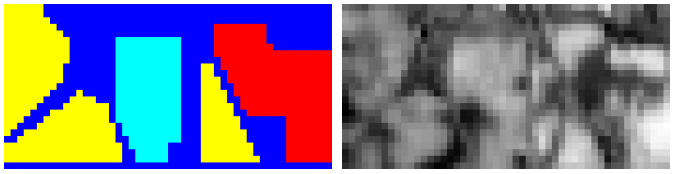


Fig. 3. The Indian Pines data was collected in IN, USA. It consists of 200 spectral bands at 20 m/pixel spatial resolution. To make the data suitable for unsupervised learning, a 50 × 25 subset of the full data set is used. (Left) GT, showing three classes. (Right) Sum across all spectral bands.

TABLE I

QUANTITATIVE RESULTS FOR CLUSTERING HSI ARE SHOWN WITH THE STRONGEST PERFORMER SHOWN IN BOLD, AND THE SECOND STRONGEST PERFORMER UNDERLINED. SRDL PERFORMS BEST ACROSS ALL DATA SETS AND METRICS, AND IN PARTICULAR OUTPERFORMS THE DLSS ALGORITHM, WHICH LACKS SPATIAL REGULARIZATION IN THE UNDERLYING DIFFUSION PROCESS. WE REMARK THAT EVEN IF R IS SET TO A DELIBERATELY SUBOPTIMAL VALUE, STRONG RESULTS ARE ACHIEVED BY SRDL (e.g., K.S.C.: $R = 10$, OA $= 0.77$; IP: $R = 4$, OA $= 0.86$; S.A.: $R = 12$, OA $= 0.90$); SEE FIG. 7 FOR ADDITIONAL ANALYSIS OF R

Algorithm	K.S.C. OA	K.S.C. AA	K.S.C. κ	I.P. OA	I.P. AA	I.P. κ	S.A. OA	S.A. AA	S.A. κ
K-means	0.36	0.25	0.01	0.43	0.38	0.09	0.63	0.66	0.52
PCA+KM	0.36	0.25	0.01	0.43	0.38	0.10	0.63	0.66	0.52
ICA+KM	0.36	0.25	0.01	0.41	0.36	0.06	0.57	0.56	0.44
RP+KM	0.60	0.50	0.43	0.51	0.51	0.26	0.63	0.66	0.53
SC	0.62	0.52	0.44	0.54	0.45	0.24	0.83	0.88	0.80
GMM	0.42	0.31	0.10	0.44	0.35	0.02	0.64	0.61	0.55
DBSCAN	0.36	0.25	0.01	0.63	0.62	0.43	0.71	0.71	0.63
HCNMF	0.36	0.25	0.00	0.41	0.32	-0.02	0.63	0.66	0.53
SMCE	0.36	0.26	0.01	0.52	0.45	0.22	0.47	0.42	0.30
MBOMS	0.74	0.70	0.65	0.57	0.50	0.27	0.70	0.81	0.65
DPC	0.36	0.25	0.00	0.58	0.51	0.26	0.63	0.61	0.54
DL	0.81	0.72	0.74	0.67	0.62	0.44	0.83	0.88	0.79
DLSS	0.83	0.73	0.76	0.85	0.82	0.75	0.85	0.90	0.81
SRDL	**0.85**	**0.75**	**0.79**	**0.89**	**0.92**	**0.83**	**0.90**	**0.93**	**0.87**

with $R = 8$. The spatial regularization in the construction of $\mathbf{P}$ leads to a smoother labeling, and SRDL improves over DLSS. However, a mistake is still made in the labeling of the proposed method, indicating that this is a challenging data set to cluster without supervision.

The *Salinas A (S.A.)* data set is shown in Fig. 1, with visual clustering results shown in Fig. 5 and quantitative accuracy results given in Table I. The proposed SRDL method was run with $R = 20$. SRDL yields the best results, and moreover, the labels recovered by the proposed method are quite spatially regular. This is likely due to the smooth spatial structure of the Salinas A crop rows, which SRDL takes advantage of by incorporating spatial proximity into the underlying Markov process.

As mentioned above, unsupervised clustering algorithms often struggle to meaningfully cluster HSI with many distinct clusters [14]. However, HSI with large numbers of classes (e.g., the full Indian Pines data set, which has 16 classes) may be segmented by partitioning the image into smaller regions, then clustering each subimage separately [2]. In the case of partitioning the Indian Pines HSI into 16 equal-sized subimages and evaluating each subimage separately before combining, SRDL (OA $= 0.8199$) substantially outperforms the next best competitor, DLSS (OA $= 0.7485$). See Fig. 6;

TABLE II

RUN TIMES IN SECONDS FOR EACH METHOD AND DATA SET. THE MBOMS METHOD WAS RUN IN C, WHILE ALL OTHER METHODS WERE RUN IN MATLAB ON A 2.7 GHz, INTEL CORE i7 PROCESSOR, AND 16 GB OF RAM

Algorithm	K-means	PCA+KM	ICA+KM	RP+KM	SC	GMM	DBSCAN	HCNMF	SMCE	MBOMS	DPC	DL	DLSS	SRDL
K.S.C.	4.10	.13	1.09	.34	178.69	6.41	112.4	.97	1315.2	.89	175.76	203.61	327.38	336.23
I.P.	.13	.01	.11	.05	32	.12	58	.19	3.02	.15	23	53	1.48	1.51
S.A.	.64	.01	.23	.17	7.08	1.13	13.53	.44	127.61	.29	6.88	8.88	19.17	18.97

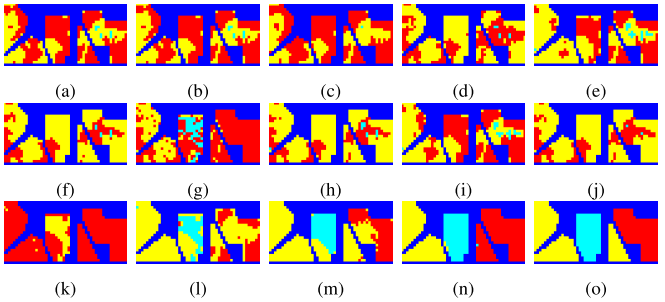


Fig. 4. On the Indian Pines HSI, the SRDL method leads to quite smooth spatial labels and has the accuracy that is optimal among all methods. However, in this case, the GT indicates that the triangular region on the lower right is labeled incorrectly by the proposed method. The smoothing imposed by SRDL—though beneficial overall—washes that region out. This weakness could be resolved in a variety of ways, perhaps most easily by oversegmenting the HSI, then querying the oversegmented class modes to determine which classes ought to be merged a posteriori. (a) K-means. (b) PCA+KM. (c) ICA+KM. (d) RP+KM. (e) SC. (f) GMM. (g) DBSCAN. (h) HCNMF. (i) SMCE. (j) MBOMS. (k) DPC. (l) DL. (m) DLSS. (n) SRDL. (o) GT.

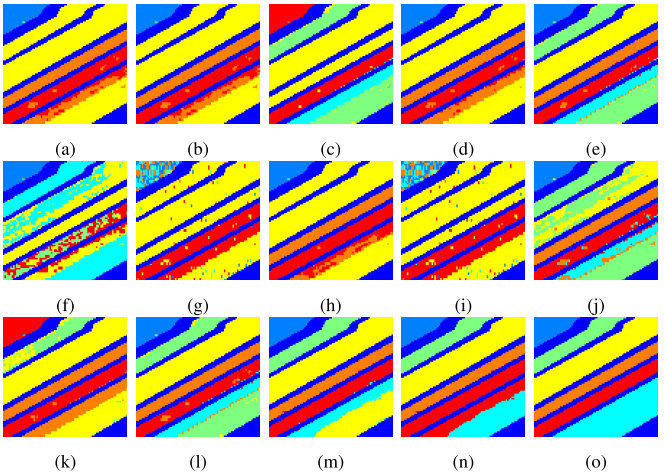


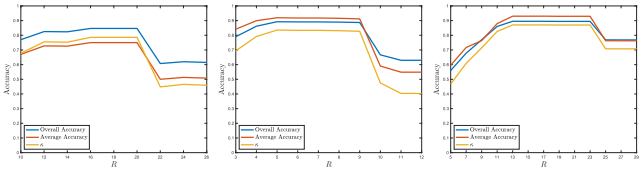
Fig. 5. For the Salinas A data set, SRDL is the optimal performer, with the DLSS, DL, and SC methods also performing strongly. The spatial regularization incorporated into the diffusion distances used for the proposed method keeps the diagonal stripes relatively far apart from each other, leading to accurate mode estimation and subsequent labeling. (a) K-means. (b) PCA+KM. (c) ICA+KM. (d) RP+KM. (e) SC. (f) GMM. (g) DBSCAN. (h) HCNMF. (i) SMCE. (j) MBOMS. (k) DPC. (l) DL. (m) DLSS. (n) SRDL. (o) GT.

we remark that recombining the clusters learned on the subimages in a consistent manner is a topic of the ongoing research.

Regarding *computational complexity*, it suffices to note that the bottleneck is in the construction of $\mathbf{P}$ (since $\mathbf{P}$ is sparse and only $m = O(1)$ eigenpairs are required). Since k nearest neighbors are sought and neighbors are constrained to live within a spatial radius R, as long as $R, k = O(1)$ with respect to n, the nearest neighbor searches for all points can be done in $O(n)$. This gives an overall complexity for the algorithm that is essentially linear in n. When $R = \Omega(n)$, indexing structures (e.g., k–d trees or cover trees) allow for fast nearest neighbor searches, yielding an algorithm quasilinear in n. The runtimes

(a) (b) (c) (d) (e)

Fig. 6. The full Indian Pines GT and sum of all spectral bands are in (a) and (b), respectively. Segmenting with SRDL (see [e]) leads to smoother and more accurate partitions than with K-means (see [c]) and the second best performer, DLSS (see [d]).



(a) (b) (c)

Fig. 7. Impact of R with $k = 100$ nearest neighbors. As R increases, empirical results improve then decline, illustrating that the optimal spatial regularization is to employ a moderate R value. Once a good R has been found, results are relatively robust. (a) Kennedy Space Center. (b) Indian Pines. (c) Salinas A.

for all methods appear in Table II, indicating that SRDL runs in similar time to SC, DL, and DLSS, and is faster than SMCE.

The crucial parameter in the proposed method is the spatial radius R, which determines how near the nearest neighbors in the underlying diffusion process must be. The impact of this parameter in terms of OA, AA, and κ is shown in Fig. 7. The plots exhibit the tradeoff typical of regularization in machine learning: insufficient or excessive regularization are both detrimental. The flat regions near the maxima in Fig. 7 suggest the proposed method is robust to the choice of R.

V. Conclusion and Discussion

Incorporating spatial regularity into the underlying diffusion geometry improves the empirical performance of DL for HSI clustering in all data sets considered. Our results suggest that for images whose labels are sufficiently smooth, there will be a regime of spatial window R in which incorporating spatial proximity improves the underlying mode detection and consequent labeling of HSI.

On the other hand, for images with many classes that are rapidly varying in space—for example, urban HSI—the proposed spatial regularization may be unhelpful or even detrimental. This is because if pixels that are nearby spatially do not have a high propensity to be in the same cluster, then regularizing the random walk spatially may negatively constrain its ability to quickly discover meaningful clusters. Indeed, motifs that are rapidly changing spatially but common throughout an image (for example urban buildings and streets) may be better learned without the spatial constraint.

We remark that an alternative approach to incorporating spatial information is to consider as underlying data points not individual pixels, but higher-order features, for example, image patches. This would integrate into the diffusion process information about detail features such as edges and textures, allowing for these fine-scale structures to be learned.

References

[1] J. M. Murphy and M. Maggioni, "Diffusion geometric methods for fusion of remotely sensed data," *Proc. SPIE*, vol. 10644, May 2018, Art. no. 106440I.

[2] J. M. Murphy and M. Maggioni, "Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1829–1845, Mar. 2019.

[3] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[4] M. Ester, H. P. Kriegel, J. Sander, and X. A. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, 1996, vol. 96, no. 34, pp. 226–231.

[5] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.

[6] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Netw.*, vol. 13, nos. 4–5, pp. 411–430, Jun. 2000.

[7] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2001, pp. 245–250.

[8] N. Gillis, D. Kuang, and H. Park, "Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2066–2078, Apr. 2015.

[9] H. Zhang, H. Zhai, L. Zhang, and P. Li, "Spectral–spatial sparse subspace clustering for hyperspectral remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3672–3684, Jun. 2016.

[10] H. Zhai, H. Zhang, L. Zhang, and P. Li, "Total variation regularized collaborative representation clustering with a locally adaptive dictionary for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 166–180, Jan. 2019.

[11] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Exploiting manifold geometry in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 441–454, Mar. 2005.

[12] E. Elhamifar and R. Vidal, "Sparse manifold clustering and embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 55–63.

[13] Z. Meng, E. Merkurjev, A. Koniges, and A. Bertozzi, "Hyperspectral video analysis using graph clustering methods," *Image Process. On Line*, vol. 7, pp. 218–245, May 2017.

[14] W. Zhu *et al.*, "Unsupervised classification in hyperspectral imagery with nonlocal total variation and primal-dual hybrid gradient algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2786–2798, May 2017.

[15] B. Basener, E. J. Ientilucci, and D. W. Messinger, "Anomaly detection using topology," *Proc. SPIE*, vol. 6565, May 2007, Art. no. 65650J.

[16] A. K. Ziemann and D. W. Messinger, "An adaptive locally linear embedding manifold learning approach for hyperspectral target detection," *Proc. SPIE*, vol. 9472, 2015, Art. no. 94720O.

[17] A. K. Ziemann, J. Theiler, and D. W. Messinger, "Hyperspectral target detection using manifold learning and multiple target spectra," in *Proc. IEEE Appl. Imag. Pattern Recognit. Workshop*, Oct. 2015, pp. 1–7.

[18] C.C. Olson and T. Doster, "A novel detection paradigm and its comparison to statistical and kernel-based anomaly detection algorithms for hyperspectral imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 108–114.

[19] C. C. Olson, K. P. Judd, and J. M. Nichols, "Manifold learning techniques for unsupervised anomaly detection," *Expert Syst. Appl.*, vol. 91, pp. 374–385, Jan. 2018.

[20] R. R. Coifman *et al.*, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proc. Nat. Acad. Sci. USA*, vol. 102, no. 21, pp. 7426–7431, 2005.

[21] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, Jul. 2006.

[22] M. Maggioni and J. M. Murphy, "Learning by unsupervised nonlinear diffusion," Dec. 2018, *arXiv:1810.06702*. [Online]. Available: https://arxiv.org/abs/1810.06702

[23] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 1601–1608.

[24] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, Mar. 2013.

[25] N. D. Cahill, W. Czaja, and D. W. Messinger, "Schroedinger eigenmaps with nondiagonal potentials for spatial-spectral clustering of hyperspectral imagery," *Proc. SPIE*, vol. 9088, Jun. 2014, Art. no. 908804.

[26] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 14, 2001, pp. 849–856.

[27] B. Merriman, J. K. Bence, and S. J. Osher, "Motion of multiple junctions: A level set approach," *J. Comput. Phys.*, vol. 112, no. 2, pp. 334–363, 1994.