NOISE-TOLERANT HYPERSPECTRAL SIGNATURE CLASSIFICATION IN UNRESOLVED OBJECT DETECTION WITH ADAPTIVE 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 spectral signature classification is a crucial step in the nonimaging detection and recognition of spaceborne objects. In classical hyperspectral recognition applications, especially where linear mixing models are employed, signature classification accuracy depends on accurate spectral endmember discrimination. In selected target recognition (ATR) applications, previous non-adaptive techniques for signature classification have yielded class separation and classifier refinement results that tend to be suboptimal. In practice, the number of signatures accurately classified often depends linearly on the number of inputs. This can lead to potentially severe classification errors in the presence of noise or densely interleaved signatures.

In this paper, we present an enhancement of an emerging technology for nonimaging spectral signature classification based on a highly accurate, efficient search engine called Tabular Nearest Neighbor Encoding (TNE). Adaptive TNE can optimize its classifier performance to track input nonergodicities and yield measures of confidence or caution for evaluation of classification results. Unlike neural networks, TNE does not have a hidden intermediate data structure (e.g., a neural net weight matrix). Instead, TNE generates and exploits a user-accessible data structure called the *agreement map* (AM), which can be manipulated by Boolean logic operations to effect accurate classifier refinement through programmable algorithms. The open architecture and programmability of TNE's pattern-space (AM) processing allows a TNE developer to determine the qualitative and quantitative reasons for classification accuracy, as well as characterize in detail the signatures for which TNE does not obtain classification matches, and why such mis-matches occur.

In this study AM-based classification has been modified to partially compensate for input statistical changes, in response to performance metrics such as probability of correct classification (Pd) and rate of false detections (Rfa). Adaptive TNE can thus achieve accurate signature classification in the presence of time-varying noise, closely spaced or interleaved signatures, and imaging system optical distortions. We analyze classification accuracy of closely spaced spectral signatures adapted from a NASA database of space material signatures. Additional analysis pertains to computational complexity and noise sensitivity, which are superior to non-adaptive TNE or Bayesian techniques based on classical neural networks.

Keywords: Automated signature detection, Pattern recognition

1. INTRODUCTION

Non-resolved detection and classification of space objects using features such as spectral signatures, or mixtures of such signatures, requires accurate, comprehensive signature classification technology [1,2]. Although passive remote sensing research has advanced significantly over the past decade, yielding imaging devices with increasing spectral coverage and resolution, the development of classifier technology has progressed more slowly. For example, the high spectral resolution produced by current hyperspectral devices facilitates the description of signatures that represent fundamental materials (spectral endmembers) comprising remotely sensed objects. This directly supports spectral discrimination based on signature-based parameters (e.g., abundance fractions). However, the demixing equations and resulting classification of abundance fraction vectors can be noise-sensitive.

In non-resolved space object detection and classification, one cannot necessarily determine spectral endmembers *a priori*. Instead, the decomposition of an object's spectral signature requires solution of demixing equations that are predicated upon *a priori* knowledge of signatures that might comprise an object's constituent materials, 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 the ability of a signature classifier to distinguish spectral endmembers.

In this paper, we discuss new developments in an adaptive classifier technology called *tabular nearest-neighbor encoding* (TNE [7,10]), which is a highly efficient, accurate paradigm for pattern recognition developed by Frontier Technology, Inc. and University of Florida. In [7.10], we showed that the development of TNE proceeded naturally from the development of signal and image compression algorithms based on vector quantization (VQ). In this study, TNE is evaluated for its applicability in distinguishing highly noise-corrupted endmember spectra in *non-imaging detection and classification* applications. Here, 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 similar spectral endmembers under conditions of high noise, when compared with a common spectral comparison technique called *Euclidean distance*, from which are derived various mean-squared error measures previously employed in assessing classification accuracy [2].

This paper is organized as follows. In Section 2, we overview the theory of tabular nearest-neighbor encoding. Section 3 provides a summary of classification test results on a database of eight materials found in domestic satellite applications, with levels of Gaussian noise corruption ranging from a standard deviation of 0.0 to 0.9 over the spectral intensity interval [0,1]. Noise, sampling density, and thresholding considerations are discussed, and TNE is found to be superior to the Euclidean distance in each test case. The space complexity of TNE is compared with array-based classifiers such as neural networks (NNs). Conclusions and suggestions for future work are given in Section 4.

2. THEORETICAL SUMMARY

We begin with an overview of tabular nearest-neighbor encoding (Section 2.1) and its implementation (Section 2.2), with complexity analysis presented in Section 2.3.

2.1. Overview of TNE

Let $\mathbf{X}$ denote an MxN-pixel image domain, and let a codebook $\mathbf{c}$ be comprised of KxL-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 disjoint KxL-pixel sampling vectors. Further assume that $\mathbf{c} \in (\mathbf{F}^{KL})^{\mathbf{N}}$, where *domain*($\mathbf{c}$) is an indexing set.

As shown in [7,10], it is possible to visualize each KxL-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 sampling vector can be seen as a point in KL-dimensional Euclidean space, denoted by $\mathbf{R}^{KL}$. This concept supports the construction of a template database, where each template contains KL elements, against which each input sample (modeled as a point in $\mathbf{R}^{KL}$) can be compared to determine its best-match template. By selecting the appropriate subset of the template database corresponding to an input (test) pattern, 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.

2.1.1. Mathematical Description. Let an image $\mathbf{a} \in \mathbf{F}^{\mathbf{X}}$ be subdivided mathematically by an indexing function h, to yield a collection of KxL-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 KxL-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, ..., B_j, ..., B_P, to which are projected each of the clusters C_i, thereby producing a collection of intervals denoted by

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

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

$$J = \{J_{i,j} \in \mathbf{R}^2 : 1 \le i \le Q \text{ and } 1 \le j \le P\}. \qquad (2)$$

Although $J_{i,j}$ is one-dimensional when **p** is a point, we assume that $J_{i,j}$ is two-dimensional, for purposes of generality.

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

$$\mathbf{d}(i,j) = \begin{cases} 1 & \text{if } p_1(J_{i,j}) \ge p_1(I_{i,j}) \text{ and } p_2(J_{i,j}) \le p_2(I_{i,j}) \\ 0 & \text{otherwise} \end{cases}. \qquad (3)$$

Step 2. As an example of pattern recognition, sum **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 **d**. The resultant scores equal the Hamming distances between **p** and each exemplar **c**(*i*) represented by cluster C_i.

Step 3. The best-match codebook exemplar is given by **c**(*choice*[*domain*(*min*(**g**))]).

2.2. Implementation

It is readily seen that TNE provides an efficient means for searching a pattern database. In imaging applications, TNE has been successfully applied to compression of a hyperspectral datacube **a** by indexing each encoding block according to the spatial configuration of its values. That is, a given pixel (**x**,**a**(**x**)) of **a** provides both spatial and grayscale information to a map $D : \mathbf{X} \times \mathbf{R} \to \mathbf{G}$, where **R** denotes *domain*(**a**) and **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 **b**. In the resulting *Q*x*KL*-pixel array **d**, which is called an *agreement map,* the *j*-th column represents a bit vector of binary matching scores between (a) value **b**(**x**) at position **x** of *domain*(**b**) indexed by *j*, and (b) all exemplar values **c**(*i*)(**x**), where *i* =1..*Q*. The exemplar that best matches **b** is given by

$$\mathbf{c}(choice(domain(min(KL - \Sigma\mathbf{d}_i)))) , \qquad (4)$$

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

Figure 1 illustrates this concept notionally by diagramming a collection of pattern clusters by their bounding boxes in pattern space *F*. Suppose we have *n* dimensions in the pattern space, such that a group of pattern clusters (shown as darkened boxes in Figure 1) that project onto the X_n^{th} axis can be assigned the value $1 \le k \le n$. We call this process *projective derivation of agreement vectors*, as depicted in Equation (2).

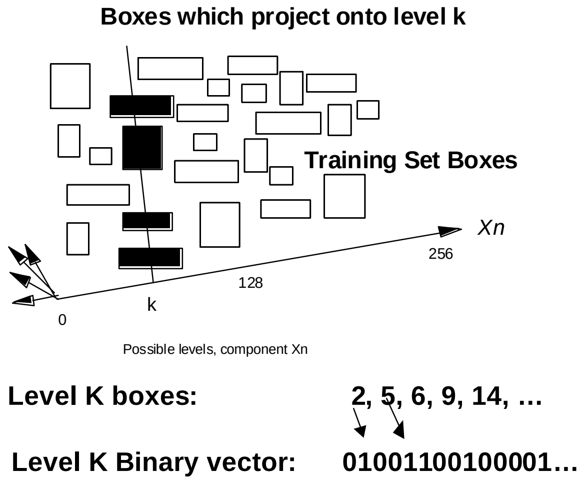


Figure 1. Derivation of agreement vectors for each quantization level along each dimension of pattern space.

The agreement vectors, formed and collected in a database called the *pointer table*, can be indexed via Equation (3). This supports construction of an agreement map, as shown schematically for binary agreement vectors in Figure 2, which shows that an agreement map is specific to a given sample at a given time. Thus, for time-variant data, higher-dimensional AMs can be constructed using this process, albeit at a cost of higher space complexity and significantly increased work requirement relative to the agreement map depicted in Figure 2. Furthermore, the agreement map is not restricted to binary values – in this study, we achieve greater noise tolerance by employing gradient comparisons that populate the AM with positive integer values.

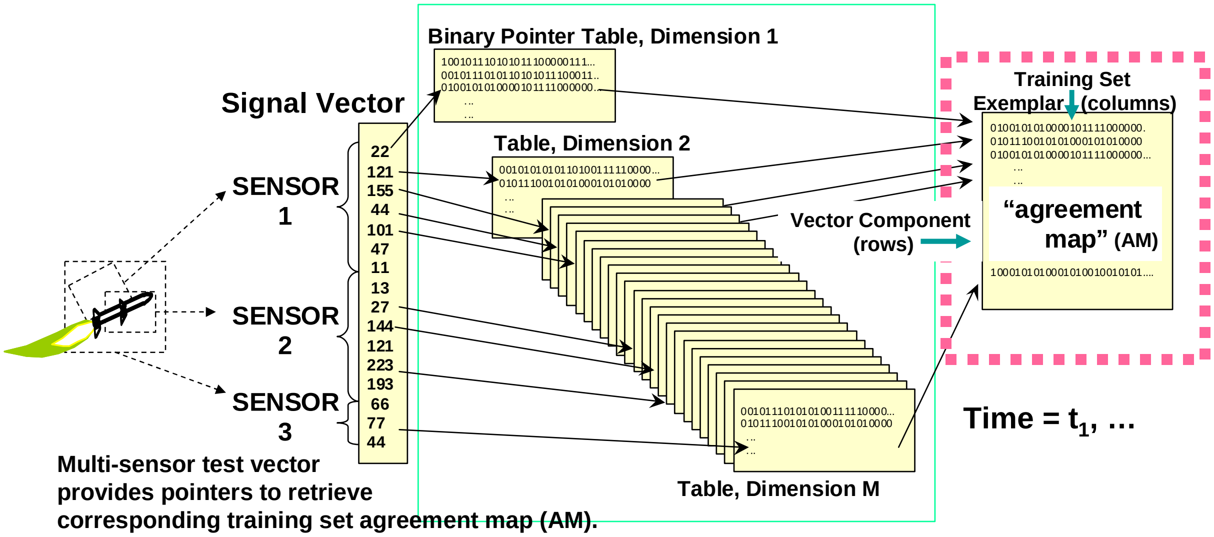


Figure 2. Specification of a single (virtual) agreement map via test vector component pointers, derived in this example from three fused sensor data types.

As mentioned previously, when variance is introduced at the input, complications occur in any known pattern classification paradigm, primarily because of sensor effects or sampling error that propagate through the classifier's arithmetic operations. In Figure 1, this would correspond to the blurring of sampling vector values, which blurs the boundaries of pattern clusters. When making TNE adaptive, we introduced a gradient matching scheme that allows the values *P* and *Q* in Equations (1) and (2) to vary with the amount and type of input noise. For example, if Gaussian noise is encountered, then *P* and *Q* are perturbed by a Gaussian distribution. This implies that the matching process depicted in Equation (3) is gradient-driven, and that the resulting agreement map can be real- or integer-valued. As a result, a wider variety of strategies for processing the agreement map are available.

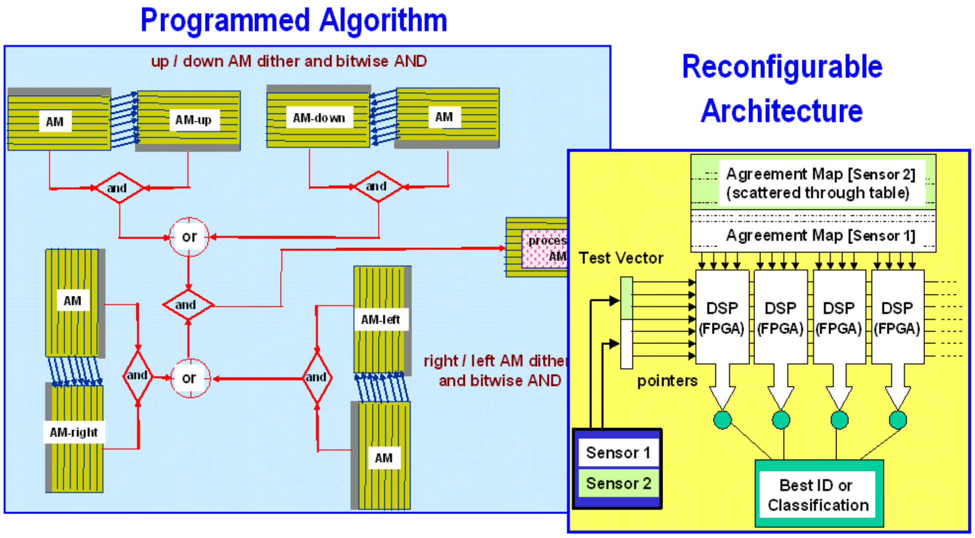


Figure 3. Example of pattern-space processing using visual programming and a reconfigurable architecture.

For example, to exploit these expanded opportunities for agreement map processing, we are currently developing a programming interface that will support visual programming of pattern space operations, as shown schematically in Figure 3. Here, the pattern-space processing algorithm uses operations such as logical *and*, bitwise or columnwise logical *or*, dithering of the agreement map values, and integer summation to remove effects of noise and clutter *in the pattern space*, versus postprocessing in the image domain. This allows highly efficient operations due to (a) simplicity of the processing operations (e.g., logical operations and integer summation) and (b) relatively small size of the agreement map, which can be restricted to one or more pattern types, for further efficiency.

An important implementational advantage of TNE is that the agreement map construction process requires I/O operations only. Also, the AM configuration is directly accessible to TNE algorithm developers at any time during the pattern classification process. Unlike other types of classifiers that have "black box" or hidden data structures (e.g., neural net weight matrices), it is possible to view and process the agreement map values as they are computed per Equation (3). This provides algorithm developers with the following three principal advantages:

1. At all times during the development process, the AM and its formation process can be interrogated to determine the effects of input variance on the pattern classification process. This is important to the results presented in this paper, due to our emphasis on making TNE noise-tolerant.
2. Examination of the AM yields two types of information about early classification: (a) correct matches, and why the template(s) matched a given input; and (b) incorrect matches, and why each template did not match input. This is an important feature of TNE that helps developers design more accurate early classification steps and classifier refinement algorithms.
3. Programming of the pattern-space operations, especially debugging of pattern-space processing algorithms, would be difficult without TNE's visible data structures.

An additional advantage of TNE is the simplicity of its operations and relatively low computational complexity, which we analyze as follows.

2.3. Complexity Analysis of TNE

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

$$W_{TNE} = MN(Q(2P+1) \text{ comparisons} + P(Q+3) \text{ additions} + P \text{ transcendentals}) . \qquad (5)$$

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 $W = 256^3 \text{x} 10^5 = 1.67$ GOPs. In contrast, the non-imaging signature recognition task reported in this paper uses *K* = 1, $L \leq 2.2 \text{ x } 10^4$, $Q \leq 800$, and |*S*| = 640, yielding *W* = 11.2 MOPs, which is well within real-time processing rates for embedded processors.

It has been argued that the preceding analysis indicates 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 computationally outperforms traditional classifiers such as Euclidean or Mahalanobis distance operators, because TNE requires no floating-point arithmetic or multiplication.

In the classifier refinement step that involves agreement map processing, TNE exhibits 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*x*Q*-bit agreement map

d, constructed per Equation (3), processing of all columns of **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 [10], $22 \leq P \leq 22{,}000$ 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., $\leq$ 6 machine cycles of a vector 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 available workstations or embedded processors.

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 3.3.

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 4. 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.

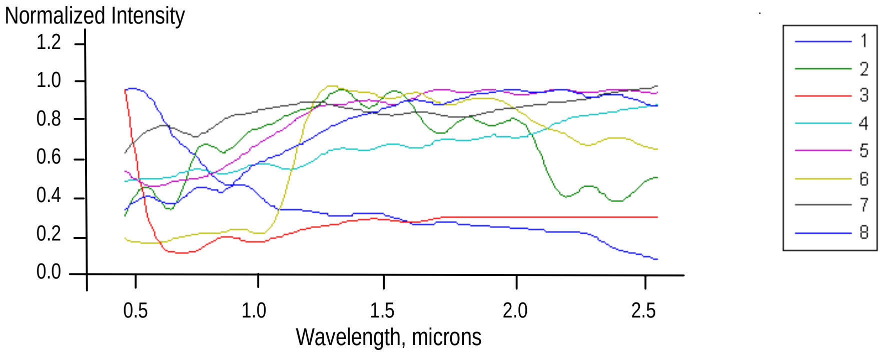


Figure 4. Normalized test spectra, adapted from NASA database of space materials.

The eight space materials were 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 primary test cases are evaluated, with the objective of comparing TNE classification accuracy against Euclidean distance classification accuracy. In the first case (Figures 5-8), 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 $0 \leq \sigma \leq 0.8$ (Figure 5) and $0 \leq \sigma \leq 0.9$ (Figures 6-8).

In the second case (Figure 6), 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 $0.01 \leq \sigma \leq 0.8$.

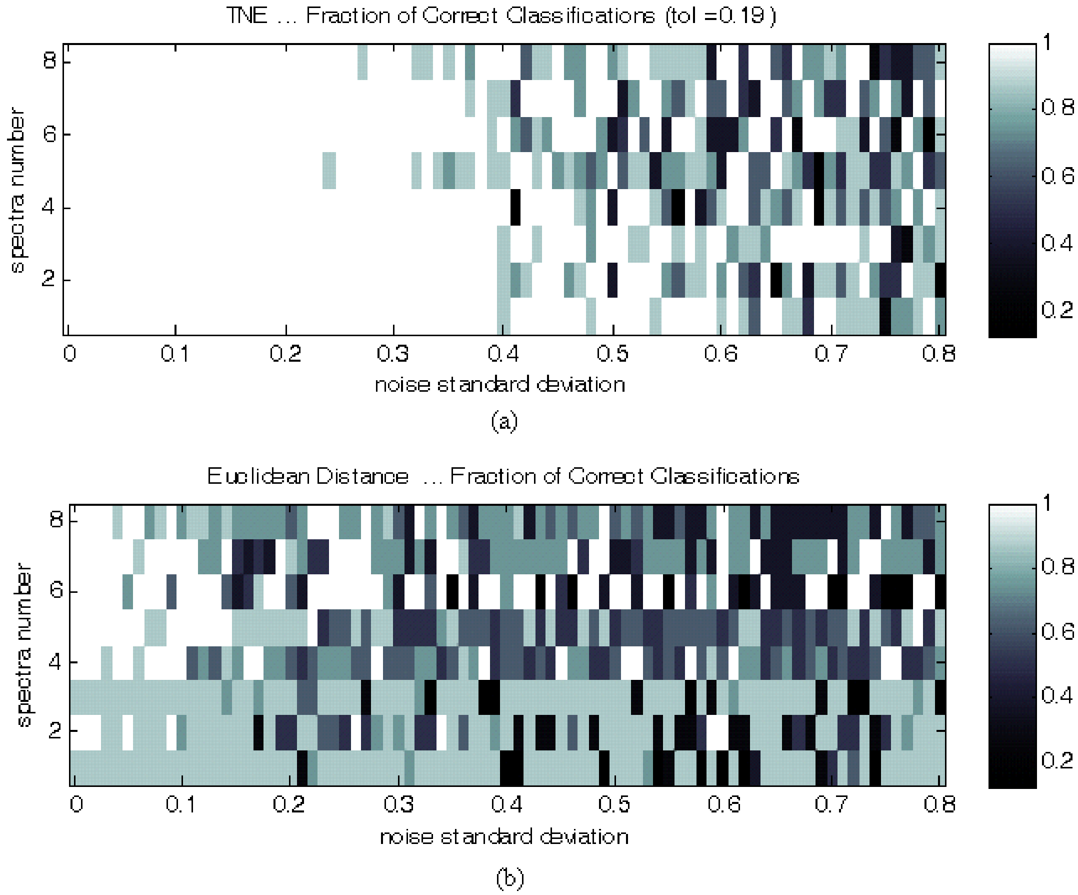
Classification accuracy was measured as follows. If a given spectrum of the $N_S = 8$ spectra is distinguished by TNE or the Euclidean distance metric from $N_S - N_C$ other spectra, then the classification accuracy is defined as

$$\eta = 1 - \frac{N_C}{N_S} . \tag{6}$$

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 allows straightforward visualization of results in terms of a grayscale or pseudo-colored image (shown in Figures 5-9), 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 5 and 6, TNE superiorly classified the eight endmember spectra, when compared with the Euclidean distance metric. In Figure 5a (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 5b) *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 5b.



(a)

(b)

Figure 5. 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.

As illustrated in Figure 6, we continued experiments in adaptation of TNE parameters by comparing existing TNE classifier results with the Euclidean distance classifier results, where the TNE agreement threshold was varied from 0.05 to 0.2. In each case, TNE performed significantly better than the Euclidean classifier (as shown by the preponderance of white pixels in the TNE performance chart, versus the Euclidean performance chart.

In contrast, the performance of the enhanced version of TNE that comprises our early prototype version of adaptive TNE is shown in Figure 7. Observe how classification scores improve for adaptive TNE, by employing gradient values in the pattern database and AM instead of binary values, as discussed in Section 2.2.

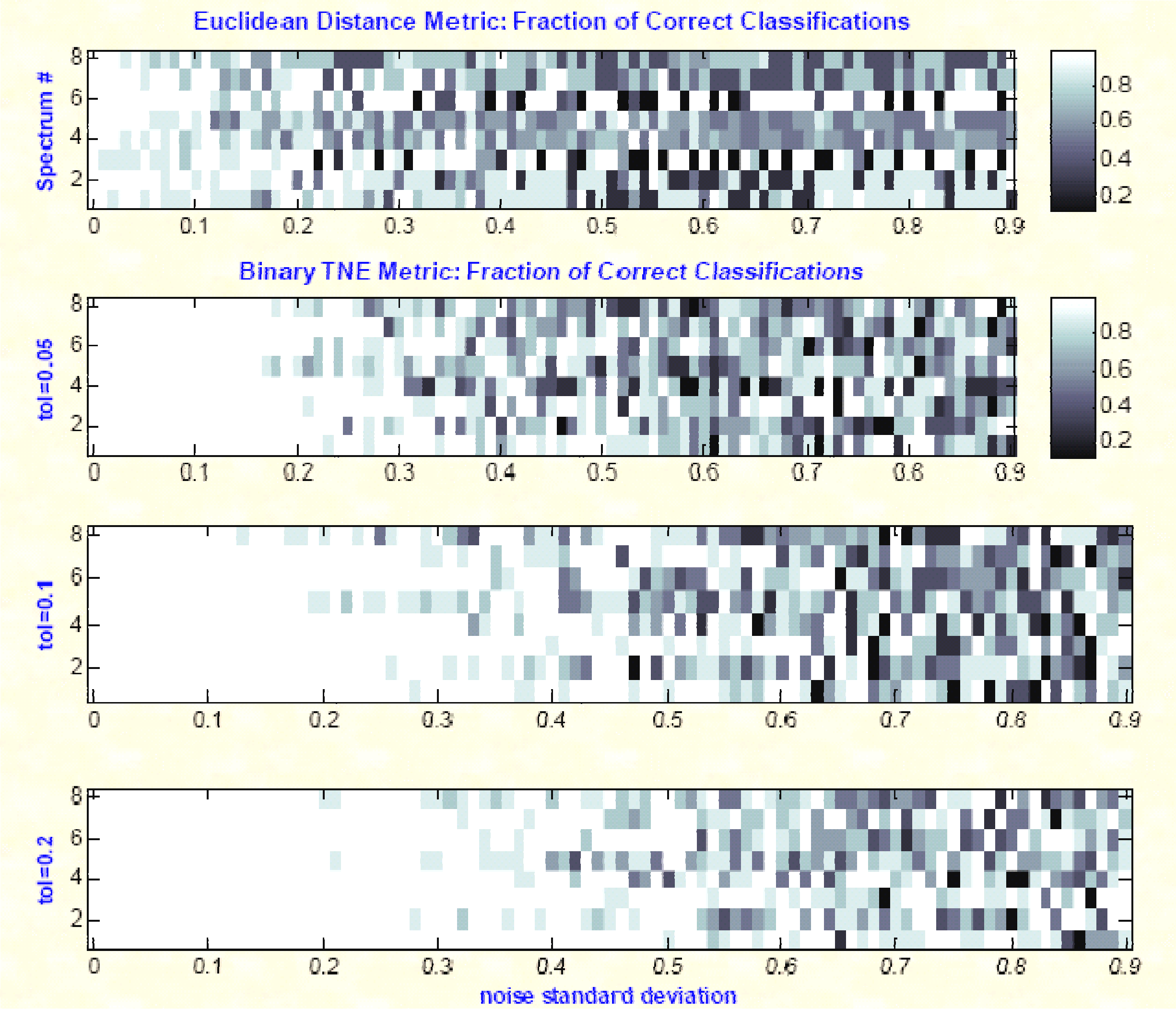


Figure 6. Performance of binary TNE versus Euclidean classifier, for three TNE agreement thresholds (T = 0.05, 0.1, and 0.2). The colored bars at the top represent classification score – a white pixel in the chart means perfect classification (P_C = 1.0) and a black pixel means totally erroneous classification.

In each of these tests, TNE superiorly classified the eight endmember spectra, when compared with the Euclidean distance metric. In Figure 7b (10X resampling), TNE correctly classified all spectra at $0 \leq \sigma \leq 0.22$, and mis-classified only two spectra at $0.24 \leq \sigma \leq 0.36$. In contrast, the Euclidean distance algorithm (Figure 7a) *never* classified all eight spectra correctly, being confounded by six of the eight spectra at very low noise levels ($0 \leq \sigma \leq 0.05$), above which Euclidean classifier performance degraded significantly (e.g., all eight spectra were mis-classified at $\sigma = 0.12$). 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 7a. However, the early prototype adaptive TNE classifier with the agreement threshold set at 0.1 correctly classified all spectra at $0 \leq \sigma \leq 0.28$, and classified all spectra with only twelve instances of erroneous classification at $0 \leq \sigma \leq 0.4$.

We next present results for the adaptive TNE classifier with gradient values in the pattern database and agreement map, as well as a very early version of a pattern replacement strategy. This allows patterns associated with low Pd and high Rfa to be deactivated (i.e., not included in classification), while patterns associated with high Pd and low Rfa are given greater weight in the classification process. A type of closed-loop optimization is thus realized, by using classifier performance as the pattern replacement metric. The results of Figures 7 through 9 again show that adaptive TNE performs superiorly to Euclidean distance and binary TNE classifiers.

In Figure 8, adaptive TNE provided significantly improved classification of the eight endmember spectra, when compared with the Euclidean distance metric, and significant improvements with respect to binary TNE. In Figure

8a (10X resampling), the Euclidean classifier nearly correctly classified all spectra at $0 \leq \sigma \leq 0.08$ (near perfect performance with 16 errors of a possible 64, where classification error metric $\varepsilon_C = 16/64 = 0.25$). In contrast, the binary TNE algorithm (Figure 8b) classified all eight spectra nearly correctly at noise levels in the interval $0 \leq \sigma \leq 0.3$, with seven errors of a possible 240 matches ($\varepsilon_C = 7/240 = 0.003$). However, the adaptive TNE classifier with the agreement threshold set at an optimal value of $T = 0.15$ (the same value as for binary TNE) nearly correctly classified all spectra at $0 \leq \sigma \leq 0.39$, with only twelve instances of erroneous classification of a possible 312 over that interval ($\varepsilon_C = 12/312 = 0.004$).

As shown previously [10], it is possible to improve TNE classification accuracy by providing more information to the classifier. We expected such improvements to occur with adaptive TNE, which occurred in our collaborative tests on the space materials database, as shown in the following discussion of Figure 9.

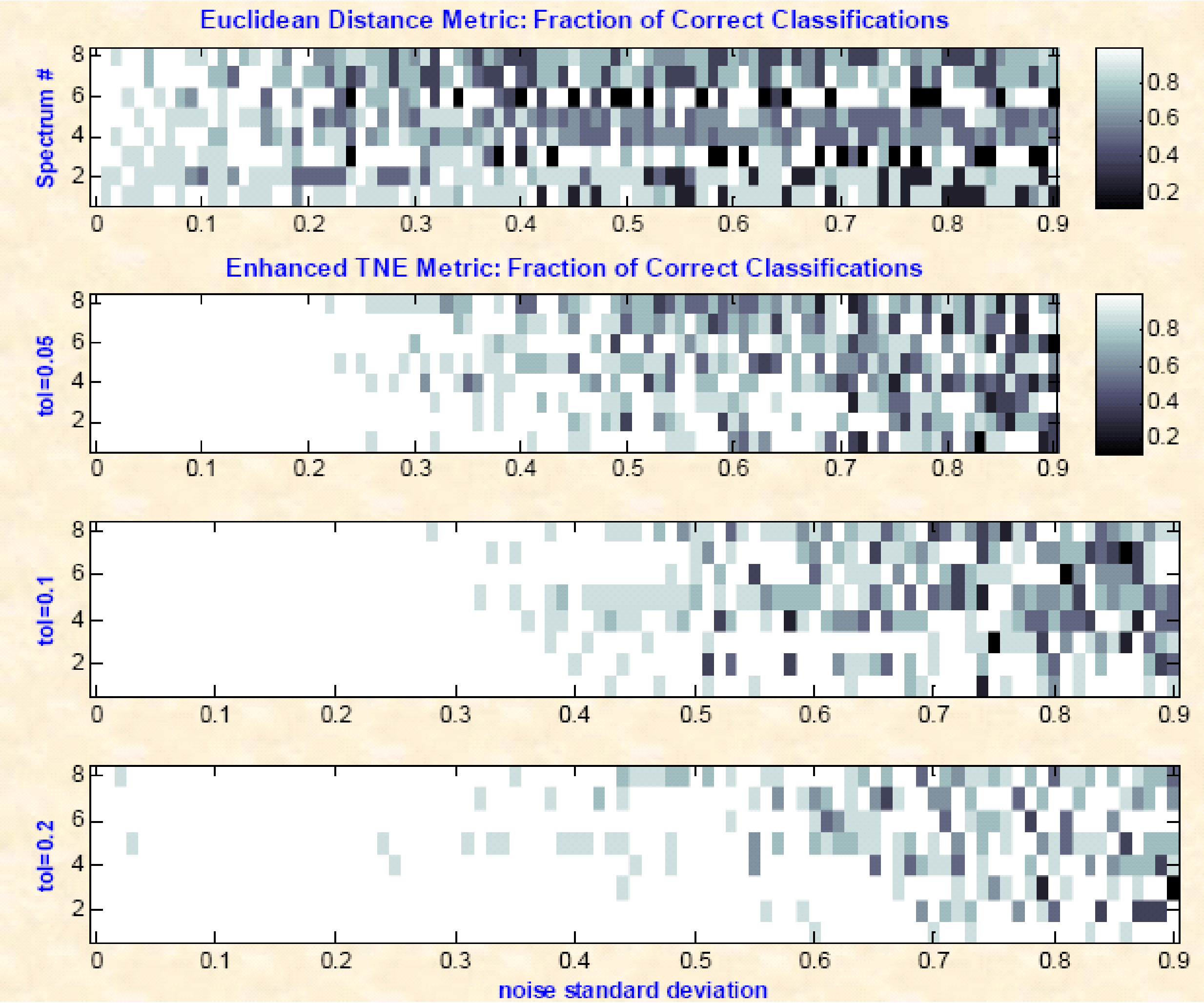


Figure 7. Performance of early prototype adaptive TNE versus Euclidean classifier, for three agreement thresholds (T = 0.05, 0.1, and 0.2). The colored bars at the top represent classification score – a white pixel in the chart means perfect classification ($P_C = 1.0$) and a black pixel means totally erroneous classification. These charts have the same format and general meaning as the charts illustrated in Figures 13 and 14.

In Figure 9 (100X resampling), adaptive TNE again provided drastically improved classification of the eight endmember spectra, when compared with the Euclidean distance metric, and significant improvements with respect to binary TNE. In Figure 9a, the Euclidean classifier nearly-correctly classified all spectra at $0 \leq \sigma \leq 0.1$, with classification error metric $\varepsilon_C = 0.35$. In contrast, the binary TNE algorithm (Figure 9b) classified all eight spectra nearly correctly at noise levels in the interval $0 \leq \sigma \leq 0.3$, with $\varepsilon_C = 0.002$). However, the adaptive TNE classifier with the agreement threshold set at an optimal value of $T = 0.15$ (the same value as for binary TNE, and in the previous test results shown in Figure 8) nearly correctly classified all spectra at $0 \leq \sigma \leq 0.42$, with $\varepsilon_C = 0.003$). This represents a consistent behavior with respect to our TNE experiments in prior studies [7,10], and approximates nearly correct classification at a signal-to-noise ratio SNR = 1.19:1 = $(2 \cdot 0.42)^{-1}$.

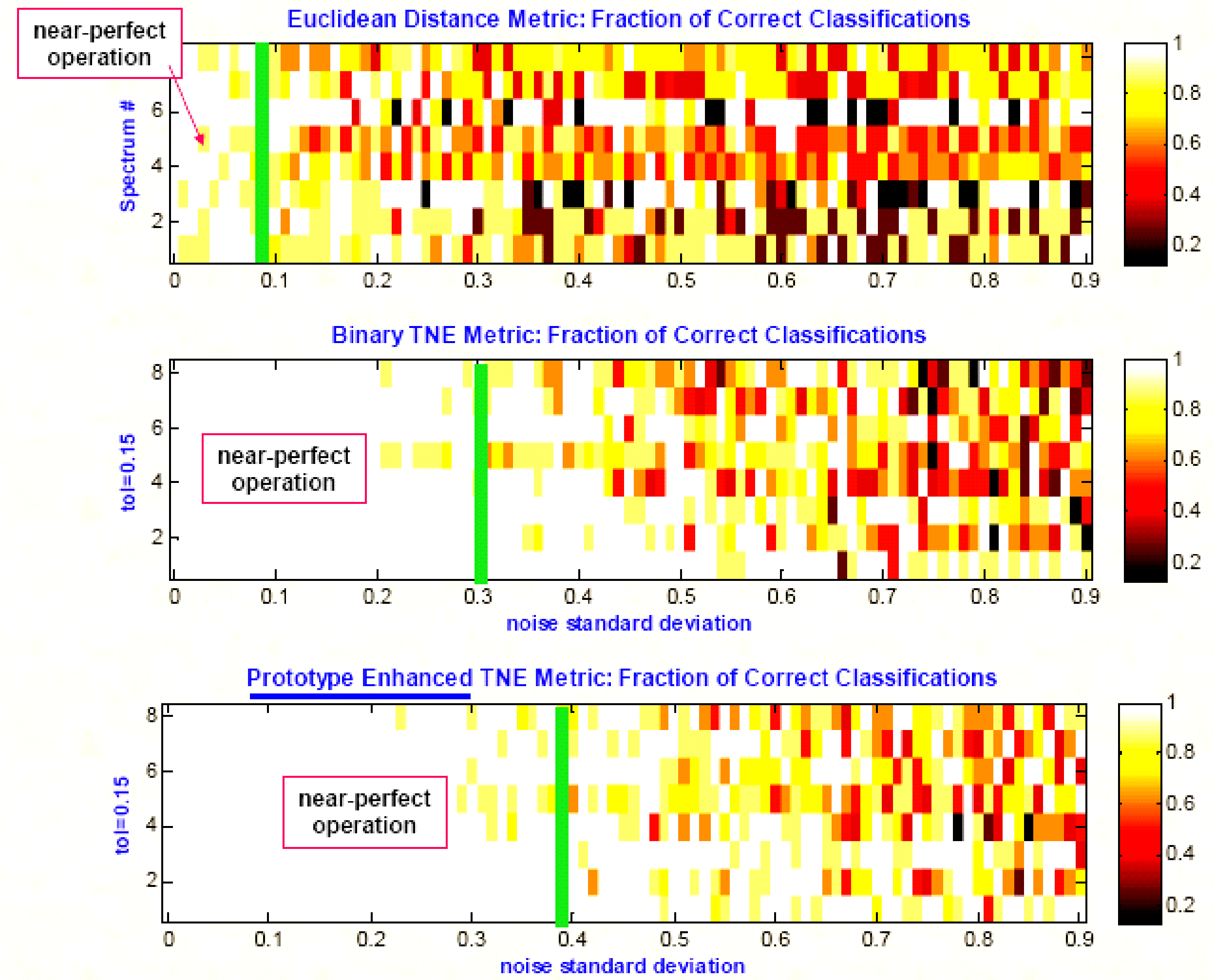


Figure 8. Performance of adaptive TNE versus Euclidean classifier, for the agreement threshold set at the optimal value of T = 0.15. The colored bars at the top represent classification score – a white pixel in the chart means perfect classification ($P_C = 1.0$) and a black pixel means totally erroneous classification. These charts have the same format and general meaning as the charts illustrated in Figures 5 and 6, with the exception that the green bar shows the upper limit of near-perfect classification.

It is interesting to compare TNE with more popular classifiers such as neural networks. TNE exhibits several advantages over classical (linear inner-product) NNs, namely, higher classification accuracy for the same number of inputs, increased information storage capacity, decreased training and classification times, and significantly reduced space requirement. Firstly, Lippman [11] stated that a classical NN with *n* inputs would be able to accurately classify no more than 0.15*n* patterns (in practice, we have found this figure to be a liberal estimate, depending on the training and test sets). However, we have shown that TNE with *n* inputs can classify with high accuracy a number of input patterns that is unrelated to *n*, which represents a distinct advantage over NNs. Secondly, an NN that stores *Q* patterns requires $\mathbf{O}(Q^2)$ space complexity, while TNE requires $\mathbf{O}(nQ)$ space, where $n << Q$ is typical and *Q* is independent of *n* as stated previously. Thirdly, classical NNs have long training times due to reverberation manifesting as multiple computational passes through the net. In contrast, TNE's training time depends only on the parameters *n* and *Q*, and training or classification requires only one pass through the algorithm given in Section 2.1.2. The TNE classification process requires $\mathbf{O}(nQ)$ I/O, Boolean logic, or integer addition operations, which can be computed in vector-parallel fashion, whereas classical NNs require $\mathbf{O}(Q^2)$ costly floating-point multiplication and division operations. Fourthly, adaptation of classical NNs require that the weight matrix be recomputed for each new pattern or group of patterns that are learned. TNE does not have this limitation, requiring only pattern replacement, with adaptation and noise tolerance built into the agreement map formation and processing as well as pattern replacement procedures.

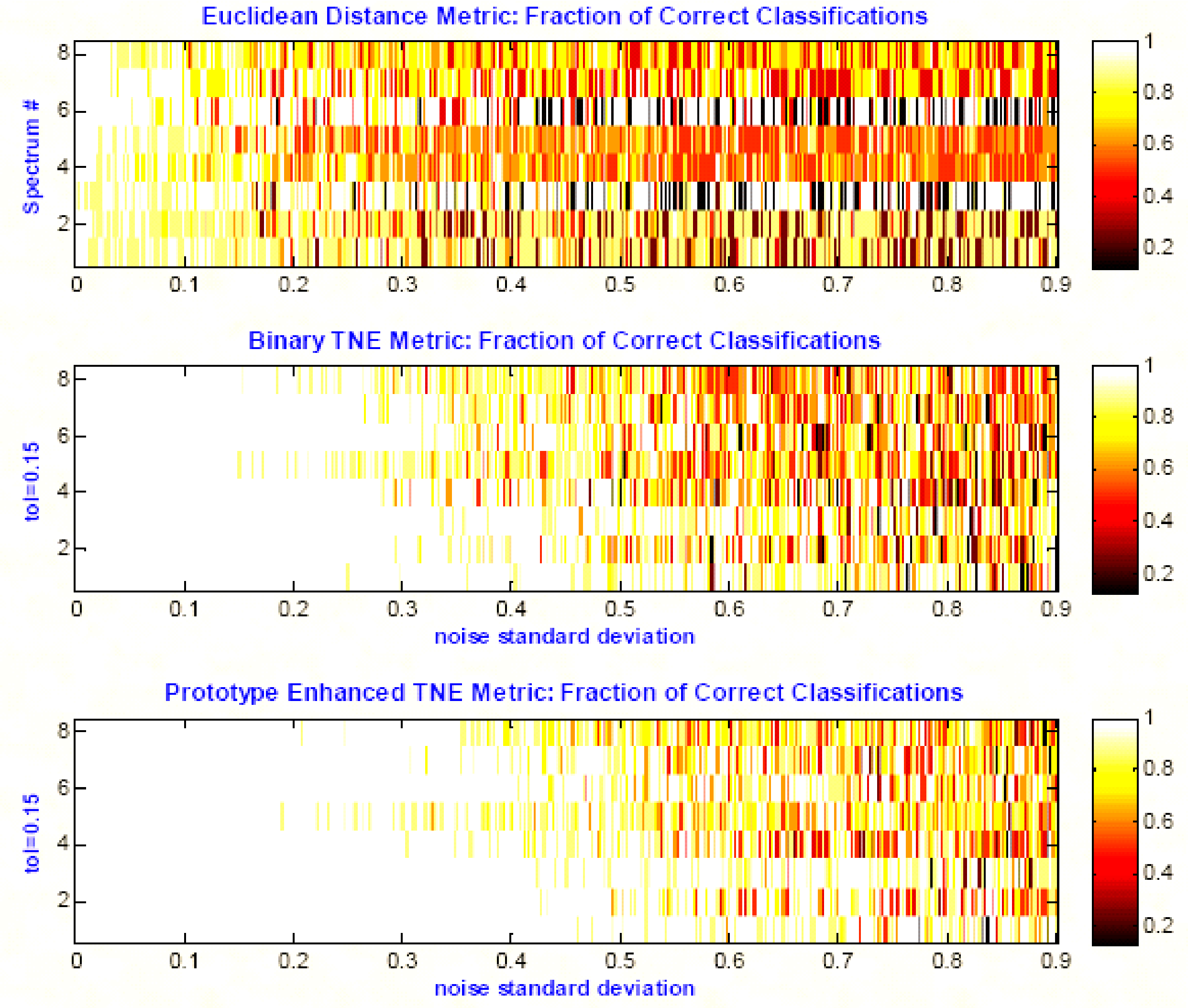


Figure 9. Performance of adaptive TNE at 100X resampling (here called enhanced TNE) versus Euclidean classifier, for the agreement threshold set at the optimal value of T = 0.15. The colored bars at the top represent classification score – a white pixel in the chart means perfect classification ($P_C = 1.0$) and a black pixel means totally erroneous classification.

4. CONCLUSIONS

Accurate non-imaging detection and recognition of space objects requires accurate classification of spectral signatures (spectral endmembers) that represent materials of which an object is comprised. In the spectral sensing process, endmembers are mixed linearly to produce a composite signature that characterizes a given object. Thus, 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 [2]. Such classifiers include metrics derived from the Euclidean distance, for example, mean-squared error.

This paper presents an emerging technology for pattern classification, called *adaptive tabular nearest-neighbor encoding* (adaptive TNE), which has proven highly successful when applied to spectral signature classification. TNE employs a highly-dimensional search space represented in terms of a relatively small 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. When the resampling factor was increased to 100X, adaptive TNE classified all spectra correctly up to the noise level $\sigma = 0.42$, which corresponds to a signal-to-noise ratio of 1.19:1, while the performance of the Euclidean distance classifier did not improve statistically. We also show that TNE performs superiorly to classical neural networks in the sense of space requirement and computational cost of training and classification, 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 adaptation techniques through closed-loop optimization of pattern replacement strategies.

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.

[10] Schmalz, M.S. and G. Key. “Hyperspectral signature classification with tabular nearest-neighbor encoding“, Proceedings of the 2007 AMOS Conference, Maui HI, 2007.

[11] Lippmann, Richard P., "An introduction to computing with neural nets", IEEE ASSP Magazine, April 1987, pp. 4-21, 1987.