A Margin Technique for Dimension Reduction with Applications to Hyperspectral Imagery*

Jing Peng[1] and Kun Zhang[2]

[1]Department of Computer Science, Montclair State University, Montclair, NJ, 07043, USA
[2]Department of Computer Science, Xavier University of Louisiana, New Orleans, LA 70125, USA
pengj@mail.montclair.edu, kzhang@xula.edu

Abstract - Target classification in hyperspectral imagery has been demonstrated to be very useful in remote-sensing applications. While spectral bands provide information for classification, they give rise to a large number of features. However, a large number of features often degrade performance. In such situations, dimensionality reduction can be very helpful. There are many such techniques in the literature, and the most popular one is Fisher's linear discriminant analysis (LDA). For two class problems, LDA can be shown to be optimal. For the multi-class case, LDA is not. As such, a multi-class problem is cast into a binary one. This formulation not only simplifies the problem but also works well in practice. However, it lacks theoretical justification. We show in this paper the connection between the above formulation and Relief feature selection, thereby providing a sound basis for observed benefits associated with this formulation. Furthermore, we propose a margin based algorithm for dimensionality reduction that addresses some of the problems facing the two class formulation. We provide experimental results that corroborate well with our analysis.

Index Terms - Classification, dimensionality reduction, Relief

I . Introduction

Target classification in hyperspectral imagery has been demonstrated to be very challenging, and at the same time to be extremely useful in many remote-sensing applications [1], [2], [3]. While spectral-reflectance measurements provide information for target detection and classification, they generate a large number of features, resulting in a high dimensional measurement space [4]. However, a large number of features often degrade classification performance. This fact is due to the curse of dimensionality. In such situations, feature extraction or selection methods play an important role by significantly reducing the number of features for building classifiers.

There are many dimensionality reduction techniques for classification in the literature. The most popular one is Fisher's linear discriminant analysis (LDA) [5]. In LDA, we are given a set of N examples $z = \{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in \Re^q$ are the q-dimensional inputs, and y_i are scalar labels.

Consider a C class problem, where m is the mean vector of all data, and m_i is the mean vector of ith class data. A within-class matrix characterizes the scatter of samples around their respective class mean vectors, and it is expressed by $S_w = \sum_{i-1}^{C} p_i \sum_{j=1}^{n_i} (x_j^i - m_i)(x_j^i - m_i)^t$, where n_i is the number of examples in the ith class, p_i ($\sum_i p_i = 1$) represents the pro-portion of class i, and t denotes matrix transpose. A between-class scatter matrix characterizes the scatter of the class means around the overall mean m: $S_b = \sum_{i=1}^{C} p_i(m_i - m)(m_i - m)^t$. Thus, LDA finds the projection matrix that maximizes the objective

$$J_F(W) = tr((W^t S_w W)^{-1} W^t S_b W) \quad (1)$$

We can obtain W that maximizes $J(W)$ by solving the generalized eigenvalue problem: $S_b w_i = \lambda_i S_w w_i$.

From the Bayes perspective, LDA is optimal for two Gaussians with equal covariances [6], [7]. However, LDA is not optimal for multiple Gaussian distributions or classes with unequal covariance matrices. To address this problem, the multiclass problem can be formulated as a binary one. This formulation not only simplifies the problem but also works well in practice. However, it lacks theoretical justification.

We show in this paper the connection between the above formulation and Relief feature selection, thereby providing a sound basis for observed benefits associated with this formulation. Furthermore, we propose a margin based algorithm for dimensionality reduction that addresses some of the problems facing the two class formulation. We show how to optimize the proposed objective with semi-definite programming and corresponding complexity. We provide experimental results demonstrating that the proposed technique is competitive against competing methods in a number of hyperspectral image data. These results corroborate well with our analysis.

II. Related Work

A large number of subspace methods have been proposed to address the computational difficulty associated with LDA when the small sample size problem occurs (S_w becomes singular) [1], [2], [8], [3], [9]. A good summary is presented in [8].

A dimension reduction technique, called linear feature extraction (LFE), based on Relief [10] is introduced [11]. In [12], a metric space dimension reduction technique is proposed. The idea is to find a linear transform such that in the transformed space total within class distance is minimized, while total between class distance is maximized. Discriminant analysis based on the average margin is proposed in [13]. The technique does not involve inverting matrices, thereby avoiding the small sample size problem. This technique is

* This research was partially supported by an US Department of Army grant W911NF-12-1-0066

closely related to our proposal, as we shall see later.

III. Two Class Formulation And Relief

As stated earlier, from Bayes perspective, LDA is not optimal for multiple Gaussian distributions or classes with unequal covariance matrices. To address this problem, the multiclass problem can be formulated as a binary problem by introducing two spaces: intraclass space and extraclass space [8]. Here, the intraclass and extraclass spaces are defined respectively as

$$\Omega_I = \{x_i - x_j | L(x_i) = L(x_j)\} \text{ and } \Omega_E = \{x_i - x_j | L(x_i) \neq L(x_j)\} \quad (2)$$

where $L(x)$ denotes the label of *x*.

The means and covariances associated with these spaces can be computed as follows: $m_I = m_E = 0$, and $\sum_I = \frac{1}{N_I} \sum_{L(x_i)=L(x_j)} (x_i - x_j)(x_i - x_j)^t$ and $\sum_E = \frac{1}{N_E} \sum_{L(x_i)\neq L(x_j)} (x_i - x_j)(x_i - x_j)^t$, where $N_I = {}^1/_2 \sum_i n_i(n_i - 1)$represents the number of samples in Ω_I, and $n_E = \sum_{L_i \neq L_j} n_i n_j$ represents the number of samples in Ω_E. Notice that the rank of either $\sum_I$ or $\sum_E$can be greater than $C - 1$.

The above formulation not only simplifies the problem but also works well in practice [8]. However, it lacks theoretical justification. Here we show its connection to RELLIEF [10], thereby providing a sound basis for observed benefits associated with this formulation.

Let *x* be an instance. We define the *near hit* or *nh* of *x* as its nearest neighbor that comes from the same class as *x*. Similarly, we define the *near miss* or *nm* as the nearest neighbor of *x* that comes from the opposite class. In [14], the hypothesis margin of *x* with respect to labeled data is defined as: $\sigma(x) = \|x - nm(x)\| = \|x - nh(x)\|$.The hypothesis margin is easy to compute and lower bounds the sample margin [14]. To be less sensitive to noise, *k* near misses/hits are often used in practice to optimize the margin for some integer *k* [11].

Let *h(x) = x – nh(x)* and *m(x) = x – nm(x).* We define two matrices, near hit S_h and S_m, as follows:

$$S_h = \sum_{i=1}^{n} h(x_i)h(x_i)^t \text{ and } S_m = \sum_{i=1}^{n} m(x_i)m(x_i)^t \quad (3)$$

Often *k* nearest neighbors are used to estimate near hit *nh(x)* and near miss *nm(x)*, respectively. Assume that *nh(x)*= m_i. That is, the near hit spans the entire class. We first write $S_h = \sum_{i=1}^{c} \sum_{L(x_j)=L_i} h(x_j)h(x_j)^t$. It follows that for class *i*, $\sum_{j=1}^{n_i} h(x_j)h(x_j)^t = \sum_{j=1}^{n_i} x_i x_i^t - n_i m_i m_i^t$. On the other hand, we can write the covariance of the intra class space as $\sum_I = \sum_{i=1}^{c} \sum_{L(x_i)=L(x_j), j<i} (x_i - x_j)(x_i - x_j)^t$ Let $\sum_i = \sum_{L(x_i)=L(x_j), j<i} (x_i - x_j)(x_i - x_j)^t$ .We have $\sum_i = n_i\left(\sum_{i=1}^{n_i} x_i x_i^t - n_j m_i m_i^t\right)$ .Thus, we can rewrite intra class covariance $\sum_I = \sum_{i=1}^{c} \frac{n_i}{n_I} \left(\sum_{i=1}^{n_i} x_i x_i^t - n_j m_i m_i^t\right)$. This shows that each class contributes proportionally to overall intra class covariance $\sum_I$. In RELIEF, on the other hand, every class contributes equally to S_h, when near hits are extended to the entire class. If every class has the same number of examples, $\sum_I$ and S_h are the same.

The relationship between extra class covariance $\sum_E$ and S_m is not as straightforward. However, let us define a space, similar to the one defined in Eq. (2)

$$\Omega_E = \{x_i - m_{-i} | L(x_i) \neq L(m_{-i})\}, \quad (4)$$

where $m_{-i} = \frac{1}{n-n_i} \sum_{L(x_j)\neq L_i} x_j$ denotes the mean of examples whose label differs from that of x_i. It is straightforward to show that $\sum_E = S_m$. Here $\sum_E$ denotes the covariance matrix of the extraclass space defined in Eq. (4) and S_m is the near miss matrix (Eq.3), where *nm*(*x*) spans the entire neighborhood, respectively.

Notice that Ω_E (Eq. 4) is not exactly the same as Ω_E (Eq. 2). However, the connections between Ω_E and S_m and between Ω_I and S_h demonstrate a close relationship between the two class formulation and RELIEF, thereby providing justification to the two class formulation.

IV. A Margin Based Solution

For two classes whose distributions are Gaussian, an objective based on the Bhattacharyya distance is proposed [7],[8]. The Bhattacharyya distance is a measure of overlap between two distributions. The Bhattacharyya distance has shown to work well even if the underlying distributions are not Gaussian [6].

For the two class formulation discussed above, the Bhattacharyya distance becomes $J_B(w) = \ln |(w^t \sum_E w)^{-1}(w^t \sum_I w) + (w^t \sum_I w)^{-1}(w^t \sum_E w) + 2I_d|$, where $I_d$is the identity matrix in the reduced space. To obtain the optimal solution, we choose generalized eigenvectors of $(\sum_E, \sum_I)$, corresponding to the largest $\lambda + \frac{1}{\lambda}$, where λ are generalized eigenvalues [7].

A closer look at the Bhattacharyya distance shows that it is an effective algorithm for selecting features along which the difference in variances between the two classes, defined by the intra-class and extra-class spaces (Eq. 2), is the largest. In many applications, this type of features can be useful. However, in the case of the intra-class and extra-class formulation, such features may not be desirable.

To address the problems implied by the above discussions, we introduce a margin based criterion and related optimization techniques. We focus on two class problems first. The multi-class case will be discussed later. The goal of LDA is to find a direction *w* that simultaneously places two classes afar and minimizes within class variations. Fisher's criterion (Eq. 1) achieves this goal. Alternatively, we can achieve this goal by maximizing

$$J(w) = tr(w^t(\lambda S_B - S_W)w), \quad (5)$$

where $\lambda > 0$ is a constant that weighs relative importance of the two terms S_B and S_W in determining the outcome of linear discriminants.

To see the proposed objective (Eq. 5) is margin based, notice that maximizing $tr(\lambda S_B - S_W)$ is equivalent to $J = \frac{1}{2} \sum_i^2 \sum_j^2 p_i p_j d(C_i, C_j)$, where p_i denotes the probability of class C_i. The intraclass distance *d* is defined as $d(C_i, C_j) = \lambda d(m_i, m_j) - tr(S_i) - tr(S_j)$, where S_i represents the scatter

matrix of class C_i, and $d(m_i, m_j)$ denotes the Euclidean distance between m_i and m_j. Thus, $= \frac{1}{2}\sum_i^2 \sum_j^2 p_i p_j d(C_i, C_j) = \frac{1}{2}\sum_i^2 \sum_j^2 p_i p_j d(m_i, m_j) - \frac{1}{2}\sum_i^2 \sum_j^2 p_i p_j \left(tr(S_i) + tr(S_j)\right)$. It can be shown that the first term is $\lambda tr(S_B)$ and the second term is $tr(S_W)$. $d(C_i, C_j)$ measures the average margin between two classes. Therefore, maximizing our objective produces large margin linear discriminants.

We note that the objective (Eq. 5) is similar to $tr(S_B - S_W)$ proposed in [13]. The key differences are (1) λ that controls the contributions of S_B and S_W; (2) how the objective (Eq. 5) is optimized; and (3) the convergence result for the proposed method.

V. Solving Objective With Semi-Definite Programming

Suppose that w optimizes the objective function (Eq. 5). So does cw for any constant $c \neq 0$. Thus we require that w have unit length. The optimization problem then becomes $\max_w tr(w^t(\lambda S_b - S_w)w)$, subject to: $\|w\| = 1$. This is a constraint optimization problem. Since $tr(w^t(\lambda S_b - S_w)w) = tr((\lambda S_b - S_w)ww^t) = tr((\lambda S_b - S_w)X)$, where $X = ww^t$, we can rewrite the above constraint optimization problem as

$$\max_X \quad tr((\lambda S_b - S_w)X) \qquad I \bullet X = 1 \quad \text{and} \quad X \succeq 0 \tag{6}$$

where I is the identity matrix and the inner product of symmetric matrices is $A \bullet B = \sum_{i,j}^n a_{ij} b_{ij}$, and $X \succeq 0$ means that the symmetric matrix X is positive semi-definite. Indeed, if X is a solution to the above optimization problem, $X \succeq 0$ and $I \bullet X = 1$ implies $\|w\| = 1$, assuming rank$(X) = 1$.

The above problem is a semi-definite program (SDP). It is a convex optimization problem, where the objective is linear with linear matrix inequality and affine equality constraints. SDPs arise in many applications, including sparse PCA, learning kernel matrices, Euclidean embedding, and others. In addition, semidefinite programming is a very useful technique for solving many problems. For example, SDP relaxations can be applied to clustering problems such that after solving a SDP, final clusters can be computed by projecting the data onto the space spanned by the first few eigenvectors of the SDP solution. For large-scale problems, there is a tremendous opportunity for exploiting special structures in the problems, as those suggested in [15], [16].

Assume rank$(X) = 1$. Since X is symmetric, one can show that rank$(X) = 1$ iff $X = ww^t$ for some vector w. Therefore, we can recover w from X as follows. Choose any column (say the ith column) of X such that $X(1,i)$, 0, and let $w = X(:,i)/X(1,i)$, where $X(:,i)$ denotes the ith column of the matrix X. Therefore, our goal is to ensure that the solution X to the above constraint optimization problem has rank at most 1. It turns out that the above formulation (6) is sufficient to ensure that the rank of the optimal solution X to Eq. (6) is one, i.e., rank$(X) = 1$.

Theorem 1: Let X be the solution to the semidefinite program (6). Also, let $rank(X) = r$. Then $r = rank(X) = 1$.

The proof is omitted due to the limit of space. Therefore, our procedure for computing w from the matrix X is guaranteed to produce the correct answer. We call our algorithm Margin LDA, or M_{LDA}. While the criterion (Eq. (5)) is different from the Fisher criterion (Eq. (1)), it is very competitive, as we shall see later in the experimental section.

For multiple class problems or matrices whose rank is greater than 1, as in Σ_I and Σ_E, a subspace with $d > 1$ dimensions must be found. We start with $A_1 = \lambda S_b - S_w$, where S_b and S_w are computed as in the two class case. We solve the problem in (6) to obtain the solution $X_1 = w_1 w^t_1$. Once we have obtained the solution $X_j = w_j w^t_j$, we deflate A_j to obtain $A_{j+1} = A_j - X_j$, from which we compute the solution $X_{j+1} = w_{j+1} w^t_{j+1}$ to A_{j+1} for $j = 1, \cdots, C-1$, where C represents the number of classes. Notice that X_js force w_{j+1} to be orthogonal to w_j s, as desired.

In terms of the computational complexity, generic methods for solving the semidefinite program (Eq. (6)) have a complexity of $O(q^3)$ [17]. In the case of $n \ll q$ (high dimensional data), one can apply QR decomposition to remove the null space of $S_t = S_w + S_b$, which does not contain any useful information. Let $r_t = rank(S_t)$. It follows that $r_t \leq n - 1$. Thus, in the resulting space, solving the semidefinite program (Eq. (6)) takes time at most $O(r_t^3)$. Since QR decomposition has a complexity of $O(qn^2)$, which dominates the overall computation, we have a complexity of $O(qn^2)$. Therefore, for a given problem, the computational complexity of solving the semidefinite program (Eq. (6)) is $O(q \min(q,n)^2)$. Note that once a problem is given, the number of classes (fixed) does not affect complexity analysis.

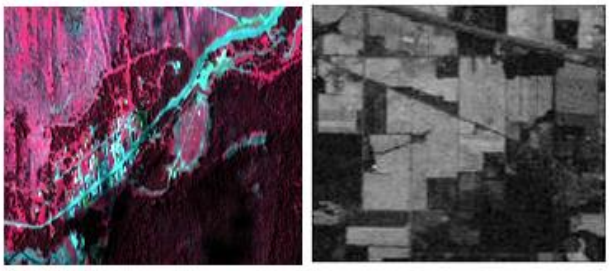

Fig. 1. Sample images of Cooke and Pines data.

VI. Experiments

In these experiments, we compare the following competing methods for dimensionality reduction in hyperspectral imagery. All procedural parameters are determined through 5-fold cross-validation. To solve the semi-definite program, we used the general purpose optimization software SeDuMi [18].

(1) DB_{IE} -Bhattacharyya distance with the intra-class and extra-class covariance matrices Σ_I and Σ_E, respectively. (2) $Fisher_{IE}$ -Fisher criterion with Σ_I and Σ_E, respectively. (3) M_{IE} - Proposed M_{LDA} algorithm (Eq. 6) with Σ_I and Σ_E, respectively. (4) $Fisher_{HM}$ -Fisher criterion with the near hit and near miss scatter matrices S_h and S_m (Eq. 3), respectively, and three

nearest neighbors are used to compute near hits and near misses. (5) M_{HM}-Proposed M_{LDA} algorithm (Eq. 6) with S_h and S_m, respectively, and three nearest neighbors are used to compute near hits and near misses. (6) M_{LDA} -Proposed M_{LDA} algorithm (Eq. 6) with the between and within class matrices S_w and S_b, respectively. (7) *LDA*-LDA algorithm (Eq. 1) with the between and within class matrices S_w and S_b, respectively. (8) *WaLuMI*-Ward's linkage strategy using mutual information [19], [20].

In the experiments, the mean and variance along each dimension are calculated using the training data. Then, the training mean and variance are used to normalize the test data. The one nearest neighbor (NN) classifier is used to obtain accuracy.

A . Data Sets

(1) **Cooke** - This data set includes a hyperspectral image and regions of interest (ROIs) indicating target locations (ground truth) in the image. The image was captured by the HyMap airborne hyperspectral sensor over Cooke City, Montana, on July 4, 2006. It consists of 280 × 800 pixels with ground sampling distance of about 3 meters and 126 spectral channels in the VNIR-SWIR range. A false color representation of the scene is shown in the left panel in Figure 1. In this experiment, we only use self test set, due to the availability of ground truth. There are seven classes in this data set. The data set can be obtained from dirsapps.cis.rit.edu/blindtest/.

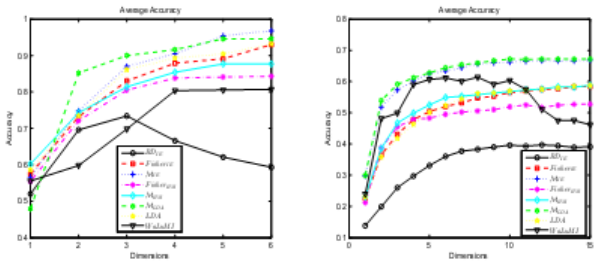

Fig. 2. Average accuracy as a function of dimensions achieved by each method on the four data sets. Left: **Cooke** and right: **Pines**.

(2) **Pines**-This data set was generated by the AVIRIS sensor over the Indian Pines test site in North-western Indiana. It consists of 145 ×145 pixels and 224 spectral reflectance bands in the wavelength range 0.4-2.5 $10^{(}$ −6) meters. In this experiment, the number of bands is reduced to 200 after removing bands covering the region of water absorption: [104-108], [150-163], and 220. The image is shown in the right panel in Figure 1. The Indian Pines scene contains mostly agriculture, forest and other natural perennial vegetation. There are sixteen classes in the dataset. Indian Pines data are available through Pursue's university MultiSpec site.

B. Experimental Results

For the **Cooke** data set, we randomly choose 5 samples from each class as training. Thus, we have a total of 35 training samples. Similarly, we randomly choose 4 from the remaining examples from each class as testing, resulting in a total of 28 testing examples. For the **Pines** data set, **Oat** has only 20 examples. Therefore, we randomly choose 10 samples from each class as training and choose randomly 10 from the remaining examples as testing. Thus, we have 160 training and 160 testing examples, respectively.

For all the methods, we obtain a projection from the training data. We then project both training and test data on the chosen features and use the one nearest neighbor classifier to obtain classification accuracy. This process is repeated 20 times and the average accuracy for each method over the 20 runs is reported.

Figure 2 shows the average accuracy registered by each method on the two data sets as a function of dimensions. Overall the M_{IE} and M_{LDA} achieve good performance in the problems that we have experimented with. M_{IE} and M_{LDA} are most competitive, followed by *LDA*. These results corroborate well with our theoretical analysis.

VII. Conclusion

In this paper, we have shown the relationship between feature selection based on intraclass and extraclass spaces and Relief, thereby providing a justification for the two class technique. We have introduced a margin criterion for dimension reduction that does not involve matrix inversion, and how our objective can be optimized using algorithms such as semi-definite programming. We have demonstrated the efficacy of the proposed technique using hyperspectral image data sets and the results show that the technique achieves competitive performance against competing methods in these data sets.

References

[1] G. Camps-Valls and L. Bruzzone, "Kernel-based methods f or hyperspec-tral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, 2005.

[2] J. Li, J. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentationusing a new Bayesian approach with active learning," *IEEE Trans.Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3947–3960, 2011.

[3] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "Asurvey of active learning algorithms for supervised remote sensing imageclassification," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 606–617, 2011.

[4] D. Landgrebe, "Hyperspectral image data analysis as a high dimensionalsignal processing problems," *IEEE Signal Processing Magazine*, vol. 19,no. 1, pp. 17–28, 2002.

[5] K. Etemad and R. Chellappa, "Discriminant analysis for recognitionof human faces," in Proc. Int'l Conf. Acoustics, Speech, and Signal Processing, 1996, pp. 2148–2151.

[6] R. Duda, P. Hart, and D. Stork, *Patten Classification, 2nd edition* . John-Wiley, 2000.

[7] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic Press, 1990.

[8] S. Zhang and T. Sim, "Discriminant subspace analysis: A fukunaga-koontz approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 10, pp. 1732–1745, 2007.

[9] L. F. Chen, H. Y. Liao, M. T. Ko, J. C. Lin, and G. J. Yu, "A new lda-based face recognition system which can solve the small sample size problem," *Pattern Recognition*, vol. 33, pp. 1713–1726, 2001.

[10] K. Kira and L. A. Rendell, "A practical approach to featu re selection," in *Proc. of 9th International Conference on Machine Learning*, 1992, pp. 249–256.

[11] Y. Sun and D. Wu, “A relief based feature extraction algorithm,” in *SIAM International Conference on Data Mining*, 2008.

[12] W. Zhang, X. Xue, Z. Sun, Y. Guo, and H. Lu, “Optimal dimensionality of metric space for classification,” in *ICML*, 2007.

[13] H. Li, T. Jiang, and K. Zhang, “E*ffi*cient and robust feature extraction by maximum margin criterion,” *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 157–165, January 2006.

[14] R. Gilad-Bachrach, A. Navot, and N. Tishby, “Margin bas ed feature selection - theory and algorithms,” in *ICML*, 2004.

[15] A. Ben-Tal and A. Nemirovski, “Non-euclidean restrict ed memory level method for large-scale convex optimization,” 2004.

[16] I. Nesterov, “Smooth minimization of non-smooth functions,” 2003.

[17] L. Vandenberghe and S. Boyd, “Semidefinite programming ,” *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.

[18] J. F. Sturm, “Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones,” *Optimization Methods and Software*, vol. 11, no. 12, pp. 625–653, 1999.

[19] A. Mart´ınez-Us´o, F. Pla, , J. S. P., and Garc´ıa-Sevil la, “Comparison of unsupervised band selection methods for hyperspectral imaging,” in *Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part I*, ser. IbPRIA'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 30–38.

[20] J. Ward, “Hierarchical grouping to optimize an objective function,” *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, 1963.