

# Linear Fusion of Image Sets for Display

Nathaniel P. Jacobson, *Student Member, IEEE*, Maya R. Gupta, *Member, IEEE* and Jeff B. Cole

## Abstract

Many remote-sensing applications produce large sets of images, such as hyperspectral images or time-indexed image sequences. We explore methods to display such image sets by linearly projecting them onto linear basis functions designed for the red, green, and blue primaries of a standard tristimulus display, for the human visual system, and for the SNR of the dataset, creating a single color image. Projecting the data onto three basis functions reduces the information but allows each datapoint to be rendered by a single color. Principal components analysis is perhaps the most commonly used linear projection method, but it is data adaptive and thus yields inconsistent visualizations that may be difficult to interpret. Instead, we focus on designing fixed basis functions based on optimizing criteria in the perceptual colorspace CIELab and the standardized device colorspace sRGB. This approach yields visualizations with rich meaning that users can readily extract. Example visualizations are shown for passive radar video and AVIRIS hyperspectral imagery. Additionally, we show how probabilistic classification information can be layered on top of the visualization to create a customized nonlinear representation of an image set.

# Linear Fusion of Image Sets for Display

## I. INTRODUCTION

**M**ULTIVARIATE images, such as hyperspectral imagery and image sets that document temporal changes, are common in remote sensing. A convenient visualization approach is to reduce the image set dimensionality to three dimensions, which can be displayed as the red, green, and blue channels of a standard tristimulus device. Human vision functions similarly: the continuous visible spectrum of light entering each point of the eye is converted into a three-component perceived color, as captured by L, M, and S cones which roughly correspond to red, green, and blue. Humans can quickly interpret color images, which can enable fast searching and comprehension of the data.

The main disadvantage to mapping multivariate vectors to colors for visualization is of course the loss of information. Every  $N \rightarrow 3$  map will be a many-to-one mapping such that some color values will represent many different  $N$ -dimensional data vectors. Human color vision suffers from the same problem; two different spectra can cause the same color sensation. However, such metameric spectra from different objects can usually be distinguished by context, shapes formed in the image, and other visual clues. The human visual system is an existence proof that dimensionality-reduction to color can be an effective way to process spatial multivariate information.

Human color vision can be modeled as a (probabilistic) projection of the incoming full spectral image onto the three spectral filters corresponding to the three cones, shown in Figure 1 (top). Each of the three cone types integrates incoming photons that are absorbed preferentially depending on the cones' sensitivity. Analogously, to reduce the dimensionality of a discrete set of multivariate images for display, it is common to project the full multivariate image set onto three basis functions. Let  $r$ ,  $g$ , and  $b$  be discrete basis functions, where  $r[n]$  denotes the  $n$ th component of the basis function  $r$ , and  $n = 1, \dots, N$ . The response in each color channel is a linear projection of the data onto each basis function so that a normalized  $N$  component data vector  $x$  is mapped to scalar color components  $R, G, B$ :

$$R = r^T x G = g^T x B = b^T x. \quad (1)$$

(Data pre-processing is further discussed in Section IV-E.)

The components  $R, G, B$  may be scaled and quantized for display. The basis functions  $r$ ,  $g$ , and  $b$  may be data-adaptive, such as the first three principal components from a principal components analysis. In this paper we focus on designing basis functions  $r, g, b$  that, like human vision, are fixed.

In previous work we proposed a basis inspired by human vision and display hardware for reducing the dimensionality of hyperspectral images for display [3]. The proposed basis synthesized what the human eye would see if its range of perceived wavelengths were stretched to cover the hyperspectral range of interest. This idea was implemented by using

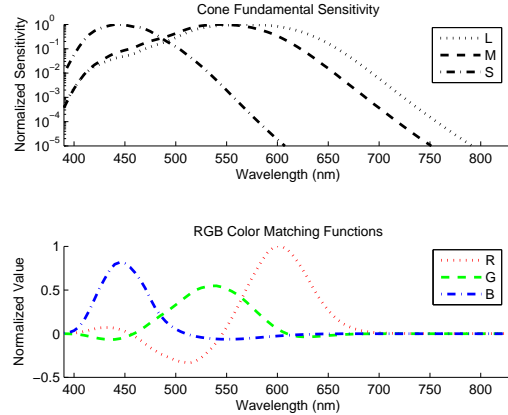


Fig. 1. **Top:** Sensitivity of cone cells in the eye. **Bottom:** CIE 1964 color matching functions (CMFs) show the required proportions of R,G,B monitor primaries needed to match pure spectral light at a given wavelength (original values from [1], converted to sRGB using transform equation from [2]).

*color matching functions.* Color matching functions specify how much of each of three primary colors (e.g. a red, a green, and a blue primary) must be mixed to create the color sensation of a monochromatic light at a particular wavelength [4]. Figure 1 (bottom) shows the color matching functions for the sRGB standard monitor red, green, and blue primaries [5]. Given hyperspectral data for the visible range, linearly projecting the visible range spectral data onto the color matching functions determines the amount of the three primary colors that, when linearly mixed, would create the same color sensation as viewing the original spectrum. By stretching the sRGB color matching functions to span the full hyperspectral data range we formed the *Stretched CMF basis*, shown in Figure 3. Applied to hyperspectral imagery, this basis creates consistent images where hue, brightness, saturation, and whitepoint all have interpretable and relevant meaning. An example AVIRIS (Airborne Visible/Infrared Imaging Spectrometer [6]) hyperspectral image rendered with the Stretched CMF basis is given in Figure 11 (top). Such visualizations avoid pre-attentive bright saturated colors, creating images with a natural palette. In Section V-D we show that by using a fixed linear basis to create a natural palette, other information can easily be added by using highlight colors.

A disadvantage of the Stretched CMF basis is that different data components are not weighted equally in the final visualization, corresponding to the fall-off of sensitivity of human vision at the edges of the visible spectrum. The effect is suitable for hyperspectral imagery systems whose signal-to-noise ratio also falls off at the ends of the spectral range. For the display of general image sets a more uniform treatment of the different data components is needed, as well as a method

to adapt a basis to the actual SNR of the data. In this paper we realize these goals.

First, related work for visualizing multivariate images is reviewed. Then, design goals for the general spatial multivariate data display problem are discussed in Section III. Based on the presented design goals, new linear bases are proposed and analyzed in Section IV. The new bases are designed to be applicable to arbitrary types of image sets, and a new method is given to effectively take into account the data signal-to-noise (SNR) ratio. Example visualizations for passive radar video and hyperspectral images are given in Section V. In Section V-D, we show how the proposed visualization can be combined with classification metadata to increase the presented information about a hyperspectral imagery.

## II. RELATED WORK

Many ideas have been proposed for multivariate visualization [7], [8]. Most approaches try to display every data component. For example, translating a multivariate data point into an icon is a popular approach; but it is difficult to render multivariate images in this manner, though some methods have been shown to be workable [9]. Healey and Enns [10] developed *pexels* to represent each multivariate pixel by a multicolored perceptual texture element. Their design goals included choosing a set of  $m$  colors to display  $m$  attributes such that any color could be detected pre-attentively, and that every color was equally easy to detect. One method to satisfy these goals was to choose five to nine colors equally-spaced along a monitor's CIE Luv gamut boundary for a fixed luminance value. In that work each attribute was mapped to a different color, and all  $m$  colors were individually displayed for each datapoint.

Projecting onto three linear basis functions creates lossy visualizations. It is common to display hyperspectral images using a lossy visualization, in particular picking three spectral bands and mapping those three images to RGB. Even when bands are highly correlated, selecting only three bands will result in less noise averaging than fusing all bands for the display. Another standard approach to displaying hyperspectral images is to calculate the principal component images and map the first three principal component images to RGB or HSV [11], [12]. Recent work forms the principal component images by maximizing the energy of wavelet coefficients in selected subbands in order to enhance edges at certain resolutions [13]; other recent work denoises the spectra using wavelets before applying PCA [14]. Using PCA for image display has a number of disadvantages (see [3] for more detail on these points): the visualization can be difficult to interpret because the colors change drastically depending on the data, color differences do not correlate strongly with data differences, the (orthogonal) principal components are usually mapped to the non-orthogonal RGB display channels, the colors may be distractingly pre-attentive, the standard saturation used in PCA display leads to simultaneous contrast problems, and the computational complexity is high. Using independent component analysis for dimensionality reduction for display has similar disadvantages [15].

A recent adaptive linear fusion technique for displaying hyperspectral imagery on RGB displays maximizes the mutual information between the original hyperspectral bands and the fused R, G, B bands [16]. Like the author's previous work [3], this maximum mutual information method uses the fact that the RGB channels are highly correlated in natural imagery and are perceived in a non-orthogonal manner. In this work, the fused image is designed to have a similar RGB covariance as natural color images [16]. This approach solves some of the problems of using PCA, but because it is an adaptive approach, the meaning of the colors must be learned for each image and the computational complexity is higher than fixed basis function approaches.

Many other techniques have been developed for image fusion, a review of multisensor image fusion for remote sensing is given in [17]. Many techniques are data-adaptive and fuse an image set into a grayscale image, which does not take advantage of human's ability to interpret color images. Many image fusion algorithms are used for a different purpose than considered in this paper, in particular, fusion is often used to estimate higher spatial resolution images by combining spectral information [18], [19], [20], [21], [22]. There is also a large literature on designing color scales that map one-dimensional data to colors that lie along a one-dimensional curve in a three-dimensional colorspace [23], [24], [25], [26]. In contrast, the focus of this paper is linearly projecting a multivariate data point onto three basis functions, so that the color projection occupies a three-dimensional gamut in sRGB colorspace.

## III. DESIGN GOALS FOR COLOR PROJECTION BASIS

In this section we consider some visualization goals for fusing a set of images to preserve information and enable interpretability.

- 1) **Summarization:** The visualization accurately summarizes the original data. Linear basis functions provide a weighted average of the data and in this sense, a summarization. To be suitable for general use, it should be possible to weight the representation arbitrarily. In most cases all components should be represented equally, but for some applications it may be desirable to emphasize components with a higher SNR or to weight recent data more heavily.
- 2) **Consistent Rendering:** The visual rendering of the data is consistent with respect to human vision such that display colors have consistent meanings.
- 3) **Computational ease:** The computation is fast enough to enable real-time usage or interactivity.
- 4) **System-optimized design:** The design is optimized for the display characteristics and the human visual system. Standard tristimulus displays can be addressed by the sRGB colorspace system [5] and interpreted in the context of human vision using a colorspace designed for perceptual uniformity, such as CIELab.
- 5) **Natural palette:** The visualization creates a natural palette of colors, producing pre-attentive colors (such as bright saturated colors) when appropriate but not creating non-informative pre-attentive distractions [27]. Too

many bright saturated colors in a visualization have been eschewed by design experts as distracting and confusing [28], [29]. In images, strong colors over large regions can induce simultaneous contrast effects which create inaccuracies in the perceived color of small color regions [30], [31], [32]. (Examples of simultaneous contrast effects can be explored on-line with the ColorBrewer visualization software [26].) As established in work using fixed basis functions for hyperspectral visualization [3], fixed basis functions can lead to a natural palette, and can be designed to avoid inappropriate bright saturated colors that form pre-attentive distractions. However, the scaling of the values for display will play a deciding role in the distribution of perceptual color properties, and if bright saturated colors are desired, they can easily be achieved by scaling the visualized colors to clip at the edges of the gamut.

- 6) **Equal-energy white point:** A data vector with the same value for each component appears gray. At the extremes, a value of zero for all components maps to black, and the maximum value for all components maps to the white point of the display.
- 7) **Equal Luma Rendering:** An equal luma rendering is one way to specify that all data components are given equal emphasis in the display. To make this notion precise, define a Kronecker data point to be an  $N$  component vector  $x[n] = \delta[n - n_0]$  where  $n$  indexes the data components  $n = 1, 2, \dots, N$ ,  $n_0$  is a fixed index  $n_0 \in \{1, 2, \dots, N\}$ , and  $\delta[n - n_0] = 0$  unless  $n = n_0$ , in which case  $\delta[n - n_0] = 1$ . If the rendered CIELab luminance of Kronecker data points is the same for all choices of  $n_0$ , we term this an *equal luma rendering*.
- 8) **Equal Chromatic Differences:** The chromatic difference between the renderings for any two Kronecker data points  $x_0 = \delta[n - n_0]$  and  $x_1 = \delta[n - n_1]$  should be the same for all equally spaced differences  $\|n_0 - n_1\|$ . This helps the viewer to correctly differentiate differences in the data based on the differences in the displayed chroma. In the proposed designs, the focus is on equal hue differences.
- 9) **Ease of Component Subset Selection:** It should be possible and consistent to ‘zoom in’ to a subset of the data dimensions and visualize some  $d < N$  data components. Or to ‘pan’ with respect to the data dimensions and view a different subset of  $d$  data components (for example, to pan over a time series and visualize  $d$  time points at once). Fixed basis functions can be stretched, compressed, or shifted to fit the number of data dimensions of interest and allow interactivity to explore dimensions of the data. The ability to pan and zoom is aided if all data dimensions are treated equally in terms of perceptual qualities of the color projection, such as luminance (Equal Luma Rendering goal) and chroma differences (Equal Chromatic Differences goal).

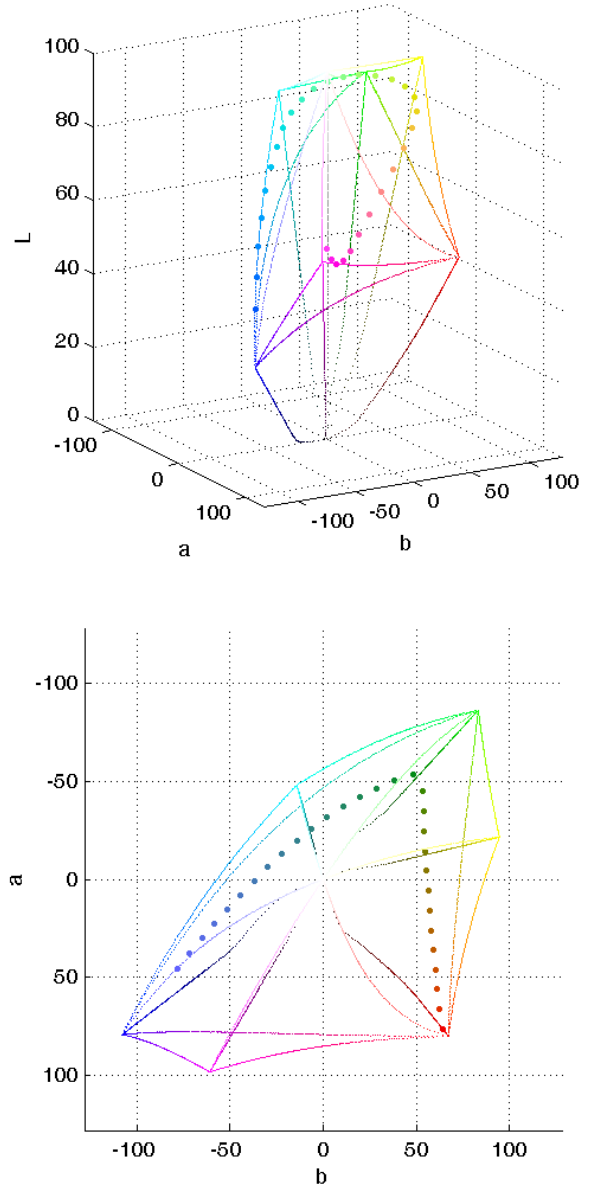
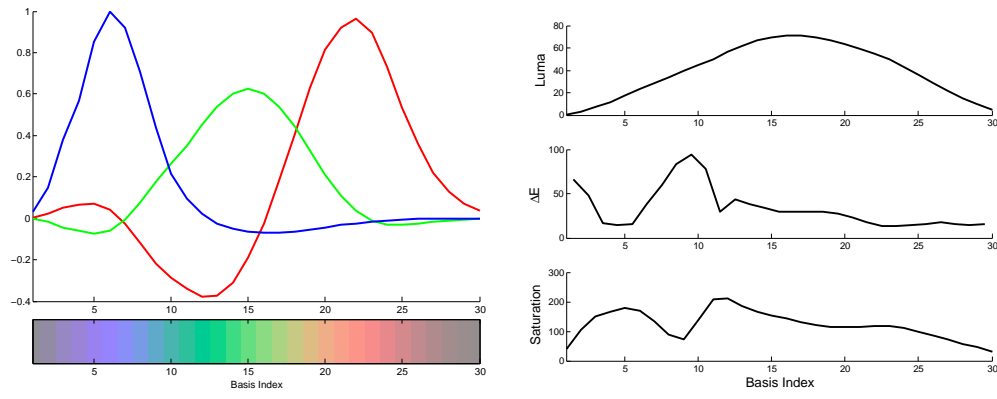
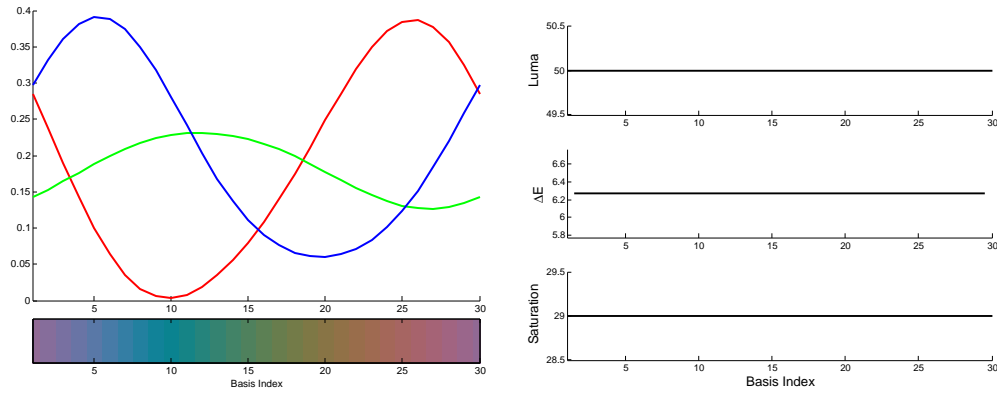


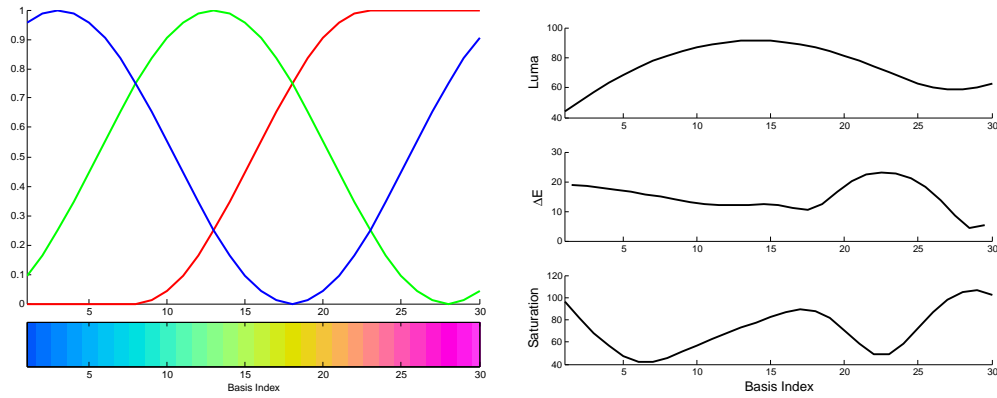
Fig. 2. **Top:** Thirty Kronecker data points, rendered without normalization using the Unwrapped Cosine basis, plotted in CIELab color space. Outlines show the sRGB gamut limits. **Bottom:** Thirty Kronecker data points, rendered with the Constant-Luma Border basis, plotted in CIELab color space (top-down view). Outlines show the sRGB gamut limits. For both figures, the blue end point is the rendered color of  $\delta[n - 1]$ , and the red end point is the rendered color of  $\delta[n - N]$ .



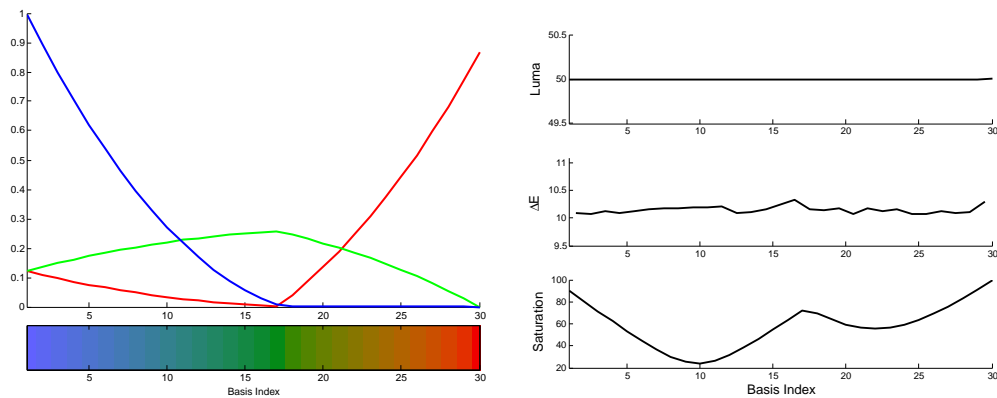
Stretched CMF basis (spectrum bar is desaturated for display)



Constant-Luma Disc basis



Unwrapped Cosine basis



Constant-Luma Border basis

Fig. 3. Several color projection bases are shown; images are designed to be viewed on a standard sRGB display.

#### IV. FIXED BASIS FUNCTION DESIGNS

First, we review the Stretched CMF basis [3], then we present three bases that were designed to achieve the goals stated in the previous section. Figure 3 shows the different bases as well as the luma and saturation for Kronecker data points for each basis. Each of the bases is constructed so that a discrete sampling with an arbitrary number of data components  $N$  can be created to project  $N$ -dimensional data onto three dimensions. Code to implement these basis function sets in Matlab is available at [idl.ee.washington.edu/projects.php](http://idl.ee.washington.edu/projects.php).

##### A. Stretched CMF

Shown at the top of Figure 3, a few aspects of the Stretched CMF basis make it suboptimal for general use. First, the middle data components are more emphasized than those at the end in terms of luminance and saturation, as shown on the right-hand side of the top of Figure 3. Second, the change of hue across the spectrum is uneven, which is best seen by the color bar on the left-hand side of Figure 3, which plots the color for Kronecker data points at each data component. Third, the Stretched CMF can lead to colors that are outside of the sRGB gamut, and thus cannot be displayed on a standard sRGB monitor, such as negative sRGB values. For applications with highly correlated data components, such as AVIRIS hyperspectral images, this is a relatively rare and minor problem. For applications with pixels that are more like Kronecker data points, such as passive radar video, this can be a major problem.

##### B. Constant-Luma Disc

The goals of equal luma rendering and equal chromatic differences can be exactly satisfied by a basis which samples a circular curve in the CIELab color space at a constant  $\Delta E$ . The parameters selected were  $L = 50$  and  $\sqrt{a^2 + b^2} = 29$ . The sum of each basis function is approximately equal, so a data vector with equal components will be displayed as almost gray using this basis.

However, practical considerations make this basis undesirable:

- **Poor System-Optimized Design:** Since all Kronecker data points result in colors which are equally saturated, only a few hues (cyans and reds) are fully saturated, while greens and purples are undersaturated at this luma value.
- **Poor Summarization:** Interpretability is decreased because the endpoint colors are nearly identical. Using less than a full circular curve to define the basis would increase interpretability, but would mean that data points with equal values for all components will no longer be rendered as grays.

##### C. Unwrapped Cosine

The shapes of the Constant-Luma Disc basis functions are roughly sinusoidal, suggesting that designing a basis in RGB space using sinusoids would retain a moderately smooth color progression, and also allow the equal-energy white point goal to be met exactly. The  $r$  sinusoid function was modified to be

a monotonically increasing curve, which makes the endpoints of the spectrum distinct without changing its sum. While the display-optimized design performance of this basis is better than was found in preliminary experiments with the piecewise linear basis function of [3], the unwrapped cosine design has strong changes in Luma across the data components, and the hue of the basis colors does not change at a constant rate over the data components. Figure 2(top) shows the CIELab colors of Kronecker data points rendered with this basis.

##### D. Constant-Luma Border

The Constant-Luma Border basis is designed for the goal of equal luma rendering, while maintaining a reasonable balance between the goal of equal chromatic differences and making the best use of the sRGB gamut. Kronecker data points are rendered in colors that follow the curve defined by linear mixtures (in sRGB space) between three pseudo-primaries that lie in the  $L = 50$  Luma plane, as shown in Figure 2 (bottom). The curve progresses from blue ( $R=0.122, G=0.122, B=1$ ) to green ( $R=0, G=0.256, B=0$ ) to red ( $R=0.866, G=0, B=0$ ). Points along the curve are sampled at approximately constant  $\Delta E$ .

The basis functions do not sum to the same value, so the equal energy white point goal is not well-met: data points with the maximum value for each component are rendered as off-white. This basis has similar strengths as the Constant-Luma Disc, but renders Kronecker data points with more saturation and has distinct endpoint colors.

##### E. Data scaling and effective gamut

In order to fit the available gamut, it is necessary to rescale the input data in a consistent way. A raw data vector  $z$  is normalized as follows to create the normalized vector  $x$  that is then linearly projected:

$$x[n] = \frac{1}{k} \left( \frac{z[n] - \widehat{\min}_z}{\widehat{\max}_z - \widehat{\min}_z} \right), \quad k = \max \left( \sum_{n=1}^N r[n], \sum_{n=1}^N g[n], \sum_{n=1}^N b[n] \right)$$

where  $z$  is the raw data and  $\widehat{\min}_z$  and  $\widehat{\max}_z$  are the smallest and largest expected values of the data.

For example, with 30-component data, a Kronecker data point will, due to normalization, render to a color that is about  $1/30$  the luminance of the display white point, too dark to easily distinguish from black. For applications where significant objects of interest are expected to have many positive components (such as hyperspectral data), the contributions from multiple Kronecker data points are integrated in the color projection, resulting in visible colors. If the data of interest is in fact Kronecker data points, then it may be desirable to scale the input data to be larger by decreasing  $k$ . Such a scaling may cause data points that have multiple positive components to be clipped in sRGB.

As an example of how the gamut of a color projection relates to the sRGB gamut, Figure 4 shows the sRGB gamut boundary in CIELab space, the rendered Kronecker data points for the Constant Luma Border basis, and the effective gamut boundary for the Constant Luma Border basis (the

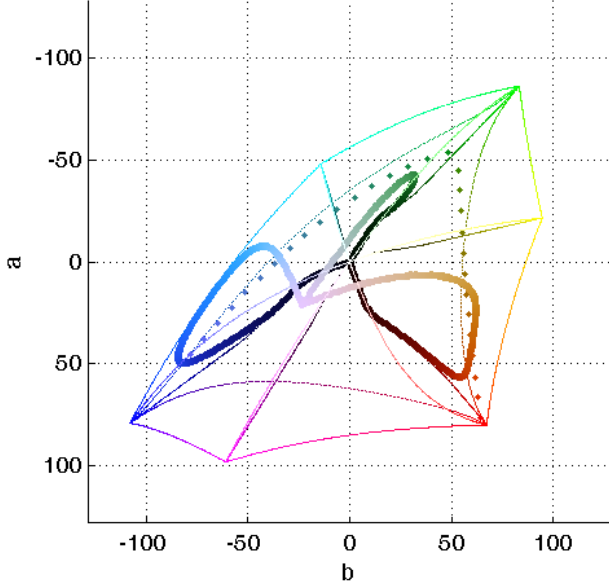


Fig. 4. In a top-down view of the CIELab color space, the figure shows the effective gamut of the Constant Luma Border basis (thick traces) and rendered colors for 30 Kronecker data points (diamonds). The sRGB gamut boundary is shown as thin traces.

three solid traces). The three solid traces were calculated by taking collections of data points that have the maximum value for a set of adjacent components, that is  $x_1 = \delta[n-1]$ ,  $x_2 = \delta[n-1] + \delta[n-2]$ ,  $x_3 = \delta[n-1] + \delta[n-2] + \delta[n-3]$ , etc. The data points were interpolated to form a continuous trace. The red trace in the figure shows the rendered colors when the set of adjacent maximum value components starts at  $\delta[n-1]$ . The blue trace in the figure shows the rendered colors when the set of adjacent maximum value components starts at  $x_1 = \delta[n-N]$ ,  $x_2 = \delta[n-N] + \delta[n-N-1]$ , etc. The green trace starts with the adjacent maximum value components as the middle components ( $x_1 = \delta[n-N/2]$ , etc.). The traces intersect when all of the components have the maximum value (rendered off-white) and when none of the components have the maximum value (rendered black).

#### F. Interpreting the color

Figures 7 and 11 are examples of image sets projected with the proposed bases. Each of the proposed bases enables the following consistent interpretation of colors to some extent:

- **Luminance:** Luminance corresponds to the sum of the data vector components.
- **Hue:** The hue of the color indicates which data components are strong. A red hue indicates that the last-ordered data components are strongest; a green hue indicates that the middle-ordered data components are strongest, and a blue hue indicates that the first-ordered data components are strongest. The colorbars in Figures 3, 7, and 8 show the rendered colors of unnormalized Kronecker data points. The colorbars act as a legend that tells the viewer

what components are strong given a particular hue in the rendered image.

- **Saturation:** A saturated color response corresponds to relatively few strong neighboring components, such as  $x = \delta[n-5] + \delta[n-6] + \delta[n-7]$ . The fewer, closer, and more intense these stronger data components are, the more saturated the color will be.
- **Whites and grays:** A white or gray patch indicates that all data components are equally strong.

In his classic work on graphics and information processing [33], Bertin divides visual information processing into three levels. The *elementary level* of processing is determining exact quantitative information such as one would read from a table. The proposed color projections do not support elementary level processing. However, the visualizations are appropriate for the *intermediate* and *overall* levels of processing: analyzing the spatially-varying qualitative relationships of the image such as identifying regions, events, and patterns in the images, and determining relationships in the image.

### V. EXTENSIONS AND APPLICATIONS

Examples are given for two sample applications: visualizing passive radar videos and hyperspectral imagery. For the hyperspectral imagery we propose a method for adjusting the basis for the SNR in Section V-B. We analyze the correlation between displayed chroma differences and the original hyperspectral angle differences for subset selection within the spectra in Section V-C and show how to effectively combine the visualization with classification metadata in Section V-D.

#### A. Passive radar videos

The Manastash Ridge Radar (run by the University of Washington Radar Remote Sensing Lab) passively images the sky by processing reflected commercial FM radio transmissions [34]. This results in a time series of images showing the range and velocity of reflecting objects. The example passive radar video used in this paper is available for viewing on the webpage [idl.ee.washington.edu/projects.php](http://idl.ee.washington.edu/projects.php). The first four frames of the video are shown in Figure 5. It is difficult to determine from watching the video how many objects there are, as the video is noisy and the signals of interest are intermittent. Also, it takes more time to view a video than a still image.

Color projection was used to collapse the sixteen time frames for one video segment into a color image, shown in Figure 7. The top image is rendered with the Constant-Luma Border basis. The bottom image is rendered with the Unwrapped Cosine basis. The strong horizontal line is due to mountain reflection of the radio signal. The strong vertical streaks are single frame artifacts caused by variations in the FM radio signal. Eight different flying objects are visible that domain experts identify as likely to be aircraft. Passive radar experts at the University of Washington judged the color projections to be very helpful and informative, in particular for the tasks of detecting and identifying flying objects.

As a comparison, the three first principal component images are shown in Figure 6, where the first principal component



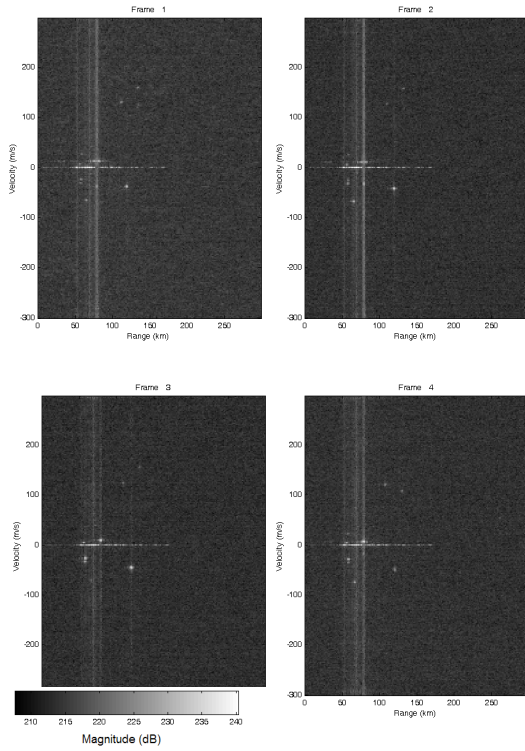


Fig. 5. First four frames from a time-indexed passive radar image set.

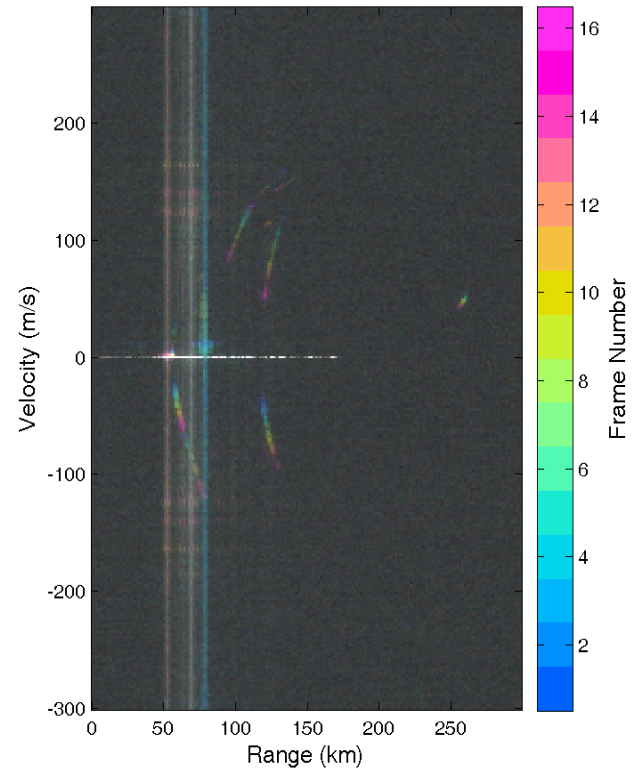
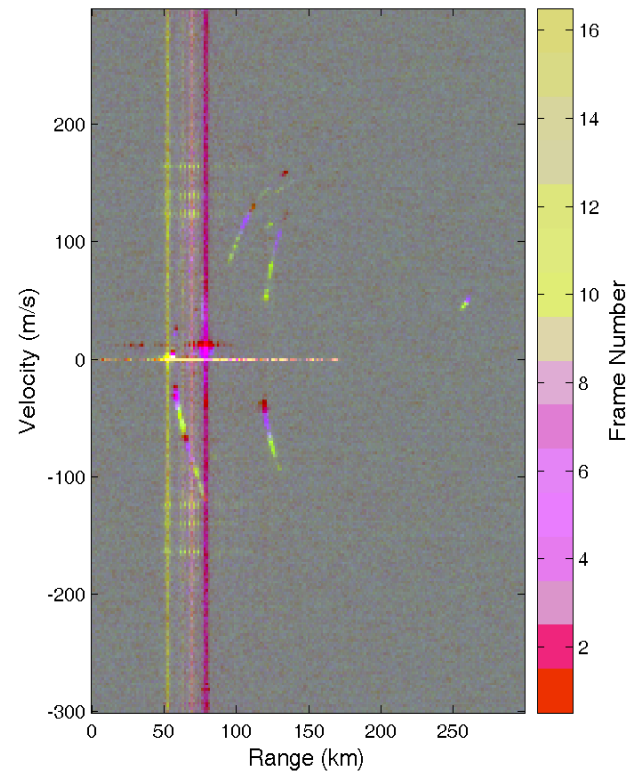
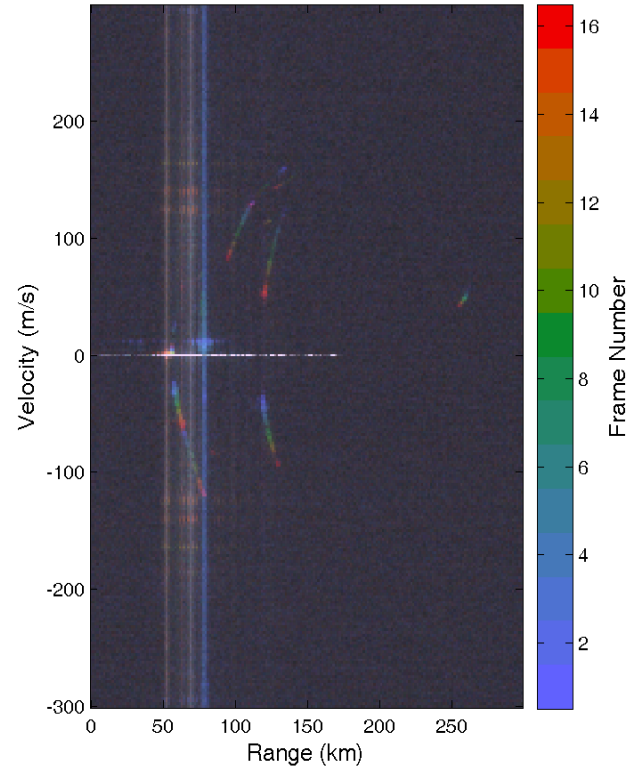


Fig. 6. Passive radar time series using the (data-adaptive) principal components basis.

Fig. 7. **Top:** Passive radar time series using Constant-Luma Border basis. **Bottom:** Passive radar time series using Unwrapped Cosine basis. Images are designed to be viewed on a standard sRGB display.



is mapped to the green display channel, the second principal component image is mapped to the red display channel, and the third principal component image is mapped to the blue display channel. The aircraft are less salient in the principal components image, and it is less easy to distinguish the different aircraft tracks.

Linear scaling of the radar range-velocity data was used for all the examples shown here in order to produce more informative composite images, although a dB scale is usually used for viewing individual passive radar frames. As a pre-processing step before projection, the expected minimum and maximum values were set to clip outliers, 0.1% of the pixels with largest and smallest magnitude.

### B. Adapting Bases to Data SNR

In some sensor systems the signal to noise ratio (SNR) varies significantly over the data components. In the case of AVIRIS the factors affecting SNR include atmospheric scattering and the sensitivity of the component spectrometers of the system. In many applications [3], [35], [36], [37] the lowest SNR bands are thrown out as they contain little usable signal.

We developed a solution to reweight a basis, adjusting the luminance and the chrominance changes to increase monotonically with SNR. We recommend starting with a basis that yields equal luminance and equal local chromatic differences, such as the Constant-Luma Border basis. Given original basis functions  $r, g, b$ , the reweighted basis functions for each component are given by:  $Ar, Ag$ , and  $Ab$ , where  $A$  is a reweighting matrix formed from the SNR values for each component.

The matrix  $A$  is designed to transfer weight from components with low SNR to components with high SNR, so that high SNR components appear brighter and have greater hue changes between them. The following constraints are used to construct  $A$ :

- The sum of the  $i$ th row of  $A$  is equal to the SNR of the  $i$ th component.
- The sum of each column of  $A$  is 1.
- All values in  $A$  are non-negative.
- All the nonzero values of  $A$  are arranged on a non-backtracking path from the top left to bottom right.

The following example shows how these constraints can be satisfied for a given SNR:

$$SNR = \begin{bmatrix} .5 \\ 2 \\ 0 \\ .25 \\ 2.25 \end{bmatrix}, \text{ then } A = \begin{bmatrix} .5 & 0 & 0 & 0 & 0 \\ .5 & 1 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .25 & 0 & 0 \\ 0 & 0 & .25 & 1 & 1 \end{bmatrix}.$$

The matrix  $A$  is built from the SNR vector row-by-row. For each row, begin at the leftmost column which does not yet sum to one, and add to it until the column sums to one or until the row sum is equal to the SNR for that component. If the SNR for that row is not exhausted but the column sums to one, then one adds to the next element in that row.

The result is an SNR-optimized basis with the same total sum for the RGB channels, but which has greater brightness

and greater chromatic differences for components with high SNR. Figure 8 shows an example of the difference between dropping low-SNR bands and using the SNR-optimized Constant-Luma Border basis. Although the SNR optimization changes the basis, as long as the SNR of the sensor system is stable the same optimized basis can be used for all data produced by the system. Thus the meaning of the colors can still be learned by a user in order to interpret the images.

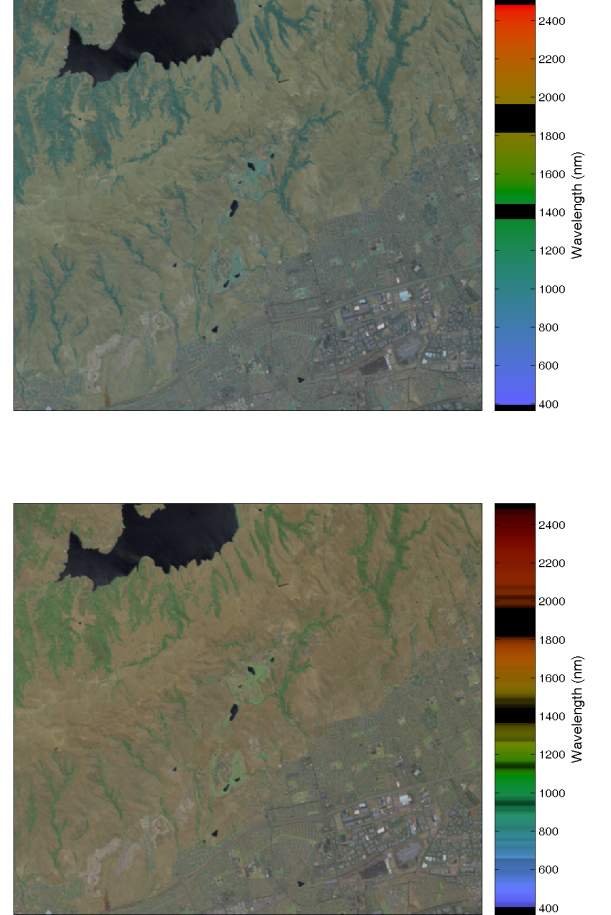


Fig. 8. **Top:** AVIRIS visualization with Constant-Luma Border basis. Bands with poor SNR are explicitly excluded. **Bottom:** AVIRIS visualization with Constant-Luma Border basis adapted to AVIRIS SNR. No bands are explicitly excluded. Images are designed to be viewed on a standard sRGB display.

### C. Projecting Subsets of Components

The proposed visualizations may also be adapted by selecting a subset of the data components to visualize, for example a subset that is expected to be most informative for a given investigation. Similar to the *window width* and *window level* [38] technique (also known as *window and level*) for viewing high dynamic range images on a low dynamic range display, using a subset of components reveals different details of the data set.

Figure 9 shows an example of the different information that is highlighted by choosing a subset of components. In

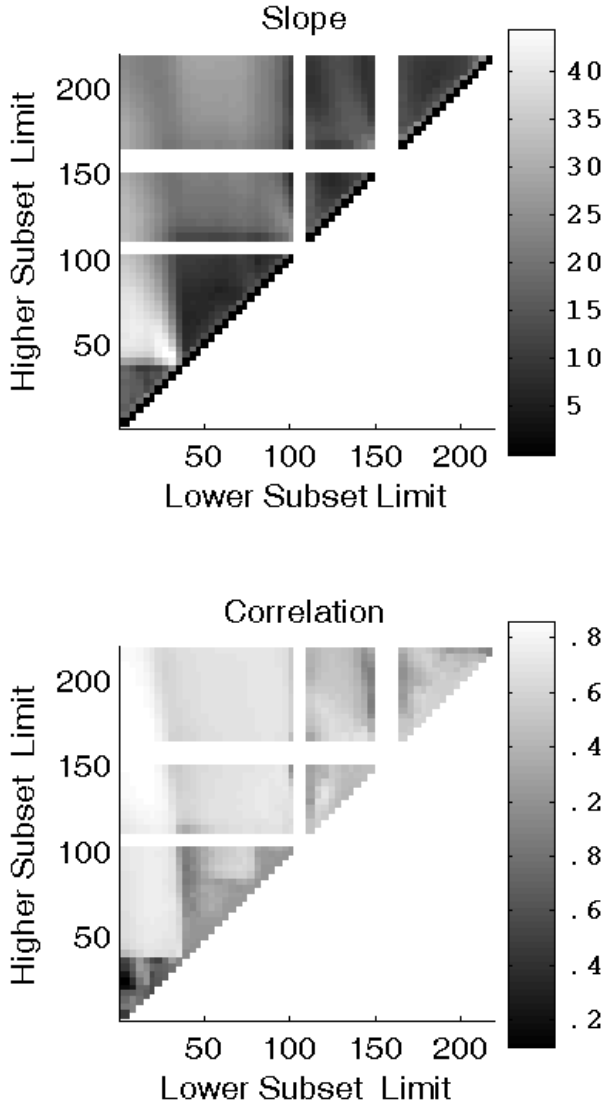
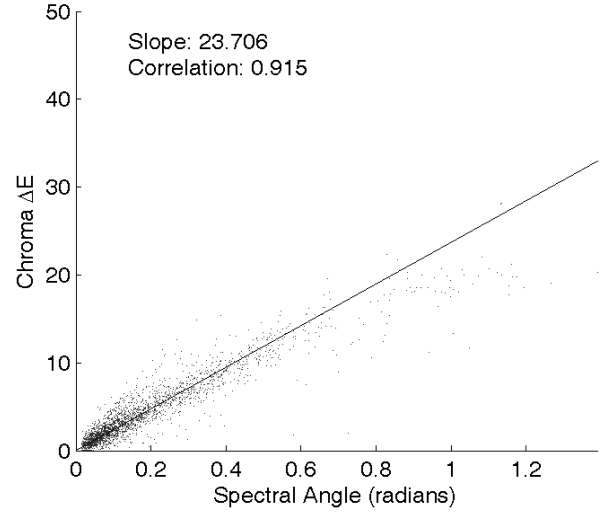


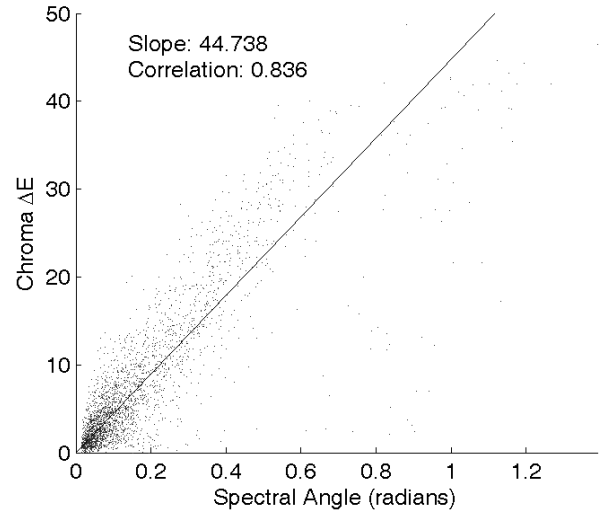
Fig. 9. Subset analysis of AVIRIS Jasper Ridge data set, projected using the Constant-Luma Border basis. Slope (**Top**) and correlation (**Bottom**) between color difference ( $\Delta E$  in the chroma plane of CIELab space) and spectral angle for different choices of low and high band.

previous work [3] we analyzed the slope and correlation of the linear regression fit between the visualization's chroma differences and the spectral angle between pairs of pixels in a hyperspectral image, as measured by the SAM [39] algorithm. The slope of the fit describes the magnitude of the chromatic differences in the visualization relative to the spectral angle differences. A large slope corresponds to a more saturated image. The correlation describes how accurately chromatic differences in the visualization represent the underlying spectral angle differences.

Figure 9 shows the slope and correlation between spectral angles (calculated using all data components) and chroma differences in color projections using different subsets of



Jasper Ridge, Constant-Luma Border, All bands



Jasper Ridge, Constant-Luma Border, bands 28-43

Fig. 10. Linear regression for two specific subset choices. **Top:** Spectral angle vs chroma  $\Delta E$  for all bands. **Bottom:** Spectral angle vs chroma  $\Delta E$  for bands 28-43.

components. The metrics show that using a subset of the components can increase the correlation and the slope between the spectral angles and the chroma differences. This can occur, for example, when many of the data components are not useful for discriminating the spectra in the image. For the example shown in Figure 10, using a reduced subset increases the slope, which is predominantly manifested as greater saturation range in the color projection (with little change in the range of hues in this case). Using a subset of the components renders differences in the original hyperspectral image more distinctly in the rendering; however the visual differences have a slightly lower correlation to the original hyperspectral differences.

#### D. Combining Color Projections with More Information

In this section a method is given for combining a color projection of a hyperspectral image with probabilistic information

about the class of material for each pixel. Suppose that one is given the probability  $p_c$  that a hyperspectral vector  $x \in \mathbb{R}^{224}$  is in material class  $c$ , where  $\sum_c p_c = 1$  and  $p_c \geq 0$  for all  $c$ . Associate an artificial color  $m_c$  with the  $c$ th material class. Then, we generate an overlay color  $m$  for the hyperspectral pixel  $x$  that is  $x$ 's expected color given the probabilistic class memberships,

$$m = \sum_c m_c p_c.$$

For each pixel  $x$ , we combine its overlay color  $m$  with the pixel's color projection color  $s$ , to create a final display color for each pixel  $f$ , where

$$f = s \left( 1 - \max_c p_c \right) + m \left( \max_c p_c \right). \quad (2)$$

That is, the overlay color  $m$  is applied to the projection image with a strength equal to the maximum class probability in the final display color. If a hyperspectral pixel has a low probability of belonging to every class, the combined color is predominantly the color projection color. Example results for this probabilistic overlay method are shown in Figure 11. For this example, small regions of the image were selected as representative for each of five material classes; these selected regions and the associated artificial colors for each class  $m_1, m_2, m_3, m_4, m_5$  are shown in the middle image of 11. Using the pixels in these selected images as training data, a neural net was trained to identify these classes. Then, the full hyperspectral image was run through the neural net to estimate the probability of each class for each hyperspectral pixel. The top image of Figure 11 is a Stretched CMF rendering of a 224 channel AVIRIS hyperspectral image without the classification information. The bottom image of Figure 11 shows the classification information overlaid on top of the rendered image, as per equation (2). Thus, the bottom image is a customized nonlinear dimensionality reduction of the original hyperspectral image to a color image, incorporating the linear image fusion of the spectral bands, and additional information about the classification in separable artificial colors.

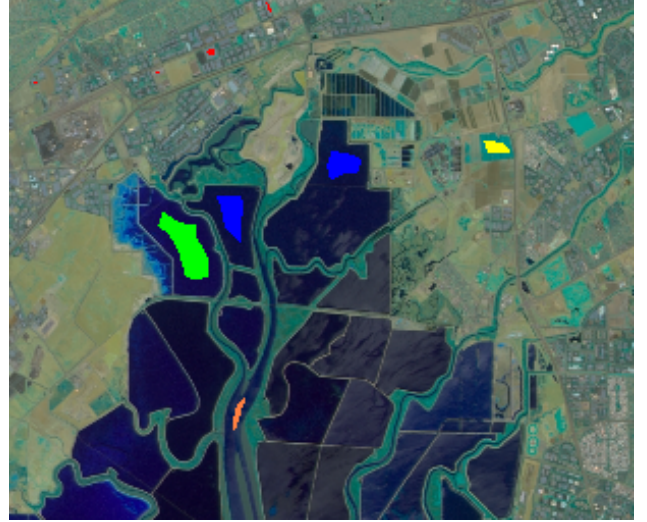
## VI. DISCUSSION

In this paper we proposed design goals and solutions for lossy visualizations of large image sets by linearly projecting the multivariate pixels onto three basis functions. The basis that best satisfies the proposed design goals is the Constant-Luma Border basis. Each data component is treated equally in terms of luma, and there are equal chromatic differences between Kronecker data points. The Constant-Luma Border functions also can create appropriate pre-attentive imagery. For example, the noisy background of the passive radar example is visualized as a textured gray, and does not draw attention. On the other hand, "spiky" events such as a reflection off an aircraft during a particular time frame yield a relatively bright saturated color. This effect is similarly delivered by the Unwrapped Cosine basis, as seen in Figure 7.

Some applications may require a specialized basis. For example, one might be more interested in recent events than



Hyperspectral Image Rendered with Stretched CMF basis



Training Regions For Each Class



Image and Overlay Showing Class Probabilities

Fig. 11. Highlighting probabilistic class membership on top of a fixed linear basis function rendering of a 224 channel hyperspectral image. Images are designed to be viewed on a standard sRGB display.



past events, then an appropriate basis would render recent events with stronger luma or saturation.

Different data pre-processing (before linear projection) may be appropriate, such as converting the data to a decibel scale or denoising. In the applications considered in this research the data components were naturally ordered: as a time series (the passive radar video) and as a spectral series (hyperspectral image). An issue not explored in this work is the ordering of data components if they are not naturally ordered. Data ordering can have a large impact on the effectiveness of any projection.

Because each multivariate data point is mapped to a single color, the technique could be used for three-dimensional data, where each voxel of the three-dimensional data has an associated multivariate data series that is then reduced to a single color.

This approach will work best when data have many highly correlated components as with hyperspectral images, or when data have relatively few components, with one or a tight cluster of strong data components (Kronecker-like data points) as with the passive radar video. Complex data sets may be difficult to visualize by this approach. The value of this kind of visualization for a particular task would need to be evaluated by a user study.

#### ACKNOWLEDGMENTS

The authors thank John Sahr, Andrew Morabito, and Melissa Meyer for passive radar data and helpful discussions.

#### REFERENCES

- [1] Colour & Vision Research Laboratories, Institute of Ophthalmology, UCL, "Colour & Vision database," <http://www.cvl.rl.ac.uk/>.
- [2] C. Poynton, "Color FAQ," <http://www.poynton.com/ColorFAQ.html>.
- [3] N. P. Jacobson and M. R. Gupta, "Design goals and solutions for the display of hyperspectral imagery," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 43, no. 11, pp. 2684–2692, November 2005.
- [4] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd ed. John Wiley & Sons, 2000.
- [5] "www.srgb.com," 2004, Hewlett Packard informational site.
- [6] R. O. G. et al., "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sensing of Environment*, vol. 65, pp. 227–248, 1998, AVIRIS signal to noise curve is given in Figure 9.
- [7] P. Wong and R. Bergeron, "30 years of multidimensional multivariate visualization," *Scientific Visualization: Overviews, Methodologies and Techniques*, pp. 3–33, 1997.
- [8] D. A. Keim, "Designing pixel-oriented visualization techniques: theory and applications," *IEEE Trans. on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 59–78, 2000.
- [9] R. F. Erbacher, D. Gonthier, and H. Levkowitz, "Color icon: a new design and a parallel implementation," *Proc. of the SPIE Conference on Visual Data and Exploration Analysis II*, February 1995.
- [10] C. Healey and J. Enns, "Large datasets at a glance: combining textures and colors in scientific visualization," *IEEE Trans. on Visualization and Computer Graphics*, vol. 5, no. 2, pp. 145–167, April-June 1999.
- [11] P. J. Ready and P. A. Wintz, "Information extraction, SNR improvement, and data compression in multispectral imagery," *IEEE Trans. on Communications*, vol. 21, no. 10, pp. 1123–1131, 1973.
- [12] J. S. Tyo, A. Konsolakis, D. I. Diersen, and R. C. Olsen, "Principal-components-based display strategy for spectral imagery," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 41, no. 3, 2003.
- [13] M. R. Gupta and N. P. Jacobson, "Wavelet principal components analysis and its application to hyperspectral images," *IEEE International Conference on Image Processing*, 2006.
- [14] S. Kaewpajit, J. L. Moigne, and T. El-Ghazawi, "Automatic reduction of hyperspectral imagery using wavelet spectral analysis," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 41, pp. 863–871, 2003.
- [15] J. Wang and C. Chang, "Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 44, no. 6, pp. 1586–1600, June 2006.
- [16] V. Tsagaris and V. Anastassopoulos, "Multispectral image fusion for improved rgb representation based on perceptual attributes," *Intl. Journal of Remote Sensing*, vol. 26, no. 15, pp. 3241–3254, August 2005.
- [17] C. Pohl and J. L. van Genderen, "Multisensor image fusion in remote sensing: concepts, methods and applications," *Int. J. Remote Sensing*, vol. 19, no. 5, pp. 823–854, 1998.
- [18] M. Choi, "A new intensity-hue-saturation fusion approach to image fusion with a tradeoff parameter," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 44, no. 6, pp. 1672–1682, June 2006.
- [19] X. Otazu, M. Gonzalez-Audana, O. Fors, and J. Nez, "Introduction of sensor spectral response into image fusion methods application to wavelet-based methods," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 43, no. 10, pp. 2376–2385, 2005.
- [20] Z. Wang, D. Ziou, C. Armenakis, D. Li, and Q. Li, "A comparative analysis of image fusion methods," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 43, no. 6, pp. 1391–1402, June 2005.
- [21] T. A. Wilson, S. K. Rogers, and M. Kabrisky, "Perceptual-based image fusion for hyperspectral data," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 35, no. 4, pp. 1007–1017, July 1997.
- [22] T. Ranchin, B. Aiazzi, L. Alparone, S. Baronti, and L. Wald, "Image fusion: the ARSIS concept and some successful implementation schemes," *ISPRS Journal of Photogrammetry & Remote Sensing*, 2003.
- [23] C. Ware, "Color sequences for univariate maps: Theory, experiments and principles," *IEEE Computer Graphics and Applications*, vol. 8, pp. 41–49, 1988, a spectrum scale is recommended to reduce simultaneous contrast effects.
- [24] L. D. Bergman, B. E. Rogowitz, and L. A. Treinish, "A rule-based tool for assisting colormap selections," *Proc. of the IEEE Visualization conference*, pp. 118–125, October 1995.
- [25] H. Levkowitz, "Perceptual steps along color scales," *International Journal of Imaging Systems and Technology*, vol. 7, pp. 97–101, 1996.
- [26] M. Harrower and C. A. Brewer, "www.colorbrewer.org," 2006.
- [27] C. Healey, K. S. Booth, and J. Enns, "Visualizing real-time multivariate data using preattentive processing," *ACM Transactions on Modeling and Computer Simulation*, vol. 5, no. 3, pp. 190–221, July 1995.
- [28] E. R. Tufte, *Envisioning Information*. Connecticut: Graphics Press, 1990.
- [29] E. Imhof, *Kartographische Geländedarstellung*. Berlin: De Gruyter, 1965.
- [30] G. M. Johnson and M. D. Fairchild, *Digital Color Imaging*. CRC Press, 2003, ch. 2, pp. 141–148.
- [31] M. Fairchild, *Color Appearance Models*. Reading, Massachusetts: Addison Wesley Inc., 1998.
- [32] M. Stone, *A Field Guide to Digital Color*. Massachusetts: AK Peters Ltd., 2003.
- [33] J. Bertin, *Graphics and graphic information processing*. Walter de Gruyter & Co., 1981, Translated by William J. Berg and Paul Scott.
- [34] A. Morabito, M. Meyer, and J. Sahr, "Improved computational performance for distributed passive radar processing through channelised data," *IEE Proc. on Radar, Sonar and Navigation*, vol. 152, no. 3, pp. 179–184, 2005. [Online]. Available: <http://link.aip.org/link/?IRS/152/179/1>
- [35] F. A. Kruse, J. W. Boardman, and J. H. Huntington, "The 1995 geology AVIRIS group shoot," in *Proceedings, 6th JPL Airborne Earth Science Workshop*, vol. 1. Jet Propulsion Laboratory, 1996, pp. 155–166.
- [36] J. P. Hoffbeck and D. A. Landgrebe, "Classification of high dimensional multispectral image data," in *Fourth Annual JPL Airborne Geoscience Workshop*, October 1993.
- [37] L. Li, S. L. Ustin, and M. Lay, "Application of AVIRIS data in detection of oil-induced vegetation stress and cover change at Jornada, New Mexico," *Remote Sensing of Environment*, vol. 94, pp. 1–16, January 2005.
- [38] J. E. Barnes, "Characteristics and control of contrast in CT," *RadioGraphics*, vol. 12, pp. 825–837, 1992, explains window and level, but uses terms "window level" and "window width". [Online]. Available: <http://radiographics.rsna.org/cgi/content/abstract/12/4/825>
- [39] F. A. Kruse, A. B. Lefkoff, J. W. Boardman, K. B. Heidebrecht, A. T. Shapiro, P. J. Barloon, and A. F. H. Goetz, "The spectral image processing system (SIPS)—interactive visualization and analysis of imaging spectrometer data," *Remote Sensing of Environment*, vol. Volume 44, pp. 145–163, May-June 1993.