

Transfer Functions for Imaging Spectroscopy Data using Principal Component Analysis

A. Broersen and R. van Liere

CWI, Amsterdam, The Netherlands

Abstract

In this paper we present a new application of the principal component analysis (PCA) to generate multidimensional transfer functions. These transfer functions are needed in the volumetric visualization of spectral data to isolate regions that contain interesting peak-shaped features. Both large and small peaks can be equally important and represent the presence of different chemical elements in a dataset. Principal component analysis separates these peaks in different uncorrelated components and can simultaneously identify spatial patterns. This approach is characterized by the direct linkage between the resulting spectral and spatial components. Our method enables us to create an opacity map from these components. One or more mappings can be selected to highlight features in three-dimensional volume visualization.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Display algorithms I.4.10 [Image Processing and Computer Vision]: Multidimensional I.5.3 [Pattern Recognition]: Algorithms

1. Introduction

The use of Direct Volume Rendering (DVR) is a well known method for the visualization of three-dimensional (3D) volumetric datasets. In most volumetric datasets, each voxel contains a scalar value that represents the density of a material on that location. For visualization, a transfer function is a mapping that assigns a color and opacity value to a scalar value. A volume renderer can draw the voxel data using the mappings specified in the transfer function. The challenge in designing an appropriate transfer function is identifying which structural properties are important for the user and which relevant features in the data should be highlighted.

Imaging spectroscopy can be used to scan the structure of chemical elements on material surfaces. In contrast to a volume consisting of 3D points of scalar values, a spectral dataset consists of two spatial dimensions and a wavelength in the third dimension. Each scalar value in the volume is interpreted as the intensity on a wavelength at a 2D position on the surface of a material. A spectral volumetric dataset is often referred to by material scientists as a *multispectral data-cube*.

Since chemical elements have a unique and known spec-

tral profile, scientists can use spectroscopy to investigate which elements are present on the surface of a material if their spectral profile can be extracted. Unfortunately, extracting a spectral profile from a data-cube is a difficult task. First, the intensity at each point in the volume consists of contributions of the wavelengths of neighboring chemical elements at that position on the surface; i.e. the measured intensity at a voxel is a linear combination of wavelengths. A robust extraction method will be needed to factor the linear combination of wavelengths into the wavelengths of each chemical element. Second, spectra characterize themselves by different levels of scale in which peaks in the spectral profile can vary in order of magnitude. For example, consider the left plot of Figure 1. The sum of all spectral profiles in the data-cube is plotted, with on the x-axis the wavelength and on the y-axis the measured intensity. For visualization purposes, the right part of the plot is magnified by a factor of 467. Various large peaks can be seen in the left part of the spectrum, while very small peaks are in the right part of the spectrum. Both types of peaks are important in the analysis of the data. The right side of Figure 1 shows the spatial distribution of spectral peaks. The value of a pixel represents the sum of intensities at each wavelength at each position

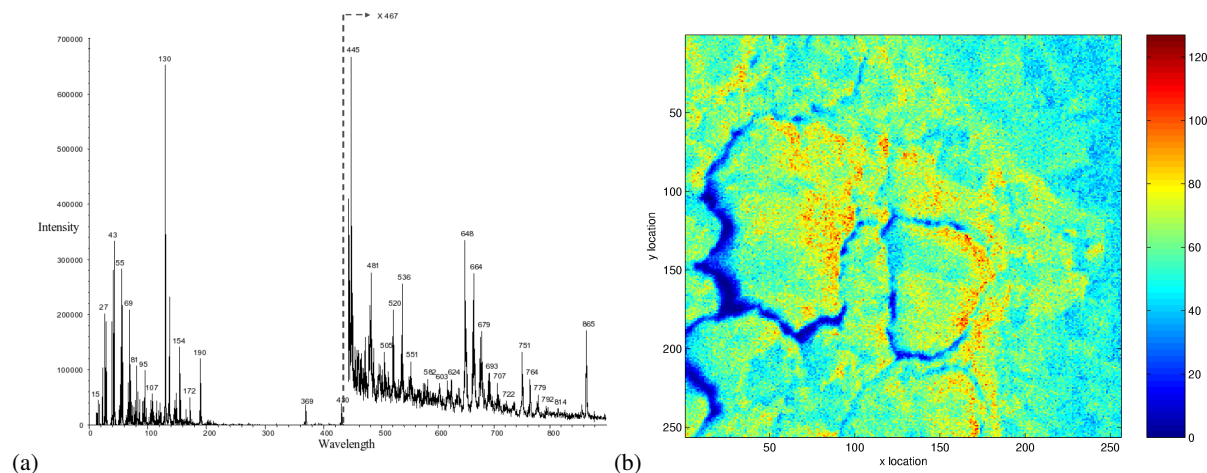


Figure 1: (a) A plot of the summation of all spectra in the data-cube from the cerebral ganglia of the pond snail. The right part of the plot is scaled by a factor of 447. (b) The spatial distribution of spectral profiles.

on the surface of the material. A color map is used to map intensity to a color.

Figure 1 is an example of how scientists use two side-by-side views to analyze the data in the data-cube. One view is the spectral view; it shows the spectral profile at all wavelengths. The second view is a spatial view; it shows the summation of the spectral profile at each position on the surface of the scanned material. It is our goal to create a data analysis environment with one integrated 3D-view to gain insight to the spatial distribution of features in the volume. This is very difficult using only the two dimensional views. This paper is the first step towards this goal; we report on how a transfer function can automatically be generated for a 3D-rendering of the data-cube.

1.1. Approach

Our approach uses Principal Component Analysis (PCA), for the generation of transfer functions of spectral data. PCA is a popular multivariate statistical method that is used to find patterns in data of high dimension. It is designed to capture the variance in a dataset in terms of principle components. Principle components are computed that define a projection that encapsulates the maximum amount of variation in a dataset and is orthogonal (and therefore uncorrelated) to the previous principle component of the same dataset. In effect, PCA is trying to reduce the dimensionality of the data to summarize the most important parts while simultaneously filtering out noise.

To use PCA, the data must be expressed in one or more variables, represented as a 2D data matrix. PCA transforms correlations in the data into a list of uncorrelated component vectors that is sorted by the amount of variability. In the case of a data-cube, PCA would be applied to orthogonal

axes of the data-cube, to find correlated spatial features at a particular wavelength, or correlated spectral features at a particular position of the surface. The resulting principal components divide the data in uncorrelated spatial features; i.e. uncorrelated chemical elements.

In our approach, however, we combine both spatial and spectral dimensions to form a 2D data matrix and apply PCA to this matrix. This results in finding correlated spatial and spectral features. This way, features are used to discriminate between boundaries of chemical elements on the material surface. We use the resulting principal component vectors to construct the transfer function. Features that have high variances can be made opaque to highlight features of interest, while features with low variances are made transparent.

We apply our method to data from two imaging spectroscopy techniques. 'Fourier transform infra-red' (FT-IR) imaging spectroscopy is a technique that employs the transmission of infra-red light through the surface of a material. Using the IR spectrum, chemical bonds and the molecular structure of organic compounds can be identified. Surface areas as small as 10-15 microns can be detected. For 'Time of flight secondary ions mass spectrometry' (TOF SIMS), a surface is bombarded with a primary beam of ions. This results in the emission of secondary ions, which are subsequently mass analyzed to generate surface mass spectra. The technique therefore provides very detailed elemental and chemical structure information. Using the latest TOF SIMS instruments, surface areas as small as 3-5 microns can be detected. While the resulting data from both techniques have different noise and scale properties, both can be analyzed similarly.

2. Related work

Early implementations of volume rendering [Lev88] using transfer functions [DCH88] are mostly applied to data resulting from CT scans. Many other areas of application could benefit from techniques developed to be used in these medical applications. For instance, Djurcilov et al. [DKLP02] uses techniques for visualizing 3D scalar datasets by combining uncertain information on top of environmental data. Uncertainty information is added to the classic volume rendering equation to highlight important features by adjusting opacity and color. One thing these datasets have in common: each data-point has a 3D spatial location in contrast with spectral data which has a 2D spatial location and a certain spectral location or wavelength. For this reason there are not many implementations of 3D visualizations for spectral data. One of the few examples of a DVR of a spectral data-cube is shown in “Visualization of Spectral Images” from [PvdH01] together with a representation with the use of iso-surfaces, which gave some unexpectedly good results using those datasets. Torson presents in [Tor89] a system for interactive analysis of 3D data-arrays, which he uses on data from spectrometer instruments for which conventional volume rendering and surface display techniques were not appropriate. He presents three reasons for not applying DVR techniques on data from imaging spectroscopy. First, the data values are not smoothly varying throughout the data-cube. Second, volume rendering cannot easily show small local data variations superimposed on broad overall variations and third, volume rendering provides only a qualitative view of the data. The system did not provide any interactive navigation tools for using volume rendering in a PC-based virtual reality like Fuhrmann et al. [FÖMH02] implemented using CT-data, but this would be the eventual goal for this presented procedure.

We try to solve the problems using DVR for spectroscopy data using PCA (see [WRR03]) to detect patterns in the data. Lasch et al. [LWMN98] already used PCA to detect patterns in FT-IR data images. Another application of PCA for pattern recognition in 3D datasets is, for example, the recognition of spatial-temporal patterns in Arctic sea ice concentration by Piwowar et al. [PDL01]. A closely related multi-variate image analysis algorithm is Independent Component Analysis (ICA). Muraki et al. [MNM00] apply ICA on multi-channel volume data from MRI scans to separate specific tissue characteristics e.g. water and fat. They train a radial basis function network with sample data from the visible female dataset to generate color transfer functions.

He et al. [HHKP96] also uses stochastic search techniques to generate transfer functions for data from MRI and CT scans, with better results than the approaches relying purely on the ‘trial and error’ of the human factor. This approach requires a minimum of computer aid compared to data-centric or image-centric approaches as described in [PLB*01]. Due to the complexity of the task and the intro-

duction of multi-dimensional transfer functions (for e.g. in [KD98], [KKH01b]) most research tends towards a semi-automatic approach in transfer function design for direct volume rendering of medical datasets. A minimum of user involvement is accomplished using direct manipulation widgets (see [KKH01a]) to create multi-dimensional transfer functions for specific datasets as in [VST*04].

Different approaches in creating appropriate transfer functions have to be considered in the relatively open area of using DVR to visualize the data-cubes from imaging spectroscopy. Existing multi-dimensional transfer functions do not handle the equally important high and low peaks in the spectral dimension very well. PCA is in use in identifying these different peaks in the resulting spectra and has already proven itself in the area of statistical pattern recognition.

3. Method

3.1. Data representation

PCA is traditionally performed on a 2D matrix with the samples of the dataset in one dimension and the different variables in the other dimension. The spectral data-cube has to be converted to such a matrix with preservation of the spectra and images as represented in Figure 2.

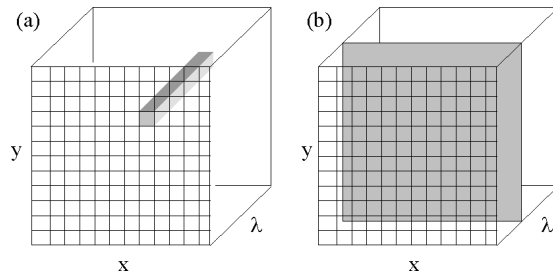


Figure 2: (a) A single spectrum and (b) a single image in the spectral data-cube.

A 2D matrix is constructed by unfolding each x by y image in the spatial dimension into an $x \times y$ -dimensional vector. This vector represents the spatial dimension at a particular wavelength. The 2D matrix, X , consists of each unfolded spatial vector, see Equation 1. Each row represents an image at a particular wavelength in the spectral dimension (denoted in Equation 1 as wavelengths $1, 2, \dots, \lambda$).

$$X = \begin{bmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,x \times y} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,x \times y} \\ \vdots & \vdots & \ddots & \vdots \\ d_{\lambda,1} & d_{\lambda,2} & \cdots & d_{\lambda,x \times y} \end{bmatrix} \quad (1)$$

The spectrum on a certain spatial location represents one column in matrix X . A slice of the data-cube at a particular wavelength represents one row.

The images in matrix X are treated as samples in the data which will result in several 'eigenimages' after applying PCA. The positive and negative peaks in the resulting eigenimages represent similarities and differences between the different spatial samples.

The spectra could also be regarded as samples to the PCA when matrix X is transposed, resulting in 'eigspectra'. The positive and negative peaks in the resulting eigenvectors represent similarities and differences between the different spectral samples.

Next, some pre-processing steps are taken to normalize the data in matrix X . First, we subtract each data value with the mean. This reduces the influence of extreme scalar data values. Second, scaling data values according to the variance. This removes big variations between values. Both rows and columns of the data matrix X are pre-processed in this way (using the formulas in 2). Consequently we can treat both rows and columns in the matrix as measurements when applying the PCA.

$$\begin{aligned}\tilde{X}_{rows} &= (X - \mu_{xy}) / \sigma_{xy} \\ X_{preprocessed} &= (\tilde{X}_{rows} - \mu_{\lambda}) / \sigma_{\lambda}\end{aligned}\quad (2)$$

3.2. Feature detection using PCA

The standard PCA algorithm is used to find orthogonal and normalized matrices for the spectral and spatial dimensions. These matrices are respectively denoted as P_{images} and $P_{spectra}$. The rows in P_{images} are the eigenvectors of the spatial dimension in the matrix $X_{preprocessed} \cdot X_{preprocessed}^T \cdot P_{images}$ is a rank sorted $\lambda \times \lambda$ sized matrix and $P_{spectra}$ is rank sorted $xy \times xy$ sized matrix.

Equation 3, shows how P_{images} is used to find Y_{images} , the by P_{images} projected data as a linear combination of the new bases of $X_{preprocessed}$. S_Y is the diagonalized covariance matrix, with the i^{th} diagonal value of S_Y being the variance of $X_{preprocessed}$ along the i^{th} principle component.

$$\begin{aligned}Y_{images} &= P_{images} \cdot X_{preprocessed} \\ \text{such that } S_Y &\equiv \frac{1}{xy-1} Y_{images} \cdot Y_{images}^T \text{ is diagonalized}\end{aligned}\quad (3)$$

Y_{images} is a $\lambda \times xy$ sized matrix. Since the principle components in P_{images} are sorted in decreasing variance, the highest contribution to the spatial dimension are the first rows in Y_{images} . The last rows with the lower associated variances are more likely to represent noise. The rows of Y_{images} are called *spatial score vectors*.

A similar approach is used to find the much larger $xy \times xy$ sized matrix, $Y_{spectra}$, the by $P_{spectra}$ projected data as a linear combination of the new basis of $X_{preprocessed}^T$. S_Y is the

diagonalized covariance matrix, with the i^{th} diagonal value of S_Y being the variance of X along the i^{th} principle component of $P_{spectra}$.

$$\begin{aligned}Y_{spectra} &= P_{spectra} \cdot X_{preprocessed}^T \\ \text{such that } S_Y &\equiv \frac{1}{\lambda-1} Y_{spectra} \cdot Y_{spectra}^T \text{ is diagonalized}\end{aligned}\quad (4)$$

The principle components with the highest contribution to the spectral dimension are the first rows in the matrix. The last rows with the lower associated variances are more likely to represent noise instead of interesting spectral features.

3.3. Transfer function generation

In the previous section we described how the principal components and score vectors are computed. These are used to find spectral or spatial features in a 3D spectral volume. For the visualization of these features, opacity of the transfer function is used. Multiple opacity functions are used for different features to isolate them from other areas.

The eigenimages and eigspectra matrices, P_{images} and $P_{spectra}$, are used to compute two new matrices, Y_{images} and $Y_{spectra}$, which project the original data into a new bases. To generate the opacity function of areas with the highest variances in the data, an addition of the first score vector with the highest spatial variance and the first score vector with the highest spectral variance is used. The motivation is that features with the highest variance in the spatial and spectral dimensions are captured in one single opacity map.

Equations 5 and 6 describe this more formally. The score vector matrices, Y_{images} and $Y_{spectra}$, are rewritten as a set of vectors. Y_{images} is the projected spatial matrix and is written as a sorted list of vectors. $Y_{spectra}^T$ is the transpose of projected spectra matrix, and is written as a sorted list of vectors. The resulting vectors are combined into one opacity transfer function. For example, Equation 6 shows how the opacity map of the first score vectors is derived.

Here, Y_{images} is a $\lambda \times xy$ size matrix and $Y_{spectra}$ is a $xy \times \lambda$ matrix, which is converted to a $\lambda \times xy$ matrix by taking the transpose. Hence, O^1 is a $\lambda \times xy$ sized matrix.

$$Y_{images} = \begin{bmatrix} i_1 \\ i_2 \\ \vdots \\ i_{\lambda} \end{bmatrix} \text{ and } Y_{spectra} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{xy} \end{bmatrix}\quad (5)$$

$$O^1 = \begin{bmatrix} i_1 \\ i_1 \\ \vdots \\ i_1 \end{bmatrix} + \begin{bmatrix} s_1 \\ s_1 \\ \vdots \\ s_1 \end{bmatrix}^T\quad (6)$$

The 3D points with the highest positive and negative values in O^1 are assigned to high alpha values. All regions in the volumetric data that contribute to this first principal component are made opaque using this 3D transparency map.

Similarly, the opacity maps of the second, third, etc. score vectors can be generated. Different mappings could be combined to display similarities or differences of multiple features in original data.

4. Applications

Our method is applied on two examples of spectral recordings. First, we describe the results of our method when applied on a dataset created with the TOF SIMS technique. It is a data set of a small section of the anterior lobe of the cerebral ganglia of the pond snail, *Lymnaea Stagnalis*. Our second example is a visualization of the brain ventricle of a mouse resulting data from FT-IR spectroscopy.

4.1. The small brains of a snail

A high spectral and spatial resolution can be obtained using the TOF SIMS. In this example atomic and molecular structures in a data set can be identified. The boundaries of different cells can be visualized when TOF SIMS is applied on a slice of the brain of the pond snail. Figure 1 already showed the resulting spectra and images from this spectral scan. Some obvious features are highlighted when our method is applied on this dataset. The first two spectral score vectors are shown in diagram a and c of Figure 3.

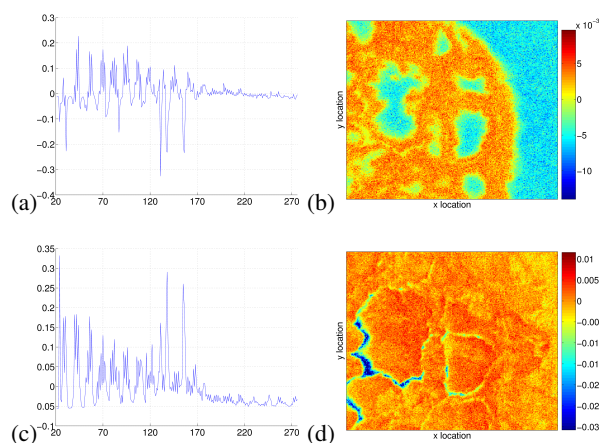


Figure 3: (a-c) The first two score vectors with (b-d) the accompanying eigenimages as a result from the PCA.

The three highest peaks in (a) represent the positive contributions in the spectral dimension of the component that isolates the largest amount of data. Figure 3b gives the corresponding spatial distribution of the material in which the cells in the data set are embedded. The second score vector

in diagram (c) has isolated a large negative peak. This peak corresponds with the red areas in the image of Figure 3d that can be identified as some larger cavities between the different cells. Other components that result from the PCA also highlight certain areas within or between cells that contain different organic compounds.

4.2. The bigger brains of a mouse

The second example is a slice of the central brain ventricle of a mouse. FT-IR spectroscopy is applied to identify different chemical functional groups. The resulting spectral and spatial components are combined in different 3D maps of which the second, third and fourth are displayed in Figure 4.

The long axes represents the spectral dimension that ends in the front on wavelength of 4000cm^{-1} . The first component mainly highlights the differences between the uninteresting regions completely filled. The red and orange regions of the components represent positive correlations in the data in contrast with the blue regions. Both regions could be of interest for the analysis of the identification the functional groups. The isolated blue region at the back in diagram (a) on wavelength 1550cm^{-1} , represents the location of amide groups, whereas the two blue regions in the front on wavelength 3300cm^{-1} and 3400cm^{-1} represent hydroxy groups and amino groups. Different regions are clearly distinguished in the resulting 3D maps. These maps can be used as an opacity map on top of the original data as shown in diagram (a) and (b) of Figure 5.

The same volumetric data can be loaded in VolView, a visualization package from Kitware. This package offers many tools to interactively create an appropriate transfer function based on the “trial and error” method with an initial estimation of color and opacity transfer function based on the histogram of the data. This initial guess for an appropriate transfer function is shown in Figure 5c. This package cannot distinguish between the spatial and spectral dimensions in the spectral data because it uses the scalar values the same way in all three dimensions. VolView cannot differentiate between small but important differences between values in the spectral dimension when they are dominated by large peaks that are present in other regions in the data-cube.

5. Discussion

The current practice in the analysis of spectral data is illustrated in Figure 1. Two dimensional plots, one for the spectral and one for the spatial information, are used to view the data set. In this paper, we have introduced a method to view spectral data in three dimensions. The three dimensional view is used to gain insight into the spatial distribution of features in the volume, which is very difficult to do using only the two dimensional views. We have discussed how a transfer function can be generated using PCA. The contribution is that PCA is applied in both the dimensions of the

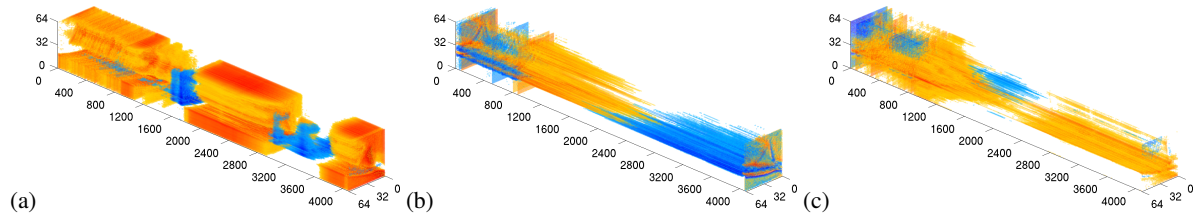


Figure 4: The (a) second, (b) third and (c) fourth component mappings with the negative contributions in blue and positive contributions in red.

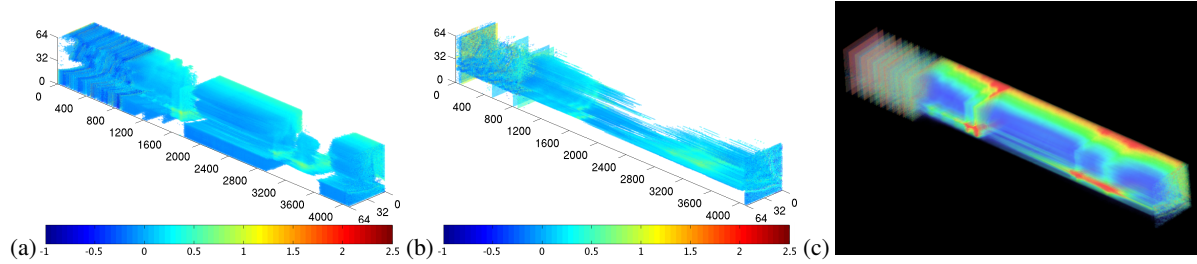


Figure 5: (a-b) Two resulting component mappings applied on the original data compared with (c) the VolView representation of the same dataset.

individual images, as well as in the dimensions of the spectra of all images.

Using PCA in both dimensions of the data cube allows us to address the two major problems that were mentioned in the introduction. First, intensity at each point in the data consists of contributions of many spectra. Using PCA in the spectral dimension, results in the principle components of a number of spectra. Second, spectra characterize themselves by different levels of scale. PCA is a technique that computes variances. Hence, the technique will find variances between spectra with very large peaks and also between spectra with low peaks.

We now discuss some pros/cons of the proposed method:

- Although the first principal component is used to highlight the data regions that of which have the highest variances, this does not necessarily mean the most interesting feature is captured by the first component. For example, a principle component could contain high variances in the spectral dimension and very low variances in the spatial dimension. As a consequence, each opacity function should still be manually checked by the user.
- Component vectors can contain negative values, although the original data does not. These negative peaks can be just as important as the positive, but it is uncertain how the composite opacity map is affected when they have a positive spectral but negative spatial contribution.
- The method normalizes the data as much as possible and reduce variances in both dimensions (see Equation 2). This is difficult to realize in both dimensions without losing the direct relation between spectral and spatial compo-

nents. There can be some outliers that cannot completely be filtered out in the pre-processing steps when their values are too deviant to successfully auto-scale the data and lose the extreme values. An alternative data-scaling technique may be used to remove these outliers automatically using a threshold function to filter out extreme values.

- The solution of $Y_{spectra}$ can be used to compute the matrix P_{images} as

$$P_{images} = \frac{Y_{spectra}}{\sqrt{\sum Y_{spectra}^2}} \quad (7)$$

This optimization has a much lower computational cost than was given in Equation 4 since $\lambda \ll x \times y$.

- As mentioned above, PCA is used to analyze spectra and highlight contrasting features in images. Even small differences between spectra and images can be detected when they are correctly pre-processed. Hence, the method can also be used to detect noise in both spectral and spatial dimensions. The 'least principal' components will contain most of the noise present in the data.
- The main advantage of using our method to detect spectral and spatial features is the direct linkage between both dimensions by using the result in one dimension to calculate the other (see Equation 6). It is not possible to automatically link components using separate analyzes by treating the spectra or images as dimension with measurements.

5.1. Future work

Our method is just the first step in creating a tool for the analysis of spectral data using direct volume rendering. In

the future, 3D separation and clustering algorithms will be incorporated to improve the definition of the opacity function. Another improvement would be the automatic identification of features by adding additional information about spectral peaks. Component vectors could be matched using this database of score vectors to label the different volumetric regions in the visualization. Storage of data sets, their features and classifications could eventually evolve into an integrated system for feature recognition and analysis. Finally, the method will be applied to other multidimensional scientific datasets which lend themselves for finding high dimensional patterns.

6. Acknowledgements

We thank Prof.Dr.Ing. R.M.A. Heeren and Drs. A.F.M. Altelaar from the Institute for Atomic and Molecular Physics (AMOLF) who prepared and supplied the spectral datasets that were used for the given examples.

This work was carried out in the context of the Virtual Laboratory for e-Science project (www.vl-e.nl). This project is supported by a BSIK grant from the Dutch Ministry of Education, Culture and Science (OC&W) and is part of the ICT innovation program of the Ministry of Economic Affairs (EZ).

References

- [DCH88] DREBIN R. A., CARPENTER L., HANRAHAN H.: Volume rendering. In *Computer Graphics (SIGGRAPH '88 Proceedings)* (Aug. 1988), Dill J., (Ed.), vol. 22, pp. 65–74. 3
- [DKLP02] DJURCILOV S., KIM K., LERMUSIAUX P., PANG A.: Visualizing scalar volumetric data with uncertainty. *Computers and Graphics* 26, 2 (Apr. 2002), 239–248. 3
- [FÖMH02] FUHRMANN A., ÖZER B., MROZ L., HAUSER H.: *VR2 Interactive Volume Rendering using PC-based Virtual Reality*. Tech. rep., Mar. 2002. 3
- [HHKP96] HE T., HONG L., KAUFMAN A., PFISTER H.: Generation of transfer functions with stochastic search techniques. In *IEEE Visualization* (1996), Yagel R., Nielson G. M., (Eds.), pp. 227–234. 3
- [KD98] KINDLMANN G., DURKIN J.: Semi-automatic generation of transfer functions for direct volume rendering. In *IEEE Symposium on Volume Visualization* (1998), pp. 79–86. 3
- [KKH01a] KNISS J., KINDLMANN G., HANSEN C.: Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *IEEE Visualization* (Oct. 2001), pp. 255–262. 3
- [KKH01b] KNISS J., KINDLMANN G., HANSEN C.: Multi-dimensional transfer functions for interactive volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 8, 4 (July 2001), 270–285. 3
- [Lev88] LEVOY M.: Display of surfaces from volume data. *IEEE Comput. Graph. Appl.* 8, 3 (1988), 29–37. 3
- [LWMN98] LASCH P., WÄSCHE W., MCCARTHY W., NAUMANN D.: Imaging of human colon carcinoma thin sections by ft-ir microspectroscopy. In *Infrared Spectroscopy: New Tool in Medicine* (1998), vol. 3257, pp. 187–197. 3
- [MNK00] MURAKI S., NAKAI T., KITA Y.: Basic research for coloring multichannel mri data. In *Proceedings of the 11th IEEE Visualization 2000 Conference* (2000), IEEE Computer Society, pp. 187–194. 3
- [PDL01] PIWOWAR J. M., DERKSEN C. P., LEDREW E. F.: Principal components analysis of the variability of northern hemisphere sea ice concentrations: 1979-1999. In *Proceedings, 23rd Canadian Symposium on Remote Sensing / 10e Congrès de L'Association québécoise de télédétection, Ste.-Foy PQ* (Aug. 2001), pp. 619–628. 3
- [PLB*01] PFISTER H., LORENSEN W., BAJAJ C., KINDLMANN G., SCHROEDER W., SOBEIERAJSKI AVILA L., MARTIN K., MACHIRAJU R., LEE J.: The transfer function bake-off. In *IEEE Computer Graphics and Applications* (May 2001), pp. 16–22. 3
- [PvdH01] POLDER G., VAN DER HEIJDEN G.: Visualization of spectral images, invisualization and optimization techniques. In *SPIE* (2001), Censor Y., Ding M., (Eds.), vol. 4553, pp. 132–137. 3
- [Tor89] TORSON J. M.: Interactive image cube visualization and analysis. In *Proceedings of the 1989 Chapel Hill workshop on Volume visualization* (1989), ACM Press, pp. 33–38. 3
- [VST*04] VEGA HIGUERA F., SAUBER N., TOMANDL B., NIMSKY C., GREINER G., HASTREITER P.: Automatic adjustment of bidimensional transfer functions for direct volume visualization of intracranial aneurysms. In *Medical Imaging 2004: Visualization, Image-Guided Procedures and Display* (June 2004), Galloway Robert L. J., (Ed.), vol. 5367, pp. 275–284. 3
- [WRR03] WALL M., RECHTSTEINER A., ROCHA L.: Singular value decomposition and principal component analysis. *A Practical Approach to Microarray Data Analysis* (2003), 91–109. 3