A Synergistic Automatic Clustering Technique (SYNERACT) for Multispectral Image Analysis

Kai-Yi Huang

Abstract

The Iterative Self-Organizing Data Analysis Technique (ISODATA) has been widely used in unsupervised and supervised classification. However, ISODATA suffers from several limitations. The user often spends much analyst time on specifying input parameters by trial and error, particularly initial cluster centers. Of more importance, an inappropriate choice of initial clusters may cause poor classification results. ISODATA is computationally intensive because of its iterative process. This study aimed to develop a synergistic automatic clustering technique (SYNERACT) that combined the hierarchical descending and ISODATA clustering procedures to avoid those limitations. The two methods were compared using multispectral digitized video images. An inappropriate choice of initial seeds for ISODATA was shown to reduce accuracies significantly. In contrast, SYNERACT was capable of determining the suitable locations for the initial clusters automatically from the data, thereby avoiding those limitations. Owing to this capability, SYNERACT was not so heavily dependent on the iterative process as was ISODATA, and thus was much faster than ISODATA. SYNERACT also matched ISODATA in accuracy. Accordingly, SYNERACT could serve as an alternative to ISODATA for multispectral image analysis.

Introduction

Clustering used for unsupervised classification is one of the most often used methods for extracting information from remotely sensed data of the Earth (Jensen, 1996). Clustering can also be used to determine the natural spectral groupings present in a data set. Thus, some of the unique classes, but with very small areal extent that might not be initially apparent to the analyst applying a supervised classifier, can be recognized as distinct units (Campbell, 1987; Lillesand and Kiefer, 2000). These classes might otherwise be incorporated into other classes, generating error and imprecision throughout the entire classification. There are two main families of clustering methods: Iterative Self-Organizing Data Analysis Technique (ISODATA) clustering and hierarchical clustering approaches (Viovy, 2000).

ISODATA (or K-means) is a widely used clustering method to partition the image data in the multispectral space into a number of spectral classes (Wharton and Turner, 1981; Jensen, 1996). It requires that the user manually specify various parameters to control the clustering process. These parameters are varied and the programs are run in an iterative fashion until the output set of clusters meets the analyst's criteria. The general principle of this method is to minimize an objective function (e.g., the sum-of-squared errors) by manipulating a set of cluster centers, whose number has to be specified by the analyst and whose initial locations are generated randomly (Viovy, 2000). More details about the clustering algorithms for ISODATA can be obtained from Swain (1978), Richards (1993), and Jensen (1996).

This type of clustering algorithm, however, suffers from two major limitations. The first limitation associated with ISODATA is the requirement of some *a priori* knowledge about the structure of the data under consideration, such as *a priori* knowledge regarding the range of optimal values for the clustering parameters—i.e., splitting and merging thresholds (Ball and Hall, 1967). Hence, the user often spends much analyst time to determine the optimal values of these parameters by trial and error. Specifically, ISODATA requires that the user specify the number of clusters, along with their initial positions, in advance (more details stated below). In practice, the actual or optimum number of clusters to choose will not be known. The strategy is to guess high and to consolidate any redundant clusters after the iterative part of the algorithm terminates, or at intervening iterations (Swain, 1978).

ISODATA requires the user to specify the initial locations of the cluster centers (also called "seeds" or "cluster means") through an educated guess simultaneously. Clustering begins with a set of arbitrarily selected pixels as cluster centers with the exception that no two may be identical (Swain, 1978). Although initial cluster centers are selected at random to assure that the analyst cannot influence the clustering, some ISODATA algorithms provide their own initial seeds (Campbell, 1987). Indeed, initial seeds are often chosen evenly spaced along a diagonal axis in multidimensional feature space because no guidance is available in general (Richards, 1993). This is a line from the origin to the point corresponding to the maximum digital number in each spectral component (e.g., 255, 255, 255 for SPOT multispectral image data). Other similar procedures used in ISODATA have been proposed in Fromm and Northouse (1976), Jensen (1996), and ERDAS (1997). As pointed out by Richards (1993), the choice of the initial seeds is not crucial to classification accuracy, but it will affect the time it takes to reach an acceptable accuracy. It does not matter where the initial cluster centers are located, as long as enough number of iterations (or processing time) is allowed (ERDAS, 1997). However, few studies have investigated the important but apparently neglected problem about how the choice of the initial seeds may adversely affect final classification results when ISODATA is used.

The second limitation is that ISODATA is computationally intensive when processing large data sets. Furthermore, ISODATA tends to suffer from performance degradation as the number of bands, the number of pixels, or the number of clusters increases (Richards, 1993; Viovy, 2000). This problem will

Department of Forestry, National Chung-Hsing University, 250 Kuo-Kwang Road, Taichung, Taiwan 402 (kyhuang@dragon.nchu.edu.tw).

Photogrammetric Engineering & Remote Sensing
Vol. 68, No. 1, January 2002, pp. 33–40.

0099-1112/02/6800–033$3.00/0

become worse with new-generation instruments—imaging spectrometers (or hyperspectral scanners)—such as AVIRIS (Airborne Visible Infrared Imaging Spectrometer) that produce a very large number of simultaneous measurements in every image pixel.

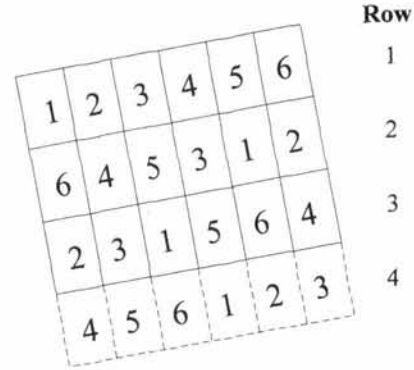
The second type of clustering method is hierarchical clustering. For the agglomerative hierarchical clustering (or hierarchical ascending) algorithm (Ward, 1963; Wishart, 1969), the amount of storage required for this approach is proportional to the number of pixels squared. Therefore, this method is not suited for classification of large sets of remotely sensed data (Wharton and Turner, 1981). For the divisive hierarchical clustering (or hierarchical descending) algorithm, it has been developed such that the data are initialized as a single cluster that is progressively subdivided. The procedure is more computationally intensive and is rarely used in remote sensing applications, because usually a large number of pixels are involved (Richards, 1993). On the other hand, Viovy (2000) has shown that the hierarchical descending method developed in his study is very efficient and does not show significant performance degradation with increasing dimension or increasing number of pixels.

This study attempted to accomplish two specific objectives using two-date video image data. The first objective was to show that the choice of initial clusters required by ISODATA significantly affected final classification results, which was contrary to the ideas suggested by Richards (1993) and ERDAS (1997). The second one was to develop a synergistic automatic clustering technique (SYNERACT) based on the principles of hyperplane, iterative optimization clustering (i.e., ISODATA or K-means), and binary tree. SYNERACT required a minimum of user input and had the ability to determine the suitable locations for the initial clusters automatically from the data set itself. Because the approach did not heavily rely on the iterative process, it was relatively efficient when processing remotely sensed data with a large number of bands or pixels. SYNERACT was thus well suited to be a surrogate for ISODATA for remote sensing applications, which was contrary to the ideas suggested by Richards (1993).

Study Area and Materials

The study area was located near Weslaco in Hidalgo County, Texas. It was a completely randomized block-designed field experiment consisting of plots of the following surface features: (1) cotton, (2) cantaloupe, (3) sorghum, (4) johnsongrass, (5) pigweed, and (6) bare soil (Figure 1). Each of the 24 plots (six treatments and four replications) measured 7.11 m by 9.14 m, making the total site dimension 42.67 m by 36.56 m (Richardson *et al.*, 1985). The fourth row (drawn with a dashed line) was excluded from the study due to damage to this portion of the video data file.

The two-date video image data were acquired on 31 May and 24 July 1983 near noon on moderately sunny days from an altitude of 900 m. The video imaging system used to collect data for the study is described in detail in Richardson *et al.* (1985). The filters and camera aperture settings used for this system are shown in Table 1. Video images were digitized to a quantization level of 256. Eight data matrices of 512 by 512 were created. The two 4-band video images were spatially registered using CONTROL POINT and WARP modules of the International Imaging System (I^2S) software package (Richardson *et al.*, 1985). Spectral bands 1 to 4 were acquired on 24 July; these same spectral bands (5 to 8) were acquired on 31 May. The wavelengths and nominal spectral locations of these bands are also shown in Table 1. Multiple-date image radiometric normalization using regression (Jensen, 1996) was performed to correct the data set used in the study, because atmospheric effects likely affected pixel brightness values of the two-date video image data (Eckhardt *et al.*, 1990).



1: Cotton
2: Soil
3: Johnsongrass
4: Cantaloupe
5: Pigweed
6: Sorghum

Figure 1. Plot identification map of the study area (not drawn to scale).

TABLE 1. THE FILTERS, WAVEBANDS, AND CAMERA SETTINGS OF THE VIDEO IMAGING SYSTEM (RICHARDSON *ET AL.*, 1985)

Filter	Wavelengths (nm)	Aperture	Band Number
Blue	420–430	f1.8	1, 5
Yellow-Green	520–550	f2.8	3, 7
Red	640–570	f1.8	2, 6
Near Infrared	850–890	f8.0	4, 8

Accuracy assessment was performed over 18 plots from row 1 to row 3 in the experimental field with the exception of row 4, as shown in Figure 1. A digital ground truth mask was used to calculate classification accuracies of the two clustering algorithms. Border areas of mask between plots were eliminated in order that incomplete pixels or mixed pixels were excluded from accuracy assessment.

Method and Rationale

SYNERACT combines the concepts of hyperplane, binary tree, and iterative optimization (K-means) clustering. The hyperplane divides a cluster into two clusters of smaller size and computes their means. The binary tree is a useful data structure that can store the clusters successively generated from each split. The first concept and the logic flow of SYNERACT are described in more detail as follows. The binary tree and related routines that build, traverse, and apply binary trees are described in Tenenbaum and Augenstein (1981). The third one can be obtained from Richards (1993).

Definition of a Hyperplane

To appreciate the development of SYNERACT, it is required to understand the concept of hyperplane. The family of linear discriminant functions (Nilsson, 1965) can be expressed in the form as follows:

$$F(\mathbf{X}) = W_1 * X_1 + W_2 * X_2 + \ldots + W_n * X_n + W_{n+1}, \quad (1)$$

where $W_1, W_2, \ldots, W_n, W_{n+1}$ are weighting coefficients. F is a linear function of the components of an augmented column vector $\mathbf{X}$.

A simple linear separation is performed by a linear discriminant function that partitions a feature space into two regions. The linear discriminant function can be viewed as a

separating surface in which the simplest form is a hyperplane (Nilsson, 1965). A hyperplane partitions a feature space into two regions defined as

$$F(\mathbf{X}) = \mathbf{W} \cdot \mathbf{X} > 0 \text{ and } F(\mathbf{X}) = \mathbf{W} \cdot \mathbf{X} \leq 0, \qquad (2)$$

where $\mathbf{W} = [W_1, W_2, \ldots, W_n, W_{n+1}]$, $\mathbf{X}^T = [X_1, X_2, \ldots, X_n, 1]$, and n is the dimension of a feature space.

Assume that there is a cluster in a feature space. A weight vector **W** is viewed to implement a linear separating surface (hyperplane) to divide a cluster in a feature space into two clusters of smaller size (children). The augmented pixel vectors of the one child-cluster have a positive dot product value with **W**, while the other child-cluster consists of pixels (lying on the other side of **W**) that have a zero or negative dot product value with **W**. The former is categorized as S_1 and the latter is categorized as S_2. Centers of the two sets are computed from the pixels in the two sets, respectively, which will be used as initial seeds for the K-means procedure. The sets of parent cluster (S) and two child-clusters (S_1 and S_2) can be defined as

$$S_1 = \{\mathbf{X} \in S | \mathbf{W} \cdot \mathbf{X} > 0\};\ S_2 = \{\mathbf{X} \in S | \mathbf{W} \cdot \mathbf{X} \leq 0\};\ S = S_1 \cup S_2.$$

Algorithm for Generating Hyperplane and Initial Clusters

The process of determining a weight vector (i.e., a straight line) and two initial cluster centers is illustrated by a hypothetical case in a two-dimensional spectral space shown in Figure 2. In general, the spatial distribution of image pixels in a two-dimensional spectral space can be thought of as a two-phase mosaic, in which patches containing high-density image pixels alternate with gaps containing low-density or no image pixels. At each node on every level of a binary tree, the process will find a line passing through the gap between two main patches and will compute two cluster (patch) means for the subsequent use.

An augmented pixel vector is defined as

$$\mathbf{X} = [\mathbf{V}^T, 1]^T, \quad \text{where} \quad \mathbf{V}^T = [V_1, V_2, \ldots, V_n]. \qquad (3)$$

Let the pixels be augmented and expressed by the general column vector **X**. One pixel **P** is chosen from this cluster so that $|\mathbf{P} - \mathbf{C}| > |\mathbf{X} - \mathbf{C}|$ for all **X**, where **C** is an arbitrarily chosen position vector in a feature space. Let **C** be the grand mean vector computed from all of the pixels in a single cluster. Under this condition, **P** and **C** define a line from the grand mean **C** to the farthest pixel **P**. Assume that there is a single cluster in a two-dimensional spectral space consisting of two spectrally distinct clusters of smaller size represented by **U** and **V**, respectively, so that $\{\mathbf{X}\} = \{\mathbf{U}\} \cup \{\mathbf{V}\}$, $\mathbf{P} \in \{\mathbf{X}\}$. Assume further that **P** comes from $\{\mathbf{U}\}$ and is denoted by $\mathbf{U}^*$, $\mathbf{U}^* = \mathbf{P}$. The next step is to find all of **V** so that $(\mathbf{V} - \mathbf{C}) \cdot (\mathbf{U}^* - \mathbf{C}) \leq 0$, $\{\mathbf{V}\} \subset \{\mathbf{X}\}$. Find $\mathbf{V}_1{}^*$ or $\mathbf{V}_2{}^*$ from $\{\mathbf{V}\}$ so that $\mathbf{V}_1{}^*$ must satisfy $|(\mathbf{V}_1{}^* - \mathbf{C}) \cdot (\mathbf{U}^* - \mathbf{C})| = 0$ and $\mathbf{V}_2{}^*$ must satisfy $|(\mathbf{V}_2{}^* - \mathbf{C}) \cdot (\mathbf{U}^* - \mathbf{C})| < |(\mathbf{V} - \mathbf{C}) \cdot (\mathbf{U}^* - \mathbf{C})|$.

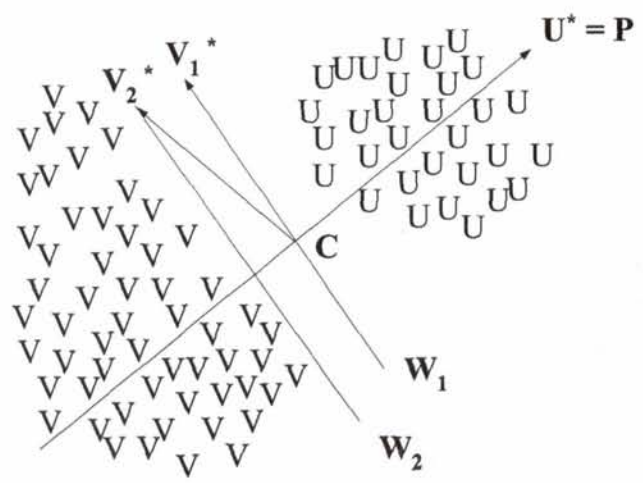


Figure 2. A two-dimensional, hypothetical case with two clusters illustrating how a hyperplane is generated.

Because the brightness value of each pixel in remotely sensed data is recorded as an integer, $\mathbf{V}_1{}^*$ cannot always be found in all cases. Thus $\mathbf{V}_2{}^*$, having the biggest negative dot product value, is used as an alternative to $\mathbf{V}_1{}^*$. Lines perpendicular to $(\mathbf{U}^* - \mathbf{C})$ and passing through $\mathbf{V}_1{}^*$ or $\mathbf{V}_2{}^*$ will be hyperplanes separating $\{\mathbf{U}\}$ and $\{\mathbf{V}\}$, as shown in Figure 2. The means of the two sets (S_1 and S_2) are computed simultaneously and will be used as initial seeds for the subsequent K-means procedure. According to the concept of the single-sided decision surface proposed by Lee and Richards (1984), these two hyper-planes are defined by equations shown as follows:

$$\mathbf{W}_1 = [(\mathbf{U}^* - \mathbf{C})^T, -\mathbf{V}_1{}^* \cdot (\mathbf{U}^* - \mathbf{C})]^T \qquad (4)$$

$$\mathbf{W}_2 = [(\mathbf{U}^* - \mathbf{C})^T, -\mathbf{V}_2{}^* \cdot (\mathbf{U}^* - \mathbf{C})]^T \qquad (5)$$

Test for Relevance of Splitting

SYNERACT will split each cluster formed at the previous separation into two clusters of smaller size. The splitting process is theoretically continued until there is only one pixel in each cluster. Therefore, this process must be controlled by two input parameters, including the maximum number of clusters to be considered (C_{max}) and the minimum percentage of pixels allowed in a single cluster (P%). Each split is tested for these two parameters *a posteriori* in order that a homogeneous cluster will not be split inappropriately. Because each cluster is the basis for an information class, C_{max} will become the maximum number of classes to be formed. Some clusters with percentage of pixels less than P% can be eliminated, leaving fewer than C_{max} clusters. The pixels in these discarded clusters will be reassigned to their original parent cluster.

This study will also attempt to use two splitting parameters proposed by Viovy (2000) for testing the relevance of a split. The first parameter is defined as the relative proportion computed from pixels assigned to parent cluster divided by pixels assigned to child-clusters based on the nearest-neighbor criterion. The second parameter is that the total variance of the two child-clusters should be smaller than the variance of the parent cluster; the ratio of variance is equal to the former divided by the latter. Thus, the split is relevant if both the proportion is less than a threshold (α) and the ratio of variance is less than a threshold (β) specified by the user. If one of the two tests described above fails, then the split is rejected. The parent cluster that is said to be stabilized is left out of the subsequent split. The thresholds of two splitting parameters were determined from a set of experiments done on simulated data sets, which were obtained by randomly generated pixels. The threshold of the first parameter $\alpha = 20$ percent and the second $\beta = 0.95$ were determined by Viovy (2000), but the user can specify them.

The Basic Algorithm of SYNERACT

The algorithm is shown in Figure 3 and proceeds in the following steps:

(1) Specify the maximum number of clusters to be considered (C_{max}) and the minimum percentage of pixels allowed in a single cluster (P%).

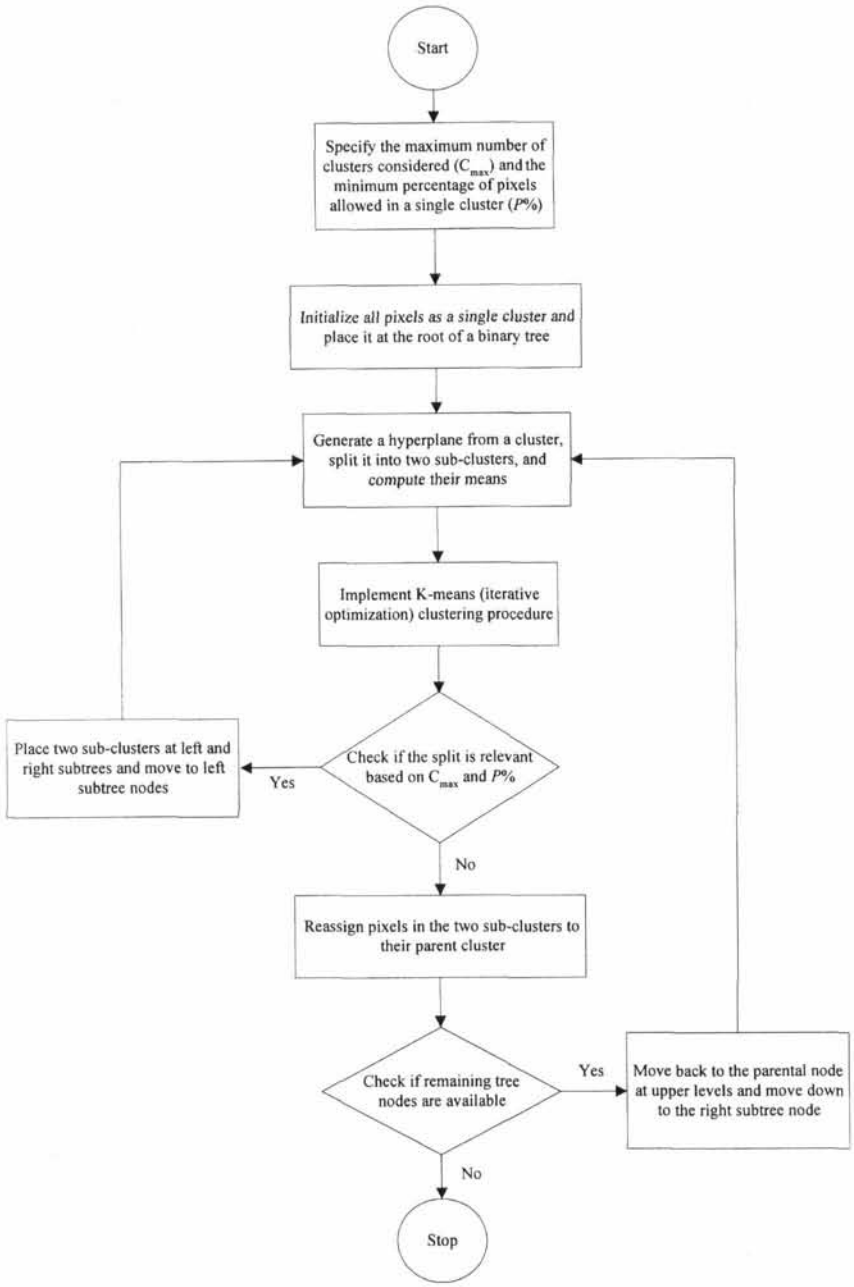


Figure 3. The flowchart showing how SYNERACT is implemented.

(2) Initialize all of the image pixels as a single cluster and place it at the root of a binary tree.
(3) Generate a hyperplane from the cluster, split the cluster into two smaller child-clusters, and compute their means that are used as initial seeds for the K-means procedure.
(4) The K-means procedure is implemented and repeated until the means of two child-clusters remain unchanged between two successive iterations.
(5) Check if the split is relevant; that is, that the number of clusters is less than C_{max} and that the percentage of pixels in a cluster is greater than P%. If so, then place two child-clusters at the left and right subtree nodes, move down to the left subtree node, and return to Step (3). Otherwise, reassign pixels in the two child-clusters to their parent cluster.
(6) Check if any remaining tree nodes are available. If available, move back to the parental (or ancestral) node at upper levels, move down to the right subtree node, and return to Step (3). Otherwise, terminate the process.

An Illustration of SYNERACT

A hypothetical case with three clusters is shown in Figure 4 to illustrate how SYNERACT works. As shown in Figure 4a, there are three clusters denoted by **1**, **2**, and **3**, respectively. These three clusters are initialized as a single cluster of bigger size, and this big cluster is placed at the root **R** of a binary tree, as shown in Figure 4b. Assume that a hyperplane $\mathbf{W}_1$ is first generated from the cluster at the root **R**, and it is subdivided into two child-clusters of by $\mathbf{W}_1$ (Figure 4a). One child-cluster (denoted by **B**) is composed of cluster **3**, and the other child-cluster (denoted by **A**) is composed of clusters **1** and **2**. The K-means procedure

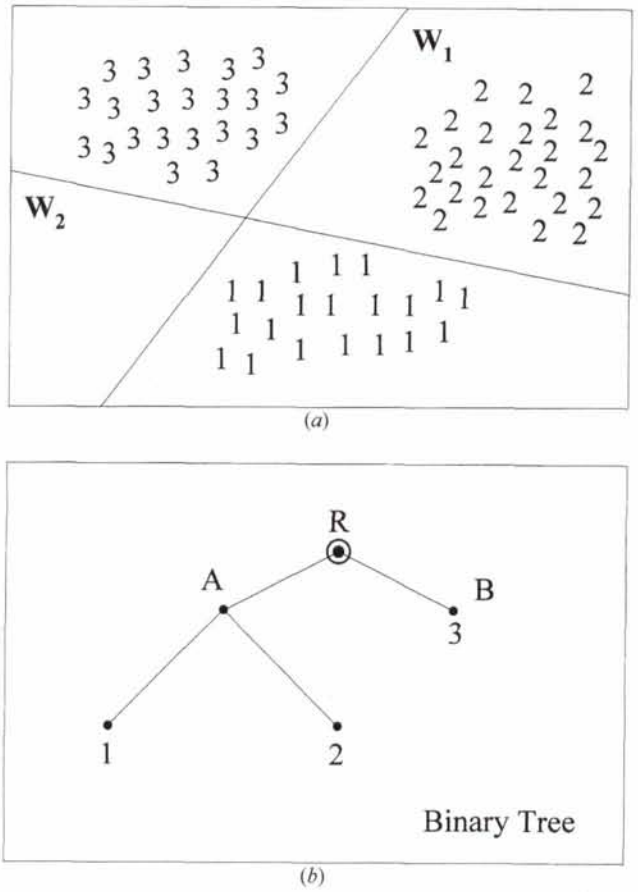


Figure 4. A two-dimensional, hypothetical case with three clusters illustrating how SYNERACT works.

is then implemented and repeated until the means remain unchanged between two successive iterations. At this point, check if the split is relevant according to the criteria: C_{max} and P% mentioned above. Because the split is relevant in the case, place these two child-clusters at the left and right subtree nodes **A** and **B** of the root **R**, move down to node **A**, and return to Step (3). A hyperplane $\mathbf{W}_2$ is generated from cluster **A**, and cluster **A** is split into clusters **1** and **2**. Again check the relevance of the split. Because the split is relevant, place them at the left and right subtree of the node **A**, move down to node **1**, and return to Step (3). A hyperplane is generated from cluster **1**, and the cluster is split into two sub-clusters. Because the split is irrelevant, reassign them to cluster **1**. Check if remaining nodes are available. Because nodes **2** and **3** are available, move upward to node **A**, move down to its right subtree node—node **2**—and return to Step (3). This process continues until cluster **3** is no longer split and no remaining nodes are available.

Results and Discussion

There were eight bands in the two-date video image data used for testing the clustering algorithms. The total number of band combinations (i.e., ${}_8C_8 + \ldots + {}_8C_1$) was equal to 255. Thus, it was impractical to test the algorithms for all of the band combinations. Because the two algorithms were developed based on Euclidean distance, this study applied the program module Separability/Euclidean Distance Measure in ERDAS Imagine 8.3.1 software to determine the appropriate bands for classification. As a result, the band combination of 2, 3, 4, and 8 with best separability was chosen for the test.

Influences of Initial Seed Assignment on Accuracy

The influence of picking initial cluster centers beforehand on classification results apparently has been neglected in most earlier studies about ISODATA, as mentioned previously. This limitation was the major concern in this study. Table 2 presents the classification results of ISODATA using six sets of initial cluster centers randomly generated from the computer program written by the author. Overall accuracies varied from the lowest at 65 percent to the highest at 85 percent; a variation of 20 percent in the accuracy was obvious. Clearly, the choice of the initial seeds was crucial to classification accuracies, because it evidently affected the results when ISODATA was used. This outcome was contrary to the ideas proposed by Richards (1993) and ERDAS (1997) mentioned previously. Indeed, a completely random assignment of initial seeds has been scarcely adopted in commercial software packages.

Note that the accuracies of johnsongrass from set 2 to set 6 were relatively low or even equal to zero; in contrast, the accuracies of pigweed and sorghum were relatively high. Although johnsongrass, pigweed, and sorghum were spectrally similar land-cover types, the band combination of 2, 3, 4, and 8 in the two-date video image data had the ability to differentiate three vegetation types according to their spectral characteristics and the results from accuracy assessment. Therefore, large variations in the accuracy were primarily due to the method of completely random seed assignment. The johnsongrass plots consisted of two major spectral classes. One spectral class co-occurred in pigweed plots and the other co-occurred in sorghum plots (i.e., one-to-many relationship). According to the rules of unsupervised classification for a one-to-many relationship stated in Lillesand and Kiefer (2000), the pixels in these two spectral classes were assigned to pigweed and sorghum, respectively, rather than to johnsongrass. Consequently, this led to poor accuracies for johnsongrass.

As shown in Figure 5, a two-dimensional, hypothetical case with two spectral clusters is helpful to illustrate this situation. Assume these two clusters are formed from two distinct but spectrally similar land-cover types. They are denoted by **A** and **B**, respectively. Assume two sets of initial seeds are generated randomly; the initial seeds in the first set are denoted by **1**, **2**, and those in the second are denoted by **3**, **4**. For the first set of initial seeds, the pixels in cluster **A** are assigned to seed **1** and the pixels in cluster **B** are assigned to seed **2** according to the nearest-neighbor criterion. By contrast, pixel **Ax** is the nearest one to seed **4** in cluster **A** and pixel **Bx** is the nearest one to seed **4** in cluster **B** for the second set of initial seeds. Because of $D_{4Ax} > D_{3Ax}$ and $D_{4Bx} > D_{3Bx}$, pixels **Ax** and **Bx** are assigned to seed **3**. Thus, all of the remaining pixels in clusters **A** and **B** are assigned to seed **3** rather than to seed **4**. The second set of initial seeds will result in the incorporation of clusters **A** and **B**, which in turn will lead to a significant loss in accuracy of the merged class (either cluster **A** or **B**).

This study also applied the method of initial seed assignment proposed by ERDAS (1997) to pick the locations of the

TABLE 2. CLASSIFICATION ACCURACIES OF ISODATA USING SIX SETS OF RANDOMLY GENERATED INITIAL SEEDS

Set of Initial Seeds Land-Cover Type	1 (%)	2 (%)	3 (%)	4 (%)	5 (%)	6 (%)
Cotton	90	92	91	92	69	67
Soil	98	98	99	99	97	98
Johnsongrass	37	61	0	0	0	0
Cantaloupe	96	67	94	65	94	96
Pigweed	97	86	98	99	97	69
Sorghum	93	95	96	97	65	62
Overall Accuracy (%)	85	83	80	75	70	65

The band combination of 2, 3, 4, and 8 was chosen for this test.

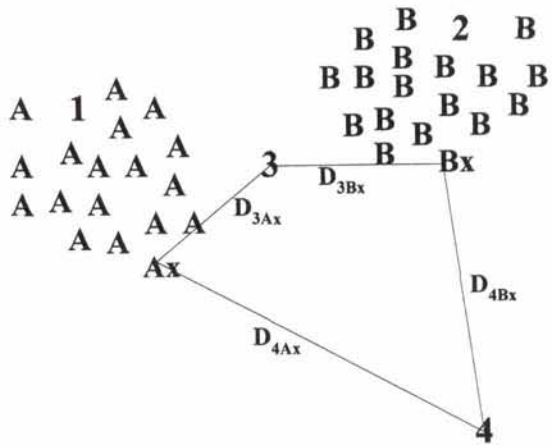


Figure 5. A two-dimensional, hypothetical case with two spectrally similar but distinct clusters illustrating the incorporation of the two clusters caused by an inappropriate initial seed assignment, which in turn will lead to poor classification results.

initial seeds required by ISODATA. Namely, the initial seeds were evenly distributed in n-dimensional spectral space between the points at the coordinates $(\boldsymbol{\mu}_1 - k^* \boldsymbol{\sigma}_1, ..., \boldsymbol{\mu}_n - k^* \boldsymbol{\sigma}_n)$ and $(\boldsymbol{\mu}_1 + k^* \boldsymbol{\sigma}_1, ..., \boldsymbol{\mu}_n + \mathbf{k}^* \boldsymbol{\sigma}_n)$, where $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\sigma}$ is the standard deviation, and k is the number of standard deviations. Table 3 presents the classification results generated from ISODATA using five sets of the initial seeds placed along vectors defined by $\boldsymbol{\mu} \pm 1^* \boldsymbol{\sigma}, ..., \boldsymbol{\mu} \pm 5^* \boldsymbol{\sigma}$, respectively. Overall accuracies varied from the highest at 93 percent for $\boldsymbol{\mu} \pm 1^* \boldsymbol{\sigma}$ to the lowest at 75 percent for $\boldsymbol{\mu} \pm 4^* \boldsymbol{\sigma}$ and $\boldsymbol{\mu} \pm 5^* \boldsymbol{\sigma}$ in this case. Clearly, ERDAS's ISODATA program still suffered from this limitation. Furthermore, the researcher did spend much "analyst time" on determining the optimal number of standard deviations from one to five in the study. In contrast, SYNERACT eliminated the need for picking the initial clusters in advance, thereby avoiding the limitation and saving much analyst time. Hence, this was a chief benefit of SYNERACT.

Influences of Initial Seed Assignment on Processing Time

Table 4 presents the lengths of processing time of ISODATA using the two methods of initial seed assignment. ERDAS's method took a much longer time than did the method of completely random seed assignment; the average length of the former was about 3.3 (67/20) times longer than that of the latter, although the former performed much better in accuracies than did the latter. As explained in the hypothetical case shown in Figure 4, inappropriate random seed assignments caused the incorporation of distinct but spectrally similar ground-cover types in the clustering process, and in turn resulted in a great loss in accuracy of the merged class (i.e., johnsongrass). The initial seeds that could not form clusters were eliminated after the first iteration, and the actual number of clusters that remained in the iterative process was less than the number of clusters initially specified. In general, the computing time of ISODATA is positively related to the number of clusters because, at each iterative step, all of the pixels in the entire data set must be checked against every cluster center (Richards, 1993). Therefore, the cascading effect of inappropriate random seed assignments frequently resulted in the acceleration, instead of prolongation, of convergence, thereby greatly reducing processing time spent on clustering.

Table 3. Classification Accuracies of ISODATA Using ERDAS's Method for Initial Cluster Allocation

$k\sigma$[1] Land-Cover Type	1σ (%)	2σ (%)	3σ (%)	4σ (%)	5σ (%)
Cotton	91	85	89	94	92
Soil	97	99	97	99	98
Johnsongrass	92	59	34	32	0
Cantaloupe	96	95	95	98	95
Pigweed	87	98	66	64	98
Sorghum	96	91	96	63	67
Overall Accuracy (%)	93	88	80	75	75

The band combination of 2, 3, 4, and 8 was chosen for this test.
[1]k = Number of Standard Deviations; $\boldsymbol{\sigma}$ = Standard Deviation.

Table 4. The Computing Time Spent by ISODATA Using the Two Methods of Initial Seed Assignment

Random Seed Assignment		ERDAS's Seed Assignment	
Set	Computing Time (Seconds)	Number of Standard Deviations	Computing Time (Seconds)
1	17	1	94
2	26	2	103
3	21	3	68
4	19	4	41
5	23	5	30
6	15	—	—
Average	20	Average	67

The band combination of 2, 3, 4, and 8 was chosen for this test.

Comparison between the Two Clustering Methods

Computing Time

Table 5 presents the lengths of computing time of two clustering algorithms, varying with the number of pixels in the data set they processed and the number of clusters generated. The lengths of computing time spent by ISODATA were 5 to 39 (T_I/T_S) times longer than those spent by SYNERACT, as the number of pixels increased from 3,400 to 17,640. As the number of pixels was fixed and the number of clusters was increased from 8 to 16, the increase in computing time for ISODATA was still much greater than that for SYNERACT. Table 6 presents the lengths of computing time of two clustering algorithms as a

Table 5. The Computing Time Spent by SYNERACT and ISODATA for Different Numbers of Pixels and Numbers of Clusters

		SYNERACT		ISODATA		
Number of Pixels	Number of Clusters	T_S (Seconds)	Number of Iterations	T_I (Seconds)	Number of Iterations	T_I/T_S
3400	8	2.1	3	11.5	30	5
	16	3.1	6	17.2	24	6
6460	8	3.6	4	24.0	35	7
	16	5.1	5	55.1	42	11
9499	8	5.2	7	54.1	56	10
	16	7.0	10	84.6	45	12
12284	16	8.8	7	258.4	105	29
	32	12.2	14	311.9	64	26
17640	16	12.3	7	478.7	138	39
	32	16.5	17	341.4	49	21

The band combination of 2, 3, 4, and 8 was chosen for this test. The initial seeds for ISODATA were placed along the vector defined by $\mu \pm 1^*\sigma$.

TABLE 6. THE COMPUTING TIME SPENT BY SYNERACT AND ISODATA FOR SEVEN SETS OF BAND COMBINATIONS

Band Combination	SYNERACT T_S (Second)	SYNERACT Number of Iterations	ISODATA T_I (Second)	ISODATA Number of Iterations	T_I/T_S
3 8	3.9	6	64.6	69	17
3 4 8	5.4	8	99.4	70	18
2 3 4 8	7.0	10	84.7	45	12
2 3 4 5 8	8.0	4	74.8	33	9
1 2 3 4 5 8	10.6	8	93.2	35	9
1 2 3 4 5 6 8	12.1	8	69.0	22	6
1 2 3 4 5 6 7 8	13.3	8	92.0	26	7

These band combinations were chosen using ERDAS Imagine 8.3.1 software according to Euclidean distance measure.
The initial seeds for ISODATA were placed along the vector defined by $\mu \pm 1^*\sigma$.
The number of pixels = 9,499; the number of clusters = 16.

function of different band combinations. The lengths of computing time spent by ISODATA were 6 to 17 (T_I/T_S) times greater than those spent by SYNERACT, as the number of bands varied from two to eight.

Note that the number of iterations performed by SYNERACT for each case in Tables 5 and 6 was remarkably fewer than that done by ISODATA. Because SYNERACT was able to determine the appropriate (or probably optimum) locations for the initial seeds automatically from the data set itself, this greatly reduced the number of iterations needed for migrating means and accelerated reaching a convergence threshold. Thus, SYNERACT was not so heavily dependent on the iterative process as was ISODATA. Furthermore, because the biggest cluster formed by all pixels placed at root was progressively subdivided, the number of pixels at every node on each level was decreased with the increase in level number of a binary tree. While performing the iterative optimization procedure, SYNERACT did not have to do the same task at each node—to check all of the pixels in the data set against every cluster centers—as ISODATA did, thereby saving much processing time. Consequently, SYNERACT was much faster than ISODATA, irrespective of which one of the three variables (the number of pixels, clusters, or bands) was increased. This was a second benefit of SYNERACT.

Ease of Use

The input parameters of SYNERACT were compared with those of ISODATA in order to understand the ease of use of these two algorithms relative to the analyst. The algorithm of ISODATA proposed by Ball and Hall (1967) requires four (or five) input parameters: the number of clusters and their initial locations, the splitting threshold, the minimum percentage of pixels in a cluster, and the merging threshold. A sophisticated ISODATA algorithm described by Jensen (1996) normally requires the analyst to specify seven parameters, which are similar to those just mentioned above. The ISODATA program of ERDAS Imagine software (Version 8.3.1) requires the user to specify four parameters similar to those shown above and to initialize cluster means along a diagonal axis or principal axis. Note that the researcher spent much more analyst time on specifying the initial seeds when using ISODATA in this study.

In contrast, SYNERACT required the analyst to specify only the two parameters already mentioned previously. As pointed out by Richards (1993), there are about two to three spectral classes per information class on the average; C_{max} should be chosen conservatively high, with a view to eliminating unnecessary clusters at a later stage. SYNERACT began by specifying this number according to the rule just stated. The second parameter, $P\%$, was used to test the relevance of a split and to eliminate unnecessary clusters. Generally, a minimum of $10n$ to $100n$ training pixels is selected for each class when any statistically based classifier is used in supervised classification, where n is the number of bands (Lillesand and Kiefer, 2000). This broad guideline can be extended to specify the second parameter, $P\%$. The optimal values of the parameter for this study ranged from 7 percent to 9 percent. Clearly, SYNERACT eliminated the need for specifying the initial clusters in advance and required a minimum of user input, thereby saving much analyst time. Accordingly, SYNERACT was more user-friendlier for the beginner than was ISODATA.

This study also attempted to use two splitting parameters proposed by Viovy (2000) for testing the relevance of a split. Most of the calculated proportions for α ranging from 25 percent to 70 percent were greater than 20 percent, and most of the calculated values for β were greater than 0.95, even greater than 1.00, for the video image data set used in the study. The two tests frequently failed at the root or the first level of a binary tree. As a result, these two splitting parameters could not be applied to this study. The possible reason is that these two parameters are not suited for a data set containing spectrally similar but distinct land-cover types, as did the data set used in this study. Furthermore, the threshold values for α (20 percent) and β (0.95) suggested by Viovy were determined from simulated data sets, instead of real remotely sensed data, and they have not been widely applied to real data set.

Classification Accuracy

Table 7 presents the overall accuracies and the accuracies of individual categories for the two clustering approaches. The overall accuracy (92 percent) of SYNERACT was only about 1 percent lower than that of ISODATA (93 percent). Nevertheless, note that the price of attaining the accuracy of ISODATA was to spend much more analyst time on determining the optimal number of the standard deviation, from one to five for this case, when ERDAS's method of initial seed assignment was used. Indeed, the analyst time spent on picking the initial seeds was much longer than the computing time taken by ISODATA in this study. The differences in accuracy for four individual categories between SYNERACT and ISODATA were less than 3 percent, except that the differences in accuracy for johnsongrass and pigweed were greater than 6 percent. The accuracy for johnsongrass associated with ISODATA (93 percent) was 7 percent higher than that for SYNERACT (86 percent), whereas the accuracy for pigweed associated with SYNERACT (91 percent) was 6 percent higher than that for ISODATA (85 percent). Accordingly,

TABLE 7. CLASSIFICATION ACCURACIES OF THE SYNERACT AND ISODATA METHODS

Land-Cover Type	SYNERACT Number of Correct Pixels	SYNERACT Accuracy[1] (%)	ISODATA Number of Correct Pixels	ISODATA Accuracy (%)	Total Number of Test Pixels for Each Type
Cotton	2026	90	2076	93	2243
Soil	2102	97	2096	97	2163
Johnsongrass	1841	86	1983	93	2130
Cantaloupe	2114	95	2119	95	2230
Pigweed	1956	91	1815	85	2143
Sorghum	2058	90	2127	93	2277
Total	12097	—	12306	—	13186
Overall Accuracy[2]	—	92	—	93	—

The band combination of 2, 3, 4, and 8 was chosen for this test.
[1]Accuracy = Number of Correct Pixels/Total Number of Test Pixels for Each Type.
[2]Overall Accuracy = Total Number of Correct Pixels/13186.

SYNERACT and ISODATA *were* equally matched in classification accuracy.

Conclusions

This study developed SYNERACT based on the principles of hyperplane, dynamical clustering, and binary tree. SYNERACT required a minimum of user input with only two parameters, thereby saving much analyst time spent on specifying input parameters by trial and error. By comparison, ISODATA was not user friendly because it required the user to spend a much longer analyst time on specifying input parameters, particularly the initial cluster centers. This study showed that an inappropriate choice of this parameter for ISODATA significantly reduced final classification accuracies, regardless of adopting the method of completely random seed assignment or ERDAS's method. This outcome obviously was contrary to the ideas pointed out by Richards (1993) and ERDAS (1997). In contrast, SYNERACT had the ability to determine this parameter automatically from the data set itself once a hyperplane splitting two clusters was generated, and thus saved much analyst time and avoided the poor classification accuracies caused by an inappropriate choice of this parameter.

SYNERACT's computing time did not increase significantly with an increase in numbers of pixels or bands; however, the reverse relation held true for ISODATA. SYNERACT made a much smaller number of passes through the data set than did ISODATA owing to SYNERACT's capability to find suitable locations for the initial seeds automatically. Thus, SYNERACT was very fast, whereas ISODATA was time-consuming. SYNERACT was able to compete with ISODATA in classification accuracy. In sum, SYNERACT was really efficient and well suited as an alternative to ISODATA for applications in remote sensing image analysis involving a large data set, which was contrary to the thoughts proposed by Richards (1993).

References

Ball, G.H., and J.D. Hall, 1967. A clustering technique for summarizing multivariate data, *Behavioral Science*, 12:153–155.

Campbell, J.B., 1987. *Introduction to Remote Sensing*, Guilford Publications, Inc., New York, N.Y., pp. 294–333.

Eckhardt, D.W., J.P. Verdin, and G.R. Lyford, 1990. Automated update of an irrigated lands GIS using SPOT HRV Imagery, *Photogrammetric Engineering & Remote Sensing*, 56(11):1515–1522.

ERDAS, 1997. ERDAS *Field Guide*, ERDAS, Inc., Atlanta, Georgia, pp. 225–229.

Fromm, F.R., and R.A. Northouse, 1976. Class: A nonparametric clustering algorithm, *Pattern Recognition*, 8(3):107–114.

Jensen, J.R., 1996. *Introductory Digital Image Processing—A Remote Sensing Perspective*, Prentice Hall, Inc., Upper Saddle River, New Jersey, pp. 197–256.

Lee, T., and J.A. Richards, 1984. Piecewise linear classification using seniority logic committee methods with application of remote sensing, *Pattern Recognition*, 17(4):453–464.

Lillesand, T.M., and R.W. Kiefer, 2000. *Remote Sensing and Image Interpretation*, John Wiley & Sons, Inc., New York, N.Y., pp. 470–615.

Nilsson, N.J., 1965. *Learning Machines*, McGraw-Hill Book Co., New York, N.Y., pp. 15–27.

Richards, J.A., 1993. *Remote Sensing Digital Image Analysis: An Introduction*, Springer-Verlag, Berlin, Germany, pp. 229–244, 265–291.

Richardson, A.J., R.M. Menges, and P.R. Nixon, 1985. Distinguishing weed from crop plants using video remote sensing, *Photogrammetric Engineering & Remote Sensing*, 51(11):1785–1790.

Swain, P.H., 1978. Fundamentals of pattern recognition in remote sensing, *Remote Sensing: The Quantitative Approach* (P.H. Swain and S.M. Davis, editors), McGraw-Hill Book Co., New York, N.Y., pp. 136–187.

Tenenbaum, A.M., and M.J. Augenstein, 1981. *Data Structures Using Pascal*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, pp. 252–274.

Viovy, N., 2000. Automatic classification of time series (ACTS): A new clustering method for remote sensing time series, *International Journal of Remote Sensing*, 21(6):1537–1560.

Ward, J.H., Jr., 1963. Hierarchical grouping to optimize an objective function, *Journal of the American Statistical Association*, 53(301):236–244.

Wharton, S.W., and B.J. Turner, 1981. ICAP: An interactive cluster analysis procedure for analyzing remotely sensed data, *Remote Sensing of Environment*, 11:279–293.

Wishart, D., 1969. An algorithm for hierarchical classifications, *Biometrics*, 22(1):165–170.

(Received 05 September 2000; accepted 08 March 2001; revised 26 June 2001)