A Temporally Adaptive Classifier for Multispectral Imagery

Jianqi Wang, Mahmood R. Azimi-Sadjadi, and Donald Reinke

***Abstract*—This paper presents a new temporally adaptive classification system for multispectral images. A spatial–temporal adaptation mechanism is devised to account for the changes in the feature space as a result of environmental variations. Classification based upon spatial features is performed using Bayesian framework or probabilistic neural networks (PNNs) while the temporal updating takes place using a spatial–temporal predictor. A simple iterative updating mechanism is also introduced for adjusting the parameters of these systems. The proposed methodology is used to develop a pixel-based cloud classification system. Experimental results on cloud classification from satellite imagery are provided to show the usefulness of this system.**

***Index Terms*—Bayes classification, cloud classification, multispectral imaging, prediction.**

I. Introduction

MULTISPECTRAL meteorological satellite imaging systems generally provide frequent, high-resolution (0.5–5 km), visible, and infrared (IR) images over large areas. They play an important role in remote sensing, weather analysis and forecasting, and military applications [1]–[7]. However, due to the large volume of data received every day, manual inspection becomes impractical. Thus, automated and efficient detection and classification systems are needed.

The major challenge in designing classification systems for multispectral imaging, besides the large volume and frequency of the data, is that as time elapses the temperature variations can cause substantial changes in the IR channel features. Additionally, owing to sun angle changes over time, the reflectivity in the visible channel will also change, leading to the variations in the visible channel features. Furthermore, the overlap among features of different areas of interests (e.g., land, clouds, water) can drastically deteriorate the classification performance of the system over time. To address these issues an adaptive classification system that can adjust its parameters to account for the temporal and environmental changes in multispectral images is needed. In [8], Tian et. al. developed a new temporal updating algorithm to address this issue for cloud classification from Geostationary Operational Environment Satellite (GOES)-8 data. A PNN was used as the classifier. To perform temporal updating, a context-based predictor is introduced to classify the current image frame based on the classification results in the previous frame. The PNN classifier, updated to the previous frame, is also used to classify the current frame. A supervised learning algorithm is used to fine-tune the PNN classifier based on those blocks for which the labels of the predictor and classifier match. An unsupervised learning algorithm is used for the rest of the blocks. This method achieved very promising results for coping with feature changes in the IR and visible channels of GOES-8 data.

This paper presents a generalization of the temporal adaptation method in [8] by providing a theoretical framework for spatial–temporal updating idea that encompasses decoupled classification and prediction methodologies using Bayesian classification and Markov-based prediction. Multispectral imaging application is selected here though the methodology can be applied to any problem that involves spatial–temporal feature changes. Based upon this framework a simple pixel-based cloud classification system is developed. Experimental results are presented that show the usefulness of the proposed algorithm for classifying GOES-8 satellite imagery data into five specific classes, namely high-level cloud, middle-level cloud, low-level cloud, land, and water.

Manuscript received July 9, 2002; revised January 23, 2003. This work was supported by the DoD Center for Geoscience Atmospheric Research at Colorado State University under the Cooperative Agreement (#DAAL01-98-2-0078) with the Army Research Laboratory.

J. Wang and M. R. Azimi-Sadjadi are with the Department of Electrical and Computer Engineering. Colorado State University, Fort Collins, CO 80523 USA (e-mail: azimi@engr.colostate.edu)

D. Reinke is with the Cooperative Institute for Research in the Atmosphere (CIRA), Colorado State University, Fort Collins, CO 80523 USA.

Digital Object Identifier 10.1109/TNN.2003.820622

II. Spatial–Temporal Adaptive Classification

As explained earlier, the temperature and reflectivity changes make the feature space highly variable over time. Nevertheless, the classifier trained on the previous image frame, can still classify the current frame relatively well as the environmental changes between two consecutive frames (20 minutes apart) are relatively small. Additionally, a large number of pixels keep their class types between two consecutive frames owing to the fact that most clouds do not evolve and/or move very rapidly. These properties are exploited to develop a new temporally adaptive classification system.

A. Decoupled Multispectral Classification Problem

Let $\mathbf{x}(n) = (\mathbf{v}(n), \mathbf{g}(n))$ be the spatial–temporal "state" for a block (or pixel) at location $\mathbf{v}(n)$ at frame (or time) n with multispectral feature set $\mathbf{g}(n) = \{\mathbf{g}_1(n), \mathbf{g}_2(n) \ldots \mathbf{g}_M(n)\}$, where $\mathbf{g}_i(n)$, $i \in [1, M]$ is the feature vector for spectral band i. It is very important to note that different ways of combining (or fusing) the feature vectors of the spectral bands yield different classification/fusion methodologies. For instance, if we had defined $\mathbf{g}(n) = [\mathbf{g}^t{}_1(n), \mathbf{g}^t{}_2(n), \ldots \mathbf{g}^t{}_M(n)]^t$, i.e., all the spectral band feature vectors are lumped into one multispectral feature

vector, this would have required only one classifier for the augmented feature vector; whereas the former arrangement necessitates using a classifier for every spectral band feature vector $\mathbf{g}_i(n)$. Now, let $P(C_k|\mathbf{x}(n))$ denote the *a posteriori* class conditional probability, then the class label $C(\mathbf{x}(n))$ of $\mathbf{x}(n)$ among N possible classes is decided based upon the maximum *a posteriori* (MAP) method as

$$C(\mathbf{x}(n)) = \arg \max_{1 \le k \le N} P(C_k|\mathbf{x}(n)). \quad (1)$$

Using the Bayes rule and the fact that $P(\mathbf{g}(n)|\mathbf{v}(n))$ is not class-dependent, (1) can be written as

$$C(\mathbf{x}(n)) = \arg \max_{1 \le k \le N} P(C_k|\mathbf{v}(n))P(\mathbf{g}(n)|C_k, \mathbf{v}(n)). \quad (2)$$

It is reasonable to assume that $P(\mathbf{g}(n)|C_k, \mathbf{v}(n)) = P(\mathbf{g}(n)|C_k)$ i.e., the multispectral feature vector $\mathbf{g}(n)$ given a class C_k is independent of the location of a specific block/pixel. Then (2) becomes

$$C(\mathbf{x}(n)) = \arg \max_{1 \le k \le N} P(C_k|\mathbf{v}(n))P(C_k|\mathbf{g}(n)) \quad (3)$$

where it is assumed that the *a priori* probabilities are equal i.e., $P(C_k) = P(C_j)$, $1 \le k, j \le N$, $k \ne j$, and further $P(\mathbf{g}(n))$ is the same for all classes.

The spatial–temporal MAP condition in (3) requires computing two separate the *a posteriori* conditional probabilities, namely $P(C_k|\mathbf{v}(n))$ that is not dependent on the spectral features, and $P(C_k|\mathbf{g}(n))$, which is location independent. The latter probability can be generated using any neural network that implements Bayes classification e.g., PNN [8]. On the other hand, estimating the former probability calls for a spatial–temporal prediction that is considered next.

B. Prediction

For two images with high spatial–temporal correlations, the task of the predictor is to predict the classification result in one image based on the results in the previous images. Clearly, there is rich temporal contextual correlation in two consecutive frames of multispectral images since the short-term changes, e.g., movement, formation and dissipation of clouds, is typically not significant. This property could be exploited to build a predictor that utilizes the spatial–temporal correlation between two consecutive frames to estimate $P(C_k|\mathbf{v}(n))$. The temporal contextual correlation between two highly correlated images can be modeled by a Markov chain. For the sake of computational simplicity, first-order Markov chain is considered here. The Markov assumption of the predictor implies that the label of block/pixel $\mathbf{v}(n)$ is solely determined by the labels of those pixels in its spatial–temporal neighborhood in frame $n-1$.

For a block/pixel at location $\mathbf{v}(n)$, let $\mathcal{R}_{n-1}(\mathbf{v}(n))$ be its spatial–temporal neighborhood in frame $n-1$ with support region geometry Γ with constituent members Γ_i, i.e., $\Gamma = \bigcup_i \Gamma_i$ where $\bigcup$ stands for union operation. Now, let $C_{\mathbf{v}(n)}$ be the class label of block/pixel $\mathbf{v}(n)$ and define $C_{\mathcal{R}_{n-1}}$ as the set of the class labels of all the blocks/pixels in $\mathcal{R}_{n-1}(\mathbf{v}(n))$, i.e., $C_{\mathcal{R}_{n-1}} = \{C_{\mathbf{v}_i(n-1)}|\mathbf{v}_i \in \mathcal{R}_{n-1}(\mathbf{v}(n))\}$. Thus, the problem of finding $P(C_k|\mathbf{v}(n))$ is equivalent to that of computing $P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma)$ that can be generated using the predictor. The derivation is given here.

Using $\Gamma = \bigcup_i \Gamma_i$ and the fact that Γ_is are mutually exclusive, the conditional probability $P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma)$ can be written as

$$\begin{aligned} P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma) &= \sum_{i=1}^{L} \frac{P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma_i)P(C_{\mathcal{R}_{n-1}}, \Gamma_i)}{P(C_{\mathcal{R}_{n-1}}, \Gamma)} \\ &= \sum_{i=1}^{L} P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma_i)P(\Gamma_i|\Gamma) \\ &= \sum_{i=1}^{L} P(C_{\mathbf{v}(n)}|C_{\mathbf{v}_i(n-1)}, \Gamma_i)P(\Gamma_i|\Gamma) \end{aligned}$$

where $L = |\Gamma|$ is the size of the neighborhood (i.e., the number of bixels/blocks), and $P(C_{\mathbf{v}(n)}|C_{\mathbf{v}_i(n-1)}, \Gamma_i)$ denotes the class transition probability from pixel $\mathbf{v}_i(n-1) \in \mathcal{R}_{n-1}(\mathbf{v}(n))$ to the pixel $\mathbf{v}(n)$ and $P(\Gamma_i|\Gamma)$ can be interpreted as the contribution of pixel $\mathbf{v}_i(n-1)$ to the determination of the class label of $\mathbf{v}(n)$. If N denotes the number of classes, the former probability can be written as

$$P(C_{\mathbf{v}(n)} = k|C_{\mathbf{v}_i(n-1)} = l, \Gamma_i) = \begin{cases} \alpha_i & k = l \\ \frac{1-\alpha_i}{N-1} & k \ne l \end{cases} \quad (4)$$

which implies that pixel $\mathbf{v}_i(n-1)$ imposes its class label on $\mathbf{v}(n)$ with probability α_i while other classes have probability $(1-\alpha_i)/N-1$. The other factor that affects the probability $P(C_{\mathbf{v}(n)}|C_{\mathcal{R}_{n-1}}, \Gamma)$ is the relation of Γ_i with Γ or $P(\Gamma_i|\Gamma)$, which can be given by

$$P(\Gamma_i|\Gamma) = K \cdot \beta^{H(\mathbf{v}_i, \mathbf{v})} \quad (5)$$

where $0 < \beta \le 1$, K is a constant that should satisfy $\sum_{1 \le i \le M} K \cdot \beta^{H(\mathbf{v}_i, \mathbf{v})} = 1$, and $H(\mathbf{v}_i, \mathbf{v})$ is the Hamming distance between $\mathbf{v}_i$ and $\mathbf{v}$. Hence, the farther the pixel $\mathbf{v}_i$ is from pixel $\mathbf{v}$, the less effect it has on its label. Thus, the predictor decides the class label of pixel $\mathbf{v}(n)$ using

$$C_{\mathbf{v}(n)} = \arg \max_{1 \le k \le N} P(C_{\mathbf{v}(n)} = k|C_{\mathcal{R}_{n-1}}, \Gamma). \quad (6)$$

In practice, however, a threshold W is chosen and those pixels satisfying $\max_{1 \le k \le N} P(C_{\mathbf{v}(n)} = k|C_{\mathcal{R}_{n-1}}, \Gamma) < W$ are not labeled ('no label' class). Note that if we choose W to be the upper limit of $P(C_{\mathbf{v}(n)} = k|C_{\mathcal{R}_{n-1}}, \Gamma)$ and let $\beta = 1$, then this predictor becomes one for which the following rule applies: if and only if all the pixels in the spatial–temporal neighborhood $\mathcal{R}_{n-1}(\mathbf{v}(n))$ belong to class k, then $C_{\mathbf{v}(n)} = k$. This property is similar to the erosion in morphological operations with the neighborhood being the structuring element. This will be used in Section III.

C. Iterative Updating

Having defined the classification and prediction processes, a simple iterative temporal updating mechanism is given to update the parameters of the classifier(s). The temporal updating method involves forming two subsets T_1 and T_2 iteratively using

the results of the classifier and predictor, respectively. These subsets are defined as

$$T_1 = \left\{ (\mathbf{x}(n), C) | C = \arg \max_{1 \le k \le N} P(C_k | \mathbf{v}(n)) \right\}$$
$$T_2 = \left\{ (\mathbf{x}(n), C) | C = \arg \max_{1 \le k \le N} P(C_k | \mathbf{g}(n)) \right\}$$

where C represents the class label of $\mathbf{x}(n)$. Additionally, let us define set

$$T = \left\{ (\mathbf{x}(n), C) | C = \arg \max_{1 \le k \le N} P(C_k | \mathbf{v}(n)) P(C_k | \mathbf{g}(n)) \right\}$$

which includes those blocks/pixels whose labels have high confidence. Clearly, the set $T_1 \bigcap T_2 \subseteq T$ can be used to retrain the whole system at every new frame. To accomplish this, an iterative updating process is devised here. The process starts with initial subsets T_1^0 and T_2^0 where T_1^0 is obtained by applying the erosion operation performed by the predictor using the classification results of the previous frame, and T_2^0 is generated by applying the classifier built from the last frame to the current frame. To determine T_2^0, we use the fact that between two consecutive frames, the variations of $P(\mathbf{g}(n)|C_i)$ are not significant and thus the estimated distribution for the last frame can give a good initial value for the current frame. Then, for any frame the process at iteration i $(i \ge 1)$ involves:

1) Use samples in $T_1^{i-1} \bigcap T_2^{i-1}$ to retrain all the classifiers. Note that sufficient number of correctly classified samples for each class are needed to reestimate the parameters of each classifier.
2) Use the updated classifiers to classify the current frame and form T_2^i.
3) Use the predictor to get T_1^i from $T_1^{i-1} \bigcap T_2^{i-1}$.
4) Iterate several times until the changes of parameters of the system do not exceed a preselected limit or until no further improvement is gained.

After the updating is completed, we use the updated system to classify the multispectral image and get the final results for the current image frames.

III. A Hierarchical Pixel-Based Cloud Classification System

The applicability of the proposed spatial–temporal updating method is tested on a cloud classification problem from GOES8 satellite images. Normally, in cloud classification, two-channel data, namely visible and IR channels are used. The visible channel provides reflectivity as well as textural information in different cloud and noncloud areas. IR channel, on the other hand, gives temperature and some texture information where the temperature is directly related to the cloud height information. In the visible channel, Land and Water will generally appear darker than clouds. In the IR channel, different cloud heights are correlated with temperature which is directly related to the digital count value.

The proposed method is applied to develop a multistage pixel-based temporally adaptive system for cloud classification. This multistage system is shown in Fig. 1. The visible and IR channel images are first applied to the "cloud detector" to differentiate between cloud and no-cloud pixels. The IR data is applied to this system only to detect the presence of thin and/or shadowed clouds that are hard to see in the visible channel. For the no-cloud pixels, a binary geographical map of land and water is used to label these pixels. Those pixels classified as clouds are subsequently applied to the IR classifier in order to classify them into low, middle and high-level clouds. In this hierarchical cloud classification system, a texture-based classifier may also be added to classify each of the three types of clouds into their associated subclasses, e.g., middle-level clouds can further be classified into Altostratus or Altocumulus classes. However, in this paper, this option is not implemented.

This pixel-based approach has several advantages over the block-based schemes [8], [9]. These advantages include: much higher resolution in the final images, no boundary effect problems (a large portion of boundary blocks lie on the boundary of the cloud types that have different heights or on the boundary of Cloud and Land, or Cloud and Water), and most importantly the simplicity of the classifiers (scalar-input). The latter property makes the updating process less difficult to implement. The price paid for these advantages is that this pixel-based system is only capable of classifying clouds into three cloud classes depending on their heights. However, as mentioned before, these three cloud types can further be classified into more specific cloud types based upon block-based classification systems that account for textural as well as spectral features.

This system could be made temporally adaptive by adjusting the parameters of the two classifiers in response to the sun angle variations and diurnal temperature changes of the land and water. The predictor uses the spatial–temporal correlations between two adjacent frames to provide the results of the current frame from the classification results of the previous frame. The iterative updating method in Section II can be used to adjust the parameters of the two classifiers. These two classifiers are designed using the Bayesian framework and initially trained using images labeled by meteorologists. The design of Bayesian classifiers involve the specification of appropriate decision thresholds.

A. Cloud Detector

For the first classifier (cloud detector) that differentiates between cloud and no-cloud pixels two decision thresholds need to be specified, one for the visible channel, and the other for the IR channel. In the visible image, Land typically has a greater intensity than Water. Nonetheless, if the Water is very rough, it will reflect more sunlight and can be almost as bright as Land. On the other hand, Clouds are almost always brighter than both (the exception is snow-covered Land or bright white sand or salt flats). Consequently, determining the right threshold between Land and Cloud is sufficient to differentiate between Cloud and Water. Thus, the task is reduced to that of determining the right threshold between Land and Cloud.

The difficulty in building this classifier is that most of the low-level clouds (e.g., Cumulus) appear in the image as isolated spots whereas meteorologists generally label images into areas. As a result, the labeled low-level cloud areas are most likely mixtures of low-level cloud and land pixels, or generally clouds and land as we can combine the three cloud types into one

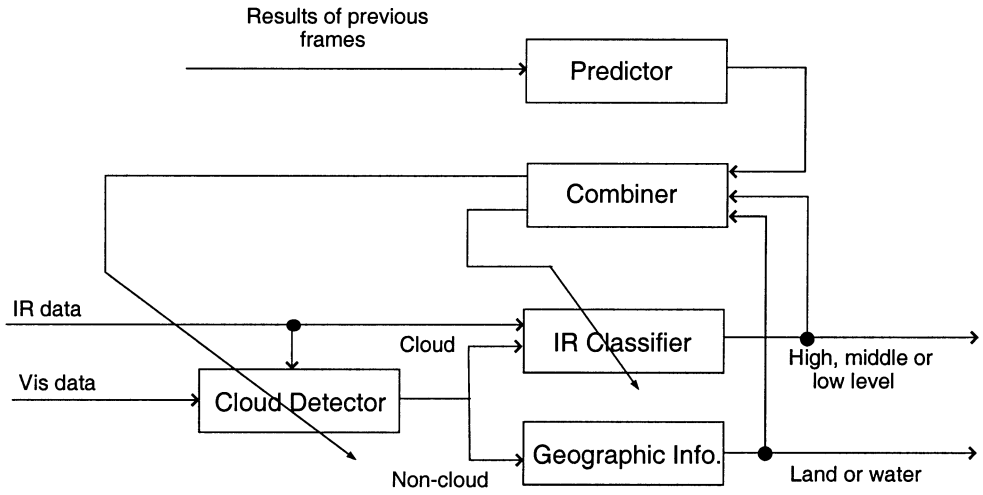


Fig. 1. Block diagram of the pixel-based cloud classification system.

"cloud" class. On the other hand, for those labeled land areas, the confidence in labeling is very high. Thus, two training data sets, one exclusively for "land" class (denoted by S_1) and one composed of mixture of "land" and "cloud" classes (denoted by S_2) can be used to construct a simple Bayesian cloud detector. Our experimental studies of the histograms of these two data sets indicated that the histogram of S_1 can be approximated by a unimodal normal PDF while that of S_2 exhibits a bimodal PDF. Now, if we let the PDFs (normal) of land and cloud pixels be $p_l : \mathcal{N}(\mu_l, \sigma_l^2)$ and $p_c : \mathcal{N}(\mu_c, \sigma_c^2)$, respectively, the PDFs of S_1 and S_2 can be expressed as

$$p_1(g_1|\phi) = p_l(g_1|\phi_l) \tag{7}$$

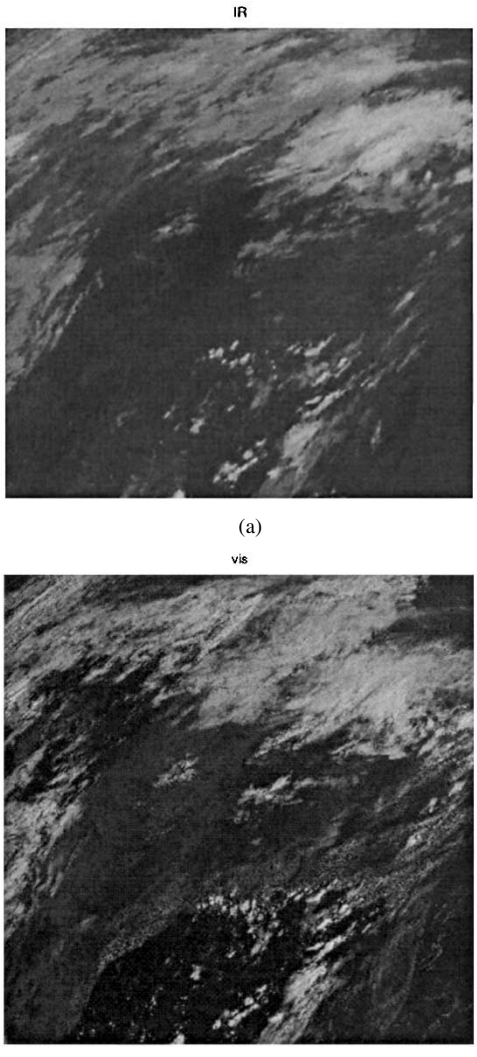
$$p_2(g_1|\Phi) = \alpha_l p_l(g_1|\phi_l) + \alpha_c p_c(g_1|\phi c) \tag{8}$$

where g_1 represents intensity of the pixels in the visible channel, $\phi_i := (\mu_i, \sigma_i^2), i = l, c$ is the parameter set of the ith distribution with μ_i and σ_i^2 being the mean and variance of the Gaussian, and $\Phi = (\alpha_l, \alpha_c, \phi_l, \phi_c)$ is the augmented set of parameters and α_l and α_c are the *a priori* probabilities of land and clouds, respectively. Since S_1 contains much more reliable information about the distribution of the intensity of land, we include it in estimating the PDFs of the two classes and keep it separate from S_2.

The log likelihood function can be written as:

$$L(\Phi) = \sum_{g_{1i} \in S_1} \log(p_1(g_{1i}|\phi_l)) + \sum_{g_{1k} \in S_2} \log(p_2(g_{1k}|\Phi)) \tag{9}$$

By maximizing the log likelihood function, we can get the parameter set Φ. The well-known Expectation-Maximization (EM) approach is an efficient way to solve this problem [10], [11]. The EM equations for this problem are given in the Appendix . Note that if we assume that cloud and land pixels have the same *a priori* probabilities, the threshold can be decided by solving $p_l(g_1|\phi_l) = p_c(g_1|\phi_c)$. As pointed out before, since in daytime, the intensity of the land is not less than that of water, this threshold can also be used to differentiate cloud and water pixels. In other words, this threshold is suitable for differentiating cloud and no-cloud pixels.



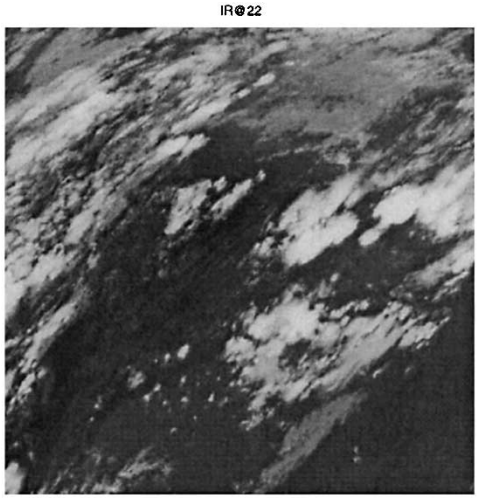
(a)

(b)

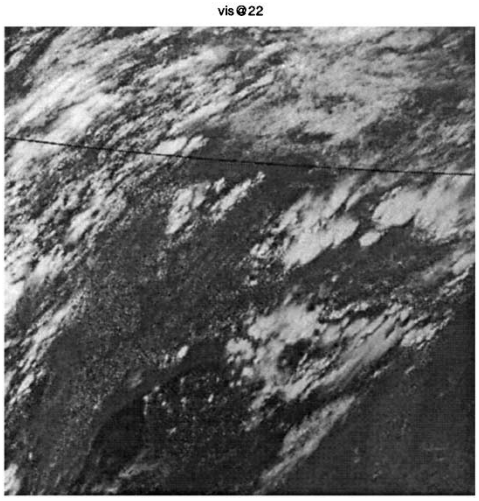
Fig. 2. Original image pair at 15:45 UTC. (a) Infrared. (b) Visible.

B. IR Classifier

The design of the IR-channel classifier is quite straightforward. The PDF of each cloud type, i.e., low-level cloud,

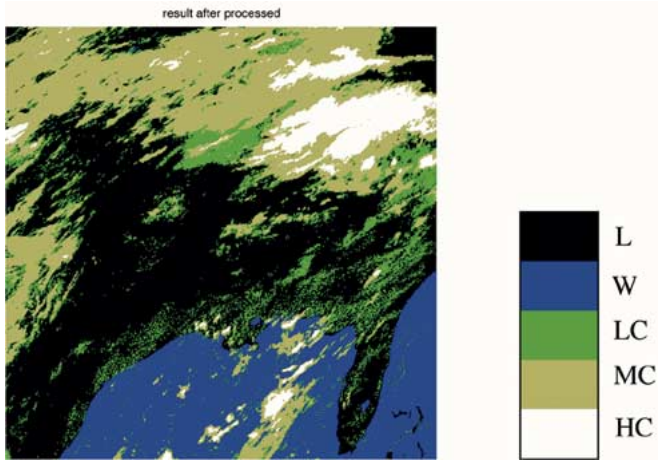


(a)



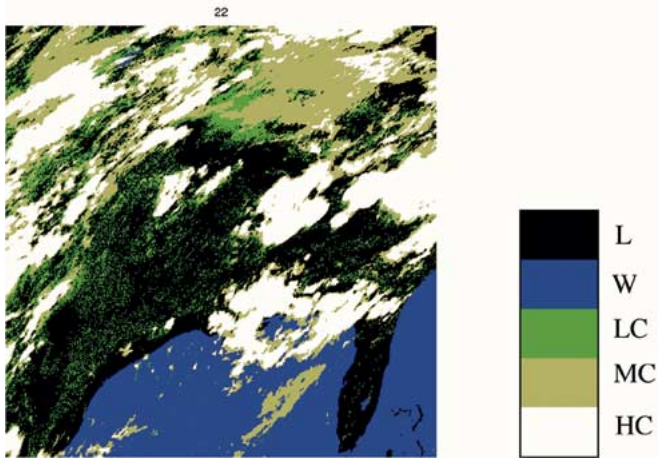
(b)

Fig. 3. Original image pair at 22:45 UTC. (a) Infrared. (b) Visible.



(a) (b)

Fig. 4. Classification result of the pixel-based method (15:45 UTC).



(a) (b)

Fig. 5. Classification results of the pixel-based method (22:45 UTC).

middle-level cloud and high-level cloud, can be approximated by a unimodal normal PDF with parameters that can be directly calculated from the labeled images. The classifier is then designed using the Bayesian framework and letting the *a priori* probability of each type be equal among all the three classes. There are two points that need to be clarified here. First, the training of the IR-channel classifier follows that of the cloud detection system, which determines the threshold between cloud and land pixels. Thus, we can use this threshold to remove the land and water pixels from the labeled low-level cloud areas. Second, since most of the thin cloud and cloud shadow belong to middle-level or high-level clouds, once the IR-channel classifier is trained, the threshold between the low-level cloud and middle-level cloud is fed back to the cloud detection system. If the intensity of a pixel in the IR image is higher than the threshold, we classify it as a cloud pixel, irrespective of its intensity in the visible image.

IV. Experimental Results

To test the pixel-based system on real cloud images, a GOES 8 satellite imagery database was used. GOES 8 satellite carries one visible and four infrared channel sensors. However, only the data from channel 1 (visible) and channel 4 (IR) are used because these two channels are the only two that are currently common among all global imaging systems. The image sequence on July 24th, 1998 during 15:45 to 22:45 UTC (Universal Time Coordinate) at one hour interval was employed in this study. These images of size 512×512 pixels (spatial resolution of 5 km/pixel) cover the Midwest and most of the Eastern part of the U.S. Lake Michigan is in the upper right corner and Florida is located in the lower right, with Gulf of Mexico in the lower center of the image. The images were labeled by two meteorologists and only those areas in which their labeling agreed were used for training and validation purpose. Figs. 2(a) and (b) and 3(a) and (b) show the first (15:45 UTC) and last (22:45 UTC) image pairs in the sequence, respectively.

The system is initially trained based upon the labeled pixels in the first image at 15:45 UTC and then continuously updated using the spatial–temporal updating mechanism. Table I shows the classification accuracies when this approach is applied to this image sequence (15:45 to 22:45 UTC). The performance of the system on the low-level clouds is not available owing

TABLE I
THE CLASSIFICATION ACCURACY IN REAL CLOUD IMAGES

$Time(UTC)$	$Land$	$Water$	$Low-level$ $Cloud$	$Middle-level$ $Cloud$	$High-level$ $Cloud$
15 : 45	0.9912	0.9994	N/A	0.9368	0.9687
16 : 45	0.9840	1.0000	N/A	0.8913	0.9755
17 : 45	0.9743	1.0000	N/A	0.9117	0.9666
18 : 45	0.9833	0.9995	N/A	0.7792	0.9440
19 : 45	0.9840	0.9917	N/A	0.7526	0.9845
20 : 45	0.9937	0.9960	N/A	0.7073	0.9292
21 : 45	0.9963	0.9999	N/A	0.6405	0.9478
22 : 45	0.9858	0.9977	N/A	0.8601	0.9613

to low confidence in the labeling of low-level cloud areas. Additionally, we also notice a drastic decline in the accuracy of the middle-level clouds from 17:45 to 18:45 UTC and in the subsequent frames. This is probably due to the fact that the cloud top temperatures are very close to the temperature used as the cut-off between Middle and High-level cloud classification by the expert who did the manual analysis. This appears to be due the fact that the significant amount of convective cloud growth is occurring which causes the same cloud element to pass from middle to high class. It is interesting to note that this discrepancy did not deteriorate the performance on the high-level clouds. Figs. 4 and 5 show the color-coded classification results of the first hour (at 15:45 UTC) and the last hour (at 22:45 UTC) in this sequence, respectively. The color assignment is shown in the color-bar in Fig. 4(b). The results demonstrate the effectiveness of the proposed spatial–temporal updating scheme. Also notice that in Fig. 3, the black line in the visible channel, is a data drop that was totally removed in the results of Fig. 5(b).

The plot of the decision threshold for separating between cloud and no-cloud pixels versus time in Fig. 6 is very insightful. As can be observed, this threshold is first increasing and reaches its peak around 20:45 UTC and then it starts to go down. This variation in the decision threshold follows the same trend as the diurnal sunlight intensity change. This is due to the fact that the intensities of land (and to some extent cloud) pixels change as the sunlight varies during the day. Hence, the decision threshold used to differentiate between land and clouds also needs to have the same trend.

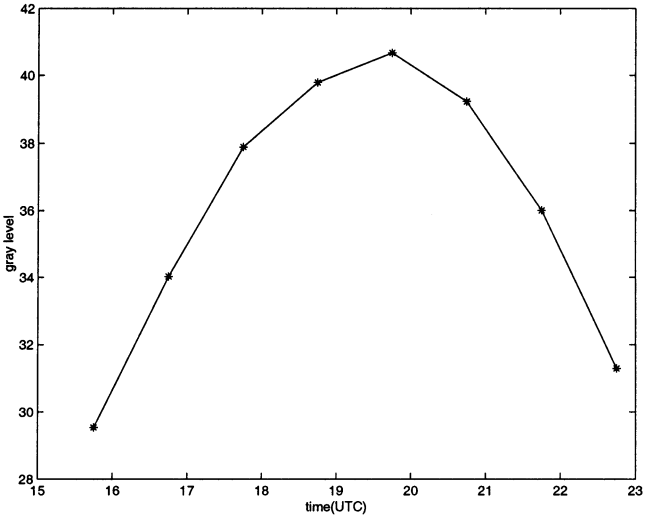


Fig. 6. The variation of the threshold between cloud and land.

V. CONCLUSION

This paper proposes a new framework for spatial–temporal updating for multispectral image classification. It was shown that the problem can be decoupled into the design of a spatial classifier and a spatial–temporal predictor. A Markov-based predictor was introduced that exploits spatial–temporal correlations in two consecutive frames. A simple iterative updating mechanism was also proposed for adjusting the parameters of the classifier(s) and the predictor. The spatial–temporal updating was used to develop a new adaptive pixel-based cloud classification system for GOES-8 satellite imagery. The system uses a cloud detector, to differentiate between cloud and no-cloud pixels, and a cloud classifier system using Bayesian methodology to classify the cloud detected areas into three possible classes depending on the cloud heights. The results on a sequence of GOES-8 images indicated the usefulness of the temporal updating scheme. Future work should include the extension of this system to handle more specific cloud classes, the inclusion of surface observations in the training of the classifiers, and the addition of corrections for solar zenith angle (visible) and limb-darkening (IR).

APPENDIX

EM Algorithm for (9)

Using EM algorithm to maximize the log likelihood function in (9) involves two steps. In the expectation (E) stage, the Q function [11] is

$$Q(\Phi|\Phi^{\text{old}}) = \sum_{i=l,c}\left[\sum_{g_{1k}\in S_2}\frac{\alpha_i^{\text{old}}p_i(g_{1k}|\phi_i^{\text{old}})}{p_2(g_{1k}|\Phi^{\text{old}})}\right]\log(\alpha_i) + \sum_{i=l,c}\sum_{g_{1k}\in S_2}\log(p_i(g_{1k}|\phi_i))\frac{\alpha_i^{\text{old}}p_i(g_{1k}|\phi_i^{\text{old}})}{p_2(g_{1k}|\Phi^{\text{old}})} + \sum_{g_{1l}\in S_1}\log(p_l(g_{1l}|\phi_l)). \quad \text{(A.1)}$$

In the maximization (M) stage, the parameters that maximize Q function are:

$$\begin{aligned}
\alpha_i^{new} &= \frac{1}{N}\sum_{g_{1k}\in S_2} w_{k,i}, \quad i = l, c \\
\mu_1^{new} &= \frac{\sum_{g_{1k}\in S_2} g_{1k}w_{k,l} + \sum_{g_{1i}\in S_1} g_{1i}}{M + \sum_{g_{1k}\in S_2} w_{k,l}} \\
\sigma_l^{2^{new}} &= \frac{\sum_{g_{1k}\in S_2} w_{k,l}(g_{1k} - \mu_l^{old})^2 + \sum_{g_{1i}\in S_1}(g_{1i} - \mu_l)^2}{M + \sum_{g_{1k}\in S_2} w_{k,l}} \\
\mu_c^{new} &= \frac{\sum_{g_{1k}\in S_2} g_{1k}w_{k,c}}{\sum_{g_{1k}\in S_2} w_{k,c}} \\
\sigma_c^{2^{new}} &= \frac{w_{k,c}(g_{1k} - \mu_c^{old})^2}{\sum_{g_{1k}\in S_2} w_{k,c}}
\end{aligned} \tag{A.2}$$

where $w_{k,i} = \alpha_i^{old} p_i(g_{1k}|\phi_i^{old})/p_2(g_{1k}|\Phi^{old})$, for $g_{1k} \in S_2$, N is the number of samples in S_2 and M is the number of samples in S_1.

REFERENCES

[1] J. J. Simpson and J. I. Gobat, "Improved cloud detection in GOES scenes over land," *Remote Sens. Environ.*, vol. 52, pp. 36–54, 1995.

[2] S. Q. Kidder and T. H. Vonder Haar, *Satellite Meteorology: An Introduction*. New York: Academic, 1995.

[3] G. S. Pankiewicz, "Pattern recognition techniques for identification of cloud and cloud systems," *Meteorol. Appl.*, vol. 2, pp. 257–271, Sept. 1995.

[4] A. Arking, "Latitudinal distribution of cloud cover from Tiros III photographs," *Science*, vol. 143, pp. 569–572.

[5] R. Koffler, A. G. DeCotiis, and P. K. Rao, "A procedure for estimating cloud amount and height from satellite infrared radiation data," *Mon. Wea. Rev.*, vol. 101, pp. 240–243.

[6] J. J. Simpson and J. I. Gobat, "Improved cloud detection in GOES scenes over the oceans," *Remote Sens. Environ.*, vol. 52, pp. 79–94, 1995.

[7] P. M. Tag, R. L. Bankert, and L. R. Brody, "An AVHRR multiple cloud-type classification package," *J. Appl. Meteorol.*, vol. 39, pp. 125–134, February 2000.

[8] B. Tian, M. R. Azimi-Sadjadi, T. H. Vonder-Haar, and D. L. Reinke, "Temporal updating scheme for probabilistic neural network with application to satellite cloud classification," *IEEE Trans. Neural Networks*, vol. 11, pp. 903–920, July 2000.

[9] B. Tian, M. A. Shaikh, M. R. Azimi-Sadjadi, T. H. Voner-Haar, and D. L. Reinke, "A study of cloud classification with neural networks using spectral and textural features," *IEEE Trans. Neural Networks,* , vol. 10, pp. 138–151, Jan. 1999.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc. Series B*, vol. 39, pp. 1–38, 1977.

[11] R. A. RedNer and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Rev.*, vol. 26, Apr. 1984.

Jianqi Wang received the B. Eng degree in automation from Tsinghua University, Beijing, China, in 1997, the M. Eng degree in pattern recognition and intelligent systems from Shanghai Jiao Tong University, Shanghai, China, in 2000, and the M.S degree in electrical and computer engineering from Colorado State University, Fort Collins, in 2002. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN.

His research interests include statistical signal/image processing, neural networks, and pattern recognition.

Mahmood R. Azimi-Sadjadi received the M.S. and Ph.D. degrees from the Imperial College of Science and Technology, University of London, London, U.K., in 1978 and 1982, respectively, both in electrical engineering with specialization in digital signal/image processing.

He is currently a Full Professor in the Electrical and Computer Engineering Department at Colorado State University (CSU), Fort Collins, where he is also serving as the Director of the Digital Signal/Image Laboratory. His main areas of interest include digital signal and image processing, target detection, classification and tracking using broadband sonar, radar and IR systems, adaptive filtering and system identification, and neural networks. His research efforts in these areas resulted in over one hundred fifty journal and refereed conference publications. He is the coauthor of the book *Digital Filtering in One and Two Dimensions* (New York: Plenum, 1989).

Dr. Azimi-Sadjadi was the Recipient of the 1999 the ABELL Teaching Award, the 1993 ASEE-Navy Senior Faculty Fellowship Award, the 1991 CSU Dean's Council Award, and the 1984 DOW chemical Outstanding Young Faculty Award. He served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and is currently serving as an Associate Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS.

Donald Reinke has extensive experience in automated processing meteorological satellite data, satellite derived cloud analyses, scientific programming, and real-time meteorological support systems management. He is currently a Research Associate at the Cooperative Institute for Research in the Atmosphere (CIRA), with a primary position of: Implementation Manager for the CloudSat Data Processing Center. His professional experience includes duty as a Research Scientist and Project Manager. He has been a project manager for a number of multi-year research programs related to the processing and analysis of digital meteorological satellite imagery and related ancillary data sets. He managed the computer lab and data processing facility, including the local area network and all hardware and software systems used by researchers at CIRA. He currently manages the Research Support group that provides data, data analysis computer applications, and technical support to CIRA. He has the responsibility for the collection of environmental data from eight worldwide meteorological satellite platforms and two data service networks. He has been a lead or co-author on more than seventy publications since arriving at CIRA.