Letters

Errata to "A Study of Cloud Classification with Neural Networks Using Spectral and Textural Features"

Bin Tian, Mukhtiar A. Shaikh, Mahmood R. Azimi-Sadjadi, Thomas H. Vonder Haar, and Donald L. Reinke

In the above paper,[1] Figs. 2, 5, and 6 were printed incorrectly. Below are the correct figures.

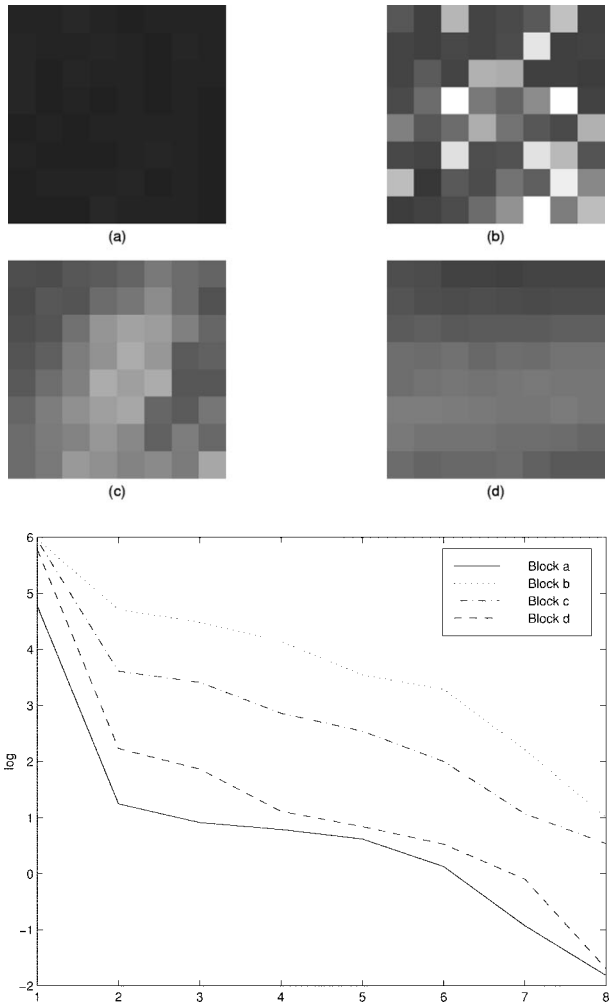


Fig. 2. Some typical 8 × 8 blocks in the GOES 8 satellite visible channel images and their corresponding SVD values in log domain. Block (a) corresponds to land while blocks (b)–(d) correspond to cumulus, stratocumulus, and thin cirrus, respectively.

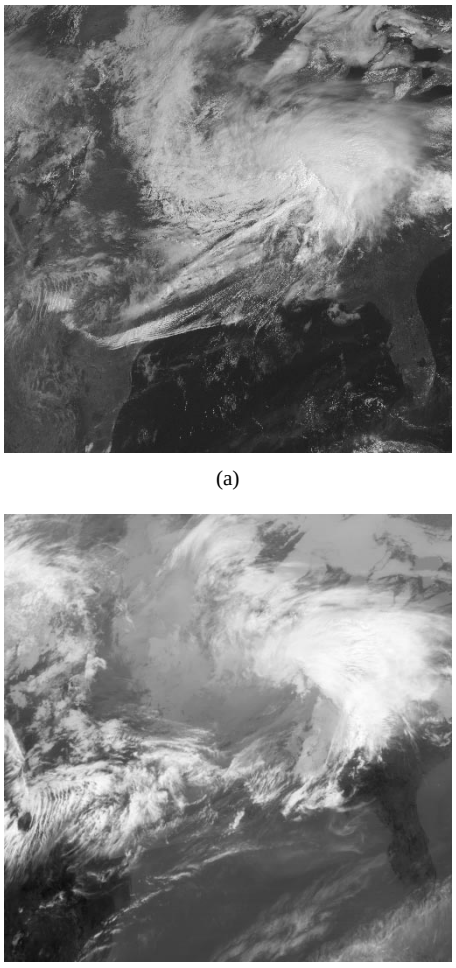
Manuscript received March 1, 1999.

B. Tian, M. A. Shaikh, and M. R. Azimi-Sadjadi are with the Department of Electrical Engineering, Colorado State University, Fort Collins, CO 80523 USA.

T. H. Vonder Haar and D. L. Reinke are with the Cooperative Institute for Research in the Atmosphere (CIRA), Colorado State University, Fort Collins, CO 80523 USA.

Publisher Item Identifier S 1045-9227(99)05313-8.

[1] B. Tian, M. A. Shaikh, M. R. Azimi-Sadjadi, T. H. Vonder Haar, and D. L. Reinke, *IEEE Trans. Neural Networks*, vol. 10, pp. 138–151, Jan. 1999.



(a)

(b)

Fig. 5. Original GOES 8 Image (Time 15:45 UTC). (a) Visible channel. (b) IR channel.

Fig. 6. Cloud/backgroud classes labeled by meteorologists.

A Study of Cloud Classification with Neural Networks Using Spectral and Textural Features

Bin Tian, Mukhtiar A. Shaikh, Mahmood R. Azimi-Sadjadi, *Senior Member, IEEE*, Thomas H. Vonder Haar, and Donald L. Reinke

***Abstract*—The problem of cloud data classification from satellite imagery using neural networks is considered in this paper. Several image transformations such as singular value decomposition (SVD) and wavelet packet (WP) were used to extract the salient spectral and textural features attributed to satellite cloud data in both visible and infrared (IR) channels. In addition, the well-known gray-level cooccurrence matrix (GLCM) method and spectral features were examined for the sake of comparison. Two different neural-network paradigms namely probability neural network (PNN) and unsupervised Kohonen self-organized feature map (SOM) were examined and their performance were also benchmarked on the geostationary operational environmental satellite (GOES) 8 data. Additionally, a postprocessing scheme was developed which utilizes the contextual information in the satellite images to improve the final classification accuracy. Overall, the performance of the PNN when used in conjunction with these feature extraction and postprocessing schemes showed the potential of this neural-network-based cloud classification system.**

***Index Terms*— Cloud classification, feature extraction, neural networks.**

I. Introduction

AUTOMATIC and accurate classification of clouds to enhance weather forecasting is one of the important applications studied in meteorology. Computer-based automatic classification systems would help the forecaster in several ways. The importance of this lies in the fact that large quantities of satellite imagery e.g., 25 GB/day from each geostationary operational environmental satellite (GOES) are generated every day. Extracting cloud field information from these images using visual/manual interpretation is a tedious and unreliable task and moreover the results are, to some extent, operator dependent. Therefore, highly efficient and robust cloud classification schemes are needed for automatic processing of satellite cloud imagery for climatological applications.

Manuscript received May 6, 1997; revised April 8, 1998 and October 8, 1998. This work was sponsored as part of DOD Geoscience (Phase II) program under Contract DAAH04 94 G0420.

B. Tian, M. A. Shaikh, M. R. Azimi-Sadjadi are with the Department of Electrical Engineering, Colorado State University, Fort Collins, CO 80523 USA.

T. H. Vonder Haar and D. Reinke are with the Cooperative Institute for Research in the Atmosphere (CIRA), Colorado State University, Fort Collins, CO 80523 USA.

Publisher Item Identifier S 1045-9227(99)00623-2.

In recent years, considerable research has been focused on the cloud classification area. A good review of the available schemes is provided by Pankiewicz [1]. Generally, two broad categories of cloud features are most commonly used in the cloud classification field: spectral and textural features. The first class of features, which plays a more important role for cloud classification, extracts the information on the cloud radiance in different spectral bands. Some of the most commonly used methods in this category include threshold-based schemes [2], histogram schemes [3], and multispectral approaches [4], [5]. The spectral features due to their physical importance (albedo, temperature) are proven to be effective and simple. However, they also encounter some problems because of the spectral similarities of certain features such as ice cloud and snow. Other factors, such as moisture in atmosphere, may also alter the multispectral characteristics and thus affecting the final judgement. The second category, i.e., textural features, distinguish certain types of clouds by the spatial distribution characteristics of gray levels corresponding to a region in one specific channel. While the spectral characteristics of clouds may change, their textural properties are often distinct and tend to be less sensitive to the effects of atmospheric attenuation or detector noise [6]. Most of the texture-based cloud classification methods in the past used statistical measures based on gray level cooccurrence matrix (GLCM) [7] and its variant, such as gray level difference vector (GLDV), gray level difference matrix (GLDM) and sum and difference histogram (SADH) [8], [9]. For example, Welch *et al.* [8] used GLCM for feature extraction to classify stratocumulus, cumulus, and cirrus clouds. Kuo *et al.* [9] used GLDV method to differentiate between clouds and ice/snow. Another important group of textural extraction schemes explores the frequency characteristics of images. Garand *et al.* [10] have examined the power spectrum of ocean cloud images while Gu and Duncan [11] evaluated autocorrelation, textural edgeness and the GLCM approach to obtain cloud textural information. They suggested a combination of textural measures in order to classify different cloud types. Gabor filter was also employed for cloud classification task by Lamei *et al.* [6] and Du [12]. Several comparative studies of these features have been conducted by Parikh [13], Gu [11], and Ohanian [14] where they suggested that GLCM provides the best features for cloud classification, while in [15] Gabor filters and Fourier features are recommended. There is no consistent and optimal feature

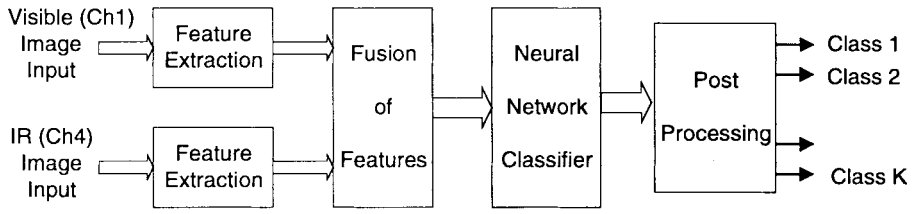


Fig. 1. Block diagram of the proposed cloud classification system.

extraction scheme determined at this time. Therefore, there is a need to develop efficient feature extraction schemes for cloud data analysis.

Another important issue in the cloud data analysis is the choice of an appropriate classifier. There are basically two types of classifiers; traditional classifiers which include: linear discriminant, maximum likelihood and k-nearest neighbor classifiers, and the neural-network classifiers which include: multilayer backpropagation neural network (BPNN), self-organizing map (SOM) and probability neural network (PNN), etc. Owing to the fact that the characteristics of clouds are highly variable and difficult to classify, neural network classifiers through their adaptive learning nature offer attractive and computationally very efficient alternatives. Lee *et al.* [16] used a three-layer BPNN for cloud classification of LANDSAT multispectral scanning system (MSS) data while PNN was examined by Bankert *et al.* [17] for classification of AVHRR imagery. In [18], traditional linear discrimination and two neural-network classifiers namely BPNN and PNN were comparatively studied for the classification of polar clouds and surface. The results showed that BPNN-based solution achieved the highest classification accuracy, while PNN falls behind within a very small accuracy range. It is worthy to mention that the BPNN-based scheme was extremely time consuming in the training phase compared to the one-pass noniterative PNN training approach [17]. On the other hand, the price one pays for this one-pass training approach is that potentially a very large network can be formed. This leads to increased storage and computational cost in the testing phase when compared with that of the BPNN-based solution. The unsupervised Kohonen SOM has also been examined for cloud classification [19]–[21].

In this paper, a neural-network-based cloud classification system is proposed (see the block diagram in Fig. 1). Several image transformation schemes namely singular value decomposition (SVD) and wavelet packets (WP's) were exploited to extract salient features of the cloud data. In addition, the conventional GLCM-based statistical features were also used for the purpose of benchmarking. The features from both the visible and IR channels were then combined together and fed to a neural-network classifier. However, these features do not remain consistent and vary at different time of the day and season. For example, certain types of clouds may look different in the visible channel due to the sun angle changes. On the other hand, land and low-level clouds can be heated up during the daytime thus looking different in the IR channel. Although the temporal issue is not considered in this paper, one possible solution is to adapt the neural network to accommodate these changes [22]. Consequently, it is important for the neural networks to have fast learning ability in order to adapt to the temporal changes. Based on this consideration, two neural-network classifiers are examined in this paper. The original PNN is improved by using Gaussian mixture models to provide much faster convergence than the BPNN-based solution. Using this scheme the computational cost in the testing phase is also greatly reduced. Owing to the fact that in most of the situations the truth maps of the clouds and background may not be available or reliable and further huge amount of satellite images are generally encountered, unsupervised Kohonen network solution is also exploited. In order to further improve the classification accuracy of the proposed cloud classification system, a postprocessing scheme is designed which utilizes the rich spatial contextual information in the satellite imagery. Finally, the effectiveness of the proposed system is analyzed on the GOES 8 satellite data, and various feature extraction schemes and classifiers were benchmarked.

II. Feature Extraction Schemes for Cloud Classification

Feature extraction is an important stage for any pattern recognition task especially for cloud classification, since clouds are highly variable and it is difficult to find reliable and robust features. According to the study in [24], trained meteorologists mainly rely on six criteria in visual interpretation of cloud images. These are brightness, texture, size, shape, organization and shadow effects. The brightness corresponds to the spectral features, which can generally be extracted rather easily. Textural features are those characteristics such as smoothness, fineness and coarseness or certain pattern associated with an image [7]. They reflect the local spatial distribution property in a certain region. The spectral and textural features are most widely used in automatic cloud classification. Other features such as size, shape and organization information attribute to the large-scale or global spatial distribution. Generally, these features are calculated after an image has been segmented [1]. The shadow information is the most difficult one for analysis and has not been fully exploited.

In the following, two feature extraction schemes, namely SVD and WP's are briefly reviewed. These schemes can extract features which contain contributions from both the spectral and textural aspects.

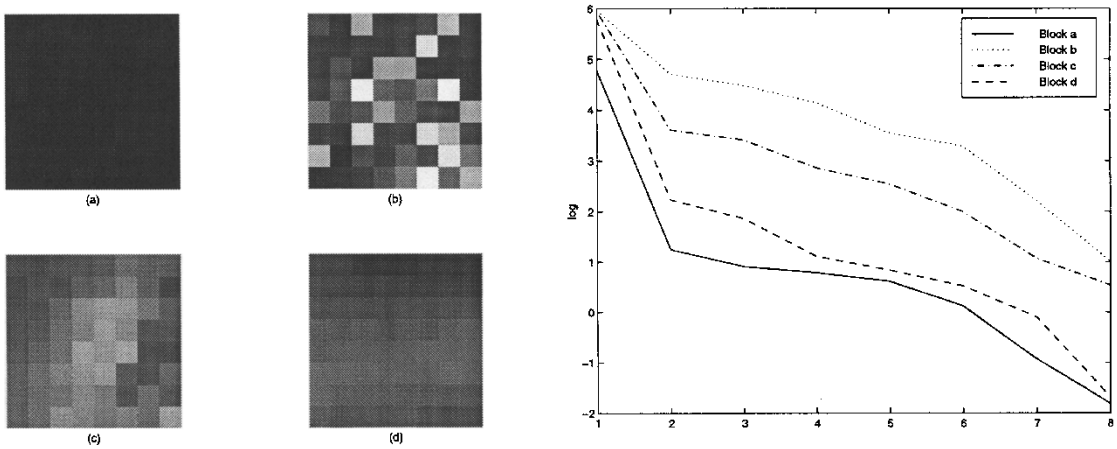


Fig. 2. Some typical 8×8 blocks in the GOES 8 satellite visible channel images and their corresponding SVD values in log domain. Block (a) corresponds to land while blocks (b)–(d) correspond to cumulus, stratocumulus, and thin cirrus, respectively.

A. Singular Value Decomposition (SVD) Scheme

SVD is a very powerful tool in image restoration, power spectrum estimation and data reduction compression areas. It is one of the best candidates for dealing with sets of equations or matrices that are either singular or ill-conditioned as it provides quantitative information about the structure of the system or linear equations. In image processing, SVD has been shown to have excellent energy-packing ability [25].

Let us consider an $N \times M$ image X where $N \geq M$. It is possible to represent this image in the r-dimensional subspace where r is the rank of X, and $r \leq M$. Let $U = XX^T$ and $V = X^TX$, be nonnegative, symmetric matrices with same eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_r$, where it is assumed that $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_r$. These eigenvalues are called the singular values of X. Now, if we form matrices Ψ and Φ from the corresponding eigenvectors of U and V, then X can be diagonalized as

$$X = \Psi \Lambda \Phi^T \tag{1}$$

where Λ is the diagonal matrix, i.e., $\Lambda = \text{diag}[\sqrt{\lambda_1}, \sqrt{\lambda_1}, \cdots \sqrt{\lambda_r}]$. Alternatively, we have

$$\Lambda = \Psi^T X \Phi. \tag{2}$$

Basically, the eigenvalues or singular values $\{\lambda_i\}_{i=1}^r$ represent the energy of image X projected on each subspace. The singular values and their distribution which carry certain useful information about the content of X, vary drastically from image to image. For an image with random textural content, e.g., white noise, its energy will spread over all the singular values. On the other hand, for a smooth image with no texture, the first singular value will be dominant while all the others are almost zero. Fig. 2 shows some typical blocks of size 8 $\times$ 8 in the GOES 8 satellite visible channel image and their corresponding eight singular values. In most of the cases, the first (largest) singular value roughly corresponds to the mean of the image thus closely relating to the spectral features, while all the other singular values provide detailed information about the spatial content of the image which somewhat relates to the textural features. Another important property of the SVD features is that they are not sensitive to the orientation of the texture, since the image X and its transpose have the same set of singular values.

Overall, singular values provide the energy information of the image as well as the knowledge of how the energy is distributed over the subspace. They contain the contributions from both the spectral and textural aspects of the image.

B. 2-D Wavelet Transform (WT) and Wavelet Packets (WP) Schemes

The 2-D wavelet transform (WT) is a straightforward extension of the 1-D case [26]-[29]. It can be viewed as performing the 1-D WT first along the x direction then along the y direction. In the first level of decomposition, any given image $x(m, n)$ is decomposed into one low-pass approximation and three added detail images which contain high-frequency information of the image in the vertical, horizontal and diagonal directions [28]. Like in the one-dimensional (1-D) case, such operation can be repeatedly applied to each subimage leading to the two-dimensional (2-D) wavelet packets (WP) decomposition [30]. 2-D WP decomposition provides a powerful tool to analyze the content of images with a good localization property both in the spatial and frequency domains. When an image goes through WP decomposition process, a full-structure tree of subimages may be formed. The subimages in the same decomposition level provide the multiple looks of the original image at different frequency band. At the higher levels, the resolution in the frequency domain will increase while the spatial resolution is gradually degraded. A block of size $M \times M$ in the original image will correspond to a reduced size block $(M/2^i) \times (M/2^i)$ at the decomposition level i. For regions with different textural content, their spectra exhibit different characteristics which in turn will be reflected in the energy distribution of the subimages through the tree structure [30], [31]. Consequently, the energy in all the subimages can be computed to form a feature vector. However, in order to remove the redundancy in the features, a feature selection procedure is needed to identify only a few nodes in the

full tree which possess the best discriminatory ability for the subsequent classification. This feature selection process is briefly described in Section IV-B.

III. Neural-Network Classifiers and Post Processing Scheme

Two different neural-network classifiers namely Kohonen SOM and PNN were studied for the cloud classification problem. The reader is referred to [32] and [33] for review on Kohonen SOM. In the following section the PNN [34], [35] is briefly discussed. The renowned expectation maximization (EM) [36] is used for the efficient training of PNN. In order to further improve the classification accuracy, a postprocessing approach is also developed to utilize the rich spatial context information in the satellite imagery.

A. Probability Neural Network (PNN)

The original PNN, which was proposed by Specht in [34], is a direct neural-network implementation of the Parzen nonparametric probability density function (PDF) estimation [37] and Bayes classification rule. It can be considered as a special case of the radial basis function neural networks. Comparing with the well-known BPNN-based solution, PNN has a very fast one-pass learning scheme, and can be retrained or updated on-line. It was also reported in [35] that PNN and BPNN have comparable generalization ability in classifying unknown patterns.

For an input pattern $\boldsymbol{x}$, the so-called optimum Bayesian classification strategy is to make the decision in such a way that the "expected risk" is minimized. For the "0-1" cost function [38] which is generally used for pattern recognition, the Bayes classifier will lead to the maximum *a posteriori* (MAP) classifier [37], i.e.,

$$C(\boldsymbol{x}) = \underset{c_i}{\operatorname{argmax}}\, P(\boldsymbol{x}|c_i)P(c_i) \qquad i = 1, 2, \cdots, K \tag{3}$$

where $C(\boldsymbol{x})$ is the class of input $\boldsymbol{x}$ which belongs to $\{c_i, i = 1, \cdots K\}, P(c_i)$ is the *a priori* class distribution, and $P(\boldsymbol{x}|c_i)$ is the *a priori* conditional distribution for class c_i. Generally, the unknown class distribution, $P(c_i)$, is decided by analyzing the physical nature of the problem, and is assumed to be uniformly distributed here without loss of generality. The key issue for the implementation of this Bayesian classifier is to extract the conditional distributions from the training data set. In [37], Parzen proved that $P(\boldsymbol{x}|c_i)$ can be estimated from all the samples in the training set which belong to the class c_i. When a Gaussian kernel is adopted, the Parzen PDF estimator can be represented by [34]

$$p(\boldsymbol{y}|c_i) = \frac{1}{N_i(2\pi)^{d/2}\sigma^d} \sum_{j=1}^{N_i} \exp\left[-\frac{(\boldsymbol{y} - \boldsymbol{x}_i^{(j)})^T(\boldsymbol{y} - \boldsymbol{x}_i^{(j)})}{2\sigma^2}\right] \tag{4}$$

where N_i is the number of samples in the training set belong to class $c_i, \boldsymbol{x}_i^{(j)}$ represents the jth sample belonging to class c_i, d is the input vector dimension and is called the "smoothing factor."

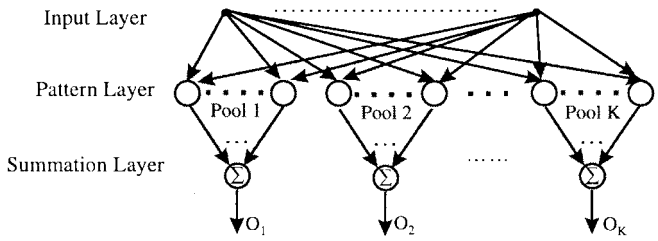


Fig. 3. The structure of PNN.

The original PNN structure proposed by Specht in [34], is a direct implementation of the above estimator. It consists of three feedforward layers: input layer, pattern layer, and summation layer [34] which are shown in Fig. 3. The input layer accepts the feature vectors and supplies them to all the neurons in the pattern layer. The pattern layer consists of K pools of pattern neurons, where K is the number of patterns. In each pool $i, i = 1, \cdots, K$, there are N_i number of pattern neurons. For the input feature vector $\boldsymbol{y}$, the output of each pattern neuron is

$$f(\boldsymbol{y}; \boldsymbol{w}_i^{(j)}, \sigma) = \frac{1}{N_i(2\pi)^{d/2}\sigma^d} \exp\left[-\frac{(\boldsymbol{y} - \boldsymbol{w}_i^{(j)})^T(\boldsymbol{y} - \boldsymbol{w}_i^{(j)})}{2\sigma^2}\right] \tag{5}$$

where $\boldsymbol{w}_i^{(j)}$ is the weight vector of the jth neuron in the ith pool, and the nonlinear function $f(\cdot)$ represents the activation function of the neurons. There are totally K neurons in the summation layer where the ith neuron, $i = 1, \cdots, K$, forms the weighed sum of all the outputs from the ith pool in the pattern layer. The weights of the summation layer are determined by the decision cost function and the *a priori* class distribution. For the "0-1" cost function and uniform *a priori* distribution, the weights will be one for all the neurons in the summation layer. For the input pattern $\boldsymbol{y}$ of unknown class, the final decision will be made by a simple comparison of all the outputs, i.e.,

$$\boldsymbol{y} \in c_k, \quad \text{if } O_k > , O_i, \qquad i \neq k, \quad i, k \in [1, K].$$

Comparing (4) and (5), it can easily be observed that the output of PNN will be proportional to the *a posteriori* probability when $\boldsymbol{w}_i^{(k)} = \boldsymbol{x}_i^{(k)}$ (under the condition that the weights of the summation layer are properly assigned based on the *a priori* class distribution). So the training phase of PNN is very straightforward. For each new training sample $\boldsymbol{x}$ belonging to class c_i, the training process is nothing but adding a new neuron in the ith pool of the pattern layer, with the weight vector equal to $\boldsymbol{x}$.

One drawback of the original PNN is that potentially a very large network will be formed since every training pattern will need to be stored. This leads to extensive storage and computational requirements during the testing phase. One natural way to improve the PNN is to reduce the number of neurons, i.e., use fewer kernels but place them at optimal places. The scheme in [39] uses learning vector quantization (LVQ) schemes for clustering the training samples. In [40], Streit *et al.* improved the PNN by using finite Gaussian mixture

model. This scheme is adopted in this paper and briefly described below.

For any class $c_i, i = 1, \cdots, K$, suppose that the class conditional distribution is modeled approximately by a Gaussian mixture, i.e.,

$$p(\boldsymbol{x}|c_i) = \sum_{j=1}^{M_i} \pi_{ji} p_{ji}(\boldsymbol{x}; \mu_{ji}, \Sigma_{ji}) \tag{6}$$

where M_i is the number of Gaussian components in class c_i and π_{ji}'s are the weights of the components which satisfy the constraint $\Sigma_{j=1}^{M_i}\ \pi_{ji} = 1, p_{ji}(\boldsymbol{x}; \mu_{ji}, \Sigma_{ji})$ denote the multivariate Gaussian density function of the jth component in class c_i and we have

$$\begin{aligned} p_{ji}(\boldsymbol{x}; \mu_{ji}, \Sigma_{ji}) &= \frac{1}{(2\pi)^{d/2}|\Sigma_{ji}|^{1/2}} \\ &\quad \cdot \exp\left\{-\frac{1}{2}(\boldsymbol{x} - \mu_{ji})^T \Sigma_{ji}^{-1}(\boldsymbol{x} - \mu_{ji})\right\} \end{aligned} \tag{7}$$

where μ_{ji} and Σ_{ji} are the mean vector and covariance matrix of the jth Gaussian component for class c_i, respectively. The Gaussian mixture model described in (6) and (7) can also be easily mapped to the PNN structure shown in Fig. 3. Since M_i is generally much smaller than the number of training samples belong to class i, N_i, the pattern layer of the PNN is therefore substantially simplified than its original version. The price paid for this simplification is that the noniterative training procedure will no longer exist. Instead, the weights of the PNN i.e., the parameter sets of the mixture model for each class, need to be estimated from the training data set.

Let $\lambda_i = \{\pi_{ji}, \mu_{ji}, \Sigma_{ji}\}_{j=1}^{M_i}$ denote the parameter set used to describe the mixture model of class c_i and $\Lambda = \{\lambda_i\}_{i=1}^{K}$ denote the whole parameter space for the PNN. There are several criteria available which can be used to estimate Λ. If we assume that the parameters in Λ are unknown fixed quantities, the ML estimation method is a suitable choice. Now suppose that the training samples drawn independently from the feature space form the set X, which can be further separated into K subset $X_i, i = 1, \cdots, K$, in which all the samples belong to class c_i. The ML estimation of the parameter set Λ is then given by

$$\Lambda^* = \arg\max_{\Lambda} \prod_{i=1}^{K} \prod_{\boldsymbol{x} \in X_i} p(\boldsymbol{x}|c_i; \Lambda). \tag{8}$$

For the computational efficiency, generally we will maximize the equivalent log-likelihood, i.e.,

$$\begin{aligned} \Lambda^* &= \arg\max_{\Lambda} \sum_{i=1}^{K} \sum_{\boldsymbol{x} \in X_i} \log[p(\boldsymbol{x}|c_i; \Lambda)] \\ &= \arg\max_{\Lambda} \sum_{i=1}^{K} \sum_{\boldsymbol{x} \in X_i} \log[p(\boldsymbol{x}|c_i; \lambda_i)]. \end{aligned} \tag{9}$$

The last step in the above equation is arrived at based upon the assumption that the conditional probability of class c_i is totally decided by the parameter set of that class, λ_i and not affected by the parameter set of the other classes. The maximization of the log-likelihood function can be done using the gradient descent scheme. However, a more efficient way is to use the well-known EM approach [36]. This method which is used here helps to achieve the ML estimation via iterative computation when the observations can be viewed as incomplete data. There are two major steps in this approach: the estimation (E) step and maximization (M) step. The E step extends the likelihood function to the unobserved variables, then computes an expectation with respect to them using the current estimate of the parameter set. In the M step, the new parameter set is obtained by maximizing the resultant expectation function. These two steps are iterated until the convergence is reached. The reader is referred to [36] for the detail description on EM algorithm. Generally, this training process converges much faster than the BPNN-based approach. To improve the final classification accuracy, a postprocessing method is developed in the following section.

B. Post Processing

In the above discussion each block in the satellite images is processed separately without considering the spatial neighborhood context. However, there is rich context information in these images i.e., some classes are likely to span over a region instead of appearing in isolated blocks. Proper utilization of such contextual information can help to improve the final classification accuracy. Therefore, a postprocessing scheme similar to that in [42] is developed here in order to take advantage of the contextual information.

For any block $\boldsymbol{r}$ in the image, where $\boldsymbol{r} = (k, l)$ is the coordination vector of that block, $\boldsymbol{x}(\boldsymbol{r})$ denotes the corresponding feature vector while $C(\boldsymbol{x}(\boldsymbol{r}))$ refers to its physical class which belongs to the label set $\{c_1, c_2, \cdots, c_k\}$. Moreover, we can define a spatial neighborhood for $\boldsymbol{x}(\boldsymbol{r})$ as $H(\boldsymbol{r}) = \{\boldsymbol{x}(\boldsymbol{r} + \boldsymbol{v})|\boldsymbol{v} \in \Psi\}$ where Ψ is the neighborhood set. Fig. 4(a) shows one typical example of Ψ. $C_H(\boldsymbol{r}) = \{C(\boldsymbol{r} + \boldsymbol{v})|\boldsymbol{v} \in \Psi\}$ is used to represent the class label of $H(\boldsymbol{r})$. For simplicity we assume that all the contextual information for block $\boldsymbol{r}$ is conveyed by the classes of its spatial neighborhood, i.e., $C_H(\boldsymbol{r})$. Now, we can define a spatial context classifier for post processing of the images, i.e.,

$$\hat{C}(\boldsymbol{x}(\boldsymbol{r})) = \arg\max_{c_i} P(c_i|\boldsymbol{x}(\boldsymbol{r}), C_H(\boldsymbol{r})) \qquad i = 1, \cdots, K. \tag{10}$$

Comparing with the general classifier in (3), this new classifier takes into account not only the feature vector in block $\boldsymbol{r}$ but also the class information in its neighborhood, i.e, the context information. For simplicity, we drop the block position variable $\boldsymbol{r}$ in the sequel. Using Bayes rule, we get

$$P(c_i|\boldsymbol{x}, C_H) = \frac{P(\boldsymbol{x}|c_i, C_H)P(c_i|C_H)}{P(\boldsymbol{x}|C_H).}$$

It is reasonable to assume that the distribution of the feature vector is solely dependent on its own class label and not affected by its spatial neighborhood, i.e., $P(\boldsymbol{x}|c_i, C_H) = P(\boldsymbol{x}|c_i)$. Also, notice that $P(\boldsymbol{x}|C_H)$ is the same for all classes,

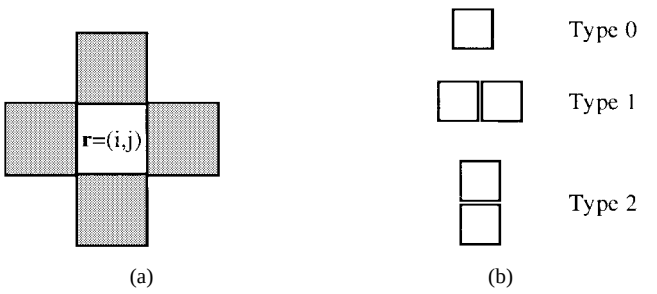


Fig. 4. (a) Block $\boldsymbol{r}$ and its spatial neighborhood blocks (shaded) $\boldsymbol{r}+\boldsymbol{v}$. In this case, $\boldsymbol{v} \in \Psi = \{(-1,0),(0,-1),(0,1),(1,0)\}$. (b) Three types of cliques for the spatial neighborhood in 4(a).

then the spatial context classifier in (10) becomes

$$\hat{C}(x) = \arg \max_{c_i} P(\boldsymbol{x}|c_i)P(c_i|C_H). \tag{11}$$

The feature vector conditional probability $P(\boldsymbol{x}|c_i)$ can be calculated using the PNN. Moreover, $P(c_i|C_H)$, which describes the spatial context conditional probability, can be modeled by Gibbs distribution [43]. The Gibbs distribution is equivalent to Markov random field modeling and has been widely used in the image processing area. The spatial context conditional probability can be expressed in the form of [43]

$$P(c_i|C_H) = \frac{1}{Z}e^{-U(c_i,C_H)} \tag{12}$$

where Z is a normalization constant and U is called the energy function given by

$$U(c_i, C_H) = \sum_{W \in N_{CL}} V_W(c_i, C_H) \tag{13}$$

$V_w(c_i, C_H)$ is called the "potential function" of the class configuration $\{c_i, C_H\}$ on a clique W. A so-called clique is a set of block locations where any pair of distinct indices in the set are neighbors to each other. For the spatial neighborhood shown in Fig. 4(a), there are totally three types of cliques that are shown in Fig. 4(b). N_{CL} denotes the set of all possible cliques for current class configuration. The form of potential function, V_w, is very important since it specifics the context information between adjacent blocks. In this paper, the potential function suggested in [42] was adopted. It is shown in [42] that after some simplifications the spatial context conditional probability can be given by

$$P(c_i|C_H) = Ae^{-2\beta(m-2)} \tag{14}$$

where m is the number of occurrence of class c_i in C_H, A is a constant which is the same for all classes and β is a parameter describing the interactions between classes of adjacent blocks. $\beta = 0$ corresponds to the situation where no context information is available for the classification. On the other hand, a larger value of β indicates stronger class correlation in the neighborhood. Generally, homogenous regions are more favored in the postprocessing.

Another issue for implementing the classifier in (11) is that the class of neighborhood, C_H, is unknown in practice. This can be circumvented by performing the postprocessing scheme iteratively. In this case, the satellite image is initially classified using the PNN and the initial results are recorded. In the post processing, the initial classification results are used to replace the unknown neighborhood class and a new classified image is generated using (11). This process can be repeated and at each iteration the resultant image of the previous iteration is used to provide the neighborhood class information. The process stops when the class differences between the resultant images in two consecutive iterations are negligible. In our experiment, this error goal was set to five blocks per image.

IV. Simulation Results and Discussions

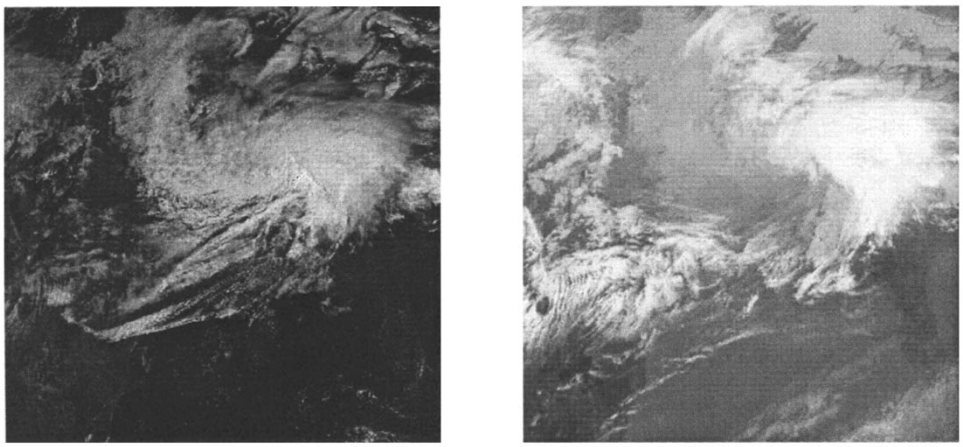
In this section, the proposed cloud classification system is examined on the GOES 8 satellite image data set. This data set and the process of labeling are first introduced. The preprocessing procedure of these data, feature extraction and selection processes are also described. In order to select the most suitable classifier and the set of features for this cloud classification problem, several neural-network paradigms and different sets of features are benchmarked without considering the context information. Finally, the postprocessing scheme is implemented and the final results are provided.

A. GOES-8 Satellite Imagery

The cloud data analyzed in this study was obtained from the GOES 8 satellite that carries five channel sensors. This study was carried out only using two channels, namely channel 1 (visible) and channel 4 (IR) since these channels are commonly used in almost all the other meteorological satellites. Nevertheless, the system can be easily expanded to accommodate more channels, if necessary. Totally six pairs of images were used in these tests. Since the purpose of this study was to examine the performance of cloud classification system for certain areas without considering the temporal changes of the data. These images covered the same geographical regions and were obtained during almost the same period of the day. Specifically, they were collected on May 1, 1995 and May 5, 1995 from 15:45 UTC[1] to 17:45 UTC. One typical image pair is shown in Fig. 5. These images with spatial resolution of 512×512 pixels cover mid-west and most of the eastern part of the United States extending from the Rocky Mountains to the Atlantic coast. The region spreads over mountains, plains, lakes, and coastal areas where clouds have some specific features. Lake Michigan is in the upper right corner and Florida is located in the lower part with Gulf of Mexico in the lower center of the image. There are a variety of cloud types in this image. For example, there are some thin cirrus in the left middle part, cirrostratus in the right middle part and low/middle-level clouds (stratocumulus and altostratus) in the center part as well as water and land areas.

Since the ground truth is not available and reliable, two meteorologists were asked to identify all possible cloud types as well as the background areas based on the visual inspection and other related information. This was accomplished with the aid of a computer software package developed solely for this purpose. Totally ten cloud/background classes were

[1] UTC stands for Universal Time Code.



(a) (b)

Fig. 5. Original GOES 8 Image (Time 15:45 UTC). (a) Visible channel. (b) IR channel.

TABLE I
NUMBER OF BLOCKS FOR EACH CLOUD/BACKGROUND CLASS IN THE LABELED DATA SETS

Warm Land (WLnd)	Cold Land (CLnd)	Warm Water (WWtr)	Cold Water (CWtr)	Stratus (St)	Low Cumulus (Cu)	Strato-Cumulus (Sc)	Alto-Stratus (As)	Cirro-Stratus (Cs)	Cirrus (Ci)
826	534	540	92	410	823	1185	569	1351	4259

found, which are: warm land (WLnd), cold land (CLnd), warm water(WWtr), cold water (CWtr), stratus (St) cumulus (Cu), altostratus (As), stratocumulus (Sc), cirrus(Ci) and cirrostratus (Cs). It should be mentioned that the way meteorologists label images differs from that of the neural-network classification system. That is, instead of labeling each block, they first try to identity certain regions and then assign that whole area into one category. As a result, it is possible that some blocks in that region may belong to different classes since the labeling is done based upon global information. Additionally, some regions have mixed cloud types hence making the classification task difficult. The regions labeled by the experts were further divided into small blocks of size 8×8 corresponding to an area of size 32 km $\times$ 32 km. The labeled results of the satellite image pair in Fig. 5 are shown in Fig. 6. All the labeled blocks formed the data set for our benchmarking. Half of the blocks randomly drawn from this data set were used for the training of the classifier while the rest were used for the testing and performance analysis. Table I gives the total numbers of blocks for each cloud/background type.

Fig. 6. Cloud/backgroud classes labeled by meteorologists.

B. Feature Extraction and Selection

In order to reduce the dimensionality of the data and extract the pertinent features for cloud classification, several feature extraction methods were investigated in this study. Besides the SVD features and WP features discussed above, the widely used GLCM-based statistical features and spectral features were also implemented for the sake of comparison.

To remove the redundancy in these features, a feature selection process was performed. This not only reduces the training time for the classifier but also may help to improve the classification accuracy due to the finite sample size. The sequential forward floating selection (SFFS) algorithm used in [8] was employed which provides a deterministic single solution leading to a suboptimal feature set. At every step, this method searches the remaining feature space iteratively and selects one feature at a time to make the new enlarged feature set which maximizes the "distance" among different classes. In this paper, the Bhattacharya distance was used to measure the class separability. Jain *et al.* [44] reported optimal performance for SFFS scheme among all the other methods they tested.

For the SVD approach, a total of 16 singular values were extracted, eight from every 8×8 block in each channel. However, after the feature selection process, only six features were chosen for the subsequent classification process. These

TABLE II
CONFUSION MATRIX FOR ORIGINAL PNN CLASSIFIER USING SVD FEATURE (%). (ROW—NEURAL-NETWORK RESULTS, COLUMN-EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 83.4%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	98.8	0.2	0.5	0	0	0	0	0.5	0	0
CLnd	0.4	96.6	0.4	0	0	1.9	0	0.7	0	0
WWtr	0	0.7	98.9	0	0	0.4	0	0	0	0
CWtr	0	8.7	0	80.4	0	0	2.2	8.7	0	0
St	0	0.5	0	0	82.9	4.4	10.2	1	0	1
Cu	2.2	1.7	0.7	0.2	2.9	78.3	1.9	2.2	0	9.7
As	0	0	0	0	12.3	1.8	59.5	4.2	1.1	21.1
Ci	1.2	0.8	0.5	0.3	0.6	0.9	1.9	84.2	8.8	0.8
Cs	0.3	0.1	0	0	0.3	0.1	0.7	18.1	80	0.3
Sc	0.8	0	0.2	0	3	7.1	10.1	2	0.5	76.2

correspond to the first, third, and fifth singular values in the visible channel and the first, third, and sixth ones in the IR channel.

The 2-D WP approach was applied to the whole image. Haar wavelet [28] was adopted here for its extreme simplicity and computational efficiency. A three-level decomposition was carried out and a full tree with a total of 85 nodes was generated for each channel. The energy of the region corresponding to 8×8 blocks in the original image was calculated for each subimage which formed the primary feature set. After feature selection process, totally ten energy components were chosen; seven components from the visible channel and three from the IR channel. Specifically, these were the energy in subimages [0,1], [1,1], [1,2], [2,3], [2,11], [3,1] and [3,42] of the visible channel and [0,1], [1,1] and [3,1] of the IR channel, where the notation $[i, j]$ represents the jth subimage in the ith decomposition level.

For the GLCM scheme [8], the gray level resolution was reduced to only 16 in order to reduce the storage and computational requirements. Four most popular textural features, namely contrast, correlation, entropy, and homogeneity, were calculated for each image setting the interpixel distance equal to one in all the four orientations, i.e., 0, 45, 90, and 145°. Finally, directionality was suppressed by averaging the extracted features over four directions to produce isotropically averaged measures. These four GLCM features were extracted from both the visible and IR images and then fused together to form a combined feature vector. Since both the SVD and WP features contain the spectral information, for the sake of fairness in benchmarking the mean of the block in both channels were also included in the feature vector. The resultant feature vector is of dimension ten and no further feature selection process was applied. Additionally, a feature set which only includes the spectral information was generated which contained the mean of each block in the two channels.

C. Comparison Study of Different Classifiers

The performance of different kinds of neural-network classifiers namely PNN and SOM were first examined for their classification accuracy. The extracted SVD features were used for this comparison due to the simplicity of this scheme. Note that since only parts of the images labeled by the meteorologist were used for benchmarking and performance comparison, the accuracy rates alone may not fully reflect the performance of the classifiers. Visual inspection of the final results is also an important criterion that was used in the evaluation process. What follows is the detailed discussion of the results and performance of each classifier.

1) PNN Classifier: Both the traditional PNN and the modified PNN (Section III-A) were implemented. The traditional PNN was trained by the one-pass noniterative scheme, i.e, adding neurons in the pattern layer with weights equal to the training samples. The smoothing parameter was determined experimentally to be $\sigma = 7$. It was found that classification result is not sensitive to the choice of this parameter as long as it is not too small $(\sigma < 1)$. Table II shows the classification results in terms of the confusion matrix for the original PNN. The results indicate that good classification (over 90%) were obtained for most background classes except cold water (CWtr), for which the poor performance is partly attributed to the lack of enough samples as well as the similarities between the cold land and cold water types. For the high level cloud classes, i.e., cirrus and cirrustratus, it was observed that most misclassifications occurred between them. This is expected since some cirrus blocks exhibit the features similar to those of cirrustratus if they are thick and smooth enough. In the data set there are also some very thin cirrus samples which span over the Gulf of Mexico (lower-left corner in Fig. 5). Owing to the thin nature of the cirrus, the region has the averaged temperature of both the cirrus and warm water, i.e., it is much warmer than the general cirrus and looks similar to the cold water and some of the low clouds. This also causes misclassifications. The classification accuracy for the low and middle level clouds, namely stratus (St), cumulus (Cu), stratocumulus (Sc), and altostratus (As) ranges around 60–83%. Some of the most confusing pairs include St versus As, Cu versus Sc, and especially Sc versus As. Although the middle level (As) clouds generally are colder than the lower level clouds, such difference may become less prominent if considering the background temperature variation. For example, some Altostraturs samples in the Gulf of Mexico may have similar temperature as the low clouds in the far North. On the other hand, the texture of (Sc) are somehow in between those of (St)/(As) and Cu types. Some small blocks of Sc have smooth texture like St/As while other blocks are very similar to Cu.

TABLE III
CONFUSION MATRIX FOR THE MODIFIED PNN CLASSIFIER USING SVD FEATURES (%). (ROW—NEURAL-NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 83.8%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	97.3	0.7	0	0	0	1.5	0	0.5	0	0
CLnd	1.9	92.9	0	0	0	4.1	0	1.1	0	0
WWtr	0.7	1.5	90	0	0.7	3.3	0	3.7	0	0
CWtr	0	0	0	84.8	0	2.2	0	10.9	2.2	0
St	0	0	0	0	74.6	4.4	14.6	0.5	0	5.9
Cu	0	0	0	0	1	84.2	1.9	1.9	0.5	10.5
As	0	0.4	0	0	3.2	3.2	66.2	3.9	0.7	22.5
C_i	0.5	0.8	0.3	0.2	0.2	1	1.5	85.5	9.2	0.8
Cs	0	0	0	0.1	0	0.1	1.2	15.6	82.5	0.4
Sc	0.2	0	0	0	1	8.6	13.9	1.5	0.5	74.3

Fig. 7. Classification results using modified PNN with SVD featuers.

For the modified PNN, several network structures with different number of Gaussian components in each class were examined. A pruning operation was also performed during the training process if the weight of certain Gaussian component was less than 0.005. The best structure was chosen based on the classification accuracies, however, this does not imply that the structure is theoretically optimal. Table III presents the classification accuracies of the modified PNN. It was found that this PNN achieved slightly better results than that of the original PNN. The overall classification rate has increased from 83.4 to 83.8%. Moreover, the structure of neural network is greatly simplified. There were totally 5297 pattern neurons in the original PNN in contrast to only 94 neurons in the modified PNN. Consequently, the computational and storage requirements in the testing phase are greatly reduced. The color-coded image based on SVD features and the modified PNN classifier is shown in Fig. 7. Visual inspection of this result indicates that different cloud/background areas are well-separated. This agrees with the meteorologist labeling with the exception of some isolated error blocks. Other error blocks occurred at the border of cloud and background areas where features in one block are the mixture of different physical classes. Overall, the resultant classified image show more promise of the system than indicated by the classification rates.

2) Kohonen SOM: SOM is an unsupervised network which clusters the inputs into certain prespecified number of unknown categories. The optimal number of output neurons (categories) is highly problem dependent and difficult to determine (exactly) *a priori*. In this study, three SOM networks, referred to as SOM1, SOM2 and SOM3, with different number of output neurons, namely, 10, 30, and 50 were tested. For each SOM network, different neighborhood functions were experimented, and the best one was chosen based on the classification accuracy. For SOM1, this was 5×2 neighborhood while for SOM2 and SOM3 these were 1×30 and 10×5, respectively. Note that mxn neighborhood implies that the neurones are arranged in a mxn grid in two dimensions. The training data set for the SOM consists of 8192 samples which were randomly chosen from the six image pairs. Each SOM was initialized using the convex combination method [46]. The training process of SOM was separated into two phases. In the global ordering phase which corresponds to the first 1000 epochs, the learning rate decreased linearly from 0.9 to 0.45, while the size of the neighborhood was kept large. Such a large learning rate and neighborhood size can help to form the topological ordering network. In the fine tuning phase which takes 19 000 epochs, the learning rate decays exponentially from 0.45 to zero while the neighborhood is linearly reduced until finally will contain just the winner neuron after 2000 epochs. After the training is completed, the output neurons were further mapped to the corresponding cloud/background class based on a labeling data set. This labeling set is a subset of the training data set for PNN classifier. The reason being in the original PNN training set the number of samples for each class was not equal, e.g., there were only 92 samples for the cold water class in contrast to 4259 samples in the cirrus class. If this set was used for labeling the SOM networks, the resultant SOM would have been significantly biased to the cirrus class while the cold water type might have been neglected. In order to circumvent this problem, we require that the maximum number of samples in each class of the labeling set does not exceed 500. For those classes in the old PNN training set which had more samples than this limit, we randomly chose 500 samples. All SOM networks were labeled using this new data set.

The classification rates for the three SOM topologies are given in Table IV. The results of SOM1 were very poor mainly because ten neurons was inadequate to represent all the clusters in the feature space. For example, there was no neuron to represent the CWtr, St, and As classes so the classification rates for these classes were zero! Increasing the number of

TABLE IV
CLASSIFICATION ACCURACY FOR THREE SOM'S WITH OUTPUT NEURONS 10, 30, AND 50, RESPECTIVELY (%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc	Overall Accuracy
SOM1	96.9	43.1	85.2	0	0	60.8	0	68.8	72.9	82.8	65.0
SOM2	94.9	84.6	98.9	76.1	37.6	61.3	27.5	73.2	74.7	71.5	72.0
SOM3	98.8	90.6	97.8	78.3	55.6	68.6	23.2	71.6	80.6	80.7	74.8

TABLE V
CONFUSION MATRIX FOR SOM2 WITH SVD FEATURES (%). (ROW—NEURAL NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 72.0%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	94.9	1.7	3.4	0	0	0	0	0	0	0
CLnd	0	84.6	1.9	1.9	0	10.1	0	1.5	0	0
WWtr	1.1	0	98.9	0	0	0	0	0	0	0
CWtr	0	15.2	0	76.1	0	0	0	8.7	0	0
St	0	2.4	0	0	37.6	24.9	22	4.9	0	8.3
Cu	10.7	10.5	1.9	0.2	0.5	61.3	1.7	4.6	0	8.5
As	0	0	0	0	5.3	12.7	27.5	0.7	0.4	53.5
Ci	2	2.1	3.3	1.6	3	3.8	0.7	73.2	8.4	1.8
Cs	0	0	0	0	2.1	0.1	4.3	17.9	74.7	0.9
Sc	1.2	0.2	0.7	0	4.6	10.8	10.3	0.8	0	71.5

output neurons to 30 certainly help to improve the overall classification accuracy from to 65.0 to 72.0% since for each class there was at least one neuron to represent it. However, there was only a slight improvement when the number of neurons was further increased to 50. Although SOM3 with 50 neurons can more accurately represent the clusters in the feature space than the SOM2 with 30 neurons, the formed clusters may not necessarily correspond to the physical labels used by meteorologist. In other words, the physical classes may not form distinct clusters in the feature space. This explains the reason for no significant improvement in the classification accuracy.

Comparing to PNN, SOM did not achieve high classification accuracy owing to the unsupervised learning nature of this network. The confusion matrix for SOM2 is provided in Table V. The color-coded image classified using SOM2 and the SVD features is shown in Fig. 8. Visual inspection of this image can easily reveal significant errors in classification. One benefit in using SOM network is that the classified image will more likely to have very smooth mutually exclusive class boundaries comparing with other classifiers. This is primarily due to the fact that each "natural cluster" is mapped to one cloud/background class as a whole.

From the above discussion, PNN is found to be the better choice in terms of both the classification accuracy and visual inspection. The modified version of the PNN slightly outperformed the original one with much less computational and storage costs. SOM's do not provide satisfactory classification accuracies.

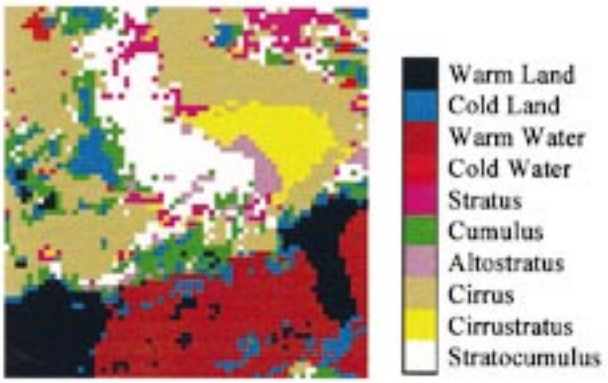


Fig. 8. Classification results using SOM2 and SVD features.

D. Comparison Study of Different Features

Besides the SVD features discussed above, various features including GLCM, WP and spectral features were investigated. The classifier used for this comparison study was the modified PNN. The classification accuracy for the WP and GLCM are provided in Tables VI and VII, respectively. Comparing with the results of SVD in Table III, it was found that WP and GLCM-based statistical features achieved similar results, both fall slightly behind SVD results as far as the overall accuracy is concerned. If the classification rate of each class is analyzed separately, the SVD features performed the best for the altostratus, cirrostratus, stratocumulus and the background, while WP is the best at discriminating cirrus and GLCM features achieved the highest rate for the stratus and cumulus types. No feature set performed consistently the best on all the categories. Moreover, the altostratus and cold water are the most difficult classes for all the three schemes.

The use of spectral feature alone was also examined on the effectiveness of the cloud classification system. The classification accuracies are given in Table VIII. The overall accuracy is the worst among all the examined features. Since all the other features contain both the textural and spectral information, this

TABLE VI
CONFUSION MATRIX FOR WP FEATURES USING MODIFIED PNN CLASSIFIER (%). (ROW—NEURAL-NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 81.2%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	92.7	0.7	0	0	0	2.9	0	2.9	0	0.7
CLnd	4.9	82	0	0	0	3	0	10.1	0	0
WWtr	12.6	1.9	83.3	0	0	0.4	0	1.9	0	0
CWtr	4.3	2.2	0	56.5	0	2.2	2.2	23.9	4.3	4.3
St	0	0	0	0	75.6	2	15.6	0.5	0	6.3
Cu	1.7	0	0	0	0.2	80	4.6	1.9	0	11.4
As	0	0	0	0	4.6	2.5	64.1	6	1.8	21.1
Ci	0.8	0.6	0	0	0	0.8	2.3	85.8	9	0.7
Cs	0	0	0	0	0.1	0	1.6	20	77.9	0.3
Sc	0.2	0	0	0	2.7	6.6	16	1.9	0.3	72.3

TABLE VII
CONFUSION MATRIX FOR GLCM FEATURES USING MODIFIED PNN CLASSIFIER (%). (ROW—NEURAL-NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 80.5%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	97.1	0.5	0.5	0	0	1.2	0	0.7	0	0
CLnd	1.9	92.1	0	0	0	2.2	0	3.7	0	0
WWtr	18.9	0.7	77.8	0	1.1	1.5	0	0	0	0
CWtr	0	2.2	17.4	39.1	4.3	2.2	2.2	30.4	2.2	0
St	0	0	0	0	82.4	0.5	6.3	1	0	9.8
Cu	0.7	0.2	0.2	0	1.2	88.6	1.9	1.7	0.2	5.1
As	0	0	0	0	11.6	2.5	40.1	4.2	3.5	38
Ci	1.3	0.1	0.4	0	1.1	1	1.6	64.7	9	0.7
Cs	0	0	0	0	0.4	0.1	1.3	19.1	77.6	1.3
Sc	0.2	0	0	0	4.2	7.1	15.9	1.5	1.4	69.8

TABLE VIII
CONFUSION MATRIX FOR THE SPECTRAL FEATURE ALONE USING MODIFIED PNN CLASSIFIER (%). (ROW—NEURAL-NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 75.5%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	96.4	0.5	0.2	0	0	2.9	0	0	0	0
CLnd	0.4	94	0	0	0	5.6	0	0	0	0
WWtr	0	0.7	98.9	0	0	0.4	0	0	0	0
CWtr	0	4.3	0	89.1	0	0	0	6.5	0	0
St	0	4.4	0	0	83.4	3.4	7.8	0	0	1
Cu	6.1	10.9	1.5	0	12.9	56.7	1.9	2.9	0	7.1
As	0	0.4	0	0	18.3	5.3	52.5	0.7	0.4	22.5
Ci	1	2.6	2	1.4	2.3	3.8	1.1	71.9	13.4	0.4
Cs	0	0	0	0	1	0.1	1.3	15.4	82.1	0
Sc	1	0.2	0	0	5.6	6.9	18.1	0.5	0	67.7

observation may indicate the contribution of textural features in cloud classification.

Overall, the SVD provides better class discrimination among the examined features followed by the WP, GLCM-based statistical features and lastly the spectral features alone. Also considering the storage and computational requirements, the SVD approach was found to be the most preferred choice for this study.

E. Results of Postprocessing Approach

To account for the spatial contextual information in the satellite imagery, the postprocessing scheme in Section III-B was examined. The satellite images were initially classified using the modified PNN based on the SVD features. The postprocessing scheme was then applied to the output of the PNN classifier. Table IX gives the resultant cloud classification accuracies. The parameter β in (14) was chosen to be 0.35 in this process. Comparing to the results without postprocessing given in Table III, an overall improvement of 5% in classification accuracy was achieved. Furthermore, the improvement was found to be somewhat consistent for all individual classes, ranging from 0.6% for warm land class to nearly 7% for stratocumulus and cumulus classes. Fig. 9 shows the postprocessed color-coded image corresponding to the original color-coded image shown in Fig. 7. As can be observed, after postprocessing a large number of the isolated blocks are removed leading to more homogeneous regions. Moreover, the boundaries between different classes are now more distinct. The results conform better to the experts labeling results in Fig. 6. The postprocessing effects in relation to the choice of parameter β was also investigated. Table X presents the overall classification rate versus β. Notice that $\beta = 0$ implies that no postprocessing was applied. It was found that larger β will lead to higher classification accuracy. However, this does not necessarily mean larger β is better. As we discussed before, larger β indicates stronger class correlation in the neighborhood which favors more

TABLE IX
CONFUSION MATRIX AFTER POST PROCESSING (%) ($\beta = 0.35$). (ROW—NEURAL-NETWORK RESULTS, COLUMN—EXPERTS LABELLING. OVERALL CLASSIFICATION RATE IS 86.3%)

	WLnd	CLnd	WWtr	CWtr	St	Cu	As	Ci	Cs	Sc
WLnd	98.1	0.5	0	0	0	1	0	0.5	0	0
CLnd	0	95.9	0	0	0	2.2	0	1.1	0	0.7
WWtr	0.7	0.4	92.6	0	0.7	2.6	0	3	0	0
CWtr	0	0	0	87	0	2.2	0	10.9	0	0
St	0	0	0	0	81	3.4	11.7	1	0	2.9
Cu	0	0	0	0	0.5	91.2	0.5	2.4	0.5	4.9
As	0	0.4	0	0	2.5	2.8	72.5	3.9	0.7	17.3
Ci	0.5	0.3	0.3	0.1	0.1	0.9	0.8	91.9	4.8	0.3
Cs	0	0	0	0.1	0	0.1	0.7	15	83.7	0.3
Sc	0.2	0	0	0	0.8	7.8	8.3	1.2	0.5	81.3

TABLE X
THE OVERALL CLASSIFICATION ACCURACY AFTER POST PROCESSING (%). THE SVD FEATURES AND SIMPLIFIED PNN WERE USED. $\beta = 0$ IMPLIES NO POSTPROCESSING

β	0	0.05	0.15	0.25	0.35	0.5	0.75	1	1.5
Accuracy	83.8	85.2	87	88	88.8	89.6	90.9	91.3	91.7

Fig. 9. Classification results after postprocessing of Fig. 7. ($\beta = 0.35$).

homogenous regions as the isolated blocks in the color-coded image will be removed. Since most of the classification errors occurred separately, such postprocessing can help to increase the accuracy. However, if β is too large, the image will be over-smoothed and the clouds in small scale regions will not be identified. As a result, the choice of this parameter presents a tradeoff between accuracy and smoothing effects. In our testing, the visual examination of all the postprocessed images led to the optimal choice of $\beta = 0.35$.

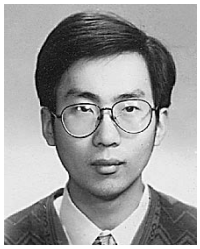
V. CONCLUSIONS

In this paper, a study is conducted on the neural-network-based solutions for the problem of cloud classification from the multispectral GOES-8 satellite imagery. A comprehensive study of unsupervised Kohonen SOM and PNN was made to select the optimal classifier for the problem in hand. These neural networks were benchmarked on their discriminating efficiency on the data labeled by the expert meteorologists. It was demonstrated that PNN achieved the highest classification rate and the best final color-coded image. On the other hand, the Kohonen SOM provided an unsupervised solution with smooth mutually exclusive class boundaries. However, due to the fact that clusters formed by the unsupervised learning may not necessarily correspond (match) to the real-world classes defined by meteorologists, the classification accuracy rates may not be meaningful. Overall, the PNN was found to provide the best solution for the cloud classification. A postprocessing approach was also developed which utilizes the spatial contextual information in the satellite imagery in order to improve the classification accuracy. Additionally, four different sets of features namely SVD, WP, GLCM and spectral were studied and their performance was compared. It was found that the spectral features alone can discriminate different clouds with reasonable accuracy while adding textural information can help to further improve the classification performance. SVD and WP achieved almost similar results while GLCM falls slightly behind. Nevertheless, none of these feature sets provide consistently good discrimination ability for all cloud/background classes. Future work should include searching for a new powerful and robust feature extraction scheme and a self-adaptation scheme for the PNN to accommodate significant temporal, geographical and seasonal changes in satellite imagery sequences. A temporal updating scheme has recently been introduced in [22].

REFERENCES

[1] G. S. Pankiewicz, "Pattern recognition techniques for identification of cloud and cloud systems," *Meteorol. Appl.*, vol. 2, pp. 257–271, Sept. 1995.
[2] W. E. Shenk and R. J. Holub, "A multispectral cloud type identification method developed for nimbus 3 mrir measurements," *Mon. Weather Rev.*, vol. 104, pp. 284–291, Mar. 1976.
[3] M. Desbois and G. Szejwach, "Automatic classification of clouds on meteosat imagery: Application to high-level clouds," *J. Clim. Appl. Meteorol.*, vol. 21, pp. 401–412, 1982.
[4] D. W. Reynolds and T. H. Vonder-Haar, "Bispectral method for cloud parameter determination," *Mon. Weather Rev.*, vol. 105, pp. 446–457, Mar. 1977.
[5] J. Parikh and A. Rosenfeld, "Automated segmentation and classification of infrared meteorological satellite data," *IEEE Trans. Syst. Man. Cybern*, vol. 8, pp. 736–743, 1978.
[6] N. Lamei *et al.*, "Cloud-type discrimination via multispectral textural analysis," *Opt. Eng.*, vol. 33, pp. 1303–1313, Apr. 1994.
[7] R. M. Haralick *et al.*, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 610–621, Mar. 1973.
[8] R. M. Welch, K. S Kuo, S. K. Sengupta, and D. W. Chen, "Cloud field classification based upon high spatial resolution textural feature

(I): gray-level cooccurrence matrix approach," *J. Geophys. Res.*, vol. 93, pp. 12, 663–12681, Oct. 1988.
[9] K. S. Kuo, R. M. Welch, and S. K. Sengupta, "Structural and textural characteristics of cirrus clouds observed using high spatial resolution Landsat imagery," *J. Appl. Meteorol.*, vol. 27, pp. 1242–1260, Aug. 1988.
[10] L. Garand, "Automatic recognition of oceanic cloud patterns. Part I: Methodology and application to cloud climatology," *J. Climate*, Jan. 1988, pp. 20–39.
[11] Z. Gu and C. Duncan, "Texture and spectral features as an aid to cloud classification," *J. Remote Sensing*, vol. 12, no. 5, pp. 953–968, 1991.
[12] L. J. Du, "Texture segmentation of SAR images using localized spatial filtering," *Proc. Int. Geosci. Remote Sensing Symp.*, Washington, D.C., 1990, pp. 1983–1986.
[13] J. A. Parikh, "A comparative study of cloud classification techniques," *Remote Sensing Environment*, vol. 6, pp. 67–81, Mar. 1977.
[14] P. P. Ohanian and R. C. Dubes, "'Performance evaluation of four classes of texture features," *Pattern Recognition*, vol. 25, pp. 819–833, 1992.
[15] M. F. Aug.eijin, "Performance evaluation of texture measures for ground cover identification in satellite images by means of a neural-network classifier," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 616–625, May 1995.
[16] J. Lee *et al.*, "A neural-network approach to cloud classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 28, no. 5, pp. 846–855, Sept. 1990.
[17] R. L. Bankert, "Cloud classification of AVHRR Imagery in maritime regions using a probabilistic neural network," *J. Appl. Meteorol.*, vol. 33, pp. 909–918, Aug. 1994.
[18] R. M. Welch *et al.*, "Polar cloud and surface classification using AVHRR imagery: An intercomparison of methods," *J. Appl. Meteorol.*, vol. 31, pp. 405–420, May 1992.
[19] A. Visa, J. Iivarinen, K. Valkealahti, and O. Simula, "Neural-network-based cloud classifier," in *Proc. Int. Conf. Artificial Neural Networks (ICANN'95)*, Ind. Session 14 (Remote Sensing), 1995.
[20] R. J. Liou and M. R. Azimi-Sadjadi, "Detection and classification of cloud data from geostationary satellite using artificial neural networks," in *Proc. IEEE Int. Conf. Neural Networks (ICNN'94)*, Orlando, FL, 1994, pp. 4327–4332.
[21] P. Blonda, G. Pasquariello, and J. Smith, "Comparison of backpropagation, cascade-correlation and Kohonen algorithms for cloud retrieval," in *Proc. IJCNN-93, Int. Joint Conf. Neural Networks*," vol. II, 1993, pp. 1231–1234.
[22] B. Tian, M. R. Azimi-Sadjadi, T. H. Vonder Haar, and D. Reinke, "Temporal updating scheme for probability neural network with application to satellite cloud classification," submitted to *IEEE Trans. Neural Networks*.
[23] M. A. Shaikh and B. Tian *et al.*, "Neural-network-based cloud detection/classification using textural and spectral features," in *Proc. IGRASS'96*, Lincoln, NB, June 1996, pp. 1105–1107.
[24] J. H. Conover, "Cloud interpretation from satellite altitudes," Air Force Cambridge Res. Lab., Cambridge, MA, Tech. Rep. AFCRL-62680, 1962.
[25] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[26] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA: SIAM, 1992.
[27] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.
[28] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[29] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Cambridge, U.K.: Wellesley-Cambridge Press, 1996.
[30] T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Processing*, vol. 2, pp. 429–441, Oct. 1993. [29] J. D. Villasenor, [31] A. Laine and J. Fan, "Texture classification by wavelet packet signatures," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 1186–1191, Nov. 1993.
[31] T. Kohonen, "The self-organizing map," *Proc. IEEE: Special Issue on Neural Networks—I: (Theory and Modeling)*, vol. 78, pp. 1464–1480, Mar. 1990.
[32] ——, *Self-Organization and Associative Memory*. New York: Springer-Verlag, 1984.
[33] D. F. Specht, "Probabilistic neural network," *Neural Networks*, vol. 3, pp. 109–118, 1990.
[34] D. F. Specht and P. D. Shapiro, "Generalization accuracy of probabilistic neural networks compared with backpropogation networks," in *Proc. IJCNN'91*, pp. 458–461, July 1991.
[35] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc.*, series B, vol. 39, pp. 1–38, 1977.
[36] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, pp. 1065–1076, 1962.
[37] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.
[38] P. Burrascano, "Learning vector quantization for the probabilistic neural network," *IEEE Trans. Neural Networks*, vol. 2, pp. 458–461, July 1991.
[39] R. L. Streit and T. E. Luginbuhl, "Maximum Likelihood Training of Probabilistic Neural Networks," *IEEE Trans. Neural Networks*, vol. 5, pp. 764–783, 1994.
[40] S. Lin, S. Y. Kung, and L. Lin, "Face recognition/detection by probabilistic decision-based neural network," *IEEE Trans. Neural Networks*, vol. 8, pp. 764–783, 1997.
[41] B. Jeon and D. A. Landgrebe, "Classification with spatio-temporal interpixel class dependency contexts," *IEEE Trans. Geosci. Remote Sensing*, vol. 30, pp. 663–672, July 1992.
[42] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution and Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, Nov. 1984.
[43] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 153–158, Feb. 1997.
[44] S. W. Miller and W. J. Emery, "An automatic neural-network cloud classifier for use over land and ocean surface," *J. Appl. Meteorol.*, vol. 36, pp. 1346–1362, Oct. 1997.
[45] R. Hecht-Nielsen, "Kolmogrov's mapping neural network existence theorem," in *1st IEEE Conf. Neural Networks*, San Diego, CA, vol. 3, pp. 11–14, 1987.



Bin Tian received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1991 and 1993, respectively. He has recently completed the Ph.D. degree at the Department of Electrical Engineering, Colorado State University, Fort Collins.

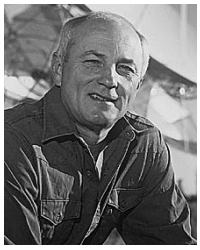
His main research interests include signal/image processing, telecommunication, and theory and applications of neural networks.

Mukhtiar A. Shaikh, photograph and biography not available at the time of publication.

Mahmood R. Azimi-Sadjadi (S'81–M'81–SM'89) received the B.S. degree from University of Tehran, Iran,, in 1977, the M.Sc. and Ph.D. degrees from the Imperial College, University of London, U.K., in 1978 and 1982, respectively, all in electrical engineering.

He served as an Assistant Professor in the Department of Electrical and Computer Engineering, University of Michigan-Dearborn. Since July 1986 he has been with the Department of Electrical Engineering, Colorado State University, Fort Collins, where he is now a Professor. He is also the director of the Multisensory Computing Laboratory (MUSCL) at Colorado State University. His areas of interest include digital signal/image processing, target detection and tracking, multidimensional system theory and analysis, adaptive filtering, system identification, and neural networks. His contributions in these areas resulted in more than 100 journal and refereed conference publications. He is a coauthor of the book *Digital Filtering in One and Two Dimensions* (New York: Plenum, 1989).

Dr. Azimi-Sadjadi is the recipient of 1993 ASEE-Navy Senior Faculty Fellowship Award, 1991 CSU Dean's Council Award, 1990 Battelle Summer Faculty Fellowship Award, and the 1984 DOW chemical Outstanding Young Faculty Award of the American Society for Engineering Education. He is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING.



Thomas H. Vonder Haar received the B.S. degree in aeronautics from Parks College of St. Louis University, MO, in 1963, the M.S. and Ph.D. degrees in meteorology from the University of Wisconsin, Madison, in 1964 and 1968, respectively.

He has taught in the Atmospheric Science Department at Colorado State University, Fort Collins, since 1970, and became a University Distinguished Professor in 1994. He became the Director of the Cooperative Institute for Research in the Atmosphere (CIRA) at Colorado State University in 1980. He has been the senior author or coauthor on more than 400 scientific publications, more than 100 in refereed journals. The range of titles covers several different research areas, with the use of measurements from meteorological satellites as a most common factor. Research work has been published in such journals as *Science, Solar Energy, Space Research, Journal of Atmospheric Science, Bulletin of the American Meteorological Society, Monthly Weather Review, Journal of Applied Meteorology, Journal of Geophysical Research, Journal of Atmospheric and Oceanic Technology*, etc. He is coauthor of *Satellite Meteorology: An Introduction* (New York: Academic, 1995). In addition, some original research work is now being included in textbooks and summaries of atmospheric scinece and atmospheric physics by authors both in the United States and abroad.

Donald L. Reinke received the B.S. degree in meteorology from the University of Oklahoma, Norman, in 1975 and the M.S. degree in atmospheric science from Colorado State University, Fort Collins, in 1982.

He has extensive experience in automated processing of meteorological satellite data, satellite derived cloud analyzes, scientific programming, and real-time meteorological support systems management. He is presently a Research Associate at Colorado State University managing a research group that provides data, data analysis computer applications, and technical support to CIRA, as well as management of the computer support resources for the 50-plus person research facility. His group also has the responsibility for 24-h/day operation of real-time satellite earthstation to collect environmental data from eight worldwide meteorological satellite platforms and two data service networks.His professional experience includes duty as a Research Scientist and Project Manager for a number of multiyear research programs related to the processing and analysis of digital meteorological satellite imagery and related ancillary data sets. He has been a lead or coauthor on more than 60 published papers.