FSPE: Visualization of Hyperspectral Imagery Using Faithful Stochastic Proximity Embedding

Safa A. Najim, *Student Member, IEEE*, Ik Soo Lim, Peter Wittek, and Mark W. Jones

***Abstract*—Hyperspectral image visualization reduces color bands to three, but prevailing linear methods fail to address data characteristics, and nonlinear embeddings are computationally demanding. Qualitative evaluation of embedding is also lacking. We propose faithful stochastic proximity embedding (FSPE), which is a scalable and nonlinear dimensionality reduction method. FSPE considers the nonlinear characteristics of spectral signatures, yet it avoids the costly computation of geodesic distances that are often required by other nonlinear methods. Furthermore, we employ a pixelwise metric that measures the quality of hyperspectral image visualization at each pixel. FSPE outperforms the state-of-art methods by at least 12% on average and up to 25% in the qualitative measure. An implementation on graphics processing units is two orders of magnitude faster than the baseline. Our method opens the path to high-fidelity and real-time analysis of hyperspectral images.**

***Index Terms*—Dimension reduction methods, hyperspectral imagery sensing, visualization.**

I. Introduction

HYPERSPECTRAL images are useful in examining the compounds and elements on the surface of the Earth, which are important for various geoscience-related applications. However, it is challenging to process and analyze hyperspectral images due to the high dimensionality of hundreds of spectral bands (> 200) per pixel position. Thus, prior to the classification or visualization of hyperspectral images, one often needs to convert high-dimensional data to a lower dimension while preserving the main features of the original space.

For this dimensionality reduction, principal component analysis (PCA) is one of the most widely used methods [1], [2]. However, assuming the linearity of the underlying manifold, PCA and similar linear projection methods fail to capture the nonlinear characteristics of hyperspectral imagery. Thus, these methods are prone to significant estimation errors [3]. To avoid the pitfalls of linear methods, nonlinear dimensionality reduction methods have attracted much attention. Examples include isometric feature mapping (ISOMAP, see [3] and [4]), and locally linear embedding (LLE, see [5] and [6]), which preserve the nonlinear features of the underlying manifold.

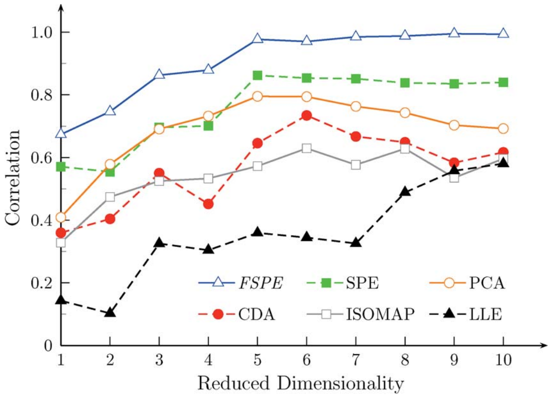


Fig. 1. Inevitable distortions in reducing the dimensionality to three or less. The dimensionality of a hyperspectral image is reduced to $10, 9, \ldots, 3, 2$, or 1 using various methods of dimensionality reduction. The quality of the dimensionality reduction is measured with a correlation metric between interpoint distances in the input data space and the projected space [1], [9]; the higher the correlation, the better. Regardless of the methods used for dimensionality reduction, the correlation at dimension 3 is lower than that at 10, implying inevitable distortions in the color display of hyperspectral images.

Dimensionality reduction for classification or segmentation applications seeks a low and intrinsic dimensionality, for instance, ten bands [5]. Dimensionality reduction for the color display of hyperspectral images is even more constrained: the reduced dimensionality cannot be more than three, i.e., the number of color bands, regardless of the data characteristics [1], [7]. By intrinsic dimensionality we mean the minimal number of variables necessary to account for all the pairwise distances in the data [8]. When the intrinsic dimensionality of hyperspectral images is higher than three, the dimensionality reduction for the color display inevitably leads to distortions or loss of spectral information and, thus, to unfaithful colors. For instance, distant points in the high-dimensional data space might be projected to nearby points in the color space. In other words, pixels of dissimilar spectral signatures will have similar colors, making the visualization unfaithful or unreliable. Fig. 1 shows that a reduction to three dimensions or less leads to lower accuracy than a reduction to higher dimensions. The resulting color display will contain inevitable distortions and misleading colors.

By proposing faithful stochastic proximity embedding (FSPE), we reduce misleading colors to make the color display of hyperspectral images more reliable or faithful. FSPE is a

Manuscript received October 15, 2013; revised January 15, 2014; accepted March 25, 2014. Date of publication June 20, 2014; date of current version August 14, 2014.

S. A. Najim is with the School of Computer Science, College of Physical and Applied Sciences, Bangor University, LL57 2DG Gwynedd, U.K., and also with the Department of Computer Science, College of Science, Basrah University, Basrah, Iraq (e-mail: safanajim@gmail.com).

I. S. Lim is with the School of Computer Science, College of Physical and Applied Sciences, Bangor University, LL57 2DG Gwynedd, U.K. (e-mail: i.s.lim@bangor.ac.uk).

P. Wittek is with the University of Borås, 501 90 Borås, Sweden (e-mail: peterwittek@acm.org).

M. W. Jones is with Swansea University, SA2 8PP Swansea, U.K. (e-mail: m.w.jones@swansea.ac.uk).

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

Digital Object Identifier 10.1109/LGRS.2014.2324631

novel dimensionality reduction method that leads to a more faithful color display than other popular methods such as PCA, ISOMAP, and LLE. Apart from a higher quality embedding, FSPE also has a computational advantage: it is easy to parallelize, and graphics processing unit (GPU)-based implementation enables interactive applications. Furthermore, we use a pixelwise metric of good visualization. Unlike the other metrics that only provide a global measure of the whole visualization with a single number [1], [9], the pixelwise metric provides a quantitative measure for each pixel of the hyperspectral images and tells us how reliable the color is per pixel. This measure is a more stringent criterion to estimate the quality of the projection. FSPE outperforms competing methods by a large margin using both the traditional correlation measure and the qualitative measure.

II. Dimensionality Reduction

For every data point $\mathbf{x}_i \in \mathbb{R}^h$ in a high-dimensional space (e.g., pixels in a hyperspectral image with h bands), the dimensionality reduction seeks its low-dimensional representation $\mathbf{y}_i \in \mathbb{R}^l$, where $l < h$ (e.g., $l = 3$ for the color display of a hyperspectral image). Dimensionality reduction is often cast as a minimization problem with the following cost function:

$$f = \sum_i \sum_{j \neq i} (r_{ij} - d_{ij})^2 W(\cdot) \tag{1}$$

where r_{ij} is the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ in the input data space, and d_{ij} is the Euclidean distance $\|\mathbf{y}_i - \mathbf{y}_j\|$ in the projection space of reduced dimensionality. For weighting function $W(\cdot)$, various forms exist [10].

With $W(\cdot) = 1$, PCA and classical multidimensional scaling use the Euclidean distance for r_{ij} [11]. The Euclidean distance underestimates the proximity of data points on a nonlinear manifold and leads to erroneous dimensionality reduction. ISOMAP addresses this problem by using the geodesic distance for r_{ij}, which better captures the nonlinear features of the data [4].

With $W(d_{ij}) = 1$ if $d_{ij} \leq d_c$ and $W(d_{ij}) = 0$ if $d_{ij} > d_c$ for a neighbor radius d_c, curvilinear component analysis uses Euclidean distances for r_{ij} but focuses on preserving the distances between nearby low-dimensional representations $\mathbf{y}_i$ [12]. Curvilinear distance analysis (CDA) is similar, except that it uses geodesic distances for r_{ij} [13].

III. FSPE

A. SPE

ISOMAP has to estimate the geodesic distances between data points, which is then followed by the eigendecomposition of the distance matrix. The distance computation and the eigendecomposition incur a substantial computing time. Although more efficient versions of ISOMAP are available [3], [14], it is still too slow, particularly for interactive visualization [1].

As the geodesic distance is always greater than or equal to its corresponding Euclidean distance, stochastic proximity embedding (SPE) takes the Euclidean distances between remote points as the lower bounds of their true geodesic distances and uses them as a means to impose global structure [15]. Avoiding the computation of geodesic distances between remote points, SPE merely requires that the distances between their low-dimensional representations do not fall below their Euclidean distances in the high-dimensional input data space.

Starting with a random initial configuration, the SPE algorithm iteratively refines the embedding by repeatedly selecting two points i and j at random and adjusting their coordinates in a manner similar to a stochastic gradient descent as follows:

$$\begin{aligned} \mathbf{y}_i &\leftarrow \mathbf{y}_i + \lambda(t) S(r_{ij}) \frac{r_{ij} - d_{ij}}{d_{ij} + \epsilon} (\mathbf{y}_i - \mathbf{y}_j) \\ \mathbf{y}_j &\leftarrow \mathbf{y}_j + \lambda(t) S(r_{ij}) \frac{r_{ij} - d_{ij}}{d_{ij} + \epsilon} (\mathbf{y}_j - \mathbf{y}_i) \end{aligned} \tag{2}$$

$$S(r_{ij}) = \begin{cases} 1, & \text{if } r_{ij} \leq r_c \\ & \quad \text{or} \\ & \text{if } r_{ij} > r_c \text{ and } r_{ij} > d_{ij}, \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $\lambda(t)$ is the learning rate that decreases over time t, ϵ is a tiny number used to avoid division by zero, r_{ij} is Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|$, and r_c is the neighborhood radius.

Preserving the nonlinear geometry of complex high-dimensional data, SPE is a fast and scalable method for dimensionality reduction. SPE, however, suffers from distortions when reducing dimensionality below the intrinsic dimensionality. As with other methods of dimensionality reduction, remote points in the input data space can end up as nearby points in the projection space.

B. FSPE

To make SPE more faithful in visualization, we propose the FSPE algorithm as follows:

$$\begin{aligned} \mathbf{y}_i &\leftarrow \mathbf{y}_i + \lambda(t) T(d_{ij}) \frac{r_{ij} - d_{ij}}{d_{ij} + \epsilon} (\mathbf{y}_i - \mathbf{y}_j) \\ \mathbf{y}_j &\leftarrow \mathbf{y}_j + \lambda(t) T(d_{ij}) \frac{r_{ij} - d_{ij}}{d_{ij} + \epsilon} (\mathbf{y}_j - \mathbf{y}_i) \end{aligned} \tag{4}$$

$$T(d_{ij}) = \begin{cases} 1, & \text{if } d_{ij} \leq d_c(t) \\ & \quad \text{or} \\ & \text{if } d_{ij} > d_c(t) \text{ and } d_{ij} < r_{ij}, \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $d_c(t)$ is a neighbor radius that decreases over time, and r_{ij} is Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|$. Starting with a random initial configuration, the proposed FSPE algorithm iteratively refines it by repeatedly selecting two points i and j at random and adjusts their coordinates according to (4).

The key difference between SPE and FSPE is in $S(r_{ij})$ and $T(d_{ij})$. Whereas the $S(r_{ij})$ of SPE is a function of distance r_{ij} in the input data space with a fixed neighborhood radius r_c, the $T(d_{ij})$ of FSPE is a function of distance d_{ij} in the projection space with a time-varying neighborhood radius $d_c(t)$. $T(d_{ij})$

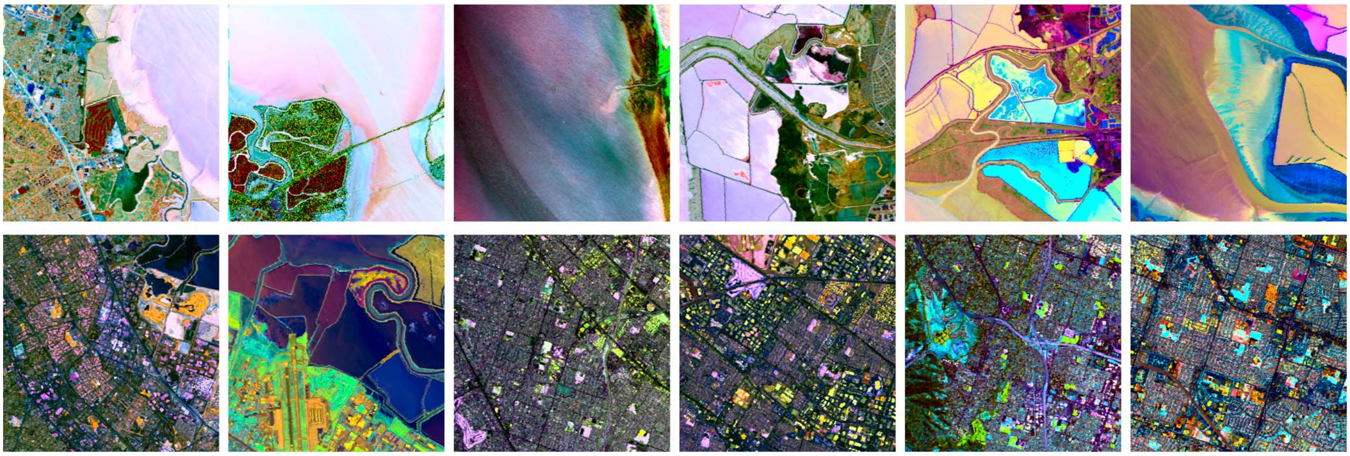


Fig. 2. We use 12 image patches for the experiments, which contain various spectral signatures, covering Moffett Field, California at the southern end of San Francisco Bay. The visualization was carried out using the proposed FSPE.

makes the projection of dimensionality reduction more faithful in the following senses.

1) It detects those "unfaithful" points that are pairwise remote in the input data space (i.e., large r_{ij}) but that are nearby in the projection space (i.e., small d_{ij}), the latter of which satisfies the condition $d_{ij} \le d_c(t)$ in (5).
2) It corrects the unfaithful points to make them more faithful, i.e., the large discrepancy $r_{ij} - d_{ij}$ adjusts them to be further apart in the projection space according to (4).

C. Local Enhancement by FSPE

Even after the completion of the projection, the FSPE algorithm is applied with a different purpose. After the projection, a user interactively selects points of interest $\{i_1, i_2, \ldots, i_p\}$ and reruns the FSPE algorithm in (4). Taking the final configuration as the new initial configuration, the update selects j at random from the set of the whole data points, but it is limited in selecting $i \in \{i_1, i_2, \ldots, i_p\}$. This local enhancement will improve the projection quality of the selected points, i.e., more reliable colors will be assigned to them.

Local enhancement by FSPE applies to any dimensionality reduction method. If one already has a configuration of points of reduced dimensionality, for instance, by PCA, then one carries out the local enhancement of selected points by FSPE, starting with the PCA-based configuration as an initial configuration.

IV. Pixelwise Quality Metric

A popular choice for estimating the quality of dimensionality reduction is a metric based on the following correlation coefficient [1], [9]:

$$\gamma = \frac{X^T Y/|X| - \bar{X}\bar{Y}}{\text{std}(X)\text{std}(Y)} \tag{6}$$

where X is the vector of all pairwise distances of the data points in the input data space, and Y is the vector of the corresponding pairwise Euclidean distances in the projection space. However, summarizing the quality of the dimensionality reduction for the entire image with a single number, this global metric cannot assess the quality of the dimensionality reduction at individual pixels. This widely varies among different pixels in a single image.

To remedy this issue, we use a pixelwise correlation metric as follows:

$$\gamma(i) = \frac{X_i^T Y_i/|X_i| - \bar{X}_i\bar{Y}_i}{\text{std}(X_i)\text{std}(Y_i)} \tag{7}$$

where X_i is the vector of all pairwise distances involving the ith pixel in the input data space, and Y_i is the vector of the corresponding pairwise Euclidean distances in the projection space.

V. Experimental Results

We implemented our method in Microsoft Visual Studio C++ 2008 with CUDA 4.2 in Windows 7. The hardware included an Intel i7-930 CPU clocked at 2.80 GHz, with 12 GB of main memory. The graphics processor was an Nvidia GTX 280 with a buffer size of 1 GB. We use 12 hyperspectral images that have various spectral signatures (see Fig. 2).

A. Correlation Metric

We tested the performance of FSPE and compared it against other projection methods using correlation metric γ. ISOMAP, LLE, and SPE involve free parameters such as the number of the neighbors k or neighbor radius r_c. For a fair comparison, we tried different values of k and r_c and only included the best result for each of these methods.

The proposed FSPE is competitive, frequently outperforming the other methods. Among the 12 test images, FSPE outperforms all the other methods with five images and comes as the second best performer with three images (see Table I). The average performance is 12% better. Note that, since FSPE outperforms PCA, we do not compare against other methods that are known to be similar to or outperformed by PCA [16], [17].

For FSPE, we use $\epsilon = 1.0 \times 10^{-8}$. The initial configuration is a random distribution of points in a 3-D cube $[0, 1] \times [0, 1] \times [0, 1]$. To avoid oscillatory behavior, learning rate $\lambda(t)$ linearly decreases from $\lambda_0 = 1.0$ to $\lambda_1 = 1.0 \times 10^{-3}$.

TABLE I
PERFORMANCE OF DIMENSIONALITY REDUCTION PROJECTIONS MEASURED ACCORDING TO CORRELATION COEFFICIENT γ. THE HIGHER, THE BETTER

	PCA	SPE	ISOMAP	CDA	LLE	*FSPE*
Mean	0.599	0.641	0.602	0.604	0.409	**0.717**
Region-0	0.691	0.696	0.525	0.550	0.325	**0.863**
Region-1	**0.901**	0.809	0.550	0.557	0.712	0.862
Region-2	0.469	**0.994**	0.817	0.900	0.136	0.904
Region-3	0.579	0.620	**0.800**	0.766	0.707	0.584
Region-4	0.673	0.641	0.646	0.648	0.368	**0.794**
Region-5	0.591	0.813	0.809	0.753	**0.885**	0.532
Region-6	0.653	0.504	0.562	**0.715**	0.516	0.620
Region-7	0.760	0.662	0.549	0.588	0.142	**0.809**
Region-8	0.442	**0.671**	0.414	0.128	0.277	0.619
Region-9	0.217	0.141	0.546	0.608	0.141	**0.867**
Region-10	0.520	**0.684**	0.560	0.487	0.324	0.360
Region-11	0.695	0.462	0.448	0.551	0.374	**0.786**

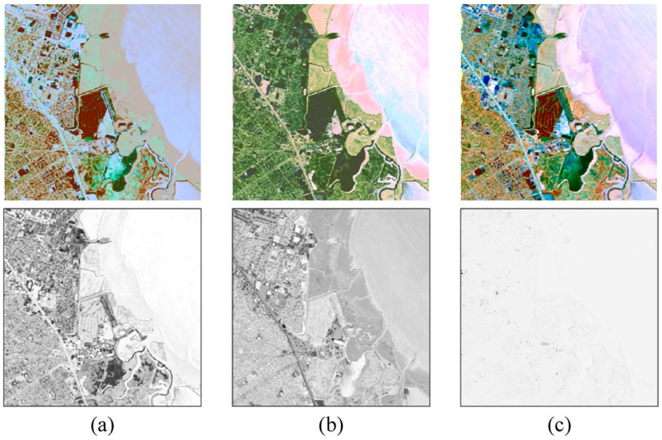
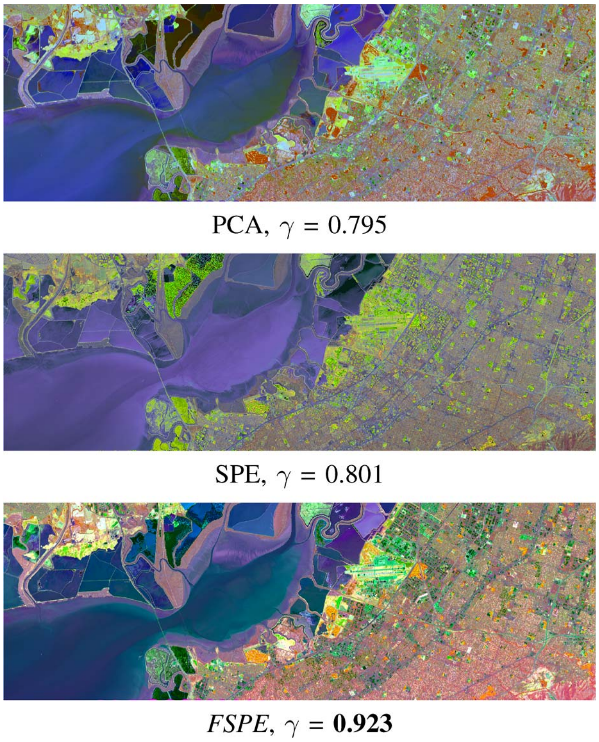

(a) (b) (c)

Fig. 3. Grayscale images of pixelwise correlation values $\gamma(i)$. The top row shows the color display of a hyperspectral image by PCA, ISOMAP, and FSPE. The bottom row presents the corresponding grayscale display of the pixelwise correlation values; the brighter, the better (i.e., higher). (a) PCA, $\gamma = 0.691$. (b) ISOMAP, $\gamma = 0.525$. (c) FSPE, $\gamma = \mathbf{0.863}$.

B. Faithful Visualization

For visualization of hyperspectral images, we use the CIE $L^*a^*b^*$ color space that is perceptually uniform. Among the three coordinates in the projection space, we choose the most varying coordinate for the L^* component and the remaining two coordinates for the a^* and b^* components, followed by a simple linear stretching such that $L^* \in [0,\ 100]$ and that both a^* and b^* have a zero mean.

Not only providing a quantitative measure of the projection quality at each pixel, pixelwise correlation metric $\gamma(i)$ is also useful during the color display and visual examination of hyperspectral images. In addition to a hyperspectral image in color, one may display a supplementary grayscale image. This image indicates the reliability of the color at each pixel i by mapping $\gamma(i)$ to a grayscale intensity, i.e., the brighter, the higher (i.e., more reliable) (see Fig. 3). This provides extra information that the overall correlation metric γ is unable to provide. Although the overall correlation value γ by PCA is higher than that by ISOMAP for example, the grayscale display shows that PCA yields pixels with $\gamma(i)$ values that are significantly lower (i.e., darker) than those of the corresponding pixels by ISOMAP (see Fig. 3). At these pixels, the quality of dimensionality reduction or the reliability of color by PCA is lower than that by ISOMAP.


PCA, γ = 0.795

SPE, γ = 0.801

FSPE, γ = **0.923**

Fig. 4. Visualization results of a large hyperspectral image of 1800 × 600 pixels by PCA, SPE, and FSPE.

Fig. 4 shows the visualization results of a large hyperspectral image (1800 × 600 pixels) by PCA, SPE, and FSPE; the proposed FSPE outperforms the others.

C. Local Enhancement

To demonstrate the local enhancement by FSPE, we choose the upper-right part of a hyperspectral image as a region of interest; the dimensionality reduction of the image has been initially carried out by CDA, which yields a low correlation value for the region. Thus, the pixel colors of the region are unreliable and even misleading. After the local enhancement by FSPE, the correlation value significantly increases, and the new colors of the corresponding pixels are more reliable and faithful (see Fig. 5).

D. GPU Implementation

Nonlinear dimensionality reduction is often too slow to compute for interactive visualization. Linear methods, such as PCA, are used due to their fast computation [1]. FSPE is competitive in computational time. For a hyperspectral image of 300 × 300 pixels with 224 spectral bands, the computation of FSPE only takes 12.416 s on a CPU; the total number of pairwise refinement steps is 9×10^7. Owing to its inherently parallel nature, the computation time of FSPE is further reduced when benefiting from the parallel computing power of low-cost GPUs. The computation of FSPE takes 0.011 s on a GPU, yielding a speed up of two orders of magnitude and making it suitable for interactive visualization.

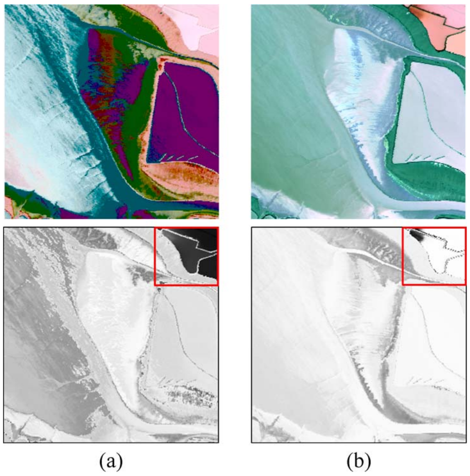


Fig. 5. Local enhancement by FSPE. (a) Color display of a hyperspectral image by CDA and its corresponding grayscale image of pixelwise correlation $\gamma(i)$. The upper-right part of it is chosen as a region of interest. Since the correlation value of the region is low ($\gamma = 0.036$), its colors appearing (relatively) homogeneous may not be reliable and misleadingly implies the homogeneity of spectral signatures. (b) After the local enhancement by FSPE, the correlation value of the region significantly increases to $\gamma = 0.890$; thus, its colors are now more reliable, showing more heterogeneity of spectral signatures.

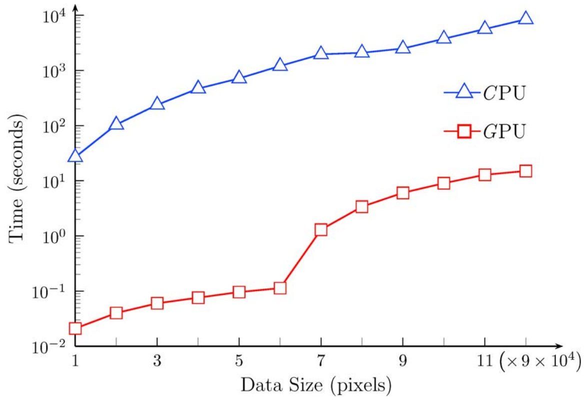


Fig. 6. Computation time of the FSPE algorithm for hyperspectral images with different sizes. The GPU implementation maintains a speed up of two orders of magnitude compared with the CPU implementation.

As other nonlinear projection methods often involve the eigendecomposition or computation of geodesic distances, they are more challenging to speed up their computation by GPUs. Fig. 6 shows the running time of the GPU implementation of FSPE compared with the CPU implementation for different data sizes, i.e., $1 \times 300 \times 300, 2 \times 300 \times 300, \ldots, 12 \times 300 \times 300$ pixels; for a hyperspectral image of a million pixels, the implementation of FSPE on a GPU takes less than 16 s.

VI. Conclusion

FSPE is a novel nonlinear dimensionality reduction method for visualizing hyperspectral imagery. While considering the nonlinear characteristics of data points in the high-dimensional spectral space, FSPE avoids the costly computation of explicitly estimating geodesic distances that are often required by other nonlinear methods. The experimental results prove that FSPE is competitive with or outperforms state-of-art methods in hyperspectral image visualization. We also demonstrate that, due to its parallel nature, FSPE takes advantage of parallel computing devices such as GPUs, leading to substantial speed ups and interactive applications. We also utilize a pixelwise metric that measures the quality of hyperspectral image visualization at individual pixels. The future work includes an optimized GPU implementation of FSPE aiming at processing a hyperspectral image of a million pixels within a second or so and improving the pixelwise metric with a weighted version. The source code of FSPE is available [18].

References

[1] M. Cui, A. Razdan, J. Hu, and P. Wonka, “Interactive hyperspectral image visualization using convex optimization,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 6, pp. 1673–1684, Jun. 2009.

[2] J. S. Tyo, A. Konsolakis, D. I. Diersen, and R. C. Olsen, “Principal-components-based display strategy for spectral imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 3, pp. 708–718, Mar. 2003.

[3] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, “Improved manifold coordinate representations of large-scale hyperspectral scenes,” *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2786–2803, Oct. 2006.

[4] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[5] A. Mohan, G. Sapiro, and E. Bosch, “Spatially coherent nonlinear dimensionality reduction and segmentation of hyperspectral images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 206–210, Apr. 2007.

[6] S. Roweis and L. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[7] Q. Du, N. Raksuntorn, S. Cai, and R. Moorhead, “Color display for hyperspectral imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1858–1866, Jun. 2008.

[8] M. Radovanović, A. Nanopoulos, and M. Ivanović, “Hubs in space: Popular nearest neighbors in high-dimensional data,” *J. Mach. Learning Res.*, vol. 11, pp. 2487–2531, Sep. 2010.

[9] N. Jacobson and M. Gupta, “Design goals and solutions for display of hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 11, pp. 2684–2692, Nov. 2005.

[10] S. France and J. Carroll, “Two-way multidimensional scaling: A review,” *IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev.*, vol. 41, no. 5, pp. 644–661, Sep. 2011.

[11] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2000.

[12] P. Demartines and J. Herault, “Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets,” *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 148–154, Jan. 1997.

[13] J. A. Lee, A. Lendasse, and M. Verleysen, “Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis,” *Neurocomputing*, vol. 57, pp. 49–76, Mar. 2004.

[14] V. de Silva and J. B. Tenenbaum, “Local versus global methods for nonlinear dimensionality reduction,” in *Advances in Neural Information Processing Systems*, vol. 15. Cambridge, MA, USA: MIT Press, 2003, pp. 705–712.

[15] D. K. Agrafiotis and H. Xu, “A self-organizing principle for learning nonlinear manifolds,” *Proc. Nat. Acad. Sci. U.S.A.*, vol. 99, no. 25, pp. 15869–15872, Dec. 2002.

[16] M. Mignotte, “A bicriteria-optimization-approach-based dimensionality-reduction model for the color display of hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 2, pp. 501–513, Feb. 2012.

[17] Z. Mahmood and P. Scheunders, “Enhanced visualization of hyperspectral images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 5, pp. 869–873, Sep. 2011.

[18] [Online]. Available: https://sites.google.com/site/fspedr/