IMAGE CLASSIFICATION

Extraction of information from remotely-sensed data has traditionally been performed through human interpretation of aerial imagery. With the development of remote sensing and computer technology, these traditional techniques have shifted from analog to digital image analysis and interpretation. Image classification is an important tool to extract thematic information from remotely sensed data of the earth surface. Recently, computer digital image processing has taken a large amount of human intervention to extract and map thematic information from remotely sensed data. Hence, digital image classification has become an inseparable component of remote sensing.

As the name indicates, *remote sensing* is the acquisition of information from a distance. It is the art and science of acquiring information about the physical and biological characteristics of an object or a phenomenon without being in direct contact with the object or phenomenon. Remotely sensed data may come from satellite images, aerial photography, radar images, or ground measurements.

Because remote-sensing technology and techniques are growing rapidly, remote sensing is becoming one of the most powerful and flexible tools for environmental management and economic development. Currently, remotely-sensed data are the primary source of information for global change and environmental a studies.

Remote sensing is a national and international priority. Many countries use remote sensing products in their daily planning, management and decision making. The continuously growing use of remotely-sensed data has empowered tremendous development and improvement in this technology. Many countries have currently their own earth observation satellites. The most internationally known satellites are the Landsat program of the National Aeronautics and Space Administration (NASA), the National Oceanographic and Atmospheric Administration (NOAA) satellite program, the ERS satellite program of the European Space Agency, the Satellite Pour l'Observation de la Terre (SPOT) satellite program of the Centre National d'Etude Spatiales, France, the RADARSAT satellite program of the Canadian Center for Remote Sensing, Canada, and satellites from other programs such as those in Russian, Japan, and Indian (1).

Remote sensing technology is a relatively new concept and, scientists are still exploring the methods and concepts to better exploit this technology. The key components of remote sensing include the physical properties of the ground features to be *imaged*, the energy that interacts with these features, and the sensor that records the energy coming from these features. The energy is known as the *electromagnetic energy*, which is composed of *wavelengths* that travels with an electric field and a magnetic field. The electromagnetic energy is measured by its wavelength or by its *frequency*. The wavelengths are ranked by their length and are represented by intervals, called *spectral bands*, on what is referred to as the *electromagnetic spectrum* (Fig. 1). The electromagnetic spectrum ranges from very short wavelengths (i.e., cosmic rays) to very long wavelengths (i.e., audio wavelengths).

The visible portion of the spectrum is the energy that humans can see with the naked eye. This portion is very small compared to the entire spectrum over which remote sensing instruments operate. This is a clear advantage of this technology as it allows us to discriminate among objects over a large spectrum over which the human eye is incapable of *seeing*. Because the physical properties of the ground features are wavelength dependent, features that look the same (i.e. respond similarly) in one region of the electromagnetic spectrum (e.g., the visible) may look completely different in another portion of the spectrum (e.g., the infrared). Therefore, the wide range of the electromagnetic spectrum allows us to detect and analyze objects in a variety of wavelengths, which increases the possibilities of their identification.

Essentially, remote sensors record the energy coming from objects on the ground. This energy may be either reflected, emitted, or transmitted from these objects in separate wavelengths, called *spectral bands* or simply *bands*. The information may be interpreted on individual bands or a combination of bands. The combination of bands (*multispectral images*) is a technique that allows the interpreter to examine a feature in multiple dimensions to increase the accuracy of extracting information. Remote-sensing data may be in analogue format (e.g., photographs) or in a digital image format. Digital image interpretation is usually performed with computers and the process is called *image classification*. The process of digital image classification usually involves a combination of *hardware* (i.e., computers) and *software* (i.e., programs) with special *algorithms* known as *classifiers*. A *pixel* (picture element) in a digital image is the spectral measurement of the corresponding area on the ground. A pixel can then be defined as a point in n-dimensional feature (spectral) space. Hence, a pixel is an n-dimensional pattern vector. The size of an area forming a pixel is determined by the sensors and is called the *spatial image resolution*. The value of each pixel is called the *spectral signature* (gray level) and is quantized during the sensing process. In analog, a digital image is similar to a matrix: a pair of coordinates and a value associated with each matrix element.

As new sensor technology has emerged over the past few years, high dimensional hyperspectral data with hundreds of bands have become available. These bands are usually measured at a series of narrow and contiguous wavelength. For example, the AVIRIS system gathers image data in 210 spectral bands in the range of 0.4-2.4 μm. Compared to the lower dimensionality images (less than 20 bands), this *hyperspectral* data potentially provides a wealth of information for identifying spectrally unique materials. The land use/cover, detection of objections, urban fringe expansion and, mudflows and landslides monitoring are some widely applications for hyperspectral data. However, it also raises the need for more specific attention to the data analysis procedure if this potential is to be fully realized (42).

Image classification is a process that clusters pixels of similar spectral response in an image into separate categories (i.e. themes) representing ground cover types (2). Thematic information can then be extracted in multiple spectral bands. *Spectral classes* are characteristics recorded in the remotely sensed data in which the pixels in

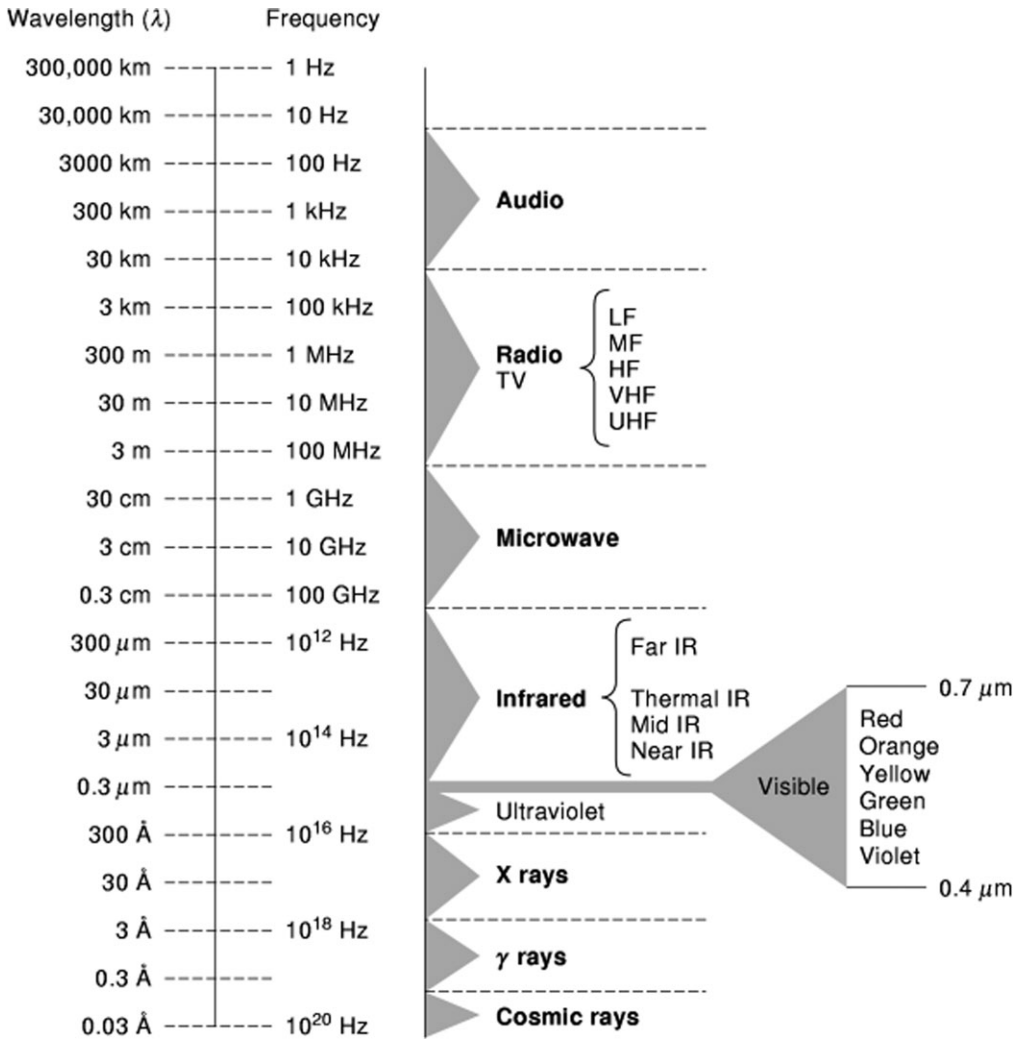

Figure 1. Schematic diagram of the electromagnetic spectrum. The visible portion is very small compared to the entire spectrum used by the remote-sensing systems, which increase our ability to discriminate among ground features that respond differently at different wavelengths.

the same category have the same or similar spectral signatures and represent the information of ground cover types. The output from a multispectral image classification system is a thematic map in which each pixel in the original imagery is classified into one of several spectral classes. Image classification may also be thought of as a labeling problem, providing each pixel with a class label.

The objective of digital image classification is to categorize automatically all the image pixels into themes that correspond to the ground cover. This is usually done through the use of multispectral bands that allow for each pixel to be analyzed in multiple dimensions for better discrimination among different features represented by these pixels. Each pixel may be identified using three recognition forms (2).

1. *Spectral* Pixels are recognized from their brightness values (also called digital values, digital numbers, spectral signatures, or gray levels).
2. *Spatial* Pixels are identified as a function of the surrounding pixels.
3. *Temporal* Certain features are easier identified in one season than in another. For example, it is difficult to discriminate between conifers and deciduous in the summer because both species have green leaves (i.e., similar spectral response), but these two species can easily be separated in the winter because deciduous trees would have lost their leaves and their spectral response would be different from that of conifers.

The two most common methods in digital image classification are *unsupervised classification* and *supervised classification*. A third method is a combination of these two, called *hybrid classification*.

A digital image classification system usually consists of two stages: the training stage and the classification stage. The training stage is used to determine the spectral signature of the optimal number of spectral classes. Given a set of classes after the training process, these labeled classes

are then used for classification in which the unknown pixel should be assigned to one of these labeled classes. A classified image appears as a mosaic of uniform parcels.

Image classification algorithms may be grouped into two main approaches to categorize a digital image into spectral classes: *pixel-based* and *region-based approaches*. In the region-based approach, the image is segmented into homogeneous regions and a set of meaningful features is defined. Once defined, image regions (blocks) can be categorized using pattern recognition techniques (3). However, image segmentation has been proven to be a difficult goal. In the pixel-based approach, three popular classification techniques have been widely used in remotely sensed data, namely, the minimum-distance, the parallelepiped, and the maximum-likelihood classifiers (4). All of these classifiers use spectral information to assign a pixel to a particular class.

UNSUPERVISED CLASSIFICATION

In this method, the entire classification is done almost automatically. This method involves algorithms that examine and cluster pixels into spectral classes (i.e., clusters) based on their spectral values. The intervention of the analyst occurs only toward the end of the classification and it is limited to regrouping spectral classes into information classes. Of course, this means that the analyst must have strong knowledge regarding the spectral characteristics of the ground features to be able to label the different classes. The analyst may use any ancillary data (aerial photographs, maps, etc.) that may be helpful to identify these classes.

Many unsupervised algorithms have been developed to classify an image into spectral classes. Most of these algorithms use two phases (passes) (2):

- First Pass Cluster building
- Second Pass Assignment of pixels to classes using the minimum distance classifier.

After these passes the intervention of the analyst is needed to regroup the classified spectral classes into information (i.e. thematic) classes.

Cluster building

To perform this phase, the analyst may be asked to provide four types of information:

- R, a spectral radius distance that will tell the computer when a new cluster should be determined (e.g., R = 15 spectral units),
- N, the number of pixels to process between each cluster merging (e.g., N = 2000 pixels),
- C, a spectral distance that determines if two or more clusters need to be merged after processing N pixels (e.g., C = 30 spectral units), and
- $A_{\max}$, the maximum number of spectral classes to identify (e.g., $A_{\max}$ = 20 clusters).

These parameters may be set to default by the algorithm if not specified by the analyst.

The algorithm starts at the origin of the image (pixel [1,1]) and begins to evaluate every pixel in the image (Fig. 2). To simplify the process, let us consider the first three pixels only of the image. Pixel 1 = [10,10], pixel 2 = [20,20], and pixel 3 = [30,20] in two bands (Band 4 and Band 5) (2). The spectral relationship between the three pixels is illustrated in Fig. 2(a).

The digital number [10,10] associated with pixel 1 represents the mean value (μ_1) of the first spectral class in the two bands. Remember that in image classification, we usually use n bands ($n > 2$). Next, the algorithm will evaluate pixel 2 ([20,20]). If the spectral distance (D) between cluster 1 (μ_1= [10,10]) and pixel 2 is greater than R (R = 15), pixel 2 will constitute a separate cluster (cluster 2) with a cluster mean (μ_2 = [20,20]). But if the distance D is less than R, then pixel 2 will be merged with cluster 1 and a new mean is computed. The new mean for cluster 1 becomes the weighted mean of pixel 1 and pixel 2 ([10+20]/2, [10+20]/2), with a weight of 2 [Fig. 2(b)].

In our example, the distance D between cluster 1 (i.e., pixel 1) and pixel 2 is 14.14, computed as:

$$\sqrt{(20-10)^2+(20-15)^2} = \sqrt{200} = 4.14\{<\}15$$

Because $D < R$, pixel 2 will not form a new cluster and it will be merged with cluster 1. The new cluster will have a new mean with a new location at [15,15]. Next, pixel 3 ([30,20]) is evaluated relative to the new cluster mean ([15,15]) using the same formula:

$$\sqrt{(30-15)^2+(20-15)^2} = \sqrt{250} = 15.81 > 15$$

Because 15.81 is greater than 15, pixel 3 will form a separate cluster with a mean value of [30,20] [Fig. 2(c)].

This process will continue until the number of processed pixels is reaches N (N = 2000, in our case). At this point, the algorithm stops evaluating individual pixels and examines the nature of spectral clusters developed so far. The algorithm computes the distance between the means of all the existing clusters. Clusters which mean is less than C (C = 30, in our case), will be merged and a new cluster mean is computed as a weighted mean of all the pixels in the new cluster. When no cluster is within 30 units of any other cluster, the algorithm will continue with the next pixel (pixel N + 1). The process will continue until the entire image is classified.

It should be noted that by adding a pixel to a cluster, the mean of the cluster is displaced less dramatically as the number of pixels increases in the cluster [Fig. 2(d)]. This is because each time a new pixel is added to a cluster, the new mean is a weighted mean of all the pixels in the cluster, not just between the new added pixel and the old mean of the cluster.

Assignment of pixels to one of the $A_{\max}$

Once the entire image is classified and number of clusters specified is reached based on the specified parameters (R and C), there will still be a number of pixels that have not meet the specified requirements and, therefore, will not belong to a cluster. Because an image classification requires

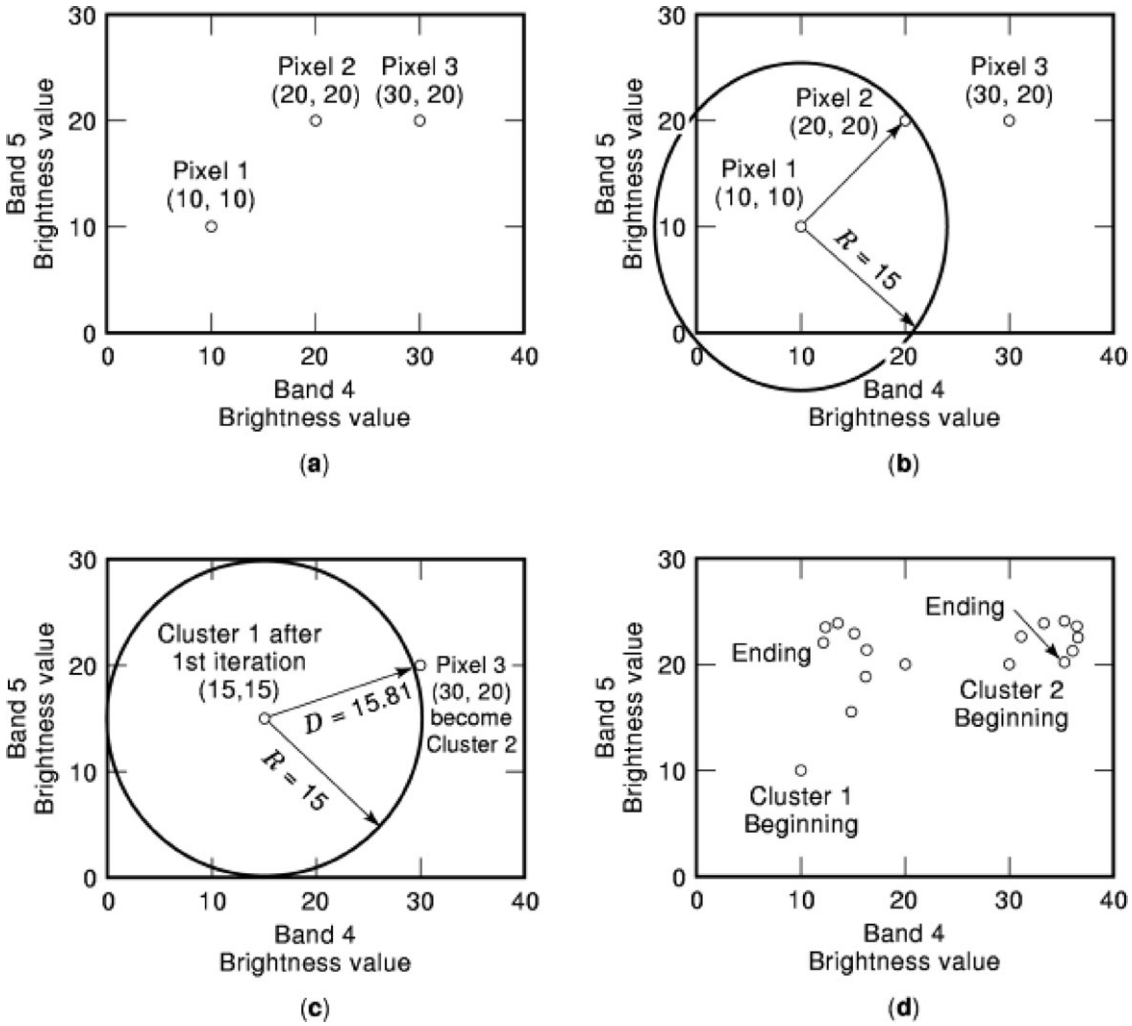

Figure 2. A two-dimensional representation of pixel clustering during the unsupervised classification process: (a) the original data showing the relationship among pixels, (b) evaluation of the spectral distance between the first two pixels, which form cluster 1 because the spectral distance (D) between the two pixels is less than the tolerance distance (R), (c) evaluation of pixel 3 against the mean value of cluster 1, which forms a new cluster (cluster 2) because the spectral distance between pixel 3 and cluster 1 is greater than R, (d) displacement of the cluster means during the several iterations of the classification process (from Jensen, Ref. 2).

that each pixel belongs to a class, the algorithm will then assign the remaining pixels using the minimum distance technique.

Establishment of information classes

Generally, the analyst produces a scatter plot combining many bands, two at a time, to analyze the spectral location of each pixel. Next, the analyst identifies the pixels on the image and labels them if possible, then regroup spectral classes that constitute one information category. It is at this stage where a thorough knowledge of the terrain properties becomes very important.

An example of spectral clusters is illustrated in Fig. 3(a) for bands 3 and 4. Because bands 3 (red) and 4 (infrared) have low correlation, they are often used to identify and regroup the spectral classes into information classes for vegetated areas. Figure 3 displays a scatter plot similar to that of the vegetation index.

Identification of spectral classes is often done interactively. The visual analysis, in conjunction with the scatter diagram, helps the analyst to regroup the spectral classes into information classes as shown in Fig. 3(b).

- Spectral class 1 forms a distinct class, water
- Spectral classes 4 and 5 are located in a spectral region contained between forest and water and, therefore, they are assigned to wetlands.

Many statistical algorithms have been developed to carry out an unsupervised classification. Some of the most commonly used are the iterative self-organizing data analysis technique (ISODATA), the K-means, and the hierarchical algorithms (5).

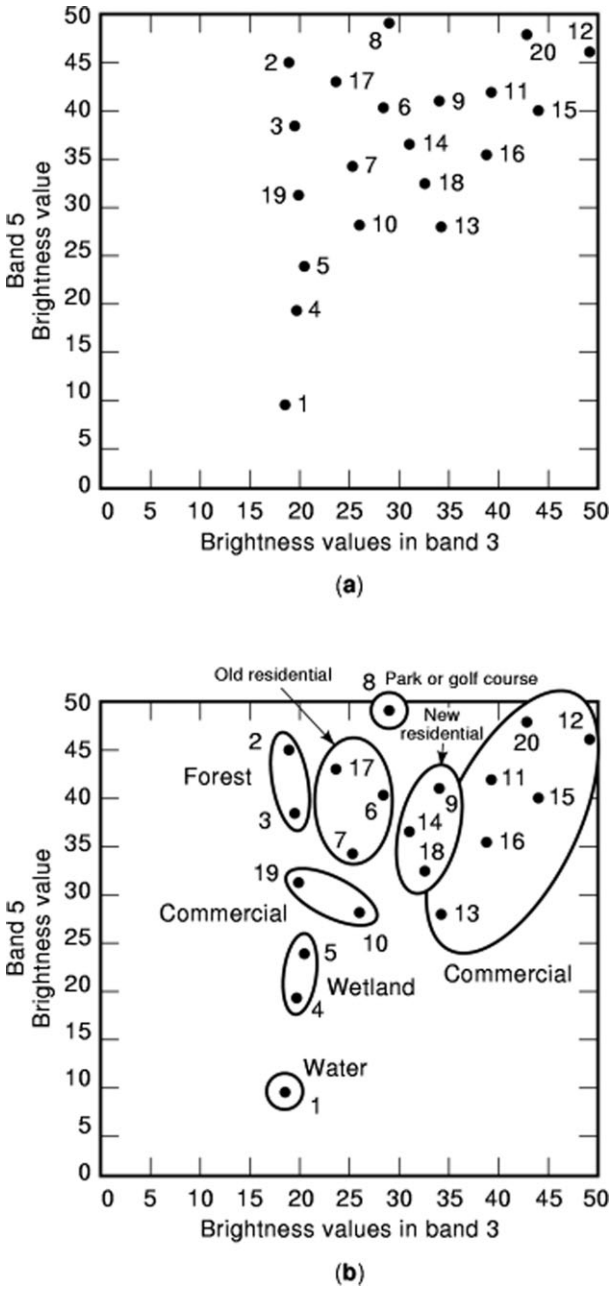

Figure 3. Spectral representation of the mean values of the final clusters formed during the unsupervised classification process using bands 3 (red) and 4 (infrared); (a) spectral clusters as determined by the classifying algorithm before been identified and labeled by the analyst and (b) identification, regrouping, and labeling of the spectral classes into information (or thematic) classes. This step requires experience and knowledge of the spectral characteristics of ground features. Photographs, field data, and any other ancillary information may be used to conduct this process (from Jensen, Ref. 2).

SUPERVISED CLASSIFICATION

To conduct a supervised classification, seven steps are necessary:

1. Develop an adequate classification scheme
2. Select *accurate* training sites
3. Extract statistical information from the training sites
4. Analyze the statistical data to select the best bands for the classification
5. Select an appropriate classification algorithm
6. Classify the image
7. Evaluate the image classification accuracy and repeat the procedure if necessary

Classification Scheme

Before classifying the image, the analyst must define the different information categories to extract. For this, a classification system must be established so that each pixel is attributed to a specific predefined class. Many systems have been developed for this purpose [e.g., USGS Land Use/Cover Classification System, US Fish and Wildlife Service Wetland Classification System, NOAA CoastWatch Land Cover Classification System (1)].

Training Site Selection

This step is critical in supervised image classification. It consists of training the computer to recognize the spectral signature of each category (class) to classify. Any error introduced at this level will be carried through the entire classification process and will lead to misclassification. To obtain satisfactory results, the sample sites must reflect the spectral characteristics of the classes they represent as accurately as possible. This means that the analyst must obtain accurate descriptive statistics for all the *spectral classes* (*cluster*) forming each *information class* (theme) to be mapped.

In other words, all the spectral classes constituting each information class must be adequately represented by the training samples used to classify the image. For example, a feature located on different soil types requires as many training sites as that of soil types to adequately be represented. This information will be used to define the decision rules, and each pixel of the image is then assigned to one of the most appropriate classes based on the decision rules in the classification phase.

Training sites are usually identified and delineated directly on the computer screen. Figure 4 shows an example of selecting training sites. Notice that the training sites have carefully been delineated inside homogenous areas to ovoid including *mixed* (or *polluted*) pixels along the edges of the areas.

The best way to represent all the information classes is to select a high number of training sites containing a small but very homogeneous number of pixels. For example, 20 sites of 40 pixels each are much better than 5 sites of 160 pixels each. In other words, it is important to avoid excluding important spectral classes to represent each information class and, at the same time, avoid including redundant training pixels to reduce the manipulation time.

Statistics Extraction and Analysis

Statistical distribution of the training sites may be analyzed graphically (qualitatively) or numerically (quantita-

Figure 4. Delineation of training sites on the computer screen. The analyst must have definite knowledge about the areas to represent and the training sites must be carefully delineated inside homogenous fields to ovoid inclusion of pixels (i.e., mixed or polluted pixels) from other categories surrounding the area under consideration. These training sites will train the classifying algorithm to recognize the spectral signature of each category; and "bad" training will confuse the algorithm and result in misclassification (from Lillesand and Kiefer, Ref. 4).

tively). Generally the combination of the two techniques is used for better results.

The statistical method is used to select the bands that give the highest degree of statistical separation among classes. The quantitative distribution is generally represented by variance-covariance and correlation coefficient tables. However, quantitative analysis alone is not always sufficient to understand the nature of the spectral data. Consequently, we combine this type of analysis with graphical analysis.

Many graphical techniques have been developed to select a combination of bands that provide the best discrimination among image features. Figure 5 shows a spectral configuration of a training site in five bands. Histogram representation is very important when the *maximum-likelihood classifier* (discussed later) is used because this classifier requires a normal distribution, which can easily be shown by a histogram representation.

Although the histograms in Fig. 5 show a good representation of individual categories, they do not give us enough information concerning the separability among the different classes. Consequently, we use what we call coincident spectral plots (Fig. 6) (4). This figure illustrates the mean spectral response of each category (with a letter: C = Corn, F = Forest, H = Hay, S = Sand, U = Urban, and W = Water) and the variance of the distribution ($\mu \pm 2$ standard deviation) for each category in each band. Figure 6 indicates that the hay and corn response patterns overlap in all spectral bands. The plot also shows which combination of bands might be best for discrimination among the different features such as bands 3 and 5 for hay and corn discrimination.

The main challenge in spectral recognition of a ground feature is to find a technique that leads to high separability among features using a minimum number of bands. When there is spectral overlap among classes, any decision used to separate among classes must be taken into consideration for two types of errors:

- Error of commission when a pixel is assigned to a class to which it should not belong.
- Error of omission when a pixel is omitted from a class to which it should belong.

If the spectral signatures of the classes are normally distributed, it is possible to use what we call *divergence* (degree of separability) to separate between two classes. Divergence between the two classes C and D is given as (2):

$$Diver_{CD} = 0.5Tr[V_C - V_D)(V_D^{-1} - V_C^{-1})] + 0.5[(V_C^{-1} + V_D^{-1}) \times (M_C - M_D)(M_C - M_D)^T)] \quad (1)$$

Where, T_r is the sum of the diagonal elements of the matrix [Y], V_C and V_D are the covariance matrices for classes C and D, V^{-1} is the inverse matrix of V, T indicates the transpose of the matrix, and M_C and M_D are the mean values for classes C and D.

Selection of an Image Classification Algorithm

Numerous classification algorithms have been developed to categorize a digital image into spectral classes. The choice of a particular technique depends on the nature of the data to be processed, the output to be derived, and the accuracy required. The most common algorithms, however, are the *minimum distance*, the *parallelepiped*, and the *maximum likelihood*.

To analyze each of these algorithms, let us consider a simple spectral diagram on which six classes are represented by the scatter plots [Fig. 7(a)]. To simplify the analysis, let us consider two bands only (MSS3 and MSS4). Also, let us assume that the pixel observations are from areas of known cover types (from training sites, for example).

Minimum Distance. Remember that among the statistical parameters extracted from an image is the mean of each class, which should be determined before the classification procedure. Class means are indicated by a plus sign in Fig. 7(a). By considering the two-band pixel values as positional coordinates (as they are portrayed in the scatter plot), a pixel of unknown identity may be classified by computing the distance (i.e., *spectral distance*) between the value of the unknown pixel and the mean of each class. The unknown pixel is attributed to the closest class. Computation of this distance is done using the Euclidean distance based on the Pythagorean theorem (2):

$$Dist = \sqrt{(DN_{i,i,k} - \mu_{c,k})^2 + (DN_{i,j,l} - \mu_{c,l})^2} \quad (2)$$

where $DN_{i,j,k}$ is the digital number of the unknown pixel (i,j) in band k, $DN_{i,j,l}$ is the digital number of the unknown pixel (i,j) in band l, and $\mu_{c,k}$ and $\mu_{c,l}$ are the spectral means for class C in band k and l, respectively.

In a multispectral (more than two bands) image classification, the same formula applies as

$$Dist = \sqrt{\sum_{k=1}^{k=n} (DN_{i,j,k} - \mu_{c,k})^2} \quad (3)$$

where n is the number of bands.

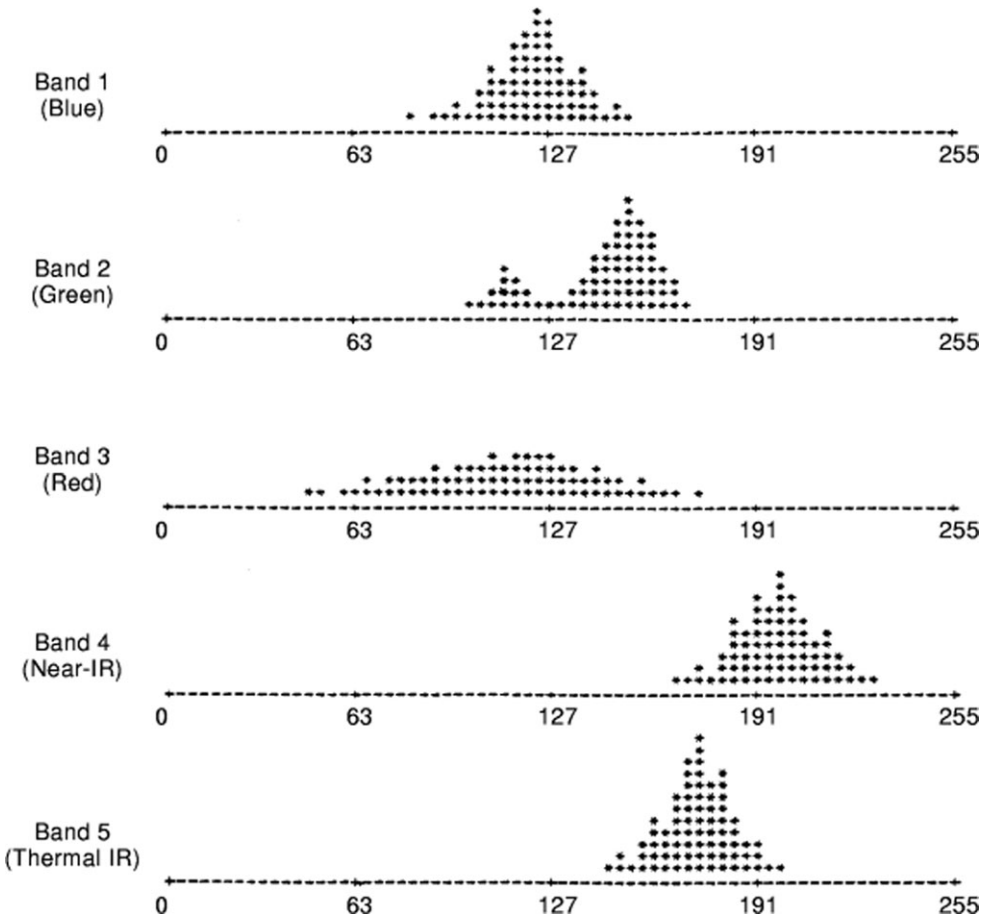

Figure 5. An example of histogram representation of the training sites to verify their normality, which is an important criterion when the maximum likelihood classifier is used. If the training sites are well selected, their histograms should be normally distributed. Note that in band 2, the histogram is bimodal, which suggests that the training site may be composed of two different subclasses, two different types of the same class, or different illumination conditions (i.e., shadowed versus nonshadowed areas) (from Lillesand and Kiefer, Ref. 4).

The minimum distance algorithm is simple and fast but insensitive to different degrees of spectral variance in the spectral response data. For example, in Fig. 7(b), the minimum-distance classifier would assign the pixel at point 2 to class "sand" in spite of the fact that the greater variability in the "urban" class suggests that pixel at point 2 should belong to "urban." Consequently, this classifier is rarely used when classes are spectrally close to one another and have high spectral variances. There exist several distance measurement techniques among which the Euclidean and city block distances are commonly used (6).

Parallelepiped Classification Algorithm. This technique introduces the sensitivity to class variance by considering the range of values in each training site. This range may be defined by the lowest and highest digital values of the training site in each training site. It may also be defined as a function of the standard deviation (SD) of each class in each band. This results in rectangular areas called *parallelepipeds* [Fig. 7(c)]. Using a threshold of 1 SD, an unknown pixel is assigned to a category if its digital number (DN) falls within the lower limit ($\mu - 1$ SD) and the upper limit ($\mu + 1$ SD) in each band.

$$\mu_{c,k} - SD_{c,k} \leq DN_{i,j,k} \leq \mu_{c,k} + SD_{c,k} \tag{4}$$

where $c = 1, 2, 3, \ldots$ is the class number, $k = 1, 2, 3, \ldots$ is the band number, $\mu_{c,k} - \mathrm{SD}_{c,k}$ is the lower limit (or the lowest value in band k for a class C), and $\mu_{c,k} + \mathrm{SD}_{c,k}$ is the upper limit (or the highest value in band k for a class C).

The parallelepiped classification is to classify a pixel based on the decision region of spectral ranges in which it lies. Once a pixel is classified to a class, it is blocked out from consideration for all the other classes. Therefore, the behavior of this algorithm is influenced by the order of spectral classes specified in the classification process. Some combinations of these classification algorithms are also used in some commercial software (7).

Sometimes, many pixels remain unclassified. In this case, assigning a pixel to a class may be done by either increasing range between the lower and upper limits (e.g., 2 SD or 3 SD) or by using the minimum distance to classify the pixels. However, the first option (i.e., 2 SD or 3 SD) may result in an overlap among rectangular areas (i.e., parallelepipeds) and pixels that are common to two or more classes would be difficult to classify. Consequently, there is a need for an algorithm that takes into account the vari-

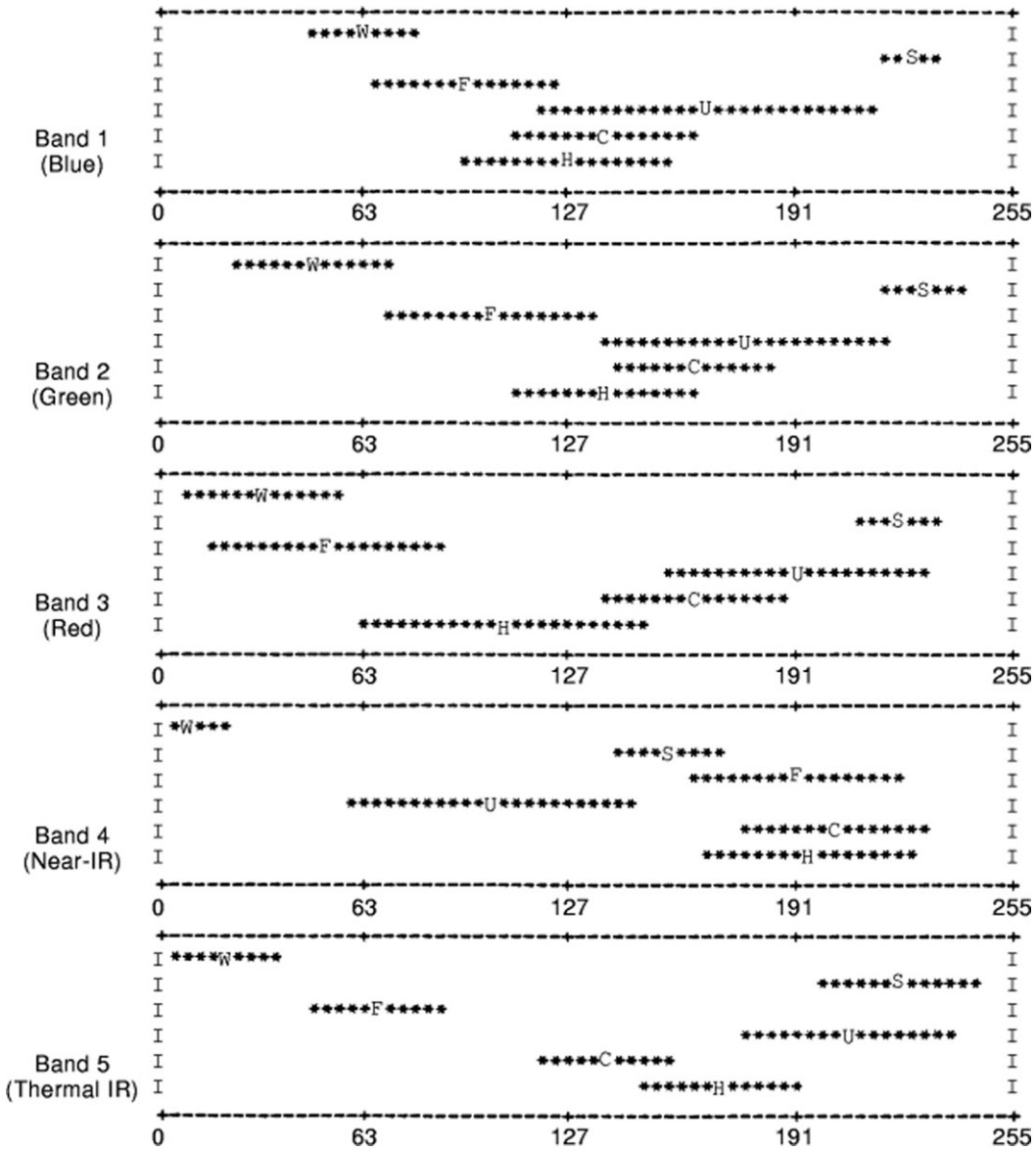

Figure 6. Coincident spectral plot representation of six classes in five bands. This diagram shows the spectral overlap among classes and helps identify the bands where confusion among classes is minimal. Note that when using multiple bands (i.e., multidimensional representation), the spectral separability among classes may be highly improved (from Lillesand and Kiefer, Ref. 4).

ability within a class and the covariability between classes. This algorithm is the *maximum likelihood*.

Maximum Likelihood. This technique uses the variance within a class and the covariance between classes to assign a pixel to a category. However, as mentioned earlier, this method requires a *normal* (*Gaussian*) *distribution* of the data, which is usually the case in most images. The normally distributed (bell-shaped) histograms are called *probability density functions*, which are used to determine the belonging of a pixel to a given category. The maximum likelihood also uses the mean ($\mu_{c,k}$) of each class (C) in the different bands (k) and the covariance matrix (V_c) of each class in the different bands. The decision rule to assign a pixel X with a digital value of $\mathrm{DN}_{i,j,k}$ in different bands to a class C is determined as the following (2):

Pixel X belongs to class C if and only if:

$$P_C \geq P_i \quad \text{where } i = 1, 2, 3, \ldots, m \text{ possible classes}$$

$$P_C = \{-0.5\, log_e[det(V_{Ckl})]\} - [0.5(X - M_C)^T V_{Ckl}^{-1}(X - M_C)] \quad (5)$$

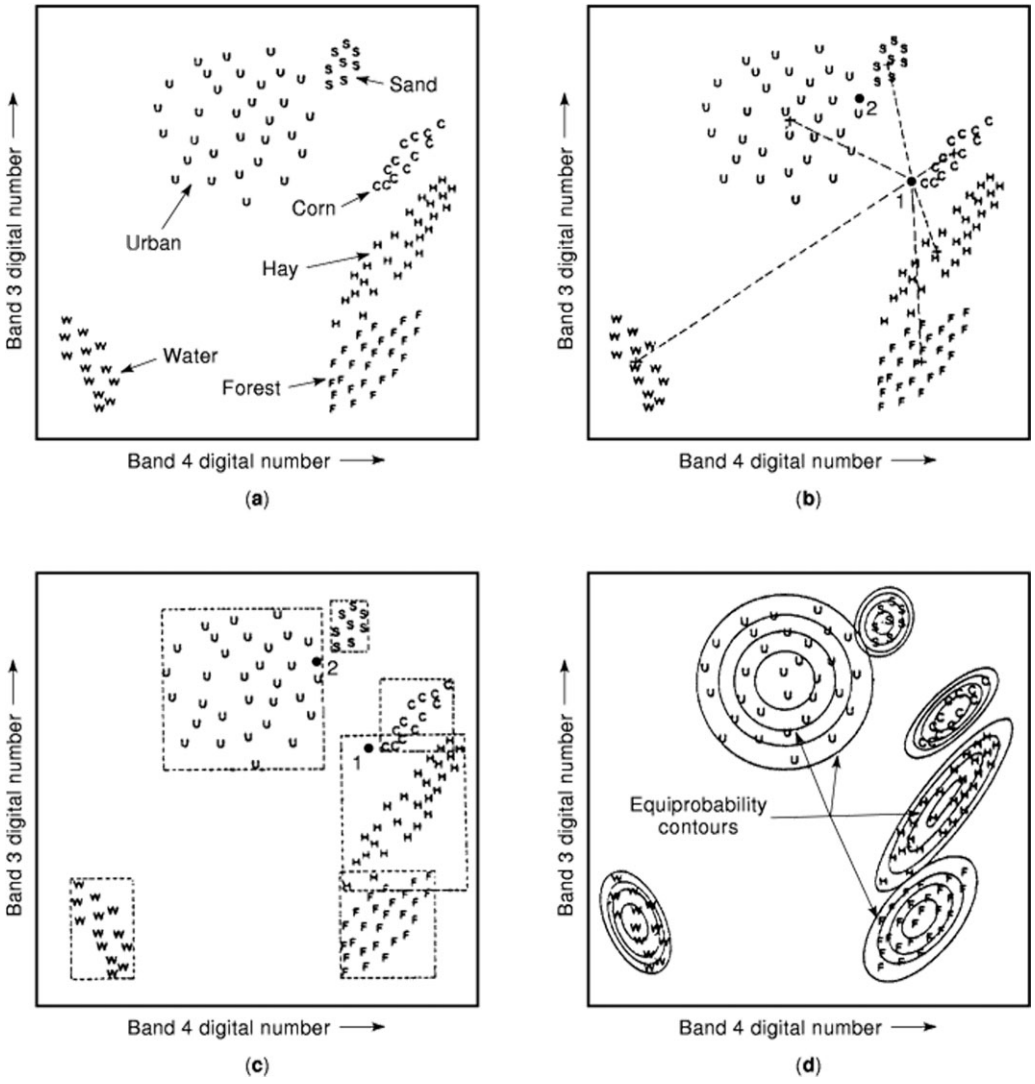

Figure 7. Scatter diagram of six classes using bands 3 and 4. (a) Spectral distribution of classes before classification, (b) the minimum distance classifier assigns a pixel to the closest class mean (represented by "+" without considering the spectral variability within or among classes, (c) the parallelepiped classifier takes into account the class variance by considering the range of values within a class, but it does not take into account the variation among classes, which causes the overlap among clases, and (d) the maximum-likelihood classifier takes into account both the variance within classes and the covariance among classes, which usually produce better results (from Lillesand and Kiefer, Ref. 4).

where

$$V_{Ckl} = \begin{bmatrix} cov_{C11} & cov_{C12} & \cdots & cov_{C1n} \\ cov_{C21} & cov_{C22} & \cdots & cov_{C2n} \\ \vdots & \vdots & \cdots & \vdots \\ cov_{Cn1} & cov_{Cn2} & \cdots & cov_{Cnn} \end{bmatrix}$$

$$M_C \begin{bmatrix} \mu_{C1} \\ \mu_{C2} \\ \vdots \\ \mu_{Ck} \end{bmatrix} \quad \text{and} \quad X = \begin{bmatrix} DN_{i,j,1} \\ DN_{i,j,2} \\ \vdots \\ DN_{i,j,k} \end{bmatrix}$$

cov_{ckl} is the covariance value of class C in the band k and band l, μ_{ck} is the mean value of class C in band k, $\text{DN}_{i,j,k}$ is the digital number of pixel i, j in band k, and det is the determinant of the matrix. X is assigned to the class that has the highest P_C.

Another way to determine the belonging of a pixel to a class using the maximum- likelihood technique is to compute its likelihood value. This simple technique is generally preferred.

$$Likelihood(X) = e^{-\left[\frac{(X-\overline{X})^2}{2(SD)^2}\right]} \tag{6}$$

where X is the pixel value being evaluated and $\overline{X}$ is the mean value of the spectral class under consideration.

Graphically, the maximum-likelihood method results in ellipsoidal curves that follow the direction of the variability of the scatter plots [Fig. 7(d)]. This technique is generally more accurate than the minimum distance and the parallelepiped if the training sites are carefully selected. The principal drawback of this method is the computation time

it takes to classify a pixel. This is particularly true when a large number of bands are involved or a large number of classes must be differentiated.

In case the analyst has some knowledge (e.g., percentage of the total area of each class) about the area to classify, he or she can employ that information to increase the probability of accurately assigning a pixel to the correct category. This technique is known as a priori probability. In this case, the maximum likelihood of a pixel to belong to a class depends upon the a priori probability of occurrence of that pixel during the classification process. Let us define some notations before deriving the maximum-likelihood classification.

$P(\omega_i)$ is the a priori probability of class ω_i.

$P(\omega_i/x)$ is the probability that x comes from class ω_i.

$P(x/\omega_i)$ is the probability (the likelihood function) of class ω_i.

$P(x)$ is the sum of the probability $P(x/\omega_i)$ for all classes.

The decision rule of the maximum-likelihood classification is to assign the pixel to the class with the highest probability (6, 8). To assign a pixel to one of the spectral classes, we have to compute the probability $P(\omega_i/x)$ for each class. Using Bayes' formula,

$$P(\omega_i/x) = P(x/\omega_i)P(\omega_i)/P(x) \tag{7}$$

The $P(x/\omega_i)$ can be estimated from the histogram of the training data for class ω_i and $P(\omega_i)$ can be estimated from maps and the a priori knowledge of the area. Since the $P(x/\omega_i)$ is computed from the histogram of each class, it will use a lot of space in memory for the storage of these histograms when we implement this classification using a computer program (6). The estimation of $P(x/\omega_i)$ may be simplified by approximating the histogram with the Gaussian function. A multivariate Gaussian function of the $P(x/\omega_i)$ is defined as

$$P(x/\omega_i) = \frac{1}{(2\pi)^{n/2}|C_i|^{1/2}} exp[-\frac{1}{2}(x-m_i)^T C_i^{-1}(x-m_i)],$$
$$i = 1, 2, \ldots, M \tag{8}$$

where m_i is the mean vector of class ω_i and C_i is the covariance matrix of class ω_i. These parameters can be calculated from the histogram of each class. Substituting $P(x/\omega_i)$ in Eq. (5) into Eq. (7), taking the natural logarithm of Eq. (7), and dropping the constant terms (8), we will have

$$P(\omega_i/x) = ln\ P(\omega_i) - 1/2\, ln|C_i| - \frac{1}{2}[(x-m_i)^T C_i^{-1}(x-m_i)],$$
$$i = 1, \ldots, M \tag{9}$$

where T is the transpose of the vector and C_i^{-1} is the inverse covariance of C_i. Equation (13) is the decision function used in the maximum-likelihood classification in which the pixel will be classified to the class having the minimum $P(\omega_i/x)$.

Selection of a classification algorithm depends on many factors that the analyst should evaluate before initializing the classification procedure. These factors may include, for example, considerations such as

- Type of the data
- Type of the final product
- Type of application
- Means available
- Accuracy requirement
- Production cost versus accuracy

HYBRID CLASSIFICATION

This technique involves both unsupervised and supervised classification methods. It incorporates the advantages of both methods while minimizing their disadvantages. Hybrid classification uses the unsupervised classification to generate unbiased and homogeneous training sites and the supervised method to assign pixels to spectral classes. This technique is usually more accurate than unsupervised or supervised techniques used alone.

There are several training algorithms used in hybrid classification. These algorithms attempt to define the optimal number of spectral classes by exploring the intrinsic and natural structure of a representative data set. This data set is sampled from the images. Many statistical clustering algorithms such as the K-means and ISODATA (8) have been widely used as unsupervised training methods. Although K-means clustering has a strong tendency to local minima and its clustering result is heavily dependent on its initial cluster distribution (9), it is a simple and widely used algorithm. The K-means algorithm is to minimize a performance index, which is based on the measurement of Euclidean distance. The algorithm is repeated several times until it converges or is already in the limit of the convergence tolerance. We assume that there are M pixels in the training data set and each pixel is represented as X. The algorithm is sketched in the following:

Step 1.Choose the number of clusters, K, and the convergence tolerance, δ.

Step 2.Generate K cluster centers $C_1(1), C_2(1), \ldots, C_k(1)$ arbitrarily.

Step 3.Distribute the samples among K clusters $C_1(i)$, $C_2(i), \ldots, C_k(i)$ at the ith iteration by the minimum distance criterion using the Euclidean Distance measure and update the centroid of each cluster for the next iteration, i.e. $C_1(i+1), C_2(i+1), \ldots, C_k(i+1)$.

Step 4.If $|C_j(i+1) - C_j(i)| \leq \delta$ for $j = 1, 2, 3, \ldots, K$, the algorithm converges. Otherwise, repeat steps 2 to 4.

From steps 3 and 4, the optimal clustering result is obtained by minimizing the objective function

$$\sigma = \frac{1}{M}\sqrt{\sum_{i=1}^{K}\sum_{x \in C_i}(x - C_i)^2} \tag{10}$$

which is the standard deviation function. The function is minimized when each pixel is equal or very close to its assigned cluster center. The ISODATA algorithm is an iterative, complex procedure, which is similar to the K-means algorithm in principle. However, the ISODATA is capable

of merging and merging clusters (8). The user needs to determine a few numbers of input parameters for the ISODATA algorithm.

The hybrid training is a combination of supervised and unsupervised training methods. If the analyst does not have a complete knowledge about the ground truth, some spectral classes are created using a supervised training approach and some spectral classes may be established using an unsupervised training method (7).

MULTIPLE CLASSIFIER SYSTEMS

Multiple classifier systems (MCS) exploit the complementary discriminatory information between each classifier and results in the better performance than that of a single classifier. Combining classifiers can further reduce the classification error by using perturbation, reweighting and combination techniques (47). The design of a MCS involves two main phases: the design of the classifier ensemble and the design of the combination function. The purpose of the first phase is to generate a set of mutually complementary classifiers. Three popular techniques are available for this purpose–bagging, boosting and the random subspace method (44–46). The principle of these techniques is to train each classifier on a different subset of the training data. The subset of the training data is generated by using some sampling techniques.

In the design of the combination function, many combination techniques have been proposed (45–49). Methods for classifiers combination include the fusion of label outputs, naïve bayes combination, Dempster-Shafer method, etc. (46). In many application areas, such as information retrieval, image processing and computational biology, analysis of high dimensional datasets is frequently encountered. For high dimensional data classification, feature extraction technique is usually applied to reduce the dimensionality. But most papers using bagging, boosting or random subspace methods to design classifiers do not discuss the effect or impact of applying feature extraction as a preprocessing step.

Some hybrid algorithms based on bagging (BG) and random subspace method (RSM) have been proposed (48). The effect of using original data and transformed data in bagging, random subspace was also investigated. Two classifier overproduction techniques, bagging and random subspace methods, are briefly introduced.

Bagging

1. Repeat for $b = 1,2,\ldots,B$:
   a. Take a bootstrap replicate X^b of the training data set X.
   b. Construct a classifier $C^b(x)$ (base classifier) on X^b.
2. Combine classifiers $C^b(x)$, $b = 1,2,\ldots,B$, by simple majority voting (the most often predicted label) to a final decision rule

Random Subspace Method

1. Repeat for $b = 1,2,\ldots,B$:
   a. select an r-dimensional random subspace $\tilde{X}^b$ from the original p-dimensional feature space.
   b. Construct a classifier $C^b(x)$ in $\tilde{X}^b$.
2. Combine classifiers $C^b(x)$, $b = 1,2,\ldots,B$, by simple majority voting to a final decision rule.

The performances of bagging and random subspace methods have been explored (48). It shows that random subspace method is good when the training sample size is less or close to the dimensionality and bagging is useful when the training sample size is greater than and close to the dimensionality (critical condition). If the training sample size is much greater than dimensionality, then both overproduction methods is useless.

CLASSIFICATION ACCURACY ASSESSMENT

Classification accuracy assessment of digital images requires a comparison between a classified image and a reference image called *ground truth* (2). Ground truth image may be generated from field data using a sampling strategy or by interpreting large-scale aerial photographs. Often, the latter method is used. The result is an image assumed to represent the ground truth correctly. Then, the classified and the reference images must be perfectly registered, having the same scale, referenced in the same coordinate system, and having the same class numbering.

Classified image	Reference image
Class 1	Class 1
Class 2	Class 2
Class 3	Class 3
.	.
.	.
Etc.	Etc.

Two methods may be used to assess classification accuracy:

- Comparison by total area
- Comparison by pixel

Comparison by total area requires the knowledge of the total area of each class in the reference and the classified images. Then, the total area of each class in the classified image is compared to the total area of the class in the reference image.

Example

Type	Reference image	Classified image
Forest	35 %	32 %
Bare soil	27 %	29 %
Water	15 %	13 %
Agriculture	23 %	26 %

If we know in advance the percentage of each category in the classified area (reference image) as specified in the preceding example, then, we may calculate the percentage of individual or total accuracy of our classification. However, this computation will not be accurate if we do not know the location of these classified areas. For example, 32 % forest may be 20 % forest and 12 % something else. The same may

	F	S	E	Ag	True
F	35	14	11	1	61
S	4	11	3	0	18
E	12	9	38	4	63
Ag	2	5	12	2	21
Predicted	53	39	64	7	163

Overall accuracy = (35 + 11 + 38 + 2) / 163 = 53%

Omission errors: (Producer's Accuracy)
35 / 53 = 66.0%
11 / 39 = 28.2%
38 / 64 = 59.4%
2 / 7 = 28.7%

Commission errors: (User's Accuracy)
35 / 61 = 57.4%
11 / 18 = 61.1%
38 / 63 = 60.3%
2 / 21 = 9.5%

Figure 8. Confusion or error matrix.

	F	S	E	Ag	True
F	3233	2379	3904	427	
S	954	702	1152	126	
E	3339	2457	4032	441	
Ag	1113	819	1344	147	
Predicted					26569

3233 = 61 × 53
2379 = 61 × 39
3904 = 61 × 64
427 = 61 × 7

954 = 18 × 53
702 = 18 × 39
etc...

Overall accuracy = (3233 + 702 + 4032 + 147) / 26569 = 30%

K = Observed value − Predicted value) / (1 − Predicted value) = (0.53 − 0.305) / (1 − 0.305) = 32%

Figure 9. κ statistics computed from the error matrix.

be true for the other classes.

Pixel by pixel comparison is the most used technique in digital image classification accuracy assessment. This technique requires a reference image to which the classified image is to be compared. The reference image may be obtained from existing data (maps, aerial photographs, etc.) or by sampling a number of pixels on the classified image and then verifying them on the ground using GPS techniques, for example. The number of pixels to be sampled is critical to meet the required accuracy. Fitzpatrick-Lins (10), suggests that the minimum sample size N to be used to evaluate the accuracy of a land-use classification map is determined from the binomial probability theory as

$$N = \frac{Z^2(p)(q)}{E^2} \tag{11}$$

where p is the expected percent accuracy, may be determined using a random sample of pixels, $q = 100 - p$, E is the allowable error, and $Z = 2$ from the standard normal deviation of 1.96 for 95% two-sided confidence level.

For example, if we want to be 95 % confident that our expected classification accuracy will be 85 %, then we need at least N sample points, computed as:

$$N = \frac{2^2(85)(15)}{5^2}$$

Usually, more points need to be sampled to be able to draw reliable conclusions on the classification accuracy.

Once, we have our reference data, we may compare it to the reference data. This results in a matrix called the *confusion* or *error matrix*. The diagonal of the matrix represent the pixels well classified and the rest of the matrix represent the pixels wrongly classified as shown in Fig. 8.

Error matrices provide valuable information on the type of errors and the confusion between classes committed during classification. For example, 14 pixels of the forest area were confused with soil, but only 4 soil pixels were confused with forest area. Also, forest and water are often confused due to the similarity in their spectral characteristics. This suggests a very careful selection of the training sites, particularly when the categories to classify have close spectral responses.

The classification accuracy may be evaluated by comparing it to that of a random classification. The parameter used to do this evaluation is called *k statistics* computed from the error matrix as shown in Fig. 9. This means that our classification is 32% better than if the image was randomly classified.

NEURAL NETWORKS AND GENETIC ALGORITHMS

Artificial neural networks (ANNs), a brain-style computation model, have been used for many years in different application areas such as vector quantization, speech recognition and pattern recognition (12, 13). In general, ANN is capable of tolerating the noise, distortion, and incompleteness of data taken from the practical applications. Researchers have developed several different paradigms of ANNs (13). These paradigms are capable of detecting various features represented in input signals. An ANN is usually composed of many nonlinear computational elements. These computational elements operate in parallel to simulate the function of the human brain. An ANN is characterized by the topology, activation function, and learning rules. The topology is the architecture of how neurons are connected, the activation function is the characteristics of each neuron, and the learning rule is the strategy for learning (14). ANN is also well suited for parallel implementations because of the simplicity and repetition of

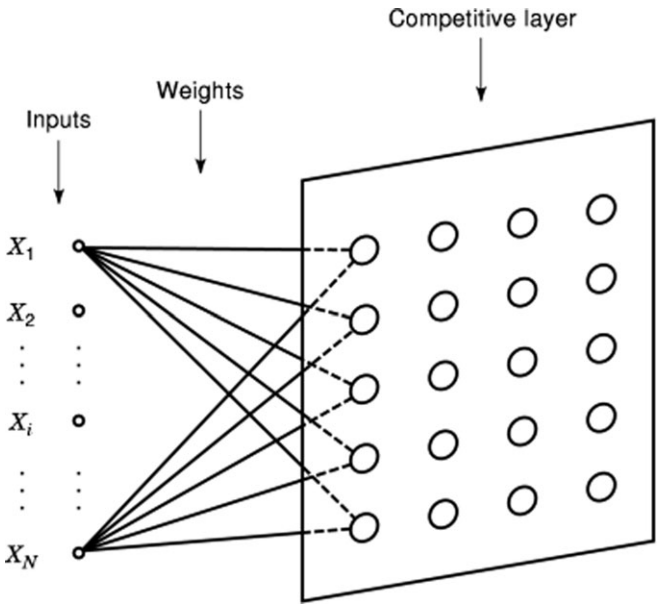

Figure 10. The $X_1, X_2, \ldots, X_i, \ldots, X_N$ are inputs, one for each component of the pixel vector corresponding to N multispectral bands. A circle denotes a neuron. Neurons are interconnected in the two-dimensional competitive layer. Each neuron defines a spectral class where its center values are stored in the connections between the inputs and the neurons.

the processing elements.

One type of these networks that possess the self-organizing property is called *competitive learning networks*. Three different competitive learning networks, the simple competitive learning network (SCL), Kohonen's self-organizing feature map (KSFM) and the frequency-sensitive competitive learning (FSCL) network, were proposed as unsupervised training methods in the hybrid image classification system (15). Similar to statistical clustering algorithms, these competitive learning networks are able to find the natural groupings from the training data set. The topology of the Kohonen self-organizing feature map is represented as a two-dimensional, one-layered output neural net as shown in Fig. 10. Each input node is connected to each output node. The dimension of the training patterns determines the number of input nodes. Unlike the output nodes in the Kohonen's feature map, there is no particular geometrical relationship between the output nodes in both the simple competitive learning network and the frequency-sensitive competitive learning network. During the process of training, the input patterns are fed into the network sequentially. Output nodes represent the trained classes and the center of each class is stored in the connection weights between input and output nodes.

The following algorithm outlines the operation of the simple competitive learning network as applied to unsupervised training. Let L denote the dimension of the input vectors, which, for us, is the number of spectral bands. We assume that a two-dimensional ($N \times N$) output layer is defined for the algorithm, where N is chosen so that the expected number of the classes is less than or equal to N^2.

Step 1: Initialize weights $w_{ij}(t)$ ($i = 1,\ldots, L$ and $j = 1,\ldots, N \times N$) to small random values. Steps 2–5 are repeated for each pixel in the training data set for each iteration.

Step 2: Present an input pixel $X(t) = (x_1,\ldots, x_L)$ at time t.

Step 3: Compute the distance d_j between the x_i and each output node using

$$d_j = \sum_{i=1}^{L} (x_i - w_{ij}(t))^2$$

where i, j, L, w_{ij} and x_i are similarly defined as in steps 1 and 2.

Step 4: Select an output node j^* which has minimum distance (i.e. the winning node).

Step 5: Update weights of the winning node j^* using

$$w_{ij}(t+1) = w_{ij}(t) + \eta(t)(x_i - w_{ij}(t)), \; i = 1, \ldots, L$$

and $\quad 1 \le j \le N \times N,$

where $\eta(t)$ is a monotonically slowly decreasing function of t and its value is between 0 and 1.

Step 6: Select a subset of these N^2 output nodes as spectral classes.

The J-M distance (16), which is a measure of statistical separability of pairs of classes is employed to evaluate the capability of the proposed methods (15).

Competitive learning provides a way to discover the salient general features that can be used to classify a set of patterns. However, there are some problems associated with competitive learning neural networks in the application of remotely sensed data. Among them are (1) the underutilization of some neurons (13), (2) the fact that the learning algorithm is very sensitive to the learning rate, $\eta(t)$, in remotely sensed data analysis, and (3) the fact that the number of output nodes in the network must be greater than the number of spectral classes embedded in the training set. Ideally, the number of output nodes should be dynamically determined in the training (learning) environment instead of being specified a priori. An unsupervised training approach combining a genetic algorithm with the K-means clustering algorithm was proposed in Ref. 17.

The genetic algorithm is used to prevent fixation to the local minima. A genetic algorithm is a global search method simulating natural evolution. This evolutionary computing approach has several applications (18, 19). Evolution begins with a population of randomly selected chromosomes (candidate solutions). Chromosomes compete with one another to reproduce based on the Darwinian principle of *survival of the fittest* in each generation of evolution. After a number of generations in evolution, the chromosomes that survived in the population are the optimal solutions. With the genetic algorithm approach, it turns out that establishing initial parameters in K-means clustering for unsupervised training is not terribly important. A similar approach has been applied to the problem of color image quantization (20). A simple genetic algorithm consists of four basic elements: namely, (1) generation of populations of chromo-

somes, (2) reproduction, (3) crossover, and (4) mutation. The operation of crossover and mutation is to move a population around on the landscape defined by the fitness function. The genetic K-means algorithm consists of the following steps. We assume that the string with the lowest mean square error (MSE^*) is the optimal solution. The algorithm will search such a string for the solution.

Step 1. Choose the number of clusters, K.

Step 2. Generate P sets of cluster centers. i.e. $S_1 = \{C_1, C_2, \ldots, C_k\}$, $S_2 = \{C_1, C_2, \ldots, C_k\}$, $S_3 = \{C_1, C_2, \ldots, C_k\}, \ldots, S_p = \{C_1, C_2, \ldots, C_k\}$. Each set having different cluster centers.

Step 3. Reproduction: Formulate a string for each set, hence P strings are generated. The inverse of the mean squared error (MSE) is used as the fitness function for each string. All strings are pairwise compared. In each comparison the string with the lowest MSE will be retained and the other one will be discarded and replaced by the first one or a new random cluster. In other words, only half of the strings in a population remains and the other half is either replaced by the remaining half or regenerated by new random clusters.

Step 4. Apply K-means algorithm: Distribute the samples among K clusters by the minimum distance criterion using the Euclidean Distance measure for each string separately and update the centroid of each cluster. Store the string with the lowest MSE (MSE^*) out of these P strings as the solution.

Step 5. Crossover: For each string, one-point crossover is applied with probability p_c. A partner string is randomly chosen for the mating. Both strings are cut into two portions at a randomly selected position and the portions are mutually interchanged as shown in the following.

$$S_k = \{u_1, \ldots, u_m\} \rightarrow \{u_1, \ldots, u_j, v_{j+1}, v_m\}$$
$$S_l = \{v_1, \ldots, v_m\} \rightarrow \{v_1, \ldots, v_j, u_{j+1}, u_m\}$$

Step 6. Mutation: Mutation with probability p_m is done on a component for each string. Either -1 or 1 is selected randomly with probability 0.5 and added to the chosen component. The mutation operation is used to prevent fixation to the local minimum.

Step 7. Repeat steps 3 to 6 for several generations.

TEXTURAL, CONTEXTUAL, AND NEURAL NETWORKS CLASSIFIERS

Per-pixel classifiers use spectral information to classify each pixel in the image. These classifiers tend to generate a salt-and-pepper appearing classified image. One of the main drawbacks of these methods is that each pixel is treated independently of its neighbors. Remotely sensed images may be regarded as samples of random processes because of the variations of object characteristics and noise (21). Thus, each pixel in the image can be regarded as a random variable. This indicates that the per-pixel classifier may not be reliable by looking at each individual pixel value for the decision purpose. To improve the classification accuracy, image classifiers that incorporate nonspectral features have been proposed in the literature (22). Image classifiers have been developed using textural, contextual, and ancillary information. Texture refers to a description of the spatial variability of tones within part of an image. Various methods have been developed to extract statistical textural features (3). Context is a measure of the relationships between the pixels in a neighborhood. Contextual information has been used in different phases in the classification process. Kittler and Foglein (23) outlined four different approaches to incorporating contextual information in image classification and developed a contextual algorithm using spatial and stochastic information. The authors also showed that the use of contextual information increases the reliability of classification. A brief review of several contextual classifiers was given by Sharma and Sarkar (24). A spatial classifier uses the sigma probability of the Gaussian distribution and the connectivity property was proposed in Ref. 25. The algorithm classifies a pixel by considering those neighboring pixels that have intensities within an adaptive s range of the pixel. Any neighboring pixel outside the adaptive s range most likely comes from a different class and, therefore, should not be included in the class which is being considered (36). Connectivity ensures that all the pixels within the sigma range are not randomly distributed (noise). To determine whether two pixels are connected, a criterion of similarity must be established. The 4-neighbors (horizontal and vertical) and 8-neighbors (plus diagonal) are commonly used in digital image processing. Markov Random Field (MRF) probability models have been widely studied to incorporate the contextual information in the image classification and segmentation process (23).

Many adaptive, non-parametric neural-net classifiers have been proposed for real-world problems. These classifiers show that they are capable of achieving higher classification accuracy than conventional pix-based classifiers (26); however, few neural-network classifiers which apply spatial information have been proposed. The feed-forward multilayer neural network has been widely used in supervised image classification of remotely sensed data (27). Arora and Foody (28) concluded that the feed-forward multilayer neural networks would produce the most accurate classification results. A Feed-forward multilayer network as shown in Fig. 11 is an interconnected network in which neurons are arranged in multilayers and fully connected. There is a value called *weight* associated with each connection. These weights are adjusted using the back-propagation algorithm or its variations, which is called *training* the neural networks (13). Once the network is well trained, it can be used to perform the image classification.

REDUCTION OF THE COMPUTATIONAL COMPLEXITY

The maximum-likelihood (ML) classifier is an optimal classification algorithm that has been widely used by the analyst. However, this classifier requires a tremendous amount of computational time involving the covariance, inverse covariance matrices and determinant of the matri-

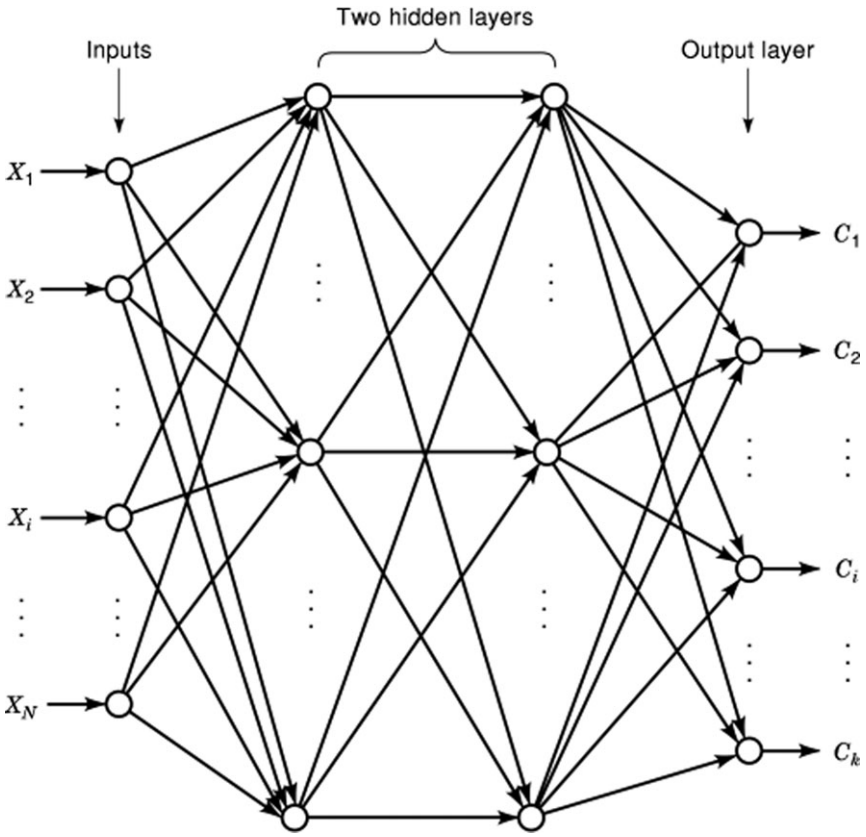

Figure 11. Architecture of a feed-forward artificial neural network usually used in multispectral image classification. The network consists of one input layer, two hidden layers, and one output layer. The number of neurons in the input layer is determined by the number of multispectral bands used in the training and classification processes. The number of neurons in the output layer depends on the number of spectral classes. The number of neurons in hidden layers is usually determined by trial and error.

ces. There are several methods, that have been developed to reduce the computational complexity of this classifier (29, 30). The canonical analysis is to decompose the inverse covariance matrix using the Cholesky factorization and it saves a significant amount of time for the multiplications. The canonical analysis of the inverse covariance and the quadratic form range theorem were used to reduce the time complexity and improve the speed of the traditional maximum-likelihood classifier in the three stage ML classifier (30). It was reported that this algorithm is three times faster than the traditional ML classifier. The ML classifier was also implemented on parallel processing architectures to speed up the processing (31, 32).

FEATURE EXTRACTION AND SELECTION

Feature extraction is to find appropriate feature representation of an object in the image. The objective of feature extraction is to extract characteristic features and hence reduce the amount of data for performing the image classification. This procedure is critical in high-dimensional hyperspectral image classification. The features should carry a maximum amount of information in representing an object, i.e. *distinguishing features*. The features should also be able to differentiate among objects. Feature extraction is essential for any successful applications of image classification systems. Figure 12 shows an example of hyperspectral image classification systems (37).

Several feature extraction methods have been proposed in the pattern recognition and image classification literature. These methods include linear discriminant analysis (LDA), principal component analysis (PCA), factor analysis, projection pursuit, independent component analysis (ICA) and transform-based approaches (38, 39). Fourier transform (FD), Hadamard transform, Haar transform, and wavelet transform are some well-known transform approaches. These methods are well explained in some textbooks and review papers (38–40). As the PCA is frequently used in the remotely-sensed image analysis, the method is briefly sketched in the following.

Principal Component Analysis

Principal component analysis (PCA) is a multivariate statistical transformation technique, which is based on statistical properties of vector representations. PCA provides a systematic means of reducing the dimensionality of multispectral data. PCA has been used in image data compression and pattern classification. To perform the PCA, a transformation is applied to a set of multispectral image data. This will result in another uncorrelated data set, in which the axes of the original data set are rotated and translated. Hence, the coordinates and the pixel values are

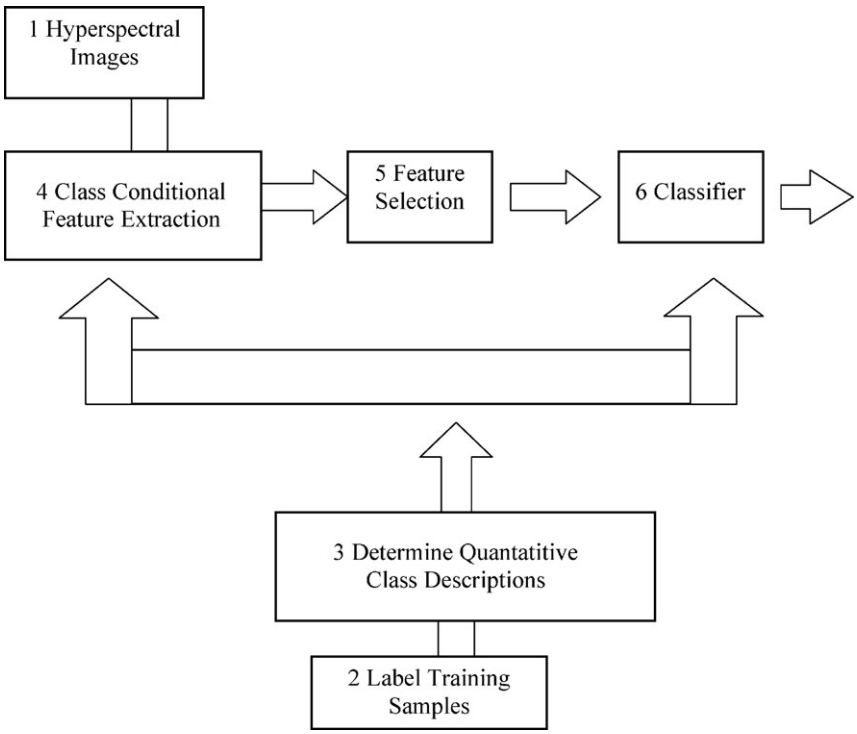

Figure 12. A schematic diagram for hyperspectral image classification.

changed. In other words, PCA is formed through a linear combination of the input bands. The new axes are parallel to the axes of the ellipse (in an n-dimensional histogram, a hyperellipsoid is formed if the distribution of each input band is normal or near normal) (33). If there is significant correlation between the original image set, most of the image information will be contained in the first few bands (principal components) after PCA transformation. These principal components are uncorrelated and independent. The first principal component (PC) contains the largest variation of the information contained in an image, which, spatially, represents the major axis of the elliptic spectral distribution of the data. The second PC contains the second largest variation of the information in an image. It represents the minor axis of the spectral ellipse, and it is orthogonal to the first PC. In n-dimensional representation, each successive PC is orthogonal to the previous PCs, describing less amount of variation that was not accounted for by the preseding PCs.

Mathematically, if $XT = (x_1, x_2, \ldots, x_n)$ is an N-dimensional random variable with mean vector M and covariance matrix C and if A is a matrix whose rows are formed from the eigenvectors of C, ordered so that the first row of A is the eigenvector corresponding to the largest eigenvalue, and the last row is the eigenvector corresponding to the smallest eigenvalue, then the PCA transformation is defined as:

$$Y = A(X - M) \tag{12}$$

where $Y = [y_1, y_2, \ldots, y_n]^T$, T is the transpose and each vector y_i is the i^{th} principal component.

As the principal components are independent of one another, a color combination of the first three components can be useful in providing maximum visual separability of image features. Therefore, principal components analysis has been used to generate a new set of data from multiple sources of data for multispectral image classification (34). Standarized principal components were also developed to improve the signal-to-noise ratio for the LANDSAT MSS data (35).

Hyperspectral sensor systems usually collect more than a hundred of spectral bands. These numerous bands may not be needed for a specific application. In addition, it has been shown that a lower dimensional feature space can improve the generalization capability of image classifiers (41). Besides PCA, many feature extraction methods have been proposed for hyperspectral images (42). Discriminant Analysis Feature Extraction (DAFE) and Nonparametric Weighted Feature Extraction (NWFE) are briefly described below.

Discriminant Analysis Feature Extraction (DAFE)

DAFE is often used for dimension reduction in classification problems. It is also called the parametric feature extraction method, since DAFE uses the mean vector and covariance matrix of each class. Usually within-class, between-class, and mixture scatter matrices are used to formulate the criteria of class separability. A within-class scatter matrix for L classes is expressed by (43):

$$S_w^{DA} = \sum_{i=1}^{L} P_i \Sigma_i = \sum_{i=1}^{L} P_i S_{wi}^{DA} \tag{13}$$

where P_i denotes the prior probability of class i, m_i is the class mean and Σ_i is the class covariance matrix. A

between-class scatter matrix is expressed as

$$S_b^{DA} = \sum P_i(m_i - m_o)(m_i - m_o)^T$$
$$= \sum_{i=1}^{L-1}\sum_{j=i+1}^{L} P_i P_j(m_i - m_j)(m_i - m_j)^T \quad (14)$$

where m_0 represents the expected vector of the mixture distribution and is given by

$$m_0 = \sum_{i=1}^{L} P_i m_i \quad (15)$$

The optimal features are determined by optimizing the Fisher criteria given by

$$J_{DAFE} = tr[(S_w^{DA})^{-1}(S_b^{DA})] \quad (16)$$

DAFE has been shown to be equivalent to finding the ML estimators of a Gaussian model, assuming that all class discrimination information resides in the transformed subspace and the within-class covariances are equal for all classes. The advantage of DAFE is that it is distribution-free but there are three major disadvantages in DAFE. One is that it works well only if the distributions of classes are normal-like distributions (43). When the distributions of classes are nonnormal-like or multi-modal mixture distributions, the performance of DAFE is not satisfactory. The second disadvantage of DAFE is the rank of the between-scatter matrix is number of classes (L) -1, so assuming sufficient observations and the rank of within-class scatter matrix is v, then only $\min(L-1, v)$ features can be extracted. We know (43) that unless a posterior probability function is specified, $L-1$ features are suboptimal in a Bayes sense, although they are optimal based on the chosen criterion. In real situations, the data distributions are often complicated and not normal-like, therefore only using $L-1$ features is not sufficient for much real data. The third limitation is that if the within-class covariance is singular, which often occurs in high dimensional problems, DAFE will have a poor performance on classification.

Foley-Sammon feature extraction and its extension can help to extract more than $L-1$ orthogonal features from n-dimensional space based on the following:

$$r_i = \max_r \frac{r^T S_b^{DA} r}{r^T S_w^{DA} r}, i = 1, 2, \ldots, n-1$$
$$\text{subject to } r_i^T S_w^{DA} r_j = 0, i \neq j$$

This third limitation can be relieved by using regularized covariance estimators in the estimating procedure of the within-class scatter matrix or by adding Singular Value Perturbation to the within-class scatter matrix to solve the generalized eigenvalue problem (37). Approximated pairwise accuracy criterion Linear Dimension Reduction (aPAC-LDR) can be seen as DAFE weighted contributions of individual class pairs according to the Euclidian distance of respective class means (37).

Nonparametric Weighted Feature Extraction (NWFE)

We know that the "local information" is important and useful for improving DAFE. The main idea of NWFE is putting different weights on every sample to compute the "weighted means" and defining new nonparametric between-class and within-class scatter matrices to obtain more than $L-1$ features (37). In NWFE, the nonparametric between-class scatter matrix for L classes is defined as

$$S_b^{NW} = \sum_{i=1}^{L} P_i \sum_{\substack{j=1 \\ j\neq i}}^{L} \sum_{l=1}^{N_i} \frac{\lambda_l^{(i,j)}}{N_i}(x_l^{(i)} - M_j(x_l^{(i)}))(x_l^{(i)} - M_j(x_l^{(i)}))^T \quad (17)$$

where $x_l^{(i)}$ refers to the l-th sample from class i, N_i is training sample size of class i, P_i denotes the prior probability of class i. The scatter matrix weight $\lambda_l^{(i,j)}$ is a function of $x_l^{(i)}$ and $M_j(x_l^{(i)})$, and defined as:

$$\lambda_l^{(i,j)} = \frac{dist(x_l^{(i)}, M_j(x_l^{(i)}))^{-1}}{\sum_{t=1}^{N_i} dist(x_t^{(i)}, M_j(x_t^{(i)}))^{-1}}, \quad (18)$$

where $dist(a, b)$ denotes the Euclidean distance from a to b. If the distance between $x_l^{(i)}$ and $M_j(x_l^{(i)})$ is small then its weight $\lambda_l^{(i,j)}$ will be close to 1; otherwise, $\lambda_l^{(i,j)}$ will be close to 0. The sum of the $\lambda_l^{(i,j)}$ for class i is 1. $M_j(x_l^{(i)})$ denotes the weighted mean of $x_l^{(i)}$ in class j and defined as:

$$M_j(x_l^{(i)}) = \sum_{k=1}^{N_j} w_{lk}^{(i,j)} x_k^{(j)}, \quad (19)$$

$$\text{where} \quad w_{lk}^{(i,j)} = \frac{dist(x_l^{(i)}, x_k^{(j)})^{-1}}{\sum_{k=1}^{N_i} dist(x_l^{(i)}, x_k^{(j)})^{-1}}. \quad (20)$$

The weight $w_{lk}^{(i,j)}$ for computing weighted means is a function of $x_l^{(i)}$ and $x_k^{(j)}$. If the distance between $x_l^{(i)}$ and $x_k^{(j)}$ is small then its weight $w_{lk}^{(i,j)}$ will be close to 1; otherwise, $w_{lk}^{(i,j)}$ will be close to 0. The sum of the $w_{lk}^{(i,j)}$ for $M_j(x_l^{(i)})$ is 1.

The nonparametric within-class scatter matrix is defined as

$$S_w^{NW} = \sum_{i=1}^{L} P_i \sum_{l=1}^{N_i} \frac{\lambda_l^{(i,j)}}{N_i}(x_l^{(i)} - M_i(x_l^{(i)}))(x_l^{(i)} - M_i(x_l^{(i)}))^T \quad (21)$$

NWFE proposes the "weighted mean" (eq. (28)) and using weighted between- and within-class vector to improve the nonparameteric discriminant analysis (NDA) (37). The extracted f features are the f eigenvectors with largest f eigenvalues of the following matrix:

$$(S_w^{NW})^{-1} S_b^{NW}$$

To reduce the effect of the cross products of within-class distances and prevent the singularity, some regularized techniques can be applied to within-class scatter matrix. The within-class scatter matrix is regularized by

$$S_w^{NW} = 0.5 S_w^{NW} + 0.5 diag(S_w^{NW}),$$

where diag(A) means the diagonal parts of matrix A.

The NWFE algorithm is sketched below

1. Compute the distances between each pair of sample points and form the distance matrix.
2. Compute $w_{lk}^{(i,j)}$ using the distance matrix
3. Use $w_{lk}^{(i,j)}$ to compute the weighted means $M_j(x_l^{(i)})$
4. Compute the scatter matrix weight $\lambda_l^{(i,j)}$

5. Compute S_b^{NW} and regularized S_w^{NW}
6. Compute the eigenvectors of $(S_w^{NW})^{-1}S_b^{NW}$ as extracted features

USE OF CLASSIFIED IMAGES IN GEOGRAPHICAL INFORMATION SYSTEMS

Remotely sensed data is the main source of data input to geographical information systems (GIS). Because computerized GIS are specialized data systems that manipulate virtually any geopreferenced digital information, classified images are usually the primary data sources for these systems for they are in digital format, thus readily usable. The main advantage of remotely sensed data is the digital format and their capability to cover large areas in short periods, which is very valuable when current and fast data are required for urgent matters. Although most recent GIS systems are vector oriented, most of these systems have the capabilities to convert data from vector to raster and vice versa. Classified images are usually thematic layers (e.g., vegetation types, soil types, geology water bodies, and land use/cover) that can be entered into a GIS for overlay analysis and modeling.

SOME IMAGE CLASSIFICATION SOFTWARE PACKAGES

Some companies that market software packages for image classification, which provide some techniques, covered in this chapter are listed below. This is not an exhaustive list of all the software products or vendors. A software package, MultiSpec, has been developed and maintained in Purdue University.

ERDAS	IMAGINE is the commercial product of ERDAS.
Intergraph Corp.	Image Analyst is the commercial product for remote sensing applications.
PCI	PCI Geomatics provides commercial products for the geospatial industry.
Purdue University	The Laboratory for Applications of Remotely Sensing (LARS) provides a software package called MultiSpec.
RSI	RADARSAT International (RSI) provides solutions based on satellite-derived data.

BIBLIOGRAPHY

1. B. Ferdinand G. Rochon *Principes et Méthodes*, vol. 1 of *Précis de Télédétection*, QuéCanada: Presses de l'Université du Qébec/AUPELF, 1992.
2. J. R. Jensen *Introductory Digital Image Processing: A Remote Sensing Perspective*, 2nd ed., Upper Saddle River, NJ: Prentice-Hall, 1996.
3. R. M. Haralick K. Shanmugam I. Dinstein Textural features for image classification, *IEEE Trans. Syst. Man Cybern.*, **SMC-3**: 610–621, 1973.
4. T. M. Lillesand R. W. Kiefer *Remote Sensing and Image Interpretation*, New York: Wiley, 1987.
5. A. K. Jain R. C. Dubes *Algorithms for Clustering Data*, Englewood Cliffs, NJ: Prentice-Hall, 1988.
6. W. Niblack *An Introduction to Digital Image Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1986.
7. MGE Advanced Imager (MAI), *User's Guide for theWindows NT Operating System*, Intergraph Corporation, Huntsville, AL,July 1995.
8. J. T. Tou R. C. Gonzalez *Pattern Recognition Principles*, Reading, MA: Addison-Wesley, 1974.
9. Q. Zhang Q. R. Wang R. Boyle A clustering algorithm for datasets with a large number of classes, *Pattern Recog.*, **24** (4): 331–340, 1991.
10. K. Fitzpatrick-Lins The Accuracy of selected land use and land cover maps at scales of 1:250000 and 1:100000, *J. Res.*, U.S.G.S., **6**: 169–173, 1980.
11. D. L. Verbyla *Satellite Remote Sensing of Natural Resources*, New York: Lewis, 1995.
12. R. P. Lippmann An introduction to computing with neural nets, *IEEE ASSP Mag.*, **27** (11): 4–22, 1987.
13. J. Hertz A. Krogh R. G. Palmer *Introduction to the Theory of Neural Computation*, Reading, MA: Addison-Wesley, 1991.
14. R. Schalkoff *Pattern Recognition: Statistical, Structural and Neural Approaches*, New York: Wiley, 1992.
15. C. C. Hung Competitive learning networks for unsupervised training, *Int. J. Remote Sensing*, **14**: 2411–2415, 1993.
16. P. H. Swain S. M. Davis *Remote Sensing: The Quantitative Approach*, New York: McGraw-Hill, 1978.
17. C. C. Hung T. Coleman P. Scheunders The genetic algorithm approach and *K*-means clustering: Their role in unsupervised training in image classification, *Proc. Int. Conf. Comput. Graphics and Imaging*, Halifax, Canada, June, 1998.
18. D. E. Goldberg *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison-Wesley, 1989.
19. J. Zhou D. L. Civco Using genetic learning neural networks for spatial decision making in GIS, *Photogrammetric Eng. Remote Sensing*, **62**: 1287–1295, 1996.
20. P. Scheunders A generic c-means clustering algorithm applied to color image quantization, *Pattern Recognition*, **30** (6): 859–866, 1997.
21. J. G. Moik *Digital Processing of Remotely Sensed Images*, Washington, D.C.: U.S. Government Printing Office, 1980.
22. P. M. Mather *Computer Processing of Remotely-Sensed Images: An Introduction*, New York: Wiley, 1987.
23. J. Kittler J. Foglein Contextual classification of multispecral pixel data, *Image Vision Comput.*, **2**: 13–29, 1984.
24. K. M. S. Sharma A. Sarkar A modified contextual classification technique for remote sensing data, *Photogrammetric Eng. Remote Sensing*, **64**: 273–280, 1998.
25. C. C. Hung A multispectral image classification algorithm based on contextual information, *Proc. 13th Ann. Int. Conf.—Comput. Science Group*, The Association of Management, Vancouver, B.C., Canada, August, 1995.
26. R. P. Lippmann Pattern classification using neural networks, *IEEE Comm. Mag.*, **27** (11): 47–63, 1989.
27. P. D. Heermann N. Khazenie Classification of multispectral remote sensing data using a back-propagation neural network, *IEEE Trans. Geosci. Remote Sensing*, **30**: 81–88, 1992.
28. M. K. Arora G. M. Foody Log-linear modelling of the evaluation of the variables affecting the accuracy of probabilistic, fuzzy and neural network classification, *Int. J. Remote Sensing*, **18**: 785–798, 1997.
29. W. Eppler Canonical analysis for increased classification speed and channel selection, *IEEE Trans. Geosci. Remote Sensing*, **14**: 26–33, 1976.
30. N. B. Venkateswarlu P. S. V. K. Raju Three stage ML classifier, *Pattern Recog.*, **24** (11): 1113–1116, 1991.
31. J. J. Settle S. A. Briggs Fast maximum likelihood classification of remotely sensed imagery, *Int. J. Remote Sensing*, **8**: 723–734, 1987.
32. K. S. Fu *Spectral Computer Architectures for Pattern Recognition*, Boca Raton, FL: CRC Press, 1987.
33. Erdas *Imagine: Field Guide*, 3rd ed., Atlanta, Georgia: Erdas, 1995.
34. M. Shimura T. Imai Nonsupervised classification using the principal component, *Pattern Recog.*, **5** (4): 353–363, 1973.
35. A. Singh Standardized principal components, *Int. J. Remote Sensing*, **6**: 883–896, 1985.
36. J. S. Lee Digital image smoothing and the Sigma filter, Computre Vision, Graphics, and Image Processing, **49** (1): 55–64.
37. B.-C. Kuo D. A. Landgrebe Nonparametric weighted feature extraction for classification, *IEEE Trans. Geosci. Remote Sensing*, **42** (5): 1096–1105, 2004.
38. S. Theodoridis K. Koutroumbas *Pattern Recognition*(2nd Ed.), Academic Press, 2003.
39. R. O. Duda P. E. Hart D. G. Stork *Pattern Classification*(2nd Ed.), John Wiley & Sons, Inc. 2001.
40. A. K. Jain R. P. W. Duin J. Mao Statistical pattern recognition: a review, *IEEE Trans.on Pattern Analysis and Machine Intelligence*, **22** (1): 4–37, 2000.
41. B. Scholkopf A. Smola *Learning with Kernels*, MIT press, Cambridge, MA, 2002.
42. D. A. Landgrebe *Signal Theory Methods in Multispectral Remote Sensing*, John Wiley & Sons, Inc. 2003.
43. K. Fukunaga *Introduction to Statistical Pattern Recognition*, Academic Press, 1990.
44. L. Breiman Bagging predictors, *Machine Learning Journal*, **24** (2): 123–140, 1996.
45. J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas," On Combining Classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.**20**, no. 3, 1998, 226–239.
46. L. I. KunchevaCombining Pattern Classifiers: Methods and Algoritms, John Wiley & Sons, Inc. 2004.
47. E. Bauer R. KohaviAn empirical comparison of voting classification algorithms: bagging, boosting, and variants, Machine learning,vv, 1–38, 1998.
48. Bor-Chen Kuo Chia-Hao Pai Tian-Wei Sheu Guey-Shya Chen Hyperspectral Data Classification Using Classifier Overproduction and Fusion Strategies, *IEEE IGARSS*, 2004

49. L. Xu A. Krzyzak C. Suen Methods of combining multiple classifiers and their applications to handwriting recognition, *IEEE Trans. on SMC*, **22**: 418–435, 1992.

CHIH-CHENG HUNG
BOR-CHEN KUO
AHMED FAHSI
TOMMY L. COLEMAN
Southern Polytechnic State University, Marietta, GA
National Taichung University, Taiching, Taiwain
Applied Analysis Inc., Billerica, MA
Alabama A&M University, Normal, AL