CLASSIFICATION OF HYPERSPECTRAL IMAGES USING SVM WITH SHAPE-ADAPTIVE RECONSTRUCTION AND SMOOTHED TOTAL VARIATION

Ruoning Li [1], *Kangning Cui* [1,2], *Raymond H. Chan* [1,2,*], *and Robert J. Plemmons* [3]

[1] Department of Mathematics, City University of Hong Kong
[2] Hong Kong Centre for Cerebro-Cardiovascular Health Engineering
[3] Departments of Mathematics and Computer Science, Wake Forest University

ABSTRACT

In this work, a novel algorithm called SVM with Shape-adaptive Reconstruction and Smoothed Total Variation (SaR-SVM-STV) is introduced to classify hyperspectral images, which makes full use of spatial and spectral information. The Shape-adaptive Reconstruction (SaR) is introduced to preprocess each pixel based on the Pearson Correlation between pixels in its shape-adaptive (SA) region. Support Vector Machines (SVMs) are trained to estimate the pixel-wise probability maps of each class. Then the Smoothed Total Variation (STV) model is applied to denoise and generate the final classification map. Experiments show that SaR-SVM-STV outperforms the SVM-STV method with a few training labels, demonstrating the significance of reconstructing hyperspectral images before classification.

Index Terms— Hyperspectral Image Classification, Image Reconstruction, Support Vector Machines, Smoothed Total Variation, Semi-supervised Learning

1. INTRODUCTION

Hyperspectral Images (HSIs) often have hundreds of electromagnetic reflectance bands collected by aircraft or satellites that contain discriminative information. HSIs can be represented by $X \in \mathbb{R}^{M \times N \times B}$, where M and N represent the spatial size and B represents the number of spectral bands [1]. With the enhancement of the spectral resolution of HSIs, machine learning approaches can be applied to investigate HSIs in land cover identification, change recognition, and mineral mapping, etc. [2, 3, 4, 5]. The ground truth annotations of HSIs are expensive, which motivates us to take less randomly picked labels for training [2].

In this work, SVM with Shape-adaptive Reconstruction and Smoothed Total Variation (SaR-SVM-STV) is introduced to classify highly-mixed HSIs in a semi-supervised way. In our previous work, SVM with Smoothed Total Variation (SVM-STV) incorporated a convex variant of the Mumford-Shah denoising algorithm with ν-SVM Classifier to utilize spatial and spectral information [2]. It has been shown that SVM-STV was highly competitive compared with state-of-the-art algorithms, illustrating the effectiveness of incorporating spatial information into classical pixel-wise algorithms. In this work, spatial information is further utilized by a preprocessing step named Shape-adaptive Reconstruction (SaR). The number of labeled pixels of each class needed to achieve satisfying performance is less than 30.

This paper is organized as follows. Section 2 introduces the related works and the motivation of this work. In Section 3, the workflow of the SaR-SVM-STV Algorithm is introduced. Section 4 shows the numerical results of SaR-SVM-STV compared with ν-SVM and SVM-STV on two benchmark HSI datasets. Section 5 concludes the experiments and discusses the planned future work.

2. RELATED WORKS

Semi-supervised methods are desirable and computational feasible since they only require a few labeled data for training. The classical semi-supervised pixel-wise algorithms, such as SVMs, performed well for HSIs with small computational cost [6]. A publicly available toolbox called LIBSVM collects the most common SVM models that can be instantly utilized, promoting the application of SVMs [7].

In general, pixel-wise classification methods can be enhanced by analyzing the spatial dependency of nearby pixels, due to the common theoretical-based assumption that the pixels are locally homogeneous in HSIs [8]. In [9], spatial-spectral features are merged to train the SVM Classifier with Mercer's kernels. K-means and Principal Component Analysis (PCA) are used in [10] to extract features, which are then fed into SVM with RBF kernels. The performance of the above two methods grows significantly compared to pixel-wise classifiers. The Smoothed Total Variation (STV) model has been used for denoising and segmentation with superb performance [2, 11, 12]. The SVM-STV in [2] firstly trains a ν-SVM Classifier to estimate the probability maps of each

[*] Corresponding Author; email: raymond.chan@cityu.edu.hk

This work was supported in part by HKRGC Grants No. CUHK14301718, CityU11301120, C1013-21GF, CityU Grant 9380101.

class, and then applies the STV model with L_1 and L_2 regularizers on the generated 3D probability tensor.

It is customary to employ spatial windows of pixels to extract features in HSIs. Adopting square-shaped fixed-size windows is a straightforward solution. A joint sparsity representation (JSR) proposed in [13] takes full advantage of the spatial information by constructing JSR on local square patches. However, pixels in a square window, particularly those on the margins or corners, may not have similar spectra as the center pixel. Nevertheless, fixed-size windows will fail when both spatially homogeneous and heterogeneous regions exist. Nested sliding windows are designed in [14] to find the best nested square window that maximizes the mean of the Pearson Correlation between window center and other pixels, and then replace the spectra of center pixels with the product of weighted Pearson Correlation vectors and spectra of pixels in the best nested windows. In [15], the shape-adaptive (SA) regions of pixels are learned by constructing local smooth irregular polygons in the first principal component. The local polynomial approximation filtering (LPA) and the intersection of confidence intervals rule (ICI) are applied to determine the size and shape of the polygons [16]. Then a shape-adaptive JSR is applied to explore the spatial and spectral information properly within SA regions [15].

In this work, instead of using JSR, we reconstruct the spectrum of each pixel using its SA region. The main difference between the SaR-SVM-STV proposed in this work and SVM-STV is that the shape-adaptive reconstruction (SaR) is applied to further utilize the spatial features of HSIs before training the ν-SVM. That is, we reconstruct the HSIs by SaR and reduce the dimension of the reconstructed data by PCA, and then perform SVM-STV on the PCA reduced data. The workflow of the SaR-SVM-STV model is summarized in Algorithm 1 and the following section.

3. METHODOLOGY

3.1. Shape-adaptive Reconstruction

The Shape-adaptive Reconstruction (SaR) is introduced as an efficient tool to further utilize spatial features as a preprocessing step. The shape-adaptive (SA) region is generated to pick surrounding pixels that are the closest to the target pixel by the LPA denoising and ICI rule [15, 16]. The Pearson Correlation coefficients $\mathrm{Corr}(x_c, x_i) = \frac{\mathrm{Cov}(x_c,x_i)}{\sqrt{\mathrm{Var}(x_c)\cdot\mathrm{Var}(x_i)}}$ are computed to estimate the correlation between the center pixel x_c and the other $n-1$ pixels x_i, where n represents the total number of pixels in x_c's SA region. Then a vector $p(x_c) \in \mathbb{R}^{n\times 1}$ is formed with the i-th entry $\mathrm{Corr}(x_c, x_i)$ when $i = 1, 2, \cdots, n-1$, and the n-th entry 1. The correlation coefficients are then rescaled as a weight vector $\hat{p}(x_c) = \frac{p(x_c)}{\mathrm{Sum}(p(x_c))}$, and the spectrum of the center pixel is reconstructed by $S_c = S_{nbs}\hat{p}(x_c)$, where $S_{nbs} \in \mathbb{R}^{B\times n}$ represents the spectra collection of the n pixels in the SA region of pixel x_c and the n-th column represents the spectrum of x_c. Thus, pixels inside SA regions will not contribute to the spectrum of center pixel uniformly, with highly-correlated pixels dominating the spectrum of center pixel as they are associated with large weights in $\hat{p}$. PCA dimension reduction is performed to further reduce the noise in HSIs and speed up the training of our proposed model. Experiments show that using the data before or after PCA produces similar classification results.

Algorithm 1: SVM with Shape-adaptive Reconstruction and Smoothed Total Variation (SaR-SVM-STV)

```
Input: HSI, $\Upsilon$ (training set),
    $\nu$ (tolerance of training error),
    $\gamma$ (scaling factor of the RBF kernel),
    $\beta_1, \beta_2$ (regularization parameters)
Output: $\hat{Y}$ (classification map)
1 Reconstructing the spectrum of each pixel with its
    Shape-adaptive Regions: $S_c = S_{nbs}\hat{p}(x_c)$, and get
    the reconstructed data $\tilde{X}$;
2 Reducing the dimension of data by PCA:
    $X_r =$PCA$(\tilde{X})$;
3 Estimating the probability tensor $V$ by using
    $\nu$-SVMs on the HSI data with parameters $\nu$ and $\gamma$:
    $V = \nu$-SVMs$(X_r, \nu, \gamma)$;
4 Applying STV model on the probability tensor $V$ to
    get the denoised tensor $U$ for each class:
    $\min_{\mathbf{u}_k} \frac{1}{2}\|\mathbf{u}_k - \mathbf{v}_k\|_2^2 + \beta_1\|\nabla\mathbf{u}_k\|_1 +$
    $\frac{\beta_2}{2}\|\nabla\mathbf{u}_k\|_2^2, \text{s.t.}, \mathbf{u}_k|_\Upsilon = \mathbf{v}_k|_\Upsilon$;
5 The classification map $\hat{Y}$ is obtained by
    $\hat{Y}_{i,j} =$MaxIndex$(U_{i,j,:})$.
```

3.2. SVM with Smoothed Total Variation

The ν-SVMs are trained by the randomly chosen labels Υ with one-against-one strategy and five-fold cross-validation, and predict the 3D probability map $V \in \mathbb{R}^{M\times N\times K}$. The k-th channel $\mathbf{v}_k = V_{:,:,k}, k = 1, ..., K$ represents the probability of the pixels belonging to the class k [2]. The toolbox of LIBSVM library is used to implement ν-SVM [7]. Then the STV model is adopted to reduce the noise in the probability tensor V by applying the alternating direction method of multipliers (ADMM) with the convergence guaranteed [11, 12, 17]. The STV model is shown in Step 4 of Algorithm 1. The $\mathbf{u}_k$ represents the smoothed probability map of class k. Meanwhile, the labels of pixels in the training set Υ should remain unchanged. The smoothed 3D tensor U is obtained after the STV noise reduction, and the final classification map $\hat{Y}$ is then produced by taking the index of the class that has the maximum probability of each pixel [2].

4. NUMERICAL RESULTS

In this section, we show the results of SaR-SVM-STV algorithm compared with ν-SVM and SVM-STV on two publicly available datasets: Indian Pines and University of Pavia (Pavia U). Details of the datasets are summarized in Table 1.

Table 1. Summary of the two HSI datasets

	Type	Size	No. Classes	Sensor	Location
Indian Pines	Forest	145×145×200	16	AVIRIS	Indiana, US
Pavia U	Urban	610×340×103	9	ROSIS	Pavia, Italy

The PCA dimension reduction is used to keep 99.9% of the variance of the reconstructed data. The best ν and γ are selected by the cross-validation of ν-SVM. The best smoothing parameters β_1 and β_2 in the STV model are tuned on the probability tensor V that maximize the sum of the overall accuracy (OA), the average accuracy (AA), and the Kappa coefficient in 10 trials. For Pavia U, $\beta_1 = 0.2$, and $\beta_2 = 1$. For Indian Pines, $\beta_1 = 0.2$, and $\beta_2 = 4$. The source code used in this work can be found at: github.com/ckn3/SA-Recon.

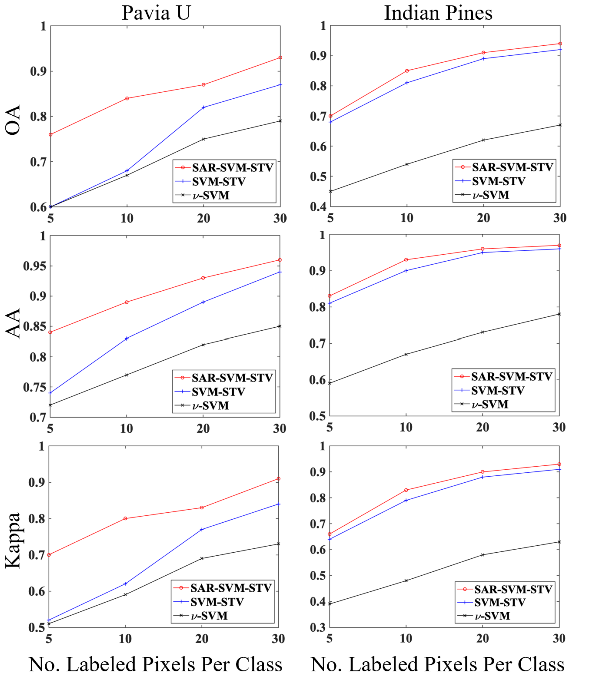
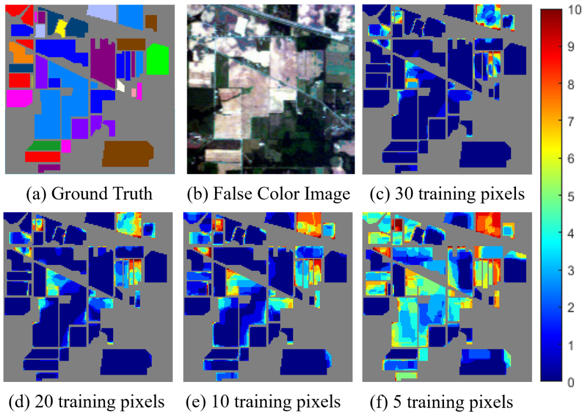


Fig. 1. The performance of ν-SVM, SVM-STV, and SaR-SVM-STV on Pavia U and Indian Pines. The two columns present the performance of Pavia U and Indian Pines, and the three rows show the overall accuracy, the average accuracy, and the kappa coefficient of the three algorithms we compare against, respectively.

Fig. 1 shows the tendency of performance while increasing the labeled pixels randomly picked from each class. The three metrics improve substantially by incorporating more training labels, and the SaR-SVM-STV model outperforms the comparison methods in all cases. Fig. 2 shows relationship between the number of labeled pixels involved for training and the count of misclassifications on Indian Pines using SaR-SVM-STV. As the training size increases, the portion of misclassifications reduces monotonically.

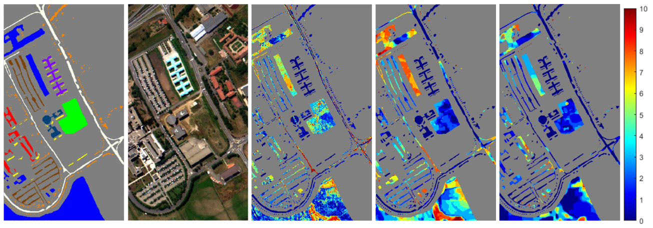


(a) Ground Truth (b) False Color Image (c) 30 training pixels

(d) 20 training pixels (e) 10 training pixels (f) 5 training pixels

Fig. 2. Misclassification heatmaps of Indian Pines using SaR-SVM-STV with different numbers of labeled pixels. Figures (a-b) show the ground truth (each color represents a class) and false-color visualization of Indian Pines. Figures (c-f) show the heatmaps when 30, 20, 10, and 5 labeled pixels per class are used. The colorbar represents the count of misclassifications in 10 trials.

Fig. 3 indicates the efficacy of incorporating spatial information. By comparing Figs. 3 (c) and (d), it can be found that although the spatial noise is reduced, the SVM-STV model is misled by misclassified pixels in the predicted map of ν-SVM when only a few labels are used. Applying the SaR and PCA algorithms smooth the HSIs before classification, and thus reduce the number of misclassified pixels in the ν-SVM step. The STV model can further tick out the misclassified pixels by enforcing spatial connectivity in probability maps as a postprocessing step, see Fig. 3 (e).



(a) Ground Truth (b) False Color Image (c) v-SVM (d) SVM-STV (e) SAR-SVM-STV

Fig. 3. Misclassification heatmaps of Pavia U by comparing the three methods when 10 labeled pixels per class are used for training. Figures (a-b) show the ground truth (each color represents a class) and false-color visualization of Pavia U. Figures (c-e) show the heatmaps of ν-SVM, SVM-STV, and SaR-SVM-STV. The colorbar represents the count of misclassifications in 10 trials.

5. CONCLUSIONS

In [2], SVM-STV has been shown to outperform many other methods. In this work, we show that adding a preprocessing step, i.e., the Shape-adaptive Reconstruction that reconstruct HSIs using spatial information, can further enhance the performance. The SaR preprocessing and STV postprocessing steps can be easily merged with other semi-supervised or unsupervised algorithms that do not utilize the spatial information of HSIs. The work planned in the near future would be combining the image processing techniques with unsupervised clustering methods [18, 19], and developing the active learning extension where a few carefully-selected annotations are used to train the model [20, 21].

6. REFERENCES

[1] M. T. Eismann, *Hyperspectral remote sensing*, SPIE, 2012.

[2] R. H. Chan, K. K. Kan, M. Nikolova, and R. J. Plemmons, "A two-stage method for spectral–spatial classification of hyperspectral images," *J. Math. Imaging Vis.*, vol. 62, no. 6, pp. 790–807, 2020.

[3] K. Cui and R. J. Plemmons, "Unsupervised classification of AVIRIS-NG hyperspectral images," in *Proc IEEE Workshop Hyperspectral Image Signal Process Evol Remote Sens*. IEEE, 2021, pp. 1–5.

[4] S. Camalan et al., "Change detection of amazonian alluvial gold mining using deep learning and sentinel-2 imagery," *Remote Sens.*, vol. 14, no. 7, 2022.

[5] M. J. Khan, H. S. Khan, A. Yousaf, K. Khurshid, and A. Abbas, "Modern trends in hyperspectral image analysis: A review," *IEEE Access*, vol. 6, pp. 14118–14129, 2018.

[6] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, 2004.

[7] C. Chang and C. Lin, "Libsvm: a library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[8] J. Liu, Z. Wu, Z. Wei, L. Xiao, and L. Sun, "Spatial-spectral kernel sparse representation for hyperspectral image classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 6, no. 6, pp. 2462–2471, 2013.

[9] G. Camps-Valls, L. Gomez-Chova, J. Muñoz-Marí, J. Vila-Francés, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote. Sens. Lett.*, vol. 3, no. 1, pp. 93–97, 2006.

[10] L. Shu, K. McIsaac, and G. R. Osinski, "Learning spatial–spectral features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5138–5147, 2018.

[11] X. Cai, R. Chan, and T. Zeng, "A two-stage image segmentation method using a convex variant of the mumford–shah model and thresholding," *SIAM J. Imaging Sci.*, vol. 6, no. 1, pp. 368–390, 2013.

[12] X. Cai, R. Chan, M. Nikolova, and T. Zeng, "A three-stage approach for segmenting degraded color images: Smoothing, lifting and thresholding (slat)," *J. Sci. Comput.*, vol. 72, no. 3, pp. 1313–1332, 2017.

[13] H. Zhang, J. Li, Y. Huang, and L. Zhang, "A nonlocal weighted joint sparse representation classification method for hyperspectral imagery," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 7, no. 6, pp. 2056–2065, 2014.

[14] J. Ren, R. Wang, G. Liu, Y. Wang, and W. Wu, "An svm-based nested sliding window approach for spectral–spatial classification of hyperspectral images," *Remote Sens.*, vol. 13, no. 1, pp. 114, 2021.

[15] W. Fu, S. Li, L. Fang, X. Kang, and J. A. Benediktsson, "Hyperspectral image classification via shape-adaptive joint sparse representation," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 9, no. 2, pp. 556–567, 2016.

[16] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, 2007.

[17] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Now Publishers Inc, 2011.

[18] S. L. Polk, K. Cui, R. J. Plemmons, and J. M. Murphy, "Diffusion and volume maximization-based clustering of highly mixed hyperspectral images," *arXiv preprint arXiv:2203.09992*, 2022.

[19] J. M. Murphy and S. L. Polk, "A multiscale environment for learning by diffusion," *Appl. Comput. Harmon. Anal.*, vol. 57, pp. 58–100, 2022.

[20] 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, 2018.

[21] S. L. Polk, K. Cui, R. J. Plemmons, and Murphy J. M., "Active diffusion and VCA-assisted image segmentation of hyperspectral images," *arXiv preprint arXiv:2204.06298*, 2022.