PROGRESSIVE CLASSIFICATION IN THE COMPRESSED DOMAIN FOR LARGE EOS SATELLITE DATABASES[1]

Vittorio Castelli *Chung-Sheng Li* *John Turek* *Ioannis Kontoyiannis*[2]

IBM T.J. Watson Research Center,
Box 704, Yorktown Heights, NY10598

Appeared in *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech and Signal Processing,* Atlanta, GA, May 1996.

ABSTRACT

We introduce a new framework for classifying large images that is more accurate and less computationally expensive than the classical pixel-by-pixel approach. This approach, called progressive classification, is well suited for analyzing large images, such as multispectral satellite scenes, compressed with wavelet-based or block-transform-based transformations. These transformations produce a multiresolution pyramid representation of the data. A progressive classifier analyzes the image at the coarsest resolution level, and it decides whether each coefficient corresponds to a homogeneous block of pixels in the original image or to a heterogeneous block. In the first case it labels the block, in the second case it recursively analyzes the region of the image at the immediately finer resolution level. Computational efficiency, compared to the classical approach, results from examinining a much smaller number of coefficients than the number of pixels in the original image. Thus, progressive classification is a prime candidate as a content-based search operator for remotely-sensed data.

1. INTRODUCTION

The effective management and retrieval of large scientific spatial data sets has become increasingly important in many fields. In particular, existing space platforms that collect remotely sensed data for earth science studies have already created new challenges for encoding, transmitting, compressing, archiving, retrieving and distributing the mass amount of datasets. New platforms will be soon operative, such as the first two Earth Observing System (EOS) satellites, scheduled to be launched in 1998 and 2000, respectively. These new platforms will generate data at the impressive average rate of 26 Mb/s [1]. Content-based video and image retrieval is a rapidly evolving area of technology that allows users to specify queries using color, texture and shape descriptors [2, 3, 4, 5]. This technology offers an appealing answer to the problem of searching earth-science databases effectively (see, for example, Samadani, Han and Katragadda [6]).

Among content-based retrieval operators, accurate land-cover classification is essential for subsequent processing of remotely sensed data [7] and it is a powerful tool for feature extraction. Here, classification of a multispectral image means the process of labeling individual pixels or larger areas of the image, according to classes defined by a specified taxonomy. The "United States Geological Survey (USGS) level I and level II land-use/land-cover data" are examples of such classification taxonomies. The former encompasses 13 classes, the latter 44.

As a typical example, studies of deforestation or global warming trends require that regions of forest and old-ice coverage be recognized from the satellite data [8]. Thus, many studies have been devoted to classifying satellite data into a number of land-cover classes [9, 10, 11], usually on a pixel-by-pixel basis. However, most of these studies concentrate on the accuracy of the classification with respect to individual images.

In this paper we focus on performing the classification in a time-efficent manner, by operating directly on remote sensing data in the compressed domain.

2. PRELIMINARIES

We investigate both block-based transforms, such as the block Discrete Cosine Transform used in JPEG [12], and subband coding transforms, such as the pyramidal wavelet transform [13, 14]. Such transforms are the base of several effective data compression schemes, in which the image is first transformed then coded using either lossless or lossy compression techniques (run-length, Huffman and Lempel-Ziv coding being examples of the first class, scalar and vector quantization being examples of the second).

The coding schemes just mentioned are designed to allow fast decompression and at the same time, both block-based transformation and subband coding schemes restructure the image in a way that makes hierarchical land cover classification possible. It is therefore natural to consider classifiers that operate on transformed data.

We shall now describe a progressive approach to land-cover classification as it applies to images transformed with a pyramidal decomposition based on biorthogonal symmetric wavelets. It is worth noting that the same methodology can be applied with obvious modifications to the other transformation schemes described above.

The pyramidal wavelet decomposition of a function is very closely related to its multiresolution approximation [15]. On digital images, the wavelet transform is often implemented by means of separable filters: the image is fil-

[1]This work was funded in part by grant no. NASA/CAN NCC5-101.

[2]Currently on leave from the Information Systems Laboratory, Electrical Engineering Department, Stanford University.

tered row-by-row using a pair of matched filters (low-pass and band-pass), and the resulting 'transformed images' are filtered again column-by-column, usually with the same pair of filters, and they are then subsampled in both directions to produce the four image *subbands* The image subband corresponding to low-pass filtering in both direction is a half-resolution double-scale version of the original image. The other three subbands capture abrupt variations in horizontal, vertical and diagonal directions. In the classical wavelet transform, the decomposition algorithm is applied recursively to the half resolution version of the image. We shall use the term *k-th level of the pyramid* to denote the approximation of the image after k iterations; level 0 refers to the original image.

The value of a pixel at level k depends on the value of $n_k(f)$ pixels at level 0, f being the length of the analysis filter, where $n_i(f) = 2[n_{i-1}(f)] + f - 2$ and $n_0 = 1$. Yet, for biorthogonal wavelets, most of the contribution comes from only $(2^k)^2$ pixels of the original image. Thus, pixels at level k correspond roughly to nonoverlapping square blocks of side 2^k at level 0 of the pyramid.

3. PROGRESSIVE CLASSIFICATION

The first step in our proposed progressive classification scheme consists of training and applying classifiers at different levels of the pyramidal decomposition. The resulting label of a pixel at level k is then used to label the entire corresponding block at level 0. The computational advantage of this approach is immediately clear: the number of samples to be classified decreases exponentially with the multiresolution level. The downside is that the classification results are by nature blocky.

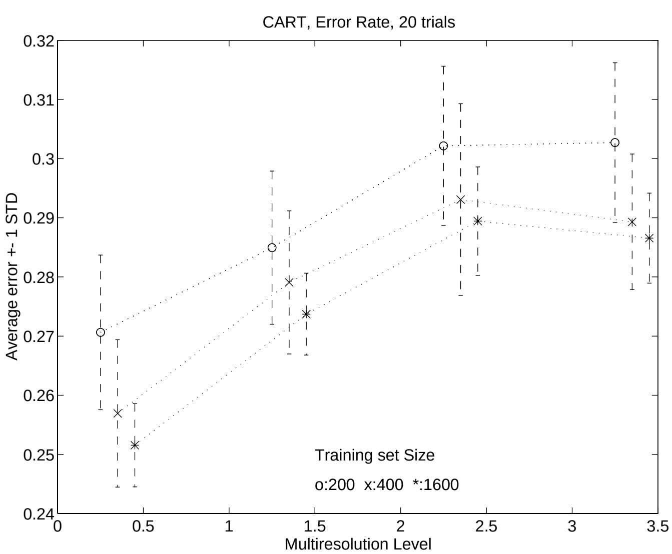

Figure 1. Error rate vs. Multiresolution Level for CART

We have studied the behavior of different types of classifiers under the described framework, in particular Gaussian classifiers (also, regrettably, known as Maximum Likelihood classifiers), k-Nearest Neighbor, Learning Vector Quantization, clustering-based schemes [16], and CART (Classification and Regression Trees) [17].

In all cases the results were surprisingly accurate: when operating on levels 1 to 3 of the pyramidal decomposition, in spite of the blocking effects, the increase in error rate was moderate: on the average the error rate at level 3 is about 4% larger than the error rate at level 0.

The dataset used in the experiments is a multispectral scanner (MSS) image of the Black Hills, taken by the Landsat 2 satellite in 1972. The image size is 512×512 pixels, and each pixel corresponds to a square on the ground of side 79 meters. The ground truth is in the form of USGS level I land-use/land-cover data. Samples from five different classes are represented in the image, namely Class 0 (Urban), Class 1 (Agricoltural Terrain), Class 2 (Range Land), Class 3 (Forest Land) and Class 6 (Barren Terrain).

Figures 1 and 2 show typical results of our experiments, obtained using CART. Analogous results have been obtained with the other classifiers described above.

Figure 1 shows the dependence of the error rate on the multiresolution level, for training sample sizes of 200, 800 and 1600. Each experiment was repeated independently 20 times, by generating at random a new training set, the error rates were recorded, the average and standard deviation were calculated and plotted.

Further investigation has revealed that pixel-based classification on the original image (level 0) is affected by the noisy nature of the data. Lowering the resolution actually filters out the noise, making the classification more robust.

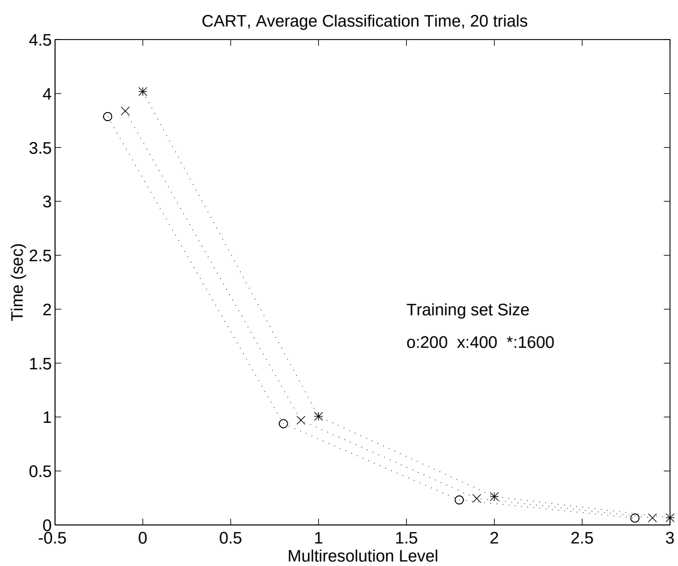

Figure 2. Classification time vs. multiresolution level for CART. The classification time decreases exponentially with the level of the multiresolution pyramid.

From the described experiments, we conclude that we are able to classify images very quickly and with reasonable accuracy by operating on low resolution approximations.

Based on these findings, we propose the following progressive approach to classification of transformed images.

For the approximation of images at level $k > 0$ we introduce a new class of samples, which we call MIX. Wavelet coefficients that correspond to blocks of pixels at level 0 having heterogeneous classification labels are labeled as MIX coefficients. MIX coefficient should not be classified at level k; instead, the corresponding 2×2 block at level $k-1$ should be analyzed. A classifier is trained for each level of the pyramid. Classifiers for levels other than 0 are trained to recognize both the original land-cover classes and the new MIX class. The image is first analyzed at the lowest resolution, say k. The non-MIX coefficients are classified at level k, and the MIX coefficients are expanded and classified by the level $k-1$ classifier; the process is iteratively repeated.

One should expect two types of benefits from the above classifier: an increase in classification accuracy and a decrease in classification time. There is clearly a tradeoff between the two benefits: one could decide to limit the number of progressive steps and increase correspondingly the classification speed at the expense of the accuracy, or do the opposite.

From the description of the algorithm, it is apparent that a crucial step is obtaining appropriate training sets. We explored two approaches to the construction of training sets for progressive classification. The first approach is the independent contruction of a training set for each level of the pyramid, the second is the joint construction of a hierarchy of training sets.

The former case, while strictly suboptimal for the problem, is simpler and more flexible. In fact, once classifiers for levels $0, 1, \ldots, L$ are trained with this scheme, one can start the progressive classification at any desired level l. The approach is suboptimal in that the training algorithm partitions the observation space into regions corresponding to the unconditional distributions of the populations at the desired level, and not on the conditional distributions given that the progressive step was taken at the immediately coarser level.

In the latter case, we fix the starting level of the progressive classification, say L, and we create a training set for that level. For each finer level, we select training points that correspond to MIX coefficients in the immediately coarser multiresolution level. Then, the classifier learns the conditional decision regions at level l given that the progressive step is taken at level $l+1$. While this approach is closer to the spirit of progressive classification, it has some drawbacks. First, it is computationally more expensive than the first approach; this, in general, is not a serious problem, since training is a one-time only operation. Also, the resulting structure is not flexible, in the sense that the starting (and ending) levels for the progressive classification are fixed. Finally, the approach requires very accurate training data, especially on the border of regions of different classification. Regrettably, in the case of satellite images very often the ground truth is known only approximately, as is the case for the data in our possession. What we have observed in practice is that the two approaches result in classifiers of comparable accuracy.

Accuracy	73%	74%	75%	76%
Progressive Classifier	800	1280	1280	1280
Standard Classifier	1280	1600	2500	4000

Table 1. Training set size as function of average accuracy for Nearest Neighbor classifier. The progressive implementation of the classifier requires a smaller training set size to achieve the same accuracy of the standard implementation.

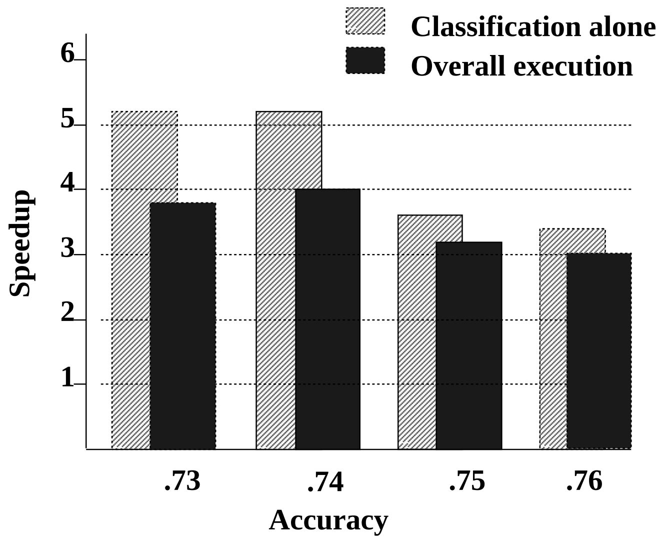

Figure 3. Speedup for Nearest Neighbor progressive classifier. The progressive classifier started at level 2 of the multiresolution pyramid. At lower accuracy a larger percentage of points are classified at coarser resolution. Thus, the progressive classifier displays higher speedup at lower accuracy.

4. EXPERIMENTAL RESULTS

We performed the experiments using an IBM workstation model 570, with 256 Kb of secondary cache and 128 Mb of main memory. The computer is based on a 50 MHz. POWER RISC processor, delivering 57.5 SPEC int92 and 99.2 SPEC fp92. The dataset used is the formerly described multispectral scanner (MSS) image of the Black Hills, taken by the Landsat 2 satellite in 1972.

Here we describe the results obtained with progressive versions of CART and of the Nearest Neighbor Classifier; the latter is based on a tree structure as described by Kim and Park in [18]. Progressiveness has been applied to levels 0 to 2 of the multiresolution pyramid, and the training sets for the different levels were constructed independently (corresponding to the first of the two approaches described in the previous section). In both cases, the progressive scheme shows better classification accuracy (by a few percent points) than the corresponding pixel-based approach on the original image (keeping the sizes of the overall training sets equal). Very significant speedup (on the order of 3 to 4 times) has been achieveded using the Nearest Neighbor (NN) classifier; recall that NN is computationally more expensive than CART.

Using CART, the current implementation of the progressive scheme does not offer significant reduction of overall execution time. This is caused mostly by the overheads deriving from the interface with our image database, that heavily penalizes decompression steps. The speedup in actual classification time is essentially the same as observed using NN.

Figure 3 summarizes the speedup achieved by our progressive algorithm applied to a NN classifier. Both average classification speedup and average overall execution speedup are depicted. The average execution time depends on the input-output costs, the pyramid inversion cost and the actual classification cost.

The overall training set size for the progressive classification algorithm is 800 points for 73% accuracy and 1280 points for the other accuracy values. Different degrees of the accuracy are obtained by controlling the amount of "progressiveness" of the algorithm, namely, by using an adaptive procedure to label samples as MIX, and by appropriately selecting the trainign set sizes for the various levels of the multiresolution.

The progressive algorithm is compared to the NN classifier that operates on uncompressed data. For the range of classification accuracy analyzed, the progressive classification scheme achieves a speedup of 3.5 to 5.3, and the overall execution time is reduced by a factor of 3 to 4. We expect better implementation of our synthesis procedure to yield overall speedups close to the speedup achieved by the classification alone.

Note that the classification accuracy of the NN operating on uncompressed data can be increased only by increasing the training set size. Table 1 summarizes the training set sizes required to achieve on average the error rates between 73% and 76%. It is apparent that, for equal training set sizes, the error rate of the progressive classifier is smaller than that of straight NN.

The progressive classification scheme can be applied to block-coded images, by using at first a very limited number of coefficients to label each block, and then a progressively increasing number of coefficients to classify those blocks labeled as MIX.

5. CONCLUSIONS

In summary, we have described a progressive approach to classification of images stored in transformed format, with particular emphasis on remotely sensed data. The progressive classifier relies on properties of the multiresolution pyramid: it starts at the coarsest level, decides whether a coefficient corresponds to a homogeneous region to a heterogeneous region in the full-resolution image. In the former case, it labels an entire block, in the latter it analyzes the image at a finer resolution level.

We have demonstrated that progressive classification is significantly faster and more accurate than the traditional pixel-based approach, thus proving a viable a tool for searching very large image databases.

REFERENCES

[1] J. C. Tilton and M. Manohar. Earth science data compression issues and activities. *Remote Sensing Reviews*, 9(1994):271–298, 1989.

[2] W. Niblack et al. The qbic project: Querying images by content using color texture, and shape. *Proc. SPIE, vol. 1908, Storage Retrieval for Image and Video Databases*, pages 173–187, 1993.

[3] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. *Proc. SPIE, vol. 2185, Storage and Retrieval for Image and Video Databases*, pages 34–47, 1994.

[4] T. Y. Hou, P. Liu, A. Hsu, and M. Y. Chiu. Medical image retrieval by spatial feature. *Proc. IEEE International Conference on System, Man, and Cybernetics*, pages 1364–1369, 1992.

[5] S. W. Smoliar and H. Zhang. Content based video indexing and retrieval. *IEEE Multimedia*, 1(2):62–72, 1994.

[6] R. Samadani, C. Han, and Katragadda L. K. Content-based event selection from satellite images of the aurora. *Proc. SPIE, Storage and Retrieval for Image and Video Databases*, 1908:50–59, 1993.

[7] E. Mesrobian et al. Extracting spatio temporal patterns from geoscience dataset. *Proc. IEEE Workshop on Visualization and Machine Vision*, pages 92–103, 1994.

[8] V. C. Barbosa, Machado R. J., and F. dos S.Liporace. A neural system for deforestation monitoring on landsat images of the amazon region. *International Journal of Approximate Reasoning*, 11:321–357, 1994.

[9] S. R. Chettri, R. F. Cromp, and M. Birmingham. Design of neural networks for classification of remotely sensed imagery. *Telematics and Informatics*, 9(3/4):145–156, 1992.

[10] R. F. Cromp and W. J. Campbell. Data mining of multidimensional remotely sensed images. *Proc. 2nd International Conference of Information and Knowledge Management*, Arlington, VA, Nov 1-5, 1993.

[11] S.R. Chettri and R. F. Cromp. Probabilisitc neural network architecture for high-speed classification of remotely sensed imagery. *Telematics and Informatics*, 10(3):187–198, Summer 1993.

[12] William B. Pennebaker. *JPEG Still Image Data Compression Standard.* Van Nostrand Reinhold, New York, 1992.

[13] Ingrid Daubechies. *Ten Lectures on Wavelets.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[14] Martin Vetterli and Cormac Herley. Wavelets and filter banks: Theory and design. *IEEE Transactions on Signal Processing*, 40(9):2207–2232, Sept 1989.

[15] Olivier Rioul and Martin Vetterli. Wavelets and signal processing. *IEEE Signal Processing Magazine*, 8(4):14–38, Oct. 1991.

[16] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, Inc., 1973.

[17] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth & Brooks/Cole, 1984.

[18] B. S. Kim and S. B. Park. A fast k nearest neighbor finding algorithm based on the ordered partition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, PAMI-8(6):761–766, Nov. 1986.