Satellite Sensor Image Classification Using Cascaded Architecture of Neural Fuzzy Network

Chin-Teng Lin, Yin-Cheung Lee, and Her-Chang Pu

***Abstract*—Satellite sensor images usually contain many complex factors and mixed pixels, so a high classification accuracy is not easy to attain. Especially, for a nonhomogeneous region, gray values of satellite sensor images vary greatly and thus, direct statistic gray values fail to do the categorization task correctly. The goal of this paper is to develop a cascaded architecture of neural fuzzy networks with feature mapping (CNFM) to help the clustering of satellite sensor images. In the CNFM, a Kohonen's self-organizing feature map (SOFM) is used as a preprocessing layer for the reduction of feature domain, which combines original multi-spectral gray values, structural measurements from co-occurrence matrices, and spectrum features from wavelet decomposition. In addition to the benefit of dimensional reduction of feature space, Kohonen's SOFM can remove some noisy areas and prevent the following training process from being overoriented to the training patterns. The condensed measurements are then forwarded into a neural fuzzy network, which performs supervised learning for pattern classification. The proposed cascaded approach is an appropriate technique for handling the classification problem in areas that exhibit large spatial variation and interclass heterogeneity (e.g., urban-rural infringing areas). The CNFM is a general and useful structure that can give us favorable results in terms of classification accuracy and learning speed. Experimental results indicate that our structure can retain high accuracy of classification (90% in average), while the training time is substantially reduced if our system is compared to the commonly used backpropagation network. The CNFM appears to be more reasonable and practical than the conventional implementation.**

I. Introduction

SATELLITE sensor images for remote sensing usually contain large spatial variation and interclass heterogeneity, especially in the applications of soils/cities distribution analysis (such as the urban-rural infringing areas) for the land development and the variation detection of clouds and volcano for weather forecasting and precaution. To tackle such sophisticated analysis and classification problems, it is important to combine different types of image features, including spectral and spatial data types and process these diverse features by an efficient data-fusion and classification scheme. The aim of this paper is to study various features of satellite sensor images and develop a nerual fuzzy network-based system that can assist us in analyzing and classifying the information from satellite sensor images automatically.

Early investigations for satellite-sensor image classification have employed autocorrelation functions [1], power spectra, relative frequencies of various gray levels on the unnormalized image [2], and the second-order gray level statistics method [3] to obtain texture features. These applications should be extended to produce the classification with arbitrary patterns, not just targets of selected blocks with different textures. Others have applied Bayesian classifier [4] and Markov Random Field [5], [6] to obtain relative frequencies of individual and neighbors among a pixel. These structures are hard to obtain the required results when samples are insufficient. Moreover, the consumption of time to do the classification should also be one of the considerations. Some researchers use neural networks [7]–[9] to produce the classification. The results show that the neural network is a feasible application to satellite sensor image classification. However, since the satellite sensor images usually contain many complex factors and mixed pixels, a high classification accuracy is not easy to attain. Especially for a nonhomogeneous region, the gray values of its satellite sensor image vary greatly and thus, the direct statistic gray values fail to do the categorization task correctly. To handle complicated satellite sensor image classification problems accurately and efficiently, we first combine three types of features including original gray values, statistically structural measurements, and spectrum features as the inputs of a classification system. In this system, we develop a new cascaded architecture of neural fuzzy network with feature mapping (CNFM) to help the clustering of satellite sensor images. In the proposed classifier architecture, CNFM, the dimension of input feature space is first reduced by a neural network, and then these condensed measurements are forwarded into a neural fuzzy network for classification. The adoption of the neural network and fuzzy logic techniques in this work is based on the fact that the former is nonparametric and able to handle diverse data, and the latter is good at processing uncertain data and partially known information. Also, they are both fast in the classification process after being well trained.

In the proposed CNFM, the dimension of feature space is first reduced by the Kohonen's self-organizing feature map (SOFM). No matter how many features and how many channels we used, each group of features in high dimension can be transformed into two-dimensional (2-D) coordinates by the 2-D Kohonen's SOFM. Aside from the benefit of reduction in input dimension, SOFM can remove some noisy areas and can avoid the following training process being overoriented to the training patterns. After the inputs have been condensed by Kohonen's SOFM, further classification will be performed by a neural fuzzy network (called SONFIN [12]). This cascaded architecture, named CNFM, is a general and effective structure that can give us favorable results in terms of classification accuracy and learning speed. Experimental result shows that

Manuscript received October 30, 1998; revised April 29, 1999.

The authors are with the Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu, Taiwan (e-mail: ctlin@fnn.cn.nctu.edu.tw).

Publisher Item Identifier S 0196-2892(00)02472-4.

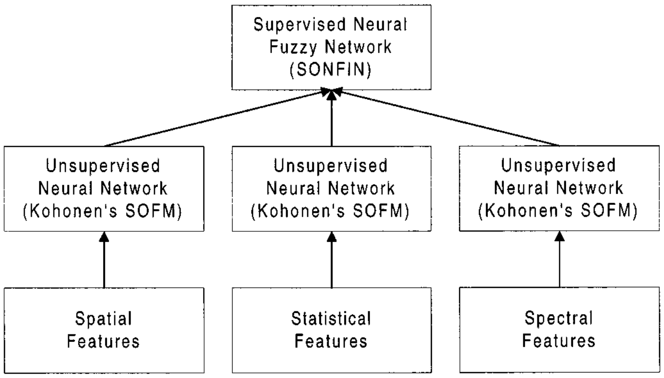

Supervised Neural Fuzzy Network (SONFIN)
Unsupervised Neural Network (Kohonen's SOFM)
Unsupervised Neural Network (Kohonen's SOFM)
Unsupervised Neural Network (Kohonen's SOFM)
Spatial Features
Statistical Features
Spectral Features

Fig. 1. Functional block diagram of the proposed CNFM.

the CNFM can reach the accuracy of 96.5% with respect to all domains of features based on the testing satellite sensor images that we had.

Fig. 1 shows the system architecture of CNFM. There are three types of inputs (features) that are spatial features of gray values, statistical features from a co-occurrence matrix, and spectral features from wavelet decomposition with N channels. Suppose there are M features in total. If we do not reduce our dimension of feature space, our network will need $(M * N)$ input nodes. In the proposed CNFM, the input dimension is first reduced by the Kohonen's SOFM and further classification is performed by a neural fuzzy network (SONFIN). In the next section, the acquirement, implementation, and modification of the three groups of features are given. In Section III, the integration of CNFM by cascading the Kohonen's SOFM and SONFIN is modeled. Experimental results are shown in Section IV. Finally, some conclusions are made in Section V.

II. Input Acquisition

The objective of this section is to clarify how the input features are chosen and what they are actually for. These input features are gray values, statistical features, and spectral features. The varieties of different gray values among channels are characteristics that can be used to classify the ground cover types from data of multispectral spaces into desired clusters. However, due to the complexities, mixed pixels, and diverse types of textures, we can make use of statistical features from a co-occurrence matrix and spectral features from wavelet decomposition to improve the accuracy of classification.

A. Features from Spatial Domain by Multispectral Data

Taking the advantage of multispectral data obtained from different satellite sensors, we can classify the data as required. Here, we choose the gray values as our spatial features. Since the varieties of different gray values among channels can be served as characteristics, they can be used to classify the ground cover types in these multispectral spaces into desired clusters. In details, the groups or clusters of pixel points are referred to as information classes, since they are the actual classes of data that a computer can recognize.

B. Features from Statistical Domain

1) Co-occurrence matrix: Texture is a major distinguishing characteristic for the analysis of many types of images. On account of the multispectral properties in satellite sensor image, texture also becomes an important characteristic for feature extraction. The gray-level co-occurrence matrices, proposed by Haralick *et al.* in 1973, constitute one of the basic approaches to the statistical analysis of texture [3] [14]. By computing a set of graytone spatial-dependence probability-distribution matrices for a given image block and extracting a set of textural features from each of these matrices, the basic model attempts to take the variation as a function of direction of spatial distance [13]–[15]. We shall also develop some structural-statistical generalizations that apply the statistical techniques to obtain the structural features of satellite sensor images.

Consider the second-order histogram $S = f(i, j, d, \theta)$, where i, j are gray values of pixels at distance d apart and θ is the angle (usually every 45°) of the line joining the centers of these pixels. These matrices are symmetric $P(\theta) = P(\theta + \pi)$ (i.e., $P(i, j, d, \theta) = P(j, i, d, \theta)$. From this matrix, a number of features can be computed. With G gray levels in the image, the dimension of the matrix will be $G \times G$. The (i, j)th element P_{ij} of this matrix is defined by [3]

$$P_{ij} = \frac{f_{ij}^{d,\vartheta}}{\sum_{i}^{N} \sum_{j}^{N} f_{ij}^{d,\vartheta}} \tag{1}$$

where $f_{ij}^{d,\vartheta}$ is the frequency of occurrence of gray levels i and j separated by a distance d and direction $\theta = 0°, 45°, 90°, 135°$. The summation is over the total number of pixel pairs N given d, in the window.

We shall compute the following texture features from the co-occurrence matrix: angular second moment (ASM), contrast (CON), inverse difference moment (IDM), and entropy (ENT). These features are among the most commonly used co-occurrence features [3], [13]–[15].

2) Simplification of Computations by Symmetric Linked-List (SLL): Since the number of operations required to compute any of the aforementioned features is proportional to the number of resolution cells (gray values being used) in the image block, co-occurrence matrices are time-consuming to compute and are memory-intensive as well. For a typical gray-valued image, assuming that window size is N and number of gray levels is G (as usual, we set $G = 256$), and then the redundancy computation is a factor of $G \times G - N(N-1) \times 2$. To overcome this problem, we use the full 256 (8 bits) dynamic range without compression and rely on efficiencies in our algorithm to overcome the computational demands of such large (256 × 256) matrices. This is in contrast to Haralick's work in [14], where the number of gray levels was compressed to a more limited dynamic range. To take advantage of the characteristics of overlapped windows and symmetric property, we can construct a symmetric linked-list (SLL) [16], which is better suited for generating co-occurrence features than a matrix approach. Another advantage for using a gray-level co-occurrence linked list is efficiency, because it does not allocate storage for those gray-level pairs that have zero

TABLE I
ORDER OF COMPLEXITY IN THE FIRST COMPUTATION OF CO-OCCURRENCE MATRIX

First Time	H	V	R	L
Co	$2(N-1)N \bullet G^2$		$2(N-1)^2 \bullet G^2$	
Asym	$2(N-1)N \bullet L1$		$2(N-1)^2 L1$	
Sym	$2(N-1)N \bullet L2$		$2(N-1)^2 L2$	

TABLE II
ORDER OF COMPLEXITY AFTER THE FIRST COMPUTATION OF CO-OCCURRENCE MATRIX

After the first time	H	V	R	L
Co	$2(N-1)N \bullet G^2$		$2(N-1)^2 \bullet G^2$	
Asym	$4N \bullet L1$	$4(N-1)\bullet L1$		
Sym	$2N \bullet L2$	$2(N-1)\bullet L2$		

probability. Each node in SLL consists of data i, data j, and a counter. Data i and j are indexes of the co-occurrence matrix, and the counter is used to count the frequency of occurrence of gray levels i and j. To increase the efficiency, each entry in SLL should be sorted. Since the co-occurrence matrix is symmetric, memory store can be reduced at least half.

The update procedure is as follows [13]–[16]. First, we construct a basic co-occurrence matrix at the left-top corner of the image and compute the required features from this matrix. Then, we subtract elements from the last column. Second, we move the window one column to the right. At the same time, we add elements from the new inserted column and compute the required features. Repeat this process column by column. When the window reaches the end of the row, we slide down to the next row. Then, the window will move in a zigzag pattern until the entire image has been covered. Although it is not easy to design the SLL, it is the fastest method to calculate the co-occurrence matrix with full color range. Moreover, this update method can be usefully applied to other applications.

3) Order of Complexity: To calculate the order of complexity, we define N to be the size of neighborhood, G to be the number of gray levels, L1 to be the length of asymmetric linked-list, and L2 to be the length of symmetric linked-list. Since we use the full 256 (8 bits) dynamic range without compression in this work, the length of symmetric linked-list L2 is almost half (i.e., the nonredundant portion) of the length of asymmetric linked-list L1. Also, we define Co to be Co-occurrence matrix, Asym to be co-occurrence matrix with Asymmetric linked-list, and Sym to be co-occurrence matrix with Symmetric linked-list. Since the co-occurrence matrix is updated to calculate the order of complexity of subtraction and addition between rows (Horizontal), between columns (Vertical), and between slant of rows and of columns, we define H to be Horizontal, V to be Vertical, L to be slant toward Left, and R to be slant toward Right.

Tables I and II show the order of complexity of the first iteration and each of the following iterations, respectively. Since there is no other consideration in overlapping area in Co, the complexity of Table II is the same as that in Table I. However, for the linked-list format, the factor of G^2 is reduced to either L1 or L2, and the factor of $2(N-1)$ is reduced to either 4 or 2, correspondingly. This is because we only need to update either two columns or two rows for the linked-list format. Furthermore, Since the length of L2 is almost half of the length of L1, the Symmetric format has the lowest computational complexity.

4) Global Texture Features from Second-Order Histogram Statistics: From the co-occurrence matrix, we can define the texture features as follows [3].

a) Entropy (ENT): The entropy computed from the second order histogram provides a standard measurement of homogeneity and is defined as

$$Entropy = -\sum_{i}^{n}\sum_{j}^{n} p(i,j)\log(p(i,j)). \qquad (2)$$

when n stands for the total levels of gray value (i.e., in our experiment, we set $n = 2^8 = 256$). Higher values of homogeneity measures will indicate less structural variations, while lower values will be interpreted as a higher probability of textural region.

b) Contrast (CON): The contrast feature is a difference moment of the P matrix and is a standard measurement of the amount of local variations presented in an image. The contrast feature on the second-order histogram is defined as

$$Contrast = \sum_{i}^{n}\sum_{j}^{n} (i-j)^2 p(i,j) \qquad (3)$$

The higher the values of the contrast are, the sharper the structural variations in the image are.

c) Angular Second Moment (ASM): The angular second moment gives a strong measurement of uniformity and can be defined as

$$Angular\ Second\ Moment = \sum_{i}^{n}\sum_{j}^{n} \{p(i,j)\}^2. \qquad (4)$$

Higher nonuniformity values provide evidence of higher structural variations.

d) Inverse Difference Moment (IDM): The inverse difference moment is a measure of local homogeneity, and is defined as

$$Inverse\ Difference\ Moment = \sum_{i}^{n}\sum_{j}^{n} \frac{1}{1+(i-j)^2/G^2} p(i,j). \qquad (5)$$

It is noticed that features such as correlation cannot be used, since if the variance becomes zero, the correlation will come to infinity.

C. Features from Spectrum Domain

The wavelet decomposition [10], [11] is a mathematical framework for constructing an orthonormal basis for the space of all finite energy signals. It can decompose input signals into multiscale details, describing their power at each scale and position, so it can discriminate the local properties corresponding to smooth and textured areas. The framework of wavelet decomposition is based on the notion of a multiresolution analysis, which consists of a chain of approximation vector spaces $V_j\ j \in Z$. In other words, just as sin and cos functions are orthogonal, so the set of scaled and dilated wavelets forms an orthonormal basis for R [10], [11]. The first basis consists of a set of scaling functions $\phi(x)$ [10]

$$\{2^{j/2}\phi(2^j x - n) | n \in Z\} \tag{6}$$

where $j < 0$ is a scaling factor that forms an orthonormal basis for V_j. Besides, there is a corresponding set of functions, a wavelet basis $\psi(x)$ [10],

$$\{2^{j/2}\psi(2^j x - n) | n \in Z\} \tag{7}$$

that forms an orthonormal basis for W_j. The spaces $W_j,\ j \in Z$ are mutually orthogonal.

In practice, the approximation and detailed projection coefficients associated with V_j and W_j are computed from the approximation coefficients at the next higher scale V_{j+1} using a quadrature mirror filter pair (QMF) and a pyramidal sub-band coding scheme. The impulse responses of the QMF pair are typically denoted by $h(n)$ and $g(n)$, where $h(n)$ is formed from the inner product between $\phi(x)$ and $\phi(2x - n)$, and $g(n)$ is formed from the inner product between $\psi(x)$ and $\psi(2x - n)$, i.e.,

$$\phi(x) = \sqrt{2} \sum_n h(n)\phi(2x - n) \quad \text{and} \tag{8}$$

$$\psi(x) = \sqrt{2} \sum_n g(n)\psi(2x - n). \tag{9}$$

The coefficients $\{h(n)\}$ have to meet several conditions for the set of basis functions to be unique and orthonormal. It can be shown that a wavelet decomposition of a signal does not require the explicit forms of $\phi(x)$ and $\psi(x)$, but only depends on $h(n)$ and $g(n)$ sequences. Moreover, filtering operations use $\{h(n)\}$ and $\{g(n)\}$ coefficients as the impulse responses corresponding to low and high-pass operations, respectively. By repeatedly convoluting each approximation signal with $h(n)$ and $g(n)$, and decimating the outputs by a factor of two, the signal is decomposed into frequency bands whose bandwidths and center frequencies vary by octaves.

In this paper, we choose the quadratic spline function as the scaling function [11], i.e.,

$$\tilde{\phi}(\omega) = \left[\frac{\sin\frac{\omega}{2}}{\frac{\omega}{2}}\right]^3. \tag{10}$$

The Fourier transform of its corresponding wavelet is

$$\tilde{\psi}(\omega) = j\omega \left[\frac{\sin\frac{\omega}{4}}{\frac{\omega}{4}}\right]^4. \tag{11}$$

Then we can deduce the coefficients $\{h(n)\}$ and $\{g(n)\}$ as follows [11], [17]:

$$h(n) = \frac{1}{8}\left[\frac{\sin\pi(n+\frac{3}{2})}{\pi(n+\frac{3}{2})} + \frac{\sin\pi(n-\frac{3}{2})}{\pi(n-\frac{3}{2})} + 3\frac{\sin\pi(n+\frac{1}{2})}{\pi(n+\frac{1}{2})} + 3\frac{\sin\pi(n-\frac{1}{2})}{\pi(n-\frac{1}{2})}\right] \tag{12}$$

$$g(n) = \sum_{u=0}^{3} 2c(u)\left[\frac{\sin(n+u+1)\pi}{(n+u+1)\pi} - \frac{\sin(n-u)\pi}{(n-u)\pi}\right] \tag{13}$$

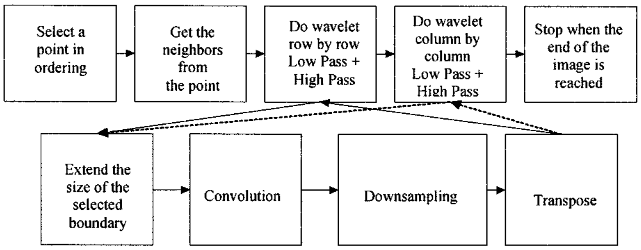

Select a point in ordering
Get the neighbors from the point
Do wavelet row by row Low Pass + High Pass
Do wavelet column by column Low Pass + High Pass
Stop when the end of the image is reached
Extend the size of the selected boundary
Convolution
Downsampling
Transpose

Fig. 2. Procedure of wavelet decomposition.

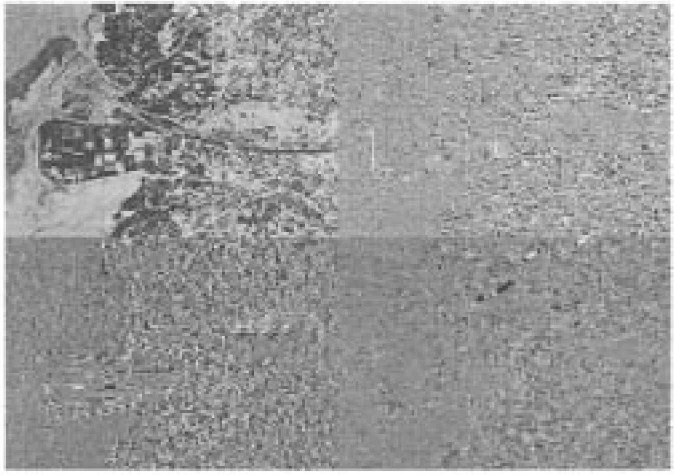


Fig. 3. Result of wavelet decomposition of a channel input, where the top left and bottom right are the result of wavelet decomposition from both low pass and from both high pass in columns and in rows, respectively. The top right image is from low pass in columns and high pass in rows. Conversely, the bottom left image is from high pass in columns and low pass in rows.

where $c(n)$ can be selected as [17]

$$c(0) = 1.909686, \quad c(1) = 1.215350, \quad c(2) = 0.344046,$$
$$c(3) = 0.122197, \quad c(4) = 0.028610, \quad \text{and}$$
$$c(n) = 0.0, \quad \text{for } n \geq 5.$$

The complete procedure is shown in Fig. 2. First, we select a point from the left-top of the image and obtain a block from the gray values of its neighbors. Second, we perform the wavelet decomposition row by row, including cases of low pass and high pass. Finally, we apply the same process column by column. This is the 2-D wavelet decomposition [17]. For the wavelet decomposition, we first extend the size of the selected block to be a square block. Then, we perform the convolution and downsample the size of the block into half. Afterwards, we transpose the image for the next wavelet decomposition. As an illustrated example shown in Fig. 3, the left-top and right-bottom images are the result of wavelet decomposition from both low pass, and from both high pass in columns and in rows, respectively. The right-top image is from low pass in columns and high pass in rows. Conversely, the left-bottom image is from high pass in columns and low pass in rows.

III. Cascaded Architecture of Neural Fuzzy Network with Feature Mapping (CNFM)

After we have obtained the features from the spatial, statistical, and spectral domains, we proceed with the training of clas-

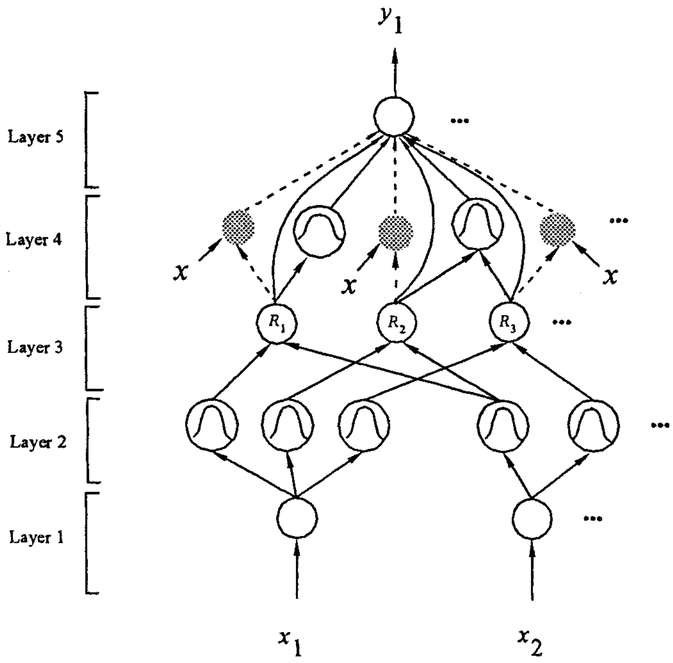


Fig. 4. Structure of Self-Constructing Neural Fuzzy Inference Network (SONFIN).

sification by our system, CNFM. It is composed of two cascaded neural networks. The former one is the unsupervised Kohonen's SOFM and the latter is the supervised neural fuzzy network (called SONFIN). From Fig. 4, we can see the detailed information of the architecture. For each channel, we obtain thirteen types of features as the inputs of Kohonen's SOFM and get the individual outputs, where the output of each SOFM is the 2-D coordinates (on the x and y axes) in the map formed by the 2-D neurons (clustering units). In this way, we transform the thirteen statistic measurements into three 2-D coordinates for each channel. Next, we use the supervised neural fuzzy network, SONFIN, to produce a refined classification. This complementary architecture can compensate the problems of inaccuracy and long training time of unsupervised and supervised neural networks, respectively. It is noticed that although the concept of cascading a supervised neural network by an unsupervised learning neural network is not an entirely new concept in the neural network realm for speeding up learning, the proposed architecture is a totally new combination. In addition to owning the advantage of speedy learning, more importantly, the proposed architecture provides a new efficient solution to the pattern recognition problems containing many diverse types of features, thus making a contribution to the domain of pattern recognition in general and remote sensing classification in particular. In more details, the most important contribution of this architecture is to improve the method of input selection. Compared to the conventional trial and error approach, the input dimension of our system is first reduced by the Kohonen's SOFM, and then each group of features from a channel is transformed into 2-D coordinates. Consequently, with benefit of input dimension reduction, Kohonen's SOFM unsupervised neural network also contributes to noise elimination. Next, we use SONFIN to overcome the inaccuracy due to Kohonen's SOFM. Hence, no matter how many features and how many channels that we use, the problem of big input space and long training time can be removed by the proposed mechanism.

A. Reduction of Input Dimension by Unsupervised Network

This subsection introduces the 2-D Kohonen's SOFM, which can map high-dimensional inputs onto a 2-D map and can filter out some noisy information [19]. In addition to mapping high-dimensional inputs onto a 2-D map, another major goal of Kohonen's SOFM is to obtain a topology-preserving map that keeps the neighborhood relations of the input patterns. We shall also modify the Kohonen's SOFM to make the input clusters be normally distributed. With the benefits of Kohonen's SOFM, the system can be adapted to the change of increase of both features and channels easily. Also, it can filter out some noisy information. The most important merit is that we can avoid taking trial and error to remove redundancy, such as by genetic algorithm, KL expansion, and correlation $(x_j^T y)/[(x_j^T x)\,(y_j^T y)]$.

The 2-D SOFM, also called a topology-preserving map, assumes a topological structure among the cluster units. There are m cluster units, arranged in a 2-D array in our application, the input signals are $n$tuples. The weight vector for a cluster unit serves as an exemplar of the input patterns associated with that cluster. During the self-organization process, the cluster unit whose weight vector matches the input pattern most closely (typically, the square of the minimum Euclidean distance) is chosen as the winner. The winning unit and its neighboring units (in terms of the topology of the cluster units) update their weights. The 2-D coordinates (on the x and y axes) of the winning unit on the 2-D map of SOFM are viewed as the output of SOFM.

To make an unsupervised neural network utilize the information concerning the distribution of patterns, the patterns must be divided into m clusters according to the occurrences of clusters. Therefore, clusters will be sufficient enough in a high-density region, whereas the number of clusters will be reduced in a sparse region. We apply this conscience [18] into the update of the Kohonen's SOFM. After the selection of a winner, i.e.,

$$y_j = \begin{cases} 1, & \text{if } ||X(t) - W_j(t)|| \\ & \quad = \min\limits_{l=1}^{m} ||X(t) - W_j(t)|| \\ 0, & \text{otherwise,} \end{cases} \tag{14}$$

where y_j is the winner, $X(t)$ is the current inputs, $W_j(t)$ is the weight vector (cluster j), and instead of updating the weights as usual, we record the usage of each cluster, i.e.,

$$p_j^{new} = p_j^{old} + \beta(y_j - p_j^{old}), \qquad j = 1, \cdots, m \tag{15}$$

where p_j is the record of usage and β is a step size. After the statistics are known, the selection of the winner will be rewritten as

$$y_j = \begin{cases} 1, & \text{if } ||X(t) - W_j(t)|| - b_j \\ & \quad = \min\limits_{l=1}^{m} ||X(t) - W_l(t)|| - b_l \\ 0, & \text{otherwise,} \end{cases} \tag{16}$$

where $b_j = \gamma((1/m) - p_j^{new})$ is an offset, andγ is a constant.

As the usage p_j of cluster j increases, its offset b_j will decrease. This reduces its competitive ability. In other words, its conscience is increased and is persuaded to give opportunity to other clusters.

B. Classification by Supervised Network

After the transformation of input space using Kohonen's SOFM has been completed, we then pass the newly composed inputs into a supervised neural network to produce a refined classification. This cascaded architecture has the ability to perform the classification of satellite sensor images even though there are very complex mixed pixels.

The neural fuzzy network that we used for satellite sensor image classification is called the self-constructing neural fuzzy inference network (SONFIN) that we proposed previously in [12]. It is worth mentioning that SONFIN is a brand new idea applied to satellite sensor image classification. The SONFIN is a general connectionist model of a fuzzy logic system, which can find its optimal structure and parameters automatically. The number of generated rules and membership functions is small even for modeling a sophisticated system. The SONFIN can always find itself an economic network size, and the learning speed as well as the modeling ability are all appreciated as compared to normal neural networks.

The structure of SONFIN is shown in Fig. 4. The five-layered network realizes a fuzzy model of the following form

Rule i: IF x_l is A_{il} and $\cdots$ and x_n is A_{in}
THEN y is $m_{oi} + a_{ji}x_j + \cdots$

where A_{ij} is a fuzzy set, m_{oi} is the center of a symmetric membership function on y, and a_{ji} is a consequent parameter. It is noted that unlike the traditional TSK model [19], where all the input variables are used in the output linear equation, only the significant ones are used in the SONFIN (i.e., some a_{ij}'s in the above fuzzy rules are zero). We shall next describe the functions of the nodes in each of the five layers of the SONFIN.

Layer 1: This layer is an input layer

$$a^{(1)} = u_i^{(1)} = x_i. \tag{17}$$

Layer 2: Each node in this layer corresponds to one linguistic label (small, large, etc.) of one of the input variables in Layer 1. With the use of the Gaussian membership function, the operations performed in this layer are

$$a^{(2)} = e^{-((u_{ij}^{(2)} - m_{ij})^2/\sigma_{ij}^2)} \tag{18}$$

where m_{ij} and σ_{ij} are, respectively, the center (or mean) and the width (or variance) of the Gaussian membership function of the jth term of the ith input variable x_i.

Layer 3: A node in this layer represents one fuzzy logic rule and performs precondition matching of a rule. Here, we use the following AND operation for each Layer-3 node

$$a^{(3)} = \prod_i u_i^{(3)} \tag{19}$$

where n is the number of Layer-2 nodes participating in the IF part of the rule.

Layer 4: This layer is called the consequent layer. Two types of nodes are used in this layer, and they are denoted as blank and shaded circles in Fig. 4, respectively. The node denoted by a blank circle (blank node) is the essential node representing a fuzzy set (described by a Gaussian membership function) of the output variable. The function of the blank node is

$$a^{(4)} = \sum_j u_j^{(4)} \cdot a_{0i} \tag{20}$$

where $a_{0i} = m_{0i}$ the center of a Gaussian membership function. As to the shaded node, it is generated only when necessary. The shaded node function is

$$a^{(4)} = \sum_j a_{ji}x_j \cdot u_i^{(4)} \tag{21}$$

where the summation is over all the inputs, and a_{ji} is the corresponding parameter. Combining these two types of nodes in Layer 4, we obtain the whole function performed by this layer for each rule as

$$a^{(4)} = \left(\sum_j a_{ji}x_j + a_{oi}\right) u_i^{(4)}. \tag{22}$$

Layer 5: Each node in this layer corresponds to one output variable. The node integrates all the actions recommended by Layers 3 and 4 and acts as a defuzzifier with

$$a^{(5)} = \sum_i a_i^{(4)} \Big/ \sum_i a_i^{(3)}. \tag{23}$$

The SONFIN can be used for normal operation at any time during the learning process without repeated training on the input-output patterns when online operation is required. There are no rules (i.e., no nodes in the network except the input/output nodes) in the SONFIN initially. They are created dynamically as learning proceeds upon receiving online incoming training data by performing the following learning processes simultaneously:

1) input/output space partitioning;
2) construction of fuzzy rules;
3) consequent structure identification;
4) parameter identification.

Processes 1, 2, and 3 belong to the structure learning phase, and process 4 belongs to the parameter learning phase. The details of these learning processes can be found in [12].

C. Cascaded Architecture of Neural Fuzzy Network with Feature Mapping (CNFM)

This subsection gives the details of how to implement the cascaded architecture of the unsupervised and supervised neural networks presented in Section III-A and III-B, respectively. The general architecture of CNFM is set up as follows. At first, we utilize k (e.g., $k = 3$), Kohonen's SOFM's to reduce the dimensions of k sets of inputs. The full (three) sets of inputs that we have here are gray values, statistical properties, and spectrum features from wavelet decomposition. Instead of using all

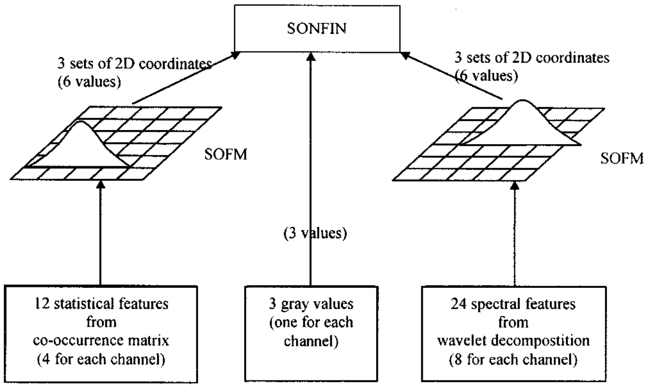


Fig. 5. System architecture of the CNFM used in our experiment.

sets of selected features as the inputs of a supervised neural network, the Kohonen's SOFM can transform each set of features into 2-D coordinates and use these low-dimension data as the inputs of the supervised neural fuzzy network, SONFIN. Not only can we get a better representation, but we can also obtain a good meaning when the 2-D coordinates serve as the inputs of a supervised neural network.

To go into a little more detail, Fig. 5 shows the architecture of our system. The first set of inputs contains three gray values, each of which comes from one of three channels and represents the features of spatial domain. The second set of inputs consists of angular second-moment, contrast, inverse difference moment, and entropy that come from co-occurrence matrices. This set of inputs represents the features of statistical domain. The last set of inputs includes energy and entropy that come from wavelet decomposition and represents the features of spectral domain. After the transformation of Kohonen's SOFM's, these three sets of inputs are reduced into 2-D coordinates. However, we do not expand a gray value into 2-D coordinates because its dimension is low enough in our test examples. Also, if we have transformed the gray values of all channels, we may remove the information of differences among channels. Since the statistical and spectral features focus on local varieties, we can apply the transformation directly and do not need to care for the information of differences among channels. As a result, we reduce the input dimension from 39 features (13 features/channel × 3 channels) into 15 features (5 features/channel × 3 channels, where each channel has one gray value and two 2-D coordinates). As shown in Fig. 5, the three types of inputs are a gray value obtained from each channel, four statistical features from each channel, and eight spectral features from each channel, where there are three channels totally. Hence, if we have not reduced our dimension of inputs, there are 39 inputs in total. It must be noted that if the number of features increase, the number of inputs will grow terribly.

After the feature domain has been reduced and transformed by the Kohonen's SOFM, we pass the condensed measurements to a supervised neural fuzzy network SONFIN. This network can perform online input space partitioning, which creates only the significant membership functions on the universe of discourse of each input variable by using a fuzzy measure algorithm [12] and the orthogonal least square (OLS) method [19]. Our objective is to use SONFIN to obtain lower mean square error (MSE) and higher learning rate. The result will be compared to that of normally used statistics and backpropagation-network schemes in Section IV.

IV. Experimental Results

Our experiment makes use of the real High Resolution Visible (SPOT HRV) satellite sensor images provided by the Earth Resource Satellite Receiving Station in R.O.C. The image size of each channel is 1024 × 1024, with full 256 gray levels. These images contain five classes, with respect to soil (Class 1), forest (Class 2), sea (Class 3), city 1—residential area (Class 4) and city 2—industrial area (Class 5). These five classes are defined entirely based on the ground truth obtained by onsite investigation. Due to the different properties of these five classes, we believe their morphologies also differ, giving them different textures. We find that some of these classes are expected to be highly separable from each other (i.e., soil, forest, and sea), but the others (i.e., city 1 and city 2) are quite similar since both of them are urban, although city 1 is the residential area and city 2 is the industrial area. The requirement of distinguishing city 1 from city 2 makes this classification problem more difficult. Our experiment will demonstrate this phenomena and show the need for combining diverse types of features in the proposed CNFM. Our objective is to classify these classes from the three-channel satellite sensor images. The results show that when we try to use all features as inputs directly without any dimension reduction to perform the training of classification on either a SONFIN or a backpropagation (BP) network, the converged MSE is so large that we cannot reduce it. Hence, we must first apply the SOFM in our CNFM to reduce the dimension of inputs and then perform the training of classification.

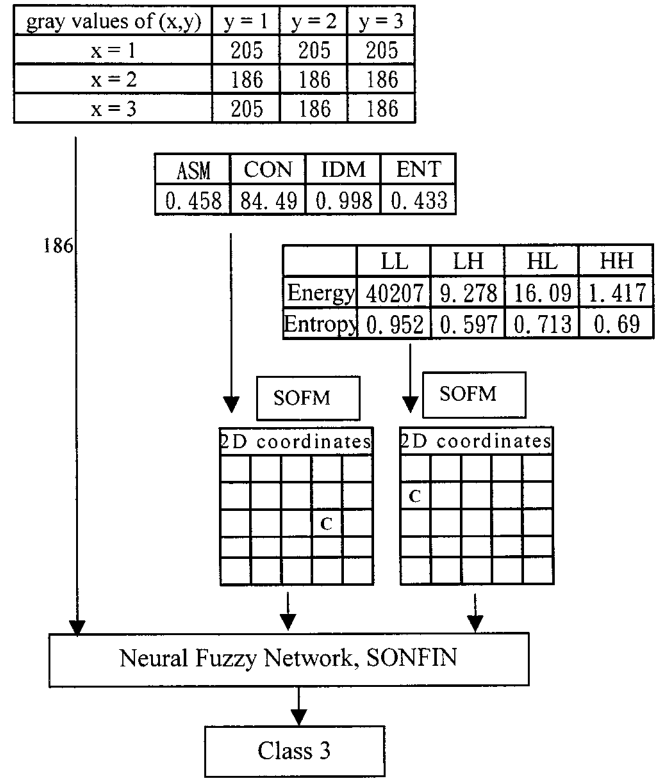
An illustrative example for classification of the features from a channel using CNFM is shown in Fig. 6, and it can give the details of the proposed system. First, we obtain the gray value of a center-pixel, 186. From the neighborhood of this pixel, we then calculate the Angular Second Moment (ASM), Contrast (CON), Inverse Difference Moment (IDM), entropies and energies from the co-occurrence matrices and wavelet transformations with a neighborhood size $N = 7$. After we have acquired the input features, we pass each group of the features into a Kohonen's SOFM. Then, each group of input features with high dimension is reduced to 2-D coordinates. Finally, we use these 2-D coordinates, combined with the coordinates from other channels, as the inputs of the neural fuzzy network SONFIN. The result we obtained is the desired cluster, and its name is Class 3. In our experiments, the learning parameters of SOFM, including neighborhood size, learning rate, and convergent rate are set as follows:

1) neighborhood size: initial value is 100 and updated by (i.e., the shrinkage term)

$$neighborhood\ size/(0.4 * epoch/1000.0 + 1);$$

2) learning rate: initial value is 0.9 and updated by

$$learning\ rate * \exp(-0.3 * epoch/50);$$



gray values of (x,y)	y = 1	y = 2	y = 3
x = 1	205	205	205
x = 2	186	186	186
x = 3	205	186	186

ASM	CON	IDM	ENT
0.458	84.49	0.998	0.433

	LL	LH	HL	HH
Energy	40207	9.278	16.09	1.417
Entropy	0.952	0.597	0.713	0.69

Fig. 6. Illustrative example for classification of the features from a channel using CNFM.

TABLE III
CLASSIFICATION RESULT WITH ALL TYPES OF FEATURES

	Class 1	Class 2	Class 3	Class 4	Class 5	Overall	Average
Class 1 (Soil)	495	0	0	4	1	99 %	96.5 %
Class 2 (Forest)	0	512	0	0	0	100 %	
Class 3 (Sea)	0	0	470	0	0	100 %	
Class 4 (City 1)	7	2	1	424	28	92.2 %	
Class 5 (City 2)	25	5	0	13	462	91.5%	

3) convergent rate: convergent rate is updated by

$$1-\frac{number\text{ of training set}}{total\ number\ of\ training\ set}.$$

In SONFIN, the learning rate $\eta = 0.001$, overlap degree $\beta = 0.25$, similarity degree $\rho = 0.6$, forgetting factor $\lambda = 1$, threshold parameters $\overline{F}_{in} = 0.04$, and $\overline{F}_{out} = 0.025$ are chosen. After 50 000 time steps of training, 18 input clusters (rules) are generated. The final converged MSE is 12.74.

Tables III–V show the results of the classification using the proposed CNFM. Here we compared the classification accuracies resulting from different types of inputs. Based on the satellite sensor images that we had, the result in Table III got the average classification accuracy of 96.5% and was the best among the others, because all information was used as inputs to CNFM in this case. Table IV shows the result of classification using only the information of gray levels and statistics and therefore, the average classification accuracy decreased to 94.9%. We can see that the classification accuracy decreased in each class. The last one is Table V, and the result was obtained with three gray values only. The average classification rate became 92.5%. Table V tells us that the gray levels alone are only good at distinguishing Class 3 (sea) from the others. By comparing Tables IV and V, we find that the statistics features do enhance the discriminative power of gray levels on Class 1 (soil) and Class 2 (forest), but not on Class 4 (city 1—residential area) and Class 5 (city 2—industrial area). By comparing Tables III and IV, we find that the spectrum features can further enhance the discrimination power of gray levels and statistics features on the two urban areas, Class 1 and Class 2. It can thus be seen that without sufficient information, classification is difficult to perform. Fig. 7 shows the ground truth of the satellite sensor image and the classified image using CNFM with three types of features. We see that CNFM can perform quite an accurate classification.

TABLE IV
CLASSIFICATION RESULT WITH GRAY VALUES AND STATISTICS FEATURES

	Class 1	Class 2	Class 3	Class 4	Class 5	Overall	Average
Class 1 (Soil)	490	0	0	7	3	98 %	94.9 %
Class 2 (Forest)	0	512	0	0	0	100 %	
Class 3 (Sea)	0	0	470	0	0	100 %	
Class 4 (City 1)	9	0	0	416	37	90.3 %	
Class 5 (City 2)	44	8	5	16	435	86.1%	

TABLE V
CLASSIFICATION RESULT WITH GRAY VALUES ONLY

	Class 1	Class 2	Class 3	Class 4	Class 5	Overall	Average
Class 1 (Soil)	462	0	3	23	12	92.4 %	92.5 %
Class 2 (Forest)	9	477	5	10	11	93.2 %	
Class 3 (Sea)	0	0	470	0	0	100 %	
Class 4 (City 1)	17	7	3	384	51	83.1 %	
Class 5 (City 2)	21	7	1	9	649	93.7 %	

Table VI shows the classification results of the k nearest-neighborhood (KNN) method [19], BP network [19], and the CNFM. The inputs of BP network and CNFM are first filtered by Kohonen's SOFM, whereas KNN uses all features directly. In other words, the compared BP network is actually in the same structure as the CNFM, but the SONFIN in CNFM is replaced by a normal BP network. We had also tried to use the normal BP network by feeding all the 39 features (13 features for each channel) into it directly as inputs without going through Kohonen's SOFM for the reduction of feature space. However, the learning of this BP network appeared very slow, and the MSE was keeping at quite a high value. That is, it was stuck at a very bad local minimum, since the network was too big. Hence, we do not present the experimental results of this BP network here. The BP network used has two hidden layers with 30 hidden nodes in each layer, trained by the standard backpropagation learning rule with momentum term, where the learning rate was

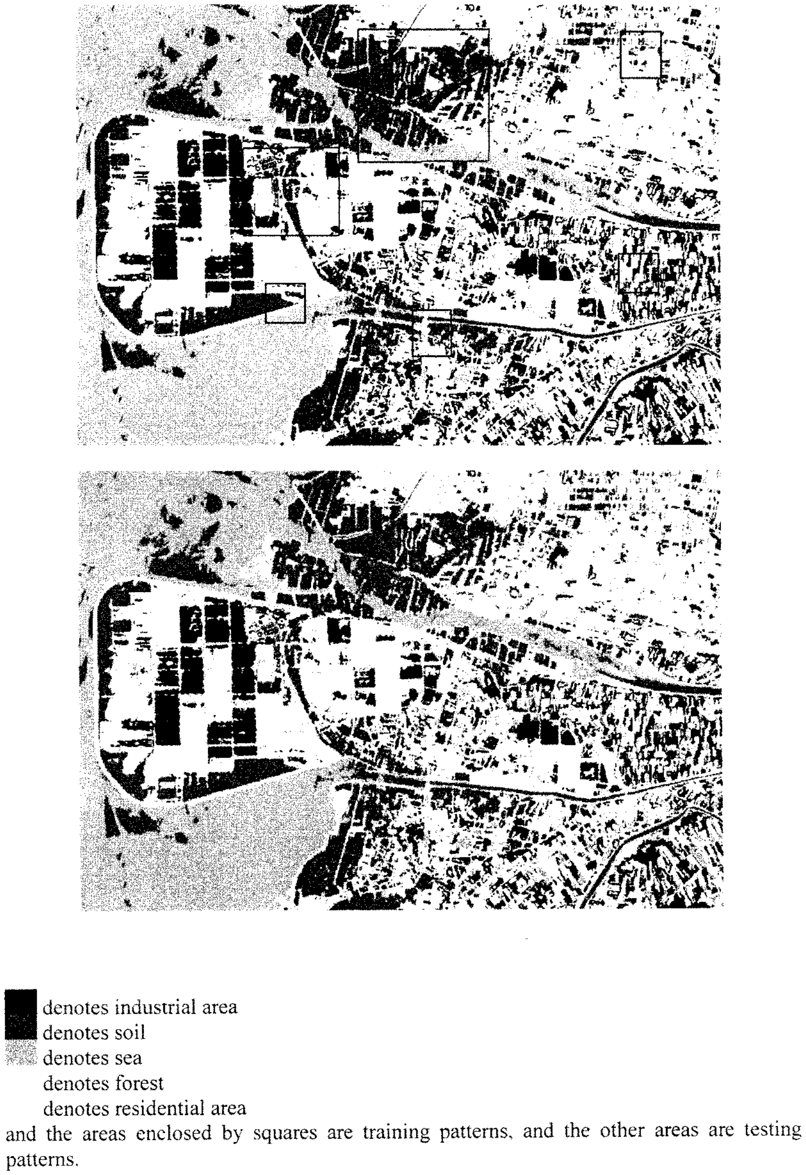

denotes industrial area
denotes soil
denotes sea
denotes forest
denotes residential area
and the areas enclosed by squares are training patterns, and the other areas are testing patterns.

Fig. 7. Ground truth of the satellite sensor image used in the experiment (top) and the classified result of CNFM (bottom), where industrial area, soil, sea, forest, residential area are all denoted, the areas enclosed by squares are training patterns, and the other areas are testing patterns.

TABLE VI
COMPARISON OF THE CLASSIFICATION RESULTS AMONG KNN, BP, AND CNFM

	KNN	BP	CNFM
Class 1 (Soil)	77%	88%	99 %
Class 2 (Forest)	82%	85%	100 %
Class 3 (Sea)	97.5%	100%	100 %
Class 4 (City 1)	73%	78.5%	92.2 %
Class 5 (City 2)	70%	77%	91.5%
Average	79.9%	85.7%	96.5%

set as 0.5, the momentum parameter was set as 0.9, and the training time was 2300 epochs (about 50 000 time steps). The network structure, as well as the learning constants, were determined after thorough trial-and-error testing for achieving the best performance (i.e., lowest MSE). The learning was stopped when the MSE almost kept constant. Table VI shows that the CNFM outperforms the KNN and BP network in average classification accuracy by 16.6% and 10.8%, respectively.

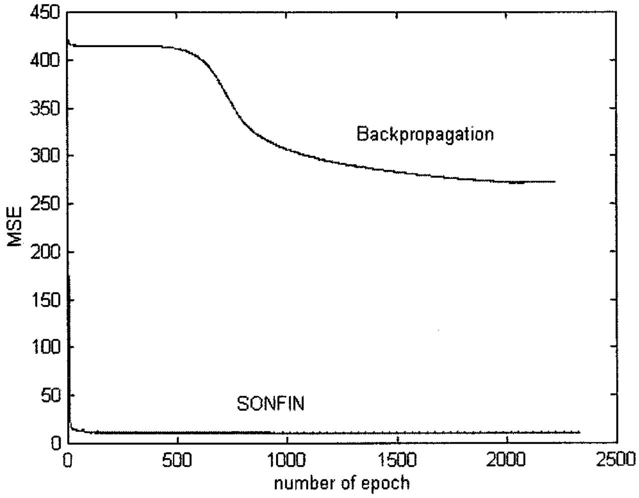


Fig. 8. MSE of backpropagation network against SONFIN.

Fig. 8 shows the accuracy and MSE of BP network against SONFIN in our CNFM. We can conclude that the BP network is insufficient to reduce the MSE due to the complexity of the image sources, but our system can tackle this problem successfully. From Tables III–VI of our experiments, we can summarize that not only the proposed diverse types of features, but also the cascaded model, CNFM, can enhance the classification accuracy of satellite sensor images of the traditional approaches. However, although the experiment based on our satellite sensor image shows that the accuracy improvement is somewhat significant, the exact extent to which the classification accuracy can be increased by the proposed scheme depends on the particular satellite sensor images tested.

V. Conclusions

The major contribution of the proposed system is to provide a good solution to the problem of scaling and selection of satellite sensor image features. Hence, in this paper, we provide some auxiliary information to handle problems of diverse types of textures and the varieties of neighborhood in the sophisticated satellite sensor image classification problems. However, the additional features will increase the computation complexity a great deal and mislead the training. To overcome this problem, we present a new cascaded system (called CNFM) that uses Kohonen's self-organizing feature map as the reduction mechanism of input dimension. After we have obtained the condensed measurements, a fine classification on them is performed by a neural fuzzy network (called SONFIN).

The major contribution of the proposed system is to solve the problem of scaling and selection of image features well. Also, we extended the mechanism of Kohonen's SOFM in the sense of reducing the high input dimension rather than filtering the training sets. Our CNFM, combining spatial, statistical, and spectral features into one system, generalizes well and can avoid the network parameters orienting to the training patterns. Furthermore, we improved the computation complexity of the co-occurrence matrix by the symmetric linked-list structure. Our future work is to enhance our cascaded system by using edge information. Edge information can be used as an outline of the distribution of clusters and to improve the classification accuracy.

References

[1] H. Kaizer, "A quantification of textures on aerial photographs," Boston Univ. Res. Lab., Boston University, Boston, MA, Technical note 121, AD 69484, 1955.

[2] E. M. Darling and R. D. Joseph, "Pattern recognition from satellite altitudes," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, pp. 38–47, Mar. 1968.

[3] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Texture features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 610–621, 1973.

[4] J. M. Boucher, P. Lena, and J. F. Marchand, "Application of local and global unsupervised Bayesian classification algorithms to the forest," *IGARSS '93, Better Understanding of Earth Environment (Cat. No. 93C3294-6)*, vol. 2, pp. 737–739, Aug. 1993.

[5] E. Rignot and R. Chellappa, "Segmentation of polarimetric synthetic aperture radar data," *IEEE Trans. Image Processing*, vol. 1, pp. 281–299, July 1992.

[6] A. H. S. Solberg, T. Taxt, and A. K. Jain, "A Markov random field model for classification of multisource satellite imagery," *IEEE Trans. Geosci. Remote Sensing*, vol. 34, pp. 100–113, Jan. 1996.

[7] Y. Hara, G. Atkins, S. . Yueh, R. T. Shin, and J. A. Kong, "Application of neural networks to radar image classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, pp. 100–109, Jan. 1994.

[8] Y. C. Tzeng, K. S. Chen, W. L. Kao, and A. K. Fung, "A dynamic learning neural network for remote sensing applications," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, pp. 994–1102, Sept. 1994.

[9] Y. C. Tzeng and K. S. Chen, "AS fuzzy neurasl network to SAR image classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 36, pp. 301–307, Jan. 1998.

[10] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, pp. 909–996, Nov. 1988.

[11] S. G. Mallat, "Multiresolution approximation and wavelets orthonormal base of $L^2(R)$," *Trans. Amer. Math. Soc. 1989*, vol. 315, pp. 69–88, Sept. 1989.

[12] C. F. Juang and C. T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 12–32, Feb. 1998.

[13] P. Gong, D. J. Marceau, and P. J. Howarth, "A comparison of spatial feature extraction algorithms for land-use classification with SPOT HRV data," *Remote Sens. Environ.*, vol. 40, pp. 137–151, May 1992.

[14] R. M. Haralick, "Statistical and structural approaches to texture," in *Proc. IEEE*, vol. 67, May 1979, pp. 786–804.

[15] D. R. Peddle and S. E. Franklin, "Image texture processing and data integration for surface pattern discrimination," *Photogramm. Eng. Remote Sensing*, vol. 57, no. 4, pp. 413–420, 1991.

[16] D. Clausi and M. Jernigan, "A fast method to determine co-occurrence texture features," *IEEE Trans. Geosci. Remote Sensing*, vol. 36, no. 1, pp. 298–300, 1998.

[17] J. W. Hsieh, M. T. Ko, H. Y. Liao, and K. C. Fan, "A new wavelet-based edge detector via constrained optimization," *Image Vis. Comp.*, vol. 15, pp. 511–527, July 1997.

[18] D. DeSieno, "Adding a conscience to competitive learning," in *Proc. IEEE ICNN*, vol. 1, San Diego, July 1988, pp. 117–124.

[19] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A Neural-Fuzzy Synergism to Intelligent Systems (with disk)*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[20] P. L. Palmer and M. Petrou, "Locating boundaries of textured regions," *IEEE Trans. Geosci. Remote Sensing*, vol. 35, pp. 1367–1371, Sept. 1997.

Chin-Teng Lin received the B.S. degree in control engineering from the National Chiao-Tung University (NCTU), Hsinchu, Taiwan, R.O.C., in 1986, and the M.S.E.E. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1989 and 1992, respectively.

Since August 1992, he has been with the College of Electrical Engineering and Computer Science, NCTU, where he is currently a Professor of Electrical and Control Engineering. He has also served as the Deputy Dean of the Research and Development Office of NCTU since 1998. His current research interests are fuzzy systems, neural networks, intelligent control, human-machine interface, and video and audio processing. He is the co-author of *Neural Fuzzy Systems—A Neuro-Fuzzy Synergism to Intelligent Systems* and the author of *Neural Fuzzy Control Systems with Structure and Parameter Learning*. He has published around 40 journal papers in the areas of neural networks and fuzzy systems.

Dr. Lin is a member of Tau Beta Pi and Eta Kappa Nu. He is also a member of the IEEE Computer Society, the IEEE Robotics and Automation Society, and the IEEE Systems, Man, and Cybernetics Society. He has been an Executive Council Member of Chinese Fuzzy System Association (CFSA) since 1995 and the Supervisor of Chinese Automation Association since 1998. He was the Vice Chairman of IEEE Robotics and Automation, Taipei Chapter, in 1996 and 1997. He won the Outstanding Research Award granted by National Science Council (NSC), Taiwan, in 1997 and 1999, and the Outstanding Electrical Engineering Professor Award granted by the Chinese Institute of Electrical Engineering (CIEE) in 1997.

Yin-Cheung Lee received the B.S. and M.S. degrees in electrical and control engineering from the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C., in 1996 and 1998, respectively.

His current research interests include pattern recognition, neural networks, remote sensing, and image analysis. He is currently a System Researcher with Formosoft Software Inc., Taiwan.

Her-Chang Pu received the M.S. degree in automatic control engineering from Feng-Chia University, Taichung, Taiwan, R.O.C., in 1998. He is pursuing his Ph.D. degree in electrical and control engineering at National Chiao-Tung University, Hsinchu, Taiwan.

His current research interests are in the areas of artificial neural networks, fuzzy systems, pattern recognition, and machine vision.