

# Unsupervised Segmentation of Hyper-spectral Images via Diffusion Bases

Alon Schclar<sup>1</sup> and Amir Averbuch<sup>2</sup>

<sup>1</sup>*School of Computer Science, The Academic College of Tel-Aviv Yaffo, POB 8401, Tel Aviv 61083, Israel*

<sup>2</sup>*School of Computer Science, Tel Aviv University, POB 39040, Tel Aviv 69978, Israel*

**Keywords:** Segmentation, Diffusion Bases, Dimensionality Reduction, Hyper-spectral Sensing.

**Abstract:** In the field of hyper-spectral sensing, sensors capture images at hundreds and even thousands of wavelengths. These *hyper-spectral* images, which are composed of hyper-pixels, offer extensive intensity information which can be utilized to obtain segmentation results which are superior to those that are obtained using RGB images. However, straightforward application of segmentation is impractical due to the large number of wavelength images, noisy wavelengths and inter-wavelength correlations. Accordingly, in order to efficiently segment the image, each pixel needs to be represented by a small number of features which capture the structure of the image. In this paper we propose the diffusion bases dimensionality reduction algorithm (Schclar and Averbuch, 2015) to derive the features which are needed for the segmentation. We also propose a simple algorithm for the segmentation of the dimensionality reduced image. We demonstrate the proposed framework when applied to hyper-spectral microscopic images and hyper-spectral images obtained from an airborne hyper-spectral camera.

## 1 INTRODUCTION

Image segmentation is the process of partitioning an image into disjoint subsets of pixels in which pixels belong to the same subset are more similar than pixels that belong to different subsets. Each subset is referred to as a segment.

A regular CCD camera provides very limited spectral information as it is equipped with sensors that only capture details that are visible to the naked eye. However, a *hyper-spectral* camera is equipped with multiple sensors - each sensor is sensitive to a particular range of the light spectrum including spectrum ranges that are not visible to the naked eye - namely, infra-red and ultra-violet. Its output contains the reflectance values of a scene at *all* the wavelengths of the sensors. Hyper-spectral cameras can be hand held (Zheludeva et al., 2015) or they can be mounted on airplanes (e.g. (Tarabalka et al., 2010)) or microscopes (Cassidy et al., 2004).

A hyper-spectral image is composed of a set of images - one for each wavelength. We refer to a set of wavelength values at a coordinate  $(x, y)$  as a *hyper-pixel*. Each hyper-pixel can be represented by a vector in  $\mathbb{R}^n$  where  $n$  is the number of wavelengths. This data can be used to achieve inferences that can not be derived from a limited number of wavelengths which are obtained by regular cameras.

Commonly, the number of wavelengths is much higher than the actual degrees of freedom of the data. Unfortunately, this phenomenon is usually unavoidable due to the inability (lack of knowledge which sensor values are more important for the task at hand) to produce a special sensor for each application. Consider for example a task that separates red objects from green objects using an off-the-shelf digital camera. In this case, the camera will produce, in addition to the red and green channels, a blue channel, which is unnecessary for this task.

Naturally, effective utilization of the wealth of wavelengths can yield segmentation results that are better than those obtained by merely using RGB data, for example, by incorporating infra-red and ultra-violet wavelengths. One can simply apply classical image processing techniques to each wavelength image *individually*. However, this will not utilize inter-wavelength connections, which are inherent in spectral signatures. Furthermore, the high number of wavelengths renders the application of segmentation algorithms to the entire hyper-spectral image useless due to the curse of dimensionality. Thus, the entire hyper-spectral cube needs to be processed in order to analyze the physical nature of the scene. Naturally, this has to be done efficiently due to the large volume of the data.

Typical hyper-spectral images contain a high de-

gree of correlation between many of the wavelengths which renders many of them redundant. Moreover, certain wavelengths contain noise as a result of poor lighting conditions and the physical condition of the camera at the time the images were captured. Accordingly, the noise and the redundant data need to be removed while maintaining the information which is vital for the segmentation. This information should be represented as concisely as possible i.e. each hyper-pixel should be represented using a small number of attributes. This will alleviate the curse of dimensionality and allow the efficient application of segmentation algorithms to the concisely represented hyper-spectral image. To achieve this, dimensionality reduction needs to be applied to the hyper-spectral image.

In this paper we propose to reduce the dimensionality of the hyper-spectral image by using the recently introduced diffusion bases (DB) dimensionality reduction algorithm (Schclar and Averbuch, 2015). The DB algorithm efficiently captures non-linear inter-wavelength correlations and produces a low-dimensional representation in which the amount of noise is drastically reduced. We also propose a fast and simple histogram-based segmentation algorithm which will be applied to the low-dimensional representation.

We use a simple and efficient histogram-based method for automatic segmentation of hyper-spectral volumes in which the DB algorithm plays a key role. The proposed method clusters hyper-pixels in the reduced-dimensional space. We refer to this method as the *Wavelength-wise Global* (WWG) segmentation algorithm.

This paper is organized as follows: in section 2 we present a survey of related work on segmentation of hyper-spectral images. The diffusion bases scheme (Schclar and Averbuch, 2015) is described in section 3. In section 4 we introduce the two phase *Wavelength-wise Global* (WWG) segmentation algorithm. Section 5 contains experimental results from the application of the algorithm to several hyper-spectral images. Concluding remarks are given in section 6.

## 2 RELATED WORKS

Segmentation methods for hyper-spectral images can be divided into two categories - supervised and unsupervised. Supervised methods segment the image using either *a-priori* spectral information of the sought after segments or information regarding the shape of the segments. Some methods use both types of information. Unsupervised segmentation techniques do

not utilize any a-priori information. The method proposed in this paper falls into the latter category.

The method in (Ye et al., 2010) uses both *a-priori* spectral information and shape information of the segments. Specifically, they use the model which is proposed in (Chan et al., 2006) which is a convexification of the two-phase version of the Mumford-Shah model. The model uses variational methods to find a smooth minimal length curve that divides the image into two regions that are as close as possible to being homogeneous. The a-priori spectral and shape information is incorporated in the variational model and its optimization.

In (Li et al., ) a variational model for simultaneous segmentation and denoising/deblurring of a hyper-spectral image which models the image as a set of three-dimensional tensors. The spectral signatures of the sought after materials is known a-priori and is incorporated in the model. The segmentation is obtained via a statistical moving average method which uses the spatial variation of spectral correlation. Specifically, a coarse-grained spectral correlation function is computed over a small moving 2D spatial cell of fixed shape and size. This function produces sharp variations as the averaging cell crosses a boundary between two materials.

In (Li et al., 2010) a supervised Bayesian segmentation approach is proposed. The method makes use of both spectral and spatial information. The two-phase algorithm first implements a learning step, which uses the multinomial logistic regression via variable splitting and augmented (LORSAL) (Bioucas-Dias and Figueiredo, 2009) algorithm to infer the class distributions. A segmentation step follows which infers the labels from a posterior distribution built on the learned class distributions. A maximum a-posterior (MAP) segmentation is computed via a min-cut based integer optimization algorithm. The algorithm also implement an active learning technique based on the mutual information (MI) between the MLR regressors and the class labels in order to reduce the size of the training set.

In (Tarabalka et al., 2010) an extension to the the watershed (Vincent and Soille, 1991) segmentation algorithm is proposed. Specifically, the algorithm is used to to define information about spatial structures and uses one-band gradient functions. The segmentation maps are incorporated into a spectral-spatial classification scheme based on a pixel-wise Support Vector Machine classifier.

### 3 THE DIFFUSION BASES DIMENSIONALITY REDUCTION ALGORITHM

The Diffusion bases (DB) dimensionality reduction algorithm (Schclar and Averbuch, 2015) reduces the dimensionality of a dataset by utilizing the inter-coordinate variability of the original data (in this sense it is dual to the *Diffusion Maps* algorithm (Coifman and Lafon, 2006; Schclar, 2008; Schclar et al., 2010)). It first constructs the graph Laplacian using the image wavebands as the datapoints. It then uses the Laplacian eigenvectors as an orthonormal system and projects the hyper-pixels on it. The eigenvectors are sorted in descending order according to their magnitude and only the eigenvectors that correspond to the highest eigenvalues are used. These eigenvectors capture the *non-linear* coordinate-wise variability of the original data. Although bearing some similarity to PCA, this process yields better results than PCA due to: (a) its ability to capture non-linear manifolds within the data by local exploration of each coordinate; (b) its robustness to noise. Furthermore, this process is more general than PCA and it produces similar results to PCA when the weight function  $w_\varepsilon$  is *linear* e.g. the inner product.

Let  $\Gamma = \{x_i\}_{i=1}^m$ ,  $x_i \in \mathbb{R}^n$ , be the original dataset of hyper-pixels and let  $x_i(j)$  denote the  $j$ -th coordinate (the reflectance value of the  $j$ -th band) of  $x_i$ ,  $1 \leq j \leq n$ . We define the vector  $x'_j \triangleq (x_1(j), \dots, x_m(j))$  to be the  $j$ -th coordinate of all the points in  $\Gamma$  i.e. the image corresponding to the  $j$ -th band.. We construct the set

$$\Gamma' = \{x'_j\}_{j=1}^n. \quad (1)$$

Let  $w_\varepsilon(x_i, x_j)$ , be a weight function which measures the pairwise similarity between the points in  $\Gamma'$ . A Markov transition matrix  $P$  is constructed by normalizing the sum of each row in the matrix  $w_\varepsilon$  to be 1:

$$p(x'_i, x'_j) = \frac{w_\varepsilon(x'_i, x'_j)}{d(x'_i)}$$

where  $d(x'_i) = \sum_{j=1}^n w_\varepsilon(x'_i, x'_j)$ . Next, eigen-decomposition of  $p(x'_i, x'_j)$  is performed

$$p(x'_i, x'_j) \equiv \sum_{k=1}^n \lambda_k v_k(x'_i) \mu_k(x'_j)$$

where the left and the right eigenvectors of  $P$  are given by  $\{\mu_k\}$  and  $\{v_k\}$ , respectively, and  $\{\lambda_k\}$  are the eigenvalues of  $P$  in descending order of magnitude. We use the eigenvalue decay property of the eigen-decomposition to extract only the first  $\eta(\delta)$  eigenvectors  $B \triangleq \{v_k\}_{k=1, \dots, \eta(\delta)}$  which contain the *non-linear*

directions with the highest variability of the coordinates of the original dataset  $\Gamma$ . We project the original data  $\Gamma$  onto the basis  $B$ . Let  $\Gamma_B$  be the set of these projections:  $\Gamma_B = \{g_i\}_{i=1}^m$ ,  $g_i \in \mathbb{R}^{\eta(\delta)}$ , where  $g_i = (x_i \cdot v_1, \dots, x_i \cdot v_{\eta(\delta)})$ ,  $i = 1, \dots, m$  and  $\cdot$  denotes the inner product operator.  $\Gamma_B$  is the reduced dimension representation of  $\Gamma$  and it contains the coordinates of the original points in the orthonormal system whose axes are given by  $B$ .

### 4 THE WAVELENGTH-WISE GLOBAL (WWG) SEGMENTATION ALGORITHM

We introduce a simple and efficient two-phase approach for the segmentation of hyper-spectral images. The first phase reduces the dimensionality of the data using the DB algorithm and the second stage applies a histogram-based method to cluster the low-dimensional data.

We model a hyper-spectral image as a three dimensional cube where the first two coordinates correspond to the position  $(x, y)$  and the third coordinate corresponds to the wavelength  $\lambda_k$ . Let

$$I = \left\{ p_{ij}^{\lambda_k} \right\}_{i,j=1, \dots, m; k=1, \dots, n} \in \mathbb{R}^{m \times m \times n} \quad (2)$$

be a hyper-spectral image cube, where the size of the image is  $m \times m$  and  $n$  is the number of wavelengths. For notation simplicity, we assume that the images are square. It is important to note that almost always  $n \ll m^2$ .

$I$  can be viewed in two ways:

1. **Wavelength-wise:**  $I = \{I^{\lambda_l}\}$  is a collection of  $n$  images of size  $m \times m$  where

$$I^{\lambda_l} \triangleq \left( p_{ij}^{\lambda_l} \right) \in \mathbb{R}^{m \times m}, 1 \leq l \leq n \quad (3)$$

is the image that corresponds to wavelength  $\lambda_l$ .

2. **Point-wise:**  $I = \left\{ \vec{T}_{ij} \right\}_{i,j=1}^m$  is a  $m \times m$  collection of  $n$ -dimensional vectors where

$$\vec{T}_{ij} \triangleq \left( p_{ij}^{\lambda_1}, \dots, p_{ij}^{\lambda_n} \right) \in \mathbb{R}^n, 1 \leq i, j \leq m \quad (4)$$

is the hyper-pixel at position  $(i, j)$ .

The proposed WWG algorithm assumes the wavelength-wise setting of a hyper-spectral image. Thus, we regard each image as a  $m^2$ - dimensional vector. Formally, let

$$\tilde{I} \triangleq \left( \pi_{i, \lambda_l} \right)_{i=1, \dots, m^2; l=1, \dots, n} \in \mathbb{R}^{m^2 \times n} \quad (5)$$

be a 2-D matrix corresponding to  $I$  where

$$\pi_{i+(j-1)\cdot m, \lambda_k} \triangleq p_{ij}^{\lambda_k}, 1 \leq k \leq n, 1 \leq i, j \leq m,$$

( $p_{ij}^{\lambda_k}$  is defined in Eq. 2) and let

$$\tilde{I}^{\lambda_k} \triangleq \begin{pmatrix} \pi_{1, \lambda_k} \\ \vdots \\ \pi_{m^2, \lambda_k} \end{pmatrix} \in \mathbb{R}^{m^2}, 1 \leq k \leq n \quad (6)$$

be a column vector that corresponds to  $I^{\lambda_k}$  (see Eq. 3).

#### 4.1 Phase 1: Reduction of Dimensionality via DB

Different sensors can produce values at different scales. Thus, in order to have a uniform scale for all the sensors, each column vector  $\tilde{I}^{\lambda_k}$ ,  $1 \leq k \leq n$ , is normalized to be in the range  $[0,1]$ .

We form the set of vectors  $\Gamma = \{\tilde{I}^{\lambda_1}, \dots, \tilde{I}^{\lambda_n}\}$  from the columns of  $\tilde{I}$  and we apply the DB Algorithm to  $\Gamma$ . We denote the dimension-reduced representation of  $\Gamma$  by  $\Gamma_B$ .

#### 4.2 Phase 2: Histogram-based Segmentation

We introduce a histogram-based segmentation algorithm that extracts objects from  $\Gamma$  using  $\Gamma_B$ . For notation convenience, we denote  $\eta(\delta) - 1$  by  $\eta$  hereinafter. We denote by  $G$  the cube representation of the set  $\Gamma_B$  in accordance with Eq. 2:

$$G \triangleq \left( g_{ij}^k \right)_{i,j=1,\dots,m;k=1,\dots,\eta}, G \in \mathbb{R}^{m \times m \times \eta}.$$

We assume a wavelength-wise setting for  $G$ . Let  $\tilde{G}$  be a 2-D matrix in the setting defined in Eq. 5 that corresponds to  $G$ . Thus,  $G^l \triangleq \left( g_{ij}^l \right)_{i,j=1,\dots,m} \in \mathbb{R}^{m \times m}$ ,  $1 \leq l \leq \eta$  corresponds to a column in  $\tilde{G}$  and  $\vec{g}_{ij} \triangleq \left( g_{ij}^1, \dots, g_{ij}^\eta \right) \in \mathbb{R}^\eta$ ,  $1 \leq i, j \leq m$  corresponds to a row in  $\tilde{G}$ . The coordinates of  $\vec{g}_{ij}$  will be referred to hereinafter as *colors*.

The segmentation is achieved by clustering hyper-pixels with similar colors. This is based on the assumption that similar objects in the image will have a similar set of *color* vectors in  $\Gamma_B$ . These colors contain the correlations between the original hyper-pixels and the global inter-wavelength changes of the image. Thus, homogeneous regions in the image have similar correlations with the changes i.e. *close* colors

where closeness between colors is measured by the Euclidean distance.

The segmentation-by-colors algorithm consists of the following steps:

##### 1. Normalization of the Input Image Cube $G$ :

First, we normalize each wavelength of the image cube to be in  $[0,1]$ . Let  $G^k$  be the  $k$ -th ( $k$  is the color index) color layer of the image cube  $G$ . We denote by  $\hat{G}^k = \left( \hat{g}_{ij}^k \right)_{i,j=1,\dots,m}$  the normalization of  $G^k$  and define it to be

$$\hat{g}_{ij}^k \triangleq \frac{g_{ij}^k - \min \{G^k\}}{\max \{G^k\} - \min \{G^k\}}, 1 \leq k \leq \eta. \quad (7)$$

##### 2. Uniform Quantization of the Normalized Input Image Cube $\hat{G}$ :

Let  $l \in \mathbb{N}$  be a given number of quantization levels. We uniformly quantize every value in  $G^k$  to be one of  $l$  possible values. The quantized matrix is given by  $Q$ :

$$Q \triangleq \left( q_{ij}^k \right)_{i,j=1,\dots,m;k=1,\dots,\eta}, q_{ij}^k \in \{1, \dots, l\} \quad (8)$$

where  $q_{ij}^k = \lfloor l \cdot \hat{g}_{ij}^k \rfloor$ . We denote the quantized *color* vector at coordinate  $(i, j)$  by

$$\vec{c}_{ij} \triangleq \left( q_{ij}^1, \dots, q_{ij}^\eta \right) \in \mathbb{R}^\eta, 1 \leq i, j \leq m. \quad (9)$$

##### 3. Construction of the Frequency *color* Histogram:

We construct the frequency function  $f: \{1, \dots, l\}^\eta \rightarrow \mathbb{N}$  where for every  $\kappa \in \{1, \dots, l\}^\eta$ ,  $f(\kappa)$  is the number of quantized color vectors  $\vec{c}_{ij}$ ,  $1 \leq i, j \leq \eta$ , that are equal to  $\kappa$ .

##### 4. Finding Peaks in the Histogram:

Local maxima points (called *peaks*) of the frequency function  $f$  are detected. We assume that each peak corresponds to a different object in the image cube  $G$ . Here we use the classical notion of segmentation - separating object from the background. Indeed, the highest peak corresponds to the largest homogeneous area which in most cases is the background. The histogram may have many peaks. Therefore, we perform an iterative procedure to find the  $\theta$  highest peaks where the number  $\theta$  of sought after peaks is given as a parameter to the algorithm. This parameter corresponds to the number of objects we seek. The algorithm is also given an integer parameter  $\xi$ , which specifies the  $l_1$  cube radius around a peak. We define the  $\xi$ -neighborhood of a coordinate  $(x_1, \dots, x_\eta)$  to be  $N_\xi(x_1, \dots, x_\eta) =$

$\{(y_1, \dots, y_\eta) | \max_k \{|y_k - x_k|\} \leq \xi\}$ . The coordinates outside the neighborhood  $N_\xi$  are the candidates for the locations of new peaks. An iterative procedure is used in order to find all the peaks. The peaks are labeled  $1, \dots, \theta$ . The output of the algorithm is a set of vectors  $\Psi = \{\vec{\rho}_i\}_{i=1, \dots, \theta}$ ,  $\vec{\rho}_i = (\rho_i^1, \dots, \rho_i^\eta) \in \mathbb{N}^\eta$  that contains the highest peaks. A summary of this step is given in algorithm 1.

#### 5. Finding the Nearest Peak to each color:

Once the highest peaks are found, each quantized color vector is associated with a single peak. The underlying assumption is that the quantized color vectors, which are associated with the same peak, belong to the same object in the color image cube  $I$ . Each quantized color is associated with the peak that is the closest to it with respect to the Euclidean distance. Each quantized color is labeled by the number of its associated peak. We denote by

$$\gamma: \vec{c}_{ij} \mapsto d \in \{1, \dots, \theta\}$$

this mapping function, where

$$\gamma(\vec{c}_{ij}) \triangleq \arg \min_{1 \leq k \leq \theta} \{\|\rho_k - \vec{c}_{ij}\|_{l_\eta}\}.$$

#### 6. Construction of the Output Image:

The final step assigns a unique color  $\kappa_i$ ,  $1 \leq i \leq \theta$  to each coordinate in the image according to its label  $\gamma(\vec{c}_{ij})$ . We denote the output image of this step by  $\Omega$ .

---

Algorithm 1: The *PeaksFinder*. Algorithm.

---

**PeaksFinder**( $f, \theta, \xi$ )

1.  $\Psi \leftarrow \phi$
  2. **while**  $|\Psi| \leq \theta$
  3. Find the next global maximum  $c$  of  $f$ .
  4. Add the coordinates of  $c$  to  $\Psi$ .
  5. Zero all the values of  $f$  in the  $\xi$ -neighborhood of  $c$ .
  6. **end while**
  7. **return**  $\Psi$ .
- 

### 4.3 Hierarchical Extension of the WWG Algorithm

We construct a hierarchical extension to the WWG algorithm in the following way: given the output  $\Omega$  of the WWG algorithm, the user can choose one of the objects and apply the WWG algorithm on the *original* hyper-pixels which belong to this

object. Let  $\chi$  be the color of the chosen object. We define  $\Gamma(\chi)$  to be the set of the *original* hyper-pixels which belong to this object:  $\Gamma(\chi) = \left\{ \left( p_{ij}^1, \dots, p_{ij}^\eta \right) \mid \Omega_{ij} = \chi, i, j = 1, \dots, m \right\}$ . This facilitates a *drill-down* function that enables a finer segmentation of a *specific* object in the image. We form the set  $\Gamma(\chi)'$  from  $\Gamma(\chi)$  as described in section 3 and run the *DiffusionBases* algorithm (Section 3) on  $\Gamma(\chi)'$ . Obviously, the size of the input is smaller than that of the original data, thus allowing the finer segmentation of the chosen object. We denote the result of this stage by  $\Gamma_B(\chi)$ . Next, the WWG is applied on  $\Gamma_B(\chi)$  and the result is given by  $\Omega_{ij}(\chi)$ . The drill-down algorithm is outlined in Algorithm 2. This step can be applied to other objects in the image as well as to the drill-down result.

---

Algorithm 2: A drill-down segmentation algorithm.

---

**DrillDown**( $\Gamma', w_\epsilon, \epsilon, \delta, \chi$ )

1.  $\Gamma_B = \text{DiffusionBases}(\Gamma', w_\epsilon, \epsilon, \delta)$  // Section 3
  2.  $\Omega_{ij} = \text{WWG}(\Gamma_B)$  // Section 4
  3.  $\Gamma_B(\chi) = \text{DiffusionBases}(\Gamma(\chi)', w_\epsilon, \epsilon, \delta)$  //
  4.  $\Omega_{ij}(\chi) = \text{WWG}(\Gamma_B(\chi))$
- 

## 5 EXPERIMENTAL RESULTS

The results are divided into three parts: (a) segmentation of hyper-spectral microscopy images; (b) segmentation of remote-sensed hyper-spectral images; (c) sub-pixel segmentation of remote-sensed images. We provide the results using the two dimensionality reduction schemes that were described in Section 3.

We denote the size of the hyper-spectral images by  $m \times m \times n$  where the size of every wavelength image is  $m \times m$  and  $n$  is the number of wavelengths. The geometry (objects, background, etc.) of each hyper-spectral image is displayed using a gray image  $\Upsilon$ . This image is obtained by averaging the hyper-spectral image along the wavelengths. Given a hyper-spectral image  $I$  of size  $m \times m \times n$ ,  $\Upsilon = \{v_{ij}\}_{i, j = 1, \dots, m}$  is obtained by

$$v_{ij} = \frac{1}{n} \sum_{k=1}^n I_{ij}^k \quad 1 \leq i, j \leq m.$$

We refer to  $\Upsilon$  as the *wavelength-averaged-version* (WAV) of the image. All the results were obtained using the automatic procedure for choosing  $\epsilon$  which is described in (Schlar and Averbuch, 2015).

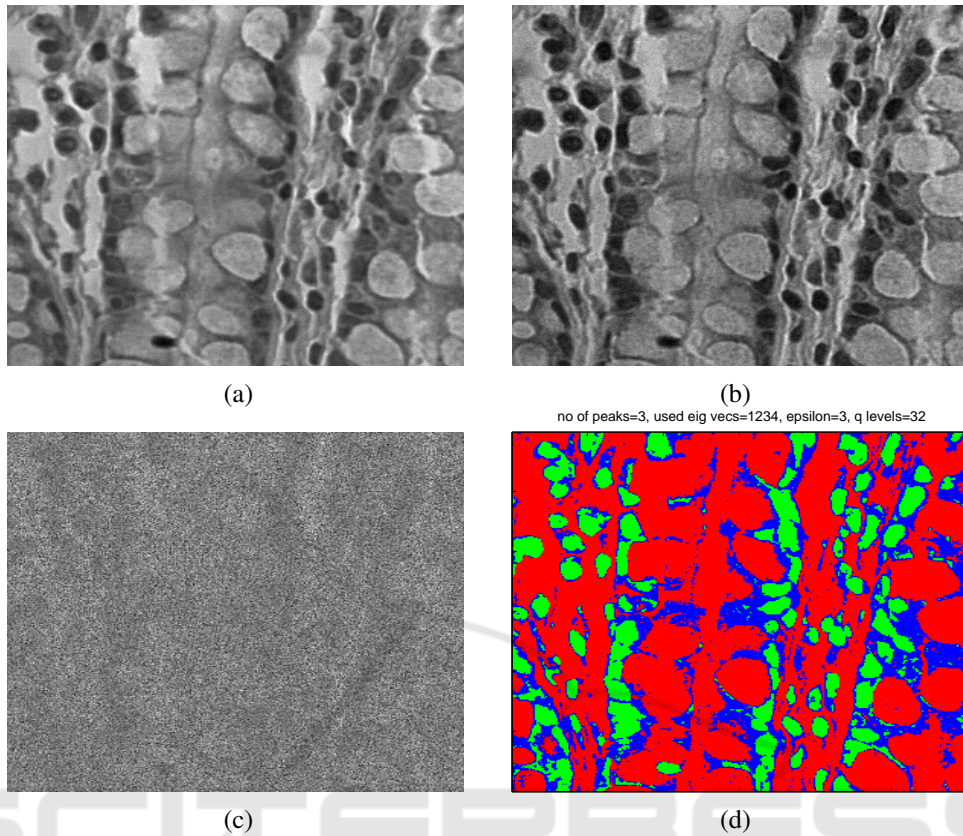


Figure 1: A hyper-spectral microscopy image of a healthy human tissue. (a) The WAV of the original image. (b) The 50<sup>th</sup> wavelength. (c) The 95<sup>th</sup> wavelength. (d) The results of the application of the WWG algorithm with  $\eta(\delta) = 4$ ,  $\theta = 3$ ,  $\xi = 3$ ,  $l = 32$ .

**Segmentation of Hyper-spectral Microscopy Images.** Figure 1 contains samples of healthy human tissues and the results of the application of the WWG algorithm on them. The images are of sizes  $300 \times 300 \times 128$  and  $151 \times 151 \times 128$ , respectively. The images contain three types of substances: nuclei, cytoplasm and glass. The glass belongs to the plate where the tissue sample lies.

Figures 1(b) and 1(c) show the 50<sup>th</sup> and 95<sup>th</sup> wavelengths, respectively. The images in the 50<sup>th</sup> through the 70<sup>th</sup> wavelengths are less noisy than the rest of the wavelengths which resemble Fig. 1(c). Figures 1(d) display the results after the application of the WWG algorithm. The algorithm clearly segments this image into three parts: the background is colored in dark gray, the cytoplasm is colored in medium shaded gray and the nuclei is colored in light gray.

**Segmentation of Remote-Sensed Images.** Figure 2 contains a hyper-spectral satellite image of the Washington DC's National Mall and the result after the application of the WWG algorithm on it. The im-

age is of size  $300 \times 300 \times 100$ . Figure 2(a) shows the WAV of the image. The image contains water, two types of grass, trees, two types of roofs, roads, trails and shadow. Figures 2(b) and 2(c) show the 10<sup>th</sup> and 80<sup>th</sup> wavelengths, respectively. Figure 2(d) is the results of the WWG algorithm where the water is colored in blue, the grass is colored in two shades of light green, the trees are colored in dark green, the roads are colored in red, the roofs are colored in pink and yellow, the trails are colored in white and the shadow is colored in black.

## 6 FUTURE RESEARCH

The results in section 5 were obtained using a Gaussian kernel. It is shown in (Coifman and Lafon, 2006) that any positive semi-definite kernel may be used for the dimensionality reduction. Rigorous analysis of families of kernels to facilitate the derivation of an optimal kernel for a given set  $\Gamma$  is an open problem and is currently being investigated by the authors.

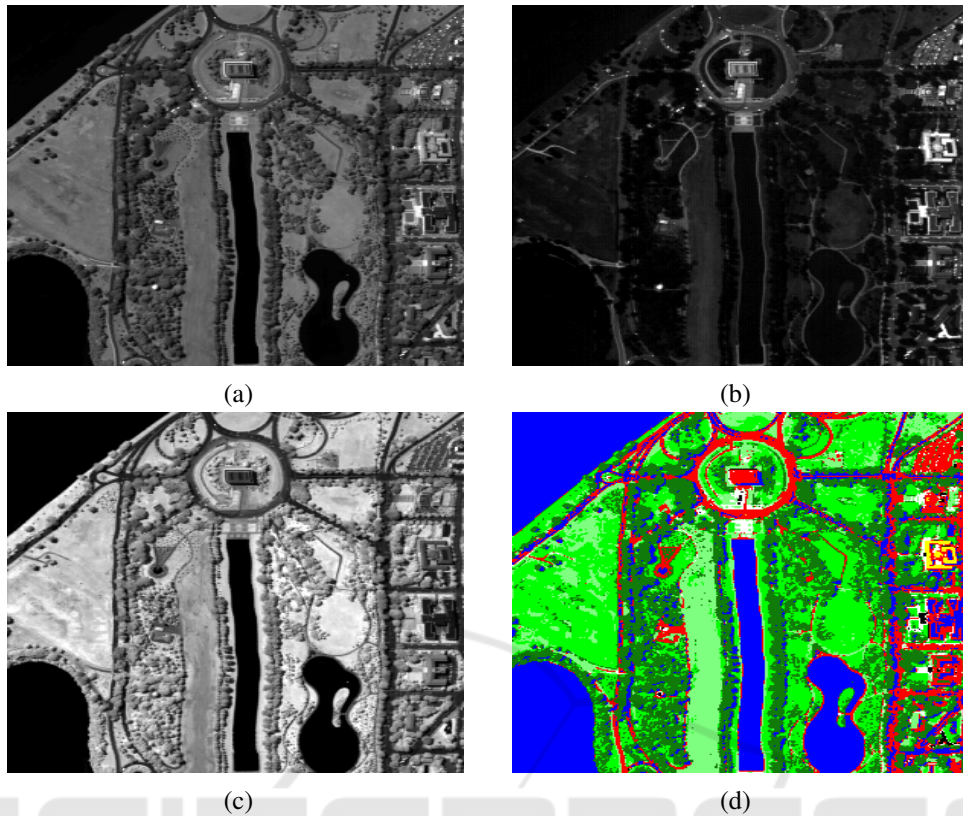


Figure 2: A hyper-spectral satellite image of the Washington DC's National Mall. (a) The WAV of the image. The image contains water, two types of grass, trees, two types of roofs, roads, trails and shadow. (b) The 10<sup>th</sup> wavelength. (c) The 80<sup>th</sup> wavelength. (d) The result after the application of the WWG algorithm using  $\eta(\delta) = 4$ ,  $\theta = 8$ ,  $\xi = 7$ ,  $l = 32$ . The water is colored in blue, the grass is colored in two shades of light green, the trees are colored in dark green, the roads are colored in red, the roofs are colored in pink and yellow, the trails are colored in white and the shadow is colored in black.

The parameter  $\eta(\delta)$  determines the dimensionality of the diffusion space. Automatic choice of this threshold is vital in order to detect the objects in the image cube. A rigorous way for choosing  $\eta(\delta)$  is currently being studied by the authors. Naturally,  $\eta(\delta)$  is *data driven* (similarly to choosing  $\epsilon$  in (Schclar and Averbuch, 2015)) i.e. it depends on the set  $\Gamma$  at hand.

## REFERENCES

- Bioucas-Dias, J. and Figueiredo, M. (2009). Logistic regression via variable splitting and augmented lagrangian tools. Technical report, Instituto Superior Tecnico, TULisbon.
- Cassidy, R. J., Berger, J., Lee, K., Maggioni, M., and Coifman, R. R. (2004). Analysis of hyperspectral colon tissue images using vocal synthesis models. In *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1611–1615.
- Chan, T., Esedoglu, S., and Nikolova, M. (2006). Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal Applied Mathematics*, 66:1632–1648.
- Coifman, R. R. and Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis: special issue on Diffusion Maps and Wavelets*, 21:5–30.
- Li, F., Ng, M. K., Plemmons, R., Prasad, S., and Zhang, Q. Hyperspectral image segmentation, deblurring, and spectral analysis for material identification.
- Li, J., Bioucas-Dias, J., and Plaza, A. (2010). Supervised hyperspectral image segmentation using active learning. In *IEEE GRSS Workshop on Hyperspectral Image and Signal Processing*, volume 1, pages 1–4.
- Schclar, A. (2008). *A Diffusion Framework for Dimensionality Reduction*, pages 315–325. Springer US, Boston, MA.
- Schclar, A. and Averbuch, A. (2015). Diffusion bases dimensionality reduction. In *Proceedings of the 7th International Joint Conference on Computational Intelligence (IJCCI 2015) - Volume 3: NCTA, Lisbon, Portugal, November 12-14, 2015.*, pages 151–156.
- Schclar, A., Averbuch, A., Hochman, K., Rabin, N., and Zheludev, V. (2010). A diffusion framework for detection of moving vehicles. *Digital Signal Processing*, 20(1):111–122.

- Tarabalka, Y., Chanussot, J., and Benediktsson, J. A. (2010). Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition*, 43(7):2367–2379.
- Vincent, L. and Soille, P. (1991). Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598.
- Ye, J., Wittman, T., Bresson, X., and Osher, S. (2010). Segmentation for hyperspectral images with priors. In *Proc. 6th International Symposium on Visual Computing, Las Vegas, NV, USA.*, volume 1, pages 1–4.
- Zheludeva, V., Polonena, I., Neittaanmaki-Perttuc, N., Averbuch, A., Gronroos, P. N. M., and Saari, H. (2015). Delineation of malignant skin tumors by hyperspectral imaging using diffusion maps dimensionality reduction. *Biomedical Signal Processing and Control*, 16:48–60.

