iSpectra Manual

iSpectra version 1.02

Christian Liebske

Institute of Geochemistry and Petrology
Department of Earth Sciences, ETH Zurich
Switzerland

February 20, 2015

Abstract

iSpectra[1] is an open toolbox for the analysis of spectral images (SI's) recorded on SEM EDS systems. The aim of iSpectra is to assign pixels with similar spectral content to phases, thus, creating true phase maps, rather than simple, but more commonly used elemental distribution maps. The strategy used in iSpectra for pixel-to-phase assignment is similar to the proprietary **PARC**[2] method described by *van Hoek et al.*[3] and adds various standard image analysis tools for optimisation of the resulting phase map. iSpectra currently supports spectral images saved in Lispix format and exports cumulative EDS phase spectra in standard EMAS file format. Pixels-to-phase assignments are carried out using an automated, threshold based algorithm to group pixels with identical elemental budgets into groups, followed by merging groups and re-assignment of pixels using count density plots (e.g. two dimensional count intensity histograms) to achieve a chemically and texturally plausible phase map. Assigned phases can be refined using various image morphology operations to account for overlapping excitation volumes across pixels located at phase boundaries. iSpectra supports batch processing and allows pixel-to-phase assignments to be applied to an unlimited amount of spectral images, thus enabling phase mapping of e.g. entire petrographic thin sections, if the SI acquisition time is affordable.

iSpectra is written in *Wavemetrics' Igor Pro* and consists of over one hundred user defined functions, which are easily accessible via a graphical user interface, therefore, making it usable without in-depth knowledge of the host software.

When using results obtained with *iSpectra* in publications or presentations, please give credit to *Liebske (submitted)* and *van Hoek et al. (2011)*.

[1] Liebske, C. (submitted) iSpectra: An Open Source Toolbox for the Analysis of Spectral Images recorded on SEM-EDS systems. Microscopy and Microanalysis

[2] **Ph**ase **R**ecognition and **C**haracterisation

[3] van Hoek, C.J.G., de Roo, M., van der Veer, G., van der Laan, S.R. (2011) A SEM-EDS Study of Cultural Heritage Objects with Interpretation of Constituents and Their Distribution Using PARC Data Analysis. Microscopy and Microanalysis, 17 (5), 656-660

Contents

1 System Requirements

iSpectra requires Igor Pro 6.1 or later versions, running either on Windows XP or Mac OS X 10.4 (or later versions of both systems). The most recent version of Igor Pro can be downloaded from `http://wavemetrics.com/`. A fully functional test version of Igor Pro can be used for 30 days free of charge. iSpectra can be downloaded from: `http://www.igorexchange.com/project/iSpectra`.

2 Installation

iSpectra comes as a single Igor procedure file (*iSpectra.ipf*), which can be integrated in the existing *Igor Pro 6 User Files* folder hierarchy to make iSpectra available in every Igor experiment. The default location of this folder is on:

- Macintosh: /Users/UserName/Documents/WaveMetrics/Igor Pro 6 User Files
- Windows XP: C:\Documents and Settings\UserName\My Documents\WaveMetrics\Igor Pro 6 User Files
- Windows 7: C:\Users\UserName\My Documents\WaveMetrics\Igor Pro 6 User Files

Place *iSpectra.ipf* or an alias/shortcut of it into the subfolder *Igor Procedures*. iSpectra will then automatically be loaded and added to Igor's top menu each time the program starts. Alternatively, the *.ipf file can simply be loaded each time when it is required, if iSpectra is not supposed to be active automatically.

iSpectra is designed to be operable with minimal knowledge of Igor Pro. However, Igor's capabilities in image processing, data analysis, visualisation and programming allow a more experienced user to extend iSpectra to personal needs or to further process iSpectra's output. If you are new to Igor you will benefit from executing the following lines from *Igor's* command line and studying the associated help files:

```
DisplayHelpTopic "Getting Started"
DisplayHelpTopic "Waves"
DisplayHelpTopic "Data Folders"
DisplayHelpTopic "User-defined Functions".
```

It is absolutely vital to use Igor's data browser, especially when running *iSpectra.ipf*, e.g.

```
DisplayHelpTopic "Data Browser".
```

3 Acquiring Spectral Images

To obtain optimal results with *iSpectra*, the acquired spectral images should fulfil certain requirements:

1) *Sufficient counting statistics.* Average maximum intensities on the major elements should ideally be in excess of 50 counts per pixel. Excellent phase maps can usually be generated on SI's with around 100 counts on the maximum peak on each pixel (as in the example data set

shown in this manual), which can be achieved on modern Si-drift detector systems in reasonable time scales. Significantly higher counts are mostly not required.

2) *Appropriate image resolution.* For most accurate pixel-to-phase assignments, the SI frame resolution and microscope magnification should be set, such that the real size of one image pixel matches the size scale of the expected interaction volume of the electron beam with the specimen. If one image pixel is significantly smaller than the interaction volume when projected onto to the sample surface, then several pixels will contain the same spectral information, which is inefficient use of acquisition time. For more details, see section 8.1. Due to memory restrictions of the 32 bit versions of Igor "out of memory" problems may arise when working with spectral images larger than XY image dimensions of 512 by 384 pixels at 2048 channels in 16 bit format.

4 Spectral Image (SI) Import

iSpectra presently (Version 1.02) supports spectral images saved in Lispix format (see `http://www.nist.gov/lispix` for more details). For NORAN EDS system the conversion program *SIconvert.exe* can be purchased from Thermo Fisher to generate such formats. Spectral images recorded on other systems may be loaded after adding appropriate import routines to iSpectra. Lispix datasets consist of two files; a binary *.raw file containing all spectra of the spectral image and a *.rpl text file (raw parameter list) containing information on the spectral image dimensions and data type. Datasets can be loaded using the **Load Raw File** menu point (note, that the *.rpl file of the Lispix file pair needs to be selected in the import dialogue) . The imported SI will be stored as 3D wave *root:cube*. At the end of import, *iSpectra* will ask for some SI acquisition properties, acceleration voltage, time per frame and number of frames and the magnification with which the SI was acquired. (Fig. 1).

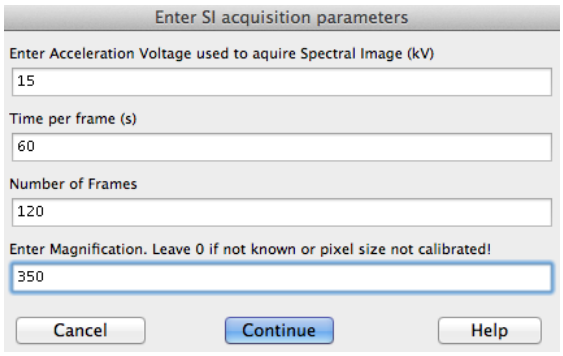


Figure 1: Parameters to be specified during import.

Specifying the magnification at which the spectral image was recorded allows size scaling of the x-y dimensions of the SI to micrometers to be applied. The size scaling implemented in *iSpectra* is based on an empirically relation between the "magnification" of the microscope (e.g. 2000 times; x 2000) and the image field-of-view width, expressed as:

$$width_{field-of-view} = 10^{(a*\log(Magnification)+b)} \quad (1)$$

where a and b are adjustable parameters which are stored as constants in *iSpectra.ipf* (see

Appendix B for details). Such parameters can be determined from e.g imaging size calibration grids at different magnifications and deriving the image width from it, followed by curve fitting. Note, that such a calibration is valid at a fixed *Working Distance*, which, however, is usually kept constant for use with EDS. If the parameters a and b are not determined then the corresponding input field in Fig. 1 should be left at zero to apply no wave scaling.

Next *iSpectra* will ask to clip unused channels. Count intensities are stored in channels with constant increments of energy. The default energy increment is 0.01 keV (see appendix B for details). However, if a spectral image contains e.g. 2048 channels but was recorded at 15 kV, then 548 channels or about 25% of the data set remain unused. Clipping such channel saves system memory and improves iSpectra's performance. (Fig. 2).

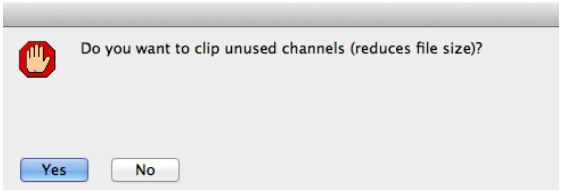


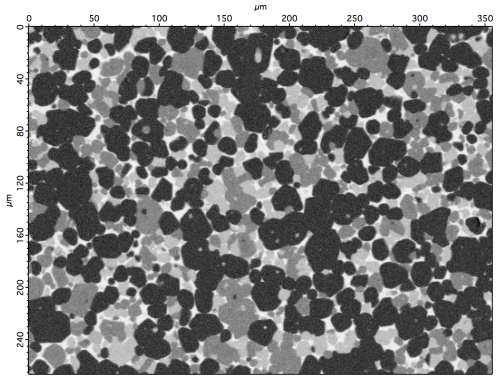
Figure 2: Unused channels of the spectral images should be clipped in order to save system memory.

Data import is finished by automatically displaying a grey scale image (see also section 5) of the SI and the **Model Setup** panel (see Fig 3). Clicking at any pixel of the image will cause the EDS spectrum at that pixel location to be displayed. Peak labelling is not added to the spectrum by default but can be activated using the **Add Line References to Top Trace** menu point.

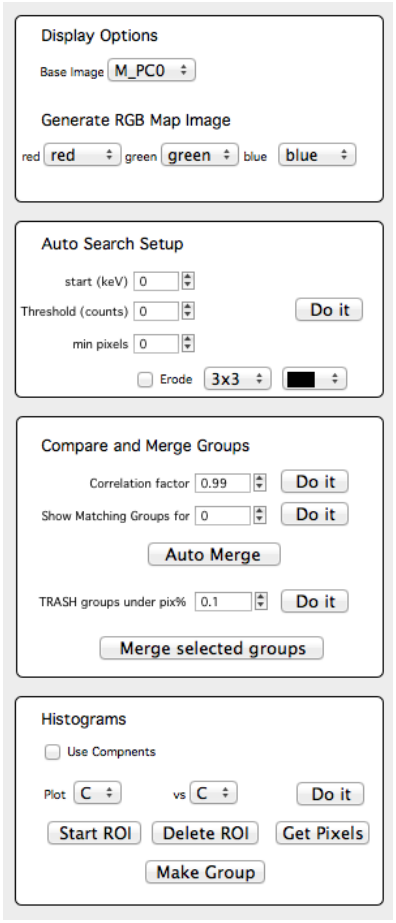
4.1 Element Energies, Maximum Pixel Spectrum and Elemental Intensity Maps

On SI import elemental intensity maps of a *predefined* list of elements will be extracted and stored as 2D waves under *root:Intensities*. Note, that most subsequent actions to assign pixels to phases, such as automated group assignments (see section 7.1) or count density plots (section 7.5), are based on these 2D intensity data waves. Elements and their x-ray emission energies are stored in the wave *root:packages:iSpectra:LineRef*. Energies are stored as numeric values while element symbols are defined as dimension labels. The content of the wave *LineRef* can be displayed, altered and saved or a custom wave can be loaded using the **Element Energies** menu points. If the existing wave *LineRef* is modified by adding an element, it should finally be sorted after energy which can be done using the menu point **Sort Line Reference Wave**. This will also cause re-extraction of the elemental intensity maps according to the modified list. The default list of elements, that is, the content of the automatically initialised wave LineRef, can be changed by modifying the string constants *energies* and *elements* of *iSpectra.ipf* (see Appendix B for details).

Working with a predefined lists of elements implies that count peaks of elemental energies that were not included in the initial element list are invisible in the following pixel-to-phase assignment (see also discussion). To identify undefined energies, a "maximum pixel spectrum" method according to Bright and Newbury (2004) is implemented, such that a spectrum is generated, which consists of the maximum x-ray counts for each energy channel found in the spectral image. Comparing this spectrum with the initially defined elemental energies easily reveals



(a) Principle component image



(b) Main Panel

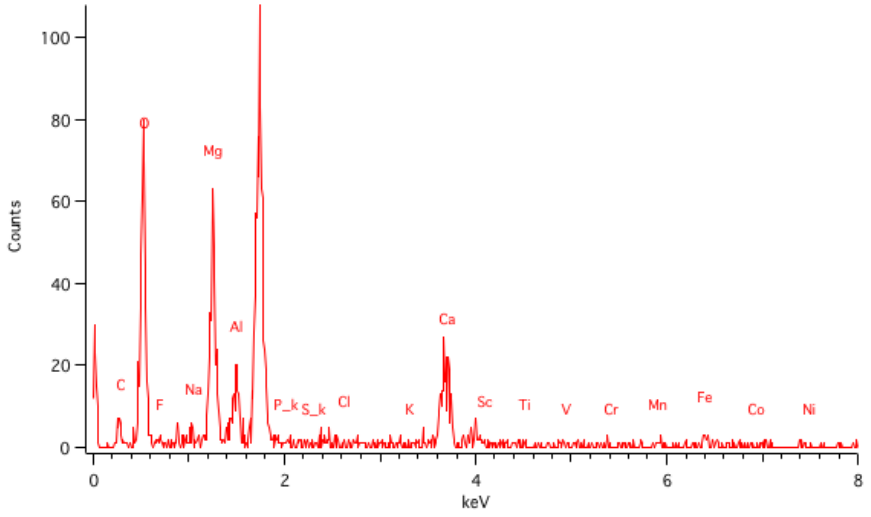


Figure 3: After import two windows will appear. The principle component image (a, see section 5 for details) and the iSpectra main panel (b) will be displayed automatically. Clicking on the image will show the EDS spectrum at this pixel location in an additional window (c).

missing emission energies. The maximum pixel spectrum can be generated using the **Show Maximum Pixel Spectrum** sub menu point under **Element Energies**.

4.1.1 Re-extracting Element Maps

iSpectra does not perform peak fitting, and will thus not deconvolute overlapping peaks, but will only extract averaged counts at a given x-ray emission energy to generate the 2D intensity maps mentioned above. By default *iSpectra* will average three channels for a particular element, that is, the theoretical channel at peak energy including the previous and next channel. Averaged counts are rounded to integer values. Averaging channels results in smoother elemental intensity distribution maps. Elemental maps can be re-extracted by choosing a different number of channels to change data smoothing. The number of channels should be an odd integer.

5 Image Display Options

Spectral imaging data sets do usually not contain any embedded image information from any of the SEMs imaging detectors, such as e.g. a back scattered or secondary electron images. However, *iSpectra* provides several possibilities to represent the scanned specimen surface to visualise chemical variability. At the end of data import the image matrix *root:packages:iSpectra:M_BaseImage* is being displayed that can contain values of one of the following options. The nature of the base image can interactively be chosen using the **Base Image** popup menu on the main panel. Note that *M_BaseImage* literally serves as base image on which groups of assigned pixels will be shown as coloured overlay. For the purpose of displaying an image all of the following image matrices are scaled to 16 bit unsigned integer values, thus whole numbers ranging from 0-65535. The available options for displaying the dataset as image matrix are summarised in Figure 4.

5.1 Principle Component Maps

The extracted element maps as described above form the basis for principle component analysis (PCA), resulting in a number of "principle component maps" (PCM's), which are generated as follows. First, all 2D element maps are deployed into 1D arrays in a column priority order. Such 1D arrays are then concatenated into a $m \times n$ matrix $\boldsymbol{A}$, where the row m corresponds to the number of pixels in the SI and column n refers the number of specified elements. Next, the arithmetic mean is subtracted from each column n of $\boldsymbol{A}$ to form a data matrix $\boldsymbol{D}$. Singular value decomposition is used to decompose matrix $\boldsymbol{D}$ into n positive eigenvalues (the squares of the singular values) and a $n \times n$ eigenvector matrix $\boldsymbol{C}$. Multiplying $\boldsymbol{C}$ with the transpose of $\boldsymbol{D}$ results in a $n \times m$ matrix $\boldsymbol{P}$. Each row of $\boldsymbol{P}$ represents a projection of each pixel onto an eigenvector, or a principle component. Extracting a row from $\boldsymbol{P}$ and re-dimensioning it into 2D results in an image matrix which resembles the variances of the different element counts in the SI and provides a way for visualising chemical variability. The rows in $\boldsymbol{P}$ are ranked in order of decreasing importance, thus, the first row is associated with the first and largest eigenvalue, which, when divided by the sum of all eigenvalues, equals a fraction of the total variance within matrix $\boldsymbol{D}$. iSpectra will extract as many PCM's until a certain cumulative variance percentage, the default value is larger 99 %, is reached. The number of extracted maps and the total cumulative variance percentage associated with these maps are written into the Igor's history window.

The default display after data import is the PCM that is associated with the largest variance of the dataset, M_PC0. Inspection of PCM's associated with low variance can be useful to locate chemical features at minor or trace abundance levels.

5.2 Sum- and RGB images

A common representation from SI data sets is the so-called sum- or summation-image. Each element of a sum image matrix consists of the the sum of all x-ray counts in corresponding pixel spectrum. Such an image, stored as *root:M_SumImage* is generated automatically during data import. Note, that the sum image usually is not well suited to display chemical variability. Another possibility, which is very well suited to visualise chemical features are RGB map images where each of the aforementioned element maps can be used as red, green or blue channel of an RGB colour image. Such images can be generated using the popup menus under the **Generate RGB Map Image** on the *iSpectra's* **Main Panel**.

5.3 Loading Other Images

An additional SEM image, as it may be recorded before or during SI acquisition on any of the available detectors of the SEM, can be loaded using the **Load SEM TIF Image** dialog. The import dialog will suggest to load images with the extension *.tif or *.tiff. However, such SEM images may have system specific extension (as e.g. *.siref or *.sitif in case of the NORAN system), and the import file type options should be set to *All Files* in this case. If the loaded image has a different XY pixel resolution compared to the SI, it will be resized to match the latter. This image will be stored as 2D matrix *root:M_TifImage*.

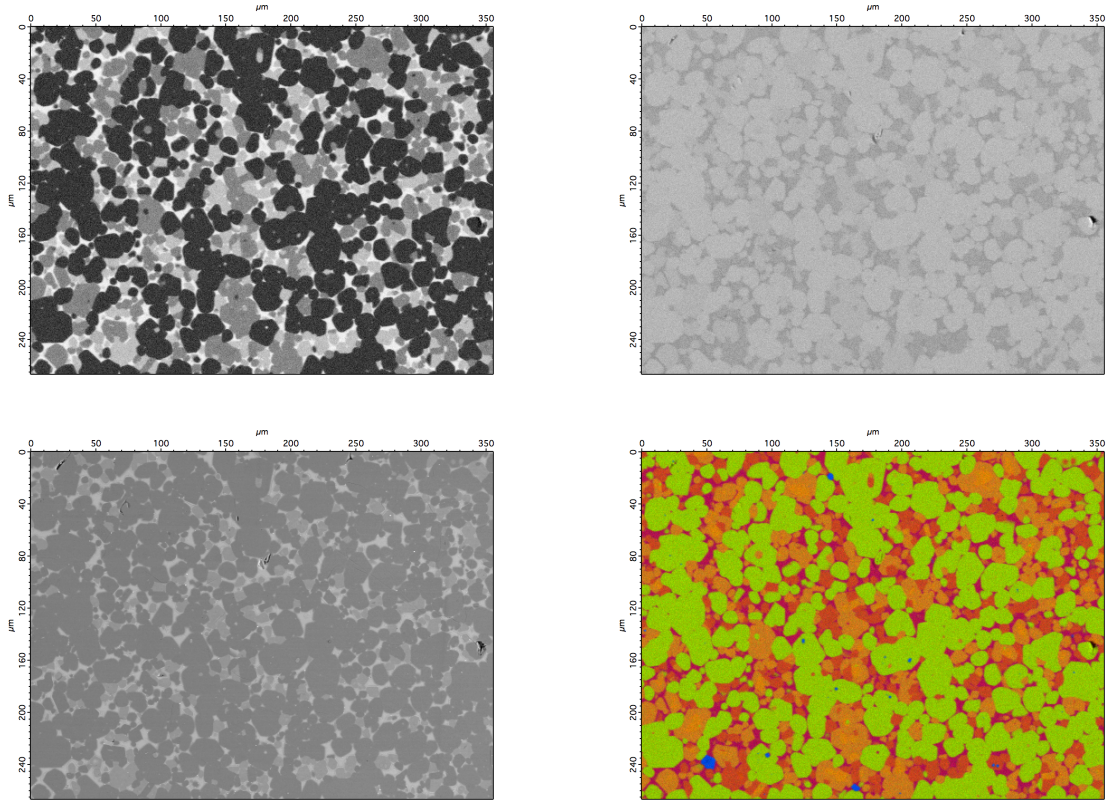


Figure 4: Displaying options for the "base image". Top-left: the default display, the principle component image associated with the larges variance of matrix $\boldsymbol{D}$ *(M_PC0). Top-right: sum-image (M_SumImage). Bottom-left: Manually imported image (M_TifImage), which was recorded before SI acquisition (here a back-scattered electron image). Bottom-right: RGB image (M_RGBImage), where Si, Mg, and Al are the red, green and blue channels, respectively. Any elemental map can be used for any channel and can be changed on the main panel.*

6 Background Subtraction of Pixel Spectra

Pixel spectra can optionally be background subtracted by applying a simple top-hat filter as described e.g. by Statham (1977) in his equation 15:

$$y_i' = \frac{1}{2M+1}\sum_{j=i-M}^{j=i+M} y_i - \frac{1}{2N}\left(\sum_{j=i-M-N}^{j=i-M-1} y_i + \sum_{j=i+M+1}^{j=i+M+N} y_i\right) \quad (2)$$

where y_i' is the background stripped count value from raw count y at channel i, and M and N are numbers of channels that define over which width the filter is active. To describe the working principle of the top-hat filter I quote from Statham (1977):

"...the "top-hat" filter essentially corresponds to taking the average over $U = 2M + 1$ channels about the channel of interest and subtracting the mean of averages over N channels on either side of this central region; the total "support" of the filter is thus $W = 2M + 1 + 2N$. The filter produces a result similar to the negative of the smoothed second derivative of the data so a single peak is transformed into a central positive lobe with negative lobes on either side and a background which is linear is completely suppressed."

Note that negative contribution to the filtered spectrum in iSpectra's implementation are set to zero. The optimum width of the channel depends on width of the element peak of interest and is therefore a function of peak resolution and thus a function of the EDS detector and acquisition properties, e.g. the time constant of the pulse processor. Additionally the peak width is a function of its emission energies, such that peaks become broader with increasing energy. For a given peak width (expressed as the full width at (peak) half maximum, FWHM), the following recommendation is given by Statham (1977):

"Since the parameters will depend directly on the peak fwhm, suitable values in general would be integers which are close to satisfying U = FWHM, W = 2 X (FWHM)."

The filter width settings for the top-hat filter algorithm in *iSpectra* can be accessed from the menu point **"Background - Show Top Hat Filter Settings**, which opens the interactive graph shown in Figure 5. Because the optimum FWHM for a given element is dependent its emission energy, *iSpectra* allows to specify two FHWM values (in keV) for two different energies. The optimum parameters M and N in equation 2 for an energy of interest (rounded to integers) are calculated from a line fit to the specified FWHM values for the given energies. The user can specify a FWHM for an element at low energy and a FWHM for an element at higher energy. In the example presented in Figure 5, the low energy value is approximated for oxygen (FWHM = 0.05 keV) and the high energy value for Cr (FWHM = 0.11 keV). Clicking the **Set** button, followed by **Filter Spectrum** filters the spectrum on the lower half of the image and shows the result on the upper half. Clicking **Apply To Cube** will apply the filter with its present settings to each pixel of the spectral image.

The filtered SI is stored under *root:Cube_BKG_rmvd*. This action also causes re-extraction of the element maps according to section 4.1. Clicking on any pixel of the base image will now show a background-stripped spectrum instead of a raw-spectrum.

Working on background subtracted spectra is advantageous because the count threshold can be set low while still largely avoiding to detect noise, specifically in the 1-3 keV range, where the continuum may be relatively high compared to peaks at higher energies. However, background

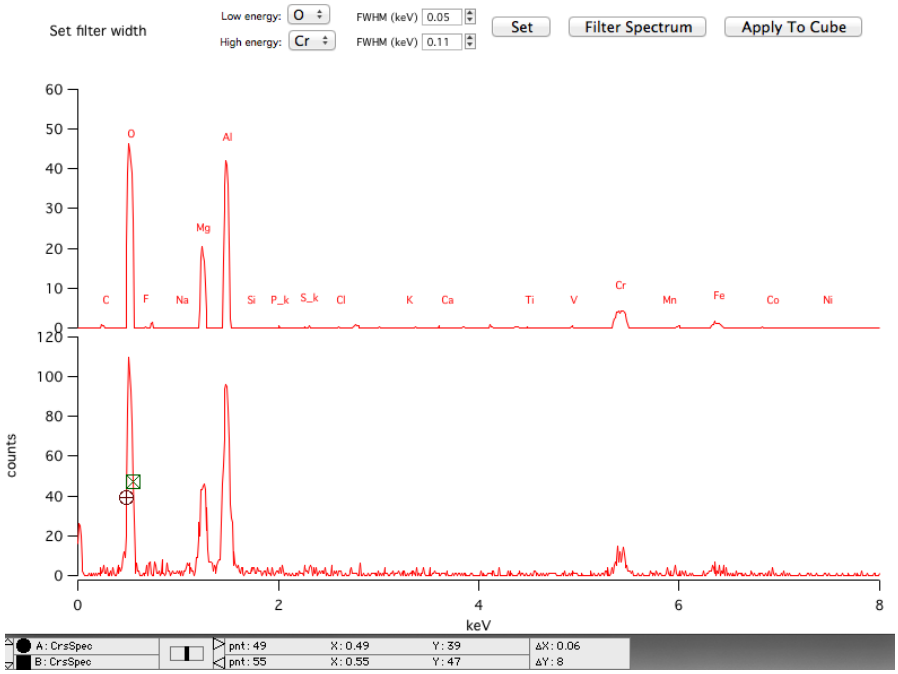


Figure 5: Setting the filter width of the top hat filter. When the graph is opened two cursors are set on the graph, which can be dragged over the spectrum. Here the cursors are set approximately on the left and right flank of the oxygen peak at 0.526 keV at FWHM level. The energy width (ΔX) can be read out in the top right corner of the cursor information panel on the bottom of the graph. For best results the low and high FWHM values should be set for elements sufficiently apart in terms of emission energy.

stripping is a computationally (relative) expensive process and by far the slowest process within iSpectra. When seeking batch processing (see below) with a large number of SI's it may be more efficient to keep working with raw spectral images.

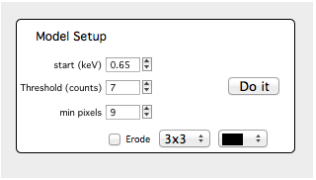
7 Pixel-to-Phase Assignment

The pixel-to-phase assignment is primarily carried out by the user, however, iSpectra as a toolbox provides various instruments and automatic modes to assist. The automatic modes are written under the premise that they should be as transparent as possible to avoid any black-box behaviour. This implies that the user has full control but also full responsibility over parameters that are being used in such modes.

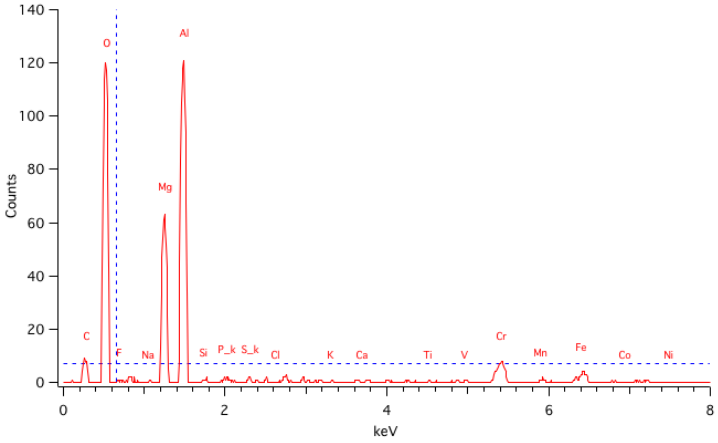
7.1 Automated Group Assignment

Pixel-to-phase assignment starts with an automatic, threshold based method, which can be set up in the **Main Panel**. The following actions can either be carried out on raw-spectra or on background-stripped spectra generated as described above.

The user defines a minimum of counts (**Threshold (counts)**) above which an element is recognised as being present at a given image pixel. In addition it is possible to define low energy cut-off (**Start (KeV)**), such that light elements can be excluded from the elements of interest. This is useful if e.g. the sample consists purely of oxide phases where the presence of oxygen is not discriminating for pixel assignments to different groups. The principle is schematically shown in Fig. 6. The spectrum is extracted from an arbitrary pixel location. The low energy cut-off and a the count threshold are represented by a blue, dashed vertical and horizontal line, respectively. Thus for the spectrum shown in Fig. 6 the elements Mg, Al, and Cr (all identified at Kα energies) are detected as being present at this given SI pixel location. The elements Mg, Al, and Cr are also referred to as the *elemental budget* of the pixel.



(a) Setting low energy cutoff and threshold values



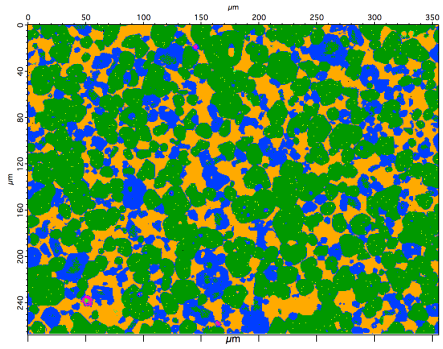
(b) Threshold values displayed on spectrum

Figure 6: Setting threshold values (blue dashed lines) for automated pixel assignment. For the present pixel Mg, Al and Cr will be marked as abundant. Oxygen is excluded because of the low energy cutoff value (start KeV). Figure modified from Liebske (2015).

Figure 7 shows the outcome of the auto group assignment after clicking the **Do It** button and using the threshold settings as shown in Figure 6. The auto-search was done on background stripped pixel spectra, generated with the filter settings according to Figure 5. Pixels with

identical elements budgets are summarised as groups. Groups are shown as overlay on the base image image representing the SI on the left side of Fig. 7. Colours on the overlay can be (de-)activated by clicking on the **Show** checkbox in the **iSpectraGroups panel** shown on the right. The other columns in the panel are the group number and a popup menu for changing the groups colours. The empty row **Phase** may be filled with individual phase names, whereas **Elements** lists the elements (the elemental budget as defined above), which have been detected in the automated group assignment. The first two groups are created by default: TRASH and EMPTY. Empty spectra may occur if no peak is detected above the energy threshold (not present in the given example). This may occur on carbon-based resin embedding material (if additionally a low energy above C Kα is selected). The group TRASH acts as bin for unassigned pixels. To avoid a too large amount of groups on chemically complex samples a minimum group pixel threshold can be set (**min pixels** in **Main Panel**), which causes groups with less than the defined amount of pixels to be merged with TRASH.

The remaining columns are dedicated to the absolute and relative (in percent) number of pixels of a group. By default groups with an abundance of less than 0.5% are marked red (see details below).



(a) Resulting group assignment

Group	Color	Phase	Elements	Show	Pixels	Pix%	EDS Pixels	
				Clear				All
0			TRASH,	☑	26	0.01		EDS
1			Mg,Al,	☑	155	0.08		EDS
2			Mg,Si,	☑	1.0262e+	52.2		EDS
3			F,Mg,Si,	☑	796	0.40		EDS
4			Mg,Al,Si,	☑	42968	21.9		EDS
5			F,Mg,Al,Si,	☑	30	0.02		EDS
6			Mg,Si,Ca,	☑	316	0.16		EDS
7			Mg,Al,Si,Ca,	☑	49633	25.2		EDS
8			F,Mg,Al,Si,Ca,	☑	20	0.01		EDS
9			Mg,Al,Cr,	☑	40	0.02		EDS

(b) The iSpectraGroups panel

*Figure 7: Left: Groups image as overlay on the base image. Right: Resulting pixels assignment summarised in the **iSpectraGroups panel**. To remove a group from the overlay, uncheck the Show checkbox. Left figure modified from Liebske (2015).*

7.2 Displaying and Comparing Cumulative Group Spectra

Cumulative group spectra can be viewed by clicking on the **EDS button** in the group panel. Note, that cumulative spectra are *always (!)* extracted from the raw data cube and never from background-stripped data. Several group spectra can be displayed in one graph. The colour of each spectrum can be synchronised with the appropriate group colour. For easy spectrum comparison cumulative spectra can be normalised by division through the number of pixels of the corresponding group, such that resulting spectra are average spectra of one pixel. This circumvents scaling problems when comparing spectra of groups with very different absolute abundances (Fig. 9).

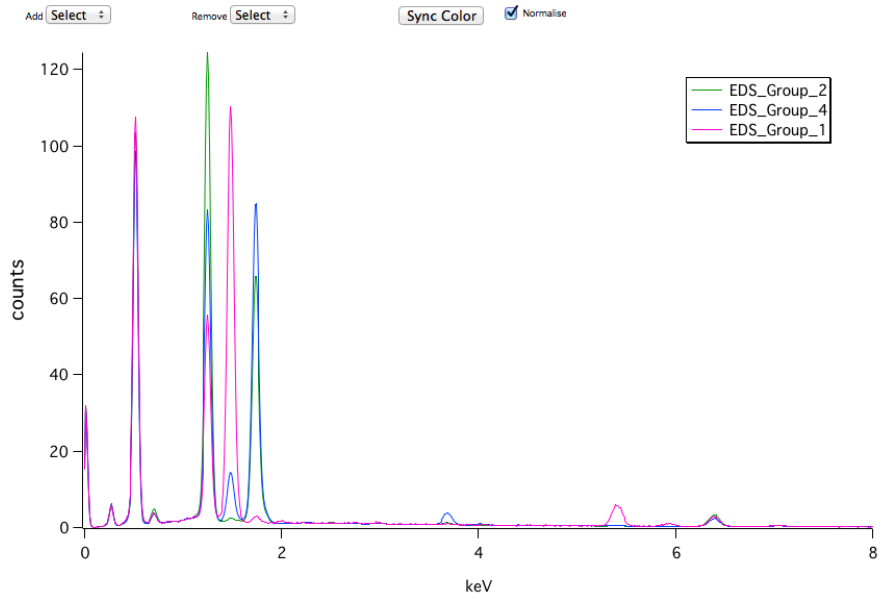


Figure 8: Clicking a EDS button in the iSpectraGroups panel activates the SpectrumDisplay graph. Existing spectra can be added or removed using the popup menus. Activating the Normaise checkbox, normalises the counts of all spectra to one average pixel.

7.3 (Auto-) Merging of Groups

The spectra of the one, chemically homogenous phase within a SI will vary slightly due to statistical variations in counts per pixel, such that intensities of one or more elements may fall just under or over the threshold limit. This can be envisaged on Fig. 6, where the Cr Kα signal is over the threshold limit but an adjacent pixel may contain a spectrum in which the Cr counts would fail to reach this value. This results in the formation of "ghost phases", which subsequently need to be merged with a group that represents the main population of pixels of the actual phase.

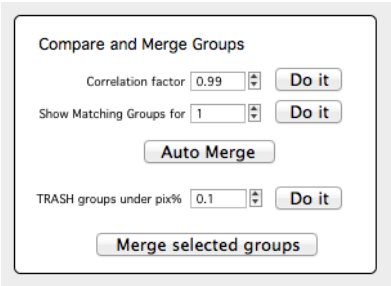
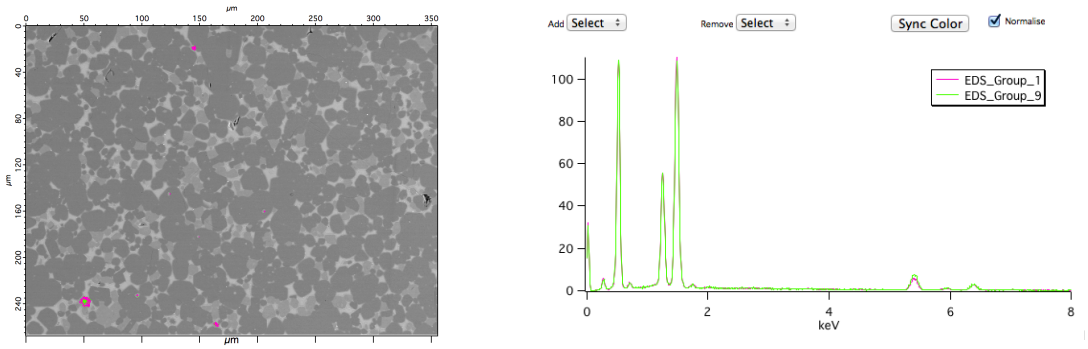


Figure 9: Section of the main panel, showing the options for (auto-) merging operations.

Identifying ghost phases is assisted by calculating cross correlation factors of cumulative group spectra to test for statistical similarities. Setting the **Correlation Factor** in the Model Setup panel to e.g. 0.99 followed by clicking **Do It** will result in the square matrix *M_CorrMatrix* in which statistically similar cumulative spectra, with cross correlation factors larger or equal than

the defined value, will be marked. A cross correlation factor of one means identical spectra (as the diagonal of the matrix indicates). More details on the cross correlation can be found in Appendix B. Marked groups may then be merged by unchecking the *Show* option in the iSpectraGroups panels of all other groups followed by hitting the **Merge Selected Groups** button in the Model Setup panel. Groups with higher group numbers will be merged into the group with the lowest group number of the selection.

Alternatively, groups with matching cross correlation factors can be displayed automatically both as group image and spectra by entering a group number in the **Show matching groups** SetVariable field. Any other group, that is associated with a matching correlation factor, is shown as demonstrated in Fig. 10. The two graphs show easily if there is textural and/or spectral correlation between those groups to justify merging. All groups of interest are set to *Show* in the *iSpectraGroups* panel, such that they can directly be merged if required.



(a) *(b)*

Figure 10: Statistically similar group spectra can be displayed and compared and textural relations reviewed. Here, the ***Schow Matching Groups*** *for group 1 has been executed and group 9 exhibits a spectral cross correlation greater 0.99. Both spectra are indeed near identical and the corresponding pixels are in textural relation with each other. Such groups can be merged.*

Identifying ghost groups manually and merging them into other groups can be time consuming especially when working with chemically complex samples. iSpectra provides two auto-merge functions: 1) Groups can be summarised automatically when they exceed a selectable spectral cross-correlation factor, e.g. greater 0.99 using the **Auto Merge** button. The cross correlations threshold will be one specified by **Correlation factor**. 2) Ghost groups typically contain only an insignificant number of pixels, which are often randomly distributed between other group pixels. Such groups can automatically be merged with trash if a certain minimum pixel percentage value (on e.g. a fraction of a percent level) is undercut. This, however, may also merge accessory phases into the group of unassigned pixels which can be avoided by applying particle analysis functions (built-in into Igor; see also next section) to detect if group pixels occur connected and appear contiguous over a adjustable number of pixels or randomly distributed. The latter groups pixels would be merged into trash but not so the former (**TRASH groups under pix%**, followed by **Do it**).

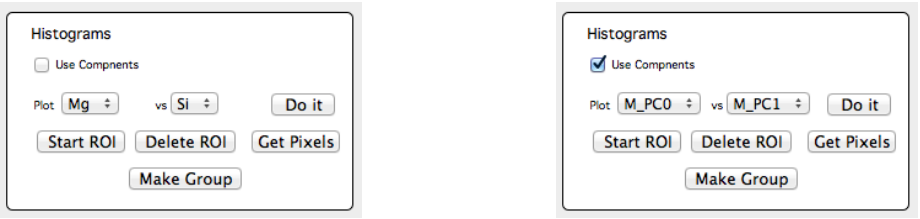
7.4 Particle Analysis

Automated group assignment in chemically complex samples may result in a large number ghost phases. Such groups typically contain a relatively low number of pixels which are often distributed randomly within or around another group with a significantly higher pixel count. *iSpectra* will mark any group with an area percentage less then 0.5% red on the *iSpectraGroups panel* (see Appendix B for details). Typically a majority of groups will initially be marked and such groups may possible be ghost phases although it can not be excluded that "true" phases are be amongst them. **Find Particles** from the menu will execute functions to analyse marked groups whether associated pixels occur localised as particle(s) or are randomly distributed. In the former case the group will be marked green in the iSpectraGroups panel.

7.5 Density Plots

A disadvantage of the threshold-based auto group assignment arises if the same elements are present in two or more phases (above the aforementioned count intensity threshold) but are abundant in different relative proportions, e.g. having different stoichiometries. Pixels covering such phases are presently treated as one group. Deconvolution of two or more different phases within one group must be carried out manually but is assisted by using density plots (also called bivariate or 2D histograms).

Density plot show the number of pixels having e.g. identical count pairs (in the case of elemental density plots) for two elements. Chemically distinct features of the sample appear as a pixel population around a certain element count pair. Density plots are being created from all groups which are set to *Show* on the iSpectraGroups panel! Note, that the automated group assignment must be done prior to using density plots. Density plots can be generated from elemental or principle component maps and be used used to manually (re-)assign group pixels.



(a) *(b)*

Figure 11: Histograms can be generated from either elemental- (left) or principle component maps (right). If the checkbox ***Use Components*** *is active, the popup menus give access to principale component instead of element maps.*

Examples of elemental and principle component density plots are shown in Figures 12 and 13 and show results for all groups, thus all pixels in the spectral image. Figure 12 is a Mg vs. Si density plot, where on Figure 13 the component maps associated with the two most significant variances (M_PC0 and M_PC1) are being used as input to create the histogram. Note, that the scaling of a component density plot takes integer values from 0-255. The original projections stored in the rows of matrix $\boldsymbol{P}$ (see above), from which the maps are being extracted, are real, negative to positive values. For the histogram purposes, such real values are scaled to 0-255 for computational convenience, as it avoids selecting bin sizes for the histogram and the back-tracing

of pixels (as described below) is greatly simplified, when working with integers. Both density plot show similar features, but are differently orientated. In both plots it becomes apparent that five distinct populations exists, four major being approximately horizontally located and a minor fifths one which plots markedly away from the others. While both, elemental and principle components density plots, provide essentially the same information, the major difference between them is that the elemental density plot requires choices to be made about the elemental content of the groups to be displayed. If other elements for the same groups are chosen as input for the histogram, such five populations may not become apparent.

A similar plot as the density plot gan be generated by activating the **Show as pixels per group** checkbox on the interactive density plot. The plot on the bottom of Figure 13 provides similar information as the density plot, but shows each pixel individually in its group colour, rather than the amount of pixels having identical component pair values. In the context of principle component analysis such a representation is usually called a score plot. This representation clearly shows that the orange group hosts two populations. Note, that this observation can be made without any knowledge on the actual elemental budgets of the groups. Thus, the final step in the present pixel-to-phase assignment is to use iSpectras ROI tools to segment out one of the two population from orange group.

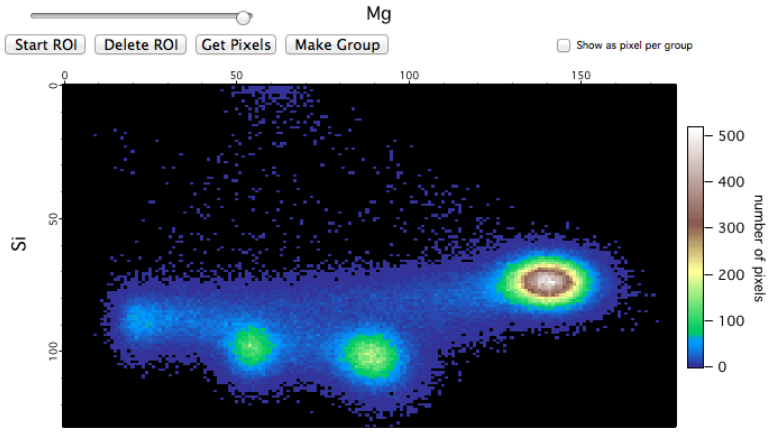


Figure 12: Elemental density plot (Mg vs. Si) for all pixels in the spectra image.

Any population of pixels on any density plot can be selected by drawing a region of interest (ROI) around it using a marquee tool. Select **Start ROI** either from the density plot window or from the **Main Panel** and start drawing the ROI. Once the polygon closes click the upper left icon in the *Show Tools menu bar* of the histogram. The drawn lines will turn yellow. Clicking then the **Get Pixels** button will open another graph with pixels within the ROI marked green (Fig. 14). A new group of pixels, which will be added to the iSpectraGroups panel, can be created by clicking **Create Group**. Any pixels that were previously associated with another group will be removed from it and added to the new group. Pixels counts in the iSpectraGroups panel will be updated immediately. Note, that the ternary plot requires working with normalised intensity ratios and does not work with absolute counts (with a bin size of one) as the two element histogram does. This has consequences for the resolution with which distinct populations of pixels can be resolved, especially with low total count rates. The ternary diagram can be used for sample exploration only - ROI tools and are presently not available.

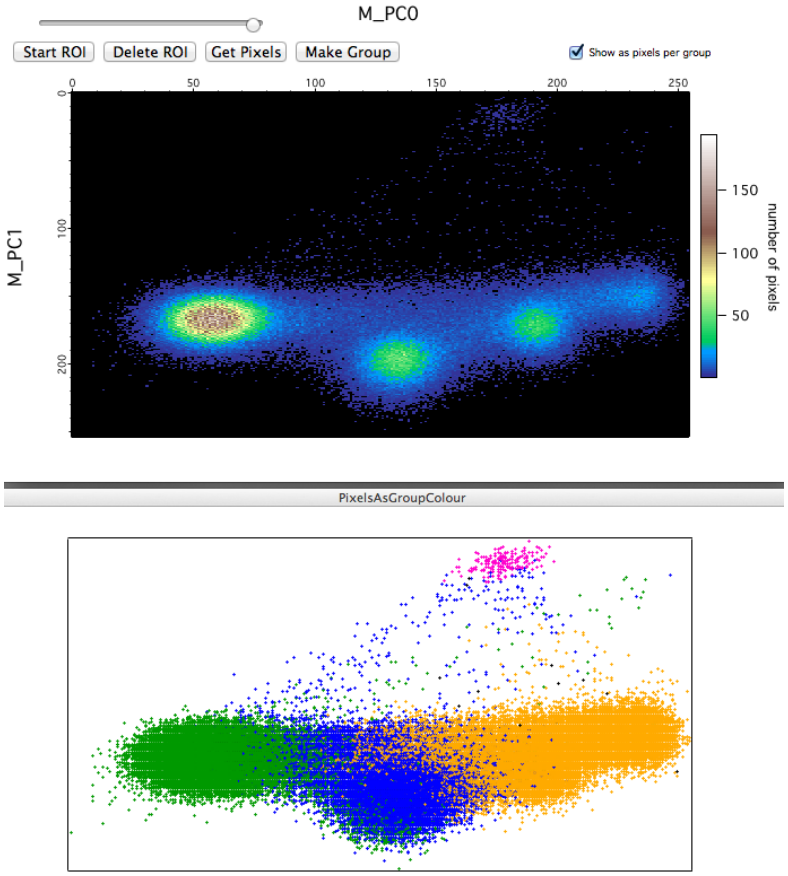
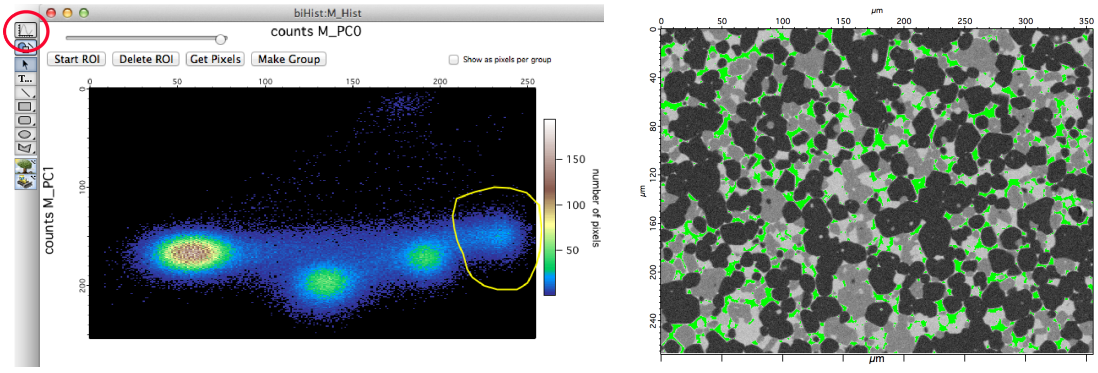


Figure 13: Elemental density plot (Mg vs. Si) for all pixels in the spectra image.



(a) *(b)*

Figure 14: Using the region of interest (ROI) tools, populations of pixels can be selected, located and added as groups. First, click "Start ROI", draw the ROI and close it by clicking the last point close to the first one, finally exist drawing mode by clicking the symbol in the top left corner (red circle in a). Then click "Get Pixels" for back tracing.

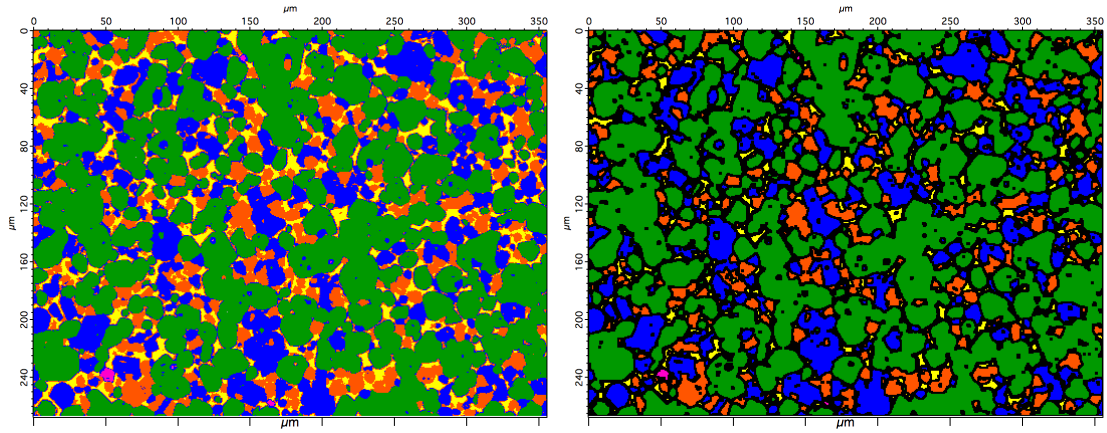
8 Morphology Operations

The assignment of pixels to phases can be refined on the basis of Igor's built-in *Image Morphology* operations. Such operations are based on nearest neighbour constraints and do not consider any spectral correlation of the affected pixels.

8.1 Bulk Erosion of Group Pixels

Cumulative spectra may be exported, once groups or phases are satisfactorily defined. However it must be considered that pixels located at phase boundaries likely contain EDS spectra that have x-ray contributions from all neighbouring phases as a result of overlapping excitation volumes. This can readily be seen in Figs. **??** and 14 where linear mixing trends in between population are a clear indication for this effect. The amount of pixels that are affected by mixed phase contributions will depend on the acquisition properties, such as acceleration voltage, image resolution and the nature of the target material. Care should be taken that the actual excitation volume of the average sample, when projected onto the sample surface, should not be significantly larger than one pixel, otherwise magnification or the resolution of the spectral image should be adjusted. A software tool called *Casino* can be used for predicting excitation volumes and can be downloaded from `http://www.gel.usherbrooke.ca/casino/`.

If the overlap of x-ray contribution from different phases can be restricted to about one pixel the cumulative phase spectra of each group can be filtered by excluding pixels located at grain perimeters. This can simply be achieved by applying a standard image processing algorithm to each group: Erosion. Erosion can be applied to all groups simultaneously by checking the **Erode** checkbox in the **Main Panel** . Fig. 15 shows the before (left) and after (right) of the erosion process based on a 3x3 square structure element and the effect on the number of pixels considered for cumulative spectra (bottom; EDS Pixels vs. Pixels). If the excitation volume is suspected to be larger than covered by a 3x3 structure element, the amount of eroded pixels can be increased by choosing a larger 5x5 (circle) structure element.



(a) Uneroded *(b) Eroded*

Group	Color	Phase	Elements	Show	Pixels	Pix%	EDS Pixels	
				Clear				All
0		un_assigned	TRASH,	☑	18	0.01	0	EDS
1		Spinel	Mg,Al,Cr,	☑	195	0.10	78	EDS
2		Olivine	Mg,Si,F,	☑	1.0101e+	51.4	80513	EDS
3		Ortho_Pyroxene	Mg,Al,Si,F,Ca,M_PC1_	☑	45724	23.3	24704	EDS
4		Cline_Pyroxene	Mg,Al,Si,Ca,F,	☑	36681	18.7	14792	EDS
5		Silicate_Melt	M_PC1_vs_M_PC0_1	☑	12980	6.6	2861	EDS

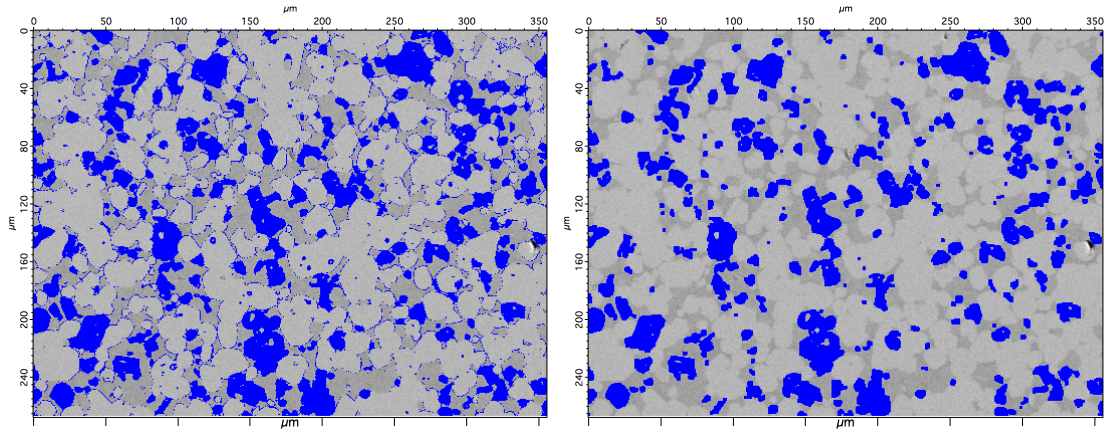
(c) EDS Pixels vs Pixels in the iSpectraGroups panel

Figure 15: The effect of bulk erosion on pixel assignment. a) before, b) after activating the erosion checkbox (eroded pixel, thus discarded pixels for cumulative spectra, are blacked out). C) the effect on cumulative pixels (EDS pixels vs. Pixels). Figure modified from Liebske (2015).

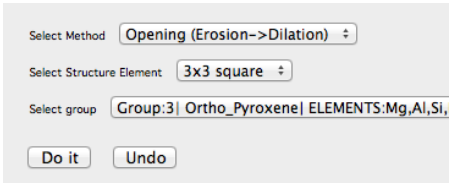
8.2 Particle Morphology of Individual Groups

Erosion, combined with complementary *dilation* (an algorithm that adds pixels) form the basis of standard image processing algorithms, the so-called *Opening* (erosion followed by dilation) and *Closing* (dilation followed by erosion) operations. Combinations of erosion and dilation (and opening and closing) can be applied to single groups using the **Correct Phase Morphology** menu point to refine a groups morphology. An application is shown in Fig. 16. The pixel assignment of the blue group is based purely on the automated, threshold based algorithm. The assignment clearly covers larger grains, but also produces artefacts around other phases, due to effects of overlapping excitation volumes. Applying an opening operation results in a much cleaner and, based on morphological considerations, much more plausible phase assignment. Any discarded pixels are moved to TRASH, whereas added pixels are subtracted from existing groups.

Closing operations can be useful if larger grains of a phase exists in which single pixels, due to insufficient counting statistics, are incorrectly associated with other groups. When turning on the standard bulk erosion filter, the missing pixels would cause large proportions of the bigger grains to be eliminated, thus reducing the amount of pixels in eroded EDS spectra significantly. For smaller particles, missing pixels within such objects may actually cause complete elimination of a phase after erosion. This can be avoided, if appropriate morphology operations are being applied.



(a) *(b)*



(c)

Figure 16: Particle morphology operations can be applied to single phases to refine phase morphologies. Methods are based on standard erosion and dilation algorithms. Result of applying image morphology to a group, which only partly outlines morphological features on the grey scale images (a) before, b) after). The outcome is a much cleaner and more plausible phase distribution. Note that these algorithms are purely based on morphology operation without considering any spectral features. Different methods and structure elements can be used from the morphology panel c). Figure modified from Liebske (2015).

Note, that any morphology operations have to be applied with care because they are not related to any spectral content of the pixels. The morphology panel support one UNDO action.

9 Single Phase Analysis

By default *iSpectra* creates cumulative EDS spectra of all pixels associated with one group when clicking the EDS button in the *iSpectraGroups panel*. However it may be desirable to compare spectra and thus chemical composition of e.g. different particles of one group. This can be done using the menu point **Single Phase Analysis**. iSpectra will prompt for the group number to be analysed for particles and a minimum count of pixels under which a particle will be discarded. Lastly the user has the option to directly write the spectra as EMSA files to disk, otherwise the spectra will be stored as waves in *root:spectra*. If export as EMSA is selected, iSpectra will ask for an existing EMSA file if the wave EmsaHeader does not exist (see section 10 for details). iSpectra will generate a binary mask *M_ParticleMask* and export/store EDS spectra including the image coordinates for the corresponding particle.

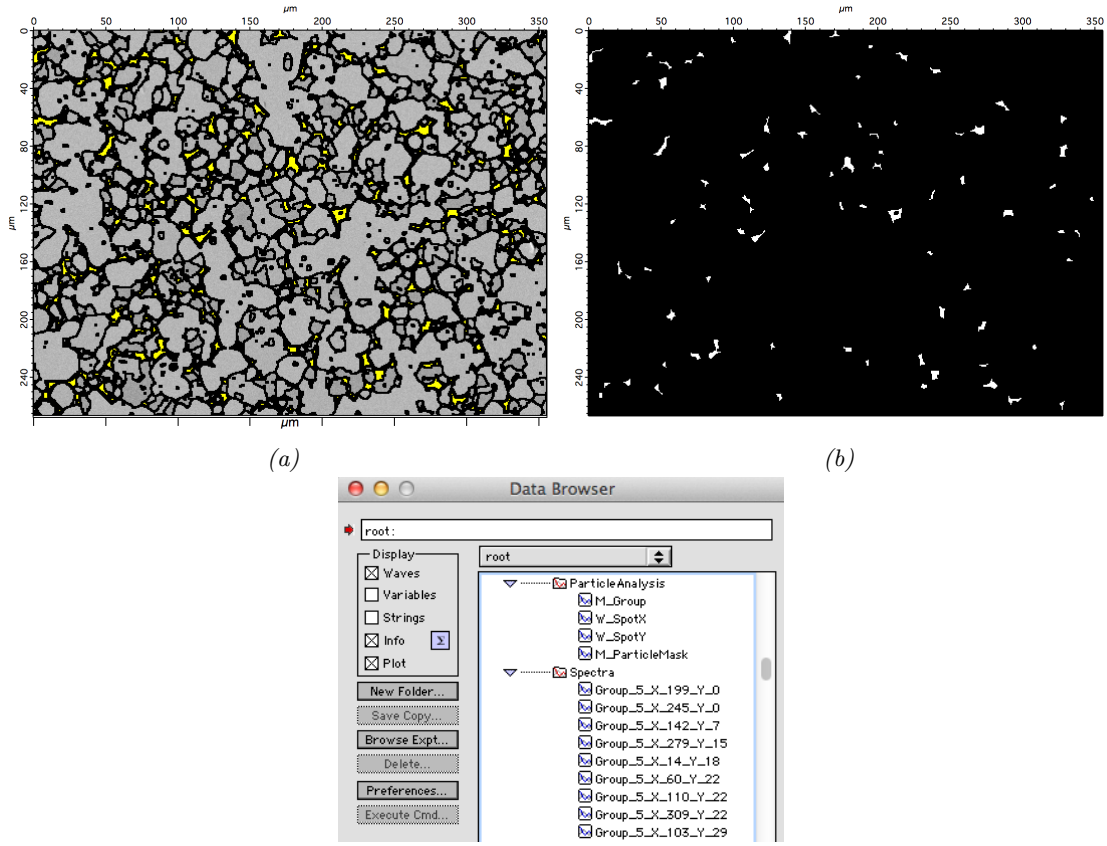


Figure 17: Outcome of Single Phase Analysis applied to group 5 after erosion (a, see also Fig 15). Single phase analysis stores spectra in root:spectra and optionally on disk as EMSA files (c). Note that each spectrum name contains the XY coordinates of the particle that it represents. A binary mask (b) of the particles will be stored under root:ParticleAnalysis.

10 Exporting Spectra and Results

iSpectra supports spectrum export in standard EMSA spectrum file format. EMSA files are text files and contain a text header with instrument settings followed by the spectrum as comma separated list of energy and intensity. Exporting spectra thus requires an appropriate spectrum which was ideally extracted from the present SI and the native spectral imaging software. *iSpectra's* export routine will ask for the location of such a spectrum during the *Select Extracted Spectrum* dialogue. The header information from this spectrum will be stored as text wave (*EmsaHeader*). As long as this wave exists, *iSpectra* will not ask again for a spectrum file. In the following *iSpectra* will also prompt to specify the directory to where the EMSA files will be written to. This path will be stored as Igor symbolic path (*SpectrumPath*) and as long as it exists, this path will be used. If the output path needs to be changed, *SpectrumPath* must be deleted which can be done under the Igor main menu *Misc-Kill Path*.

The export options comprise export of all and selected spectra (set to *Show* on the iSpectra-

Groups panel) or a full export including the sum- and RGB images (if the latter exists) and eroded and un-eroded group images as TIF files. Note that spectra will be eroded before export. Additionally a snapshot of the iSpectraGroups panel will also be saved as TIF. For convenient review all image files are summarised in *Summary.htm*, an HTML file which can be opened with e.g. a standard internet browser. The sum- and group images are scaled, regardless of their native resolution, to a constant image width (see Appendix B for changing the width). For full export a new symbolic output path needs to be specified (*SavePath*). The output folder name is by default "Cube".

The graphical representation of the group distribution, saved as *M_Groups.tif*, displays all groups. However, it is possible to generate TIF images that just show one phase overprinted on a grey scale image such as *M_SumImage* or *M_TifImage*. To generate an appropriate 3D RGB wave, which can be saved as TIF using Igor's *Save Image* dialog, execute:

```
iS_MakeGroupTif(GroupNumber, ImageMatrix, "Name")
```

where *GroupNumber* is the actual number of the group to be displayed on a *ImageMatrix*. The resulting image will be called "Name".

10.1 Applying a Colour Data Base

Consistently looking images in terms of colour assignments to given phase names can be generated when building a colour data base using the **Colour Data Base** menu points. A colour data base is a 2D text wave, *root:M_ColourDataBase*, containing phase names which are linked to 16 bit RGB colours. If the phase names in the *iSpectraGroupsPanel* are matching the names listed in the data base (not case sensitive), then the colours in the present SI pixel assignment will be changed to match the colours of the data base. Note, that the colour data base will convert any entered name into a standard object name, i.e.,

```
DisplayHelpTopic "Standard Object Names"
```

11 Saving Intermediate Results

iSpectra does generally not support any UNDO action (with the exception of the morphology operation). Regularly creating restore points (**Create Restore Point**) allows the user to go back to any previously defined state of pixel assignment. There is no restriction on the number of restore points (except for system memory) and any previously saved ones can be accessed using the **Load Restore Point** menu option.

12 History Wave and Batch Processing

All actions taken to assign pixels to groups are recored in the wave *:packages:iSpectra:History*. This wave can be viewed using **Show History** from the **History Actions** submenu. An example of the content of this text wave is shown in Fig. 18. The history wave differs from the information stored using the **Create restore point** option, such that it enables to re-create *the way pixels were assigned*, rather than just the actual pixel assignment (this is what restoring does).

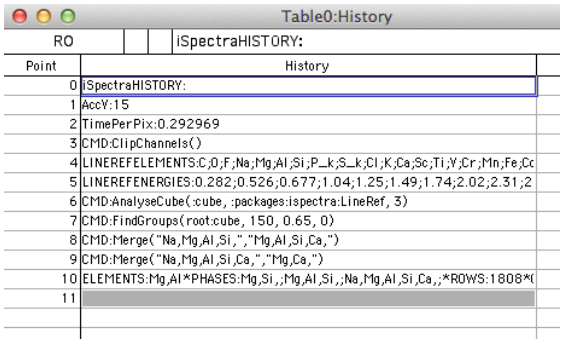


Figure 18: Example of a history wave: the wave contains, e.g. information about the element list or which groups have been merged. Row 10 stores information on count density plots, e.g. which elements were selected, active groups, histogram dimensions and the selected region of interest (ROI)

Therefore, once all pixels have been satisfactorily assigned, saving the history wave (**Save History**) to disk (as *.itx file) is the recommended option to save the data set, as this avoids saving the larger 3D spectral image, which would otherwise exist both as *.raw file and as 3D wave within the Igor experiment. Re-creating the pixel assignment can be done (when starting with an empty experiment) by loading the spectral image and subsequently executing the **Apply History** menu option. Note, that *Save History* ist different from simply choosing *Igor's Save Waves* dialog: In the former case, *iSpectra* will add information to the saved wave, such as e.g. the phase names and the groups colour assignments. Thus the content of the History is altered upon saving.

As the history wave contains a recipe how to assign pixels to phases, it can in principle be applied to any other SI that is expected to contain the same phases if it was recored under identical acquisition properties. **Setup Batch Processing** leads the user through several input dialogs for various file paths, parameters and output folder options in the following order:

1. Path to a valid *iSpectra History file*.
2. Path to an *EMSA file containing an appropriate spectrum header* (see section 10).
3. Path(s) to *RPL file(s)*; these are the complementary files to the SI data sets saved as *.raw files. Select all files that should be processed in batch.
4. *iSpectra* will ask how to deal with groups which do not contain a manually entered phase name. See section 12.1 for details. Clicking yes in the first popup menu leads to an input dialog for two parameters: First, the pixel percentage under which any un-named group will be merged automatically with TRASH, and second, whether other un-named groups with abundances above this percentage should be analysed if they occur as particles (see also section 7.4). If mode 1 is selected (particle analysis), each remaining un-named group will be merged with TRASH if it does not appear localised but rather randomly distributed over the sample surface. If a particle is found, *iSpectra* will write this into the history window.
5. Batch processing provides the possibility to export *RGB* images. If this is required, the elements (as named in *root:Intensities:*) for red, green and blue channels must be entered as semicolon separated list (NO white space allowed), thus e.g. "Si;Mg;Al;". If left empty, no RGB images are being exported.

6. Enter a *folder name*: This will be name of the folders on disk in which all exported data will be placed.

7. Lastly, the *output directory* needs to be specified. This dialog is skipped if there is an Igor symbolic path named *SavePath*.

iSpectra will store the above information as string variables in the text wave *root:BatchSetup*. At the end of the dialog sequence batch processing can either be started immediately or later using the **Start Batch Processing** option from the menu. The wave *BatchSetup* can be easily be modified (or saved, or just reviewed) without having to go through the dialog sequence again.

Once started, batch processing will write its progress into Igor's history window. Note that the computers energy saving options, such as (disk-) sleep mode or screen saver settings may cause batch processing to stop executing when those time limits are reached. Such settings should be turned off.

In addition to EMSA spectrum files, TIF images and HTML output, batch processing produces two additional files:

- *BatchSummary.txt*: A plain text file containing all information stored in the iSpectra-Groups panels, which is useful for further data processing. The content is also available as 2D text wave *BatchSummary*. A figure legend listing phase names, colours and the cumulative modal proportions of those phases can be generated from the *BatchSummary* wave using the menu point **Make Legend From Batch Summary**.
- *logfile.txt*: A plain text file echoing the output *iSpectra* has written into Igor's history window during batch processing. This file can be used to review problems in pixel-to-assignment as they can arise due to unsuccessful merging operations.

12.1 Details on Batch Processing

Reproducing pixel-to-phase assignments is based on a) elemental intensity maps generated by averaging count channels (section 4.1), b) merging groups after threshold-based group finding, and c) group definitions based on count density plots. The merging operation merges groups according to elemental contents (see column *Elements* in the *iSpectraGroups* panel). Taking the groups 1 and 9 (see Figs. 7 and 10) as example, means that group "Mg,Al,Cr," is merged into "Mg,Al,". However, if in another SI processed in batch, "Mg,Al," does not exist, then the merging operation will report a warning to Igor's history window and to *logfile.txt*. Such unsuccessful merging operations can lead to undefined phases, in particular if a group, to which other groups are supposed to be added (merged), is a ghost phase, which may not be present in other SI's. Note that groups are always merged into the group with the lowest group number. Phase definition based on count density plots provide a more reliable basis and should be the preferred method when batch processing is sought.

When running batch processing it is essential to assign names to all defined or recognised phases as shown in Fig. 15 (Spinel, Olivine etc.). Groups without phase names can be treated specially during batch processing (see point 5 in the previous section). When applying a history wave to a new data set a number of ghost phases will likely occur, thus creating additional groups in the *iSpectraGroups* panel. Such groups can automatically be merged with TRASH based on a pixel percentage threshold value. In addition un-named groups can be analysed with the particle analysis algorithms, such that groups that occur localised will not be merged with TRASH but

will be marked as *UnknownPhase* in the *iSpectraGroups* panel. Note that *UnknownPhase* is followed by a consecutive number, which is set to zero before processing each spectral image in batch. This means that e.g. *UnknownPhase_0* in two different SI's may contain different elemental budgets.

12.2 Create Montage

If batch processing ran over several spectral images recorded in automated grid mode the resulting TIF images can be arranged to represent the entire grid in one image file using **Create Montage**. The dialog will ask for the individual files to be assembled. The images will not be stitched but only be concatenated in the appropriate orientation. The position of an individual image is determined from the file name, e.g. *M_Groups_[Grid@3 1]*, whereas the column and row positions are identified by a preceding @ an a succeeding] symbol, separated by white space. Other file names need to be changed accordingly or an error message will occur. The dialog will ask for the number of pixels overlapping in X and Y direction. The resulting image matrix is called *M_ImageMontage*. If the overlap is not corrected properly the operation can simply be repeatedly executed from the command line using:

```
iS_Stitch(:ImageFolder, x, y)
```

where x and y need to be replaced by appropriate numbers of pixels which can be found by visual inspection of the displayed image. *M_ImageMontage* needs to be saved manually using Igors *Save Image* dialog.

References

Bright, D. and Newbury, D. (2004). Maximum pixel spectrum: a new tool for detecting and recovering rare, unanticipated features from spectrum image data cubes. *Journal of Microscopy*, 219:186–193.

Liebske, C. (2015). ispectra: an open source tool box for the analysis of spectral images recorded on scanning electron microscopes. *Microscopy and Microanalysis, submitted*.

Statham, P. (1977). Deconvolution and background subtraction by least-squares fitting with prefiltering of spectra. *Analytical chemistry*, 49:2149–2154.

A Appendix: Data Folder and Wave Content Reference

This section gives a brief description of folders and waves which are created when using iSpectra.

A.1 Folder *root:*

- Cube: 3D 8 or 16 bit unsigned integer wave (depending on the maximum counts) holding the spectra image. X, Y dimension correspond to SI image resolution Z holds the spectra.
- Cube_BKG_rmvd: 3D 8 or 16 bit unsigned integer wave (depending on the maximum counts) holding the *holding background stripped* pixel spectra. This wave is only being created when applying background subtraction.
- M_SumImage: unsigned 16 bit 2D image matrix containing the sum of each EDS spectrum, normalised to values ranging from 0-65535. Is by default displayed as image as graphical SI representation.
- M_TIFImage: optional TIF image as unsigned 16 bit 2D image matrix, normalised to values ranging from 0-65535.
- M_RGBImage: optional RGB image as unsigned 16 bit 3D image matrix (red, green and blue as layers), normalised to values ranging from 0-65535. This image is generated when using the popup menus under *Generate RGB Map Image*.
- M_Groups: Each pixels contains the groups number to which it has been assigned. Eroded pixels will be set to number 301.
- M_GroupsErodeUndo: Copy of M_Groups before erosion is applied. Is used to re-create M_Groups if the **Erode** checkbox is unchecked and subsequently killed.
- M_GroupsMorphologyUndo: Copy of M_Groups before morphology operations are applied. Only keeps track of one action. Is used to re-create M_Groups if the **UNDO** button in the **Morphology** panel is clicked and subsequently killed.
- M_ImageMontage: Will be created when running *Create Montage* on exported tif images from batch processing.
- M_CorrMatrix: square matrix which is created when setting the *Correlation Factor* in the *Main Panel* . Cross correlation factors of cumulative spectra are shown if the values are higher than the specified value, otherwise the entry will be *NaN* (not a number).
- M_ImageMontage: Will be created when running *Create Montage* on exported tif images from batch processing.
- BatchSetup: This text wave will be created when running batch processing and contains all required parameters and paths to files.
- M_ColourDataBase: 2D text wave linking given phase names to RGB values. This wave can be created to loaded using the *Color Data Base* menu points.

A.2 Folder *root:packages:iSpectra:*

This folder holds a number of global variables mostly stored in numeric or text waves. Most of them use dimension labels for access in iSpectra code

- Settings: Holds a number of global variables, such as the numeric values of the *Main Panel* . Uses dimension labels.
- M_BaseImage: 16 bit 3D image matrix (red, green and blue as layers), normalised to values ranging from 0-65535. This wave is being displayed as base image on which groups of pixels are shown as overlay. The content of M_BaseImage corresponds either to M_SumImage, M_TIFImage or M_RGBImage depending on the popup *Select Image*.
- LineRef: Wave holding reference energies for map extraction. Dimension labels contain element names/abbreviation. Note that some elements can not be addressed with their actual symbols (specifically P, S, As) because of name conflicts with Igor operations.
- History: Text wave to which commands or parameters are written which describe the pixel assignment. This wave can be used to reproduce the current state of assigned pixels.
- M_Model: 2D text wave to store information displayed in iSpectraGroups panel, such as detected elements but also numeric entries such as RGB values or pixel counts. Uses dimension labels.
- CrsSpec: Spectrum at any position of the SI, when the base image is clicked.
- CrsSpec_BKG_rmvd: Optional top-hat filtered CrsSpec.
- ElementLabel: Text wave that is being created when using the *Add line reference to top trace* to label peaks. The wave contains the elements at each channel position.

A.3 Folder *root:Intensities:*

Contains element maps of all elements defined in LineRef. Elements from this list of waves can be selected for count density plots.

A.4 Folder *root:Components:*

The folder contains waves that are being created by the principle component analysis during data import. The names of the matrices correspond to the names given in section 5.1 but with a preceding *M_*.

- M_A: data matrix containing all element maps from folder root:intensities expanded into columns of M_A.
- M_D: as M_A but with means subtracted from all columns.
- M_C: eigenvector matrix of M_D created by singular value decomposition of M_D.
- M_P: projection of each pixel onto an eigenvector in rows.
- W_CumulativeVAR: 1D wave containing the cumulative variance of the eigenvalues.
- M_PC0...M_PCn: Principle component maps.

A.5 Folder *root:IntPlotData:*

This folder is created when using count density plots.

- M_ActivePhases: Mask of pixels which are active and on which the 2D histogram will be calculated.
- M_Hist: The 2D histogram.
- M_PixFromDensityPlot: contains the pixels after drawing the ROI and clicking **Get Pixels**.
- M_ROIMask: Mask of the region of interest.
- W_xx: This 1D wave is created when activating the *Show as pixel per group* checkbox on the interactive density plot graph. W_xx contains the x data for the plot *PixelsAsGroupColors*.
- W_yy: like above; containing y data.
- W_Groups: 1D wave containing the group numbers of each element in W_xx and W_yy.
- M_GroupColours: 2D wave containing RGB values of the group colours.

A.6 Folder *root:Spectra:*

This folder contains all spectra which are being created when clicking the EDS button on iSpectraGroups panel or running single phase analysis. If a spectrum of a given name exists it will be overwritten. All existing spectra within this folder can be added to the Spectrum comparison panel using the **Add spectrum** popup menu.

A.7 Folder *root:ParticleAnalysis:*

This folder will only be creating when using single phase analysis.

- M_Group: Mask of the selected group.
- M_ParticleMask: Mask of identified particles.
- W_SpotX, W_SpotY: XY coordinates of the identified particles.

A.8 Folder *root:Restore:*

The folder is created when setting restore points. The folder will contain subfolder named after the given restore point names. Each of the subfolders contains three waves: M_Groups, M_Model, and Settings. When loading a restore point each of those wave will replace the present versions in root: and root:packages:iSpectra.

A.9 Folder *root:ImageFolder:*

This folder will be created when using *Create Montage* and contains the individual images to be stitched to one single image matrix.

B Appendix: Constants in iSpectra

iSpectra.ipf contains a number of string and static numeric constants at the very top of the procedure file. These constants can be used to adjust iSpectra to the needs of the user but can mainly remain untouched. It follows a brief description of such constants with their default values.

- *elements = C;O;F;Na;Mg;Al;Si;P_k;S_k;Cl;K;Ca;Ti;V;Cr;Mn;Fe;Co;Ni;* Semicolon separated list of default elements used in iSpectra. When *iSpectra* is initialised, the content of this string is written to the wave *LineRef*. Element symbols MUST NOT interfere with existing Igor operations (this is why *S* and *P* cannot be used) and MUST follow the rules for Igor *Standard Object Names*, e.g. execute:

  ```
  Displayhelptopic "Standard Object Names"
  ```

  for more details. Element names can be descriptive, e.g. *Re_M_alpha_1* is a valid name for an emission energy of 1.842 keV. The default list its suitable for the analysis of geological relevant material; the authors area of interest.

- *energies = ...*; Semicolon separated list of energies of the default elements in the same order as listed in *elements*.

- *avgChannels = 3*; Default number of channels to be averaged at given element position; can be manually overridden using the *Re-extract Element Maps* menus point.

- *xPerChannel = 0.01*; default x- (energy) spacing of cube in keV; set by spectral imaging system; 0.01 keV for Thermo NSS 7 system.

- *OverVoltage = 1.5*; Overvoltage factor; e.g. function FindGroups() will ignore energies above 10 kV when spectrum is acquired at 15 kV. Takes into accounts lower intensities at higher voltages.

- *maxPhases = 100*; applied in function FindGroups(); prevents unrealistic number of groups in automated search.

- *MinPhaseFraction = 0.5*; pixel percent abundance under which a phase is marked red on iSpectraGroups panel.

- *MaxkeVSpecVal = 8*; default maximum keV value shown in displayed spectra.

- *minParticleArea = 9*; minimum number of contiguous pixels among one group above which the groups is recognised to appear as particle.

- *nChannelsForCrossCorrelation = 650*; number of channels that is used for cross correlation calculations, lower is faster, higher is slower but more accurate.

- *IntervalForCrossCorrelation = 3*; sampling interval for cross correlation. Higher is faster but less accurate; only every *i'th* channel is considered in the cumulative spectrum which is used to calculate cross coloration factors.

- *HTMLimageWidth = 400*; Width in number of pixels of images written to HTML output file. All images will be scaled to this value, regardless of their native resolution.

- *totPCVarianceMax = 99*; principle component analysis will extract as many principle component maps until a certain cumulative variance percentage (=totPCVarianceMax) (associated with those maps) is reached. The default is 99 (%).

- *MagCal_A = -0.999* and *MagCal_B = 5.093*; fit constants of equation 1 to relate image width (in microns) to image magnification. Needs calibration for accurate usage.