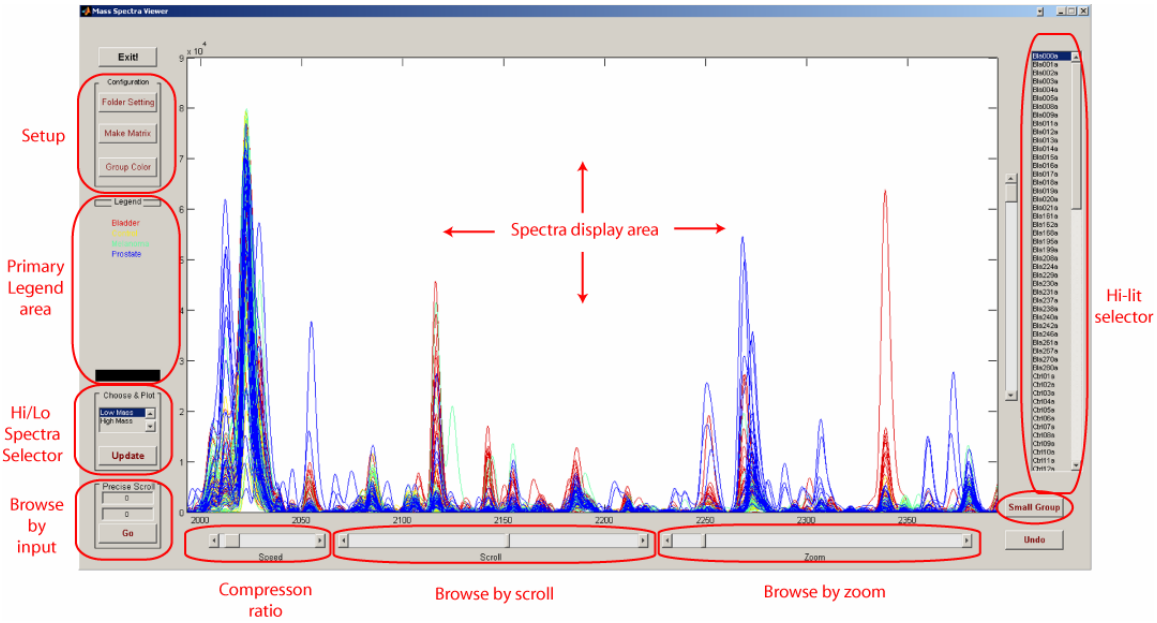
Instruction for Mass Spectra Viewer

1. A general description

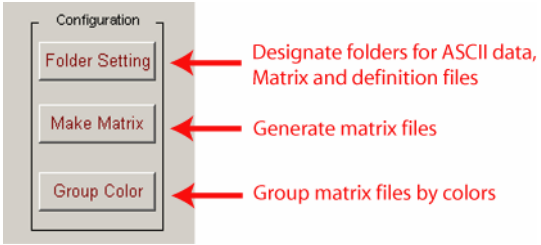


Basic functions

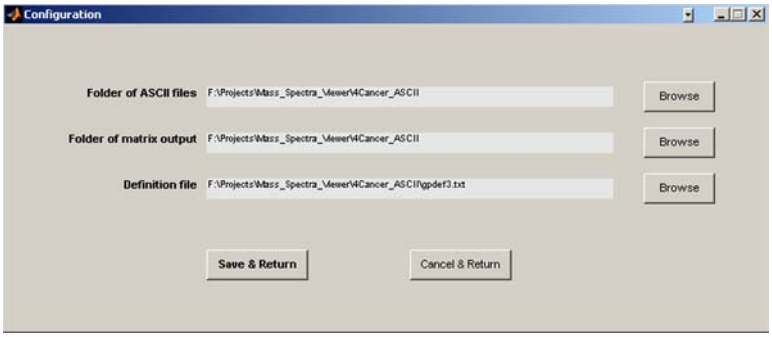
- Setup and Pre-process
- View, browser and undo
- Grouping, highlighting and labeling

2. Setup and Pre-process

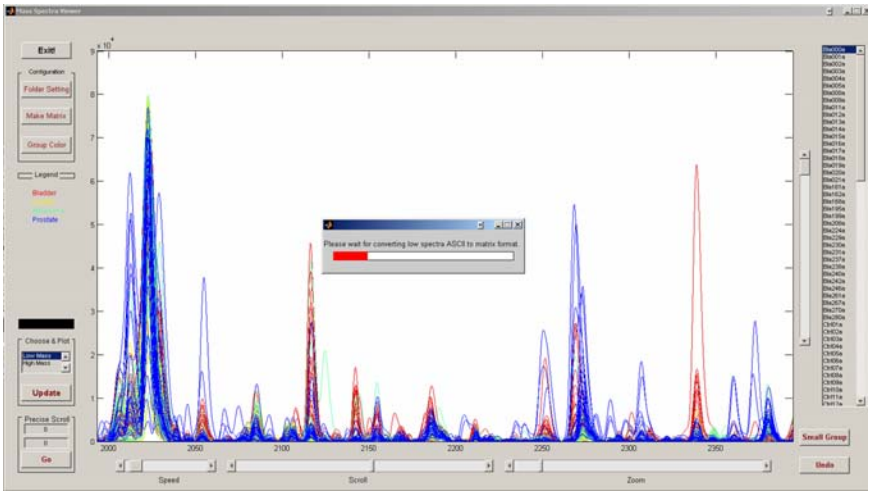
These procedures must be done before browsing the data. But most of time, they need to be run only once for a specific matrix set.



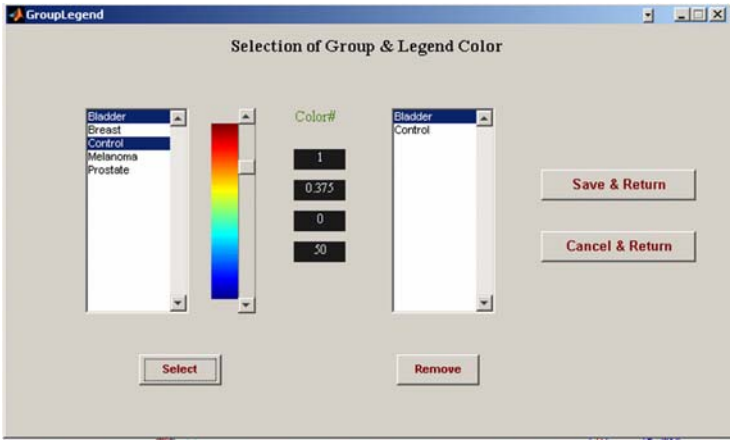
a. Folder Setting: if *cfg.txt* is found in current directory, previous settings will be loaded. Otherwise, current folders will be used instead. Click on **Browse** to find desired folders.



b. Matrix generation: click on **Make Matrix**. Program will read ASCII files and convert them into Matlab internal matrix format. A progress bar will show the percentage of files finished. Depends on number of ASCII files, it could take a while. No other operation is needed from user.

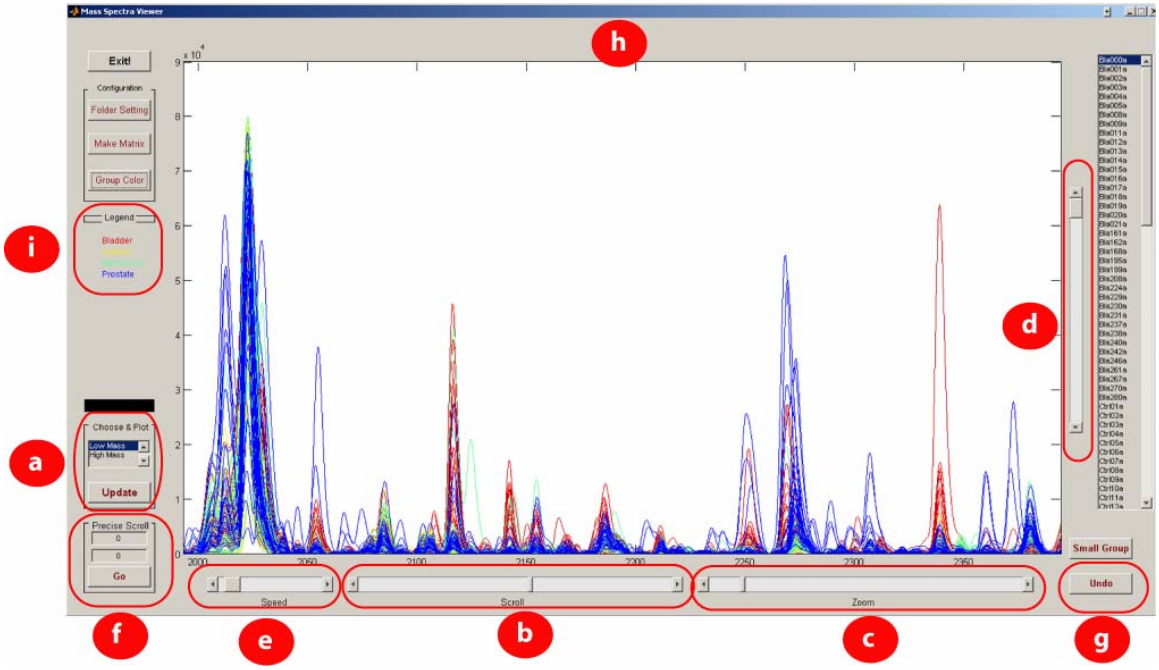


c. Group Color: You can choose one or more candidates and assign a color. Its RGB values and index in a 64-step colormap will be shown for reference. The results will be saved in an internal file *gpidx.txt*. Click on **Save** or **Cancel** to return to main window.

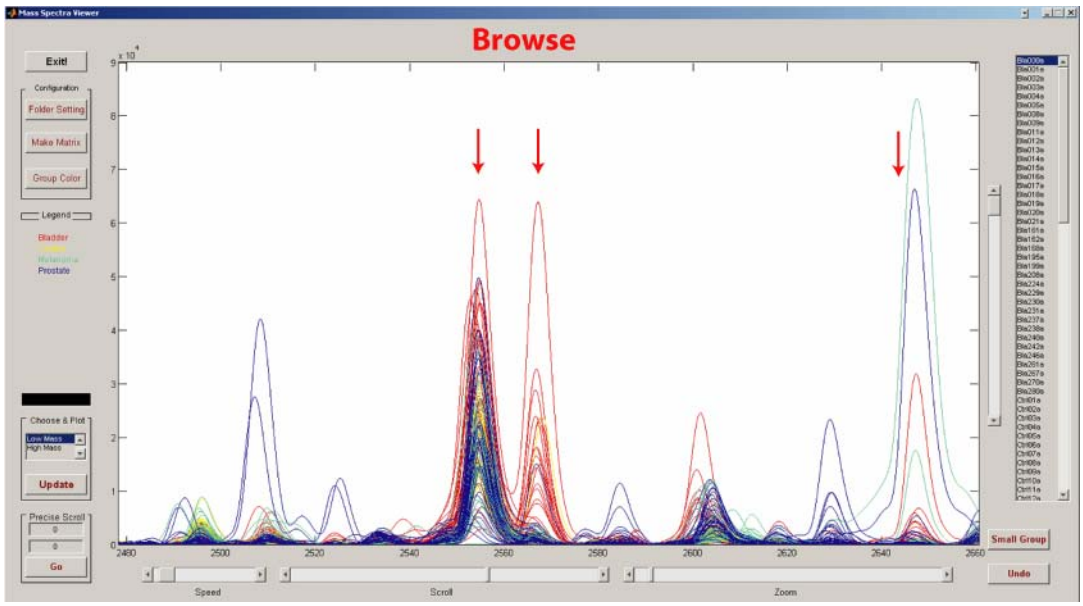


3. Browse

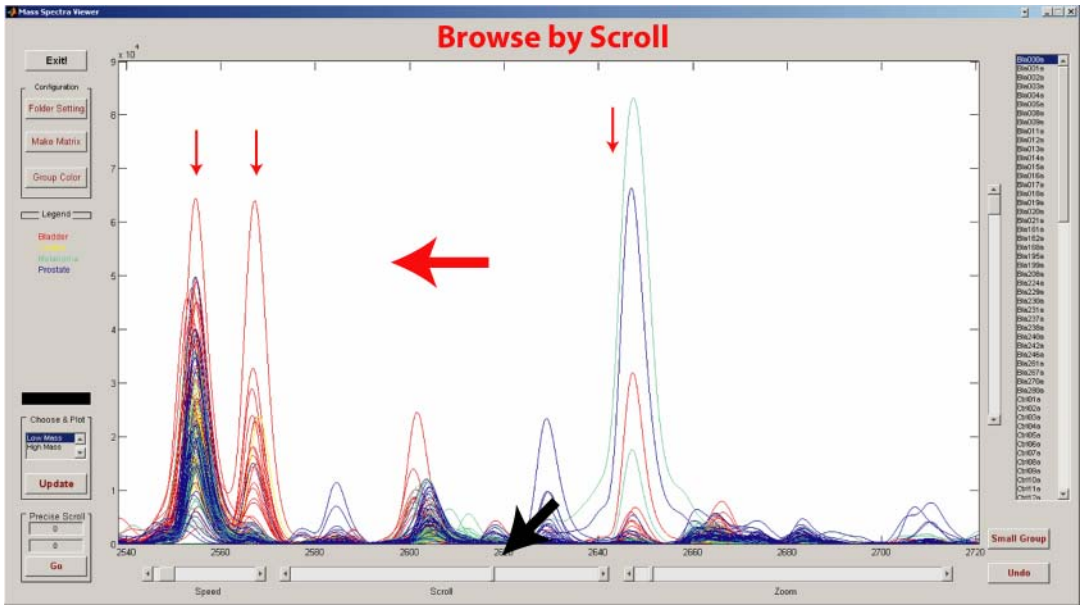
See A to H for major functions.



a. Select **Low** or **High Spectra**, and **Update**. Matrix files will be loaded and plotted.

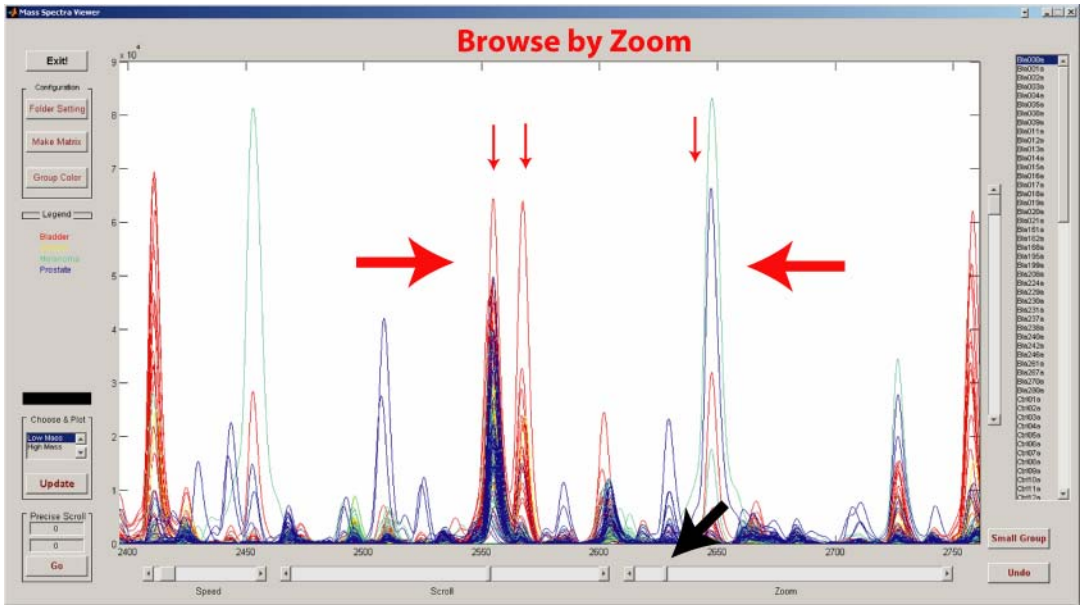


b. Use **Scroll** slider to browse along x-axis (mass value).



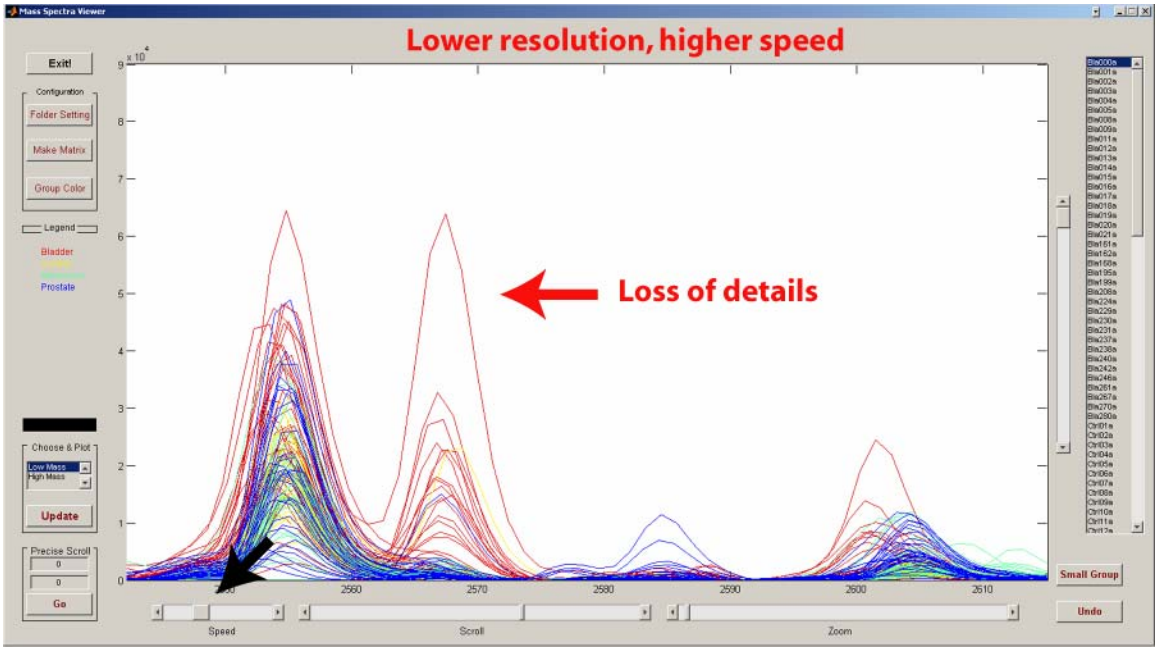
- In Windows version, you may use **Left / Right arrow keys** to scroll along mass value, and use **Upwards / Downwards arrow keys** to zoom in and out.
- In Mac version, you can click on **Scroll / Zoom** slider, and then use **Left / Right arrow keys** to move along mass value or to zoom in/out.

c. Use **Zoom** slider to zoom in/out along x-axis (mass).

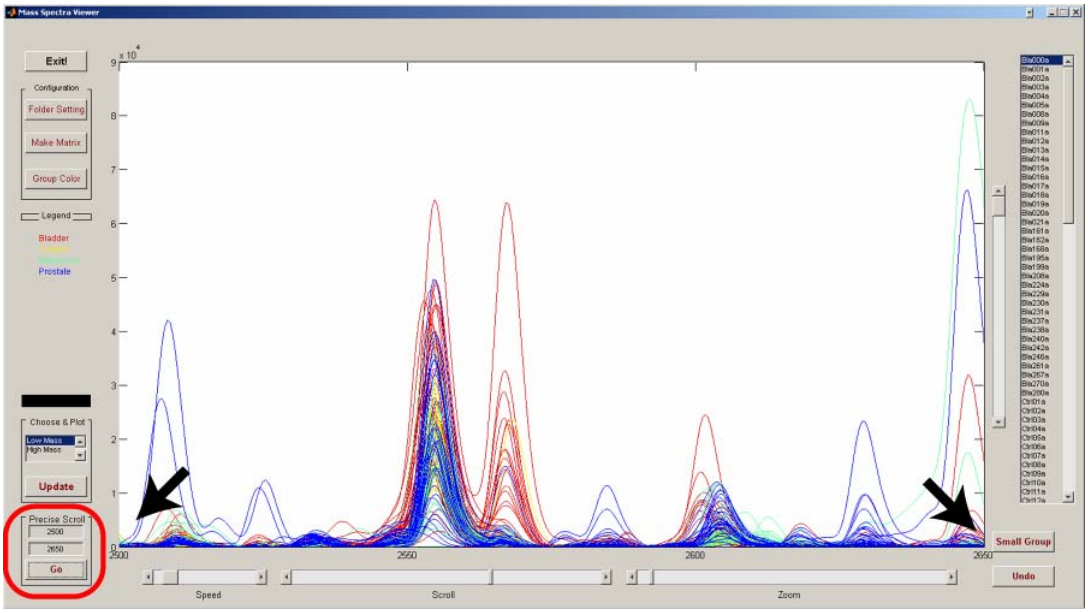


d. Use horizontal slider to zoom in/out along y-axis (intensity). Picture not shown.

e. Use **Speed** slider to change the resolution. Note: Lower resolution, faster speed.



f. Input mass range in **Precise Scroll** to display the spectra of interest.

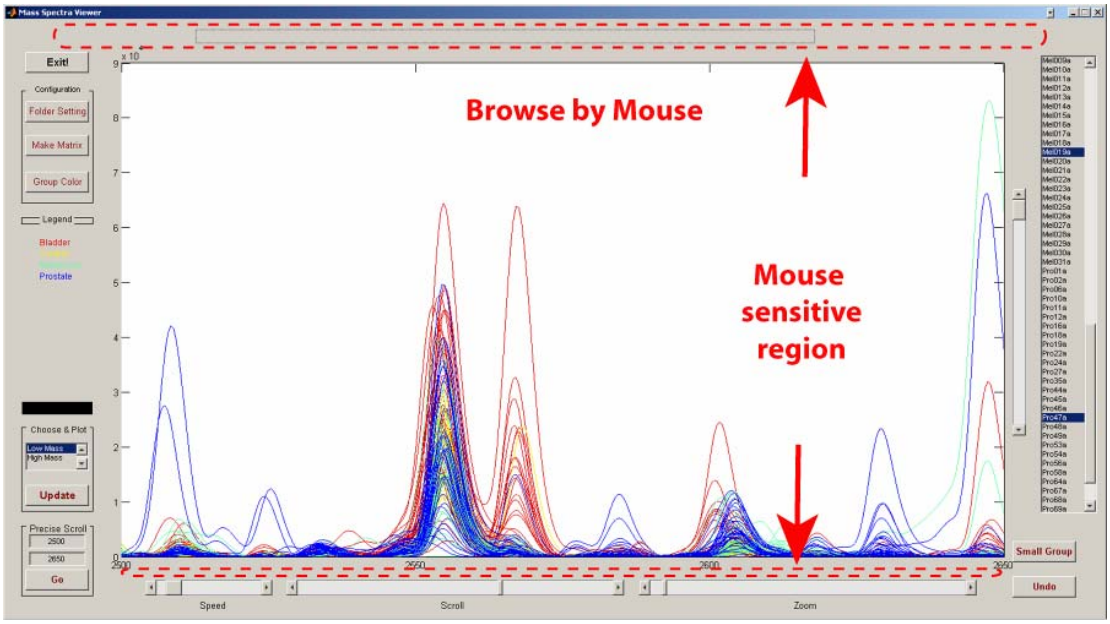


g. **Undo** to restore up to 40 previous operation statuses.

Note: (1) Matlab will escape some middle steps to catch the newest status, if the user acts too quickly. (2) Undo reservoir is run like a first-in-last-out stack. The earliest step will be overwritten if more than 40 actions are recorded.

h. Browse by mouse: you can click-&-drag to zoom in a specific region.

Note: due to functional conflicts, you can only drag a virtual line within top or bottom grey regions. Drag or click on plot region will activate highlighting function (see below)



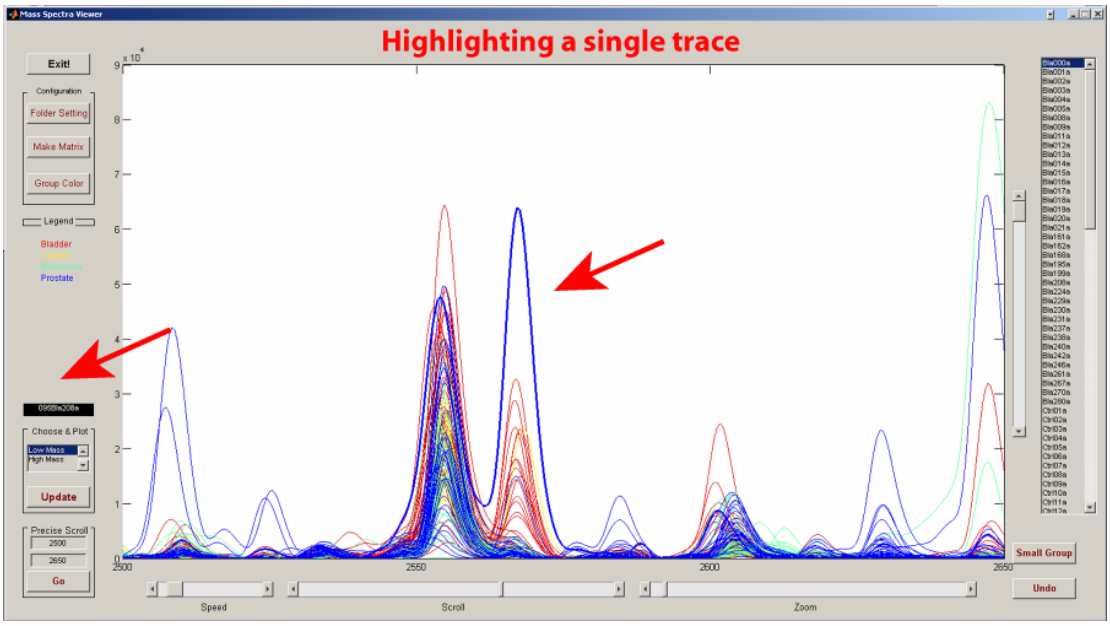
4. **Highlighting and labeling**:

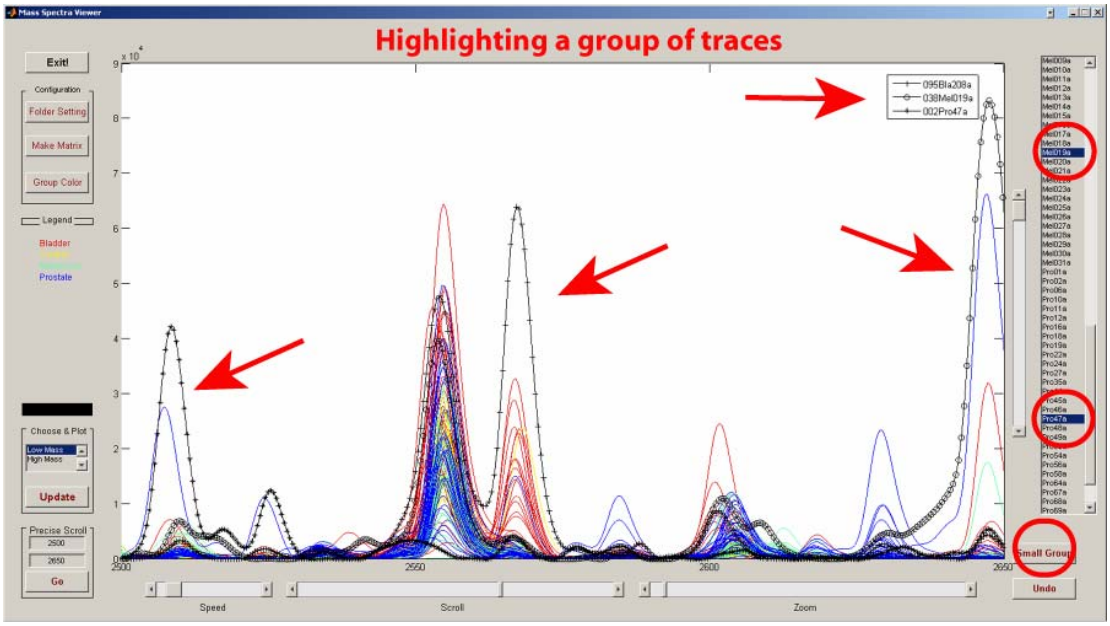
Functions are implemented for two major labeling purposes – (i) by *a trace curve* and (ii) by *trace name(s)*.

a. Labeling by a trace name: click on a trace. It will be highlighted in blue. Trace name is shown on the left panel. Click on a blank space will restore to its original color. The algorithm is to find selected curve in an internal objects list, save and change its current status, finally put it in the front of objects list so that curve is shown above all others.

b. Labeling by trace name(s): use mouse, as well as **Shift** or **Control** key, to select one or more trace(s) in right list box. Then hit **Small Group**. Selected traces will be black-coded and shown with legends. The algorithm is to reload all matrixes from hard drive, match trace name(s) with *Samplename* (see *Matrix files*), and finally plot identified curves over previous ones.

Note: Loading and plotting commands take longer time. You must click on **Update** to clean these highlights.





5. **Exit**

   Click on **Exit** to quit the program. It makes sure that all memory used by internal functions will be released.

Project Report – Visual Interface for Mass Spectra Analysis

Title: Visual Interface for Mass Spectra Analysis

Client Laboratory: Tempst Laboratory

Location: RRL 553

Date begun: June 10, 2004

Date finished: July 27, 2004

Contact person: John Philip

Summary: This project is aimed to develop a prototype of visual interface for mass spectra analysis on Matlab platform. The application, Mass Spectra Viewer (MSV), enables the researcher to load and review a large amount of spectra data in a reasonable speed. Client is adopting this interface in replace of manufacturer's propriety software for daily analysis. A long term goal of this project is to develop an integrated package with signal processing and statistical tools, as well as visual interface, for analyzing mass spectra data obtained from on-going MALDI/TOF studies. In this version of MSV new and enhanced functions include highlighting trace(s), zooming / scrolling by keyboard, restoration of previous statuses, operational optimization, and more precise scroll control.

Features:

- Streamlined visual interface for smooth operation.
- Quick rendering thousands of spectra traces.
- Group samples by colors, legends and line styles.
- Highlighting single / multiple trace(s).
- Browsing traces by scrolling and zooming.
- Browsing traces of a precise range defined by keyboard input.
- Implement of keyboard operation.
- Cross-platform compatibility at source code level, tested in both Mac and Windows environment.

Tested software / hardware environment

- Matlab V7.0 R14 for Windows

a. Windows XP SP1, HP XW 8000 with Xeon 3.06GHz CPU, 2GB RAM and 1600*1200 monitor.

- Matlab V7.0 R14 for Mac
  a. OS X 10.3, Power Mac G4 with 733MHz CPU, 768MB RAM and 1600*1200 monitor.
  b. OS X 10.3, Power Mac G5 with 2.0GHz CPU, 2GB RAM, 1680*1050 monitor.
  c. OS X 10.3, Power Mac G5 with 2.0GHz CPU, 4GB RAM, 1920*1200 monitor.

Package components

- Main program: ***masspectraviewer.m***
- Supporting programs:
  a. ***GroupLegend.m*** – grouping samples by colors.
  b. ***ChgDirectory.m*** – defining folders of ASCII, matrix and definition files.
  c. ***marklegend3.m*** – highlighting and labeling traces.
  d. ***map.mat*** – containing internal 64-step color map.
- Internal files, in plain text format. They are generated and used by main and supporting programs.
  a. ***cfg.txt*** – containing directory information generated by *ChgDirectory.m*
  b. ***gpidx.txt*** – containing sample types and associated color codes.

Data files (Provided by user)

- Definition file – containing names and classifications of traces. Definition file must be in TXT format and is organized in three space-separated columns – (a) numerical index, (b) trace name (without _1 or _2 addendum and *.ASCII* extension) and (c) classification. For examples:

1	001Pro70a	Prostate
2	002Brn47a	Brain
3	003B13a	Breast
4	003Mel021a	Melanoma
…	…	…

- ASCII data files – containing mass spectra data in TXT format. ASCII data file is organized as two space-separated columns (Column α and β), which refer to mass value and spectrum intensity on this position, respectively.

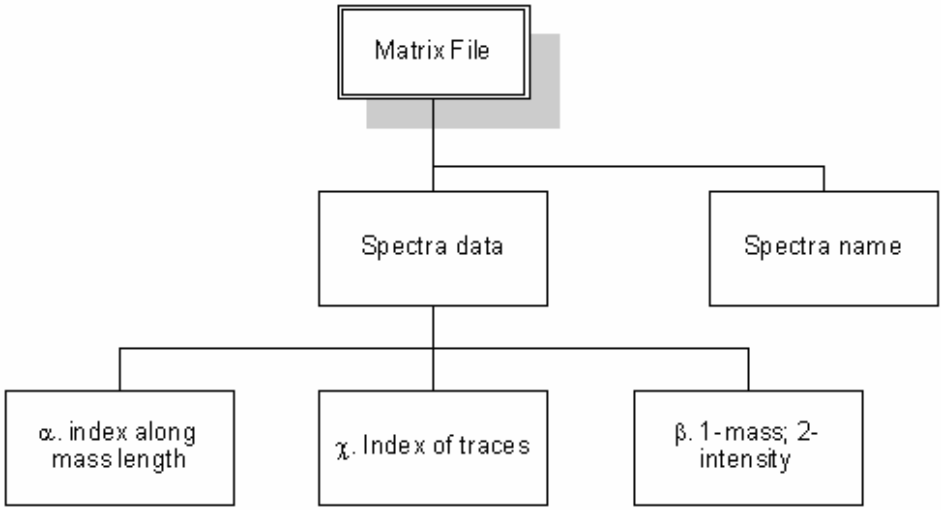
- MAT data files – intermediate files generated from ASCII data files for internal usage. See below for details.

Algorithm consideration & optimization

- Matlab was chosen as software platform because its high-level programming language is capable of processing large data set. It also has many toolboxes for signal and imaging processing. However, GUI programming in Matlab environment is not strong and flexible enough for large scale development. Furthermore, graphic render engine in Matlab is too slow for fast plot of thousands of traces.

- To achieve the goals of this project, following steps had been proven very effective:
  a. Converting spectra data from ASCII-format to Matlab internal matrix. It accelerates the loading process. See below for details of matrix.
  b. Using OPENGL option as the render engine. It accelerates the plotting speed.
  c. Reducing time of re-plotting the data. The whole data set is plotted only once. Browser functions such as scroll and zoom are achieved by rescaling the axis.
  d. Reducing data points by scale down the resolution. This approach is especially useful for a data set with 500 or higher traces.
  e. Upgrading to faster computer. New computer usually comes with faster CPU, system bus and graphic board.

- Matrix file is constructed as a Matlab *cell* container. A simple diagram of its organization is illustrated as following:



Matrix files can be access by using **load** command, e.g., temp=load('Prostatematrix.mat'). Two variables will be shown in Matlab workspace, temp.matrix and temp.samplename. *Temp.matrix* is 3-way matrix [α, χ, β], where α, β are two columns of mass and intensity (see *ASCII data files*), and χ is index of traces. *Temp.samplename* is a one way matrix containing name of each sample indexed by χ.

Note: sometimes a sample is lost in the original collection even though its name shows in the definition file. In such a case, a blank matrix will be recorded.

Limitations

- Due to the restriction of memory management, it is impossible to review a large data set (>300 samples) using Windows version. A warning of 'Out of Memory' will be reported by Matlab.
- For more efficient use of memory, you might want to divide samples (e.g., 3800 samples of brain cancer) into several small groups, and then assign the same color to these groups. Sometime in this way you may avoid '*Out of Memory*' problem caused by initializing a huge size of matrix.
- ASCII-format spectra file must use '.ascii' as the extension. Future version will allow more options.
- In definition file, the keyword describing one cancer type should **NOT** be a subset of another cancer type. For example, when you try defining Breast cancer as '*Br*', it could be also considered as *Brain* cancer. Use *Breast* instead.