Information extraction and knowledge discovery from high-dimensional and high-volume complex data sets through precision manifold learning

Erzsébet Merényi*, William H. Farrand†, Robert H. Brown‡, Thomas Villmann§, Colin Fyfe¶
*Department of Electrical and Computer Engineering, Rice University, Houston, Texas
Email: erzsebet@rice.edu
†Space Science Institute, Boulder, Colorado
Email: farrand@spacescience.org
‡Lunar and Planetary Laboratory, University of Arizona, Tucson, Arizona
Email: rhb@lpl.arizona.edu
§Institute for Psychotherapy and Psychosomatic Medicine, Computational Intelligence Group
University of Leipzig, Leipzig, Germany
Email: thomas.villmann@medizin.uni-leipzig.de
¶Applied Computational Intelligence Research Unit (ACIRU)
School of Computing, University of Paisley, Paisley, Scotland, UK
Email: colin.fyfe@paisley.ac.uk

***Abstract*—Finding critical information in large and complex data volumes is a challenge increasingly posed by real systems such as NASA's space and Earth science missions. Many of these real systems are also desired to operate highly autonomously, using extracted information directly, for decisions, or for collaborations among missions. One example of this would be spacecraft or rover navigation based on scientific findings from continuously collected data, through on-board computation. This underlines the importance of the quality of information extraction: it must be intelligent enough to produce critical details; reliable; robust; and fast. The most interesting and powerful data, collected in space and Earth science missions as well as in many other areas are characterized by high dimension, high volume, and complicated internal relationships among the variables that are recorded for the purpose of capturing the structure of the data. However, while precise extraction of the data sructure facilitates the best possible knowledge discovery, few methods exist that measure up to the complexity of such data. We focus on three of the quality aspects of information extraction listed above: intelligent data understanding, reliability, and robustness, through precision manifold learning, and point out the benefits for autonomous oprations.**

***Index Terms*—Intelligent data understanding; Data mining; On-board decision support; Manifold learning; Self-Organizing Maps; High-dimensional data.**

I. Background and motivation

Extraction of critical information from continuously collected data such as in mission scenarios, is imperative for a system's decision making and subsequent response. The most interesting and powerful data, collected in space and Earth science missions as well as in other areas such as biomedical and clinical research, security and fraud monitoring, are characterized by high dimension, high volume, and complicated internal relationships among the variables that are recorded for the purpose of capturing meaningful information that can be transformed to knowledge. Hyperspectral imagery from Earth and space science projects; combination of measurements from multiple sensors; stacked time series of genetic microarrays; and homeland scurity data bases are examples.

Many real systems needing identification of key information in large and complicated data sets are also desired to operate in a highly autonomous fashion, using extracted information and discovered, distilled knowledge directly for decision making, for collaborations among missions including ground components, or for alerts. Some example scenarios are a) unmanned spacecraft operations or rover navigation, seeking to return data of high scientific value from planetary missions; b) on-ground large remote sensing archives that must be processed for discoveries or for finding particular known phenomena; c) (near-)real time oprations of spacecraft or reconaissance vehicles in battlefields, or high-throughput medical and security screening.

The quality of the information extraction is extremely important for the full exploitation of such data for effective decision support. The need for advanced data exploitation capablities is further stressed by the interest in autonomous and/or (near-)real time operations. However, few existing methods have the power to deal with the class of data described above.

At the same time when many aspects of on-board computation for autonomous navigation, including hazard avoidance, pointing precision, high performance, reliability, fault tolerance, etc., are already a reality [1], as stated by [2], "... the most exciting mission opportunities will not be realized without on-board intelligence ...", "... robotic explorers may pass by innumerable scientifically interesting sites, but without

the requisite intelligence to recognize them as such, they are simply bypassed and never seen by planetary scientists." The MER rovers, for example, did not have on-board processing of science data with sufficient intelligent understanding to recognize a rare mineralogy or other scientifically relevant surface features, thus could not have made an autonomous decision to stop and examine it. Today's orbiters do not have this capability either, or even just the capability to alert to an interesting event and send the related data (or data product) to Earth with high priority, for preferential human evaluation and intervention. Since we already live in the era of high-performance, reliable, miniaturized and radiation hardened computing facilities, suitable for autonomous on-board operations [2], the withholding factor is primarily the lack of sufficiently intelligent and sophisicated data understanding methods with demonstrated reliability and robustness.

The interest to improve this situation is expressed by existing projects to develop automated analysis systems for scientific data, within NASA sponsored research. Recent examples include ADaM (Algorithm Development and Mining system, http://datamining.itsc.uah.edu/adam/index.html) and EVE (EnVironMent for on-board processing, http://eve.itsc.uah.edu) [3], [4] and spectral pattern recognition algorithms for mineral detection by Gilmore *et al.* [5], Gazis and Roush [6], and Ramsey *et al.* [7]. While these works make significant contributions toward on-board processing of scientific data (see a summary in [8]) they also illustrate that the difficulties of the pattern recognition tasks involved are great and can force limited applications. Systems developed to recognize one specific surface feature from a selected subset of the available data (for lack of capabilities to deal with multiple features from all available data), will not recognize other important species. Systems using conventional algorithms may not be able to extract detailed enough knowledge from complex, high-dimensional data, and may miss important events.

II. Intelligent data understanding with HyperEye

The above underline the *extreme importance of the capability to fully exploit a given data set, and the quality of the extracted information*. To enable the best possible data exploitation and knowledge generation, an autonomous data understanding subsystem (envisioned as part of a spacecraft, rover, or ground-based archival system) must have the following properties:

1) It must be intelligent enough to deliver high quality information / knowledge, characterized by
   a) high level and precision of useful detail;
   b) repeatability and reliability;
   c) self-assessment of quality, and feedback to the knowledge extraction engines to improve performance.

   This requires precise learning of the structure of the acquired, often very high-dimensional, data manifold, finding *all* (often a large number of) natural clusters including rare ones, and categorizing them into known and unknown classes. It is desirable that the system can perform both unsupervised clustering for novelty detection, and supervised classification for known classes of interest, simultaneously. For clustering, the ability of faithful delineation of all clusters, regardless of the distribution of their size, density, shape, etc., capturing of fine intricate structure in the data, is critical. For supervised classification, precise discrimination among many classes with potentially subtle differences between their feature vectors, is imperative.
2) A data understanding subsystem must also be capable of continuous learning and adaptation to new situations, since in a space exploration scenario (as well as in many others) data are acquired continuously;
3) It must be fast (real or near-real time).

These concepts are represented in HyperEye, our manifold learning environment.

A. HyperEye as a manifold learning subsystem

HyperEye is a collection of neural and other related algorithms for coordinated "precision" mining of complicated and high-dimensional data spaces, envisioned to support autonomous decision making or alerting as outlined in Figure 1. It is designed for both the discovery of all clusters, including rare or novel, surprising features in multi- and hyperspectral images, as well as for general surface cover mapping of all relevant spectral species. This focus is highly motivated since virtually every planetary mission and Earth-orbiting satellite carries spectral imagers now, in recognition of the fact that the extremely rich data imaging spectrometers provide enable discrimination among almost all surface materials. HyperEye algorithms, however, are equally applicable to many other types of "stacked vector" data, including fused disparate data.

In this paper we concentrate on the detail and quality of the extracted information, as stipulated in points 1) and 2) above.

In Figure 1, left, the HyperEye Intelligent Data Understanding (IDU) subsytem is envisioned embedded in a spacecraft or rover system, processing data acquired by sensor subsystem(s) from the environment. In this example scenario, the sensor subsystem is a hyperspectral imager, and the environment is a planetary surface. HyperEye has simultaneous unsupervised clustering and supervised classification capabilities at the heart of which are sophisticated non-standard neural learning processes, discussed in the next Section. On this level of operation, the important point is that the IDU subsystem can generate alerts from both unsupervised clustering (upon detection of novel signatures) and from supervised classification (upon finding known interesting species). How the alerts are used and handled should be defined within the embedding system (navigation control, for example).

The top level details of the HyperEye IDU subsystem are shown in Figure 1, right: the Artificial Neural Network (ANN) algorithmic core, the main types of data products, and communication of extracted knowledge, in various forms and on various levels of detail, to on-board decision making and/or

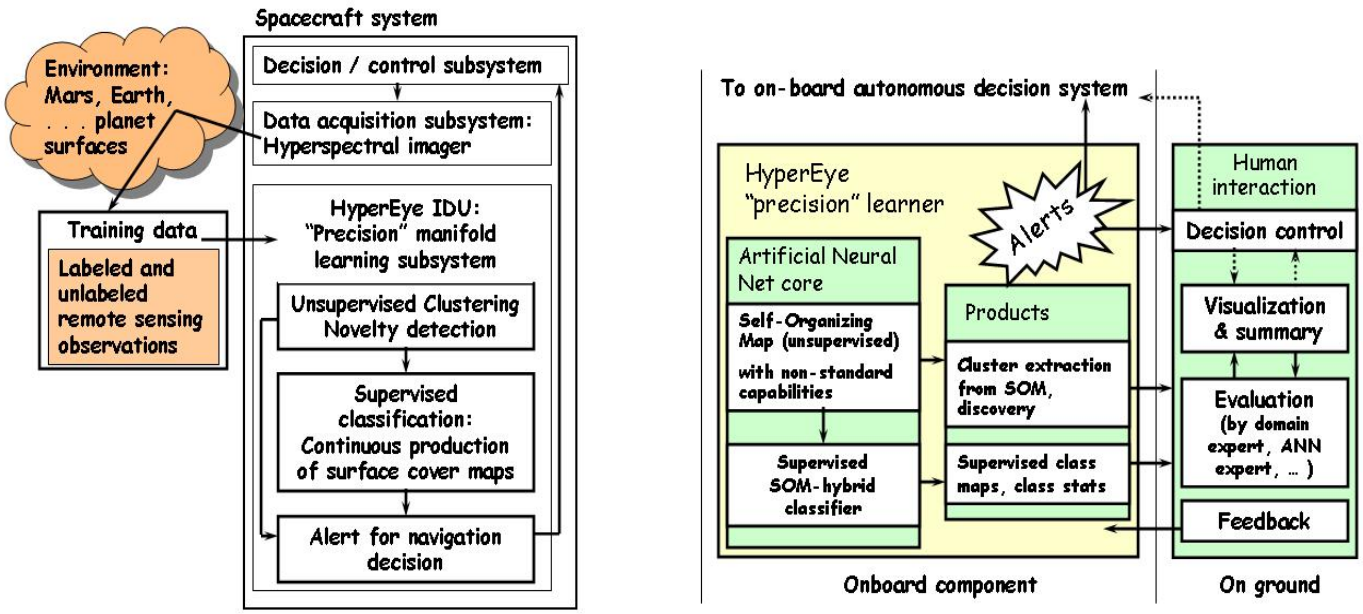

Fig. 1. **Left**: The HyperEye Inteligent Data Understanding (IUD) susbsystem embedded in a spacecraft environment, generating scientific information for decision making (in this example for navigation control) through precision manifold learning. Knowledge can be extracted using all available data, for maximum discovery potential. Both unsupervised and supervised learning and prediction can be performed simultaneously and continuously. Alerts can be generated by either modality, and passed on to the decision and control system. **Right**: The algorithmic components of HyperEye, data products, and their relationship to on-board and ground-based decision making and control, as well as to feedback and control for the learning and information extraction processes.

to humans on the ground for feedback. All acquired data can be digested for continuous unsupervised learning of the data manifold structure. This is done by Self-Organizing Maps (SOMs) and related cluster extractor modules, which have non-standard features, and which are central to the sophistication we achieve with HyperEye. These features have been developed, or adapted from recent theories and engineered to practical use, by us. Key details will be discussed in Section II.B. The learned structure of the data, seen up to the present, can be summarized and passed on to a supervised classifier, which utilizes the knowledge of the natural cluster structure of the data for its own learning of labeled data. For example, the underlying known cluster structure helps avoid learning of inconsistent labels, and also helps learning of class boundaries with greater precision than from a small amount of labeled data alone. We call this classifier an SOM-hybrid ANN because the SOM is essentially used as a hidden layer in it. Another advantage of the support by the unsupervised clustering is that the supervised classifier can be trained with a much smaller number of labeled training samples than some other supervised classifiers, including the popular and powerful Back Propagation (BP) neural network, and it is much easier to train (does not get easily trapped in local minima as do classifiers with gradient descent learning). This help from using unlabeled data is very different from the approach taken by Langrebe *et al.* (see, *e.g.,* [9]), where unlabeled data are gradually folded into the training set of the supervised classifier by labeling them according to the class predictions of the same supervised classifier. While this idea is interesting and has some (idealized) statistical justification it has not been demonstrated for high-dimensional data, nor for data containing many classes with subtle differences.

The advantage of this combination of unsupervised and supervised learning is that discoveries can be made continuously, and information can be drawn out from the SOM (in which case we are doing pure unsupervised clustering), or from the categorization (supervised) layer which is trained with labeled data "on top" of the implicit cluster knowledge passed on from the SOM. The SOM is not affected by the training of the categorization layer, thus its pristine and current knowledge of the data structure is always available for revision and augmentation of the existing class labels.

Labeled data can be provided in advance or during spacecraft operation from known libraries, or generated through on-ground human evaluation of cluster summaries returned by HyperEye. New classes can be added to the supervised classifier as deemed useful. Retraining for new classes does not need to be done from scratch, and it is a much lighter load than with a BP network. The neural classifiers in HyperEye, similarly to a BP network, learn a model of the data from training samples, which provides for more flexible predictions than a fixed rule based AI system can implement and, in general, results in a better success rate. This is especially true for high-dimensional data.

In the rest of this report we discuss some of the custom features of the SOM(s) we developed, as these are the main enablers of the sophistication of HyperEye. We give examples of data analysis capabilities, and some comparison to other methods. The referenced works by the present authors, containing more technical details, are easily accessible on-line at

http://www.ece.rice.edu/∼erzsebet/publist-Merenyi.pdf .

B. Custom features of HyperEye manifold learning

The foundation of HyperEye precision data mining is self-organized manifold learning. Self-Organizing Maps (SOMs) are intended to mimic the information processing of the cerebral cortex, where stimuli percieved from the environment are organized in a 2-dimensional surface, for very fast and very precise pattern retrieval and recognition. The basic version of the heuristic SOM algorithm, as invented by Kohonen [10], is the following. Let V denote the d_V-dimensional input data manifold, and A be the rigid, d_A-dimensional, SOM grid of Processing Elements (PEs, or neurons), where d_A is usually 1 or 2. PEs are indexed by their d_A-dimensional grid locations $\mathbf{r}$. Each PE $\mathbf{r} \in A$ has a weight vector $\mathbf{w_r}$ attached to it, which is a quantization prototype, initially with random elements. The SOM learning performs an adaptive vector quantization (VQ) by cycling through these steps many times: for any $\mathbf{v} \in V$ input a winner PE $\mathbf{s}$ is selected by

$$\mathbf{s} = \underset{\mathbf{r}}{argmin}||\mathbf{v} - \mathbf{w_r}|| \tag{1}$$

and then the weights are adapted according to

$$\Delta\mathbf{w_r} = \epsilon h_{\mathbf{rs}}(\mathbf{v} - \mathbf{w_r}) \tag{2}$$

The *neighborhood function* $h_{\mathbf{rs}}$ defines the extent to which weights are updated, as a function of the grid distance of PE $\mathbf{r}$ from $\mathbf{s}$. $h_{\mathbf{rs}}$ is often a Gaussian function centered over the winner, but the neighborhood can be defined many different ways [10]. In equation (2) the learning rate ϵ is globally defined, *i.e.,* it is the same for all PEs for a given time step and its value is independent of any local properties of the map.

This quantization differs from other VQ algorithms in two ways. It produces an optimal placement of the prototypes in data space for best approximation of the density distribution of the data. In addition, the prototypes become ordered on the SOM grid in a topology preserving fashion: prototypes that represent data points close to one another in data space will be close to each other in the SOM grid, and conversely, prototypes close in the SOM grid will represent similar data vectors. (This assumes that no topology violations occur during learning. We will briefly discuss below some research related to the recognition and remediation of topology violations in SOM learning.) This is a very powerful feature, allowing the detection of contiguous groups of similar prototypes in the SOM grid, which collectively represent clusters of similar data. Cluster boundaries can be identified based on the (dis)similarities (typically Euclidean distances) of the prototypes vectors (not the distances of their SOM grid locations!). SOM clustering does not require an initial guess of the number of clusters (unlike many clustering algorithms), nor does it require any particular initial definition of the quantization prototypes.

Many successful applications of SOMs have been reported in the last 20 years. The original Kohonen SOM (KSOM), however, was found suboptimal for high-dimensional data with complicated structures. We mention two interesting issues here.

Given an SOM with a fixed size, and K natural clusters in the data (where K is unknown prior to SOM learning), the "real estate" (the number of SOM prototypes) that can be dedicated to the representation of each data cluster is limited. In principle, if the SOM places the prototypes optimally, the *pdf* of the data should be reproduced most faithfully and all clusters (small or large) should have an areal representation proportional to their size. Theoretical analyses revealed, however, that the KSOM inherently "warps" the grid representation: instead of a linear relationship between the *pdf* of the data, P, and the distribution of the SOM prototypes in data space, Q, which is expressed by

$$Q(\mathbf{w}) = cP(\mathbf{w})^{\alpha} \tag{3}$$

where $\alpha = 1$, it realizes a functional relationship where $\alpha = 2/3$ in eq (3) [11]. The effect of this can be the loss of some clusters when the real estate is tight. For high-volume, complicated data this is always a concern, since the computational cost increases nonlinearly with the size of a 2- (or higher-)dimensional SOM grid. We use a newer SOM variant called *conscience algorithm* [12], which effects $\alpha = 1$ by a heuristics [13]. An additional benefit of the "conscience" is that one needs only to use an SOM grid neighborhood of a radius of 1 for weight updates in eq (2). This results in lighter computational load and faster learning. We also adapted a new theory to effect a *magnification* of SOM representation areas for rare events, without having to know whether rare clusters exist in a data set. This is done by forcing an $\alpha < 0$ value in eq (3), and it enhances the detectability of low-frequency data. This theory was originally proven for a rather restricted class of data. We demonstrated an extended applicability through carefully designed simulations [13]. An example of this capability is the detection of very rare materials at the Mars Pathfinder landing site, as explained in Figure 2. Full details can be found in [14] and in [15].

The Pathfinder images have 8 bands, representing a moderate dimensional input space. HyperEye can effectively handle data of much higher dimensionality. Figure 3 highlights several very small spatial clusters that were discovered from an AVIRIS hyperspectral image of an urban area, using all, ∼200 spectral channels. All extracted surface features (clusters) were verified from aerial photographs or by other inquiry. Additional details are given in [16]. This study also contrasts the power of our SOM processing with ISODATA clustering. ISODATA confuses cluster assignments in many cases where the SOM cleanly delineates homogeneous surface areas (buildings, golf course, different types of roofs, roads). The mean spectra of all the 35 clusters the SOM discovered, and of the 21 clusters ISODATA produced (shown in [16]) underline significant difference between the two methods. ISODATA not only finds a smaller number of clusters, it does not discover the clusters with the most interesting and unique signatures! This is especially noteworthy in light of the fact

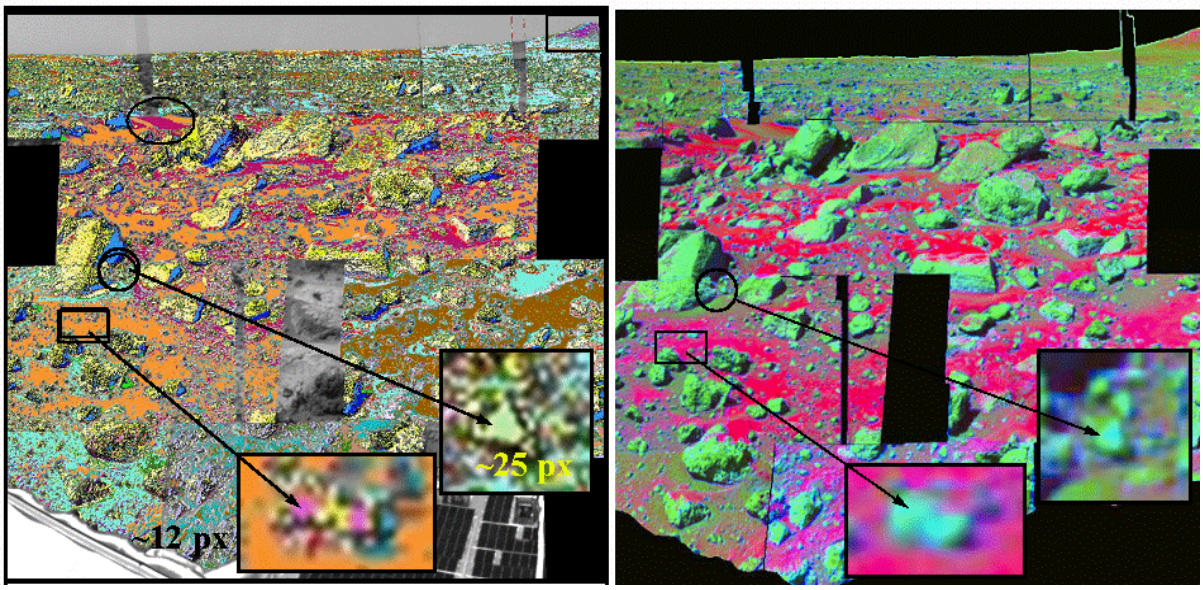

Fig. 2. **Left:** Rare surface materials on Mars mapped with HyperEye precision manifold learning from SuperPan octant S0184 (left eye) collected by the Imager for Mars Pathfinder. In this (unsupervised) cluster map, the indicated tiny areas contain a relatively pristine, undifferentiated material termed "black rock" by scientists. This material has a deep 1-μm absorption (olivine or pyroxene) and has been found in very low abundance at the Pathfinder landing site. Our clustering not only found black rock, but split it into the two subspecies shown in the insets by pale green and hot pink colors. (Please note that both of these colors are unique but to see that among 28 different colors clearly one needs to display the original cluster maps on a high-quality computer screen.) This distinction is justified by the mean spectral shapes of these subclusters (shown in [14]): one has a deeper band centered at 1 μm, the other seems to have its band center beyond 1 μm thus indicating different (undifferentiated) mineralogies. Details can be found in [14]. Note also that a large number of other surface materials have simultaneously been delineated (28 species). Such comprehensive mapping from the Mars Pathfinder data was not done before our work because of the challenges posed by the data. **Right:** Linear mixture model of the same S0184 SuperPan octant. Both black rock occurrences are outlined in the same green color, with no further distinction. The variety of surface materials is also much less pronounced that in the SOM cluster map.

that in a lower-dimensional (8-band) image of the same urban environment ISODATA produced a cluster map remarkably similar to that produced by an SOM [16]. (The ISODATA map was not quite as detailed as the SOM map but the clusters either matched or were superclusters of those in the SOM clustering, without confusion.) It is an indication of our general experience with SOMs, that the advantages of SOM-based methods over conventional ones become more pronounced with increasing data dimensionality and complexity.

Another important issue we discuss is the extraction of clusters from a learned SOM. By computing the (data space) distances between prototype vectors that are adjacent in the SOM grid and visualizing these distances over the grid cells (U-matrix representation [17]), it seems fairly straightforward to delineate cluster boundaries, and in many cases it is so. For high-dimensional data with many natural clusters, especially with widely varying cluster statistics (variable size, density, shapes) and non-linear separability, the detection of cluster boundaries becomes more complicated (*e.g.,* [18]). The representation of cluster (dis)similarities based solely on the weight (prototype) distances in data space (such as in *e.g.,* [17], [19]) is no longer sufficient for detailed and confident detection of clusters. This problem generated considerable research in recent years, partly because the challenge is intriguing from a manifold learning point of view, but just as importantly because full automation of cluster extraction from SOMs can only be done (in general, for data of high complexity) by overcoming this problem. The problem is worth the effort because the SOM, as shown in the above examples (where we used semi-automated, visualization based approaches to extract clusters) does acquire detailed and accurate knowledge about a complicated manifold, in contrast to many other clustering methods including ISODATA. Our challenge is to decipher the SOM's knowledge, and to automate the cluster extraction for autonomous applications.

The structure of a manifold, once quantization prototypes are determined and Voronoi tessellation performed with the given prototypes, can be described (on the prototype level) by the so-called Delaunay triangulation, which is a graph obtained by connecting the centroids of neighboring Voronoi cells [20]. (This underlines the importance of the optimal placement of the prototypes.) The binary Delaunay graph can thus help discover connected and unconnected parts of a manifold (*i.e.,* clusters). With simple data structures this works well. With increased data complexity and noise it becomes very important to portray how *strongly* various parts of the data space are connected. Because of its binary nature the Delaunay graph will indicate connections caused by a few outliers or by noise between otherwise well separated clusters! Some research started to target this issue recently, to represent the connectivity relations in a manifold in order to more precisely delineate clusters. These works, however, are either limited to situations where the SOM prototypes outnumber the data vectors [21], or to data spaces with low dimensions [22], [23].

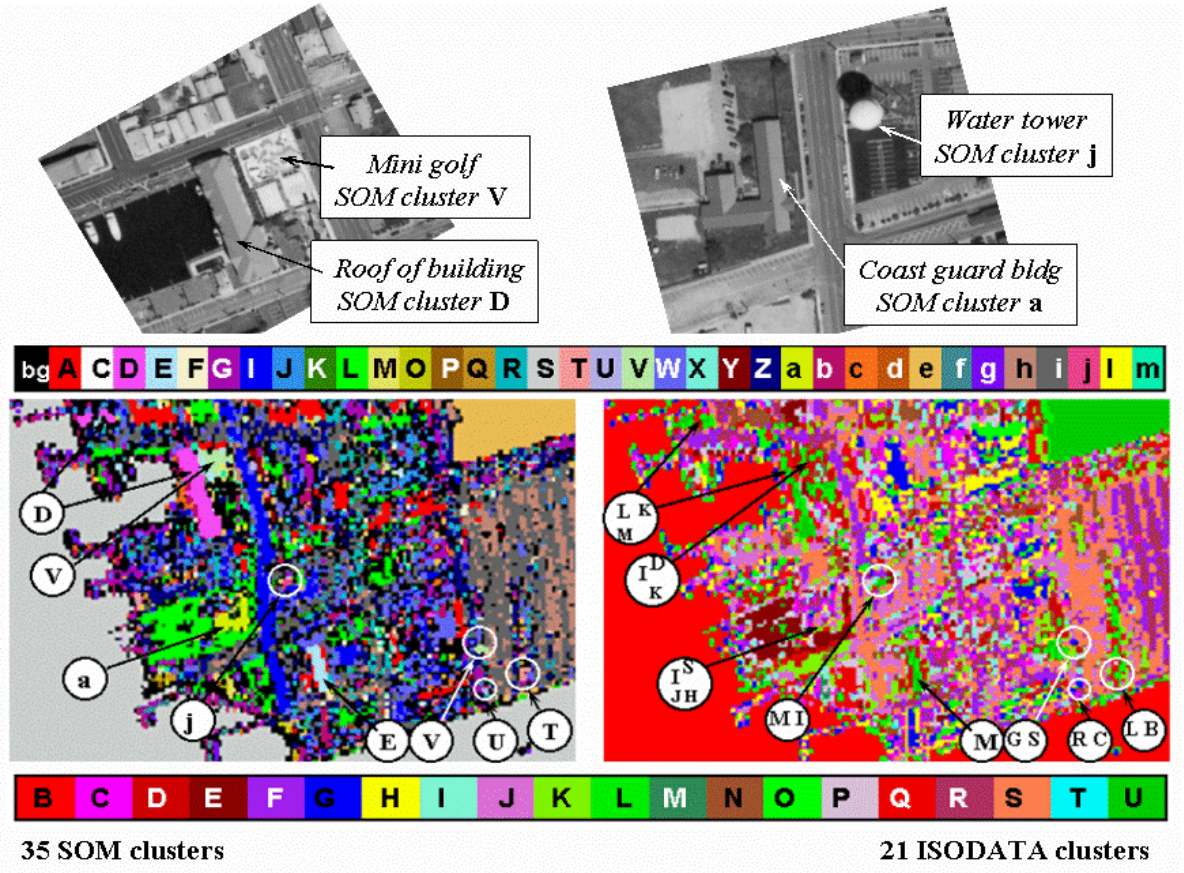

Fig. 3. Details of cluster maps, for a subsection of Ocean City, Maryland, produced from an AVIRIS image using all remaining, ∼190, bands after removal of irrecoverably corrupted bands due to atmospheric attenuation. The spatial resolution is approximately 4 m/pixel. **Left color panel:** 35 SOM clusters, with corresponding color codes at the top. The label "bg" indicates unclustered pixels. The image contains part of Ocean City, surrounded by sea water (light grey, S) with board walks extending into the water, small harbors, roads and paved plazas (dark blue and dark grey hues I, J, Z), a large open parking lot at the right (dark grey and mauve colors, i and h), beach sand (ocher, e), vegetation (green colors), and buildings with various roof materials (red (A), hot pink (D), light blue (E), yellow-green (a), T, U, and more). **Right color panel:** ISODATA clusters, with labels and color codes shown at the bottom. ISODATA leaves no pixels unclustered. Clusters and their colors are different from the SOM map. (ISODATA produced a maximum of 21 clusters even when it was allowed a considerably larger number of clusters.) In both figures, arrows point to the exact same locations. The labels in the white circles in each figure are given according to the label scheme of the respective clustering and indicate the cluster(s) assigned to the spatial feature the arrow points at in the SOM map. The full clustering can be seen in [16]. Here we make a few selected comparisons. ISODATA confuses clusters in spatial entities where the SOM assigns homogeneous labels. An example of this is the class D in the SOM map, pointing to a building (shown in the aerial photo inset at the top left), with a roof that has prominent iron oxide absorptions. ISODATA assigns this building into three different clusters, none of which have signatures with resemblance to iron oxides. (Signatures are not shown here but the full sets are displayed in [16].) Another example is a semi-U shaped building (also seen in the top right inset) with the label "a" in the SOM map, which has a very distinct spectral signature. Yet ISODATA fails to delineate it, confusing four different clusters in the footprint of this building. None of the signatures of those four clusters (I,S,J,H in the ISODATA map) has any similarity to the true spectra at this location. Finally, we point out two tiny spatial clusters: "j" (cherry color), and U (lilac) in the SOM map. "j" is a 6-pixel feature, only occurring here and at one location within the other image segment we analyzed (not shown here). In both cases, this turned out to be a water tower, as identified from aerial photographs (inset at top right). U points to a sharply delineated 3-pixel feature, with spectral signatures very different from surrounding pixels. This feature was identified as a coast guard lookout tower from a local map. ISODATA did not discover any of these, or other interesting rare spatial features in spite of their distinctive spectral signatures. These and other interesting cases are discussed in detail in [16].

Clearly, neither approach is sufficient for our goals. We are developing a novel knowledge representation that expresses the manifold connectivity strengths, for any data dimension, by showing local data densities overlain on the SOM grid, as illustrated in Figure 4 [24]. The connectivity strength between any pair of prototypes is defined as the number of times one of them is the closest matching prototype to a data point and the other is the second closest matching prototype. The use of this concept has shown advantages over existing schemes for moderate-dimensional data sets, and is under further testing and development for high-dimensional data. This knowledge representation will lend itself to automation in contrast to the "modified U-matrix" we currently use for extracting clusters semi-automatically from visualizations of prototype distances. While the latter — as the examples show — is successful in producing sophisticated details, it would be very hard to automate the human interaction involved.

We stipulated reliability and self-assessment of information extraction quality as essential characteristics of an Intelligent Data Understanding system. While it is fairly straighforward to set quality measures for supervised classifications, such mea-

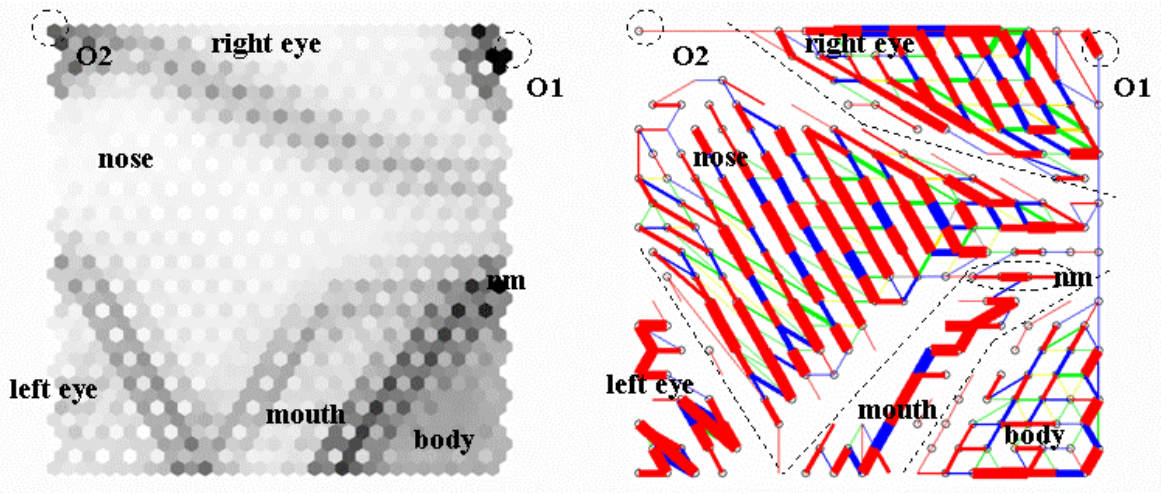

Fig. 4. **Left:** Delineation of clusters in a learned SOM grid by U-matrix representation. The SOM in this case has a hexagonal grid, and the data set learned is in the shape of a 2-dimensional "clown", artificially created to contain clusters of various sizes, shapes and densities (eyes, nose, mouth, body, with three subclusters in the left eye). Along with the hexagonal SOM of the learned prototypes it was provided to us by [18]. A U-matrix computed and overlain on this SOM outlines the main cluster structure by the high (dark) fences, as seen from the various segments annotated with the names of the respective parts of the "clown". This analysis is shown in detail in [18]. **Right:** Connectivity graph (a Delaunay graph with weighted edges instead of binary connections), explained in the text, computed from the same SOM prototypes as the ones on the left, and draped over the same SOM grid. Each line segment connecting two prototypes is drawn with a width proportional to the "connectivity strength" (local data density) between those prototypes. This representation thus provides a better resolved picture of the relative connectedness of various parts of the manifold. For example, the nose and the right eye obviously form submanifolds strongly connected inside but clearly disconnected from each other (white gaps). Dramatic improvement over the U-matrix is shown in the left eye, where the three known subclusters are easily detected. In contrast, the U-matrix representation on the right hides these substructures.

sures are harder to define, but just as important, for unsupervised clustering. To assess the goodness of a clustering without external knowledge (ground truth) is especially important in autonomous environments. The quality of a clustering can be measured, in principle, by assessing how well it matches the natural partitions of a data set. This can provide feedback for an iterative clustering method, to keep improving the clustering until the quality indicator no longer increases. For this purpose, many *cluster validity indices* have been proposed (see, *e.g.,* [25], [26] and references therein). They measure to what extent it is true that all data vectors within any cluster are closer to each other than to any data vector in any other cluster. In our experience, however, existing indices often misjudge complicated clusterings. This is caused by the metrics they use for within-cluster scatter and for between-clusters separation, which are the main components commonly combined in all validity indices. For example, the popular Davies-Bouldin index [25] employs centroid distances for separation measure, which results in favoring spherical clusters. Some indices [27] use data densities, alone or in addition to distances, to better assess clusters of various sizes and shapes. We found a number of widely accepted indices inadequate for assessing our cluster maps, and we are developing new indices designed to provide more faithful measures by taking into account the connectivity relations (as expressed by our connectivity graph) among high-dimensional clusters of widely variable statistics [26].

Cluster identification from an SOM only works well if the SOM learning was truly topology preserving: that prototypes that are neighbors in data space (centroids of adjacent Voronoi cells) end up at adjacent grid locations in the SOM. (For exact definitions of neighboring protoypes and topology preservation, see [20], [31].) Topology violating mapping can cause "twisted" SOMs, in which clusters will not be detected correctly. It is therefore important (especially for expensive runs with large data sets and in automated regime) to monitor the topology violations during learning and apply a remedy. There are many interesting aspects of such monitoring and remediation, involving the development of appropriate measures for topology violations. These are based on the expectation that in a topology preserving map a prototype is neighbor only to those prototypes (Voronoi centroids) in data space, which are in adjacent grid cells in the SOM, and vica versa. Depending on dimensionality mismatch between data space and SOM and noisiness of the data, this expectation will be violated to various degrees. For example, in a 2-dimensional rectangular SOM grid each prototype has eight grid neighbors, whereas a prototype can have more than eight neighbors in a high-dimensional or very noisy data space. The extent of a topology violation manifests in the grid distance (folding length) of two prototypes that are Voronoi neighbors, and in the strength of their connectivity. For example, in Figure 4, the prototype labeled O1 is connected to the "body" with a blue line through almost the entire SOM grid. (From the figure it is not obvious that this is not a sequence of connecting short line segments but we know it from the connectivity data.) However, this connection is caused by a single outlier data point (hence the thin line), therefore we can dismiss it as a non-important violation. A strong connection (thick line segment) with a large folding length would, in contrast, indicate an important violation. One way to examine violations (beside

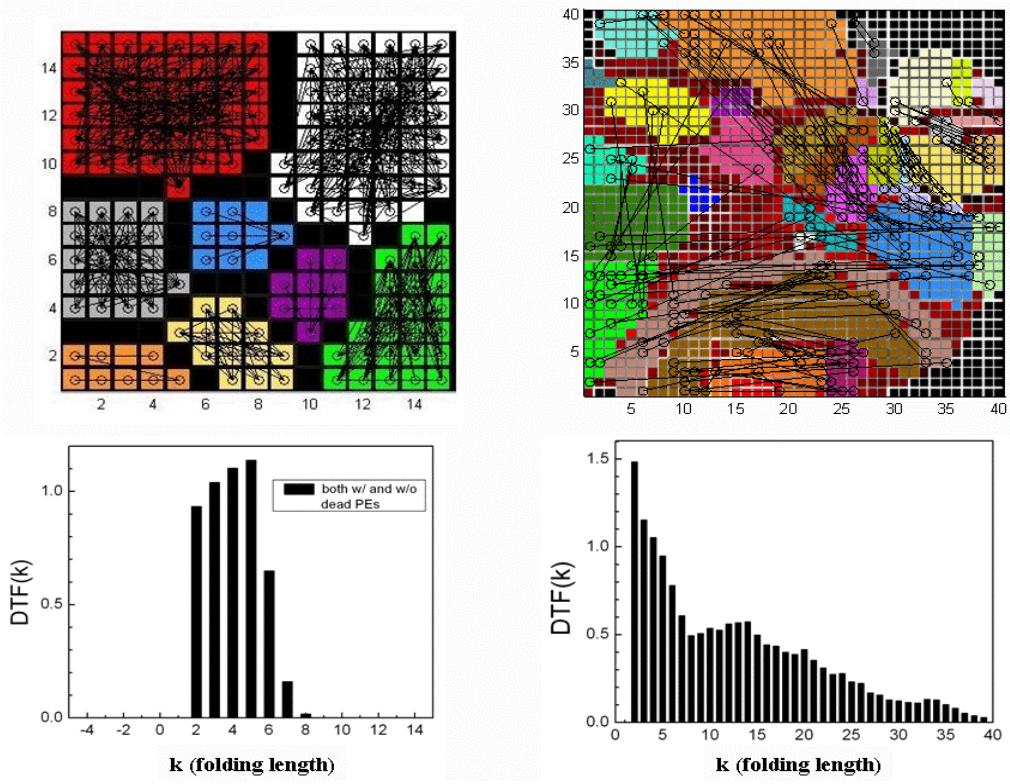

Fig. 5. Computation and representation of topology violations, for monitoring the correctness of SOM learning. **Left**: Top: Violations in an 15 x 15 SOM that learned a low-noise synthetic data set containing eight known clusters. The cluster labels known to us (but not known to the SOM) are superimposed as color coded regions. The black "gaps" contain prototypes whose Voronoi cells are empty, indicating discontinuities in the data space. Violating connections between any two prototypes are indicated by a black line segment with circles at the ends. Here, only the extent (the folding length) is shown for the violating connections, the severity of the violation (the connectivity strength) is not visualized, to avoid obscuring the structural details. Bottom: A summary measure, the Differential Topographic Function [28] reveals the distribution of violations as a function of the folding length. In this case, the longest extent of any violation is eight, which coincides with the diameter of the largest cluster. So if clusters are extracted at this stage of the learning and it is seen that all violations remain within the clusters one can conclude that *for clustering purposes* the SOM learning is sufficiently mature. Even though the violations within clusters can disappear with more learning, that will not change the cluster boundaries, and therefore computational power would be wasted. If our goal is *pdf* matching, then this level of topology violation is not acceptable, and learning should be contnued until all violations have a folding legth of one. **Right**: Top: Violating connections of strength greater than a threshold of 15 (mean plus 1 std of all connection strengths), in a 40 x 40 SOM that was learned with a 194-band AVIRIS image comprising approximately 250,000 data vectors. This data set is more noisy than the synthetic data used in the example at left. The high dimensionality and noise produce many violations, at every folding length (bottom plot). The number of violations decreases with increasing folding length, and this trend should continue with more learning. The violations with strengths exceeding the threshold, seen here, are not overwhelming considering the size of the data set, and in relation to the clusters extracted at this stage (after 300,000 learning steps). Many violating connections "profile" the onion skin structure of the clusters at the bottom part of the SOM, and some others, at the upper right clusters (pale colors), mostly extend to neighboring clusters only as these clusters are members of a slowly varying series of spectral species. In spite of obvious flaws, which indicate that more learning should be done, this clustering is already in a remarkable agreement with the known spatial distribution of materials in the input image (shown in [29], [30]

what is illustrated through Figure 4, and is further elaborated on in [24]), is the Differential Topographic Function [28], developed from the Topographic Function [31]. While we illustrate it through visualization in Figure 5, visualization is not needed for effective utilization and therefore this can be applied in automatic monitoring. For example, quantitative summaries of the inter- and within-cluster violations (shown in Figure 5) report on the state of the SOM and a temporal development of these summaries (improving or worsening trend) provide feedback on the learning. Depending on the extent, severity and history of the violations, the learning may need to be restarted with different parameters; or a change in learning parameters is necessary to accelerate a slowly improving trend; or the learning may need a new start with a larger SOM. There are other, more sophisticated (and more expensive) remediations that can be applied if warranted [32].

After these details that show how much effort goes into the precision engineering of SOMs and what capabilities result, we give an example of a many-class supervised classification from $\sim$200-band AVIRIS imagery. The geologic area is Cataract Canyon (in the Grand Canyon), where a landslide hazard study was undertaken as part of a NASA Solid Earth and Natural Hazards Program grant project (PI Victor Baker, U Arizona). The primary purpose of our classification was to map layers in canyon walls with various clay mineralogies as it had been hypothesized that different clays contribute differently to the debris-flooding potential of hill slopes. We show, in Figure 6, the resulting class map, and spectral signatures of 15 of the 28 surface cover classes that were mapped. (Readers interested in more specifics including relevant geologic details are referred

to [33].) The fine discrimination and sharp delineation of these classes, characterized by rather subtle differences in their spectral signatures, were possible because of the predetermined cluster structure by the SOM in the hidden layer of the supervised classifier.

Lastly, we want to briefly mention one important related aspect: feature extraction or dimensionality reduction. Methods that can take up the challenges we demonstrated above are scarce. Dimensionality is frequently reduced before clustering or classification to accomodate very rich data to algorithms that cannot handle high dimensionality and complexity. This, however, often results in losing discovery or discrimination potential ([29], [34], [16]). For this reason we advocate the use of full dimensionality for retention of discovery potential, and for the establishment of benchmarks for classification. However, in situations such as supervised classification, where we know excatly what we are looking for, *intelligent* feature extraction that takes into account the classification goals, can be extremely beneficial. For this purpose HyperEye has a recently developed neural *relevance learning* module, which performs non-linear feature extraction coupled with the training of a classifier, and they are jointly optimized through a feedback loop. It has shown significant performance for high-dimensional and highly structured data spaces such as hyperspetral imagery [35], [36]. We are in the process of maturing this method through further applications.

III. Discussion and Future Work

We presented a concept of on-board decision support with HyperEye as an Intelligent Data Understanding subsytem that extracts critical scientific information from data collected by scientific instruments. By communicating distilled relevant knowledge it is envisioned to contribute to science driven decisions or alerts such as needed for on-board navigation control, or automated search in large archives. In such situations the scope and the quality of the extracted information is of paramount importance. We demonstrated some of the current capabilities of HyperEye that we believe can provide smart novelty detection as well as precise detection of a wide variety of known targets of interest, from high-dimensional and complicated data.

While the core functionalities (clustering and classification) of HyperEye produce demonstratedly high quality results, there are outstanding issues to be addressed in order to minimize the need for humans in the processing loop. We discussed two important components of this envisioned autonomous IDU subsystem that are incomplete at present: the full automation of cluster extraction from a learned SOM, and the self-assessment of the quality of clustering. With the current readiness, SOM knowledge (including the prototypes and data density counts for each prototype, which is a small amount of data) would be sent to Earth (to a human operator) from time to time (or on demand), cluster boundaries extracted semi-automatically by a human analyst, and cluster statistics computed from the clustered prototypes. This allows novelty detection (since the prototypes of a cluster of data are very similar to the actual data), and decision about appropriate actions. We have an automated SOM clustering module that works well for simple cases such as shown in Figure 5, left, but its performance has to be improved for more complex data [37]. Self-assessment of clustering quality is easy to do at present in an algorithmic sense, but the judgement of available cluster validity indices is unsatisfactory. We are working on remedying this situation [26].

Interpretation and labeling of newly discovered clusters will need human interaction even when cluster extraction will be fully automated. In the long term, it would also be desirable to automate this as much as possible, since labeling can be an extremely time consuming task given the increasing amount of data and knowledge obtained from Earth and space science missions. One approach would be to create semantic models for planetary data, populate with available data (such as spectral libraries, instrument characteristics, previous analysis results) and capture their known relationships. This can help identify a material represented by a "novel" cluster, or ascertain true novelty of it. While a system like this does not exist at present, there are at least partial examples to build on.

Neural network processing is very slow with sequential computers. Implementation in massively parallel hardware (on the level of natural granularity of ANN computation), is key to the acceleration of this processing by several orders of magnitude. This is essential for on-board operations, but it is also important for processing large data sets on Earth such as terrestrial archives. It would, in addition, speed up algorithm development considerably by enabling faster turnaround and testing. High quality clustering of a hyperspectral AVIRIS image can take a couple of days on a regular Sun/Spark workstation. The same could be done under one minute with a massively parallel designated board with currently existing technology [38]. Near-future chips using newer nanotechnology will be even faster, truly enabling real-time application. This, however, is a non-trivial and expensive task for the types and sizes of neural networks we use, outside the scope and beyond the current resources of the work involving HyperEye. We expect that further advances in nanotechnology and interest in real-time neural processing can change this in the not too distant future.

In closing, we add that the methods presented here can be applied directly to similar data such as stacked time series of gene microarrays or spectral images of biological tissues. They can also be applied to other data (such as fused disparate data as in security data bases) with appropriate modifications to ingestion, summarization and housekeeping functions.

Acknowledgement

The HyperEye algorithm development and data analyses have been supported by grants NNG05GA94G and NAG5-10432 from the Applied Information Systems Research Program, and by grant NAG5-13294 from the Mars Data Analysis Program of NASA's Science Mission Directorate. This work

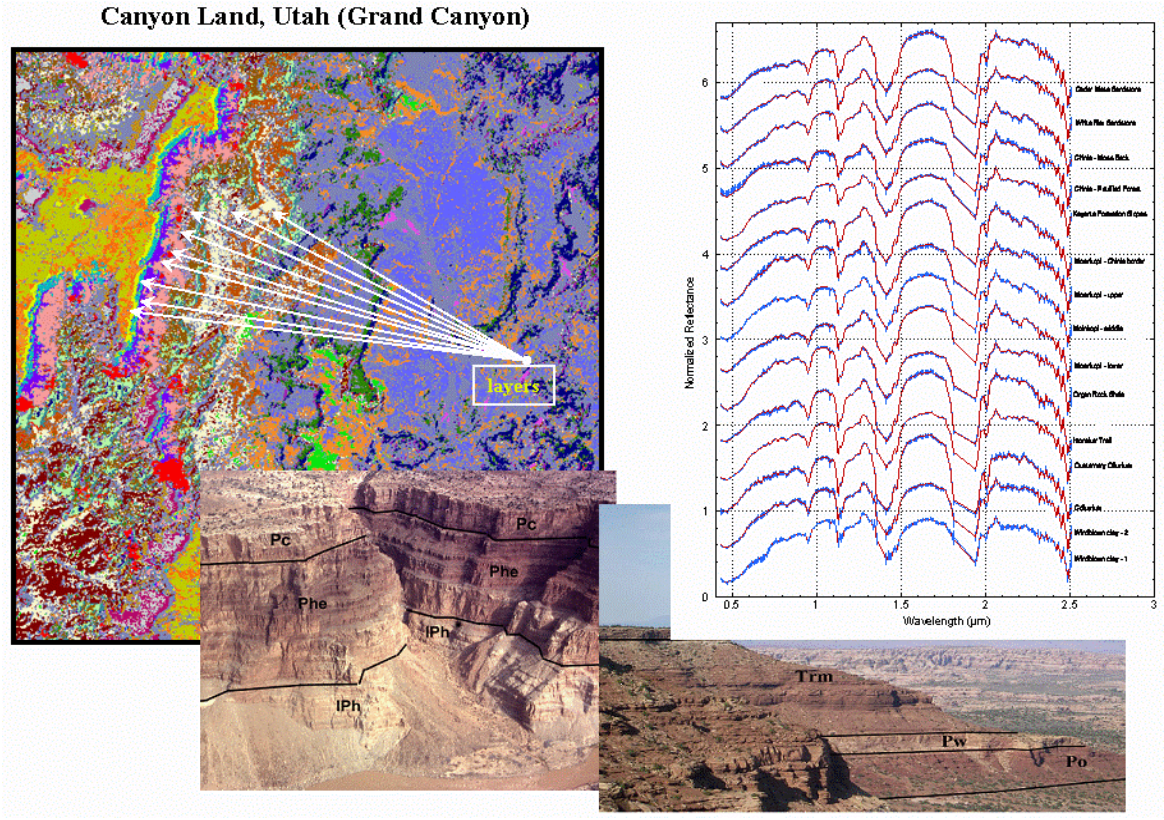

Fig. 6. **Left**: Supervised class map of 28 surface cover types in Cataract Canyon (part of Grand Canyon), Utah, from a 196-band AVIRIS hyperspectral image, using all bands remaining after removing bands with irrecoverable signals due to atmospheric attenuation. Of special interest are a series of layered geologic formations of the Grand Canyon, shown in various colors (blue, turqoise, yellow, yellow-green, orange, and others to the right of the blue classes), running down verticaly in the middle of the class map, and then continuing with displacements. **Bottom:** Some of the corresponding physical layers in the Grand Canyon. **Right**: Mean spectra of training sets (blue), and of the predicted classes (red) for 15 of the classes seen on the map at left. The graphs are vertically offset for viewing convenience. The standard deviation of the training classes are shown by vertical bars for each spectral channel. The red mean spectra of the predicted classes are, in most cases, virtually indistinguishable from the training means, indicating tight classification. These spectra represent a situation where precise discrimination of many species was needed, with subtle but meaningful differences in their signatures. Details of this geologic mapping (including the names of the layers, illegible here) are described in [33].

involves contributions by graduate students Kadim Tasdemir and Lily Zhang (Rice University) and by former graduate students Abha Jain and Major Michael Mendenhall, as well as software development by former staff member Philip Tracadas and undegraduate ECE major Allen Geer. It also reflects much appreciated stimulation and collaborations by space and Earth science collaborators. HyperEye software development utilizes Khoros [39] and NeuralWare, Inc. [40] libraries.

References

[1] E.W. Tunstel, A.M. Howard, and T. Huntsberger, "Robotics challenges for space and planetary robot systems," in *Intelligence for Space Robotics*, A.M. Howard and E.W. Tunstel, Eds., TSI Press Series, pp. 3–20. TSI Press, 2006.

[2] R. Some, "Space computing challenges and future directions," in *Intelligence for Space Robotics*, A.M. Howard and E.W. Tunstel, Eds., TSI Press Series, pp. 21–42. TSI Press, 2006.

[3] S.J. Graves, "Creating a data mining environment for geosciences," in *Proc. 34th Symposium on the Interface of Computing Science and Statistics, Montreal, Canada*, 17–20 April 2002, vol. http://www.interfacesymposia.org/interface/I02/I2002Proceedings /GravesSara/GravesSara.presentation.ppt.

[4] J. Rushing, R. Ramachandran, U. Nair, S. Graves, R. Welch, and H. Lin, "ADaM: a data mining toolkit for scientists and engineers," *Computers and Geosciences*, vol. 31, pp. 607–618, 2005.

[5] M.S. Gilmore, M.D. Merrill, R. Casta no, B. Bornstein, and J. Greenwood, "Effect of Mars analogue dust deposition on the automated detection of calcite in visible/near-infrared spectra," *Icarus*, vol. 172, pp. 641–646, 2004.

[6] P.R. Gazis and T. Roush, "Autonomous identification of carbonates using near-ir reflectance spectra during the february 1999 marsokhod field tests," *J. Geophys. Res.*, vol. 106, no. E4, pp. 7765–7773, April 25 2001.

[7] Joseph Ramsey, Paul Gazis, Ted Roush, Peter Spirtes, and Clark Glymour, "Automated remote sensing with near infrared reflectance spectra: Carbonate recognition.," *Data Min. Knowl. Discov.*, vol. 6, no. 3, pp. 277–293, 2002.

[8] E. Merényi, L. Zhang, and K Tasdemir, "Min(d)ing the small details: discovery of critical knowledge through precision manifold learning and application to on-board decision support.," in *Proc. IEEE Intl Conference on Systems of Systems Engineering (IEEE SoSE 2007)*, San Antonio, TX, April 16–18 2007, IEEE.

[9] Q. Jackson and D.A. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 12, pp. 2664–2679, December 2001.

[10] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin Heidelberg New York, 1997.

[11] H. Ritter, "Asymptotic level density for a class of vector quantization processes," *IEEE Trans. on Neural Networks*, vol. 2, pp. 173–175, 1991.

[12] D. DeSieno, "Adding a conscience to competitive learning," in *Proc. Int'l Conference on Neural Networks (ICNN), July 1988*, New

York, 1988, vol. I, pp. I–117–124.

[13] E. Merényi, A. Jain, and T. Villmann, "Explicit magnification control of self-organizing maps for "forbidden" data," *IEEE Trans. on Neural Networks*, vol. 18, no. 3, pp. 786–797, May 2007.

[14] E. Merényi, W.H. Farrand, and P. Tracadas, "Mapping surface materials on Mars from Mars Pathfinder spectral images with HYPEREYE.," in *Proc. International Conference on Information Technology (ITCC 2004)*, Las Vegas, Nevada, 2004, pp. 607–614, IEEE.

[15] E. Merényi, A. Jain, and W.H. Farrand, "Applications of SOM magnification to data mining," *WSEAS Trans. on Systems*, vol. 3, no. 5, pp. 2122–2128, 2004.

[16] E. Merényi, B. Csató, and K. Taşdemir, "Knowledge discovery in urban environments from fused multi-dimensional imagery," in *Proc. IEEE GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (URBAN 2007).*, P. Gamba and M. Crawford, Eds., Paris, France, 11–13 April 2007, IEEE Catalog number 07EX1577.

[17] A. Ultsch, "Self-organizing neural networks for visualization and classification.," in *Information and Classification — Concepts, Methods and Applications*, R. Klar O. Opitz, B. Lausen, Ed., pp. 307–313. Springer Verlag, Berlin, 1993.

[18] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 586–600, May 2000.

[19] M.A. Kraaijveld, J. Mao, and A.K. Jain, "A nonlinear projection method based on Kohonen's topology preserving maps," *IEEE Trans. on Neural Networks*, vol. 6, no. 3, pp. 548–559, 1995.

[20] Th. Martinetz and K. Schulten, "Topology representing networks," *Neural Networks*, vol. 7(3), pp. 507–522, 1994.

[21] G. Polzlbauer, A. Rauber, and M. Dittenbach, "Advanced visualization techniques for self-organizing maps with graph-based methods," in *Proc. Intl. Symp. on Neural Networks (ISSN05)*, 2005, pp. 75–80.

[22] M. Aupetit, "Visualizing the trustworthiness of a projection," in *Proc. 14th European Symposium on Artificial Neural Networks, ESANN'2006, Bruges, Belgium*, Bruges, Belgium, 26-28 April 2006, pp. 271–276.

[23] M. Aupetit and T. Catz, "High-dimensional labeled data analysis with topology representing graphs," *Neurocomputing*, vol. 63, pp. 139–169, 2005.

[24] K. Tasdemir and E. Merényi, "Data topology visualization for the Self-Organizing Map," in *Proc. 14th European Symposium on Artificial Neural Networks, ESANN'2006, Bruges, Belgium*, Bruges, Belgium, 26-28 April 2006, pp. 125–130.

[25] J.C. Bezdek and N.R. Pal, "Some new indexes of cluster validity," *IEEE Trans. System, Man and Cybernetics, Part-B*, vol. 28, no. 3, pp. 301–315, 1998.

[26] K. Tasdemir and E. Merényi, "A new cluster validity index for prototype based clustering algorithms based on inter- and intra-cluster density," in *Proc. Int'l Joint Conf. on Neural Networks (IJCNN 2007)*, Orlando, Florida, USA, August 12–17 2007.

[27] M. Halkidi and M. Vazirgiannis, "Clustering validity assessment using multi representatives," in *Proc. of SETN Conference, Thessaloniki, Greece, April*, 2002.

[28] L. Zhang and E. Merényi, "Weighted differential topographic function: A refinement of the topographic function," in *Proc. 14th European Symposium on Artificial Neural Networks (ESANN'2006)*, Brussels, Belgium, 2006, pp. 13–18, D facto publications.

[29] E. Merényi, "Precision mining of high-dimensional patterns with self-organizing maps: Interpretation of hyperspectral images.," in *Quo Vadis Computational Intelligence: New Trends and Approaches in Computational Intelligence (Studies in Fuzziness and Soft Computing, Vol 54, P. Sincak and J. Vascak Eds.)*. 2000, Physica Verlag.

[30] T. Villmann, E. Merényi, and B. Hammer, "Neural maps in remote sensing image analysis," *Neural Networks*, vol. 16, pp. 389–403, 2003.

[31] Th. Villmann, R. Der, M. Herrmann, and Th. Martinetz, "Topology Preservation in Self–Organizing Feature Maps: Exact Definition and Measurement," *IEEE Transactions on Neural Networks*, vol. 8, no. 2, pp. 256–266, 1997.

[32] H.-U. Bauer and Th. Villmann, "Growing a Hypercubical Output Space in a Self–Organizing Feature Map," *IEEE Transactions on Neural Networks*, vol. 8, no. 2, pp. 218–226, 1997.

[33] L.Rudd and E. Merényi, "Assessing debris-flow potential by using aviris imagery to map surface materials and stratigraphy in cataract canyon, Utah," in *Proc. 14th AVIRIS Earth Science and Applications Workshop*, R.O. Green, Ed., Pasadena, CA, May 24–27 2005.

[34] J. A. Benediktsson J. R. Sveinsson and et al., "Classification of very-high-dimensional data with geological applications," in *Proc. MAC Europe 91*, Lenggries, Germany, 1994, pp. 13–18.

[35] M.J. Mendenhall and E. Merényi, "Generalized relevance learning vector quantization for classification driven feature extraction from hyperspectral data," in *Proc. ASPRS 2006 Annual Conference and Technology Exhibition*, Reno, Nevada, May 5–8 2006, p. 8.

[36] M.J. Mendenhall and E. Merényi, "Relevance-based feature extraction for hyperspectral images," *IEEE Trans. on Neural Networks, under review*, 2007.

[37] K. Tasdemir and E. Merényi, "Considering topology in the clustering of Self-Organizing Maps," in *Proc. 5th Workshop on Self-Organizing Maps (WSOM 2005)*, Paris, France, September 5–8 2005, pp. 439–446.

[38] M. Porrmann, U. Witkowski, and U. Rückert, "Implementation of self-organizing feature maps in reconfigurable hardware," in *FPGA Implementations of Neural Networks*, A. Omondi and J. Rajapakse, Eds. Springer-Verlag, 2005.

[39] J. Rasure and M. Young, "An open environment for image processing software development," in *Proceedings of the SPIE/IS&T Symposium in Electronic Imaging*, Pasadena, CA, February 14 1992, vol. 1659.

[40] Inc. NeuralWare, *Neural Computing, NeuralWorks Professional II/PLUS*, 1993.