Available online at www.sciencedirect.com
SCIENCE DIRECT®
Infrared Physics & Technology 45 (2004) 191–200
INFRARED PHYSICS & TECHNOLOGY
www.elsevier.com/locate/infrared

Segmentation of multi-dimensional infrared imagery from histograms

Jerry Silverman [a,*], Stanley R. Rotman [b,c], Charlene E. Caefer [c]

[a] Solid State Scientific Corp., 27-2 Wright Road, Hollis, NH 03049, USA
[b] Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, P.O. Box 653, Beer-Sheva, Israel
[c] Air Force Research Laboratory, AFRL/SNHI, 80 Scott Road, Hanscom Air Force Base, MA 01731, USA

Received 11 May 2003

Abstract

We present a technique for segmenting multi-dimensional data cubes based on multi-dimensional histograms. The histograms are formed from single gray-scale image reductions of the data cube such as principal component images. A segmentation is effected by associating each pixel with one of the peaks in the histogram. No spatial constraints are imposed and no training pixels are required. The following refinements to this simple process are described: proper weighting of the different principal components as a function of the peak shape; and automatic methods based on an entropy measure to generate a reasonable segmentation at a specified number of levels. Examples from both visible and infrared hyperspectral data will be shown.

Keywords: Hyperspectral; Segmentation; Histograms; Principal Components; Entropy; Gaussian model

1. Introduction

Segmentation of a multi-dimensional data cube to a digitized single image is useful in several ways such as: operator display, as a preliminary classification of the scene, or as the first step in an object cueing application [1]. We describe here a simple technique for segmenting imagery from data cubes such as multi-spectral or hyperspectral imagery; no training pixels are required and the only operator intervention is to specify the number of desired segmentation levels.

Single gray-scale images have been segmented in several ways. One approach is to isolate edge points from interior points in the image on the basis of strong gradients in the brightness values. Connecting the edge pixels will then determine the image segments [2]. A second approach is to segment images by grouping together contiguous pixels with similar brightness [3]. If we relax the demand for physical connectivity between pixels of the same segment, we arrive at a third approach, histogram-based segmentation [4]: the image is segmented through an analysis of the peaks and valleys present in the histogram of the image. In the context of scene or material classification, the

* Corresponding author. Tel.: +1-781-377-3295; fax: +1-781-377-4814.
E-mail address: jerry.silverman@hanscom.af.mil (J. Silverman).

1350-4495/$ - see front matter
doi:10.1016/j.infrared.2003.08.004

third approach is the most useful as disjoint regions may physically represent the same material and should be assigned the same segmentation level.

In this paper, we present a novel way to extend histogram segmentation to multi-dimensional images. We first compress the multi-dimensional information (typically spectral profiles) into single image reductions of the data cube such as principal component images. A multi-dimensional histogram is then constructed; an analysis of the statistical distribution of the points allows us to segment the image based on the histogram extrema.

The paper is organized as follows: Section 2 details the basic segmentation method and gives examples in the VNIR, MWIR and SWIR; Section 3 describes the Gaussian model used to characterize the histogram peaks; Section 4 refines the mapping into integer bins required to generate the histograms; Section 5 presents discussion, conclusions and future directions.

2. Histogram-based segmentation

Classification, which is based on spectral clustering, and segmentation of multi-dimensional data are closely linked because of the spectral/spatial correlations of natural scenes. As mentioned above, one standard technique in segmentation from a single gray-scale image is through the extrema of the image histograms. The present segmentation method combines and extends these concepts by forming multi-dimensional histograms from "divergence images".

By divergence images, we refer to single gray-scale image reductions of a data cube where the closer the spectral similarity of two pixels, the closer will be their final gray-scale value. We have explored three types of divergence images:

1. By forming an image from the values of some spectral distance measure of every pixel from a chosen characteristic pixel. Various divergence images are generated depending on the choice of the characteristic pixel and the spectral distance measure.
2. By forming an image of the spectral distance measure of each pixel from a chosen eigenchroma from principal component (PC) analysis [5].
3. By using the principal component images (eigenimages) from the PC analysis as the divergence images.

For the present paper, we will use only divergence images of type 3 since this involves no user choice of distance measure and generates good segmentations. We describe the segmentation technique by a 2D histogram example, easily extended to additional dimensions. Our example is from a ground-based image of a tent taken with a VNIR chromotomographic hyperspectral imaging sensor (CTHIS) [6]. The VNIR CTHIS is a staring-mode, framing camera system, which encodes both spatial and spectral information on every frame of data. Data cubes ($256\times256\times74$ bands) are computed using an approach similar to the limited-angle tomography techniques used in medicine. Fig. 1(a) and (b) shows the first two PC eigenimages. By mapping or scaling the floating point PC values into integer bins (discussed in more detail later), we form a 2D histogram from the dual values of each pixel in the two components (Fig. 1(c)). We next locate peaks in this histogram, taken as maxima of at least value "2" in a 3 by 3 neighborhood (Fig. 1(d)); peaks are then weeded to remove duplicates within two pixels of each other (result not shown). These N weeded maxima are lexigraphically labeled from 1 to N. The histogram space is then divided into N regions (Fig. 1(e)) by computing the closest peak to each point in the histogram space using a Euclidean measure and labeling that point with the digital label of the closest peak. Based on such an N-region template, each pixel in the data cube is then assigned the digital label of its corresponding point in the 2D histogram/template and segmentation to N levels results (Fig. 1(f)).

The next figure shows a similarly obtained segmentation of an MWIR image taken with another locally-designed chromotomographic imaging spectrometer [7]. The optical implementation is basically the same for all of the various CTHIS systems. The CTHIS projections are obtained by

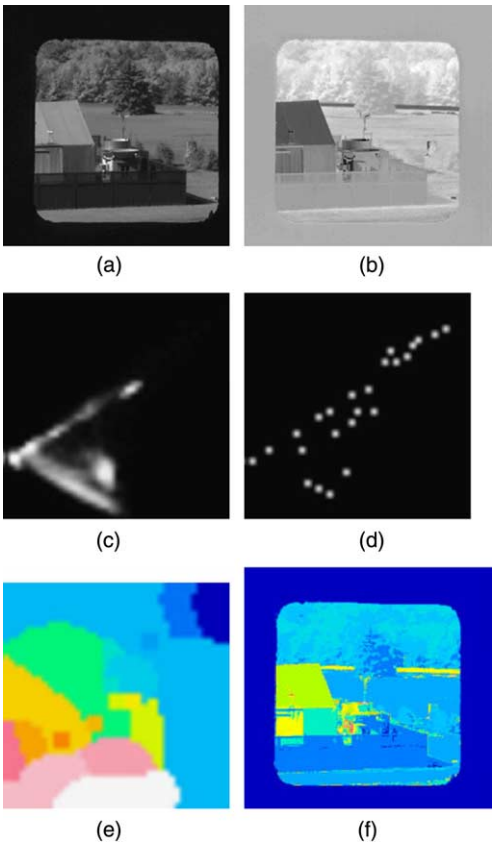

Fig. 1. (a) First principal component eigenimage of visible data cube, (b) second principal eigenimage, (c) two-dimensional histogram: brightness indicates number of pixels contributing, (d) peaks of two-dimensional histogram, (e) template and (f) 14-level segmented image.

imaging while the direct view prism is rotated on the optical axis of the telescope. The projections are then used to reconstruct a $256\times256\times74$ band hyperspectral data cube. Fig. 2(a) and (b) are the first two eigenimages, while Fig. 2(c) and (d) shows segmentations to 9 and 14 levels. Note that the number of final segmentation levels tends to increase, up to a point, with the number of chosen integer bins so that gradations from coarse to finer segmentations result (see Section 5). In our software implementations, the user specifies a desired number of segmentation levels and the starting number of integer bins (scaling factor) is adjusted downward until this desired number (or slightly less) is achieved.

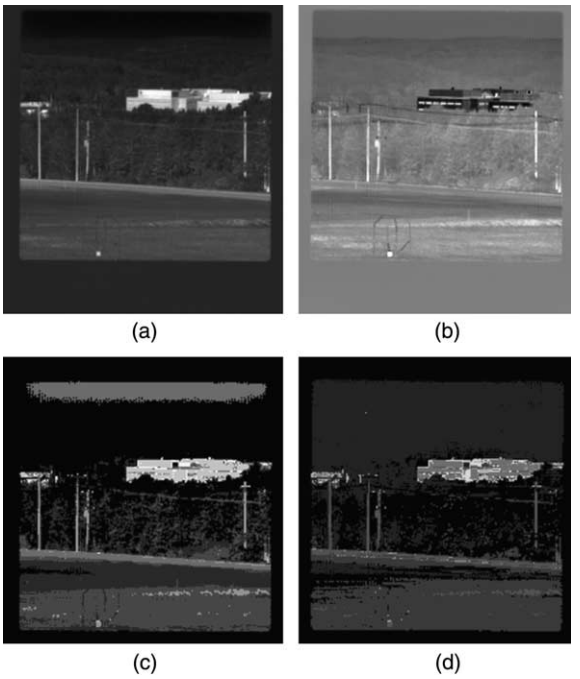

Fig. 2. (a) First principal eigenimage of MWIR hyperspectral image, (b) second principal eigenimage of MWIR hyperspectral image, (c) nine level segmentation of MWIR hyperspectral image and (d) 14 level segmentation of MWIR hyperspectral image.

Fig. 3 is an example in the SWIR for a HYDICE image [8]. Here we formed a 3D histogram from three eigenimages and extending the 2D technique in a straightforward fashion, we generated the 13 level segmentation shown in Fig. 3.

The examples in Figs. 2 and 3 incorporate some recent refinements to the simple technique described above [9] which address two issues. First, the nearest peak as measured by Euclidean distance used in forming the template (Fig. 1e) in effect weights each peak and each component equally. We shall deviate from this measure by means of a Gaussian model. Secondly, our float-to-integer mapping before histogram generation has been done by linear scaling. We have switched to well-developed techniques of histogram-based nonlinear mappings rather than linear scaling, with the final segmentations guided by an entropy measure.

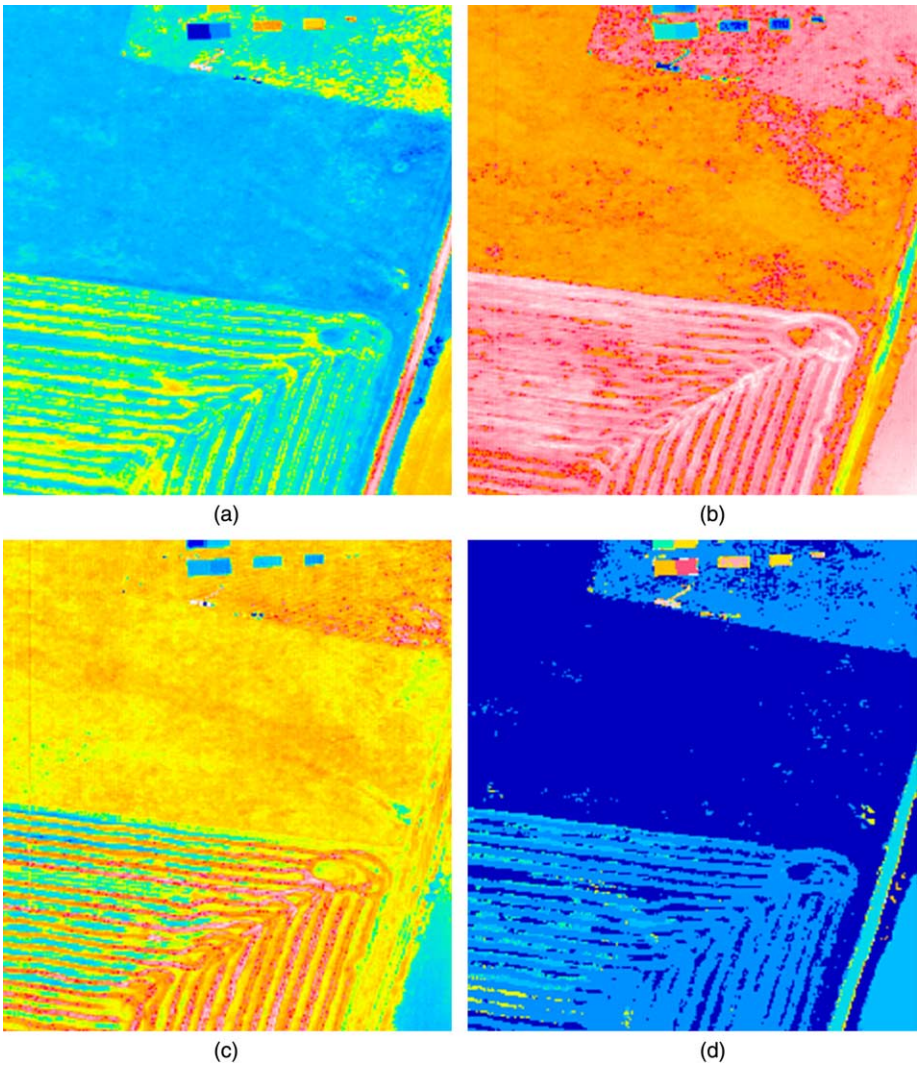

Fig. 3. (a) First principal eigenimage of HYDICE hyperspectral image, (b) third principal eigenimage of HYDICE hyperspectral image, (c) fourth principal eigenimage of HYDICE hyperspectral image and (d) 13 level segmentation of HYDICE hyperspectral image from three-dimensional histogram.

3. Gaussian model

When we model the histogram peaks by a Gaussian approximation, we take the peak height as the co-efficient and the peak position as the Gaussian mean. The less obvious choice is what to use for peak width or standard deviation. We use the co-histogram [10] of each component independently (in its integer mapped form) to estimate variance. The co-histogram, which is one type of co-occurrence matrix, is a histogram plot of the association of the gray level of a central pixel with the gray levels of its 3 by 3 neighbors. For example, a value of 20 at co-ordinate (40, 45) reflects 20 occurrences of a pixel of value 40 with a neighbor of 45 in the component image. Note that the point (45, 40) must also have a value of 20, which leads to mirror symmetry along the diagonal.

Fig. 4 shows the co-histograms of the two eigenimages in Fig. 1. The variance computed from

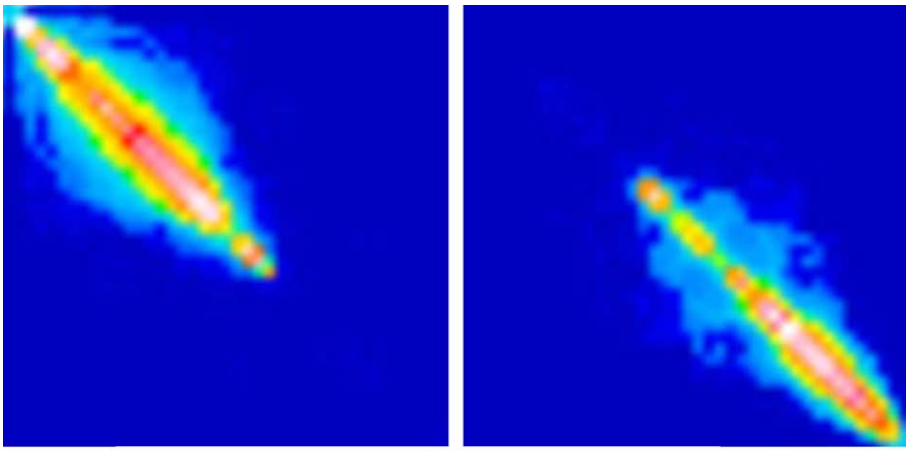

Fig. 4. (a) Co-histogram of the first principal eigenimage of Fig. 1 image and (b) co-histogram of the second principal eigenimage.

the values along (either) the vertical or horizontal line, with the diagonal value taken as the mean, is used to estimate the natural width in each component dimension of that peak. For the 2D case, we assume a simple Gaussian with no correlations between the component images and no cross terms for each peak.

$$P = A\exp[-(x_1 - m_1)^2/2\sigma_1^2]\exp[-(x_2 - m_2)^2/2\sigma_2^2] \tag{1}$$

P is a measure of the influence of the peak at general location in the histogram x_1, x_2. A is the peak height of the 2D histogram located at position (m_1, m_2) in the 2D histogram; the variances σ_1^2 and σ_2^2 are independently evaluated from the component co-histograms. For example, for a peak at (40, 30) in the 2D histogram, one computes the σ_1^2 variance from the horizontal (or equivalently vertical) line of value equal to 40 in the first co-histogram and likewise for the σ_2^2 variance at value equal to 30 in the second co-histogram.

Instead of the minimum distance criterion, we can now assign a pixel with values x_1, x_2 to a peak of maximum estimated probability at that location, i.e. the peak that yields a minimum value to the expression:

$$\frac{(x_1 - m_1)^2}{\sigma_1^2} + \frac{(x_2 - m_2)^2}{\sigma_2^2} - 2\ln A \tag{2}$$

Peaks (distributions) with large maximum (A in Eq. (2)) and/or large variance extend their influence to greater distances. Keep in mind that Eq. (2) also governs the relative influence of two components (which share the same value of A) through the σ_1^2 and σ_2^2 terms. This is shown clearly in Fig. 5 in which we segment the tent image on the basis of the first and seventh eigenimage: the latter component and its co-histogram is shown in Fig. 5(a) and (b). In Fig. 5(c) and (d), we compare the segmentation based on Euclidean distance and one using expression 2, respectively. Since the variances associated with the second histogram component are greater than those of the first, in the second segmentation, the influence of the seventh (noisy) eigenimage is down-weighted (in the sense of less constraints on the second component coordinate in regard to its peak association) leading to a very evident improvement Fig. 5(d). Less dramatic but still significant improvements in segmentation quality occur in switching to our Gaussian model even with two or three high quality PC(s) used to form the histogram.

4. Entropy-guided mappings

We next discuss the second major refinement to the initial segmentation procedure, which addresses how to map the floating-point PC data into

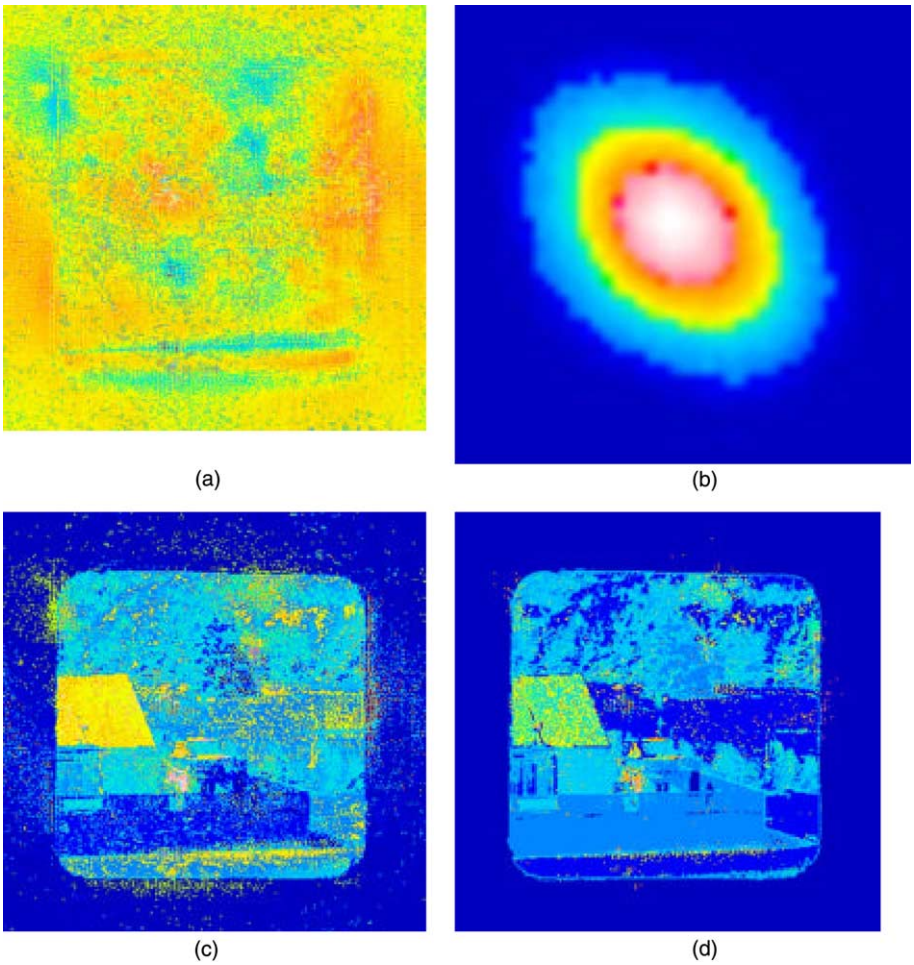

Fig. 5. (a) Seventh principal eigenimage, (b) co-histogram of seventh principal eigenimage, (c) segmentation from first and seventh components based on Euclidean distance model and (d) segmentation based on Gaussian model using co-histograms to estimate parameters.

the integer bins needed to generate the histograms. This issue is especially important when a small number of segmentation levels is desired, say less than 15, and the image consists largely of low spatial frequency background, but with occasional high spatial frequency areas at levels removed from the background. The MWIR hyperspectral image of Fig. 2, where the building is such a high spatial frequency region, will serve as an example to explore this issue.

In our initial formulation of the histogram segmentation technique [9], we used linear scaling and iterated the specified range of integer bins until the user-desired number of segmentation levels was achieved. The only other user choice was to fix the minimum number of pixels at a maximum histogram location accepted as a peak (default "2" used initially). We observed that increasing this value would have striking effects on the consequent segmentation (see for example Fig. 4(c) and (d) in [9]). However, choosing the peak definition that gives the most pleasing segmentation is both subjective and empirical and we prefer the alternative scaling technique outlined next.

Based on past research on the display of infrared imagery [11], which treats mapping from high

dynamic range histograms (the "raw" histogram) to 8-bit histograms (the display histogram), we introduced a nonlinear mapping, histogram projection (HP), which treats each occupied level of the raw histogram equally. This was in contrast to the standard technique of histogram equalization (HE) which allots integer range in the display histogram in proportion to histogram height in the raw histogram. The HE mapping tends to lump high frequency spatial regions at sparsely occupied levels into a few display levels. We further introduced a more generalized mapping, plateau equalization (PE), in which by introducing a plateau or saturation level into the raw histogram computation, one generates mappings which are intermediate between those of HP (plateau = 1) and those of HE (plateau above the maximum value in the raw histogram).

We have adopted the plateau mapping technique for the present scaling problem as follows. First we linearly scale the PC floating point values into a large integer range (typically 0–1000) which retains the basic statistics of the floating point values. Then we employ PE to map from the "raw" histogram of 1000 to the much lower integer ranges needed to form the histograms used in the segmentation techniques (ranges from 5 to 50 integer bins are typical for the 2D histograms.) To automate the process of choosing the plateau level that gives the "best" result at a user-specified number of final segmentation levels, we employ an entropy measure. Our experience is that the first eigenimage of a PC analysis generally provides a high contrast view of the scene with reasonable balance between high and low spatial frequency regions. Assuming segmentation to N levels is sought, we linearly scale the first eigenimage to N levels in order to compute the *baseline entropy* of this component by the expression:

$$E = \sum_{0}^{N-1} -(f_i \log 2(f_i)) \tag{3}$$

where f_i is the fraction of pixels at level i. We next run our algorithm with the desired subset of PC(s) over a range of plateau levels, typically 1, 5, 10, 15, 20, 25 and 30. At each plateau level, we generate the histograms produced by the corresponding PE mapping and iteratively change the final number of integer bins downward until we obtain the specified N level segmentation (or just below). The entropy of the final segmentation is computed with Eq. (3). After the segmentations associated with each plateau level are generated with their associated entropy values, the preferred final result is taken as that segmentation with entropy closest to the *baseline entropy* of the first PC.

In Fig. 6(a) and (b), we show the results of this technique for the plateau = 1 result (close to linear scaling if the PC floats form a smooth continuum) as compared to the plateau level result with the best entropy match. In Fig. 6(b) with the plateau level at 30, the entropy measure has guided us to a

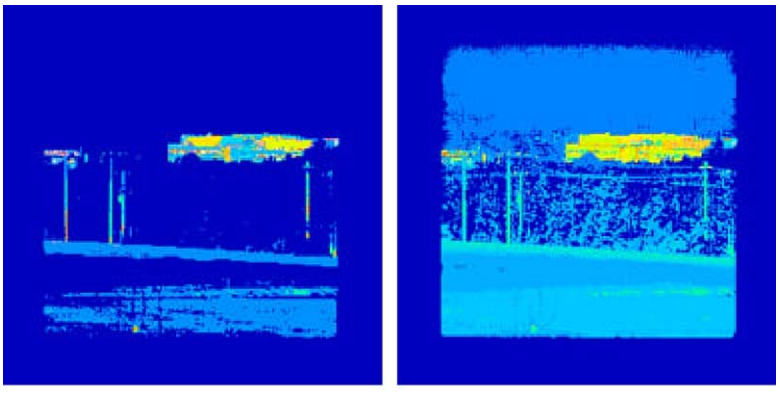

Fig. 6. (a) Segmentation of Fig. 2 image to 12 levels scaling the data linearly and without using the entropy metric and (b) segmentation to 12 levels with histogram-based scaling of the data using the entropy metric.

segmentation with clean differentiation among the field stop, sky, building, trees, and ground, similar to the first PC image. In contrast, in the linear-scaled result of Fig. 6(a), the sky, trees and part of the ground are combined into one segment.

5. Discussion

We have described a simple and robust technique for segmenting multi-dimensional data cubes based on multi-dimensional histogram formation. The histogram approach avoids any spatial constraints and no training pixels are required. The only user-specified parameter is the desired number of segmentation levels leading to a series of segmentations from fine to coarse. Such segmentations are a form of unsupervised classification based on pixel clustering. As such, our technique is simpler with less user intervention than computing techniques such as the K-means method [12] or hierarchical descending methods [13,14]. Clustering and compression are combined in these histogram-based segmentations in that spectrally similar pixels are reduced to two or three similar values in the principal eigenimages with the

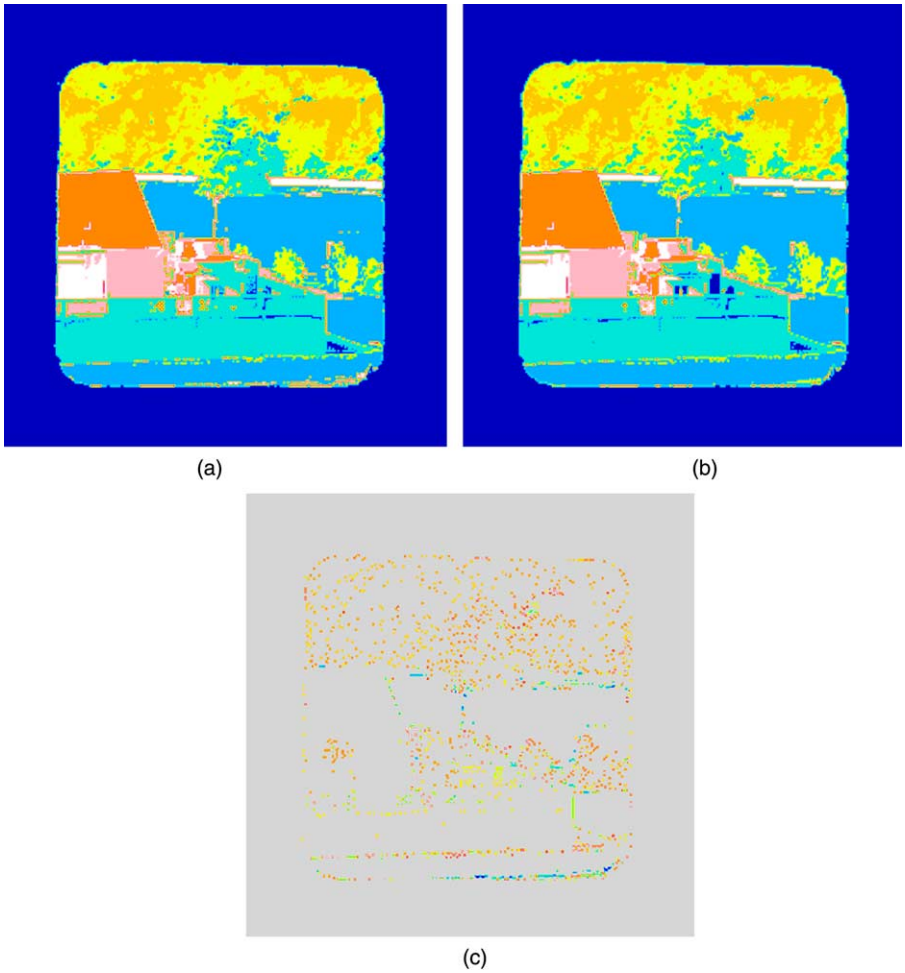

Fig. 7. (a) Segmentation to eight levels of visible data cube of Fig. 1, (b) K-means segmentation to eight levels and (c) difference image of two segmentations.

peaks in the histogram locating the centers of such groups. Given that the PC technique generates components in order of their variance, we anticipate that the technique should be robust and noise resistive.

In Fig. 7, we show two 8 level segmentations using the present methods Fig. 7(a) and the widely used K-means algorithm [12] (Fig. 7(b)). The K-means algorithm is a classifier where the operator specifies a initial seed set of K centers, i.e. the profiles of K pixels, assigns each pixel to the nearest center using some spectral distance measure, updates the centers by averaging over the profiles of the assigned pixels, and iterates the process. In producing our K-means result, we selected eight pixels randomly from each segmentation level from the 7a segmentation as the initial seeds and allowed 10 iterations using a Euclidian measure. At a casual glance, the two segmentations are quite similar; however a difference image (Fig. 7(c)) indicates that some 2000 pixels, 3% of the total are indeed different. These are concentrated at boundary regions and in the foliage. One anticipates that the K-means would provide more accurate final classifications than the present histogram technique; the latter, however, could provide reasonable starting seeds for the former. Huang [14] presents evidence that this choice affects the final classification.

Several issues will be the focus of future work in enhancing our segmentation methodology. The first is the desire to automate or at least constrain the user choice of number of clusters or segmentation levels. We are exploring using the data itself and the evolution with scaling parameter of the histogram peaks to find stable or preferred number(s) of clusters. A second issue arises in the absence of ground truth: namely, how to move beyond a subjective user assessment of the quality of a segmentation and formulate an objective merit function for comparing segmentations with similar numbers of levels. Using locally obtained MWIR images with ground truth, we plan to explore the several issues mentioned: namely, extraction of data-indicated number of clusters; a merit function to evaluate segmentation quality; and robustness to data quality.

Acknowledgements

The cameras used to collect the visible and MWIR data were designed and fabricated by William Ewing, Toby Reeves and Steven DiSalvo of our laboratory. The HYDICE data was provided by the Spectral Information Technology Application Center (SITAC). We are grateful to Linda Bouthillette for graphic assistance and to Virgil Vickers for critical review of the manuscript. This work was carried out under Air Force Task 2305BN00. We would like to acknowledge partial support of the Paul Ivanier Center for Robotics and Industrial Production, Beer-Sheva, Israel. This work was performed while one of the authors (SRR) held a National Research Council Research Associateship Award at the Air Force Research Laboratory at Hanscom AFB.

References

[1] D.W. Paglieroni, D.E. Perkins, Automatic extraction of closed pixel clusters for target cueing in hyperspectral images, Proc. SPIE 4473 (2001) 51–61.

[2] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison Wesley, Reading, MA, 1993.

[3] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.

[4] F.M. Wahl, Digital Image Signal Processing, Artech House, Boston, 1987 (Chapter 5).

[5] J.A. Richards, J. Xiuping, Remote Sensing Digital Image Analysis: An Introduction, Springer-Verlag, Berlin, 1999 (Chapter 6).

[6] J.E. Murguia, T.D. Reeves, J.M. Mooney, W.S. Ewing, F.D. Shepherd, A.K. Brodzik, A compact visible/near infrared hyperspectral imager, Proc. SPIE 4028 (2000) 457–468.

[7] J.M. Mooney, V.E. Vickers, M. An, A.K. Brodzik, High throughput hyperspectral infrared camera, J. Opt. Soc. Am. A 14 (1997) 2951–2961.

[8] The HYDICE imagery, taken over the CART/ARM Site Lamont, was provided by the Spectral Information Technolgy Application Center (SITAC). HYDICE is a push-broom imaging spectrometer with 210 spectral bands over the 0.4 to 2.5 micron spectral range.

[9] J. Silverman, C.E. Caefer, J.M. Mooney, M.M. Weeks, P. Yip, An automated clustering/segmentation of hyperspectral images based on histogram thresholding, Proc. SPIE 4480 (2002) 65–75;
J. Silverman, S.R. Rotman, C.E. Caefer, Segmentation of hyperspectral images based on histograms of principal components, Proc. SPIE 4816 (2002) 270–277.

[10] R. Haralick, K. Shanmugam, I. Dinstein, Texture features for image classification, IEEE T. Syst. Man Cyb. 3 (1973) 610–621.
[11] V.E. Vickers, Plateau equalization algorithm for real-time display of high-quality infrared imagery, Opt. Eng. 35 (1996) 1921–1926.
[12] R.A. Schowengerdt, Remote Sensing, Academic Press, Boston, 1997 (Chapter 9).
[13] N. Viovy, Automatic classification of time series (ACTS): a new clustering method for remote sensing time series, Int. J. Remote Sens. 21 (2000) 1537–1560.
[14] K. Huang, The use of a newly developed algorithm of divisive hierarchical clustering for remote sensing image analysis, Int. J. Remote Sens. 23 (2002) 3149–3168.