Waveband selection for hyperspectral data: optimal feature selection

Invited Paper

David Casasent[a] and Xue-Wen Chen[b]

[a]Dept. of Electrical & Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213
[b]Dept. of Electrical & Computer Engineering, California State University, Northridge, CA 91330

ABSTRACT

Hyperspectral (HS) data contains spectral response information that provides detailed chemical, moisture, and other descriptions of constituent parts of an item. These new sensor data are useful in USDA product inspection and in automatic target recognition (ATR) applications. However, such data introduces problems such as the curse of dimensionality, the need to reduce the number of features used to accommodate realistic small training set sizes, and the need to employ discriminatory features and still achieve good generalization (comparable training and test set performance). HS produces high-dimensional data; this is characterized by a training set size (N_i) per class that is less than the number of input features (HS λ bands). A new high-dimensional generalized discriminant (HDGD) feature extraction algorithm and a new high-dimensional branch and bound (HDBB) feature selection algorithm are described and compared to other feature reduction methods for two HS product inspection applications. Cross-validation methods, not using the test set, select algorithm parameters.

Key words: feature extraction, feature reduction, feature selection, hyperspectral data, product inspection.

1. INTRODUCTION

Hyperspectral (HS) data is very powerful and provides excellent detailed information useful for many USDA product inspection applications. Initial HS results have been achieved in determining wheat grain quality [1-5], in analyzing the quality of rice [6-7], in analyzing crops and weeds to control herbicide use [8-9], in determining moisture in soybean seeds [10], sugar content in potatoes [11], sausage quality [12], raisin quality [13], fat content in salmon [14], almond nut quality [15-16] and corn quality [17].

Use of HS data introduces new problems due to the high dimensionality N_F of the input data (the spectral response at N_F different wavelengths λ produces N_F features). When the number of training samples N_i (per class i) does not satisfy [18] $N_i > 10N_F$, then *generalization* problems are expected [19], i.e, training and test set classification performance P_c differ. The training set size is rarely adequate and thus this issue is of concern in HS data processing. This causes poor estimates of matrices such as the covariance matrix C and its inverse C^{-1} required in many discrimination algorithms. Thus, we address *dimensionality reduction* (reducing the number of features N_F used) to achieve good generalization. We also address use of discriminating features (discriminating functions); much product inspection HS work [1-14] and other HS processing [20] have surprisingly used principal component analysis (PCA) related methods that are intended for compression, not for discrimination; these methods achieve dimensionality reduction and good generalization, but yield poor P_c. Most prior food inspection work [1-14] has involved methods such as the partial least square (PLS) method [34] to locate the presence and amount of specific chemicals or elements in a given product; this does not require discriminating two types of products as our present applications do. Our applications are quite different and involve classification of good and bad products (discrimination). *Our proposed features and algorithms provide improved P_c as well as good generalization*. In all algorithms, various parameters must be selected; this is often done ad hoc; we utilize *cross-validation methods to achieve algorithm parameter selection* (using a validation set of data, separate from the test set data) for our almond database.

In this paper, we present two new feature reduction algorithms for high-dimensional (HD) HS data. They achieve dimensionality reduction and generalization plus discrimination. Most linear discriminant analysis (LDA) methods used in high dimensional space require the reduction of feature space first in order to make the covariance matrix invertible; this generally uses principal component analysis (PCA) methods, which reduce the feature space without considering the loss of discriminating information. These PCA/LDA methods have been used for face recognition [21, 22], and for HS processing. We compare our new feature extraction algorithm to PCA, LDA, and PCA/LDA methods. Our high-dimensional generalized discriminant (HDGD) feature extraction algorithm extracts discriminant features and achieves generalization without an initial PCA reduction of the original feature space, since this results in a loss of discriminating information. *Feature extraction* algorithms use all original features. For speed reasons, *feature selection* (use of only several of the original high-dimensional features) is often needed; our high-dimensional branch and bound (HDBB) feature selection algorithm achieves this. It is preferable to forward selection methods [23, 24] to select the best λ features that are widely used in HS processing. We compare our HDBB algorithm (and our optimal modified MBB algorithm) to exhaustive search and the standard BB [25] methods (these three methods are optimal) and to use of forward selection (FS) and Kullback-Leibler distance [26] (KLD) to order and select the best reduced set of features.

Sect. 2 describes the databases used. Various background methods and issues are discussed and demonstrated in Sect. 3. Our new algorithms are then presented (Section 4) followed by initial test results (Sections 5, 6, and 7).

2. DATBASES

Initial results are presented for two USDA HS databases. In the almond nut database, the central region of each almond is illuminated by a quartz lamp and the transmission spectra from from λ = 710 to 1390 nm from two different fiber optic transmission spectrometers, as detailed elsewhere [15, 16]. The data was obtained in N_F = 137 spectral bands equally spaced at $\Delta\lambda$ = 5 nm increments. The database consists of the responses for 454 almonds. The problem is to classify almond nuts as good or bad (in terms of concealed internal damage that affects taste) from HS data. The 454 nuts are divided into a training set of 228 nuts (173 good and 55 bad nuts) and a test set of 226 nuts (172 good and 54 bad); we intentionally made the training set and test set approximately equally-sized. For this database, there is not a sufficient number of samples for each class, e.g., only 109 bad nuts with only 55 bad nuts in the training set (this is less than the number N_F = 137 of features); thus, this is a high-dimensional problem and feature reduction is needed. Normalization of this HS data is needed because of the wide variations in the spectral transmittance of different nuts due to their skin quality, nut thickness, and nut shape [15, 16]. Each data sample is normalized by dividing its spectral response at each λ by the average λ response for that sample; this normalization is separately repeated for each sample (training and test sets). To select parameters for our algorithms, *cross-validation methods* [27, 28] are used on a validation set. We use 1/6 of the training set as a validation set; this selection is randomly repeated six times with a different training and validation set, and P_c(train) and P_c(valid) results are averaged and analyzed to select parameters with the best average P_c(valid) performance using a nearest neighbor (NNB) classifier. We use P_c over six sets of training and validation set data, since the validation set size ($1/6 \times 228 = 38$) is very small. We use this to select the final number of features used in our new algorithm, the number of PCA features used in PCA and PCA/LDA algorithms, and in selecting the noise parameter in our HDGD algorithm. The test set is always separate from the training and validation sets in all database tests. *Use of a validation set to select algorithm parameters does not seem to be employed in prior HS work*.

An initial set of corn kernels was also used with hyperspectral data provided by T. Pearson (ARS, Kansas) and with aflatoxin level data provided by D. Wicklow (ARS, Peoria, Il). For the corn kernel database, the central 3 mm diameter of each kernel was illuminated with a 100 W quartz lamp in a fiber-optic spectrometer and the kernel transmission spectra from 500 to 950 nm was obtained. After spectrometer measurements were obtained, each kernel sample was chemically analyzed and the amount of aflatoxin in each kernel (in units of ppb, parts per billion) was obtained. The corn kernels were grouped into good (0 ppb aflatoxin) and bad (≥ 100 ppb aflatoxin) samples. We have HS data for an initial set of 385 corn kernels in N_F = 86 spectral bands from 525 to 950 nm, each band covering an equally spaced width of $\Delta\lambda$ = 5nm. Data in the first five bands (from 500 - 520 nm) was removed because they are noisy. Of the available 385 corn kernel samples, 343 kernel samples are good and have no aflatoxin (0 ppb) and the remaining 42 samples are bad (infested) and have an aflatoxin level ≥ 100 ppb. These represent all presently available good and bad corn kernel samples. In this product inspection problem, the purpose is to classify each corn kernel as either good (aflatoxin negative, or no aflatoxin present) or bad (aflatoxin positive, or aflatoxin present). The same normalization used for the almond database was employed for the corn database. This is a high-dimensional problem with N_F = 86 features, which is larger

than the total number of bad samples 42. Since there are so few bad samples, *bootstrapping P_c scores* are used. We used 172 of the 343 good samples and half (21) of the bad samples as the training set and the remaining 171 good and 21 bad samples are used as the test set. This training and test set selection is repeated 100 times with a different random choice of 172 good and 21 bad training set samples selected each time; the average P_c scores for these 100 tests are used to select algorithm parameters and for final test results. The purpose is to classify each kernel as good or bad. USDA regulations require an average aflatoxin in level of < 20 ppb in a 5 kg random sample. Since a single kernel can have a very high aflatoxin in level (> 10,000 ppb), inspection of a larger sample is preferable and this should be possible using HS methods. Our present purpose is to determine if HS data can separate whole corn kernels. Initial results indicate that this is possible with HS data if a larger training set is available, so that the training set is representative of the test data. For the present database and our initial results, we analyze both the training and test set performance to select algorithm parameters. Prior work [17] on this corn kernel database used leave-one-out scoring (which yields questionable results) and used a Mahalanobis-distance classifier.

3. BACKGROUND AND PROBLEM DEFINITIONS

3.1 Poor Generalization

The Fisher discriminant is a widely used linear discrimination analysis (LDA) method [29]. It involves transforming the original $\boldsymbol{x}$ data of high dimension (the number of spectral features) to a new 1-D feature space $y = \boldsymbol{\varphi}^T\boldsymbol{x}$ in which good/bad products are better separated. The transformation $\boldsymbol{\varphi}$ is chosen to maximizing the Fisher ratio

$$J = \frac{\text{difference of means of projections}}{\text{sum of scatter of projections}} = \frac{\varphi^{\mathrm{T}}\mathbf{R}\varphi}{\varphi^{\mathrm{T}}\mathbf{C}_{\mathrm{w}}\varphi}, \tag{1}$$

where $\boldsymbol{R} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}}$ is the between-class covariance matrix ($\boldsymbol{\mu}_i$ is the mean vector of class i) and $\mathbf{C}_{\mathrm{w}} = \mathbf{C_1} + \mathbf{C_2}$ is the within-class covariance matrix, where $\mathbf{C_1}$ and $\mathbf{C_2}$ are the covariance matrices for good and bad classes 1 and 2. The solution φ is well-known to be the dominant eigenvector solution to the generalized eigenvalue problem

$$\mathbf{R}\varphi = \lambda\mathbf{C}_{\mathrm{w}}\varphi\,. \tag{2}$$

If $\mathbf{C}_{\mathrm{w}}$ is non-singular, the solution is

$$\varphi = \mathbf{C}_{\mathrm{w}}^{-1}\,(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\,. \tag{3}$$

Since R has rank 1, there is only one eigenvector solution. The new feature is the scalar $y = \boldsymbol{\varphi}^T\boldsymbol{x}$. For our cases, $\mathbf{C}_{\mathrm{w}}$ is ill-conditioned, but is not singular. For this reason, P_c for LDA is expected to have poor generalization, as we now show. As Table 1 shows P_c(train) is nearly perfect. However, because $\mathbf{C}_{\mathrm{w}}$ is ill-conditioned, small changes or the addition of noise to the training set can drastically change $\mathbf{C}_{\mathrm{w}}$ and $\mathbf{C}_{\mathrm{w}}^{-1}$ and hence $\boldsymbol{\varphi}$ [30]. The test set data can be viewed as the training set data with small changes; thus, quite different test set performance results and hence poor generalization, as Table 1 shows.

Table 1. Almond HS P_c results using the LDA feature

P_c(train)	P_c(test)
99.6%	83.6%

3.2 Reduced Dimensionality Methods (PCA and PCA/LDA)

The standard approaches to achieve good generalization involve dimensionality reduction of the feature space to a smaller space such that $N_i > 10N_F$ is satisfied. These methods are variations of principal component analysis (PCA) or Karhunen-Loeve (KL) analysis (Partial-least squares (PLS) is a related method for different applications). We now review PCA for dimensionality reduction. The full covariance matrix **C** of all the data (in both classes) is calculated. To reduce the dimensionality of the data from N_F to $d < N_i$, while preserving the most information in the original data, we calculate the d dominant eigenvectors of $\boldsymbol{C}$ and denote them as $\boldsymbol{v_i'}$. We form a $N_F \times d$ matrix $\mathbf{A}$ with the $\boldsymbol{v_i'}$ as its columns (each of dimension N_F). We then transform the original $\boldsymbol{x}$ data (of dimension N_F) to a reduced $d < N_F$ dimensional space with new features $\mathbf{y} = \mathbf{A}^T\boldsymbol{x}$. This projects the high-dimensional original data onto the eigenvectors $\boldsymbol{v_i'}$ to produce new features of reduced dimension. To demonstrate that these PCA features provide generalization, we consider the almond database. From cross-validation, we found that m = 6 PCA features was best. Table 2 shows the classification results using PCA features. As seen, generalization is excellent. However, the scores do not represent the best P_c. This is expected, since PCA uses only the dominate eigenvectors; these contain the information that is common to both the good and bad classes of data. It is best for representation of both classes of data (hence generalization is good); but it is not best for discrimination between the two classes of data (hence, P_c is poor). In practice, the lower eigenvectors are expected to contain information that is more useful for discrimination between the two classes of data. Thus, *use of dominate vectors (PCA, KL, PLS, etc) is not expected to give good P_c results*. This vital point is emphasized as it seems to largely not be appreciated.

Table 2. HS almond P_c results using m = 6 PCA features

P_c(train)	P_c(valid)	P_c(test)
85.1%	83.8%	84.1%

To improve the P_c of PCA features, LDA has been applied to the PCA features; an LDA combination of PCA features is produced. This is referred to as PCA/LDA feature extraction. Table 3 shows the results obtained for the almond database. Generalization is good, as expected, due to the PCA feature reduction step. However, P_c performance is not noticeably better than when only PCA features were used. This is due to the fact that the data is not linearly separable in LDA feature space. This is not unexpected. This notes yet another disadvantage of LDA feature extraction: it only provides one resultant feature. An NNB classifier is used in Table 2-3 and utilizes the more than one PCA features in Table 2, thus improving its P_c. Our new HDGD feature extraction algorithm also provides more than one feature. This is expected to improve results, since now higher-order decision surfaces can be used in the classifier.

Table 3. HS almond P_c results using the PCA/LDA feature (with m = 8 PCA features)

P_c(train)	P_c(valid)	P_c(test)
85.5%	85.6%	85.8%

3.3 Waveband Selection

For different HS applications, the use of fewer λ is preferable, as it can lead to faster sensor systems. With LEDs or laser diodes at only several λ used, more detected light in certain λ bands is possible; hence, the total integration time can be reduced, and thus the number of samples inspected per second can be increased. Similarly, a set of several optical filters can be used to allow rapid imaging at several λ onto several detector arrays. In waveband selection, we select a reduced number of the original features. Several summaries of feature selection methods exist [31, 32], with no method being clearly best for high-dimensional data (although versions of forward selection perform well). Recall that optimal feature selection is an N-P complete problem. Only an exhaustive search can determine the best set of m features out of N_F original features [33]. Forward selection has been found [32] to be one of the better methods to order a set of features by which are best. However, it exhibits nesting problems. It adds additional features to previously selected features such that the combination gives better performance than the prior set of features did. However, the best set of 3 features may

not contain the best set of 2 features, etc. Forward selection only orders the set of features by which are best; it will not tell us how many features to use (we use cross-validation methods for this). We also considered use of the Kullback-Leibler distance (KLD) [38] to select a subset of m features, since it selects features which are most different (and does not require inversion of **C**); however, the most different features are not necessarily best for discrimination. Exhaustive search methods are the only optimal methods to select the best subset m of n original features. The branch and bound (BB) algorithm [25] is an efficient optimal solution for the best feature subset. However, neither method is computationally practical for high-dimensional feature data.

4. HIGH-DIMENSIONAL FEATURE REDUCTION ALGORITHMS

4.1 High-Dimensional Generalized Discriminant (HDGD) Feature Extraction Algorithm

For high-dimensional feature extraction, we rewrite $\mathbf{C}_w$ as $\mathbf{C_w} = \sum_{\mathbf{i}=1}^{\mathbf{n}} \lambda_\mathbf{i} \mathbf{v_i} \mathbf{v_i^T}$, where λ_i and $\mathbf{v}_i$ are the *i*-th eigenvalues and eigenvectors of $\mathbf{C}_w$. The matrix inverse is then $\mathbf{C_w^{-1}} = \sum_{\mathbf{i}=1}^{n} \frac{\mathbf{v_i} \mathbf{v_i^T}}{\lambda_\mathbf{i}}$ and the LDA solution becomes

$$\boldsymbol{\varphi} = \sum_{i=1}^{n} \frac{\mathbf{v}_i \mathbf{v}_i^T}{\lambda_i} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = \sum_{i=1}^{n} \frac{\mathbf{v}_i^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}{\lambda_i} \mathbf{v}_i \,. \tag{4}$$

From this spectral decomposition of $\mathbf{C}_w$, we see that $\lambda_\mathbf{i} = \mathbf{v_i^T C_w v_i}$ and that the denominator λ_i in (4) is the variance of the projections of the two classes onto the corresponding eigenvector. Eq. (4) shows that the LDA solution is just a weighted linear combination of the n eigenvectors $\mathbf{v}_i$ of the within-class covariance matrix $\mathbf{C}_w$. The linear combination is heavily weighted by the smallest eigenvectors (these are unreliable) and this leads to an unreliable (noise sensitive) estimate of $\mathbf{C_w^{-1}}$ and of φ and hence poor generalization. Thus, LDA involves projecting the data onto all of the eigenvectors $\mathbf{v}_i$ of $\mathbf{C}_w$ and then linearly combining all of these projections.

In our new HDGD algorithm, we use only some of the eigenvectors $\mathbf{v}_i$ and as our features we use the projections onto these $\mathbf{v}_i$. Thus, we produce multiple output features, rather than just one (as in LDA) and we thus expect better performance. To select the $\mathbf{v}_i$ to use, we add zero-mean Gaussian noise to the training set data, we note the amount $\Delta\lambda_i$ by which the eigenvalue λ_i for a given $\mathbf{v}_i$ changes. For each eigenvector $\mathbf{v}_i$, we calculate the new performance measure

$$J_i = | \mathbf{v_i^T} (\boldsymbol{\mu_1} - \boldsymbol{\mu}_2) |^2 / (\lambda_i + | \Delta\lambda_i | / \lambda_i) \,. \tag{5}$$

We order the $\mathbf{v}_i$ by J_i and select the number of HDGD features and the amplitude of noise by cross-validation. In (5), the numerator of J_i is large if the mean separation of the projections of the two classes is large (i.e., if discrimination is good) and the denominator of J_i is small if $\Delta\lambda_i$ is small (i.e., if generalization is good, if the eigenvalue is reliable) and if the variance of the projection (λ_l) is small. Thus, the $\mathbf{v}_i$ so selected have good discrimination, low scatter, and good generalization.

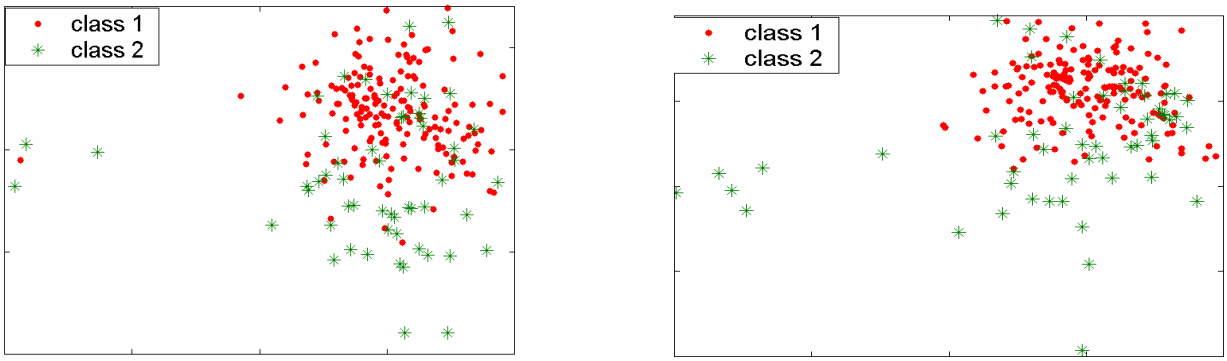
4.2 High-Dimensional Branch and Bound (HDBB) Feature Selection Algorithm

To select the best m out of N_F high-dimensional features to use, we use KLD to select an initial subset of 30 features (more are possible, as the number of subsequent calculations is not excessive). We then use a modified branch and bound (MBB) algorithm to select the best m of N_F features. This is our new HDBB feature selection algorithm. This is still an optimal search algorithm over the reduced set of initial features. It is faster than BB and much faster than an exhaustive search. We use cross-validation to select m. To describe our MBB algorithm, we now review the basic BB algorithm [25].

In the BB algorithm, a search tree is produced with n - m levels to select m features out of n, and one feature is omitted at each level of the tree. The algorithm first follows one complete path of n - m nodes to the bottom of the tree and for that path (set of m features) it computes an estimate (a bound B) on the criteria function J being used. Larger J values are better. J is then evaluated at the nodes at a given level of the tree; if J < B for a given node, then J need not be evaluated at nodes under that node (J decreases as we proceed down the tree, since fewer features are then used, i.e., more features are omitted). In our MBB algorithm, we use the Bhattacharya distance as the criteria function J. The modifications in our MBB algorithm include: efficient evaluation of paths through the tree with no branches [35], starting searches at levels 1/4 and 1/2 of the way down the tree, use of an ordered set of features (nodes) by which are best using forward selection, and calculation of a good initial bound B (using the m best features chosen by forward selection). At the upper levels of the search tree, we do not expect J < B, since only one or two features (the best ones by forward selection) are omitted. Thus, starting searches further down the tree speeds-up the search (as J must typically be evaluated for most of the nodes at the top of the tree). Ordering the features by some estimate of which features are best (the original BB algorithm uses an arbitrary order) significantly speeds up the search since the best features are at the top of tree and omitting these "better" features first is expected to produce a J < B sooner (hence, J need not be evaluated for nodes below such a node). Rapid selection of a good initial large B (using features ordered by any estimate of which are best, such as forward selection) also makes J < B more likely to occur earlier in the tree search (thus, allowing evaluation of J to be omitted for regions of the tree).

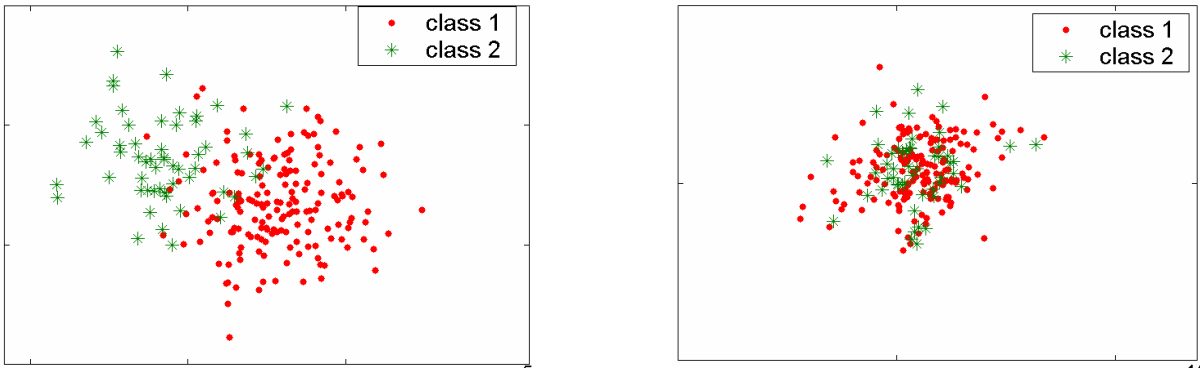
5. FEATURE EXTRACTION ALGORITHMS (ALMOND DATABASE)

For visualization, we project the original feature space (137) onto a 2-D space. Figure 1 shows the PCA projections for the two best PCA vectors (those with the largest eigenvalues of C_w) for the training and test sets; Figure 2 shows projections onto the two eigenvectors with the two smallest eigenvalues of C_w for the training and test sets; Figure 3 shows the projections onto the two best vectors for our HDGD algorithm for the training and test sets. Class 1 samples are shown as '•' and class 2 samples as '⋆'. As seen, the samples in the two classes after projection onto the PCA vectors overlap considerably (thus, poor discrimination is expected); training samples in the two classes after projection onto the eigenvectors corresponding to the smallest eigenvalues of C_w are well separated, but test samples are not (this clearly demonstrates that eigenvectors corresponding to smallest eigenvalues are sensitive to noise or small changes; thus, we expect poor generalization in LDA); for our HDGD algorithm, the samples in the two classes are well separated for both the training and test sets (thus, we expect good generalization and good discrimination).



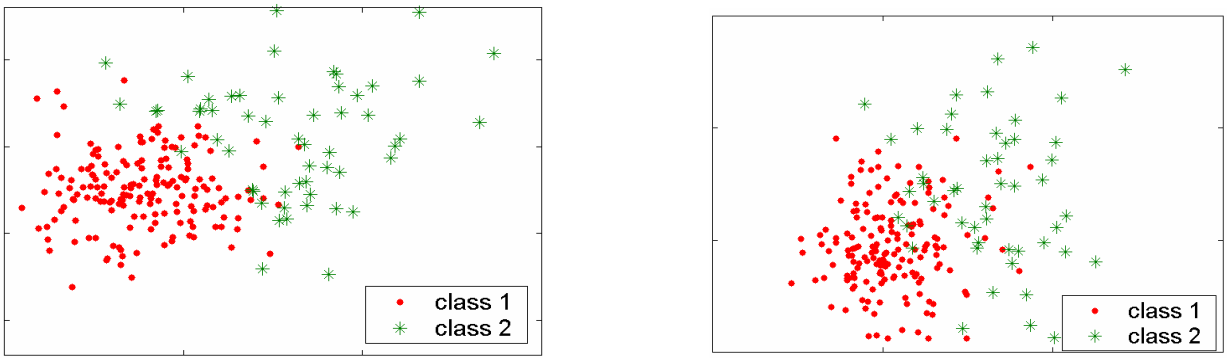
(a) (b)

Figure 1. 2-D projections onto the two best PCA vectors for the almond database. (a) training set and (b) test set.



(a) (b)

Figure 2. 2-D projections onto the two smallest eigenvectors of C_w for the almond database. (a) training set and (b) test set.



(a) (b)

Figure 3. 2-D projections onto the two best HDGD algorithm vectors for the almond database. (a) training set and (b) test set.

From cross-validation tests, we selected: the σ_n noise level used in the HDGD algorithm ($\sigma_n = 0.01\sigma_s$, where σ_s is the variance of the training set), the number of final features in PCA feature extraction (m = 5) and in the PCA step (m = 8) in the PCA/LDA algorithm, and the number of our HDGD features used (m = 4). Table 4 summarizes P_c scores for the four feature extraction algorithms. As expected and noted earlier, generalization is poor for LDA features, generalization is excellent but P_c is poor for PCA features and P_c is poor for PCA/LDA features. Our new HDGD features perform best by a significantly 6 - 7%. *Thus, almond nut classification is possible using HS data and our new HDGD feature extraction algorithm.*

Table 4. Classification results using PCA, LDA, PCA/LDA, and HDGD algorithms on the almond database.

	LDA	PCA	PCA/LDA	HDGD
Pc(train)%	99.6	85.1	85.5	92.5
Pc(test) %	83.6	84.1	85.8	90.7

6. FEATURE SELECTION ALGORITHMS (ALMOND DATABASE)

To compare the computation times for our modified Branch and Bound (MBB) algorithm to these for the BB and exhaustive search algorithms, we first used the KLD algorithm to reduce the number of original λ wavebands from 137 to 30. The three optimal feature selection algorithms were then used to select different smaller feature subsets from the reduced 30-dimensional feature space. Figure 4 is a semi-log scale plot, in which a logarithmic (base 10) axis is used for the Y-axis. It shows the number of times that the three algorithms had to evaluate the criteria function J for different

numbers of final selected features. The number of calculations of J is a measure of the speed of the different algorithms. By definition, exhaustive search evaluates all possible subsets. When a small number of features is selected, the exhaustive search algorithm is faster than BB based methods. When the number of selected features exceeds three, our MBB algorithm is more efficient than both the exhaustive search algorithm and the basic BB algorithm. When more than four features are selected, both BB based algorithms are faster, with our MBB algorithm being the most efficient and with the number of calculations of J in the exhaustive search algorithm increasing exponentially, as expected.

We used both forward selection (FS) and KLD to select the set of 30 features to which our fast MBB algorithm was applied to produce different numbers m of final features. Table 5 lists the best 1 through 7 features selected by our MBB algorithm from the 30 initial FS and KLD features. The numbers represent the number of the feature (out of 137). Each list of best features is quite different; however, only 8 of the 30 features selected as best by the two algorithms were the same **(**features 1, 46, 86, 90, 93, 101, 103, 130).

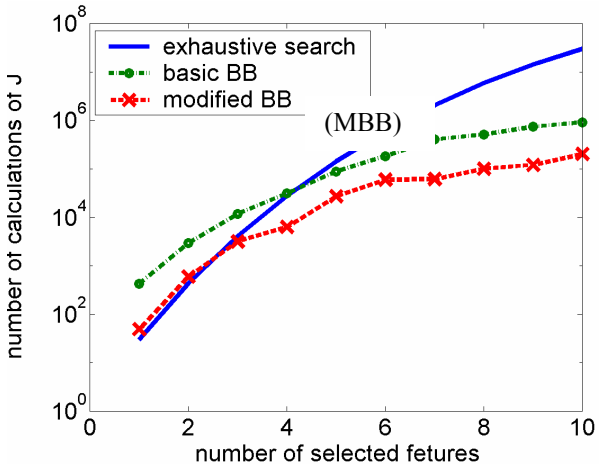


Figure 4. Comparison of the computation efficiency of the optimal feature selection algorithms on the almond database.

Table 5. Optimal subsets for the almond database

The number of features	FS/MBB	HDBB (KLD/MBB)
1	43	40
2	21, 100	12, 54
3	3, 86, 98	1, 23, 40
4	28, 39, 43, 107	1, 23, 40, 78
5	10, 11, 42, 43, 53	1, 23, 40, 72, 78
6	11, 43, 53, 54, 93, 94	12, 35, 40, 46, 54, 103
7	11, 46, 49, 53, 94, 95, 103	1, 23, 40, 46, 72, 78, 103

The FS/MBB data shows the "nesting" problem in FS feature selection, i.e., the best subset of four features (28, 39, 43, 107) does not contain any feature in the best subset of three features (3, 86, 98) or two features (21, 100). The subsets selected by the HDBB algorithm (the MBB algorithm used to select a subset of features from the 30 features selected by KLD) are seen to be very different from those selected by FS/MBB. For the best subsets of one to three features selected by the HDBB algorithm, there is also little similarity. Our HDBB algorithm selects two new features 12 and 35 in the subset of six features, which are not shown in the subsets of three to five features; feature 12 is selected in the subset of two features. Thus, *an additional optimal algorithm (such as MBB) is needed* to select the best subset of m final features out of an initial larger set of n = 30 features selected by some suboptimal algorithms to order features (such as FS or KLD). Features 1, 23, 40 etc. seem to consistently be in most of the best HDBB subsets of three or more features. Adjacent features seem to be useful as the FS/MBB algorithm selects features 42 and 43 in the subset of five features and features 53 and 54 in the subset of six features. In the set of 30 best bands selected by KLD, adjacent bands are also

selected (bands 85, 86, 87 and 88). Thus, combining adjacent λ responses to reduce the number of features is not advisable and the $\Delta\lambda$ = 5 nm initial resolution seems to be needed.

To select the final number of features m to use, we used cross-validation for the KLD/MBB feature subsets in Table 5 and found the best P_c(valid) to occur with m = 6 features. We thus choose m = 6 features for all algorithms and calculated P_c for each case to confirm which feature selection method performance best. Table 6 lists the results. Each algorithm selected different subset of the six best features. The best six features ordered by KLD perform worse than the best six features selected by FS for this database. However, the best six features chosen by an optimal selection method (such as MBB) from a larger set of 30 best FS or KLD features give better performance. *Such optimal selection methods are thus needed and our HDBB algorithm (using KLD to select the initial subset of features) performs best.*

Table 6. Classification results with 6 selected features for the almond database

Algorithm	KLD	FS	FS/MBB	HDBB
Features selected	1, 46, 55, 78, 101, 128	11, 43, 48 53, 90, 130	11, 43, 53 54, 93, 94	12, 35, 40 46, 54, 103
P_c(train)%	77.2	82.3	84.2	86.0
P_c(test)%	79.2	81.8	83.5	85.9

Thus, feature selection (using only six specific λ features vs 137) is also useful for HS inspection (with a reduction of ≈ 4% in performance for this database). Feature selection is needed to achieve the high-speed inspection rates needed.

It is interesting to compare the subset of the best four features selected by our HDBB algorithm with the best subset of four features chosen from all 137 original features using the optimal algorithms. We applied our MBB algorithm and exhaustive search to the full original data (137 features) to select the best subset of four features. We then applied our HDBB algorithm (MBB applied to 30 KLD features). Table 7 lists the subsets of the best four features selected by the three algorithms. Table 7 also notes the number of times J was evaluated for the three algorithms (this is a measure of the computation load for each algorithm) and the associated computer time required for each algorithm on a 250 MHz Pentium PC. Our MBB algorithm is about twice as fast as an exhaustive search, while our HDBB algorithm is much faster (by a factor of 1600) than exhaustive search. As can be seen, *the HDBB algorithm is much faster* than both the exhaustive search and MBB algorithms (when applied to all 137 original features). The best subset of four features chosen was the same for the MBB and exhaustive search algorithms (as expected). The best subset selected by our HDBB algorithm has three common features (1, 23, and 40) with the other algorithm. The only different feature is that band 81 was chosen by the optimal algorithms and band 78 was selected by our HDBB algorithm. All methods generated the same *J* value (= 0.1064) for the best subset of four features. Band 81 was not selected as one of the 30 best features by the KLD algorithm and thus it could not be included in the final four HDBB features. Analysis showed negligible difference in the response for bands 78 and 81 over the 228 samples in the training set. Thus, the four best bands selected by all three algorithms are essential equivalent.

Table 7. The subset of four features selected from the original 137 features for almond HS data

Algorithm	Number of calculations of J (times)	Four features selected
Exhaustive search	14,043,870 (seven days)	1, 23, 40, 81
MBB	7,541,999 (four days)	1, 23, 40, 81
HDBB	15,614 (six minutes)	1, 23, 40, 78

7. INITIAL HS CORN KERNEL DATABASE RESULTS

Recall that all of these data used bootstrapping and are the average of 100 different training and test set scores.

7.1 Feature Extraction Results

For the HDGD algorithm, we chose $\sigma_n = 0.02\sigma_s$ for the added noise level. We increased the number of features used and selected m for the different algorithms that gave the largest P_c(test). This resulted in use of eight final PCA features, four features in the PCA portion of the PCA/LDA algorithm, and seven of our HDGD features. Table 8 lists the

total P_c test set scores for the four different feature extraction algorithms. Our HDGD features perform best. Surprisingly, PCA features perform quite good. PCA/LDA features perform poorly, since use of more than one final feature seems essential.

Table 8. Test results for the four feature extraction algorithms for the corn kernel database

Algorithm	PCA/LDA	LDA	PCA	HDGD
P_c(test)%	89.4	93.4	95.5	96.7

These initial results are encouraging. New neural net classifiers are expected to provide even better results [36, 37], but will require a larger infested kernel database.

7.2 Feature Selection Results

For completeness, feature selection was also addressed to confirm that it also appears possible. Table 9 summarizes the test set results. These tests are very extensive; they involved analysis of 100 different sets of optimal λ bands (one for each of the 100 choices of the training and test set) and employing the MBB algorithm for each of the different training sets. From P_c(test) bootstrapped data, we found that five features were best (for our HDBB algorithm) and thus, as before, all P_c scores used five final output features. Average scores over 100 runs for our HDBB algorithm are slightly higher than for feature extraction (P_c = 97.5% versus 96.7). Thus, feature selection seems possible using only five λ bands (allowing faster inspection rates). Tests on a larger database are thus merited.

Table 9. Test results for the feature selection algorithms (using the best five features) for the corn kernel database

Algorithm	KLD	FS	FS/MBB	KLD/MBB (HDBB)
P_c(test)%	90.2	93.8	94.0	97.5

8. SUMMARY AND CONCLUSIONS

Our new feature extraction (HDGD) and feature selection (HDBB) algorithms for high-dimensional data were described. On two HS databases, we found them preferable to others. Out feature selection algorithm uses an improved branch and bound algorithm. Such an optimal subset feature selection algorithm is needed as we have shown; standard methods cannot provide the best ordered set of features, since nesting problems occur in optimal subsets of different sizes. Our new techniques appear very useful for product inspection using HS data. Other tests have shown their use on several different ATR object detection databases.

ACKNOWLEDGEMENT

We thank Dr. Tom Pearson (ARS, Manhattan, Kansas) for providing the HS databases and Dr. Donald Wicklow (ARS, Peoria, IL) for supplying the aflatoxin corn ppb data.

REFERENCES

1. D. Archibald, D. Funk, F. Barton II, "Locally weighted regression for accessing a database containing wheat grain NIR transmission spectra and grain quality parameters*" Proc. SPIE,* vol. 3543, pp.141-151, 1999.
2. D. Archibald, C. Thai, and F. Dowell, "Development of short-wavelength near-infrared spectral imaging for grain color" *Proc. SPIE* vol. 3543, pp. 189-198, 1999.
3. F. Dowell, M. Ram, and L. Seitz, "Predicting scab, vomitoxin, and ergosterol in single wheat kernels using near-infrared spectroscopy" *Cereal Chem*. vol. 76(4), pp. 573-576, 1999.

4. F. Dowell, J. Throne, and J. Baker, "Automated nondestructive detection of internal insect infestation of wheat kernels by using near-infrared reflectance spectroscopy" *J. Econ. Entomol*. vol. 91(4), pp. 899-904, 1998.
5. F. Dowell, "Detecting vitreous and non-vitreous durum wheat kernels using near-infrared spectroscopy" in *1999 ASAE Annual International Meeting*, paper No. 993082, 1999.
6. A. Adams and D. Herden, "Spectral reflectance of rice seedings" *Proc. SPIE* vol. 3543, pp. 259-264, 1999.
7. C. Villareal, N. De La Cruz, and B. Juliano, "Rice amylose analysis by near-infrared transmittance spectroscopy" *Cereal Chem*. vol. 71(3), pp. 292-296, 1994.
8. E. Vrindts and J. Baerdemaeker, "Optical weed detection and evaluation using reflection measurements" *Proc. SPIE* vol. 3543, pp. 279-289, 1998.
9. J. Favier, D. Tsheko, D. Kennedy, A. Muir, and J. Fleming, "Discrimination of weeds in brassica crops using optical spectral reflectance and leaf texture analysis" *Proc. SPI*E vol. 3543, pp. 311-319, 1998.
10. D. Lamb and C. Hurburgh, "Moisture determination in single soybean seeds by near-infrared transmittance" *Trans. of the ASAE*, vol. 34(5), pp. 2123-2129, 1991.
11. M. Mehrubeoglu and G. Cote, "Determination of total reducing sugars in potato samples using near-infrared spectroscopy" *Cereal Foods World*, vol. 42(5), pp. 409-413, 1997.
12. M. Ellekjare, T. Isaksson and R. Solheim, "Assessment of sensory quality of meat sausages using near-infrared spectroscopy" *J. of Food Science*, vol. 59(3), pp. 456-464, 1994.
13. C. Huxsoll, H. Bolin, and B. Mackey, "Near infrared analysis potential for grading raisin quality and moisture" *J. of Food Science*, vol. 60(1), pp. 176-180, 1995.
14. H. Sollid and C. Solberg, "Salmon fat content estimation by near infrared transmission spectroscopy" *J. of Food Science*, vol. 57(3), pp. 792-793, 1992.
15. T. Pearson, "Spectral properties and effect of drying temperature on almonds with concealed damage" *Lebensm.-Wiss. U.-Technol*, vol. 32, pp. 67-72, 1999.
16. T. Pearson, "Use of near infrared transmittance to automatically detect almonds with concealed damage" *Lebensm.-Wiss. U.-Technol*, vol. 32, pp. 73-78, 1999.
17. T. Pearson, D. Wicklow, E. Maghirang, F. Xie, and F. Dowell, "Detecting aflatoxin in single corn kernels by using transmittance and reflectance spectroscopy," submitted to *Lebensm. -Wiss. U. -Technol* .
18. A. Jain, R. Ruin, and J. Mao, "Statistical pattern recognition: a review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22(1), pp. 4-37, 2000.
19. E. Baum and D. Haussler, "What size net gives valid generalization?" *Neural Computation*, vol. 1, pp. 151-160, 1989.
20. M. J. Muasher and D. A. Landgrebe, "The K-L expansion as an effective feature ordering techniques for limited training sample size," *IEEE Trans. Geosci. Remote Sensing*, vol. *GE-21*, pp. *438-441*, Oct. 1983.
21. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Trans. on Patt. Anal. Machine Intell*., vol. *19*, no. *7*, pp. *711-720*, July, 1997.

22. K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *J. Optical Soc. Am. A*, vol. 14, pp. 1724-1733, 1997.
23. W. Hruschka, Data Analysis: wavelength selection methods. In: P. Williams and K. Norris, (Eds), *Near-infrared Tehnology in the Agricultural and Food Industries*, St. Paul, MN: Americal Association of Cereal Chemists, inc. 1987.
24. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press Inc. New York, 2nd Ed., 1992.
25. P. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," *IEEE Trans. Comput*., vol. 26, pp. 917-922, 1977.
26. T. Cover and J. Thomas, *Elements of Information Theory*, Wiley Interscience, New York, NY, 1991.
27. P. Williams and K. Norris, *Near-infrared Tehnology in the Agricultural and Food Industries*, St. Paul, MN: Americal Association of Cereal Chemists, inc. 1987.
28. K. Fukunaga and R. Hayes, "Estimation of classifier performance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11(10), pp. 1087-1101, 1989.
29. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, pp. 115-118, 1973.
30. G. Golub and C. Van Loan, *Matrix Computations*, the Johns Hopkins Press Ltd., London, 3rd Ed., pp. 80-82, 1996.
31. J. Doak, *An evaluation of feature selection methods and their application to computer security*, Technical report CSE-92-18, University of California, Davis, Department of Computer Science, 1992.
32. A. Jain and D. Zongker, "Feature selection: evaluation, application, and small sample performance," *IEEE Trans. Pattern Analysis and machine Intelligence*, vol. 19 (2), pp. 153-158, 1997.
33. T. Cover and J. Campenhout, "On the possible orderings in the measurement selection problem," *IEEE Trans. Systems, Man, and Cybernetics*, SMC-7(9), pp. 657-661, 1977.
34. S. Word, H. Martene, and H. Word, "The multivariate calibration problem in chemistry solved by the PLS method," in: *Proc. Conf. Matrix Pencils*. A. Ruhe, B. Kagstrom, eds. Lecture Notes in Mathematics. Springer Verlag, Heidelberg, pp. 286-293, 1983.
35. B. Yu and B. Yuan, "A more efficient branch and bound algorithm for feature selection," *Pattern Recognition*, vol. 26, no. 6, pp. 883-889, 1993.
36. D. Casasent and X. Chen, "New training strategies for RBF neural networks for X-ray agricultural product inspection," *Pattern Recognition*, vol. 36(2), pp. 535-547, 2003.
37. D. Casasent and X. Chen, "New data clustering for RBF classifier of agriculture products from X-ray images," *Proc. SPIE*, vol. 3837, pp. 232-241, 1999.
38. J. DeBonet and P. Viola, "Texture recognition using a non-parametric multi-scale statistical model," *1998 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 641-647, 1998.