

HYPERSPECTRAL SIGNATURE CLASSIFICATION WITH TABULAR NEAREST-NEIGHBOR ENCODING

Mark S. Schmalz

*Department of Computer and Information Science and Engineering
University of Florida, Gainesville FL 32611-6120
mssz@cise.ufl.edu*

Gary Key

*Frontier Technology, Inc.
Altamonte Springs, FL 32714-2019
gkey@fti-net.com*

ABSTRACT

Accurate classification of multiple or partial multispectral or hyperspectral signatures is a crucial task in the nonimaging detection and recognition of space objects. Previous approaches to signature classification have been based on artificial intelligence techniques such as Bayesian networks, rule-based systems, and linear operators or neural networks (NNs) expressed in terms of the algebra $(\mathbf{R}, +, \times)$. Unfortunately, in these methods the separation of tightly interleaved pattern classes tends to be suboptimal, and the number of signatures that can be accurately classified often depends linearly on the number of classifier inputs. Further, such classifiers tend to be static and therefore brittle, leading to potentially significant classification errors in the presence of noise or densely interleaved signatures, as well as decreased performance in the presence of input nonergodicities.

In previous publications, the first author and his colleagues have shown that the performance of classical neural nets can be improved upon by the use of autoassociative morphological memories (AMM), a construct similar to Hopfield autoassociative memories defined on the lattice algebra $(\mathbf{R}, +, \vee, \wedge)$. Unlimited storage and perfect recall of noiseless real valued patterns has been proven for AMMs [1]. However, AMMs suffer from sensitivity to specific noise models, which can be characterized as erosive and dilative noise. It has further been shown that AMMs can be based on dendritic computation, which yields improved classification accuracy as well as class segmentation/separation ability in the presence of highly interleaved, noisy signature data [2].

In this paper, we present an improved paradigm for signature classification based on a classify-before-detect approach to pattern recognition. This technique, called *Tabular Nearest-Neighbor Encoding* (TNE), was developed for image compression using vector quantization, and has been successfully applied to a wide variety of problems in time series prediction, speech analysis, and signature classification/recognition. We show that TNE can achieve accurate signature classification in the presence of noise and closely spaced or interleaved signatures. In particular, we examine two critical cases: (1) classification of materials in spaceborne satellites using their spectral signatures, and (2) classification of multiple closely spaced signatures that are difficult to separate using TNE as well as customary distance measures. In each case, test data are derived from a NASA database of space material signatures. Additional analysis pertains to computational complexity and noise sensitivity, as well as adaptation of TNE to input sampling density, via processing of an internal Boolean array called the *agreement map*.

Keywords: Automated signature detection, Pattern recognition

1. INTRODUCTION

Non-resolved detection and classification of space objects can be achieved by spectral analysis, which depends on accurate and precise spectrometry, physically faithful data sampling techniques, and accurate, comprehensive classifier technology. Although passive remote sensing technology has advanced significantly over the past decade, yielding imaging devices with increasing spectral coverage and resolution, the development of classifier technology has not necessarily kept pace. For example, the high spectral resolution produced by current hyperspectral devices facilitates identification of fundamental materials (spectral endmembers [3-6]) that comprise remotely sensed objects, thus supporting spectral discrimination based on parameters (e.g., abundance fractions) derived from such measures.

However, in non-resolved space object detection and classification, it is not necessarily possible to determine endmembers *a priori*. Instead, the decomposition of an object’s spectral signature requires solution of demixing equations that are predicated upon prior knowledge of the materials that might comprise an object, as well as their relative abundances in an hypothetical object representation, for example, a digital signature obtained by telescopic spectrometry [3,4]. Thus, the object classification problem in the non-resolved case depends primarily on whether or not the classifier that is applied to the hypothetical abundance fractions can distinguish the spectral endmembers. If this is not possible, then the object signature cannot be compared accurately or precisely with signatures from a catalogue of objects.

In this paper, we discuss recent developments in an emerging classifier technology called *tabular nearest-neighbor encoding* (TNE), which is a highly efficient, accurate paradigm for pattern recognition developed by Frontier Technology, Inc. [7]. The development of TNE proceeded naturally from the development of signal and image compression algorithms based on vector quantization, as described in Section 2.1. In this study, TNE is evaluated for its applicability in non-resolved space object characterization, in particular, for distinguishing endmember spectra. We assume the case of *non-imaging detection and classification*, where a spectrometer is affixed to a telescope that does not detect a space object as an image, but as a collection of scalar intensities at different wavelengths. We analyze the ability of TNE to classify these endmembers in relationship to a common spectral comparison technique called *Euclidean distance*, from which is derived mean-squared error.

This paper is organized as follows. In Section 2, we overview the theory of tabular nearest-neighbor encoding in relationship to vector quantization, then describe its practical implementation in salient detail. Section 3 provides a summary of classification test results on a database of eight materials found in domestic satellite applications. Noise, sampling density, and thresholding considerations are discussed, and TNE is found to be superior to the Euclidean distance in each test case. Conclusions and suggestions for future work are given in Section 4.

2. THEORETICAL SUMMARY

We begin with an overview of vector quantization (Section 2.1) that is couched in terms of *image algebra*, a rigorous concise notation that unifies linear and nonlinear mathematics in the image domain [8]. We then proceed to a mathematical description of tabular nearest-neighbor encoding (Section 2.2), with complexity analysis presented in Section 2.3.

2.1. Overview of Vector Quantization

Let \mathbf{X} denote an $M \times N$ -pixel image domain, and let a codebook \mathbf{c} be comprised of $K \times L$ -pixel exemplars, for convenience. Let \mathbf{a} be an image on \mathbf{X} , whose m -bit values are in the set $\mathbf{F} = \mathbf{Z}_{2^m}$ and which can be expressed as the mapping $\mathbf{a} : \mathbf{X} \rightarrow \mathbf{F}$, denoted in image algebra as $\mathbf{a} \in \mathbf{F}^{\mathbf{X}}$. Assume that this image is mathematically subdivided into $K \times L$ -pixel encoding blocks. Further assume that $\mathbf{c} \in (\mathbf{F}^{KL})^{\mathbf{N}}$, where $\text{domain}(\mathbf{c})$ is an indexing set.

Denote a VQ-compressed image as $\mathbf{a}_c \in \mathbf{N}^{\mathbf{Y}}$, where \mathbf{Y} denotes a subset of \mathbf{X} (for example, an M/K by N/L -pixel domain), such that each index in $\text{range}(\mathbf{a}_c)$ is in $\text{domain}(\mathbf{c})$. Let an indexing function $h : \mathbf{X} \rightarrow \mathbf{Y}$ map a point in the source domain to a point in the compressed domain, such that a point \mathbf{x} in the domain of the \mathbf{y}^{th} encoding block (itself a subset of \mathbf{X}) is mapped to $\mathbf{y} \in \mathbf{Y}$. Further assume that h has a dual $h^* : \mathbf{Y} \rightarrow 2^{\mathbf{X}}$, which returns the domain of the \mathbf{y}^{th} encoding block, from which the compressed pixel value $\mathbf{a}_c(\mathbf{y})$ is derived.

Given the preceding discussion, a VQ transform can be expressed at a high level as

$$T : (\mathbf{Z}_{2^m})^{\mathbf{X}} \times (\mathbf{Z}_{2^m}^{KL})^{\mathbf{N}} \rightarrow \mathbf{N}^{\mathbf{Y}} , \quad (1)$$

such that $\mathbf{a}_c = T(\mathbf{a}, \mathbf{c})$. In practice, the codebook is understood, so we write $\mathbf{a}_c = T(\mathbf{a})$. Decompression is similarly expressed via the dual transformation

$$T^* : \mathbf{N}^{\mathbf{Y}} \times (\mathbf{Z}_{2^m}^{KL})^{\mathbf{N}} \rightarrow (\mathbf{Z}_{2^m})^{\mathbf{X}} , \quad (2)$$

which is invoked to yield an approximation to the source image denoted by $\mathbf{b} = T^*(\mathbf{a}_c) \approx \mathbf{a}$. Since VQ is a lossy transformation, the inverse transform T^{-1} does not necessarily exist.

In order to understand the relationship between VQ and TNE, we need to consider the effect of codebook size. For example, we note that when VQ is applied to image compression applications, the compression ratio CR of $T: \mathbf{F}^X \rightarrow \mathbf{G}^Y$ is given by

$$CR(T) = \frac{|\mathbf{X}|}{|\mathbf{Y}|} \cdot \frac{siz(\mathbf{F})}{siz(\mathbf{G})} \quad (3)$$

where $|S|$ denotes cardinality of the set S , and $siz(S)$ denotes the number of bits required to encode S (for example, in terms of linear quantization).

Given a codebook domain \mathbf{U} , the compression ratio of the VQ transform T can be expressed in terms of its domain compression ratio CR_D and its range compression ratio CR_R , as follows:

$$CR_D = \frac{|\mathbf{X}|}{|\mathbf{Y}|} = \frac{|\mathbf{X}|}{|\mathbf{X}|/KL} = KL \quad \text{and} \quad CR_R = \frac{siz(\mathbf{F})}{siz(\mathbf{G})} = \frac{m}{\log|\mathbf{U}|} . \quad (4)$$

The compression ratio of a VQ transform, absent of the codebook, is thus given by:

$$CR = CR_D \cdot CR_R = \frac{KLm}{\log|\mathbf{U}|} . \quad (5)$$

Given CR , K , L , and m , the codebook size, also denoted by Q for brevity, is constrained by the following relationship:

$$Q \equiv |\mathbf{U}| \leq 2^{KLm/CR} . \quad (6)$$

Implementationally, it is important to note that the VQ codebook can be large, necessitating efficient search algorithms in the computational implementation of Equation (1). Again assuming encoding in terms of the KL -element encoding block or sample vector, we observe that Q can be computed from CR and other parameters, so we can use Q to estimate VQ codebook search overhead.

For example, if linear search of codebook exemplars is implemented sequentially, then each of $|\mathbf{Y}| = |\mathbf{X}|/KL$ source blocks are compared with each of Q codebook exemplars comprised of KL pixels, for a total cost of $Q \cdot |\mathbf{X}|$ comparisons. Since KL pixel-level comparisons are usually combined with $KL-1$ operations per block, the minimum overhead exclusive of I/O is given by $2Q \cdot |\mathbf{X}|$ operations. If a more computationally involved matching criteria such as Euclidean distance implemented as mean-squared error (MSE) are employed, then the work involved in determining a best-match exemplar for each source block can increase considerably. For example, blockwise MSE computation would require $2KL-1$ additions, $KL+1$ multiplications, and one square root per encoding block per codebook exemplar. Denoting $N \equiv |\mathbf{X}|$ for brevity, we have the following linear-search work requirement:

$$2NQ \text{ additions} \leq W_{cs}^{lin} < 2NQ \text{ additions} + NQ \text{ multiplications} + NQ/KL \text{ square roots} . \quad (7)$$

If logarithmic search is employed, then it is well known that one can in principle decrease the preceding estimate by a factor of $\log(Q)/Q$, via a tree-structured architecture.

2.2. Tabular Nearest-Neighbor Encoding

It is possible to visualize each KL -element sampling vector as part of a subset of a larger collection of vectors from which examples can be abstracted for purposes of pattern matching. For example, each $K \times L$ -element sample vector can be seen as a point in KL -dimensional Euclidean space, denoted by \mathbf{R}^{KL} . This concept support the construction of a database of templates, each one containing KL elements, against which each input sample (modeled as a point in \mathbf{R}^{KL}) can be compared to determine its best-match exemplar or template in the database. By selecting the appropriate subset of the template database corresponding to the input patterns, a highly efficient comparison with a relatively small number of templates can be implemented.

In practice, this database of patterns can be indexed very efficiently to yield a comparison approach that involves small amounts of floating-point or integer computations, and is primarily I/O-intensive. By using a broader collection of templates, or by varying sampling density, tradeoffs between computational cost and classification accuracy can be achieved in terms of practical constraints on the pattern classification process.

This basic concept that underlies TNE indicates a comprehensive approach to pattern matching which we first overview (Section 2.2.1), then describe in detail mathematically (Section 2.2.2).

2.2.1. Overview of TNE. Assume that the spectral classification process requires the comparison of multiple data blocks or *test vectors* from different sensor/classifier assets with a library of target-related exemplar or *training set vectors*. This library does not need to be large initially, and can be augmented adaptively during the pattern classification process. TNE begins processing by placing thresholding values called *acceptance bounds* around each component value of each test vector, then determine the number of components from each training set exemplar that fall within these bounds. A binary vector is generated for each training set exemplar, as shown in Figures 1, 3 and 4. This binary vector indicates the components of the test vector that agree with a particular database exemplar. Note that the intervals of agreement can be all equal or weighted, as appropriate. At the heart of the TNE paradigm is a procedure for very efficiently deriving this *agreement map* for the entire pattern library without significant computation – only a few logical or integer arithmetic operations are required.

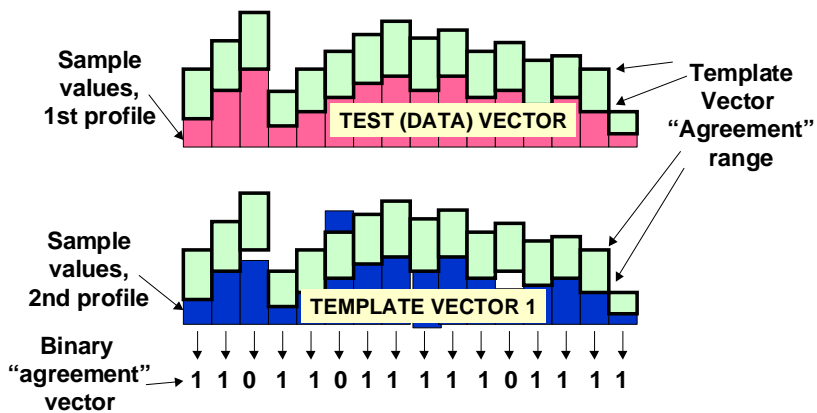


Figure 1. TNE derives a binary agreement vector, where a 1 signifies that the associated components agree within a pre-specified tolerance.

As Figure 2 indicates, the TNE algorithm constructs a binary vector for each quantization level along each of the *KL* coordinate axes (vector block dimensions). This binary vector contains a bit for each exemplar in the training set. If the *KL*-dimensional boundary corresponding to the agreement intervals about each training set exemplar projects onto the associated quantization interval, a 1 is assigned, otherwise a 0.

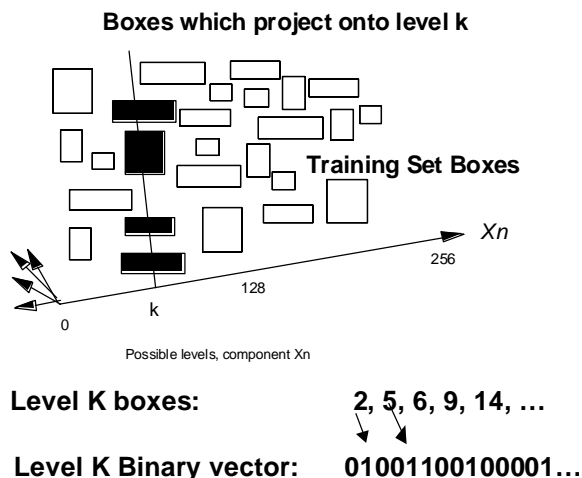


Figure 2. Derivation of binary agreement vectors for each quantization level along each dimension.

The utility of these binary agreement vectors is that once they are constructed, a *virtual agreement map* for any new test vector can be specified immediately by simply using the *KL* component values of a test vector as pointers into the larger, pre-existing table. Again, recall that this table can be prespecified or computed adaptively. Also, as

shown in Figure 3 binary agreement vectors can be combined logically to facilitate inferences with respect to test vector classification.

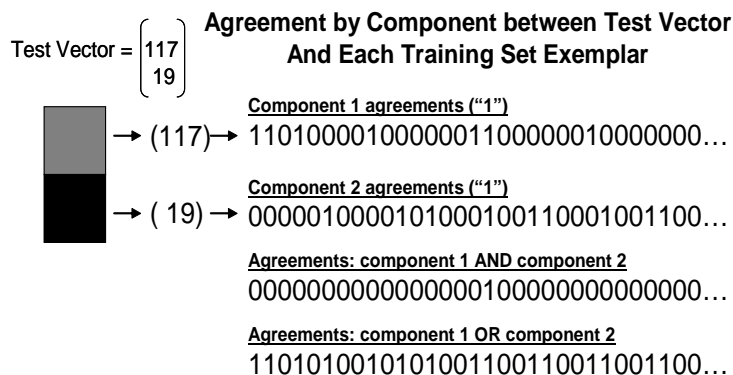


Figure 3. Example of a two-dimensional system (i.e., two pixels per sampling vector) that illustrates utility of the binary agreement vectors. Note that only one training set exemplar matches both components of the test vector.

Each dimension of the search space \mathbf{R}^{KL} can be any numerically represented object. We have thus successfully applied TNE to a wide variety of multi-data and multi-sensor classification problems. For example, detection of a launch- or boost-phase event can be structured as shown in Figure 4, where multiple sensors that each output multiple values (e.g., spectra) which can be represented in terms of the agreement map. Here, each dimension of the search space has its own binary pointer table, from which the binary agreement vector is derived via I/O operations only. Thus, TNE is an efficient paradigm for multi-data and multi-sensor fusion, as well as pattern classification.

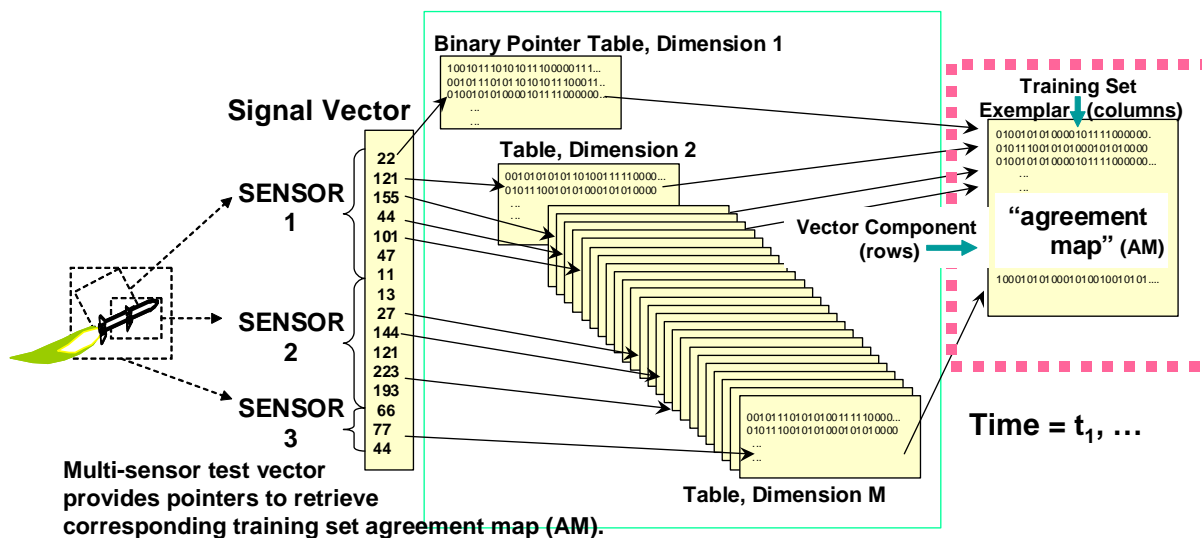


Figure 4. Specification of a single (virtual) agreement map via test vector component pointers, derived in this example from three fused sensor data types.

In order to understand the relationship of TNE to spectral signature classification, we next discuss the mathematical implementation of TNE, then proceed to a complexity analysis in Section 2.3.

2.2.2. Mathematical Description of TNE. Let an image $\mathbf{a} \in \mathbf{F}^X$ be subdivided by an indexing function h to yield a collection of $K \times L$ -pixel encoding blocks $A = \{\mathbf{b}(\mathbf{y}) : \mathbf{y} \in \mathbf{Y}\}$, where $\mathbf{Y} \subset \mathbf{X}$. Let a codebook \mathbf{c} be formed from A , such that \mathbf{c} contains Q $K \times L$ -pixel exemplars, each of which represent a cluster C_i , where $i = 1..Q$. Let a feature space representation F have axes $B_1, B_2, \dots, B_j, \dots, B_P$, to which are projected each of the clusters C_i , thereby producing a collection of intervals denoted by

$$I = \{I_{i,j} \in \mathbf{R}^2 : 1 \leq i \leq Q \text{ and } 1 \leq j \leq P\}. \quad (8)$$

2.2.2.1. Algorithm. Let an encoding block $\mathbf{b}(\mathbf{y})$ be represented by a point or region \mathbf{p} in feature space F . Let \mathbf{p} be projected to axes $B_j, j = 1..P$, to yield a collection of intervals denoted by

$$J = \{J_{ij} \in \mathbf{R}^2 : 1 \leq i \leq Q \text{ and } 1 \leq j \leq P\}. \quad (9)$$

Although J_{ij} is one-dimensional when \mathbf{p} is a point, we assume that J_{ij} is two-dimensional, for purposes of generality.

Step 1. Let I and J be processed by an operation that compares the extent of J_{ij} with the extent of I_{ij} , such that a $P \times Q$ -element bitmap \mathbf{d} is formed, as follows:

$$\mathbf{d}(i, j) = \begin{cases} 1 & \text{if } p_1(J_{ij}) \geq p_1(I_{ij}) \text{ and } p_2(J_{ij}) \leq p_2(I_{ij}) \\ 0 & \text{otherwise} \end{cases}. \quad (10)$$

Step 2. As an example of pattern recognition, sum \mathbf{d} rowwise, and subtract P as to yield scores $\mathbf{g}(i) = P - \Sigma \mathbf{d}_i$, where \mathbf{d}_i denotes the i -th row of \mathbf{d} . The resultant scores equal the Hamming distances between \mathbf{p} and each exemplar $\mathbf{c}(i)$ represented by cluster C_i .

The best-match codebook exemplar is given by $\mathbf{c}(\text{choice}[\text{domain}(\min(\mathbf{g}))])$.

2.2.2.2. Observation. It is readily seen by comparison of Sections 2.1 and 2.2.2.1 that TNE provides an efficient means for VQ codebook search. In imaging applications, TNE can be used to compress a hyperspectral datacube \mathbf{a} by indexing each encoding block according to the spatial configuration of its values. That is, a given pixel $(\mathbf{x}, \mathbf{a}(\mathbf{x}))$ of \mathbf{a} provides both spatial and grayscale information to a map $D : \mathbf{X} \times \mathbf{R} \rightarrow \mathbf{G}$, where \mathbf{R} denotes $\text{domain}(\mathbf{a})$ and \mathbf{G} is an indexed set of pointers to Q -bit Boolean vectors stored in database D . Each vector represents one of the KL pixels of a given encoding block \mathbf{b} . In the resulting $Q \times KL$ -pixel array \mathbf{d} , which is called the *agreement map*, the j -th column represents a bit vector of binary matching scores between (a) value $\mathbf{b}(\mathbf{x})$ at position \mathbf{x} of $\text{domain}(\mathbf{b})$ indexed by j , and (b) all exemplar values $\mathbf{c}(i)(\mathbf{x})$, where $i = 1..Q$. The exemplar that best matches \mathbf{b} is given by

$$\mathbf{c}(\text{choice}(\text{domain}(\min(KL - \Sigma \mathbf{d}_j)))) , \quad (11)$$

where \mathbf{d}_i denotes the i -th row of \mathbf{d} .

2.3. Complexity Analysis of TNE

Assuming that the TNE codebook cluster projections are precomputed, projection of \mathbf{p} to the axes of F would naively require $\mathbf{O}(P)$ arithmetic and transcendental operations per source block, for example, P sine operations and $2P$ additions. Comparison of I and J requires $2PQ$ comparisons per source block, with $P(Q+1)$ additions required to produce \mathbf{g} . Similarly, Q comparisons are required to find the best-match exemplar in \mathbf{c} . In principle, the work required by a naïve implementation of the TNE codebook search over \mathbf{a} is given by:

$$W_{TNE} = MN(Q(2P+1) \text{ comparisons} + P(Q+3) \text{ additions} + P \text{ transcendentals}). \quad (12)$$

It is readily verified that the precomputation of D is the burdensome step in the TNE algorithm, which can be compared to the overhead of codebook construction in VQ. For example, if each encoding block has KL pixels each having G graylevels, then G^{KL} block configurations are possible. Comparison of these configurations with the Q codebook exemplars yields a total cost of $W = \mathbf{O}(KLQG^{KL})$ comparison operations. Given typical values in hyperspectral imagery of $K, L = 16$, $Q = 256$, and $G = 256$, it is easily verified that W is prohibitively large. Hence, it is reasonable to determine the subset S of the G^{KL} block configurations that occurs in a given training set. Given S , W can be reduced to $\mathbf{O}(KLQ \cdot |S|)$ comparison operations. For example, if $|S| = 10^5$ and the proportionality constant in the complexity estimate of W is set to unity for purposes of simplicity, then $W = 256^3 \times 10^5 = 1.67$ GOPs. In contrast, the non-imaging signature recognition task reported in this paper uses $K = 1$, $L \leq 2.2 \times 10^4$, $Q \leq 800$, and $|S| = 640$, yielding $W = 11.2$ MOPs, which is well within real-time processing rates for existing workstations.

It has been argued that the preceding analysis misleads the reader into believing that TNE is not a computationally efficient classifier. In practice, the efficiency of TNE derives from the ability to construct the agreement map from a precomputed collection D of binary pointer tables, using *I/O operations only*. This allows TNE to run efficiently on machines with large local or shared memory models, where the majority of D is memory-resident. In such cases, we have determined that TNE significantly outperforms traditional classifiers such as Euclidean or Mahalanobis distance operators, because TNE requires no floating-point arithmetic or multiplication.

Furthermore, in the classifier refinement step that involves agreement map processing, TNE exhibits very significant efficiency gains over the aforementioned distance-based classifiers, because only bitwise logical operations as well as integer column sums are required for processing of the TNE agreement map. For example, given a $P \times Q$ -bit agreement map \mathbf{d} , constructed per Equation (10), processing of all columns of \mathbf{d} by pointwise logical operators requires PQ logical operations. Formation of a column sum requires a maximum of P incrementations.

Since current workstation employ processors with efficient vector computation, the actual cost of these logical and integer operations is very low. For example, in the tests described in Section 3, $22 \leq P \leq 22,000$ and $8 \leq Q \leq 800$, so 32-fold parallelism in bitwise logic operations reduces the dominant work to a lower limit of $5.5 = 8(22) / 32$ invocations of a logical vector operation (e.g., ≤ 6 machine cycles of an Itanium processor) and an upper limit of $550,000 = 800(22,000) / 32$ invocations (implying approximately 550,000 machine cycles on the same type of processor), exclusive of I/O overhead. As in the preceding analysis of computational work, this timing constraint is well within real-time processing capabilities of existing workstations.

3. APPLICATION OF TNE TO SIGNATURE CLASSIFICATION

The TNE algorithm was tested against the common Euclidean distance measure, from which metrics such as MSE are derived. The composition of our signature database is given in Section 3.1, with test procedures given in Section 3.2. Classifier performance results are discussed in Section 3.4.

3.1. Signature Database

A test database of eight spectral signatures, adapted from the NASA database of space material signatures, was selected, as shown in Figure 5. The spectral materials are as follows: (1) Hubble aluminum, (2) Hubble green glue, (3) Solar cell, (4) Black rubber edge, (5) Bolts, (6) Copper stripping, (7) Hubble honeycomb side, and (8) Hubble honeycomb top. Spectra were subsampled from original NASA data at 0.1 micron wavelength intervals.

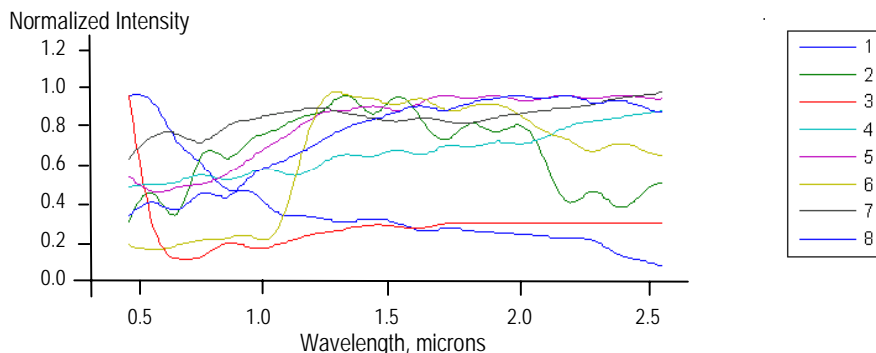


Figure 5. Normalized test spectra, adapted from NASA database of space materials.

These materials are chosen because their signatures differ in the visible and near-infrared (NIR) spectral regions, but five materials are similar in the IR region, while three materials (#2, #3 and #8) differ significantly in the IR from the remaining spectra.

3.2. Test Procedures

In this paper, two illustrative test cases are evaluated, with the objective of comparing TNE classification accuracy against Euclidean distance classification accuracy. In the first case (Figure 6), data sampled at 0.1 micron wavelength intervals were resampled via linear interpolation at 0.01 micron intervals, to provide a denser test set of 220 points per spectrum. The resampled test set was perturbed by random, Gaussian-distributed additive noise at standard deviation ranging from $0 \leq \sigma \leq 0.8$.

In the second case (Figure 7), data sampled at 0.1 micron intervals were resampled via linear interpolation at 0.0001 microns, to provide a denser test set of 22,000 points per spectrum. As in the first case, the resampled test set was perturbed by random Gaussian-distributed additive noise at standard deviation $0.01 \leq \sigma \leq 0.8$.

Classification accuracy was measured as follows. If a given spectrum of the $N_S = 8$ spectra is distinguished by TNE or the Euclidean distance metric from $N_S - N_C$ other spectra, then the classification accuracy is defined as

$$\eta = 1 - \frac{N_C}{N_S} \quad (13)$$

Thus, if $N_C = 0$, then $\eta = 1.0$. Similarly, if $N_C = 7$, then $\eta = 1 - 7/8 = 1/8 = 0.125$. This simple metric thus allows straightforward visualization of results in terms of a grayscale or pseudo-colored image (shown in Figures 6 and 7), of which the x -axis (rows) denotes spectrum by number ranging from 1 through 8, and the y -axis (columns) denotes the level of additive noise.

3.3. Experimental Results

As shown in Figures 6 and 7, TNE superiorly classified the eight endmember spectra, when compared with the Euclidean distance metric. In Figure 6a (10X resampling), TNE correctly classified all spectra at $0 \leq \sigma \leq 0.23$, and mis-classified only two spectra at $0.24 \leq \sigma \leq 0.36$. In contrast, the Euclidean distance algorithm (Figure 6b) *never* classified all eight spectra correctly, being confounded by materials #1 and #3 at very low noise levels ($0 \leq \sigma \leq 0.03$), above which Euclidean classifier performance degraded significantly (e.g., five of eight spectra were mis-classified at $\sigma = 0.11$). Above, $\sigma = 0.15$, the Euclidean classifier exhibited progressively more severe deficits, as shown by the black pixels (seven out of eight spectra confused with test spectrum) in Figure 6b.

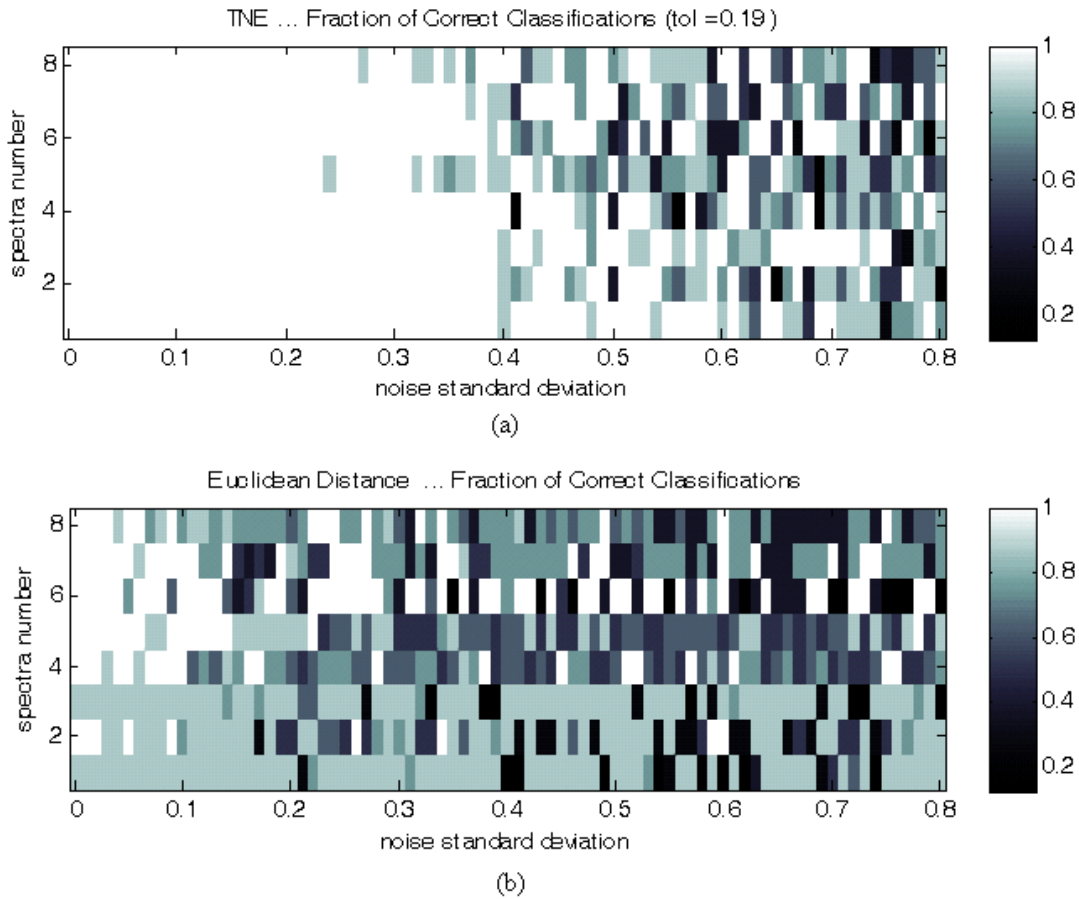


Figure 6. Comparison of (a) TNE classification accuracy with (b) Euclidean distance classification accuracy, for eight spectra (shown in Figure 5), sampled at 0.1 micron wavelength intervals, then resampled at 10X to yield 220 samples per spectra. White pixels in the performance chart indicate perfect classification of a given spectrum, while black pixels indicate very poor classification, as shown by the gradient bar to the right of each performance chart.

Interestingly, TNE performs significantly better on larger samples, that is, when the resampling factor is high. For example, in Figure 7a (1000X resampling), *all* spectra are classified correctly by TNE up to the high noise level $\sigma = 0.45$, which is a significant improvement over the 10X resampling case shown in Figure 6a. Among all tests performed in this case (640 tests = 80 noise levels x 8 spectra), only 15 tests resulted in mis-classification, which

indicates a probability of correct TNE classification of $P_C = 1 - 15/640 = 0.977$, as opposed to $P_C = 1 - 207/640 = 0.677$ (207 mis-classifications in Figure 6a) for a 10X resampling factor.

In contrast, the Euclidean distance classification performance was no better in Figure 7b as opposed to Figure 6b, and its P_C was worse than chance over all 640 measurements in each test case. Due to small-sample statistical constraints imposed by the weak law of numbers, there was no statistically significant difference in the performance of the Euclidean distance metric between Figure 6b and Figure 7b. Also, it is visually apparent in Figure 7a that TNE's classification performance improved drastically with respect to Figure 6a.

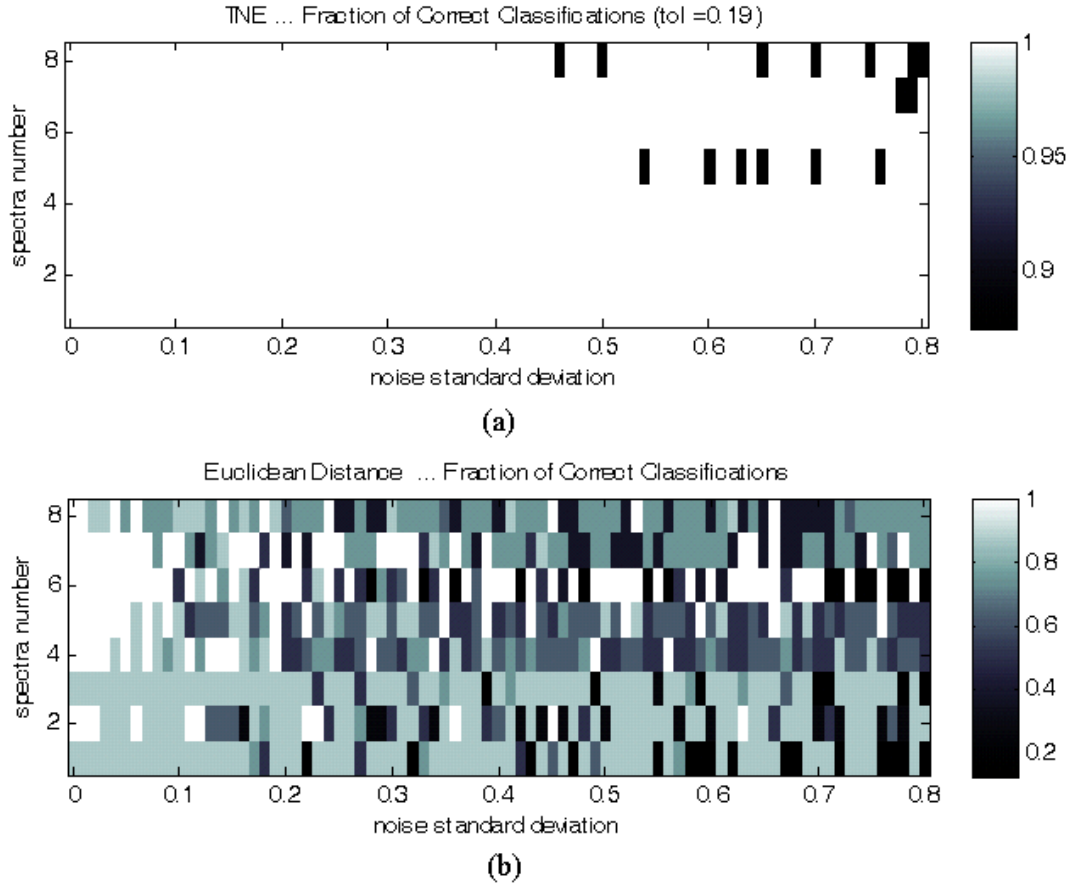


Figure 7. Comparison of (a) TNE classification accuracy with (b) Euclidean distance classification accuracy, for eight spectra (shown in Figure 5), sampled at 0.1 micron wavelength resolution, then resampled at 1000X to yield 22,000 samples per spectra. White pixels in the performance chart indicate perfect classification of a given spectrum, while black pixels indicate very poor classification, as shown by the gradient bar to the right of each performance chart.

The reason for TNE's improved performance in Figure 7a, as opposed to Figure 6a, is the increased amount of data available for processing in the TNE agreement map **d** (per Equation 10). Because we are using a simple Hamming distance metric to compute matching scores in the agreement map processing step, each sampling vector, and therefore each column of the agreement map, has 10X more data (Figure 6) than the original 22-point spectrum shown in Figure 5, and 1000X the data (Figure 7) of each original spectrum. As a result of increasing the number P of rows in **d**, the signal-to-noise ratio of the Hamming distance metric improves by two orders of magnitude between Case 1 (shown in Figure 6) and Case 2 (Figure 7). This supports much higher resolution in discriminating correct matches, per the notional diagram of Figure 1. Additionally, this indicates that TNE would be expected to perform better on hyperspectral than multi-spectral data.

A further advantage of TNE versus other types of published classifiers is that the TNE agreement map tells us (a) which patterns were classified correctly or nearly correctly, with what error (if any), and why this classification occurred; as well as (b) which patterns were classified incorrectly, with what error, and why. As a result, we are able to adaptively configure the TNE agreement map to intelligently maximize the classification probability P_C .

4. CONCLUSIONS

Accurate non-imaging detection and recognition of space objects requires accurate classification of spectral signatures that comprise pure materials of which an object is comprised. In the spectral domain, the signatures of these materials are called *endmembers*, can be mixed to produce a composite signature that characterizes a given object. It follows that if one cannot discriminate accurately among spectral endmembers, then one cannot accurately discriminate object signatures. Unfortunately, the vast majority of classifiers currently in use in pattern recognition practice perform poorly when spectral signatures are closely spaced, interleaved, or significantly noise-corrupted [9]. Such classifiers include metrics derived from the Euclidean distance, for example, mean-squared error.

This paper presents an emerging technology for pattern classification, called *tabular nearest-neighbor encoding* (TNE), which has proven highly successful when applied to spectral signature classification. Developed initially as an efficient search engine for vector quantization in image and signal compression, TNE employs a highly-dimensional search space represented in terms of a binary array called the *agreement map* (AM). Processing of the AM requires only bitwise logical operations and a few integer additions or incrementation operations, which can be performed in vector-parallel fashion. TNE is thus highly efficient, and very well suited for implementation on parallel or embedded processing architectures.

In this paper, we demonstrate that TNE performs superiorly to the Euclidean distance measure when classifying noise-corrupted signatures of eight spectra adapted from the NASA database of space materials. In particular, TNE classifies all signatures completely correctly up to Gaussian noise level at standard deviation $\sigma = 0.23$ over the unitary intensity interval, given a 10X input resampling factor. Over the 640 tests (Case 1) conducted for this resampling case, TNE's probability of classification was $P_C = 0.677$, while the Euclidean classifier yielded P_C no better than chance. When the resampling factor was increased to 1000X, TNE classified all spectra correctly up to the noise level $\sigma = 0.45$ at $P_C = 0.977$ for the 640 tests in Case 2, while the performance of the Euclidean distance classifier did not improve statistically. We also show that TNE performs superiorly in the sense of computational cost, requiring only I/O operations, bitwise logical operators, and integer incrementation or addition to achieve the aforementioned classifier performance levels. Future work emphasizes development of more intelligent agreement map processing techniques, as well as enhancement of TNE's adaptive classification capabilities.

5. REFERENCES

- [1] Schmalz, M.S. and G.X. Ritter. "Hyperspectral endmember extraction and signal classification with morphological networks", in Proceedings of the AMOS 2006 Conference, Maui HI, 2006.
- [2] Ritter, G.X and M.S. SCHMALZ. "Learning in lattice neural networks that employ dendritic computing", in Proceedings of the IEEE 2006 Conference on Fuzzy Systems, 2006.
- [3] Adams, J.B., M.O. Smith, and P.E. Johnson, "Spectral mixture modeling: A new analysis of rock and soil types at the Viking Lander 1 site", Journal of Geophysical Research, Vol. 91, 8098-8112, 1986.
- [4] Boardman, J.W., "Automated spectral unmixing of AVIRIS data using convex geometry concepts", AVIRIS Workshop Proceedings, JPL Publication 93-26, Jet Propulsion Laboratory, Pasadena, CA, pp. 11-14, 1993.
- [5] Roberts, D.A., M. Gardner, R. Church, S. Ustin, G. Sheer, and R.O. Green, "Mapping chaparral in the Santa Monica mountains using multiple endmember spectral mixture models", Remote Sensing of the Environment, Vol 65, 267-279, 1998.
- [6] Ritter, G.X. and M.S. SCHMALZ. "Fast autonomous endmember detection in hyperspectral images based on lattice theory", in Proceedings of the Eighth International Conference on Natural Computing, JCIS Multiconference, Salt Lake City UT, 2006.
- [7] Key, G., M.S. SCHMALZ, F.M. Caimi, and G.X. Ritter. "Performance analysis of tabular nearest neighbor encoding algorithm for joint compression and ATR", in *Proceedings SPIE* Vol. 3814, 115-126, 1999.
- [8] Ritter, G.X. and J.N. Wilson. Handbook of Computer Vision Algorithms in Image Algebra, Second Edition, CRC Press, Boca Raton, FL, 2000.
- [9] Schmalz, M.S., Recent Advances in Error Analysis Techniques and Systems for Astronomical Image Processing, Proceedings of the 2002 AMOS Conference, Maui HI, 295-305, 2002.