ParticleAnalysis2 GUI Manual

Ryan C. Moffet
University of the Pacific, Department of Chemistry and Biochemistry

Objective and Brief Summary:
The objective of this manual is to guide the budding analyst through the various stages of scanning transmission X-ray microscopy (STXM) data analysis as applied to particles on a substrate. A graphical user interface (GUI) has been constructed to make these procedures more accessible to those new to STXM or MATLAB. Moreover, the ParticleAnalysis2 GUI can also speed up the use of frequently used routines by removing the tedium of entering codes into the command line or running batch scripts. Within the ParticleAnalysis2 GUI there are three basic levels of data abstraction: Raw Data Processing, Processed Data Analysis, and Dataset Exploration and QC. Many of the Raw Data Processing and Processed Data analysis routines stem from codes published previously with some minor changes. New to ParticleAnalysis2 is the concept of a Dataset where the analyst can distinguish various samples that each have numerous STXM regions of interest (ROIs). The analyst can explore all particles in the dataset using an interactive plot whereby all particles for a particular sample are shown; the analyst may easily view the raw data for the particles or remove the particles from the data set if there are problems. These new routines were instrumental in a recent study on the morphological characteristics of internally mixed soot particles and it is hoped that they may serve to advance such scientific studies in the future.

1. STXM Instrument and Data Analysis Background:

This MATLAB application was written for data acquired using scanning transmission X-ray microscopy coupled with near edge X-ray absorption spectroscopy (STXM-NEXAFS)[1]. STXM-NEXAFS instruments are currently used exclusively at synchrotron facilities and all produce data having the same format. The current application was born out of a need to perform analysis on large numbers of particles that have been deposited on substrates; thus, the name "**Particle**Analysis2". Some of the code included in ParticleAnalysis2 has been published on the mathworks website previously[2] and much of the research and development has been documented in Moffet et al. 2010[3].

The only type of data files ParticleAnalysis2 operates on are "stacks" or images that will be turned into stacks. To formally describe what an image or stack is, consider that the STXM obtains an image by raster scanning the sample while recording photon counts at discrete positions (x_m,y_n). The photon counts are stored in a matrix:

$$I_{nm} = I(y_n, x_m).$$

The STXM-NEXAFS instrument is a *spectromicroscopic* technique, meaning that images over the same field of view are collected *as a function of energy*. The resulting data is represented by a three dimensional matrix called a stack (short for image stack):

$$I_{nm} = I(y_n, x_m, E).$$

It is worth noting that another data analysis package, aXis 2000, is also available for working with the data files generated with STXM-NEXAFS. aXis200 has the advantage of working with a larger variety of STXM file types such as linescans, spectra, and stacks. While ParticleAnalysis2 does not contain built-in routines to handle all STXM data types, MATLAB itself can be used to manipulate the data as well (as well as the output from ParticleAnalysis2). In the future, we plan on including some GUI functions to handle spectra generated from the routines here. This document serves to guide the user through the ParticleAnalysis2 GUI. It is expected that the analyst already knows about the STXM instrument and the sort of data it gives. For resources covering more fundamental aspects of the STXM instrument and the type of data it gives, I direct the reader to the references in this section.

2.1.1 Setting up your directory

The most crucial step when starting out a project is to select a location on your computer to store your data for analysis. I recommend that you make the directory path to your data as simple as possible. A folder directly off of the C: drive is preferred. For example:

C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData

Is where I will store the data for this manual. I will include the data in the *TestData* folder in this release of the software. Within the test data folder, you should have a separate folder for each sample that you have collected data on. Here, a "sample" almost always constitutes a sample collected onto a substrate (TEM grid, Si_3N_4 window…). So, in my TestData folder, I have included three separate folders for three different samples. The folders in my case are:

C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1027_hole2
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1227_hole8
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9

It is wise to indicate the sample time or some other systematic naming system for your folders because this will help your analysis later.

IT IS ESSENTIAL THAT YOU DO NOT CHANGE YOUR DATA DIRECTORY STRUCTURE IF YOU WANT THE DATASET (LEVEL3 DATA ABSTRATION) GENERATED HERE TO CONTINUE TO BE LINKED.

2.1.2 Populating Your Directory with STXM Data

Stack Data:
Each of the three folders I made in the preceding section is populated with STXM stack data. Each STXM stack is stored in a folder with a very particular naming system. In the case of data collected from beamline 5.3.2, the folders following the naming scheme:

BBB_YYMMDDSSS

where BBB is the beamline number, YY are the last two digits of the year, MM is the month, DD is the day and SSS is the data file number. So, for example, the first data in the folder 1027_hole2 is named ***532_120722000***, which indicates it was the first data collected on beamline 5.3.2 on July 22, 2012. Within this folder are two different file types: .hdr and .xim files. The .hdr file is an ASCII file containing metadata about the stack (pixel step size, image size, ring current, etc.). The .xim files are simple ASCII files with numbers representing photon counts.

Images to be Stitched Together Into a Stack:
Each image file has both a .hdr and .xim file associated with it. These should be kept under the sample folder, following this example, in the *1027_hole2* folder.

2.2.1 MATLAB Startup and ParticleAnalysis2 Installation:

This software currently requires the user to have MATLAB with the image processing toolbox. If you haven't done so already, start matlab. Future versions may have the ability to run without the need for a MATLAB license. Nevertheless, there is no guarantee this software will work on your computer and I do not offer technical support. If you are having problems, it might be useful to note that I have written this software on the following version:

```
>> ver
-----------------------------------------------------------------------------------------
--------------
MATLAB Version: 9.0.0.341360 (R2016a)
MATLAB License Number: 892748
Operating System: Microsoft Windows 7 Home Premium  Version 6.1 (Build 7601: Service
Pack 1)
Java Version: Java 1.7.0_60-b19 with Oracle Corporation Java HotSpot(TM) 64-Bit Server
VM mixed mode
-----------------------------------------------------------------------------------------
--------------
MATLAB                                                Version 9.0         (R2016a)
Image Processing Toolbox                              Version 9.4         (R2016a)
Optimization Toolbox                                  Version 7.4         (R2016a)
Statistics and Machine Learning Toolbox               Version 10.2        (R2016a)
```

Note that the ">>" symbol indicates the MATLAB command prompt.

If you have downloaded the files from the MATLAB website onto your computer, you need to tell MATLAB where this is by using the "Set Path" button under the "Home" Tab. Once you hit the set path button, a window will pop with several options. Select "Add with subfolders…" button and navigate to the folder containing the ParticleAnalysis2 files.

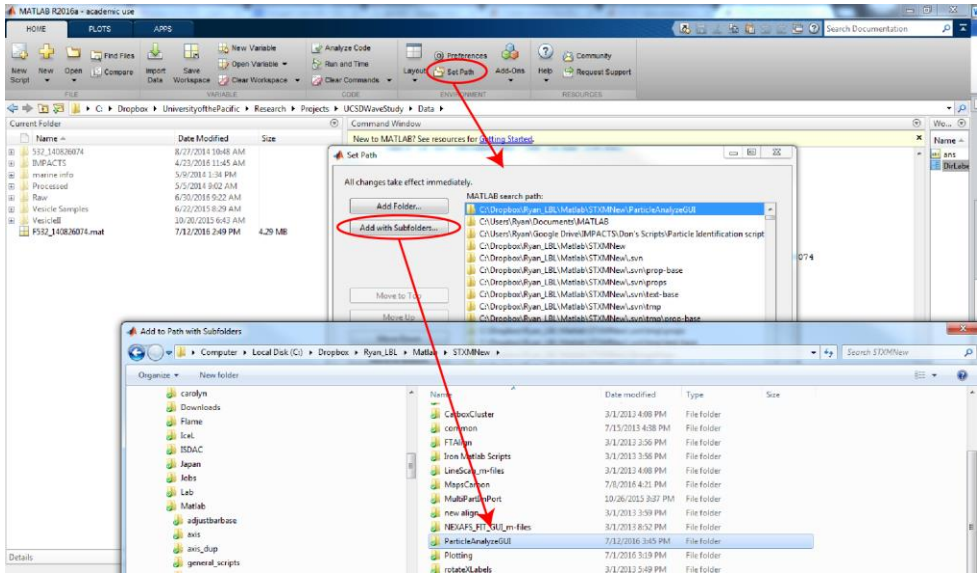


Figure 1. Window illustrating buttons necessary to set the MATLAB path.

2.2.2 ParticleAnalysis2 Startup and Overview

ParticleAnalysis2 can be opened by either double clicking on the ParticleAnalysis2 icon or by entering:

```
>> ParticleAnalysis2
```

in the MATLAB command prompt. Once you enter this command in the MATLAB prompt, the ParticleAnalysis2 GUI should appear.

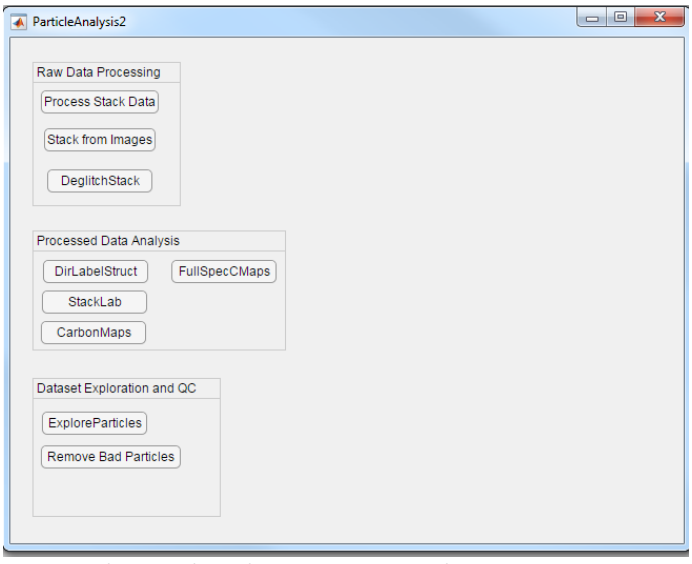


Figure 2. The ParticleAnalysis2 GUI main window.

There are three main panels in the ParticleAnalysis2 GUI that reflect the different levels of data abstraction. The first level, “Raw Data Processing”, deals with the raw ASCII files generated by

the STXM. Specifically, the Raw Data Processing routines 1) import the data into MATLAB 2) align the images to correct for the drift that occurs during acquisition 3) converts the aligned stack to optical density (OD) and 4) saves the transformed data into the same folder as the raw data. The second level of data abstraction is processed data analysis that allows the analyst to perform operations on individual stacks that were processed in the first panel. The operations involved in the second level of abstraction include, viewing of the individual stack data, producing composition maps and generating a full *data set*, which leads to the third abstraction in the third panel. In the third panel, the analyst can view all of the particles from each of the samples and perform operations on these particles such as view the raw data and flag them for removal from the data set. The remainder of the manual is devoted to the use of the routines contained in each of these three main panels.

3.1 Raw Data Processing

The following sections provide detail on the buttons contained within the Raw Data Processing panel in the ParticleAnalysis2 GUI main window (Figure 2).

3.1.1. Process Stack Data

The most essential type of data processing used for particle analysis is the processing of raw stack data. Stacks may contain anywhere from 2 to hundreds of single energy images. The Process Stack Data button allows the user to import, align, and save this stack data in individual .mat files that can be used later (Processed Data Analysis).

To process (import, align, and save) stacks:

1. press the "Process Stack Data" button in ParticleAnalysis2 main window (Figure 2). This brings up a screen entitled "Choose Directories to Process" (Figure 3, below)
2. highlight the stack folder(s) you want to process and press the "Add→" button.
3. When you have selected all the folders you would like to process press the "Done" button.
4. Choose whether or not to suppress figures for each folder by clicking "yes" or "no" in the dialog box. If you choose not to suppress figures, the program will attempt to run CarbonMaps.m and produce the figures shown in Figure 5.

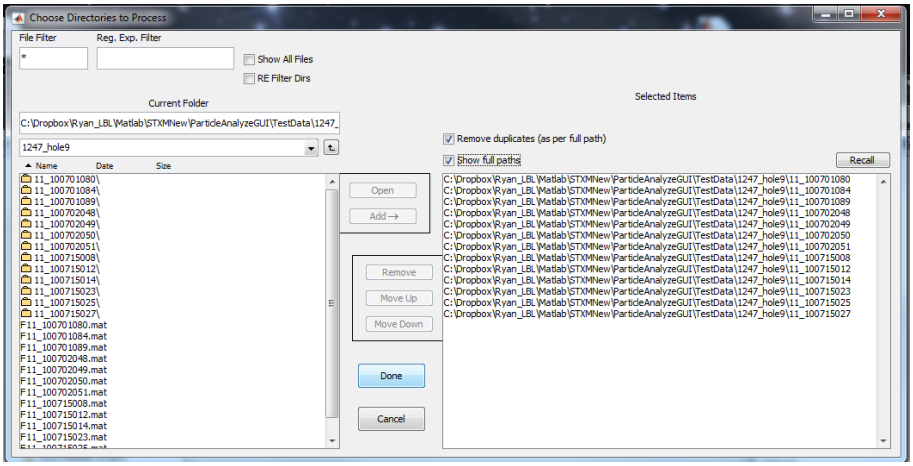


Figure 3. Folder selection window associated with the "Process Stack Data" button.

Once the file processing starts, you should see the header file names start to appear on the matlab prompt (Figure 4). After each stack has been processed, you should see that it is saved in the same directory as the raw stack folder (Figure 4). If a crash occurs, it is likely due to the format of the input data; you can see the filename of the last datafile processed at the bottom of the file list in the MATLAB prompt (Figure 4).

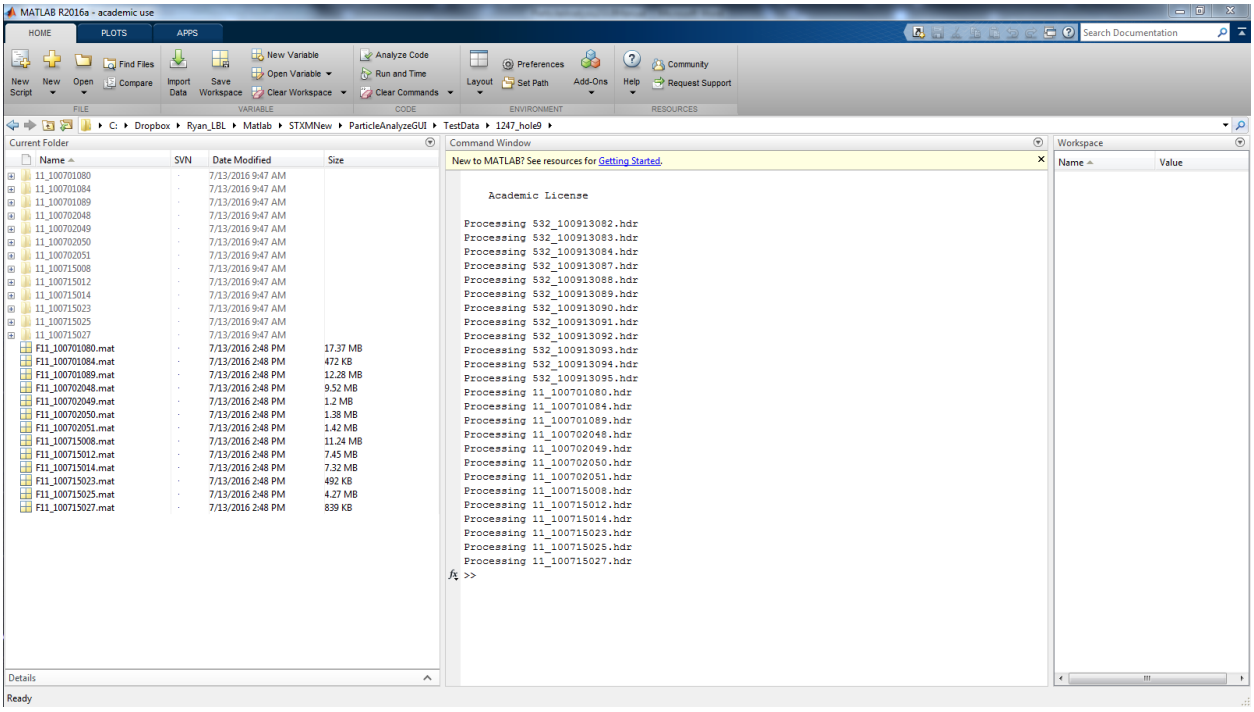


Figure 4. Main MATLAB window showing processed file list in the command window and the saved .mat files in the current folder window.

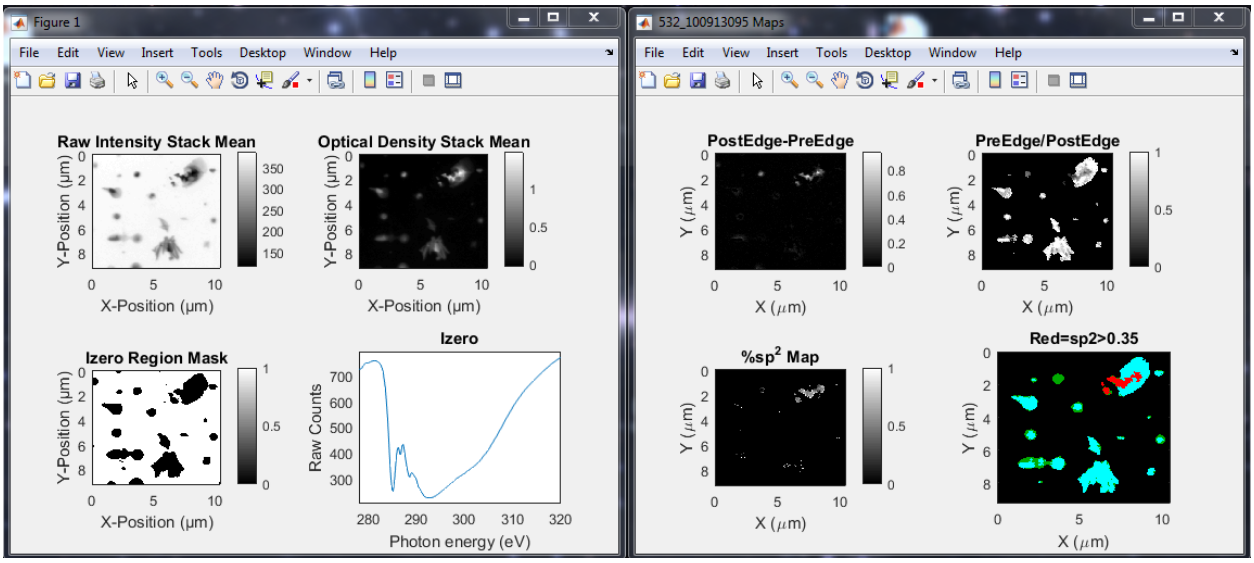


Figure 5. Figures generated when the option not to suppress figures during stack data processing.

The .mat files are named according to the original stack folder name generated by the microscope; it is these files that you can use with the second function set in ParticleAnalysis2. You can view the contents of the .mat files generated by double clicking on them in the MATLAB "current folder" window (lefthand panel, Figure 4). To view the data structure (Snew) at the MATLAB command line:

```
>> Snew

Snew =

eVenergy: [111x1 double]
Xvalue: 10.4939
Yvalue: 9.2457
spectr: [231x260x111 double]
particle: '532_100913095'
Izero: [111x2 double]
```

This data structure, S, contains a vector of energy points (`S.eVenergy`), the width of the image (`S.Xvalue`), the height of the image (`S.Yvalue`), the aligned stack with pixel values converted to optical density (`S.spectr`), a data file indicator (`S.particle`) and the I_0 spectrum (`S.Izero`). To access any of these variables at the command line, simply type "`S.variable`". For example to access the `Xvalue` variable:

```
>> S.Xvalue

ans =

    8.3500
```

3.1.2 Stack from Images

Sometimes it is necessary to combine individual single particle images to make a single stack. For this operation:

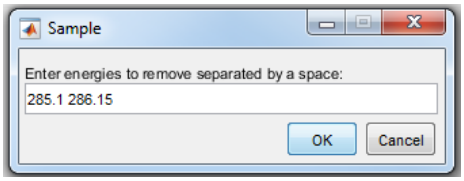
1. Press the "Stack from Images" button.

2. Select all .hdr and .xim files (using Shift+left click) to be combined.
3. Press "Open"

Once the above steps are complete, a figure showing the average raw intensity image, average optical density image, I_0 region mask and I_0 spectrum (left hand panel of Figure 5) is generated. Furthermore, as with "DeglitchStack" and "Process Stack Data" buttons, a .mat file containing the data structure is saved as described in section 3.1.1.

3.1.3 Deglitch Stack

Occasionally it is necessary to remove an image from a stack due to a focusing problem at one or more energies. For this, we use the "DeglitchStack" button:



1. Press the "DeglitchStack" button.
2. Select data file to remove image from
3. Enter the energy points separated by a space.
4. Press "OK". The data file is processed in the same manner as described in 3.1.1

3.2 Processed Data Analysis

Now that .mat files are generated for each stack, we can perform some basic operations including stack exploration (StackLab), carbon mapping (CarbonMaps), full spectral carbon mapping, and dataset generation (DirLabelStruct). This section covers these operations.

3.2.1 STACKLab

STACKLab was the first MATLAB GUI written for STXM data and has been fantastically documented by its author Tobias Henn[2a]. The "StackLab" button in the ParticleAnalysis2 GUI simply allows the user to run STACKLab on the data structures generated in section 3.1. Of course, after loading the data file, the analyst can simply type:

```
>> STACKLab(S)
```

in the MATLAB command window to bring up the STACKLab stack exploration GUI. Alternatively, from the ParticleAnalysis2 GUI,

1. Press the "StackLab" button.
2. Double click on the data file you wish to view.

After these two steps, the STACKLab window will appear. You may refer to the STACKLab documentation[2a] to learn more about generating spectra and images from STACKLab.

3.2.2 CarbonMaps

To generate composition maps at the Carbon K-edge as described in (REF) using the ParticleAnalysis2 GUI,

1. Press the “CarbonMaps” button
2. Double click on an appropriate .mat file generated using the routines described in section 3.1 of this document

At this point a carbon maps figure (shown in on the right panel of Figure 5) is shown. To run CarbonMaps on the data structure Snew from the MATLAB command window, one could simply type:

```
>> Snew=CarbonMaps(Snew)

Snew =

eVenergy: [111x1 double]
Xvalue: 10.4939
Yvalue: 9.2457
spectr: [231x260x111 double]
particle: '532_100913095'
Izero: [111x2 double]
TotC: [231x260 double]
sp2: [231x260 double]
SootEccentricity: [3x1 double]
SootMajorAxisLength: [3x1 double]
SootMinorAxisLength: [3x1 double]
SootConvexArea: [3x1 double]
SootArea: [3x1 double]
LabelMat: [231x260 double]
PartLabel: {1x17 cell}
PartSN: [17x1 double]
Size: [1x17 double]
CompSize: [17x4 double]
PartDirs: {17x2 cell}
RGBCompMap: [231x260x3 double]
Maps: [231x260x3 double]
BinCompMap: {1x3 cell}
```

You can see that many new fields were added to the data structure! To help you figure out what each of these fields are, type:

```
>> help CarbonMaps
```

3.2.3 FullSpecCMaps

While the CarbonMaps routine produces maps from a minimal set of images (namely 278, 285.4, 288.6, and 320 eV), FullSpecCMaps operates on a carbon stack collected using high spectral resolution. A high resolution carbon spectrum can be scrutinized for a few other components such as potassium and carbonate. FullSpecCMaps is executed in exactly the same way as CarbonMaps and example output is shown in Figure 6. These same maps were presented in a study of particles from Sacramento, and a discussion of some aspects of that data can be found there[4].

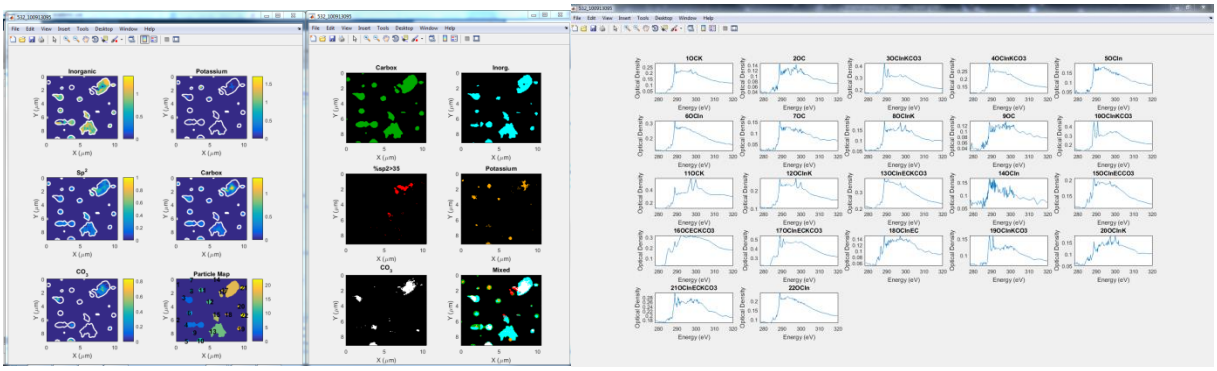


Figure 6. Output figures from FullSpecCMaps

3.2.4 DirLabelStruct

At some point, you will have processed all of your data into .mat files and performed some basic qualitative analysis. Typically, statistics need to be built up on a particular set of samples and this is where DirLabelStruct comes in. DirLabelStruct basically runs CarbonMaps, crops out individual particle maps/images, performs morphological operations, etc. on each of the sample directories. The output to DirLabelStruct contains reduced data in the form of a data structure. Lets perform DirLabelStruct on the sample data directories we created in 2.1.1:

1. Press the "DirLabelStruct" button. When this is done, the dialog box in Figure 7 shows up.
2. Enter your choice for the sp2 threshold. Typically 35% is used for the detection of elemental carbon/soot/black carbon
3. Give your data set a name
4. Choose which cropped image type to use. The RGB option gives binary maps that are produced from the CarbonMaps routine while the SVD option gives the singular value decomposition using OC, IN and EC region spectra as input to the SVD routine.
5. Press "GO!" and the dialog
6. The window shown in the left hand panel of Figure 7 appears. Highlight the sample directories that you want to process on the lefthand side of the dialog box and press "Add".
7. Press "Done".
8. A dialog box appears entitled "choose a place to save output". Navigate to an appropriate folder and save the output. I suggest that you save the output in a logical location rather than just placing it on the desktop or some other location of high entropy. For this example, we will choose:
   C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData
9. Press "Done"

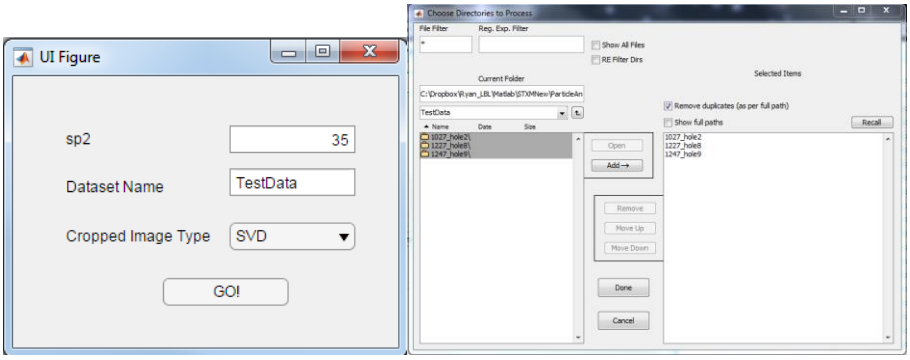


Figure 7. Dialog boxes invoked upon DirLabelStruct button press.

At this point, the routine goes to work and you can view the status on the MATLAB command window:

```
…
…
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100702050.mat
# of Stacks = 16, # of Maps = 55
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100702051.mat
# of Stacks = 17, # of Maps = 55
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715008.mat
this is not the carbon edge
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715012.mat
this is not the carbon edge
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715014.mat
this is not the carbon edge
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715023.mat
too few images for CarbonMaps
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715025.mat
this is not the carbon edge
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F11_100715027.mat
too few images for CarbonMaps
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F532_140825055.mat
# of Stacks = 18, # of Maps = 55
C:\Dropbox\Ryan_LBL\Matlab\STXMNew\ParticleAnalyzeGUI\TestData\1247_hole9\F532_150618084.mat
# of Stacks = 18, # of Maps = 56

DirLabelS =

                LabelCnt: [4x3 double]
                PartSize: {[1x345 double]  [1x748 double]  [1x66 double]}
                   label: {[1x345 double]  [1x748 double]  [1x66 double]}
                  CmpSiz: {[345x4 double]  [748x4 double]  [66x4 double]}
              SootCarbox: {[1x345 double]  [1x748 double]  [1x66 double]}
             TotalCarbon: {[1x345 double]  [1x748 double]  [1x66 double]}
                  Carbox: {[1x345 double]  [1x748 double]  [1x66 double]}
                     Sp2: {[1x345 double]  [1x748 double]  [1x66 double]}
                  OutRad: {[11x5 double]  [11x5 double]  [11x5 double]}
                  RadStd: {[11x5 double]  [11x5 double]  [11x5 double]}
                 SingRad: {{1x5 cell}  {1x5 cell}  {1x5 cell}}
            SootDistCent: {[1x23 single]  [1x139 single]  [1x22 single]}
    SootDistCentInscribed: {[1x23 single]  [1x139 single]  [1x22 single]}
                  SootEcc: {[26x1 double]  [191x1 double]  [35x1 double]}
                  SootMaj: {[26x1 double]  [191x1 double]  [35x1 double]}
                  SootMin: {[26x1 double]  [191x1 double]  [35x1 double]}
                 SootCvex: {[26x1 double]  [191x1 double]  [35x1 double]}
                 SootArea: {[26x1 double]  [191x1 double]  [35x1 double]}
            CroppedParts: {{345x1 cell}  {748x1 cell}  {66x1 cell}}
              ImageProps: {[345x4 double]  [748x4 double]  [66x4 double]}
                 PartDirs: {3x1 cell}
                   PartSN: {[345x1 double]  [748x1 double]  [66x1 double]}
```

The DirLabelS data structure contains many outputs that you can learn more about by entering the following into the MATLAB command prompt:

```
>> help DirLabelMapsStruct
```

The DirLabelMapsStruct is the data set that can be visualized in section 3.3.

3.3 Dataset Exploration and QC

To perform any of the functions in the Dataset Exploration and QC panel, you must have generated the DirLabelS data structure covered in section 3.2.4 of this manual.

3.3.1 Explore Particles

To visualize all particles in any particular sample:

1. Press the "ExploreParticles" button
2. A dialog box entitled "Pick a .mat file with the processed dataset" appears. Select the .mat file generated in section 3.2.4 of this manual.
3. A dialog box requesting you to choose a sample appears. You can select one, a select few, or all samples to show the cropped particles.
4. Press "OK" when your selection is complete. This will bring up as many figures as there are samples that you selected. Each figure contains all of the cropped particles for the selected dataset. An example of of this figure is shown in Figure 8.
5. The figure that is brought up is interactive. This means that you can click on any particle to bring up a dialog box that asks you: "What would you like to do?". Your options are to:
   a. "Explore": this runs CarbonMaps and STACKLab so that you can view the raw data and CarbonMaps output.
   b. "Delete Particle": this stores the particle serial number(s) as a vector in the DirLabelS data structure on your computer. The field in the DirLabelS structure is called "ToDelete". Note that the particle will not be deleted right away.

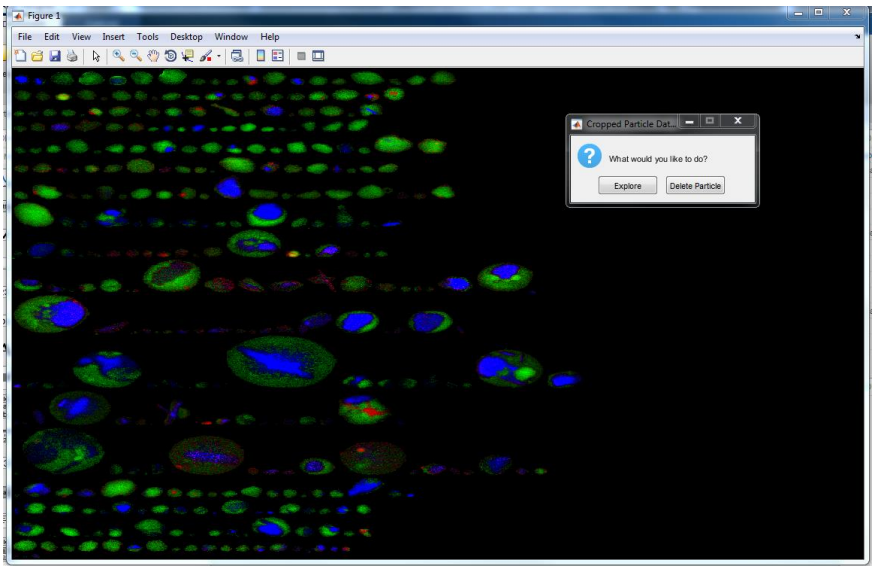


Figure 8. An example of the figure generated upon "ExploreParticles" button press.

3.3.2 Remove Bad Particles

If you selected "Remove Bad Particles", the particle serial number was saved to the DirLabelS.ToDelete field in your .mat file. In order to remove the selected particle(s) from your dataset, you must press the "Remove Bad Particles" button in the ParticleAnalysis2 GUI main window. Once you press the Remove Bad Particles button, a dialog box will pop up asking you to pick a .mat file with the processed dataset. Following the example in this manual, you can select TestData.mat. Once you select your dataset, the particles will be removed from your dataset and a new file will be saved named; following our example, the new data file will be named TestDataQC.mat

4. Concluding Remarks and Future Developments

ParticleAnalysis2 combines a number of routines used frequently in our lab and by our collaborators for a number of years. There are many more routines that can be easily included in this GUI, and I foresee at least a few more releases. Moreover, the GUI and manual were somewhat hastily written, so I foresee some refinement in the very near future. Things on our immediate "to do" list include principle components analysis/clustering for individual stacks, spectral averaging and deconvolution, and query based searching and statistical analysis on the DirLabelS structure. More can also be done to generalize this GUI to work with data at other edges or even from other instruments (such as SEM). If you have any ideas or codes you would like to share, we are always open to suggestions and collaboration.

Acknowledgements
I would like to thank those who provided code to this project. Tobias Henn, of course, invented STACKLab, developed the SVD routine, and contributed to many of the early codes still used today. The original particle cropping code was copied directly (and modified) from a matlab forum post by Anton Semechko. Tolga Birdal's code for calculating the maximum inscribed circle was also crucial for the morphological analysis. Last but not least I would like to thank my

many mentors through the years. There are probably lots of people I am forgetting, and, for that, I sincerely apologize.

References:

1. Kilcoyne, A. L. D.; Tyliszczak, T.; Steele, W. F.; Fakra, S.; Hitchcock, P.; Franck, K.; Anderson, E.; Harteneck, B.; Rightor, E. G.; Mitchell, G. E.; Hitchcock, A. P.; Yang, L.; Warwick, T.; Ade, H., Interferometer-controlled scanning transmission X-ray microscopes at the Advanced Light Source. *Journal of Synchrotron Radiation* **2003,** *10*, 125-136.
2. (a) Henn, T. R.; Moffet, R. C. STXM data analysis script collection - STACKLab. https://www.mathworks.com/matlabcentral/fileexchange/24006-stxm-data-analysis-script-collection-stacklab; (b) Moffet, R. C. STXM Spectromicroscopy Particle Analysis Routines. http://www.mathworks.com/matlabcentral/fileexchange/29085-stxm-spectromicroscopy-particle-analysis-routines.
3. Moffet, R. C.; Henn, T.; Laskin, A.; Gilles, M. K., Automated Chemical Analysis of Internally Mixed Aerosol Particles Using X-ray Spectromicroscopy at the Carbon K-Edge. *Anal Chem* **2010,** *82* (19), 7906-7914.
4. Moffet, R. C.; Rodel, T. C.; Kelly, S. T.; Yu, X. Y.; Carroll, G. T.; Fast, J.; Zaveri, R. A.; Laskin, A.; Gilles, M. K., Spectro-microscopic measurements of carbonaceous aerosol aging in Central California. *Atmos Chem Phys* **2013,** *13* (20), 10445-10459.