IsoScope User's Guide

Introduction:

IsoScope is an open source user-friendly software package written in MATLAB to visualize and perform data analysis of Mass spectrometry imaging (msi) data. Although, throughout the software development process, it was designed and only tested to work with Bruker's solariX XR FTICR mass spectrometer of designated data format, it can also take the standard .imzML data format, but may require additional testing for compatibility with data generated from other instrument. Like many other commercial and open-source MALDI image software, IsoScope has the basic functionalities of data parsing, processing, automatic peak detection, image browsing, mass spectrum browsing, ROI selection and data extraction, batch processing of a list of m/z, image overlay and registration, image segmentation, and more. In addition, IsoScope has the unique feature of analyzing isotope labeled imaging data (including 13C, 15N, 2D and 13C15N double tracer), providing biological insightful views of enrichment and labeling fractions of the tissue sample. All data analysis features are explained in further detail in this User's Manual provided with the application.

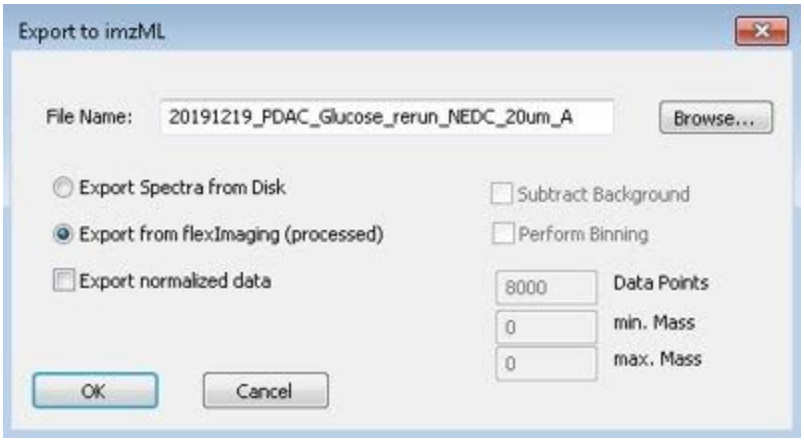
Section 1: imaging data parsing and preparation

Imaging data needs to be converted to '.mat' file format before being loaded into IsoScope. This '.mat' file serves as an intermediate temp file, which is the processed data (only containing the centroid peak list) and therefore is much smaller in size. This brings the convenience especially when the original data size is very large and frequent reloading of the same data is needed. The '.mat' file contains a matlab data structure named 'msi', which is defined for IsoScope's own use (see appendix for more details). To convert to '.mat', there are two methods to choose from:

Method 1 (slow): .imzML/.ibd → .mat

This method takes the fleximage processed .imzML file of standard msi format, performs peak detection of each mass spectrum on the fly and converts the data into matlab data structure. This method is slow for large data set, because each of the steps mentioned below: 1) load into fleximaging 2) export to imzML, 3) peak detection are dealing with the large data size that can take > 1 hour.

1. Open fleximaging, load imaging data (.mis), wait until the loading completes
2. From the main menu, select File → export to imzML, choose Export from flexImaging (processed). Note: If Export normalized data (optional) is checked, pay attention to which normalization method is used in fleximaging (None, TIC or RMS). The intensity scales will change accordingly.



3. Run IsoScope, click the ".ibd" icon  to load imzML parsing tool
4. Load in .imzML or .ibd file (same file name, either one will work)
5. Preview the data (optional) and choose the signal threshold if further data reduction is desired.
6. One click convert to create .mat

Method 2 (fast): "peaks.sqlite" → .mat

This method takes advantage of the peaks.sqlite file inside the private .d data folder that contains the already processed data after peak detection by the vendor's software. All it does is data format conversion so that matlab application can easily access the data. This data conversion is fast and usually takes only a few minutes.

1. Run the python code "quick_parse.py" located in /IsoScope/imzML parsing tool (it's convenient to load the code in spyder and click run),
2. When a file dialog pops up, select the .mis file from the imaging data folder.
3. Wait until 'Done', '.mat' file will be automatically created with the same filename as .mis.

More description:

Method 1 is aligned with the standard .imzML

Section 2: Image extraction & visualization

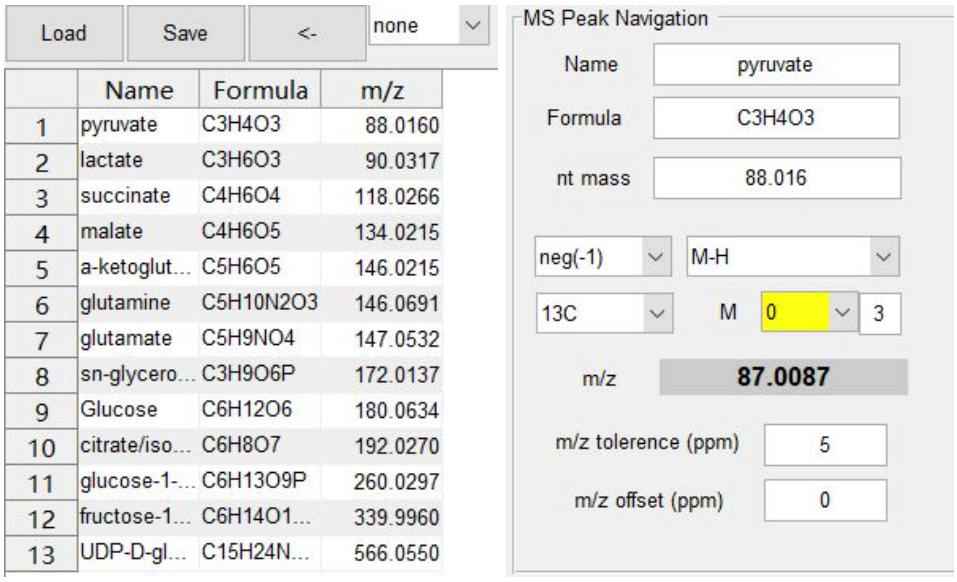
I. Load data (.mat)

After the .mat file is created, using the first icon  to load into IsoScope. An image will appear with the default initial setting. For multi-file selection, all the .mat files must come from the same coordinate (scans on the same slide, but saved separately), and they will be integrated into one image.

II. m/z peak selection.

This could be done by either of the following two methods:

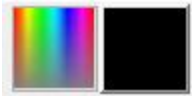
- using the MS peak Navigation tool to manually define a peak
- click on a row from the pre-loaded peak list table (note: it provides the neutral mass, formula and name, which will update the values in MS peak navigation).



Besides entering the neutral monoisotopic mass, you will also need to specify the charge state(e.g., +1 or -1), ion format (parent ion or common adducts), and isotopes (M=0 means without isotope. the number to its right is the maximum allowed isotope number, which is equivalent to C number if 13C is chosen, for instance). m/z shows the targeted m/z calculated based on the inputs above, from which, image @m/z will be extracted and displayed in the main axe of the image tab..

III. Color scheme and color contrast adjustment

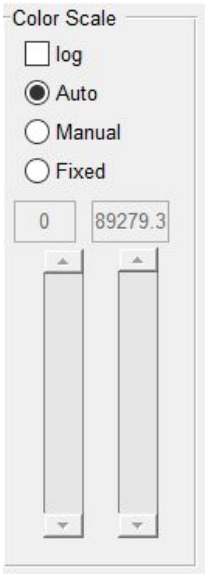
A. Color schemes



- Click the first rainbow button to select from a list of predefined colormaps, or you can customize your own colormap using the tool provided therein.
- Click the second color button to specify the background color of your image (color for pixels not covered by the laser scan), Sometimes, it's important to use different color to discriminate between pixels with zero signal and pixels out of range(N/A))

- A new button is added at toolbar  for selecting monocolor colormap, with black background color as default.

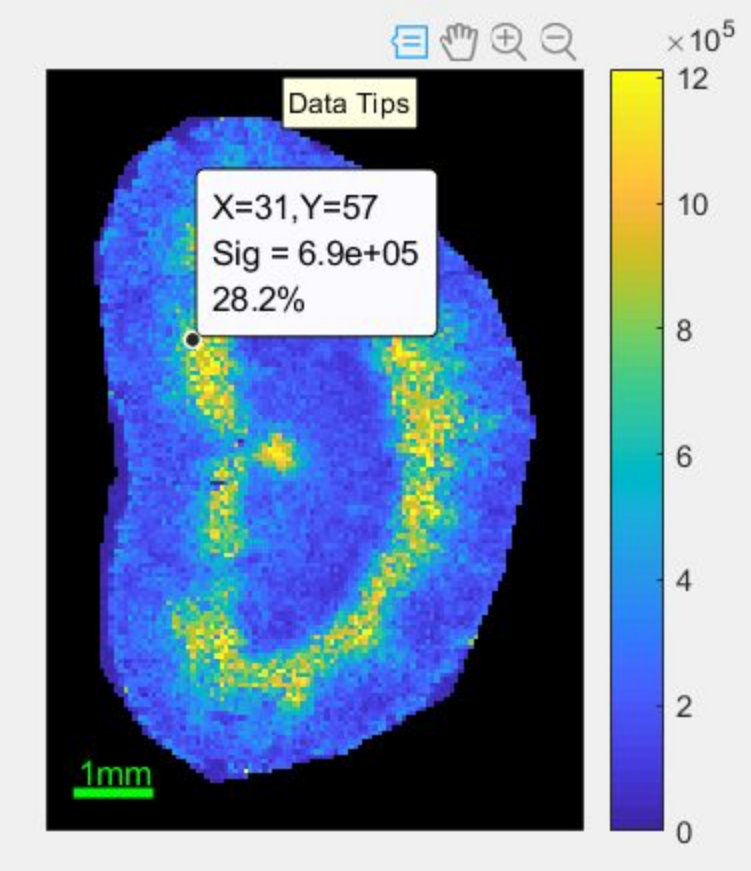
B. Color contrast and scale adjustment



There are three modes to adjust the color scales:

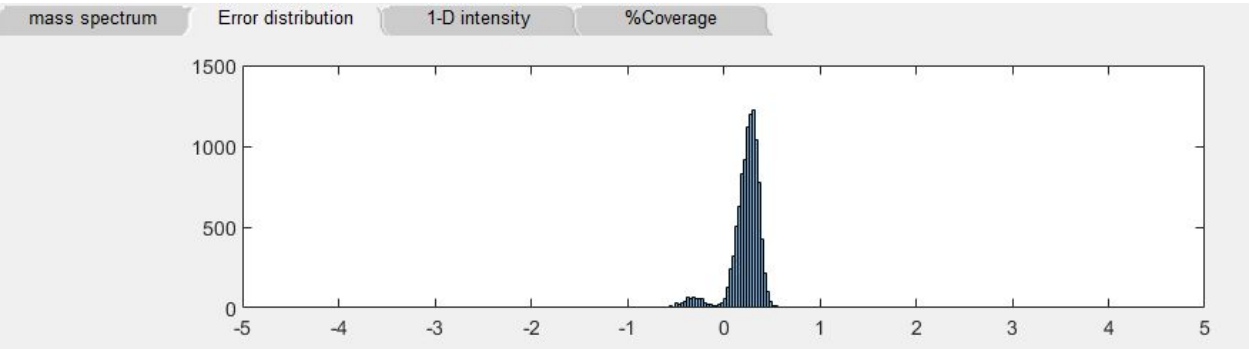
- Auto mode(recommended): sets 99% percentile to be the saturated intensity in order to remove hot spots (top 1%), which usually provides the most balanced overview for color contrast.
- Manual mode: allows users to use the two scroll bars to adjust the lower and upper thresholding intensities for the color mapping (lower bound>=0, upper bound<=top signal, which varies by m/z), which provides more flexibility.
- Fixed mode: allows user to enter arbitrary fixed numbers above the two scale bars for the lower and upper bounds. Upper bound is not restricted by the top signal detected (this is useful when plotting fractional images, such as users may want to fix the upper bound to be =100% all the time)

IV. Image resize, Fetch data point on the image



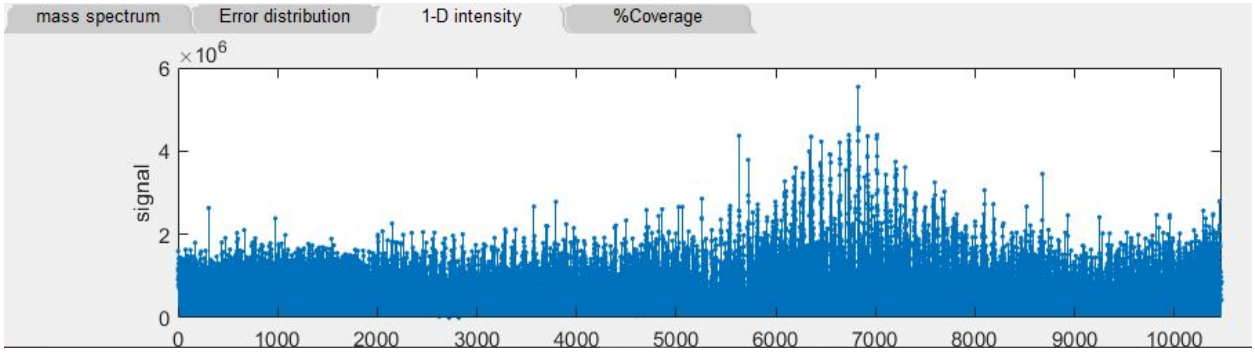
Use the built in image tool to add Data tips, pan, zoom in/out and show image of desired location/magnitude. Data tip shows the X, Y location of the cursor, its signal, as well as the percentage signal normalized to the highest. The green scale bar will automatically change accordingly. Use the toggle button  to turn the scale bar on/off.

V. Image quality assessment

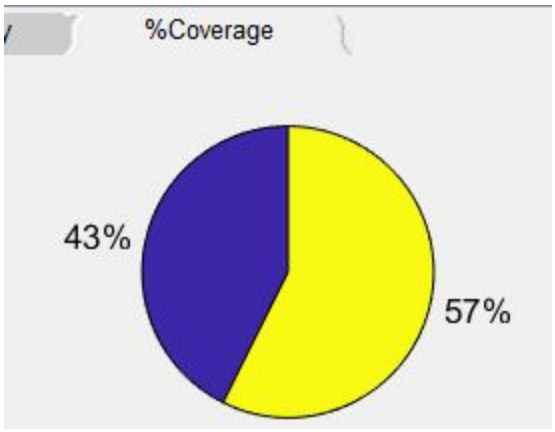


Use the error distribution tab to check the histogram of the m/z error. Ideally it should be a narrow gaussian distribution centered around 0 ppm.

- A shift of the center ppm indicates mass calibration error.
- A Doublet, if consistently observed across many different m/z, might indicate lock-mass problem; elsewise, it may be due to coexistence of multiple m/z peaks (double check on the mass spectrum)
- Non-normality or broad and flat distribution usually suggests non-real peak (noise)

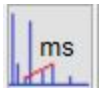


1-D intensity plot is sorted by the scanning pixel ID, which may reflect signal variations over the scanning time (long term & short term), check the average signal level, as compared to the detection limit (noise level ~2e4)

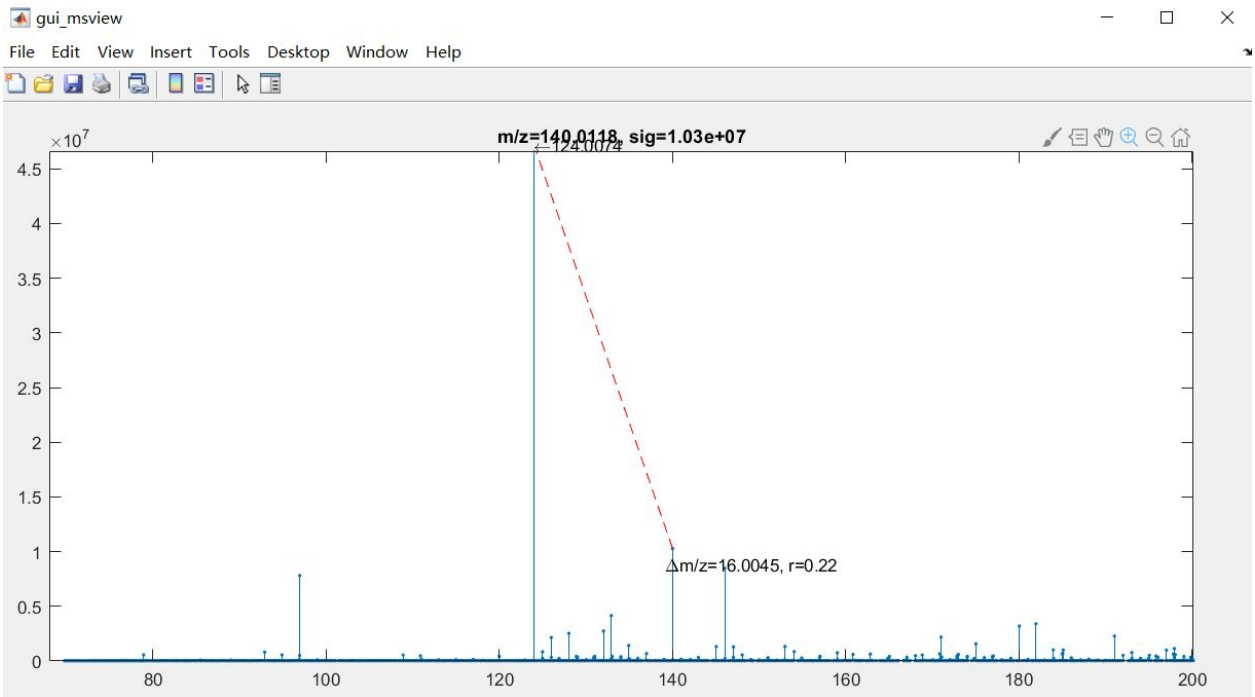


Coverage shows the percentage of non-zero signals of the image.

VI. MS spectrum browsing tool

Click the icon 

to launch the MS spectrum browsing tool. Double click on the spectrum to select the nearest peak of you mouse location as the reference peak, and move around to view nearby peaks and peak relationships (m/z different and intensity ratio).



Section 3: Isotope labeling based analysis

Clicking on the first button will calculate the image data for the selected peak m/z (M=0) as well as it's isotopes (M=1 up to M=n), based on which isotope labeling is selected (13C, 15N, 2D or 13C15N) in the MS peak navigation panel. After the calculation is complete, the next three toggle buttons (abs, ratio, frac) will be enabled, allowing the users to select which type of image to show.

- Abs: shows the images of absolute intensities.
- Ratio: shows the enrichment ratio for M = i, which equals Mi/(M0+...Mn)) for each pixel. Mi denotes the absolute intensity for m/z at M = i; Note: the final output of the ratios are after natural isotope correction.
- Frac: shows the labeling fraction image, which is the weighted average of all the labeling ratios, equal to Σ(i*Mi)/n for each pixel. Note: this image does not change with M

The dropdown box for M in the MS peak navigation panel should be used in combination here to display the images for different M and different type. For example, if the user wants to see the enrichment ratio image for pyruvate (13C M=2), set the dropdown box next to M as 2, and press the ratio button (will turn green)

If ROI selection is made (see details in the next section). The averaged values for absolute intensities, enrichment ratio after natural isotope correction and labeling fraction for each ROI will be shown sheets (3rd tab)

Image | Weight | Sheets

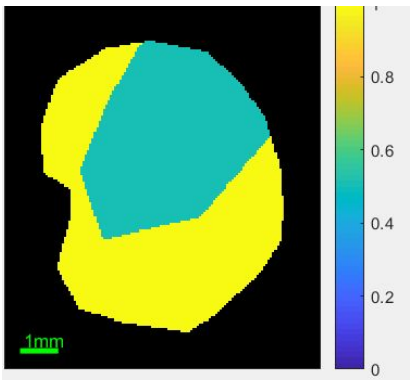
	Name	Formula	mz	r1	r2	r3	r4	r5	r6	r7
1	lactate_M0	C3H6O3	89.0244	6.8419e+04	8.3110e+04	7.1477e+04	7.1476e+04	5.1683e+04	6.9132e+04	6.8278e+04
2	lactate_M1	C3H6O3	90.0278	3.0052e+03	2.4882e+03	3.4403e+03	2.4965e+03	2.3452e+03	2.8407e+03	2.9655e+03
3	lactate_M2	C3H6O3	91.0311	1.7144e+03	2.0462e+03	1.8347e+03	1.5613e+03	1.3175e+03	1.5718e+03	1.4773e+03
4	lactate_M3	C3H6O3	92.0345	1.0177e+04	1.5852e+04	1.0740e+04	1.2237e+04	7.8297e+03	1.0095e+04	7.2230e+03

	Name	Formula	mz	r1	r2	r3	r4	r5	r6	r7
1	lactate_M0	C3H6O3	89.0244	0.6812	0.8195	0.6704	0.8081	0.5045	0.6855	0.5255
2	lactate_M1	C3H6O3	90.0278	0.0312	0.0183	0.0350	0.0208	0.0247	0.0260	0.0327
3	lactate_M2	C3H6O3	91.0311	0.0216	0.0175	0.0221	0.0164	0.0168	0.0172	0.0205
4	lactate_M3	C3H6O3	92.0345	0.1059	0.1271	0.1146	0.1008	0.0821	0.0943	0.0879

	Name	Formula	mz	r1	r2	r3	r4	r5	r6	r7
1	lactate	C3H6O3	92.0345	0.1308	0.1448	0.1411	0.1186	0.1016	0.1145	0.1124

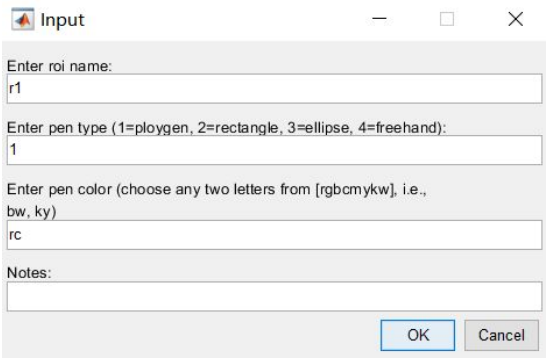
* This is A Special function upon user's request : Adding weight when calculating the labeling fraction which is normalized to (e.g., serum from LC/MS data)

Click  to draw ROI and change the weight on the image, it will redirect you to the weight tab. The calculated labeling fraction image will be normalized to this weight map (the default weight map is 1 for everywhere)



Section 4: ROI drawing and selection

Click the ROI icon to manually add an ROI, a dialog box will pop up asking for user's input

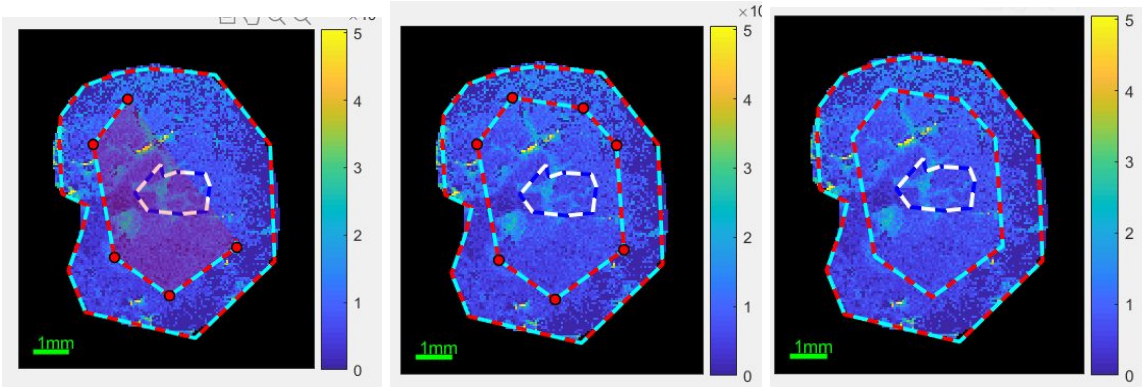


1. Enter a unique name for roi.
2. Choose pen type. The default pen is polygon, you can also choose rectangle, ellipse or freehand
3. Pen color is specified by two letters, (e.g. ‘rc’ represents ‘red-cyan’, ‘kw’ represents black-white)
4. Notes: enter anything.
5.

After clicking OK, wait for the mouse to turn a cross on the image before starting drawing ROI.

- For polygon pen type, do “click -click-...” and “double click” to close the loop,
- For rectangle and ellipse pen type, start from one corner, do “drag and drop”
- For freehand, press and hold the mouse to free draw, and release the mouse to close the loop.

After ROI is drawn, you still have a chance to do some adjustment on the nodes, add/deletes notes. Do a final double click on any of the edges to confirm.



Left: During the ROI drawing, Middle: after ROI drawing is done, right: confirmed by double clicking

A new ROI item will be added to the table below the image.
You can:

1. Click on the Name to change the name;
2. Click on the PenType to do modifications of the selected ROI, nodes will appear. Use double click to confirm. The signal value, which is the mean intensity of the ROI, will update accordingly.

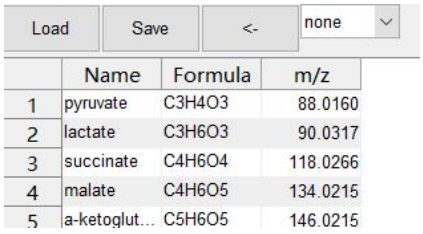
3. Click on the Notes to add or modify the notes
4. Click on [x] to delete the ROI;

	Name	PenType	Size	Signal	Notes	delete
1	r1	polygen	10015	1.0571e+06		[x]
2	r2	polygen	530	2.0956e+06		[x]
3	r3	polygen	2637	1.1067e+06		[x]
4	r4	polygen	1834	1.3149e+06		[x]
5	r5	rectangle	13932	7.9901e+05		[x]
6	r6	freehand	8398	9.6501e+05		[x]
7	r7	freehand	2378	8.6177e+05		[x]

The load and save icons next to ROI icon are used to save the user defined ROI group on the disk for future reload. It must match the image data coordinate. (Loading the previously saved ROI of a different image data will cause error)

Section 5: batch tools

Note: The use of batch tool requires at least one ROI.
The batch tool is based on the peak list provided in the table.



	Name	Formula	m/z
1	pyruvate	C3H4O3	88.0160
2	lactate	C3H6O3	90.0317
3	succinate	C4H6O4	118.0266
4	malate	C4H6O5	134.0215
5	a-ketoglut...	C5H6O5	146.0215

Use the load button to load another user defined peaklist (in .csv format). Or Use  to select one or more peaks from the current peak list. The advance use of this function will be discussed in the next section.

The first batch tool goes over all the peaks and exports their absolute intensity images and data in excel file. When group exporting the images, users will be asked to input the number of images (per row and per column) per page.

Note: The color scale for individual image will be auto adjusted, use fixed color scale instead, if you want the color scales of all output images to be comparable.

The second batch tool goes over all the peaks, calculates the isotope labeled enrichment and labeling fraction ratios for each peak, and exports to an excel file.

Section 6: Targeted and untargeted analysis

I. Targeted analysis

 It is based on the peak list provided. It will calculate the intensity correlations between every two peaks, creating a correlation matrix, from which, the most related peaks (m/z) to the selected peak can be identified. Due to the high computational cost, it will only randomly sample a portion of the image pixels to speed up, which turned out to be effective. The default number is set at 500 pixels (a dialog box will pop up for user to modify this number)

Peaks will be color coded based on the image quality score. Users can use the dropdown menu to sort the peaks based on the selected ColumnName, a second selection will reverse the sorting.

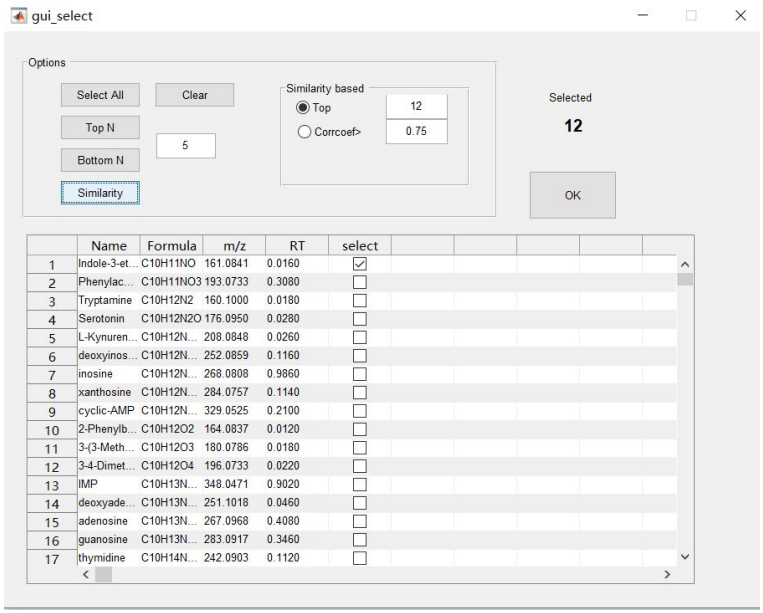
	Name	Formula	m/z	RT
1	Indole-3-et...	C10H11NO	161.0841	0.0160
2	Phenylac...	C10H11NO3	193.0733	0.3080
3	Tryptamine	C10H12N2	160.1000	0.0180
4	Serotonin	C10H12N2O	176.0950	0.0280
5	L-Kynuren...	C10H12N...	208.0848	0.0260
6	deoxyinos...	C10H12N...	252.0859	0.1160
7	inosine	C10H12N...	268.0808	0.9860
8	xanthosine	C10H12N...	284.0757	0.1140
9	cyclic-AMP	C10H12N...	329.0525	0.2100
10	2-Phenylb...	C10H12O2	164.0837	0.0120
11	3-(3-Meth...	C10H12O3	180.0786	0.0180
12	3-4-Dimet...	C10H12O4	196.0733	0.0220
13	IMP	C10H13N...	348.0471	0.9020
14	deoxyade...	C10H13N...	251.1018	0.0460
15	adenosine	C10H13N...	267.0968	0.4080
16	guanosine	C10H13N...	283.0917	0.3460
17	thymidine	C10H14N...	242.0903	0.1120
18	Ribothymi...	C10H14N...	258.0852	0.8300
19	IDP	C10H14N...	428.0134	0.0200
20	adenosine...	C10H14N...	427.0199	0.0160
21	dAMP	C10H14N...	331.0682	0.0540
22	AMP	C10H14N...	347.0631	1

Select a peak of interest, and then click  to select a subset from the peaklist,

- Manual selection by click the checkbox
- Click Top N or Bottom N (specify N in the box next to it), usually do sorting before this type of selection
- Similarity: select the top few (default is 12) peaks that have similar looking images to the peak of interest.

After clicking OK, only the selected peak subset will show up in the peak list, for further analysis or batch output. Use the 

button to go back to the original peak list.



II. Untargeted analysis:

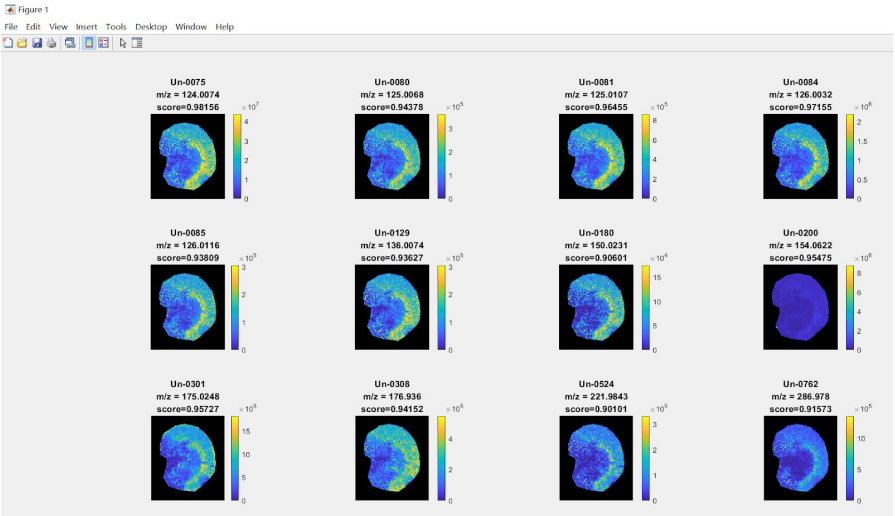
It will perform a spectra alignment and accumulation, followed by a peak picking algorithm to find the most reliable features in the dataset. An untargeted peak list will be generated in the peaklist table, however, with unknown formula and given systematic names (Un-XXXX). Then, a simple version of untargeted annotation workflow applies, which provides suggested annotations based on the common peak-peak mass differences (e.g., 13C, 15N isotopes, Na-H, K-H HCl adducts) and database m/z matching. Those with Formula names have the database m/z match.

Load | Save | <- | none

	Name	Formula	m_z	score	anno
1	Un-0001		73.5382	0.3440	
2	Un-0002		74.2405	0.3420	
3	Un-0003		74.5319	0.3700	
4	Un-0004		74.8046	0.3200	
5	Un-0005		77.5052	0.2880	
6	Un-0006		79.9664	1	
7	Un-0007		80.5222	0.2180	
8	Un-0008	HO3S	80.9646	0.5320	
9	Un-0009		82.4268	0.2020	
10	Un-0010	C4H6O2	86.0368	0.2320	
11	Un-0011	C3H4O3	88.0161	0.5480	
12	Un-0012		88.5197	0.2240	
13	Un-0013		88.7981	0.2580	
14	Un-0014	C3H7NO2	89.0477	0.7540	
15	Un-0015	C2H2O4	89.9953	0.2600	
16	Un-0016	C3H6O3	90.0317	0.7800	
17	Un-0017		91.9903	0.2660	
18	Un-0018		93.0418	0.2620	
19	Un-0019		93.5615	0.4820	
20	Un-0020		93.9353	1	

The remaining procedure is analogous to the targeted workflow. Below shows an example of identifying group images of the highest similarities to the selected peak of Un-0075 using the combination of : untargeted analysis 

, similarity based peak selection  followed by batch output | > .



Section 7: Image output

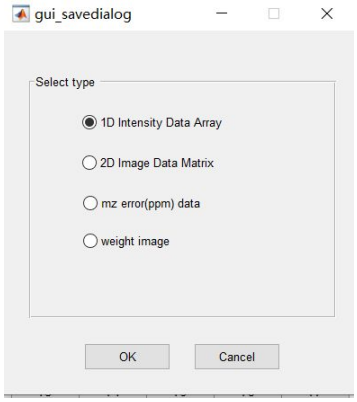
Use 

to replicate the displayed image in another window for saving. Save as '.png/.tif/.bmp/.jpg' for regular picture format or choose '.fig' for Matlab figure.

Use the toggle buttons 

to control whether ROIs or scale bars are to be displayed and saved.

Click on the  icon from the toolbar to save the image data in the numerical format as an excel table (.xlsx). You have the following options:



1. 1D intensity Data (image intensity array sorted by scan ID)
2. 2D image Data (an image intensity matrix)
3. ppm error data (array sorted by scan ID)
4. *Weight image data (weight map matrix)

These are specially defined functions, which are used for developers to debug or test before adding new functionalities. They can be used to access the msi data structure. See Appendix for more information.

Section 8: Image registration

Maldi images can be overlaid with high-resolution microscopic images for better view and more precise ROI drawing. An image registration based ROI drawing tool is built into isoscope. Click  icon from the tool bar to load the image registration gui tool:

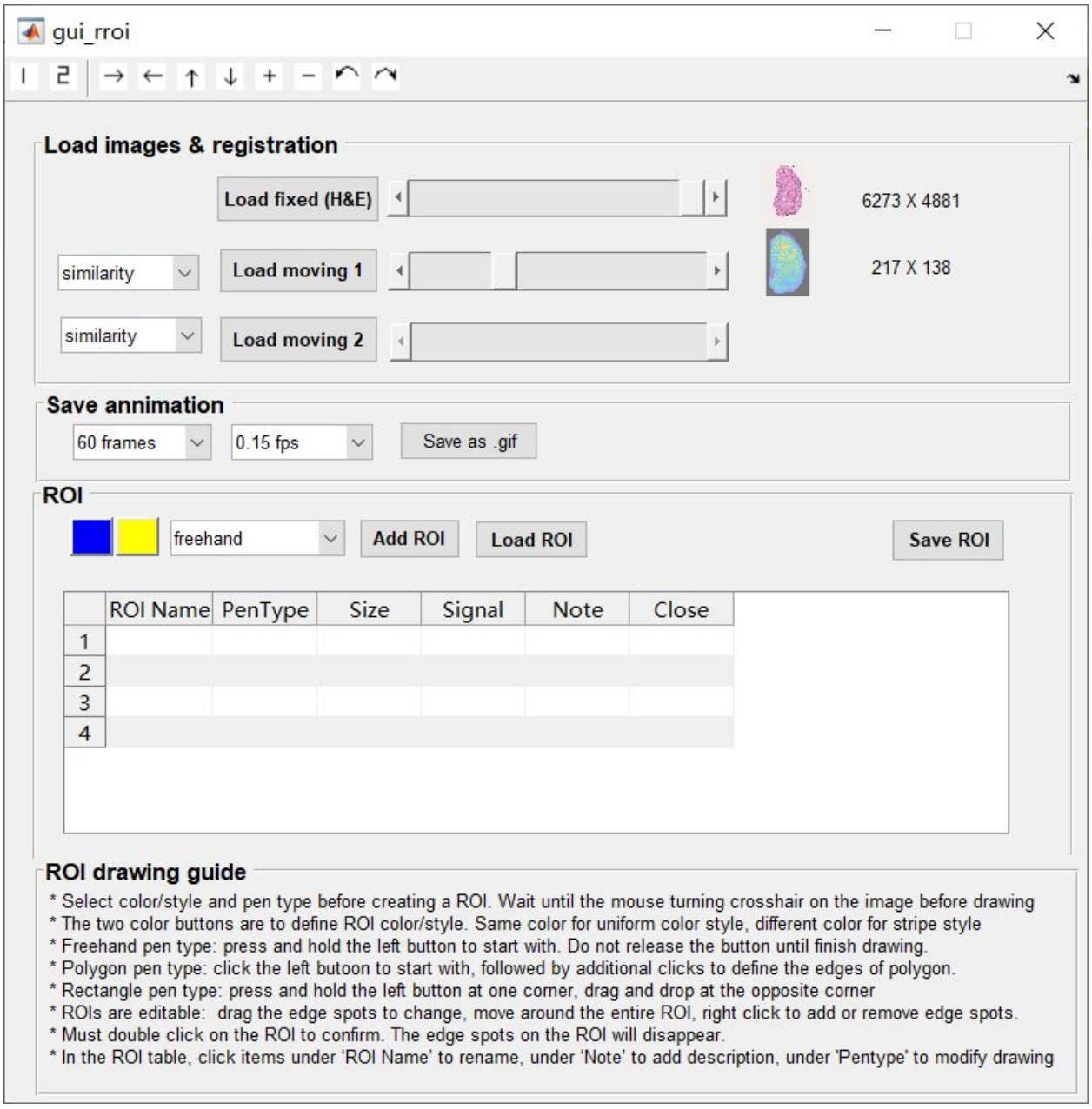
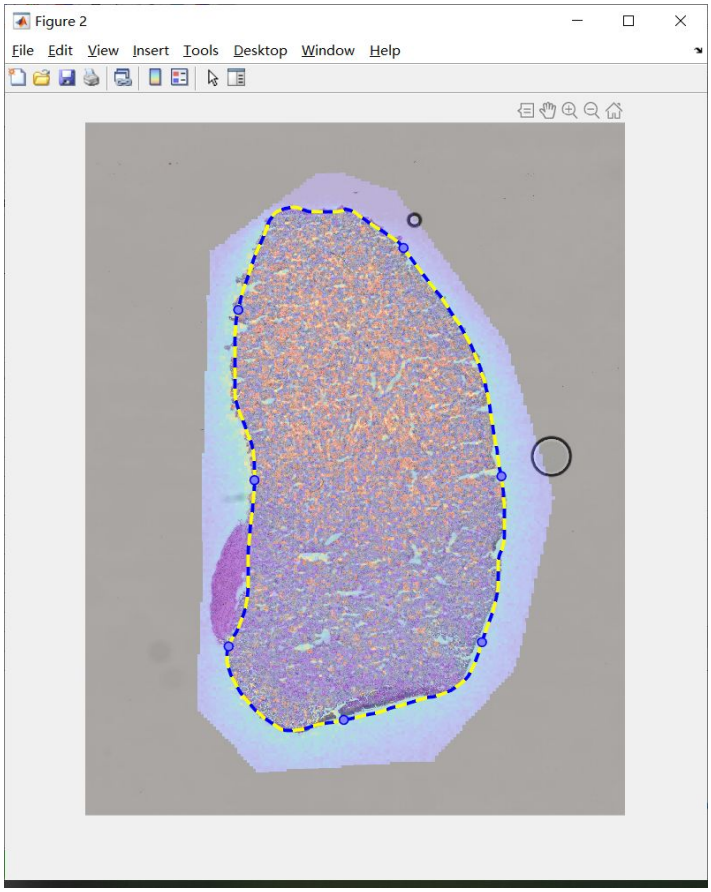


Image registration is based on manual selection of teaching points using the built-in matlab function “cpselect” that creates the transformation matrix. Users can choose from “similarity” or “affine” transformation. The high-res image should always be the fixed images to be loaded first. The MALDI image can be loaded from a pre-saved picture or directly from Isoscope if it is open. After manual teaching points selection, the blended image will be shown in a separate window and the transparencies for each can be adjusted with the scroll bar.

An ROI tool is available to select region of interest on the loaded image, either with only the high-res H&E loaded, or the overlay. For the latter, the roi groups are saved as “.rroi” file, which contain both the registration result and the ROI information. This rroi also can be loaded into

isoscope. As a major application, users can load in high-res H&E image and draw ROI on it very precisely. And then,after overlay with the low-res MALDI image, the ROIs drawn on H&E can be mapped to MALDI image for data analysis. For better overlay, an intermediate layer of image (e.g., Post-MALDI fluescense image) can be added as a moving image, so that, fixed image <-> moving1<-> moving 2. This is helpful for enhancing the registration accuracy, since in some cases, direct overlay between H&E and MALDI is hard (no convincing teaching points with local characteristics can be found).



Section 9: Image Clustering

Image Clustering for MALDI data can involve various algorithms. This functionality in IsoScope is still in its early stage. The current workflow involves segmentation of the MALDI images to group pixels of similar mass spectrum in signals (high dimensional pixel to pixel distance) done by K-mean Clustering. The clustering result is highly dependent on peak picking (untargeted analysis and peak selection) and choice of K. Note that clustering will be based on the peak list shown in the isoscope gui table. So a logical procedure is to run untargeted analysis first to get a full peak list truncated at predefined signal percentage cutoff. User can further select or

deselect peaks using  to alter the peak list. Finally, click the  icon on the toolbar, select k value for K-mean and p value for PCA dimensionality reduction (can help achieve better smoothness) to perform clustering.

Appendix I: Major classed and data structures

A. Mzpk Class:

Mzpk is a class for defining an m/z peak with great flexibility. Class definition can be found in 'mzpk.m'. Below is an instance of the class as an example.

```
Mzpk with properties:

        name: 'citrate'
     formula: 'C6H8O7'
        mass: 192.0270
           T: {[1×3 table]  [14×3 table]  [1×3 table]  [14×3 table]  [5×3 table]}
           z: -1
     isoName: {'13C'  '15N'  '2D'  '13C15N'}
     isoMass: [1.0033 0.9970 1.0063]
     isoType: 1
     addType: 1
           M: 6
      offset: 0
         ppm: 5
         mz_: 197.0398
    addList_: {14×1 cell}
     MList_: {'0'  '1'  '2'  '3'  '4'  '5'  '6'}
          M_: 7
    addName_: {'M-H'}
      range_: [197.0388 197.0408]
  atomcount_: [6 0 8 7 0 0]
       maxM_: 6
```

There are 3 ways to create an instance of this class, specifying the first three properties, (i.e, name, formula, mass).
1. mypk=Mzpk(mass), where, mass is the neutral mass in double type. Name and formula will be left blank.
2. mypk=Mzpk(pk), where, pk is a structure containing pk.name,pk.formula, pk.mass
3. mypk=Mzpk(formula), where, formula is the molecular formula in char type, neutral mass will be automatically calculated and filled in, name will be identical to formula.

The rest properties can be specified separately. See details below:

T: is an array of table, containing predefined adduct types for different ion charge modes (-2, -1, 0, 1, 2).
z: is the ion charge mode (default = -1);
isoName: lists the currently supported isotope labeling types
isoType: is the index of isoName, default = 1, i.e, 13C.
addType: is the index of table items in T{i}, i= z+3.
M: isotopomer number, default = 0, i.e., parent peak. M is an integer scalar up to maxM_ which is determined by the formula and isoType. In a special case when isoType=4, M can be a two element vector, specifying isotopomers for 13C followed by 15N.
offset: added m/z shift in ppm
ppm: m/z tolerance in ppm.
mz_: dependent property, m/z of the ion shown in the mass spectrum. It's automatically calculated based on mass, M, isoType, addType, T.
addList_: dependent property, adduct names fetched from T{i}. Just for convenience.
MList_: dependent property, enumerate all the isotopomers in char.
M_: dependent property, integer index of isotopomer starting from 1. For single tracer, M_=M+1, for 13C15N double tracer, M_ is still 1-d.
addName_: dependent property, selected adduct name.
range_: dependent property, m/z range calculated from m/z_ and ppm
atomcount_:dependent property, atom counts of C,N,H,O,S,P found from formula.
maxM_: dependent property, maximum M.

B. msi data Structure:

Snapshots of the intermediate data needed by the application are stored in msi data structure. Below shows an example. msi can be accessed by:

```
msi=getappdata(handles.figure1,'msi')
```

```
struct with fields:

              fname: 'C:\Users\xxing\Documents\Github\Imaging Data\Kidney.mat'
               data: [1×9933 struct]
                res: 60
            padding: [10 10 10 10]
                 pk: [1×1 Mzpk]
               cmap: [64×3 double]
           metadata: [9933×2 double]
            imgdata: [159×112 double]
          alphadata: [159×112 double]
                ref: [1×1 imref2d]
              idata: [9933×1 double]
            errdata: [9933×1 double]
           coverage: 0.0055
              wdata: [159×112 double]
  select_idata_type: 0
           isoidata: []
          currentID: 1
                ms: [1×1 struct]
              CLim: [0 1.4928e+05]
           bgColor: [0 0 0]
          scaleobj: [1×1 Pscale]
           handles: [1×1 struct]
              imgC: [159×112×3 uint8]
          errscore: 1.2264e-04
```

msi.data is an array of structure, which stores all the parsed and processed data from raw, including metadata (x, y) and mass spectra data (peak_mz, peak_sig), see example below.
msi.res is pixel resolution in micrometer.
msi.cmap is colormap
msi.metadata makes a copy of the metadata from msi.data and is a 2d array
msi.imgdata stores the current image intensity data, depending on the mz peak settings
msi.alphadata is the mask to distinguish between pixels with and without image data.
msi.idata is the 1d intensity data
msi.errdata stores the central mz error in ppm for each pixel
msi.coverage is calculated %coverage of non-zero intensity pixels for the current image
msi.wdata is image weight mask of user's input for normalization in calculating fraction labeling
msi.isoidata stores all the isotope images (M=0,1,..N) after the batch process, including the normalized ones after natural isotope correction. It's empty if not iso processed.
msi.currentID is the pixel ID of user's selection by mouse click on the image
msi.ms is the mass spectrum of the current pixel
msi.CLim is the color limit setting calculated
msi.scaleobj is the instance of Pscale class for the scale bar shown on the image.
msi.handles is the handles of the GUI application.
msi.imgC is the color image of the original size.

1x9933 struct with 5 fields

Fields	id	x	y	peak_mz	peak_sig
1	1	536	121	*1x5498 do...*	*1x5498 sin...*
2	2	537	121	*1x5914 do...*	*1x5914 sin...*
3	3	538	121	*1x5984 do...*	*1x5984 sin...*
4	4	539	121	*1x6106 do...*	*1x6106 sin...*
5	5	540	121	*1x6118 do...*	*1x6118 sin...*
6	6	541	121	*1x4985 do...*	*1x4985 sin...*
7	7	542	121	*1x5687 do...*	*1x5687 sin...*
8	8	543	121	*1x6187 do...*	*1x6187 sin...*
9	9	544	121	*1x5739 do...*	*1x5739 sin...*
10	10	545	121	*1x5654 do...*	*1x5654 sin...*
11	11	546	121	*1x5619 do...*	*1x5619 sin...*
12	12	547	121	*1x5509 do...*	*1x5509 sin...*
13	13	548	121	*1x5819 do...*	*1x5819 sin...*
14	14	549	121	*1x5966 do...*	*1x5966 sin...*
15	15	550	121	*1x5812 do...*	*1x5812 sin...*
16	16	551	121	*1x5889 do...*	*1x5889 sin...*

C. ROI Class

ROI is a class encapsulating info needed for each ROI and methods for creating ROI, modifying ROI, calculating signals, etc. See below and example and details in ROI.m. roigrp is an array of ROI class instances, which can be accessed by:

roigrp=getappdata(handles.figure1,'roigrp')

```
ROI with properties:

      ax: [1×1 Axes]
     pen: 'polygen'
     ref: [1×1 imref2d]
     plt: [1×1 Polygon]
     tag: 'r1'
      id: 1
      BW: [159×112 logical]
    edge: [7×2 double]
     sig: 4.3841e+06
    size: 2550
  weight: ' '
    note: ''
      c1: 'r'
      c2: 'c'
```

D. pks data structure

pks stores the current peak list that isoScope is dealing with, specifically, ordering keeps track of the peak ordering changes upon sorting.

To access, use: pks=getappdata(handles.figure1,'pks')

```
struct with fields:

    header: {'Name'  'Formula'  'm/z'}
      data: {13×3 cell}
     sdata: [13×1 struct]
      pkid: 10
  ordering: [1 2 3 4 5 6 7 8 9 10 11 12 13]
    corref: [13×13 double]
  filename: 'list00.xlsx'
```

E. Pscale Class

Pscale handles the scale bar shown on the image. It automatically changes to suitable scale display and locations in conjunction with the zoom in and out tool. See Pscale.m for details. An instance of Pscale is stored in msi structure. (msi.scaleobj)

```
Pscale with properties:

       ax: [1×1 Axes]
       ln: [1×1 Line]
      txt: [1×1 Text]
      xsp: 0.0500
      ysp: 0.0500
      lnp: 0.1500
     xpos: 'left'
     ypos: 'bottom'
    color: 'g'
      xtp: 0.0100
      ytp: 0.0300
  visible: 'On'
```

Appendix II: Description of functions

To be filled in.

Appendix III: Demo of using scripts

Example 1: starting from .mat, load data, setup mz peak, calculate isoImages of user's choice, display the image.

```
% This demo script displays absolute or enrichment ratio image for Citrate 13C isotope M=6
% In step 2 and 3 choose to run either a or b, but not both.


%% Step 1, load data
fname='..\Imaging Data\kidney.mat';  % change the path to your data location
load (fname);
padding=[20,20,20,20]; % add paddings around data region after a default tight cropping
msi=msi_get_metadata(msi,padding); %initialize, get metadata,ref and alphadata
%% Step 2: Define a mass peak
mypk=Mzpk('C6H8O7');  %initiate a peak with inputting formular
mypk.name='Citrate';  %type in name, optional
%% Step 2a alternative: select a mass peak from pre-defined pks table
```

```
% fname='list00.xlsx'; %load metabolite known list
% pkid=10; %pick a peak (10th peak in the list which is citrate)
% pks=table2struct(readtable(fname)); %create a peak list structure
% mypk=Mzpk(pks(pkid));

%% Step 3 Add peak settings
mypk.z=-1; % ion mode (default is -1)
mypk.ppm=5; % m/z tolerance in ppm (defalt is 5)
mypk.offset=0;  % added m/z offset in ppm (default is 0)
mypk.isoType=1; % choose isotope type: (1=13C (default), 2=15N, 3=2D, 4=13C15N). This
setting only matters for isotopes(M>0)
mypk.M=6;  % specify which isotopomer to display (default is 0)
mypk.addType=1; % there's a predefined adduct list stored in T, which differs by ion mode,
(default is 1: no adduct).
% type mypk to display all settings of the m/z peak

%% step 4  run this code to get the isoimage data and display images for enrichment ratio
display_type=2; %1=absolute, 2=enrichment ratio, 3=labeling fraction;
msi=msi_get_idata(msi,mypk); % get intensity data
msi=msi_get_isoidata(msi,mypk); %get intensity data for isotope data (natual isotope corrected)
msi=msi_select_idata(msi,mypk.M_,display_type); %copy the selected data to msi.imgdata
%% Step 4a. alternative: run this code to get the image data for absolute intensity
% msi=msi_get_idata(msi,mypk); % get intensity data
% msi=msi_update_imgdata(msi); % update imgdata
%%  Step 5. display settings
 cmap=parula;  %specify colormap
 colorscale=[0,0.45]; %specfy colorscale, normalized to the 1
 scalebarOnOff='on';
 resize=4;  % image resize magnification factor
%% Step 6.  run code below to display the image
 msi.imgC=msi2rgb(msi.imgdata,msi.alphadata,cmap,colorscale);
 f=figure('units','normalized');
 ax=axes(f, 'units','normalized');
 imobj=imshow(msi.imgC,msi.ref,'parent',ax);
 ax.Colormap=cmap;
 ax.CLim=colorscale*max(msi.idata); %color limit
 colorbar(ax)
 scalebar=Pscale(ax);
 scalebar.visible=scalebarOnOff;
 f.OuterPosition=[0 0 1 1]; %maximize figure
 saveas(f, '..\results\fig1.png')
```