Hyperspectral Band Selection for Multispectral Image Classification with Convolutional Networks

Giorgio Morales and John Sheppard
Gianforte School of Computing
Montana State University
Bozeman, MT 59717

Riley Logan and Joseph Shaw
Department of Electrical and Computer Engineering
Montana State University
Bozeman, MT 59717

***Abstract*—In recent years, Hyperspectral Imaging (HSI) has become a powerful source for reliable data in applications such as remote sensing, agriculture, and biomedicine. However, hyperspectral images are highly data-dense and often benefit from methods to reduce the number of spectral bands while retaining the most useful information for a specific application. We propose a novel band selection method to select a reduced set of wavelengths, obtained from an HSI system in the context of image classification. Our approach consists of two main steps: the first utilizes a filter-based approach to find relevant spectral bands based on a collinearity analysis between a band and its neighbors. This analysis helps to remove redundant bands and dramatically reduces the search space. The second step applies a wrapper-based approach to select bands from the reduced set based on their information entropy values, and trains a compact Convolutional Neural Network (CNN) to evaluate the performance of the current selection. We present classification results obtained from our method and compare them to other feature selection methods on two hyperspectral image datasets. Additionally, we use the original hyperspectral data cube to simulate the process of using actual filters in a multispectral imager. We show that our method produces more suitable results for a multispectral sensor design.**

I. Introduction

Optical remote sensing systems have a long history of collecting image data for many diverse applications, ranging from lab-based analysis of food quality and safety [1] to space-based contributions to archaeology [2]. The cornerstone of these systems is the exploitation of spatial and spectral information contained within the captured imagery. Though spatial information within an image can provide useful information, spectral data plays a central role in identifying and classifying objects in a scene. To address the need for rich spectral information, optical remote sensing systems come in many forms, ranging from simple multispectral imaging (MSI) systems [3], [4] to hyperspectral imagering (HSI) systems [1], [5]. In an MSI system, several distinct spectral bands are captured, often outside the visible spectrum. These systems are useful for capturing information in known areas of the spectrum. For example, if an application requires the detection of vegetation, a multispectral imager may only be required to capture the reflectance at $680\,\mathrm{nm}$ (red) and $800\,\mathrm{nm}$ (near-infrared), two commonly used spectral channels, to capture the chlorophyll content [6]. In contrast to MSI, HSI systems often capture hundreds of contiguous spectral bands.

Though powerful, the spectrally dense images captured during HSI come at the price of high data density, large file size, and increased computational complexity, which represent computational limitations when storing and processing these types of images. Returning to the example above, instead of capturing two distinct spectral bands to detect vegetation, a hyperspectral imager would capture data from hundreds of bands surrounding the wavelengths of interest. In such situations, the complexity introduced by the HSI system may be unnecessary if similar detection results can be achieved with fewer spectral channels. However, in many applications, relevant wavelengths are not known *a priori*. The ability to determine the most important wavelengths in a hyperspectral image would greatly simplify the data capture and processing requirements. Namely, it would enable using multispectral imagers in place of hyperspectral imagers, greatly reducing complexity and cost. Unfortunately, selecting salient wavelengths from an HSI system is not a trivial task.

In this paper, we propose a feature selection method to identify the most relevant spectral bands given an HSI classification problem. Our feature selection method consists of two steps: the first is a novel pre-selection method that we call inter-band redundancy analysis (IBRA). It assesses the degree of collinearity between each spectral band and its neighbors in order to approximate the minimum number of bands we need to move away from a band to find spectral bands with sufficiently distinct information. The distribution of this distance metric across the spectrum helps us to identify a reduced set of independent bands that act as the centroid of their corresponding regions in the spectrum. The second step is called greedy spectral selection (GSS) and consists of calculating the information entropy of each pre-selected band to rank its relevance. Then, we train a classifier using the top k pre-selected bands (where k is the number of desired bands) sorted according to their corresponding entropy. Finally, we remove from our selection the band that shows the most severe indication of multicollinearity and repeat the process taking into account the next available band to verify if the classification performance improves.

Having selected a reduced number of spectral bands from the original hyperspectral image, we train a new classification model based on a reduced-parameter convolutional neural network [7] that achieves high performance. We hypothesize

that it is possible to apply the combination of inter-band redundancy analysis and greedy spectral selection to select a small number of wavelengths (∼5–10) that will lead us to train more efficient HSI classifiers than the compared methods.

II. Related Work

Several dimensionality reduction techniques have been applied in the past as a natural pre-processing step for HSI classification problems. This is done to avoid unnecessarily high time complexity when processing large volumes of data and to reduce redundancy of the data, which could impair the performance of a classifier [8]. Dimensionality reduction techniques rely on feature extraction or feature selection approaches; the former apply linear or non-linear transformations to extract specific features from the original data, while the latter select the most useful subset of the features of the data without transforming them.

Among feature extraction methods, principal component analysis (PCA) and its variants (e.g. folded-PCA and kernel PCA) are some of the most commonly used methods to remove spectral redundancy and reduce the dimensionality of the raw data [9]. On the other hand, feature selection methods select a subset of spectral bands without modifying the data or projecting it into a new basis. The aim of this work is to identify which spectral wavelengths from the original hyperspectral spectrum are most responsive or relevant for a particular classification task without modifying the data, which is why we prefer feature selection methods over feature extraction methods. Additionally, identifying a reduced subset of relevant spectral bands allows for a better understanding of the optical properties of the materials and provides information that is useful when designing cheaper task-specific multispectral imagers. For example, if an application demands the identification of a certain type of plant, a feature selection method will identify the wavelengths that change most due to absorption from the unique pigmentation in the plant.

Given the advantages of feature selection, several methods have been proposed for hyperspectral image classification, one of the most common being ranking-based methods. These methods estimate the importance of each spectral band using such metrics as the variance inflation factor (VIF) [10] in order to select the top-ranked bands. We will use the idea of calculating the VIF value to measure collinearity, but the spectral bands will not be ranked based on this simple measure alone. Other methods use the estimated band relevance as part of an optimization approach, as proposed by Wang et al. [11], where the optimal clustering framework (OCF) separates the data into clusters, ranks them according to a selected measure (e.g. information entropy), and selects those with higher rank values. Furthermore, Wang et al. [12] also proposed a fast neighborhood grouping method for hyperspectral band selection (FNGBS) that partitions the data into several groups using Euclidean distance as a similarity measure, and then obtains the most relevant and informative bands using local density and information entropy measures.

In recent years, two feature selection approaches for HSI classification have gained more attention: partial least squares discriminant analysis (PLS-DA) [13] and genetic algorithms [14]. For instance, a recent method for bandwidth selection is known as Histogram Assisted Genetic Algorithm for Reduction in Dimensionality (HAGRID) [15]. This method maintains a population of index vectors identifying some specific number of wavelengths and fits a Gaussian mixture model to the converged population to identify the main wavelengths with their associated filter bandwidths.

Alternatively, some model-based approaches have been proposed in the context of deep learning. For instance, Taherkhani *et al.* [16] proposed regularizing the convolutional filters of the first layer of the convolutional neural network (CNN) using a group LASSO algorithm in order to sparsify the redundant spectral bands. Similar attempts, although not explicit feature selection methods, have been carried out in works such as [17], and [18], where a spectral-wise attention mechanism in the form of a fully-connected layer is applied to the inner convolutional layers of the network with the objective of emphasizing informative spectral features and suppressing less useful spectral features.

III. Materials and Methods

A. Datasets

We will use two datasets: an in-greenhouse controlled hyperspectral image dataset called "Kochia" and a well-known remote sensing HSI dataset called "Indian Pines" (IP).

The Kochia dataset consists of images of the weed kochia (*Bassia scoparia*) that were collected and analyzed by Scherrer *et al.* [19] with the aim of learning to differentiate between three different classes of herbicide-resistance of this weed: 1) herbicide-susceptible, 2) glyphosate-resistant, and 3) dicamba-resistant, where glyphosate and dicamba are two components commonly found in commercial herbicides. The images were captured using a Resonon Pika L hyperspectral imager with 300 spectral channels across a spectral range of 387.12 nm to 1,023.50 nm, resulting in a spectral resolution of approximately 2.12 nm. The kochia samples were illuminated using diffuse sunlight in a greenhouse setting. A total of 76 hyperspectral images of kochia with varying ages and spatial resolutions were captured at the Montana State University Southern Agricultural Research Center (SARC). Each image contains three kochia leaves of the same herbicide-resistance class with a height of 900 pixels and width ranging from 700–1,200 pixels.

The Indian Pines dataset [20] is an aerial 145×145 - pixel image of the Indian Pines site in Northwestern Indiana. It was acquired using the Airborne Visible / Infrared Imaging Spectrometer (AVIRIS) sensor [21] and it originally had 224 spectral bands in the wavelength range 380-2,510 nm, resulting in a spectral resolution of approximately 9.5 nm. The number of bands was reduced to 200 after removing 24 bands covering the region of water absorption. The data are divided into 16 classes containing agriculture, forest, and other natural perennial vegetation.

B. Data Pre-Processing

Images of kochia leaves were captured in raw digital numbers recorded by the Pika L hyperspectral imager, meaning the data cubes required pre-processing before they could be analyzed. For these experiments, data pre-processing was limited to reflectance correction for the Kochia dataset. To accomplish this, we converted the raw digital numbers to reflectance values using a 99% Spectralon panel as a reflectance reference. Specifically, the calculation of reflectance begins by selecting the pixels in the image which contain the Spectralon reflectance target. Each pixel in this region contains 300 digital numbers; one digital number for each of the captured spectral channels. We then take the average value of all pixels within the region at each spectral channel, leaving us with a single, averaged digital number for each spectral channel. This process leaves us with a digital number that represents 99% of the reflected light for each spectral channel. Finally, we calculate the spectral reflectance at each pixel as follows:

$$\rho = \left(\frac{DN_{scene} - DN_{dark}}{DN_{target} - DN_{dark}} \right) \rho_{target},$$

where ρ is the spectral reflectance, DN_{scene} represents the digital number values captured in the image, DN_{target} represents the averaged digital numbers of the reflectance target obtained through the process outlined above, DN_{dark} represents the dark current or background signal generated through sporadic electron generation in the imager's sensor, and ρ_{target} represents the reflectivity of the reflectance target.

We manually extracted 6,316 overlapping patches with a window size of 25×25 pixels from each of the 76 kochia images. Furthermore, we reduced the number of spectral bands within each patch from 300 to 150 by averaging adjacent pairs of bands, which can be interpreted as $2\times$ spectral binning, where the resulting spectral resolution of each channel has been modified from approximately $2.12\,\mathrm{nm}$ to $4.24\,\mathrm{nm}$. As one of the goals of this work is to aid in the design of multispectral imaging systems, and it is unlikely that optical filters with a bandwidth less than $20\,\mathrm{nm}$ will be used, decreasing the overall spectral resolution is unlikely to affect our results. Thus, this process itself gives us an upfront reduction in dimensionality that greatly reduces the potential overfitting impact in our following analysis.

Since the IP dataset consists of a single large image, we have to divide it into small patches so that each patch represents one class. Thus, we extracted square patches using a 5×5 pixel window around each pixel. Furthermore, we only collected patches around those pixels with an assigned label. By doing so, the new IP dataset has 10,249 patches.

Finally, we applied z-score normalization (mean equal to 0 and standard deviation equal to 1) onto each spectral band for both the Kochia and Indian Pines datasets.

C. Inter-Band Redundancy Analysis

The first step of our method is to reduce the inter-band redundancy by selecting a subset of representative spectral bands. We utilized a filter-based selection method whereby we iteratively calculate the Variance Inflation Factor (VIF) [22] between pairs of bands in order to determine how correlated they are; that is, to verify the presence of collinearity between them. We call this Inter-Band Redundancy Analysis (IBRA).

The VIF value between two bands, b_1 and b_2, is calculated based on the R-squared value from the Ordinary Least Square (OLS) regression model built by taking one of the bands as a dependant variable (b_1) and the other as the independent variable (b_2). Specifically, $VIF(b_1, b_2) = 1/(1 - R^2_{b_1,b_2})$. A high VIF value means that the independent variable and the dependent variable explain the same variance within the dataset and are redundant. We will consider VIF values greater than a threshold θ as representing the presence of collinearity in the model. In the literature, the recommended values of θ are between 5 and 10 [23], so we test different values of $\theta \in [5, 12]$ to observe how the performance is affected and to choose the best θ for a given classification task.

While some methods, such as that proposed by [10], use the VIF metric to identify and remove redundant spectral bands from a given set directly, our approach is novel and distinct in that we use it as part of a pre-selection step, assessing the collinearity degree between each band and its local neighbors iteratively in order to find independent salient bands. Thus, using the VIF metric, we calculate the number of bands $d_{left}(x)$ we need to move away to the left side from the x-th band in order to find bands sufficiently different from band x; similarly, we calculate the number of bands $d_{right}(x)$ we need to move away to the right side from the x-th band in order to find bands sufficiently different from that in band x (see Algorithm 1). In this algorithm, we calculate the difference $d(x) = |d_{left}(x) - d_{right}(x)|$ for each spectral band to determine how this difference is distributed across the spectrum. Let N be the total number of spectral bands within the dataset. Then `getVIF(`x, y`)` calculates the VIF value between bands at positions x and y. We construct `table`, which is a symmetric matrix that stores the pre-computed VIF values between pairs of bands in order to avoid recalculations. Then `getLocalMinima(`x`)` is a function that retrieves the position of the local minimum points of the vector x. We consider only local minimum points where $d(x) < 5$; otherwise, they will not be suitable bandwidth centers.

The distribution of the variable $d(x)$ is used to find clusters with similar bands and their corresponding cluster centers. Since we are interested in choosing suitable bandwidth centers, we choose bands that are similar to both the left and right sides; that is, the difference $d(x)$ has to be minimum. Fig. 1 shows the distribution of the variable $d(x)$ for the Kochia dataset given different thresholds θ. From this, we observe that each distribution consists of a series of "V" patterns. In this context, a local minimum—the center of a "V" pattern—represents a salient band that explains the variance within the dataset in a way similar to its neighbors. Even though all the bands within a "V" pattern are similar, the band at the leftmost position is more similar to those bands on the left side. Similarly, the band at the rightmost position of the

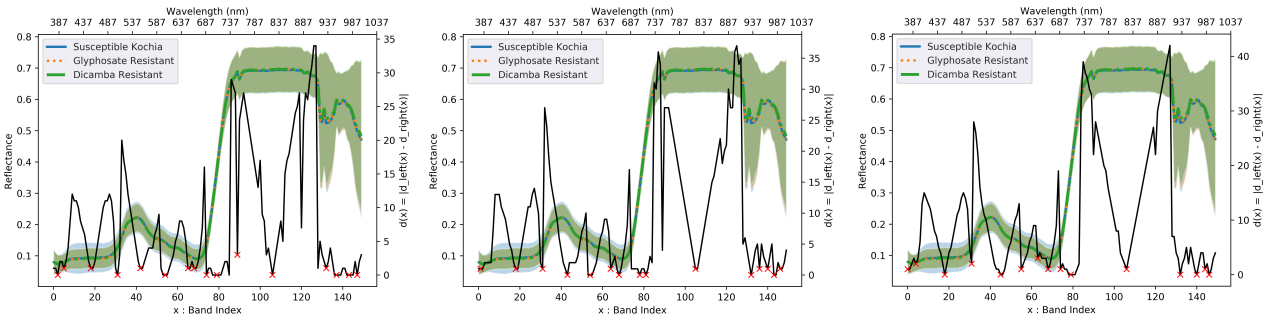

Fig. 1. Spectral response and Spectral index vs. distance d plots for the Kochia dataset using different VIF thresholds. **(a)** `th=12` **(b)** `th=10`. **(c)** `th=8`. Local minima in the three graphs are indicated with an '×'.

Algorithm 1 Calculating the interband redundancy

function INTERBANDREDUNDANCY(θ)
  $d_{left} \leftarrow []$
  $d_{right} \leftarrow []$
  $table \leftarrow zeros(N, N)$ // creates an $N \times N$ matrix of zeros
  for all $band \in (0, N)$ **do**
    // Check left side
    $t \leftarrow 1$
    $vif \leftarrow \infty$
    while $(vif > \theta) \wedge ((band - dt) > 0)$ **do**
      if $table[band, band - t] = 0$ **then**
        $table[band, band - t] = getVIF(band, band - t)$
        $table[band - t, band] = table[band, band - t]$
      $vif = table[band, band - t]$
      $t \leftarrow t + 1$
    $d_{left} \leftarrow [d_{left} - 1]$
    // Check right side
    $t \leftarrow 1$
    $vif \leftarrow \infty$
    while $(vif > \theta) \wedge ((band - dt) < N)$ **do**
      if $table[band, band + t] = 0$ **then**
        $table[band, band + t] = getVIF(band, band + t)$
        $table[band + t, band] = table[band, band + t]$
      $vif = table[band, band + t]$
      $t \leftarrow t + 1$
    $d_{right} \leftarrow [d_{right} - 1]$
  $d \leftarrow |d_{left} - d_{right}|$
  $preselection \leftarrow getLocalMinima(d)$
  return $d, preselection$

"V" is more similar to those bands on the right side, while the band corresponding to the local minimum is similar to both sides, acting as a centroid. In general, we keep the bands corresponding to a local minimum in the plot of spectral index vs. $d(x)$ and remove the rest since they are redundant.

D. Band Selection Using Pre-Selected Bands

In Sec. III-C we showed how to pick a set of independent candidate bands from the spectrum based on collinearity. Here, we employ a wrapper-based method to select the best combination of bands $S_f \in \mathbb{Z}^k$, given the desired number of bands k, from the set of available candidate bands $S_c \in \mathbb{Z}^{N'}$ (where $N' \ll N$ is the number of bands pre-selected by IBRA), which greatly reduces the search complexity. We call this process Greedy Spectral Selection (GSS).

The first step is to rank each band $s_c \in S_c$ according to some criterion. In this work, we use information entropy to calculate an initial relevance score for each spectral band. Given a band s_c, which is considered a discrete random variable with a bit depth of 14 bits, its entropy $H(s_c)$ conveys the average level of information inherent in s_c and is defined as follows:

$$H(s_c) = -\sum_{z \in \Omega_{s_c}} P(z) \log P(z),$$

where $P(z)$ is the probability mass function of random variable z, and Ω_{s_c} is the space that encompasses all the possible values that can occur in the spectral band s_c.

Other methods, such as those proposed by Wang et al. [11], [12], use information entropy to select the most representative band within each cluster. Our approach is different in the sense that we already picked the most relevant band from each cluster (defined as a local minimum in the plot of spectral index vs. distance $d(x)$); now, we rank the pre-selected bands based on information entropy in order to select a subset S_f of k bands. Thus, initially, S_f consists of the top-k bands of S_c with the greatest entropy.

Even if calculations show that collinearity does not exist between pairs of bands, it is possible that three or more bands are highly correlated—a phenomenon known as multicollinearity [23]. In that case, we check for the presence of multicollinearity within the k selected spectral bands the same way as described in Sec. III-C with collinearity. With this, we employ the following greedy algorithm for band selection (see Algorithm 2): First, we calculate the average classification performance (F1 score) when using the current k selected bands; to do this, we perform a 5×2 cross-validation and calculate the average F1 score obtained on the 10 validation folds. Then, after calculating the VIF value for each of the currently selected bands, we remove the one with the greatest VIF value and consider the next available band with the greatest entropy. With this new subset of k bands, we train a new classifier and verify if the average classification performance has improved. We repeat this process until there are no more available bands or when we find a significant drop in performance. Finally, we select the combination of bands that showed the best classification performance.

Note that Algorithm 2 selects k band indices from a list of candidate band indices S_c. Function `getEntropy(`S_f`)` returns the entropy value of each of the candidate bands in S_c.

Algorithm 2 Greedy spectral selection

```
function SELECTBANDS(S_c, k)
    H ← getEntropy(S_c)
    S_c.sort(key = H)
    S_f ← S_c[1 : k]
    S_c ← S_c[k + 1 : end]
    F1 ← trainSelection(S_f)
    bestS_f ← S_f
    while length(c) > 0 do
        listVIF ← getVIFMulti(S_f)
        index = getMax(listVIF)
        S_f[index : end] ← S_f[index + 1 : end]
        S_f.append(S_c[1])
        S_c ← S_c[2 : end]
        newF1 ← trainSelection(S_f)
        if newF > F1 then
            bestS_f ← S_f
            F1 ← newF1
        else
            if newF1 <= F1 − 0.05 then
                break // stop if a drop of 5% is found
    return bestS_f
```

Then, S_c`.sort(key=`H`)` decreasingly sorts the elements of S_c with respect to their corresponding entropy values. Function `getVIFMulti(`S_f`)` returns a list with the VIF value for each band in the list of selected bands S_f. Next `getMax(`l`)` returns the position where the maximum value in a list l was found. Finally, `trainSelection(`S_f`)` returns the average F1 score evaluated using the bands in S_f and 5×2 cross-validation.

E. Convolutional Neural Network Architecture

For all of our experiments, we used a modified version of the Hyper3DNet network [7], which is a 3D-2D CNN architecture specifically designed to solve HSI classification problems using a reduced number of trainable parameters. Furthermore, experimental results demonstrated relative superiority of this architecture over state-of-the-art architectures.

In this paper, our modified network, referred to as Hyper3DNet-Lite, is a simplification of the original Hyper3DNet architecture. The difference with the original architecture is that its 3-D feature extractor consists of two 3-D convolutional layers instead of a densely connected block with four layers; additionally, its 2-D spatial encoder has three layers instead of four. The Hyper3DNet-Lite architecture used for the Kochia dataset is detailed in Table I, where N denotes the number of spectral bands in the input, "SepConv2D" denotes a 2-D separable convolutional layer, and "ReLU" denotes a rectified linear unit activation layer (where $ReLU(x) = \max(0, x)$). The only difference with the network used to process the IP dataset is that, since the input image is smaller (5×5 pixels), the stride used in the last two "SepConv2D" layers is $(1, 1)$ instead of $(2, 2)$ to avoid dimensionality inconsistencies.

The simplified architecture of the Hyper3DNet-Lite network becomes especially suitable for datasets that use just a few spectral bands, given that these datasets do not require models with a high level of complexity to process them, unlike those datasets that use all the available spectral bands. In this way, we avoid overparameterization, which results in our models being less prone to overfitting.

TABLE I
HYPER3DNET-LITE ARCHITECTURE FOR THE KOCHIA DATASET.

Layer Name	Kernel Size	Stride Size	Output Size
Input	—	—	(25, 25, N, 1)
Conv3D + ReLU	(3, 3, 3)	(1, 1, 1)	(25, 25, N, 16)
Conv3D + ReLU	(3, 3, 3)	(1, 1, 1)	(25, 25, N, 16)
Reshape	—	—	(25, 25, 16N)
SepConv2D + ReLU	(3, 3)	(1, 1)	(25, 25, 320)
SepConv2D + ReLU	(3, 3)	(2, 2)	(13, 13, 256)
SepConv2D + ReLU	(3, 3)	(2, 2)	(7, 7, 256)
GlobalAveragePooling	—	—	256
Dense + Softmax	—	—	# classes

Previously, we also experimented with other types of classifiers (i.e. support vector machines, random forests, and feedforward neural networks) to use in the GSS process. However, due to the fast convergence rates and the substantial improvements on performance, we continued to use CNNs over the other types of classifiers.

F. Multispectral Sensor Design

The previous steps are used to select the most relevant spectral bands from the original hyperspectral data cube. However, knowing which wavelengths are the most useful for a given application allows for the design of compact multispectral sensors instead of using a full hyperspectral sensor. To accomplish this, we use the original data cube and simulate applying optical filters to capture data from a multispectral imager.

To do this, we generate k Gaussian distributions, taking the position of the spectral bands selected by the GSS method as the centroids. The bandwidth of these distributions is set to five bands or, equivalently, $20\,\mathrm{nm}$, to represent a common optical filter bandwidth. The simulated multispectral reflectance measurement is obtained by multiplying the original hyperspectral data cube by the corresponding Gaussian distribution generated for each band, then integrating under the resulting response curve to get a single reflectance value. This process is repeated for each of the k Gaussian distributions.

IV. EXPERIMENTAL RESULTS

For the sake of consistency and fairness, we used the same configuration (i.e., network architecture, optimizer, and batch size) for all the networks trained in our experiments. While this strategy does not guarantee the best possible results, it allows us to compare the behavior of different band selection methods under the same conditions. All CNNs were trained using the Adadelta optimizer [24], which is a gradient descent method based on an adaptive learning rate, so that there is no need to select a global learning rate manually. The mini-batch size was set to 128. The last layer of the CNNs used a softmax activation function, and we employed a categorical cross-entropy loss function. Furthermore, we used 5×2-fold stratified cross-validation to train and evaluate all networks. Note that z-score normalization was applied to each training set while the exact same scaling was applied to their corresponding validation set. In order to analyze the behavior of our models, we calculated

TABLE II
PERFORMANCE WITH AND WITHOUT IBRA PRESELECTION ($\theta = 10$).

Dataset	# Bands	OA	Prec	Rec	F1	# Param.
Kochia	150	98.46 ± 0.29	98.66 ± 0.26	98.55 ± 0.31	98.60 ± 0.28	561,475
	17	97.05 ± 0.47	97.25 ± 0.45	97.17 ± 0.46	97.21 ± 0.44	258,035
Indian Pines	200	99.42 ± 0.18	99.32 ± 0.29	99.47 ± 0.28	99.39 ± 0.27	1,274,464
	31	99.49 ± 0.14	99.38 ± 0.34	99.56 ± 0.19	99.47 ± 0.23	338,880

four metrics on the validation sets: accuracy (OA), macro-average precision ($Prec$), macro-average recall (Rec), and F1 score. The source code and datasets are available online[1].

In the following sections, we compare the results of using our inter-band redundancy analysis strategy alone and our greedy spectral selection strategy after pre-selection. We also compare our results with several state-of-the-art methods for bandwidth selection. For all our experiments, we select a reduced number of spectral bands k, as our objective is to design simple task-specific multispectral sensors.

A. Training Pre-Selected Bands

Previously (Fig. 1) we showed some examples of applying the pre-selection method using IBRA on the Kochia dataset using three different VIF thresholds (12, 10, and 8), which reduced our search space from 150 bands to 19, 17, and 16 bands, respectively. Table II gives the number of pre-selected bands for both the Kochia and IP datasets when using a VIF threshold of 10; it also gives the average performance for the four metrics and corresponding standard deviations using the Hyper3DNet-Lite network when training on the full hyperspectral spectrum and only the pre-selected bands. The number of parameters required to train each network is reported in the last column.

B. Greedy Spectral Selection

Next, we applied the GSS method for each of the sets of IBRA-selected bands using different VIF thresholds $\theta \in [5, 12]$. Then, we selected the classifier that achieved the best classification performance based on the mean F1-score obtained after a 5×2-fold cross-validation.

For the Kochia dataset, we considered six and ten bands in order to evaluate the trade-off between the number of bands and performance. For the IP dataset, we selected only five bands. In addition, for each dataset, we experimented with different dataset sizes (i.e. 100%, 75%, 50%, and 25%) to evaluate how consistent the band selection results are under different data set sizes.

For the Kochia dataset, when selecting six bands, the best results were obtained using a VIF threshold of ten ($\theta = 10$) and the wavelengths of the selected bands (in nm) were $[391.2, 463.3, 569.3, 675.3, 730.4, 993.3]$ for each of the four dataset size variations. When selecting ten bands, the best results were obtained using a VIF threshold of $\theta = 7$ when using 100% and 50% of the dataset, and $\theta = 8$ when using 75% and 25% of the dataset. The wavelengths of the bands selected for $\theta = 7$ were $[387.0, 404.0, 463.3, 518.4, 569.3, 654.1, 675.3, 700.8, 722.0, 1006.0]$ and the only difference with respect to the bands selected for $\theta = 8$ was the selection of the wavelength $577.8\,\mathrm{nm}$ instead of $569.3\,\mathrm{nm}$. Table III shows the performance using IBRA and GSS on the full Kochia dataset, where the bold entries represent the best VIF threshold, band selection, and average F1 score.

[1]Codebase: https://github.com/GiorgioMorales/HSI-BandSelection.git.

TABLE III
GREEDY SPECTRAL SELECTION ON THE KOCHIA DATASET.

k	VIF	Selected bands (nm)	OA	Prec	Rec	F1
6	12	[395.5, 463.3, 565.1, 700.8, 722.0 , 993.3]	92.44 ± 0.71	92.76 ± 0.80	92.79 ± 0.67	92.76 ± 0.72
	11	[395.5, 408.2, 463.3, 586.3, 662.6, 700.8]	90.74 ± 1.05	91.56 ± 0.97	91.54 ± 1.06	91.54 ± 1.01
	10	**[391.2, 463.3, 569.3, 675.3, 730.4, 993.3]**	92.69 ± 0.53	93.24 ± 0.52	93.08 ± 0.49	**93.15 ± 0.49**
	9	[391.2, 463.3, 569.3, 700.8, 730.4, 993.3]	92.40 ± 0.63	92.67 ± 0.63	92.77 ± 0.59	92.71 ± 0.59
	8	[387.0 , 404.0 , 463.3, 577.8, 700.8, 722.0]	92.58 ± 0.63	93.05 ± 0.65	93.08 ± 0.57	93.06 ± 0.59
	7	[387.0, 404.0, 463.3, 569.3, 700.8, 722.0]	92.07 ± 0.89	92.52 ± 0.89	92.55 ± 0.79	92.53 ± 0.83
	6	[387.0 , 404.0 , 463.3, 586.3, 700.8, 717.7]	92.00 ± 0.61	92.57 ± 0.54	92.52 ± 0.64	92.53 ± 0.57
	5	[387.0 , 463.3, 586.3, 645.6, 700.8, 722.0]	91.03 ± 1.04	91.79 ± 1.14	91.75 ± 0.91	91.76 ± 1.01
10	12	[395.5, 408.2 , 463.3, 518.4, 565.1, 616.0, 675.3, 700.8, 722.0, 993.3]	96.31 ± 0.69	96.57 ± 0.55	96.49 ± 0.73	96.53 ± 0.64
	11	[395.5, 408.2, 463.3, 565.1, 662.6, 675.3, 700.8, 713.5, 726.2, 993.3]	96.18 ± 0.41	96.48 ± 0.29	96.31 ± 0.46	96.39 ± 0.36
	10	[391.2, 463.3, 518.4, 569.3, 658.4, 675.3, 717.7, 730.4, 993.3, 1006.]	95.83 ± 0.36	96.10 ± 0.38	96.06 ± 0.32	96.08 ± 0.34
	9	[391.2, 463.3, 518.4, 569.3, 616.0, 671.1, 700.8, 717.7, 730.4, 993.3]	96.16 ± 0.56	96.48 ± 0.50	96.37 ± 0.54	96.42 ± 0.52
	8	[387.0, 404.0, 463.3, 518.4, 577.8, 654.1, 675.3, 700.8, 722.0, 1006.0]	96.47 ± 0.38	96.79 ± 0.36	96.66 ± 0.37	96.72 ± 0.35
	7	**[387.0, 404.0, 463.3, 518.4, 569.3, 654.1, 675.3, 700.8, 722.0, 1006.0]**	96.69 ± 0.35	96.92 ± 0.38	96.95 ± 0.34	**96.93 ± 0.35**
	6	[387.0, 404.0, 463.3, 586.3, 649.9, 679.6, 700.8, 717.7, 730.4, 1001.8]	95.91 ± 0.50	96.34 ± 0.44	96.12 ± 0.47	96.22 ± 0.45
	5	[387.0, 463.3, 586.3, 645.6, 700.8, 722.0, 832.2, 946.7, 980.6, 1001.8]	95.06 ± 0.54	95.44 ± 0.52	95.33 ± 0.56	95.38 ± 0.53

TABLE IV
GREEDY SPECTRAL SELECTION ON THE INDIAN PINES DATASET.

VIF	Selected bands (nm)	OA	Prec	Rec	F1
12	[484.6, 627.2, 703.3, 750.8, 1017.1]	97.96 ± 0.33	98.21 ± 0.43	98.32 ± 0.33	98.25 ± 0.35
11	[541.7, 570.2, 703.3, 750.8, 1017.1]	97.55 ± 0.29	98.05 ± 0.29	97.95 ± 0.29	97.98 ± 0.22
10	**[484.6, 617.7, 703.3, 750.8, 1017.1]**	98.08 ± 0.43	98.26 ± 0.42	98.39 ± 0.43	**98.32 ± 0.39**
9	[541.7, 617.7, 703.3, 817.4, 1017.1]	98.28 ± 0.35	98.24 ± 0.47	98.06 ± 0.59	98.11 ± 0.43
8	[589.2, 627.2, 703.3, 817.4, 1017.1]	98.04 ± 0.30	98.19 ± 0.46	98.06 ± 0.46	98.10 ± 0.35
7	[551.2, 570.2, 703.3, 817.4, 1017.1]	98.01 ± 0.24	98.29 ± 0.18	98.36 ± 0.42	98.31 ± 0.26
6	[560.7, 703.3, 817.4, 912.5, 1017.1]	97.06 ± 0.49	97.49 ± 0.58	97.63 ± 0.48	97.53 ± 0.48
5	[560.7, 712.8, 807.9, 912.5, 1017.1]	96.73 ± 0.55	97.46 ± 0.53	97.00 ± 0.61	97.19 ± 0.49

For the IP dataset, the best results were obtained using a VIF threshold of ten ($\theta = 10$) and the wavelength of the selected bands were $[484.6, 617.7, 703.3, 817.4, 1017.1]$ for all the four dataset size variations. Table IV shows the performance using IBRA and GSS on the full IP dataset.

C. Comparative Results

Finally, we compared our IBRA-GSS method to three other methods: OCF [11], HAGRID [15], and PLS-DA [25]. For OCF, we used the normalized cut objective function along with information entropy ranking, as they showed the best performance. For HAGRID, we used a grid search to choose the following hyperparameters: a crossover rate of 0.25, a mutation rate of 0.05, a tournament size of 5, a population size of 1,000, and 300 iterations. To analyze the effectiveness of the feature selection methods, we compared the performance of four CNNs, each trained on the features selected by the four methods. This comparison was carried out using the same network architecture, hyperparameters, and other configurations for all of the methods. Finally, to determine if the difference in performance scores was statistically significant, we performed a paired t-test using the F1 scores at the $\alpha = 0.05$ level.

The method comparison is shown in Table V for the Kochia dataset and in Table VI for the IP dataset, with the best performing metrics highlighted in bold. Here, the first row of each method represents the results obtained after training a model using the original selected bands (identified as "original band selection"), while the second row represents the results obtained after using simulated filters that take the position of the selected bands as their central wavelengths (identified as "multispectral filter simulation"). The simulated filters used for the Kochia dataset were $20\,\mathrm{nm}$ while for the IP dataset were $50\,\mathrm{nm}$. According to the t-test, our method performed significantly better than the other four methods in each of the cases. Although not shown due to space limitations, we also tested the four dataset size variations, as explained previously, and found that the improvement in performance of our method over the others was still statistically significant even when the dataset size was reduced. Additional experiments with other values of k showed that the improvements of GSS over the compared methods remained consistent.

TABLE V
FEATURE SELECTION METHOD COMPARISON — KOCHIA.

Bands	**6**				**10**			
Method	**OA**	**Prec**	**Rec**	**F1**	**OA**	**Prec**	**Rec**	**F1**
FNGBS	84.32 ± 1.78	84.85 ± 1.77	84.37 ± 1.72	84.59 ± 1.73	93.78 ± 0.77	94.17 ± 0.84	93.99 ± 0.73	94.08 ± 0.78
	86.98 ± 0.84	87.35 ± 0.80	86.91 ± 0.91	87.10 ± 0.83	94.19 ± 0.47	94.54 ± 0.47	94.27 ± 0.51	94.39 ± 0.48
PLS-DA	84.77 ± 1.83	85.15 ± 1.89	84.69 ± 1.82	84.89 ± 1.82	94.36 ± 0.51	94.86 ± 0.55	94.67 ± 0.47	94.76 ± 0.49
	88.41 ± 0.79	88.85 ± 0.62	88.37 ± 0.96	88.59 ± 0.78	95.10 ± 0.68	95.44 ± 0.59	95.18 ± 0.67	95.30 ± 0.63
OCF	90.48 ± 0.57	90.92 ± 0.62	90.81 ± 0.44	90.86 ± 0.49	94.87 ± 0.51	95.23 ± 0.52	95.11 ± 0.46	95.16 ± 0.47
	92.42 ± 0.67	92.75 ± 0.66	92.66 ± 0.66	92.70 ± 0.65	94.62 ± 0.73	95.00 ± 0.65	94.80 ± 0.64	94.89 ± 0.64
HAGRID	91.71 ± 0.83	92.25 ± 0.78	92.17 ± 0.84	92.20 ± 0.80	94.50 ± 0.81	94.81 ± 0.78	94.69 ± 0.72	94.74 ± 0.74
	92.48 ± 0.62	92.91 ± 0.53	92.89 ± 0.58	92.89 ± 0.54	95.14 ± 0.51	95.49 ± 0.48	95.18 ± 0.51	95.33 ± 0.47
GSS	**92.69 ± 0.53**	**93.24 ± 0.52**	**93.08 ± 0.50**	**93.15 ± 0.49**	**96.69 ± 0.35**	**96.92 ± 0.38**	**96.95 ± 0.34**	**96.93 ± 0.35**
	93.32 ± 0.68	**93.80 ± 0.64**	**93.74 ± 0.66**	**93.76 ± 0.64**	**96.21 ± 0.49**	**96.51 ± 0.45**	**96.40 ± 0.44**	**96.45 ± 0.44**

TABLE VI
FEATURE SELECTION METHOD COMPARISON — INDIAN PINES.

Method	**5 bands**			
	OA	**Prec**	**Rec**	**F1**
PLS-DA	96.68 ± 0.86	96.83 ± 0.99	95.62 ± 0.94	96.11 ± 0.74
	97.17 ± 0.60	97.30 ± 0.79	96.66 ± 1.03	96.90 ± 0.84
OCF	96.68 ± 0.56	97.34 ± 0.76	96.34 ± 0.98	96.77 ± 0.82
	97.02 ± 0.58	97.73 ± 0.51	97.14 ± 0.63	97.39 ± 0.48
HAGRID	96.74 ± 0.54	97.06 ± 0.75	96.34 ± 1.03	96.65 ± 0.88
	97.03 ± 0.75	97.24 ± 0.86	96.72 ± 1.45	96.91 ± 1.21
FNGBS	97.49 ± 0.34	97.86 ± 0.36	97.64 ± 0.71	97.72 ± 0.5
	97.34 ± 0.65	97.94 ± 0.52	97.75 ± 0.46	97.82 ± 0.44
GSS	**98.08 ± 0.43**	**98.26 ± 0.42**	**98.39 ± 0.43**	**98.32 ± 0.39**
	98.24 ± 0.39	**98.56 ± 0.38**	**98.43 ± 0.42**	**98.48 ± 0.36**

V. DISCUSSION

Using our IBRA method, we identified sets of influencing spectral bands for both datasets. These pre-selected bands explain the variance of their neighbors in the original spectrum with a VIF value greater than a threshold $\theta \in [5, 12]$; therefore, keeping them and removing the other bands allowed us to avoid spectral bands that did not contain useful information for performing classification. That is, our method effectively identifies those spectral bands that carry information for performing classification while discarding redundant spectral bands. Results shown in Table II demonstrate that it is possible for a model trained on the subset of spectral bands determined by IBRA, to achieve high accuracy values ($\sim$ 97–99%) similar to those obtained when using the full spectrum.

Our GSS method uses information entropy to identify which bands are more relevant among the pre-selected bands. However, if we need to select at most k bands, the subset of bands with the greatest saliency values may not be the best selection. For instance, for the Kochia dataset, if $k = 6$, the wavelengths of the bands with the highest entropy values are $[391.2, 463.3, 518.4, 616.0, 658.4, 675.3]$; however, line **??** in Algorithm 2 detects strong multicollinearity between wavelengths $616.0, 658.4$, and 675.3. Rather than using redundant bands, we select a more diverse subset if this helps to improve the classification performance.

From Tables V and VI, we see that our method achieved the highest performance on both datasets, which confirms our hypothesis. The results remain consistent even when considering different dataset sizes. In addition, Table V shows that there is a more noticeable gap in performance between our method and the others when selecting ten bands, rather than when selecting six bands. This confirms that the fewer spectral bands we select, the harder the task will be; however, multispectral imagers generally become more practical computationally as the number of spectral channels becomes smaller.

Finally, it is worth noting that, with our method, the classification performance resulting from the "original band

selection" approach is very similar to the performance with the "multispectral filter simulation" approach, unlike some of the compared methods. This can be explained by the way we selected the first band candidates using IBRA. That is, a spectral band corresponding to a local minimum in the plot of spectral index vs. distance $d(x)$ (Fig.1) acts as a centroid because it is similar to the spectral bands located on either side. Therefore, if we take this local minimum as the central wavelength of a multispectral filter, generate a Gaussian distribution around it by considering a standard bandwidth, and integrate under the curve, then we obtain reflectance values similar to those of the central band. Given that the simulated multispectral filter and the original spectral band present similar information, their classification performance is similar. This is convenient for a multispectral sensor design, as we would like the central wavelength of a filter to be the most representative.

VI. Conclusion

Data captured by an imaging system is often processed to make observations and classifications about the world around us. The spatial and spectral content of the images obtained is key to analyzing the data, with spectral content playing a central role. However, the dense spectral information collected by a hyperspectral imager is not required for every application, and managing such data can be computationally expensive. The ability to determine the most relevant wavelengths for a given application enables using simpler multispectral imagers in place of hyperspectral imagers. This simplification would not only lead to economic savings, as fewer specialized storage and processing devices are required but also clarity and time-savings when analyzing data.

To allow for this simplification, we presented a method for selecting salient wavelengths from a hyperspectral data cube based on two main steps: A pre-selection step that identifies a subset of independent spectral bands (IBRA) and a final greedy selection step based on information entropy (GSS). Experimental results showed our band selection method generally outperformed other commonly-employed feature selection methods on the Kochia and Indian Pines datasets.

Finally, we showed that the inter-band redundancy method does not only reduce the search space considerably, but it also provides potential filter centers that are suitable for the design of multispectral sensors. Hence, another outcome of this work is the aid in the design of compact multispectral imagers that will assist in applications such as automatically identifying herbicide-resistance biotypes of the weed kochia.

In the future, we plan to explore incorporating an attention mechanism into our CNN architecture. When combined with the band selection method proposed here, we expect further computational savings by enabling a more adaptive approach to using the selected bands. We also plan to explore implementation issues associated with developing multispectral sensors based on the designs recommended through these methods.

References

[1] D. Wu and D.-W. Sun, "Advanced applications of hyperspectral imaging technology for food quality and safety analysis and assessment: A review — Part I: Fundamentals," *Innovative Food Science and Emerging Technologies*, vol. 19, 2013.

[2] M. J. Giardino, "A history of NASA remote sensing contributions to archaeology," *Journal of Archaeological Science*, vol. 38, no. 9, pp. 2003–2009, 2011.

[3] S. A. Mathews, "Design and fabrication of a low-cost, multispectral imaging system," *Appl. Opt.*, vol. 47, no. 28, pp. F71–F76, Oct 2008.

[4] C. Yang, "A high-resolution airborne four-camera imaging system for agricultural remote sensing," *Computers and Electronics in Agriculture*, vol. 88, pp. 13 – 24, 2012.

[5] A. F. Goetz, "Three decades of hyperspectral remote sensing of the earth: A personal view," *Remote Sensing of Environment*, vol. 113, pp. S5 – S16, 2009.

[6] J. Xue and B. Su, "Significant Remote Sensing Vegetation Indices: A Review of Developments and Applications," *Journal of Sensors*, 2017.

[7] G. Morales, J. Sheppard, B. Scherrer, and J. Shaw, "Reduced-cost hyperspectral convolutional neural networks," *J. of Applied Remote Sensing*, vol. 14, no. 3, p. 036519, 2020.

[8] W. Sun and Q. Du, "Hyperspectral band selection: A review," *IEEE Geoscience and Remote Sensing Magazine*, vol. 7, no. 2, pp. 118–139, 2019.

[9] M. P. Uddin, M. A. Mamun, and M. A. Hossain, "Feature extraction for hyperspectral image classification," in *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, 2017, pp. 379–382.

[10] F. Castaldi, A. Castrignanò, and R. Casa, "A data fusion and spatial data analysis approach for the estimation of wheat grain nitrogen uptake from satellite data," *International Journal of Remote Sensing*, vol. 37, no. 18, pp. 4317–4336, 2016.

[11] Q. Wang, F. Zhang, and X. Li, "Optimal clustering framework for hyperspectral band selection," *IEEE Trans Geosci and Remote Sens*, vol. 56, no. 10, pp. 5910–5922, 2018.

[12] Q. Wang, Q. Li, and X. Li, "A fast neighborhood grouping method for hyperspectral band selection," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–12, 2020.

[13] L. Feng, S. Zhu, F. Liu, Y. He, Y. Bao, and C. Zhang, "Hyperspectral imaging for seed quality and safety inspection: a review," *Plant Methods*, vol. 15, no. 91, pp. 2631–2641, 2019.

[14] S. Li, Z. Xia, and C. Zhang, "Sensitive wavelengths selection in identification of *Ophiopogon japonicus* based on near-infrared hyperspectral imaging technology," *International Journal of Analytical Chemistry*, vol. 2017, 2017.

[15] N. Walton, J. Sheppard, and J. Shaw, "Using a genetic algorithm with histogram-based feature selection in hyperspectral image classification," in *Proc. ACM Genetic and Evolutionary Computation Conference*, ser. GECCO '19, 2019, p. 1364–1372.

[16] F. Taherkhani, J. Dawson, and N. M. Nasrabadi, "Deep sparse band selection for hyperspectral face recognition," 2019, arXiv:1908.09630.

[17] B. Fang, Y. Li, H. Zhang, and J. Chan, "Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism," *Remote Sensing*, vol. 11, no. 2, p. 159, 2019.

[18] X. Gao, Y. Zhao, L. Dudziak, R. Mullins, and X. Cheng-Zhong, "Dynamic channel pruning: Feature boosting and suppression," in *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, 2019.

[19] B. Scherrer, J. Sheppard, P. Jha, and J. A. Shaw, "Hyperspectral imaging and neural networks to classify herbicide-resistant weeds," *Journal of Applied Remote Sensing*, vol. 13, no. 4, pp. 1 – 15, 2019.

[20] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, "220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3," Sep 2015.

[21] W. M. Porter and H. T. Enmark, "A System Overview Of The Airborne Visible/Infrared Imaging Spectrometer (Aviris)," in *Imaging Spectroscopy II*, G. Vane, Ed., vol. 0834, International Society for Optics and Photonics. SPIE, 1987, pp. 22 – 31.

[22] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. New York: Springer Science+Business Media, 2013.

[23] D. Belsley, E. Kuh, R. Welsch, and R. Wells, *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*, ser. Wiley Series in Probability and Statistics - Applied Probability and Statistics Section Series. Wiley, 1980.

[24] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, arXiv:1212.5701.

[25] M. Marker and W. Rayens, "Partial least squares for discrimination," *Journal of Chemometrics*, vol. 17, pp. 166–173, 2003.