Code for Ocean Drilling Data (CODD)

Ocean drilling cruises and post-cruise analyses have been generating large volumes of data almost continuously since the DV GLOMAR Challenger first sailed on Leg 1 of the Deep Sea Drilling Project in 1968 with 16 scientists aboard. Over the decades onboard instrumentation has increased in sophistication, shipboard parties have grown in size, and two dedicated drilling vessels plus mission-specific platforms now operate routinely throughout the world oceans. Each ocean drilling cruise is an exemplary multidisciplinary research program in itself and results provide the seeds for further investigations into a myriad of earth science questions.

The proliferation and diversity of the data collected both during and after drilling cruises can at times be somewhat overwhelming for the individual scientist. Data are now freely available through online data bases maintained by the ocean drilling infrastructure for cruise results (e.g. LIMS), by other national efforts (e.g. NGDC) or community efforts (e.g. PANGAEA). However, a unified and consistent system for integrating disparate data streams such as micropaleontology, physical properties, core images, geochemistry, and borehole logging has not been widely available. In this document we describe a system that we have developed over the years. It takes advantage of the versatile graphical user interface and analytical functions contained in the IGOR™ graphing and analysis program commercially available from Wavemetrics, Inc.

Adapting software originally developed for personnel use for the wider community is not a simple task, so we ask at the outset for your patience. If you have problems, questions, or encounter bugs (unfortunately not unexpected) please contact either Roy Wilkens or Anna Joy Drury.

Roy H Wilkens
Hawaii Institute of Geophysics &
Planetology
University of Hawaii
rwilkens@hawaii.edu

Anna Joy Drury
MARUM
University of Bremen
ajdrury@marum.de

Table of Contents

Introduction and Overview

IGOR is a powerful program, but it is not simple and users need to spend a little time understanding how it works. As this is not intended to be an IGOR manual, we assume some general familiarity with the program in the following set of instructions. Those who are totally new to IGOR should at least read the **Introduction** and **GUIDED TOUR 1** in the pdf manual that can be accessed through the HELP menu of an open IGOR experiment and/or view the instructional videos on the [Wavemetrics](#) website. Investment of a couple of hours at the outset will make life much simpler as you use our functions.

Data and Image Naming Protocols

Success using these data handling macros can be greatly enhanced by adhering to a fixed set of naming protocols. Failure to do so will regularly lead to error messages, usually having something to do with "wave does not exist". 99% of problems most people have with the basic functions are related to naming irregularities. **Note** that we use the "U" prefix even for older drilling sites as IGOR doesn't like data waves names beginning with numerals. **Note** also that at present we use "CSF" and "CCSF" for original (aka mbsf) and spliced (aka mcd) depths regardless of the usage in effect when the data were collected.

Images are defined simply by ***Hole_CoreNumber***.

Core numbers less than 10 need a lead 0 (zero). Examples: U1334A_02 or U851B_12. Reduced-size images that are used for most plotting purposes have a prefix ("S2_") added e.g. S2_U1334A_02 or S2_U851B_12. The prefix is added when the function to create small images is run or when the core image is initially constructed from section images.

Image Folders are named using ***Hole_S1/S2***, with S1 used for full resolution images and S2 used for reduced size images. Examples U1334A_S1, U851B_S2. Each hole has a separate folder.

Data are defined by ***Hole_Technique_DataInformation***.

Examples: U1338A_RSC_AStar, U1338A_GRA_GRA, U1220A_MS_Core. Users may define Technique and Data any way they wish, but the Technique name cannot have an underscore because many of our functions use the first 2 underscores to define the ***Hole*** and ***Technique*** for further processing.

Data Folders are defined by ***Site_Technique***

Examples U1333_MS, U848_XRF, U807_GRA Data waves must be stored in folders that are identified by the technique. Unlike the images, data of the same Technique from all holes are kept in the same folder.

Although IGOR is generally not case sensitive, some of the macros dealing with graph labels and such seem to be so. Best practice would be to always use uppercase letters defining holes in image or data wave names. If you have existing data with lowercase hole letters you need to use the *Rename* function under the *Data* menu to change them. You cannot change the case of a letter directly in the Data Browser.

The IGOR Unpacked Data Structure

  IGOR has 2 ways to save an experiment - in either Packed or Unpacked structures. The simplest is the Packed structure, which holds all data and history within a single file. For uncomplicated uses this is probably the preferred mode. However, in the case of Ocean Drilling data, where an individual experiment may share data with a number of other experiments, the Unpacked structure seems to be the best form, although it can sometimes be confusing when transferring experiments from one computer to another or when files have been moved around on a single computer. It is possible to share data from external folders within a Packed experiment, meaning that a User might use the packed format, but still share large data folders such as the core images.

Unpacked experiments consist of one small file of instructions plus a folder, generally with the same name, that contains individual binary ".ibw" files for data and images in the experiment as well as several text files such as History, Notebooks, and Procedures. The small file with the experiment name has the identifier ".uxp". Note below the GRA, MS, and other Technique subfolders. They hold the individual data waves.

Name	Size	Date Modified
▼ 🗾 U1338_Data Folder	9.7 MB	Jun 6, 2013 12:26 AM
📄 history	33 KB	Jun 6, 2013 12:26 AM
⬛ miscellaneous	53 KB	Jun 6, 2013 12:26 AM
📄 Notebook0.txt	2 KB	Jun 2, 2013 11:34 PM
📄 Procedure	5 KB	Jun 6, 2013 12:26 AM
▶ 📁 U1338_GRA	2.8 MB	Jun 6, 2013 12:26 AM
▶ 📁 U1338_MS	2.8 MB	Jun 6, 2013 12:15 AM
▶ 📁 U1338_NGR	750 KB	Jun 6, 2013 12:15 AM
▶ 📁 U1338_RSC	3.2 MB	Jun 6, 2013 12:15 AM
▶ 📁 U1338_Splice_Data	6 KB	Jun 6, 2013 12:17 AM
⬛ variables	2 KB	Jun 6, 2013 12:26 AM
🌱 U1338_Data.uxp	10 KB	Jun 6, 2013 12:26 AM

**U1338_Data.uxp and U1338_Data Folder. Binary data files
are inside the subFolders.**

▼ 📁 U1338_RSC	3.2 MB	Jun 6, 2013 12:15 AM
U1338A_RSC_AStar.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_BStar.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Core.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_CSF.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Hole.ibw	62 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Interval.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Section.ibw	49 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Site.ibw	110 KB	Jun 4, 2013 11:53 PM
U1338A_RSC_Type.ibw	62 KB	Jun 4, 2013 11:53 PM
U1338B_RSC_AStar.ibw	60 KB	Jun 4, 2013 11:53 PM
U1338B_RSC_BStar.ibw	60 KB	Jun 4, 2013 11:53 PM

Partial Contents of U1338_RSC

Sharing Data Between Experiments

One of the nice concepts of the IGOR environment is the ability to use the same data in numerous experiments without the need to have multiple copies and very large (in Mbytes) experiments. This is accomplished by **sharing** IGOR binary waves. Sharing means that data modified in one experiment will be modified in other experiments sharing those data. In practice we usually initiate separate experiments for loading images and data, and for splicing. New experiments then share data from them.

Individual binary files can be shared in an experiment using the Data/Loadwaves command, but there are functions that simplify the process accessed under External_InOut in either the *CODD_Data* or *CODD_Image* menus. As an example, consider an open experiment that needs to include data and images that have already been created in other experiments.

Run Load_Shared_Data or Load_Shared_Folder. You can either create new copies of the data in the open experiment or share data from other experiments. Usually sharing is the way to go, but copying at times can be a good idea (see below).

The Load_Shared_Data function will allow the user to choose a data folder from another experiment and share its contents in a designated folder (or root:) of the current experiment.

Load_Shared_Folder duplicates the chosen folder in the open experiment and shares the data as well.

In order to facilitate easy sharing of data and images it is best to keep data and images in folders defined as above. Only coring matrices and offset tables are kept in the root folder. This protocol keeps the experiment "clean" and allows for quick loading and sharing of files from different experiments.

There are times when the user might want to "play" with data and not necessarily have results shared with all experiments. In that case it would be appropriate to copy data rather than sharing it. If, after working with the data, the user decides that the results should replace the original waves in a shared data folder, the user can run
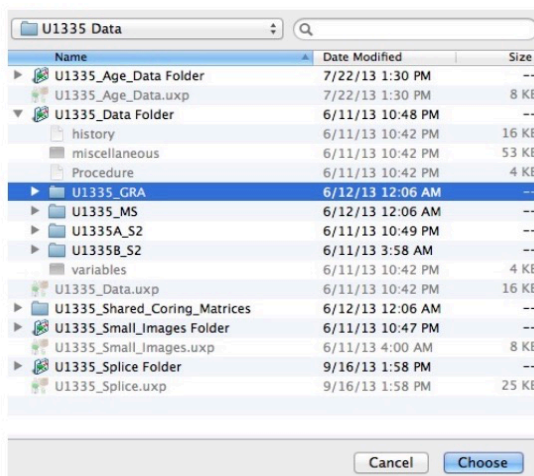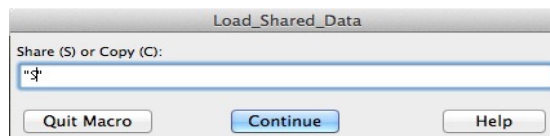
Choosing to share U1335 GRA data.
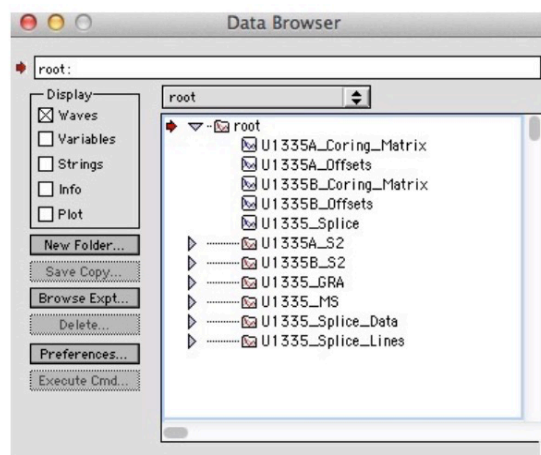
U1335_Splice experiment. S2 images, GRA, and MS data are shared from the U1335_Data and U1335_Small_Images experiments.

Save_Waves_To_Ext_Fldr. This macro allows the user to save (and share) waves to a folder outside of the current experiment. If the user selects the folder that contains the original waves, the originals will be replaced by the new versions and shared with all linked experiments.

Getting Help

     Help files are available by choosing "Help" when a dialog is opened by a function or by selecting "Help Topics" from the IGOR Help menu and finding the listed function. The help dialogs are also listed alphabetically in the file **CODD Help.pdf** accompanying this user guide.
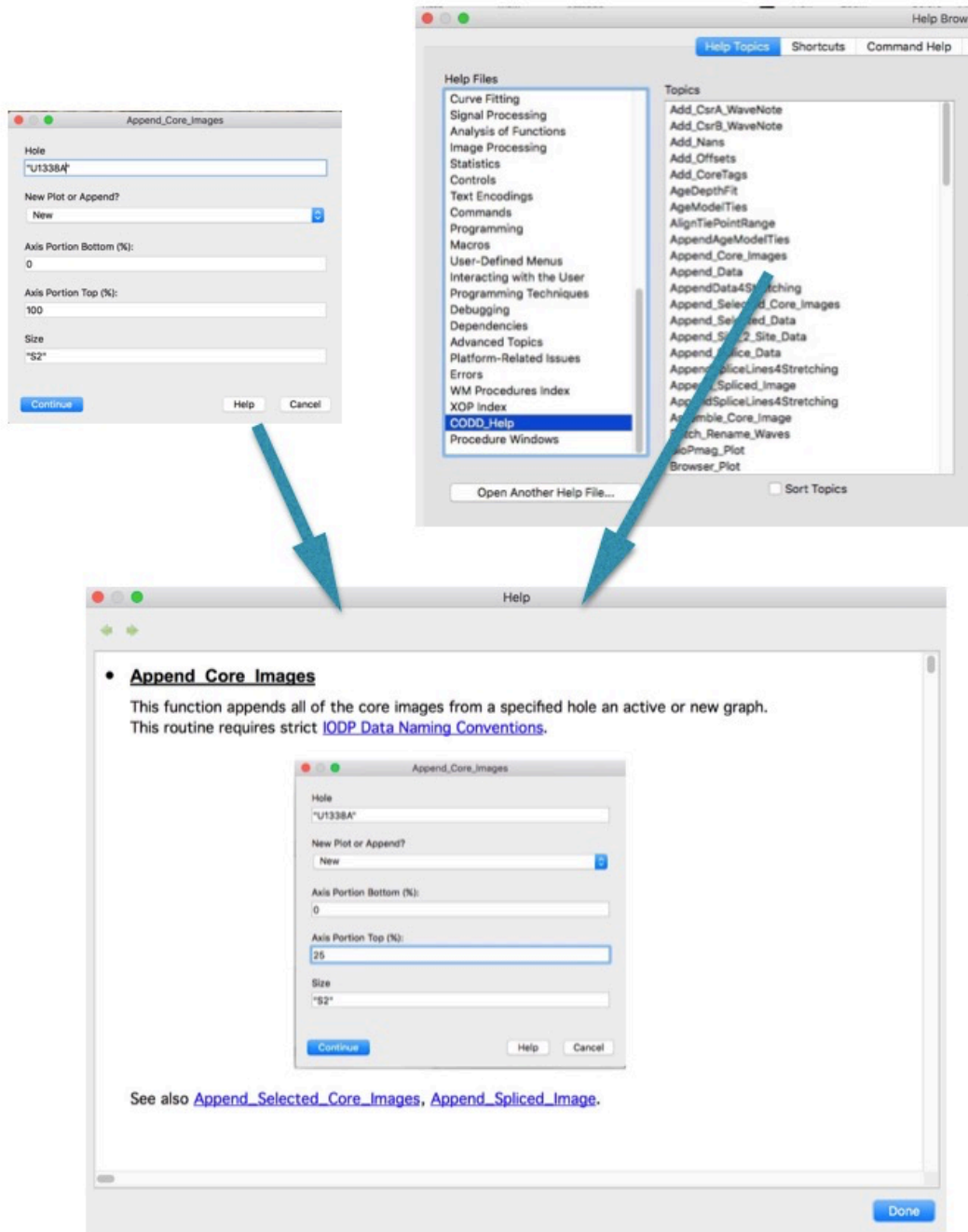
Initiating an Experiment

Loading the Functions

Functions are contained in a number of files that are usually inside a single folder. The function files have the identifier ".ipf" but are actually text files. The folder containing the functions used here are in the folder "CODD Functions" and included subfolders. Individual functions can be opened from inside an experiment by either using the "File/Open File/ Procedure" dropdown menu or by double-clicking the ".ipf" file while the experiment is open. *Includes* files have been set up to load multiple functions automatically (see below).

In order to let IGOR know where functions are stored, follow these steps:
1- Store the "CODD Functions" folder in a convenient place on your computer. Do not use the application folder as it will be overwritten during application updates.
2- Make an alias (shortcut) of the folder.
3- Place the alias in the folder "User Procedures" within the Wavemetrics folder that was created in the User's "Documents" folder when IGOR was initially loaded.

When IGOR is opened it registers the locations of all of the functions within the "CODD Functions" folder. (Note: Users of IGOR 6.3 and older may experience problems. If so, place an alias for "CODD Functions" in the User Procedure folder within the IGOR application folder instead of the folder in Documents.)

There are also "HELP" files (.ihf) that offer help for many of these functions. They are in a file in "CODD Functions" named "CODD_Help.ihf". An alias of this folder needs to be put into the "IGOR Help Files" folder within the Documents/Wavemetrics folder.

Includes Files

Includes files are files that contain calls to a number of functions, so that by opening a single *Includes* file a user may open multiple function files. At the start of any experiment the user must open

<center>Includes_CODD_Menus</center>

Three new CODD menus will appear in IGOR that contain many of the functions discussed here.
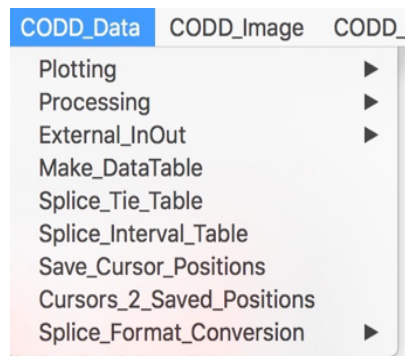
Other *Includes* files are opened for specific tasks such as creating a coring matrix or loading images. Once those specific tasks are finished the *Includes* file should be killed (closed) to avoid overlapping shortcuts in different menus.

Creating a Coring Matrix

       Virtually all ocean drilling data, with the exception of some downhole measurements, are keyed to depths defined by Core-Section-Interval. The functions we developed for ocean drilling data depend on Coring Matrices to store core and section depth data. These are 3 dimensional waves that hold CSF, CCSF, and length data for each core section. They are essential for depth calibration of core images and for offsetting and splicing of both images and data.

To form a coring matrix the user must first download core section data. Open the appropriate database and locate the "Core Section Summary". Request the data for a single hole and then copy the table that results from the request. Open a blank table in IGOR and past the data into it. The table should look something like this, depending on which database you are using:

Point	ExpW	Site	Hole	Core	Type	Sect	Recovered_len	Curated_length	Top_depth_CSF	Bottom_depth	Top
0	321	U1338	A	1	H	1	1.5	1.5	0	1.5	
1	321	U1338	A	1	H	2	1.05	1.05	1.5	2.55	
2	321	U1338	A	1	H		0.18	0.18	2.55	2.73	
3	321	U1338	A	2	H	1	1.5	1.5	2.7	4.2	
4	321	U1338	A	2	H	2	1.5	1.5	4.2	5.7	
5	321	U1338	A	2	H	3	1.5	1.5	5.7	7.2	
6	321	U1338	A	2	H	4	1.5	1.5	7.2	8.7	
7	321	U1338	A	2	H	5	1.5	1.5	8.7	10.2	
8	321	U1338	A	2	H	6	1.5	1.5	10.2	11.7	
9	321	U1338	A	2	H	7	0.75	0.75	11.7	12.45	
10	321	U1338	A	2	H		0.2	0.2	12.45	12.65	
11	321	U1338	A	3	H	1	1.5	1.5	12.2	13.7	
12	321	U1338	A	3	H	2	1.5	1.5	13.7	15.2	
13	321	U1338	A	3	H	3	1.5	1.5	15.2	16.7	
14	321	U1338	A	3	H	4	1.5	1.5	16.7	18.2	
15	321	U1338	A	3	H	5	1.5	1.5	18.2	19.7	
16	321	U1338	A	3	H	6	1.5	1.5	19.7	21.2	
17	321	U1338	A	3	H	7	0.78	0.78	21.2	21.98	

(Table0:ExpW,Site,Hole,Core,Type,Sect,... — cell R7 = A)

All the information that is needed are waves identifying Core, Section, Length, and Top CSF (Depth).

Load Includes_Coring_Matrix and execute Create_Coring_Matrix from the "CODD_Data" menu. Pulldown menus will ask you to identify the data waves necessary to form a Coring_Matrix. A new data wave will be formed with the name of U1338A_Coring_Matrix. This wave has 3 layers; the uppermost layer contains the CSF (mbsf) depths to the top of each section of each core;

Create_Coring_Matrix dialog:

Hole: "U1338A"
CSF or mbsf Wave: Top_depth_CSF_A__m_
Core Number Wave: Core
Section Number Wave: Sect
Section Length Wave: Recovered_length__m_
Replace CC as Section 9?: Y

[Continue]   [Help]   [Cancel]

Table0:U1338A_Coring_Matrix

Row	U1338A_Corin 0	U1338A_Corin 1	U1338A_Corin 2	U1338A_Corin 3	U1338A_Corin 4	U1338A_Corin 5	U1338A_Corin 6	U1338A_Corin 7	U1338A_Corin 8	U1338A_Corin 9
0										
1		0	1.5							2.55
2		2.7	4.2	5.7	7.2	8.7	10.2	11.7		12.45
3		12.2	13.7	15.2	16.7	18.2	19.7	21.2		21.98
4		21.7	23.2	24.7	26.2	27.7	29.2	30.7		31.45
5		31.2	32.7	34.2	35.7	37.2	38.7	40.2		41
6		40.7	42.2	43.7	45.2	46.7	48.2	49.7		50.52
7		50.2	51.7	53.2	54.7	56.2	57.7	59.2		59.96
8		59.7	61.2	62.7	64.2	65.7	67.2	68.7		69.43
9		69.2	70.7	72.2	73.7	75.2	76.7	78.2		78.78
10		78.7	80.2	81.7	83.2	84.7	86.2	87.7		88.55
11		88.2	89.7	91.2	92.7	94.2	95.57			96.96
12		97.7	99.2	100.7	102.2	103.7	105.2	106.7		107.47
13		107.2	108.7	110.2	111.7	113.2	114.7	116.2		116.9
14		116.7	118.2	119.7	121.2	122.7	124.2	125.7		126.57
15		126.2	127.7	129.2	130.7	132.2	133.7	134.8		135.53
16		135.7	137.2	138.7	140.2	141.7	143.2	144.7		145.45
17		145.2	146.7	148.2	149.7	151.2	152.7	154.2		155.01

the middle layer contains the lengths in meters of each section; the 3rd layer is used later and will contain the CCSF (mcd) Hole-2-Hole correlation depths to the top of each section. As a default we are designating the Core Catcher as Section # 9. In the rare instance where a hole has a core with > 8 sections, see the Appendix.

To make a coring matrix for another hole simply paste the new values over the ones just loaded and run the function again with the Hole designation changed. Once the Coring Matrices are formed the waves loaded from the coring summary should be deleted as they will no longer be used. Kill the table and delete the waves.

The final step is to export the Coring Matrices into an external folder as IGOR binary files. This can be done easily by running Save_Waves_2_External_Folder. This step is not absolutely necessary as coring matrices may be shared from the folder of the experiment you are using to form the matrices, but we have found it convenient to put the matrices and the core offsets that result from hole-2-hole correlation in a common folder.

NOTE: The 1st hole at each site during the DSDP did not have a letter designator. That means that the progression of holes starts with U572, U572A, U572B, etc. To keep life simple we use the convention of assigning to the 1st hole the letter X so that the progression becomes U572X, U572A, U572B etc.

Depth Calculation Using the Coring Matrix

There are functions provided for calculation of CSF and CCSF under the CODD_Functions menu, but it can also be done manually.

To get CSF from the matrix on the command line use:
    "print *MatrixName* [Core#] [Section#] [0] + Interval/100"
e.g. for U1338A 2H-3, 23 :

```
●print U1338A_Coring_Matrix[2][3][0] + 23/100
  5.93
```

The "0" inside the 3d set of brackets indicates the top (CSF) layer. For CCSF you would use "2" rather than "0" for the bottom (3^d) layer.

Loading and Calibrating Images

      Core images can be used for much more than a simple visual reference when they are properly captured and depth-calibrated within an advanced data analysis software system such as IGOR™. Image loading is the most complicated part of the process because over the years the image data have been captured aboard different drilling ships by different methods and are presently stored in different databases with different naming conventions. Once the images of core sections have been loaded the process is the same for all DSDP, ODP, and IODP data.

      The way we have been processing the images involves 3 steps - the first to load the section images, a second to assemble whole core images, and (optionally) a third to reduce the core image size, particularly from the high resolution line-scans.

      The three different kinds of image data available to date include core table photos and images acquired with two different flatbed line scanners. Each of the image formats requires slightly different processing to come to a common format for further work. Some of the preliminary steps are a little more tiresome than others, but at least they will only need to be performed once.

Core Table Photos

      Core table photos are digital scans of photographic images taken of all of the sections of the single core laid out on a table shortly after the cores were split. They are the lowest resolution source of images and the most cumbersome to process as each of the core sections must be "cut" from the image. Additionally, the lighting in the core table photos is slightly uneven, with the center of the image somewhat brighter than the edges. There is a correction for the uneven lighting that can be employed by the user after the initial section image cuts. It is described in a later section of this guide. The following is a detailed example of the proper steps to process a Core Table Photo image.

Go to: http://iodp.tamu.edu/janusweb/imaging/photo.shtml

Core Photos

Data Request Form

| Submit Request | Clear Form | Help | Report Options |

Leg	85
Site	572
Hole	A
Core	
Depth	to meters
Latitude	to degrees (South -90 to 0; North 0 to 90)
Longitude	to degrees (West -180 to 0; East 0 to 180)

| Submit Request | Clear Form | Help | Report Options |

Fill in the appropriate information and submit the request. You should then see something like this:

Core Photos -- Query Results

Data Results

Click here to add or modify query parameters.

Leg	Site	H	Cor	T	Depth (mbsf)	Format	DPI	Image Link
85	572	A	1	H	0.000	PDF	300	Core_1
85	572	A	2	H	9.500	PDF	300	Core_2
85	572	A	3	H	19.000	PDF	300	Core_3
85	572	A	4	H	28.600	PDF	300	Core_4
85	572	A	5	H	36.300	PDF	300	Core_5
85	572	A	6	H	45.900	PDF	300	Core_6
85	572	A	7	H	55.200	PDF	300	Core_7
85	572	A	8	H	64.100	PDF	300	Core_8
85	572	A	9	H	72.300	PDF	300	Core_9
85	572	A	10	H	81.800	PDF	300	Core_10
85	572	A	11	H	91.400	PDF	300	Core_11
85	572	A	12	H	100.700	PDF	300	Core_12
85	572	A	13	H	109.900	PDF	300	Core_13
85	572	A	14	H	119.100	PDF	300	Core_14
85	572	A	15	H	128.300	PDF	300	Core_15
85	572	A	16	H	137.100	PDF	300	Core_16
85	572	A	17	H	145.300	PDF	300	Core_17

Download the core table images by right clicking on the image link and selecting download. The downloaded file name will contain the core identifier.

The older data are stored in PDF format. The PDF images open in IGOR at a poor resolution, so convert core image PDFs to JPGs. Use a fairly good resolution for the JPGs, like 300 dpi or so. Programs like Acrobat or Photoshop can do this in batch mode (if you can figure out how it works...).

Load Includes_Core_Table_Photos. The Core_Boxes menu will be added to IGOR. Run Load_Core_Photo {shortcut ⌘-1} to load a JPG core table photo. (Note: this shortcut and others noted below are set when opening the *Includes* file.) This image will be subsequently replaced by a new image every time a new core table photo is loaded. Should you inadvertently close the core box image, you can reopen it with the graph macro included under the **Windows** menu.

Core_Boxes	Help
Load_Core_Photo	⌘1
Set_Section_Ends	⌘2
Section_Cut	⌘3
Assemble_Core_Image	⌘4
Rotate_Box_Image	
Batch_Lighting_Correction	
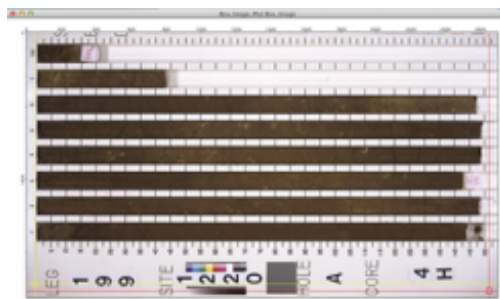
A core table photo displayed after running
Load_Core_Photo

Before proceeding, be sure that the appropriate **Coring_Matrix** is in the "root:" folder. The process of cutting out the section images requires the Coring_Matrix for scaling to CSF/mbsf.

There are 3 steps to assembling a core image from a core box photo:

1. Setting the X position (in pixels) of the bottom of each section (Set_Section_Ends {shortcut ⌘-2}).
2. Setting the position of the top of each section, the section width (in pixels), and cutting out a depth-scaled section image (Section_Cut {shortcut ⌘-3}).
3. Assembling the core image (Assemble_Core_Image {shortcut ⌘-4}).

(1) Use the B cursor to set the position of the lower ends (right hand side) of each of the core sections. This is done by aligning the vertical crosshair of the B cursor with the end of a section and running Set_Section_Ends {shortcut ⌘-2}. The cursor itself does not need to move in the Y direction, only in the X direction. Each section is done in turn until all ends have been identified. In practice, once the cursor has been moved to set the bottom of Section 1, subsequent full sections can be set by using the horizontal arrow keys. The section number will automatically increment each time the macro is run. When the bottom of the CC is defined the user must replace the automatic section number with "9".
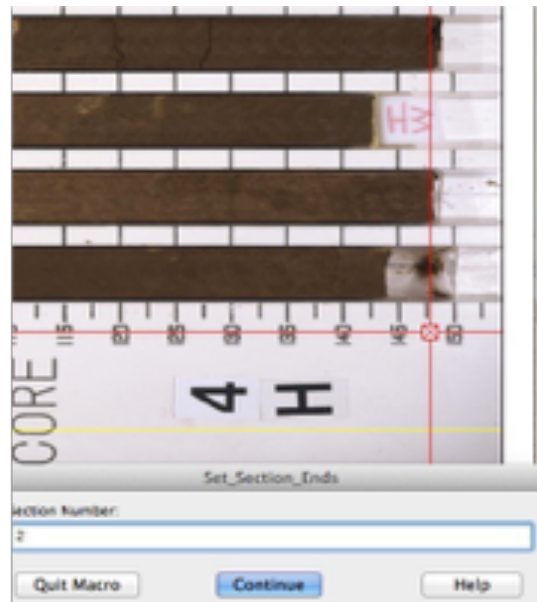
Detail showing B cursor (red) lining up the end of section #2 and the Set_Section_Ends macro input.

Before going on to step (2) you will need to determine the **Width** (in pixels) of one of the sections. Do this by using the A and B cursors and noting the ΔY value in the cursor information lines below the image. This value usually seems to be around 100, but depends on the resolution used. when scanning the photo.

| Y: 1290 | Z: (206,187,195) | ΔX: 80 |
| Y: 1403 | Z: (203,194,189) | ΔY: 113 |
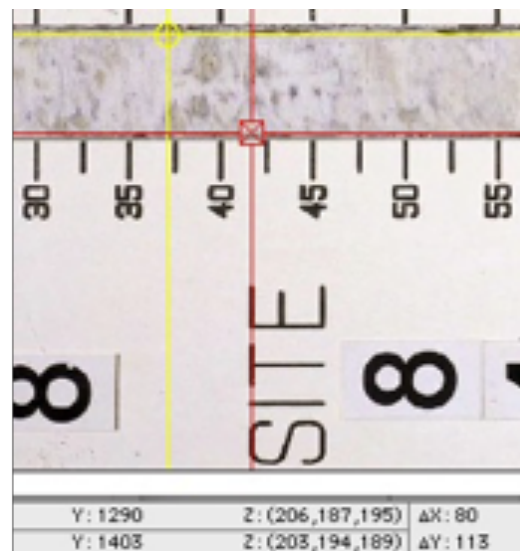
Detail illustrating the use of Cursors A & B to determine the width in pixels of a core section. Note ΔY in lower right corner.

14
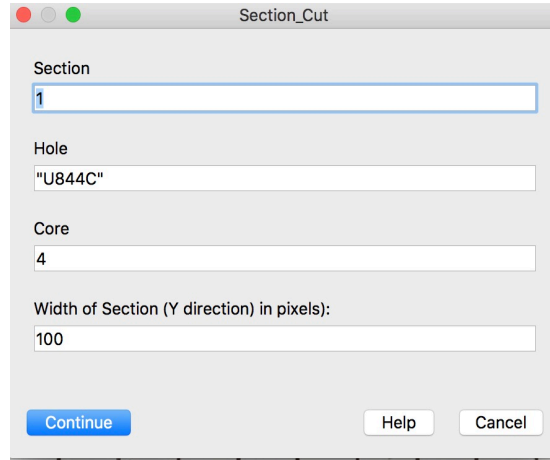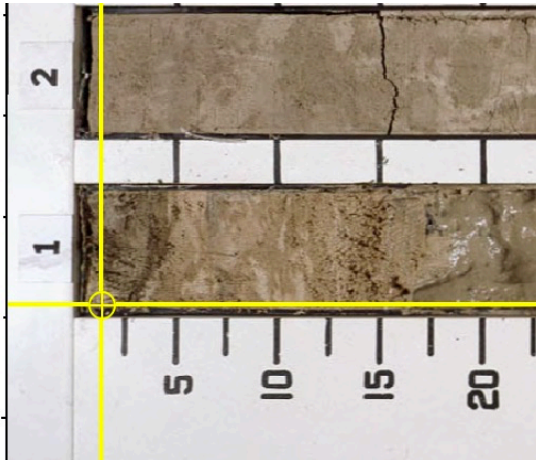
(2) Unlock Cursor A by moving it with the mouse or running CursorBOff (Option-⌘-9). Set the A cursor in the lower left corner of Section 1 and run Section_Cut {shortcut ⌘-3}. The first time through you need to enter the Width value at the bottom of the prompt window. Subsequent runs of Section_Cut will advance the A cursor in the Y direction based on the Width value so that only the arrow keys are needed to define the lower left corner of each section. Finish cutting all sections.

(3) Run Assemble_Core_Image {shortcut ⌘-4}. Be sure to choose **Y** for the "**Save K Files?**" prompt. The result will be a composite core image (displayed) and a subdirectory of the scaled section images. Saving the cut-out section images will mean that you will never again have to repeat the above process.
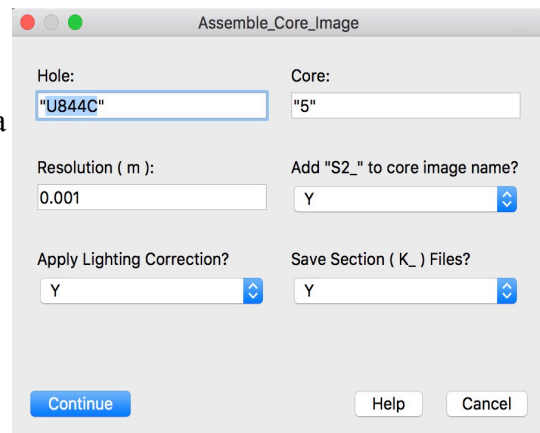
Core Box Photo section images need to be corrected for uneven lighting.This is done while assembling the core image (the original cut out section images are not changed) or later as a batch process using the saved sections. If you have an old, slow computer you may want to use the batch process as it can run while you are doing something else. The lighting correction process is discussed in detail in a later section.

The core image is now depth scaled (roughly 500 µm depth resolution) and can be manipulated as with other data. The core and section images are three-dimensional matrices of RGB color information. The upper (0) layer is R, the middle (1) layer G, and the bottom (2) layer B.

Capture all of the core images for a given Site by repeating the above from Load_Core_Photos onward for each core.

Old Scanner Images
        The original line scanner on the JOIDES Resolution was first employed during Leg 199. At times during its initial use problems with standardized settings resulted in images that were not suitable for entire core or entire site correlation. Fortunately, core table photos were also taken when the older line scanner was employed and can be used in the case of problems with the line scan images. Later legs may or may not have the same problem. At some point the protocol for the scanner was finalized and the images are useful for analysis.

Section images that were scanned with the older line scanner during Leg 199 through Expedition 312 included the meter measuring stick and a core label. Prior to loading these images into IGOR copies of the section images need to be cropped so that only the image of the core material is present, as shown to the right. Once the images have been cropped and put in a folder separate from the originals the process for images from both the old scanner and the newer scanner are the same.

New (Cropped) Scanner Images
        A new scanner was installed before Expedition 320. It was both faster and had a higher resolution than the older scanner described above. Images from the new scanner were also saved in a trimmed format that makes these newer images the easiest to process.

The images for each individual hole to be loaded need to be stored in a separate folder on your computer. Once this is done the next step is to identify the path to that folder for the image loading macro. Go to **Misc/New Path...** to get the following screen:
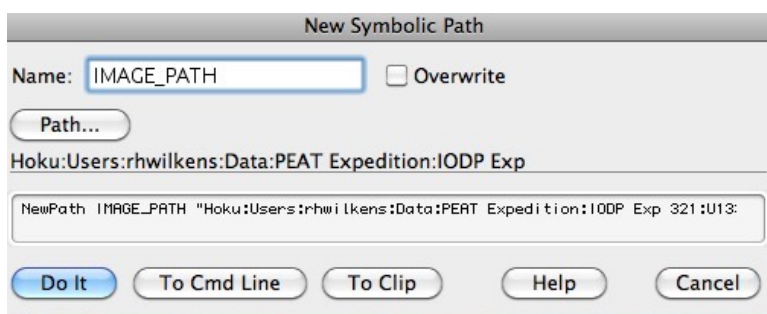
"**IMAGE_PATH** " is the default name used for whichever hole you are processing. When you change to a different hole you will need to kill the old Image_Path and establish a new one for the correct folder. Click the "Overwrite" box in the window above and choose the new path.

Open "Includes_Core_Image_Assembly" and a new "Image_Assembly" menu will appear under the CODD_Image menu. Be sure that the Coring_Matrices are loaded into the root: directory and then run "Universal_Image_Loader".
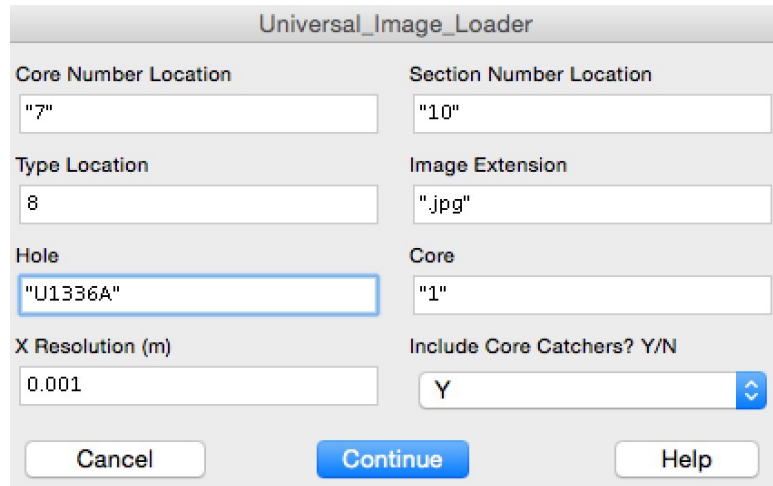
The Core, Section, and Type locations are the positions in the image filename containing the appropriate information. For example, values in the screen above are correct for the file:

U1336A-1H-1-A_SHLF875841_20090605031641_trim.jpg

Remember to count spaces beginning at 0 and not 1. Also remember that the location numbers will change for cores numbered greater than 9. For 2 digit core numbers the location is expressed as 2 numbers e.g. "7,8":

U1338A-12H-1-A_SHLF875841_20090605031641_trim.jpg

Input values are carried forward each time the macro is run and the core number should increment so user input is minimized. Image loading creates section image waves with a "K_" prefix. The "K_.." waves are temporary holders of the individual depth calibrated section images for assembly into a core image. The User can set the desired X (depth) resolution. Native for the newest scanner is something like 30 µm (overkill). The "K_" waves can be saved or discarded during the core image assembly process. Hint - the section images from the line scanners at full resolution, if saved, rapidly overwhelm IGOR with experiments exceeding 1 Gbyte for only a few cores. The only section images that should be saved are the ones cut from core table photos, which are much lower resolution and would be time consuming to recreate.
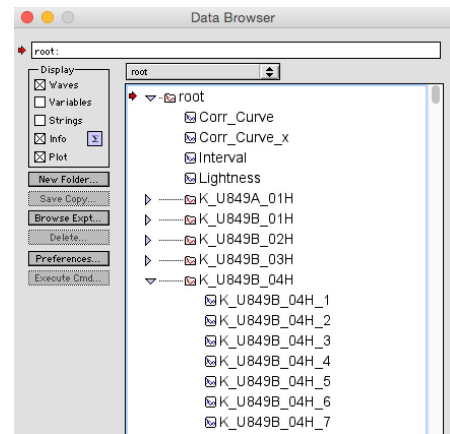
Normally the core image is assembled immediately after loading the images as a check on the loading process (see next step).

Assembling the Core Image

Run Assemble_Core_Image.

Full resolution for the newer line scanner is approximately 50 microns. This is not recommended due to size limitations and the time it takes to draw extremely high resolution images on the computer screen. Choose a reasonable resolution and then say "Yes" to adding "S2" to the image name to skip the step that reduces image size (see next section).

The Lighting Correction is used for the images cut from core table photos and is discussed elsewhere. Lighting Correction is unnecessary for line scan images. Likewise, saving the "K_" files is recommended **only** for the core table photo cuts.
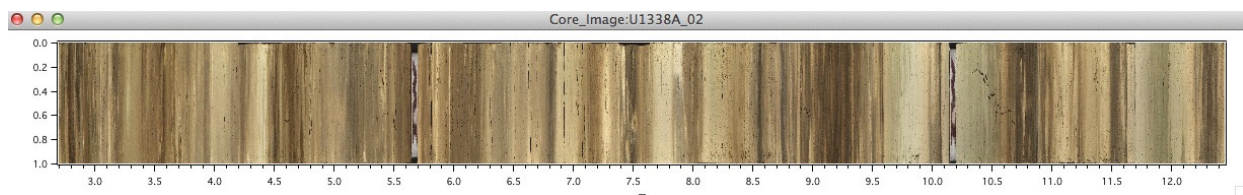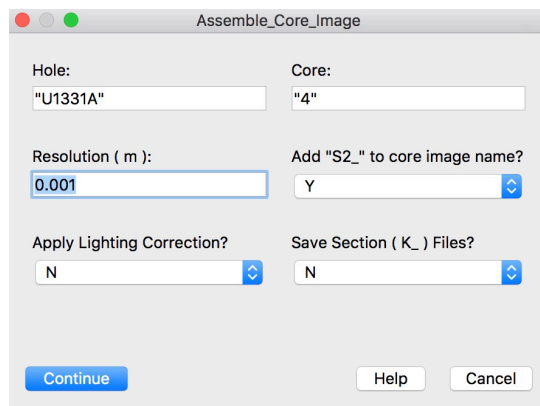
A composite image of the core will be generated and displayed as a check on the process. The Y axis of the image has 101 points scaled arbitrarily from 0 to 1. The depth resolution should be as input. Continue to run the Section Image Load and Core Image Assembly macros until all of the images from a single hole have been processed. The resulting images will be stored in a Folder named "Hole"_S1 for high resolution or "Hole"_S2 for lower resolution.

Line scanner images, because of their potential for high-resolution, can be quite large. The 40 core images of U1338A at full resolution, for example, result in an IGOR™ experiment of a little more than 800 Mbytes. This means that if you want really high resolution images (50 or 100 microns) each hole needs to be handled as a separate experiment. To process another hole therefore, you need to open a new experiment and return to the first step of the core scan image loading process. The next section has instructions on how to create lower resolution images that are more appropriate for most plotting and analysis requirements.

IGOR Hint: Once everything is set up it is possible to batch process multiple scanner-derived cores by copying the macro commands for loading and assembling images from History into a text editor or an IGOR notebook and changing core numbers:

```
•Universal_Image_Loader("7","10",8,".jpg","U1338A","2")
•Assemble_Core_Image("U1338A","2","n","n")
•Universal_Image_Loader("7","10",8,".jpg","U1338A","3")
•Assemble_Core_Image("U1338A","3","n","n")
….
•Universal_Image_Loader("7,8","11",9,".jpg","U1338A","20")
•Assemble_Core_Image("U1338A","20","n","n")
```

Copy all of the lines and paste them into the command line. Hit return and walk away, but remember that you need to change the loader command for core numbers of 10 or greater.

Reducing Image Size

Open a new experiment. Copy core images from a single hole into the new experiment using the Data Browser or share the images from the original image loading experiment. The "Browse Experiment" tab allows you to open other experiments and drag (copy) waves and data folders into the "root" folder of the present experiment. (Note: in IGOR 7™ the user must hold down the "Option/alt" key when selecting "Browse Experiment" to access unpacked experiment folders.)

Load Includes_Small_Images and then run S2_Reduce_Image_Size in the CODD_Image menu.
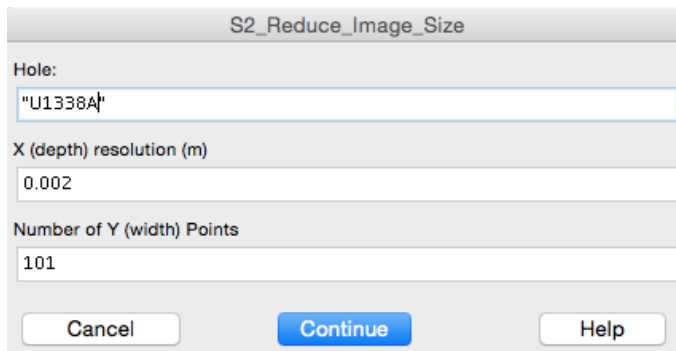
New images will be formed that begin with "S2_" and will be stored in a new data folder titled by Filename and "_S2" (e.g. U1338A_S2). We find that 1 or 2 mm depth resolution and 101 Y points works well for the small images, yielding decent resolution and yet keeping the core images small enough to include multiple holes in a single experiment. Delete the original higher resolution images that you copied/shared into the experiment.

Now repeat the process for the next hole until the new experiment has all of the core images that are reduced resolution. The data browser should look something like this (the Offsets waves are for composite sections and may not be present - see below):
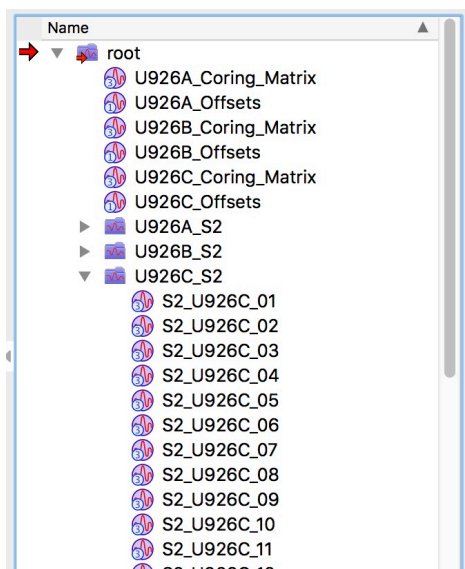
Example of U926 Small Images experiment data browser. Reduced size images are placed in separate folders.

Loading Data

In loading data from pre-Expedition sites a problem you will encounter is that the site designators from older sites are numeric. There is no "U" before the site number. IGOR™ does not like wave names beginning with a numeral, so we have adopted the protocol of using "U" for all sites, no matter how old. Two simple solutions are to correct the problem in older datasets in a spreadsheet before loading the data or load the data, delete the numeric "Site" wave, duplicate another text wave such as the "Hole" wave, rename it to "..._Site", and assign the proper designation by entering the direct command, e.g. U572A_GRA_Site = "U572" on the command line.

Point	U926C_GRA_	U926C_GRA_	U926C_GRA	U926C_GRA	U926C_GRA_	U926C_GRA_	U926C_GRA_CSF	U926C_GRA_CCSF	U926C_GRA_GRA
0	U926	C	1	H	1	78	1.28	3.46	1.47
1	U926	C	1	H	1	79	1.29	3.47	1.44
2	U926	C	1	H	1	81	1.31	3.49	1.42
3	U926	C	1	H	1	81	1.31	3.49	1.40
4	U926	C	1	H	1	84	1.34	3.52	1.42
5	U926	C	1	H	1	84	1.35	3.53	1.43
6	U926	C	1	H	1	87	1.37	3.55	1.43
7	U926	C	1	H	1	88	1.38	3.56	1.46
8	U926	C	1	H	1	90	1.40	3.58	1.48
9	U926	C	1	H	1	91	1.41	3.59	1.49
10	U926	C	1	H	1	93	1.43	3.61	1.51
11	U926	C	1	H	1	94	1.44	3.62	1.52
12	U926	C	1	H	1	96	1.46	3.64	1.50
13	U926	C	1	H	1	99	1.49	3.67	1.52
14	U926	C	1	H	1	102	1.52	3.70	1.54
15	U926	C	1	H	1	104	1.54	3.72	1.51
16	U926	C	1	H	1	105	1.55	3.73	1.52

Data may be loaded by opening a blank table by selecting "Windows/New Table.." from the Windows dropdown menu or by typing "edit" on the command line. Then cut and paste data from a spreadsheet or web data report from one of the online databases. There are also a number of data file formats recognized by IGOR™ and accessed under the dropdown "Data/Load Waves".

A rare problem occurs when the first core of a hole has only a core catcher, which is designated as "CC" in the online Section data. IGOR™ decides when data are loaded whether they are numerical or text waves from the first data point. If the first point is "CC" the wave will be loaded as text - which will cause no end of problems later when trying to use Section in calculations. The way to get around this problem is to open the data file in another program and enter the number "9" as the first point in the Section data. It is no good to make the change after loading into Igor as the wave has already been designated as text.

As noted in the section on naming protocols, data wave names must reflect Hole, Technique, and DataInformation. Use the ReName function under the Data menu to change the names of newly loaded data waves. DataInformation includes Site, Hole, Core, Type (H, X, or R), Section, Interval (in cm - not m), CSF, CCSF (if available), and the measurement(s). In the case of techniques such as XRF there may be many different measurements (DataInformation) while those such as GRA will have only one.

Depths - at the moment we are only using CSF (aka mbsf) for the unshifted depths and CCSF (aka mcd) for shifted values regardless of what they were originally named. Other depths, like rmcd, can be loaded, but they won't work at present with some of the automatic plotting macros (next section).

Displaying Images and Data

     Images and data can, of course, be plotted using simple commands directly in IGOR™. However, the macros described in this and the **Image and Data Splicing (** for splices) section will save users a lot of formatting by automatically fixing axis names. Additional display macros depend on the automatic axes names to perform operations like affixing name tags to cores or adding lines that define splice sections.

Making a Single or Multi-Hole Image Plot

     Run Append_Core_Images to display all of the images from a single hole at one time. Choose whether to make a new plot or append the images to the active graph.

If you are displaying images or data from multiple holes on the same graph it is useful to specify where they will plot on the Y axis using the "Axis Portion" input line. For example, a 3 hole display might plot Hole A from 0-32, Hole B from 33-65, and Hole C from 66-100. You can also change axis plot ranges for multiple axes using the **Axis** tab in the **Modify Axis** window. Leave room for a plot of the spliced image as well if you are planning a summary figure.

If you only desire plotting a subset of images from a hole, first select the images in the Data Browser and then run Append_Selected_Core_Images. A dialog will check to make sure you have the correct waves selected and then ask about the Y axis.
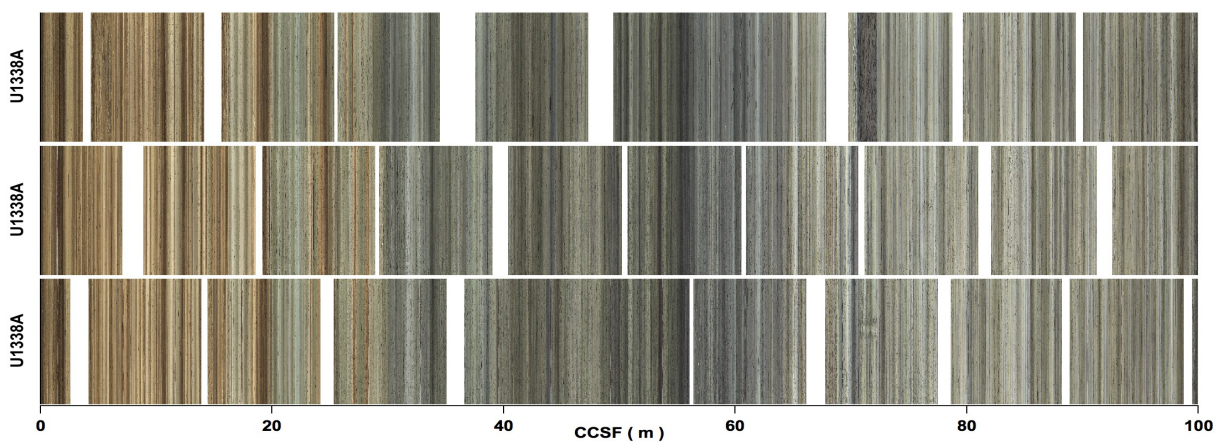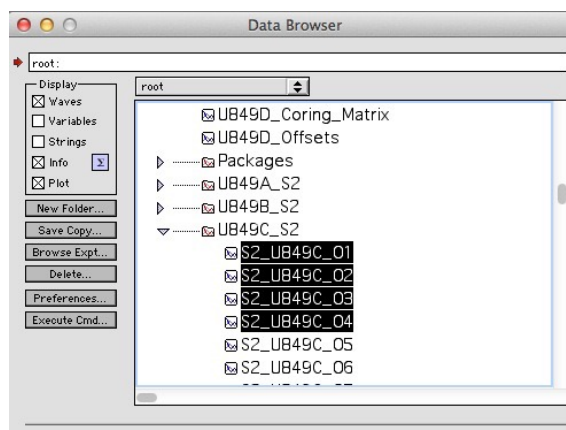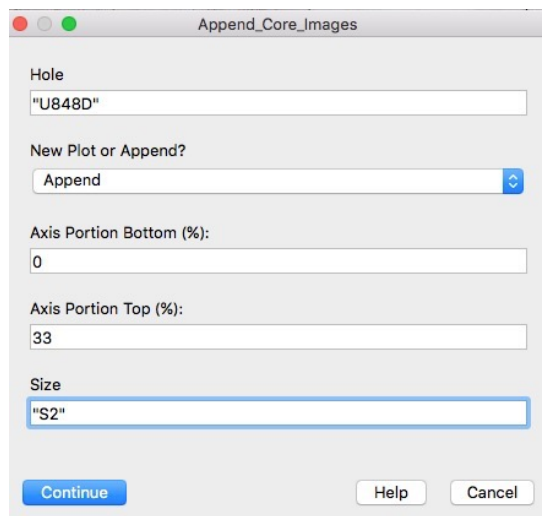
Core Images for Site U1338 plotted against CCSF.

(Note the **Live Update** box on the upper right of IGOR™'s **Modify Axis** window. This is a convenient way to see if the values you have entered will produce the result you desire. However, if this box is checked it will redraw the graph each time you enter a value. With an image-heavy graph this can take time if you are making multiple changes. Best practice is to leave the box unchecked until you have finished entries, then check the box to see your result. Uncheck it again before finishing with the window.)

Making a Single or Multi-Hole Data Plot

The procedure for data plotting is much the same as for image plotting. Run Append Data and choose whether to make a new plot or append to the active graph. Besides the Hole, the user must input the Technique, Data, depth scale, and Axis_Portion. Automatic axes names are used when either L or R are chose for the axis. The names combine the Hole and the Data. The user may also select an existing axis if desired. In that case the axis portion will be the same as that of the existing axis regardless of what is input.
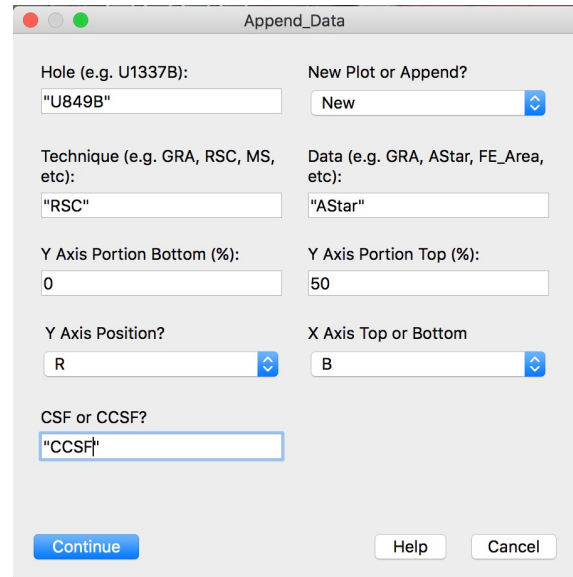
There is also an Append_Selected_Data function that works by selecting **data** waves in the Data Browser in the same manner as the selective image macro described above - select the data wave(s) and the macro will ask which depth scale to use, CSF or CCSF. In the case of multiple selections the macro will ask about the Y axis separately for each wave to be plotted.

A third data plotting function, Browser_Plot, will plot any 2 waves of equal length against each other. Select the waves in the DataBrowser and run the function. The vertical axis will be labelled with full name of the wave selected if an existing axis is not selected while the X axis remains unnamed.
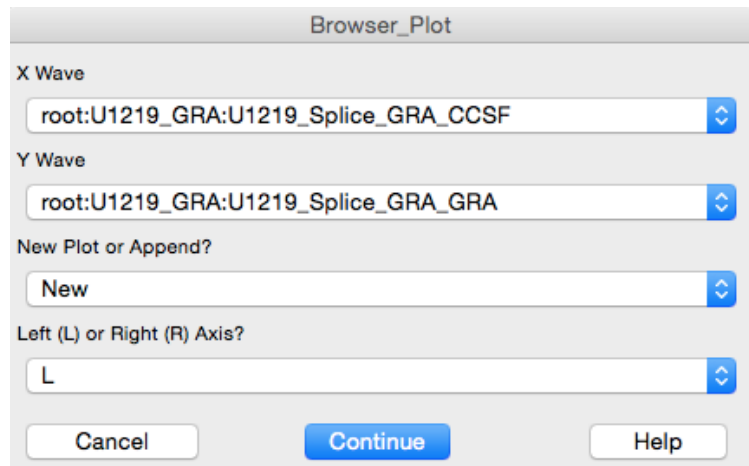
The data initially are auto-scaled which can result in different holes plotted against different Y axis ranges. Use Set_Data_Range to adjust all of the data axes at once.

The data may also be plotted on top of the core images. To do this, simply perform the above steps while the core image plot is the active graph.

Additional data may be added to the data or image+data plots by repeating the procedure. Use the macro Hide_Trace to hide or display individual sets of traces.
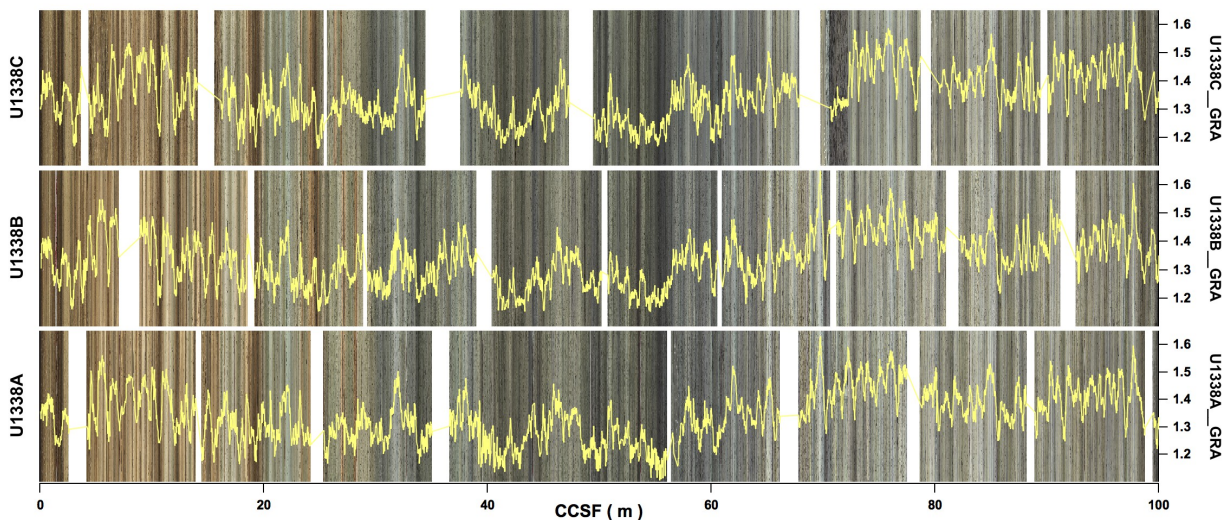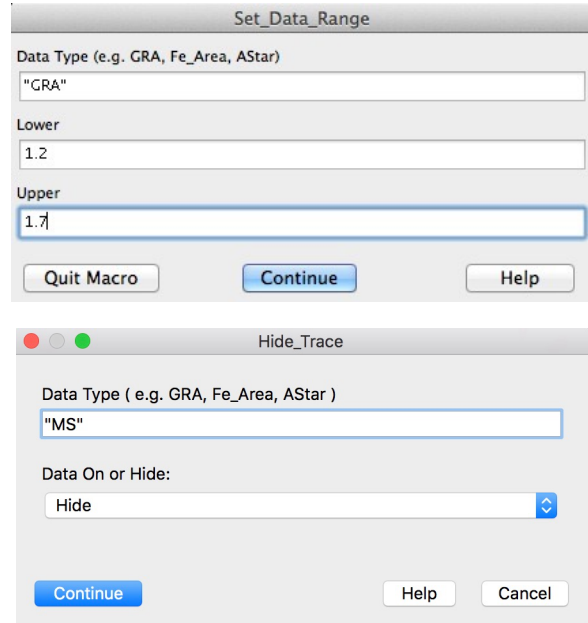
U1338 GRA data plotted on top of core images.

Adding and Removing Image Tags

Tags may be added to individual core images using Add_CoreTags. X and Y positions are relative to the upper left corner of each core image. Increasing X moves the label to the right and increasing negative Y values move the label down. If you want to change the label positions you should first use Remove_CoreTags to erase existing tags. Individual tags can be moved by dragging with the cursor.
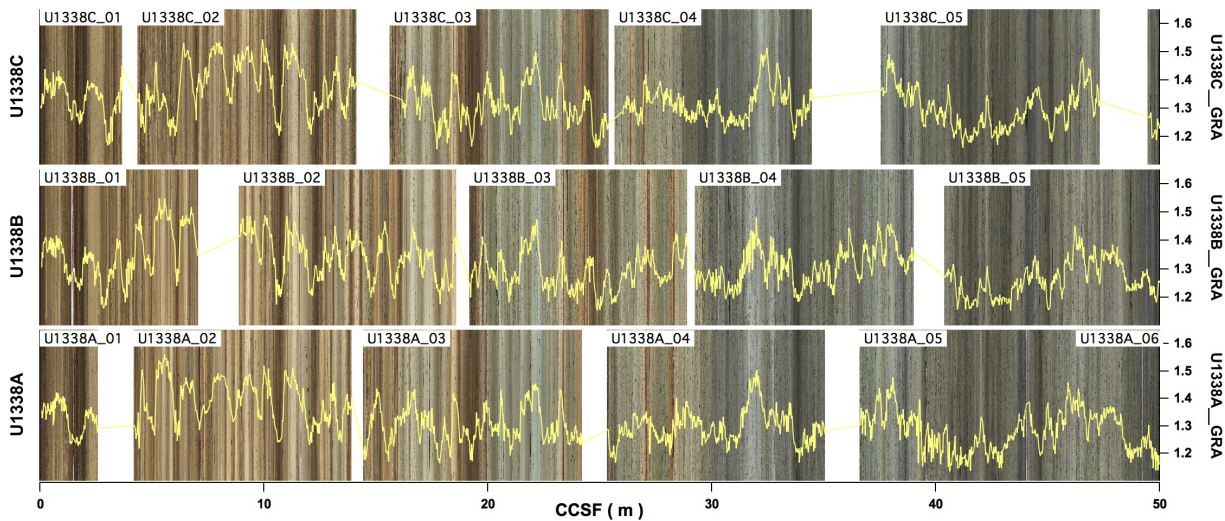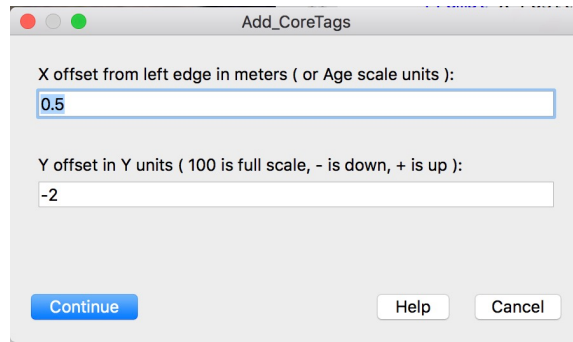
U1338 Images, GRA data, and Core Tags.

Image and Data Splicing

(Note: This section deals with functions to splice data and images once the offsets and splice ties have been established. <u>Creating</u> the splice ties is covered in another document.)

We prefer to splice data and images in a separate experiment. To do this, open a new experiment. Next use Load_Shared_Data or Load_Shared_Folder to share the coring matrices and data or image folders as explained in the Introduction section of this guide.:

In order to create a spliced image or to splice data you'll need 2 additional bits of information. The 1st comes from the Affine (or offsets) table in the 2nd is supplied by one of the Splice Tables.
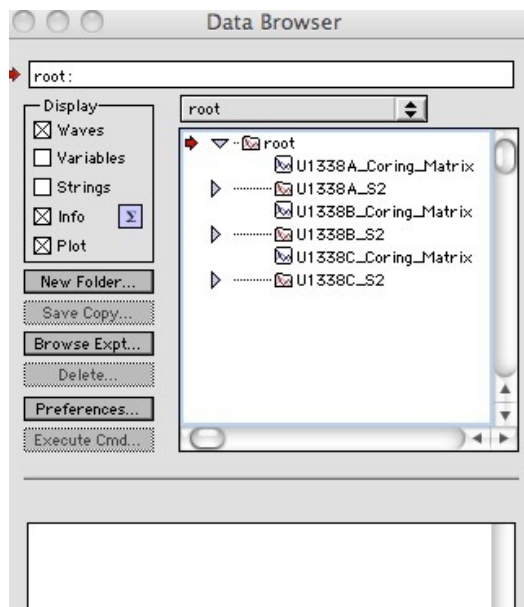
Offset Tables

Form "U1338A_Offsets", "U1338B_Offsets", etc in root: and add offset data for each core of each hole. <u>Be sure to leave row 0 blank and start Core 1 on row 1</u>. Also check to be sure that there were no core numbers skipped if you are cutting and pasting offsets from another source. Some tables omit offsets for cores with no recovery. If splicing hasn't been done, make the Offset waves and fill them with 0's. Use Save_Waves_To_Ext_Fldr to save the offset waves to the same folder holding the Coring Matrices.

Adding Offsets to Images and Data

Run Add_Offsets for each hole.This sets the third layer of the Coring Matrix to the CCSF depth of the sections of the hole.

Run CSF_2_CCSF. This shifts the scaling of the core images to the CCSF depth scale. There is a companion routine CCSF_2_CSF to return the core images to the original CSF depths.

Run Calculate_CCSF_Wave for each hole and each Technique. A new CCSF data wave will be created in the appropriate folder. If a CCSF wave already exists it will be overwritten.

Note on Splice Intervals and Tie Points

Early in 2014 a new protocol for reporting splice data was initiated by IODP. It revolves around the splice intervals rather than the position of the ties. Our macros now use either version of the splice data. The splice data are stored in separate data
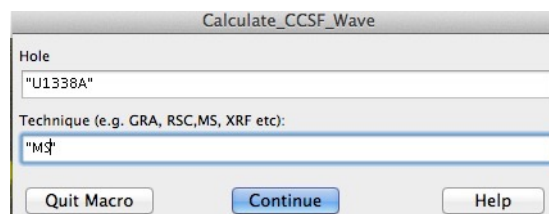
Site_Bottom	Hole_Bottom	Core_Bottom	Type_Bottom	Section_Bottom	Interval_Bottom	CSF_Bottom	CCSF_Bottom	TIE	Site_Top	Hole_Top	Core_Top	Type_Top	Section_Top	Interval_Top	CSF_Top	CCSF_Top
U844	C	1	H	6	13	7.63	7.59	TIE	U844	B	2	H	2	98	6.98	7.59
U844	B	2	H	6	34	12.34	12.95	TIE	U844	C	2	H	3	18	12.28	12.95
U844	C	2	H	6	75	17.35	18.02	TIE	U844	B	3	H	2	83	16.33	18.02
U844	B	3	H	6	105	22.55	24.24	TIE	U844	C	3	H	3	81	22.41	24.24
U844	C	3	H	7	8	27.68	29.50	TIE	U844	B	4	H	2	111	26.11	29.50
U844	B	4	H	7	21	32.71	36.11	TIE	U844	C	4	H	1	121	29.31	36.11
U844	C	4	H	7	1	37.11	43.91	TIE	U844	B	5	H	3	143	37.43	43.91
U844	B	5	H	6	100	41.50	47.98	TIE	U844	C	5	H	2	92	40.02	47.98
U844	C	5	H	6	54	45.64	53.60	TIE	U844	B	6	H	3	113	46.63	53.60
U844	B	6	H	6	125	51.25	58.23	TIE	U844	C	6	H	2	17	48.77	58.23
U844	C	6	H	5	6	53.16	62.62	TIE	U844	B	7	H	1	111	53.11	62.62

Site_Top	Hole_Top	Core_Top	Type_Top	Section_Top	Interval_Top	CSF_Top	CCSF_Top	Section_Bottom	Interval_Bottom	CSF_Bottom	CCSF_Bottom	TIE
U844	C	1	H	1	0	0.00	0.00	6	13	7.63	7.59	ANCHOR
U844	B	2	H	2	98	6.98	7.59	6	34	12.34	12.95	TIE
U844	C	2	H	3	18	12.28	12.95	6	75	17.35	18.02	TIE
U844	B	3	H	2	83	16.33	18.02	6	105	22.55	24.24	TIE
U844	C	3	H	3	81	22.41	24.24	7	8	27.68	29.50	TIE
U844	B	4	H	2	111	26.11	29.50	7	21	32.71	36.11	TIE
U844	C	4	H	1	121	29.31	36.11	7	1	37.11	43.91	TIE
U844	B	5	H	3	143	37.43	43.91	6	100	41.50	47.98	TIE
U844	C	5	H	2	92	40.02	47.98	6	54	45.64	53.60	TIE
U844	B	6	H	3	113	46.63	53.60	6	125	51.25	58.23	TIE
U844	C	6	H	2	17	48.77	58.23	5	6	53.16	62.62	TIE
U844	B	7	H	1	111	53.11	62.62	5	100	59.00	68.51	TIE
U844	C	7	H	1	149	58.09	68.51	6	14	64.24	74.66	TIE

Splice Tie Table (top) and Splice Interval Table (bottom)

folders with the Site number and either "_Splice_Intervals" or "_Splice_Ties". Splicing functions will let the user decide which to use. There are also functions under the CODD_Data menu and the CODD_Image menu to convert from one format to the other and to display each table. Use exactly these names for the data waves.

The Splice Interval Table requires the following columns: Site_Top, Hole_Top, Core_Top, Type_Top, Section_Top, Interval_Top, CSF_Top, CCSF_Top, Section_Bottom, Interval_Bottom, CSF_Bottom, CCF_Bottom, and TIE.

The Splice Tie Table requires the following columns: Site_Bottom, Hole_Bottom, Core_Bottom, Section_Bottom, Interval_Bottom, CSF_Bottom, and CCSF_Bottom, Tie, Site_Top, Hole_Top, Core_Top, Type_Top, Section_Top, Interval_Top, CSF_Top, and CCSF_Top.

Splicing Images and Data

Once all of the images have been shifted to CCSF scaling and the CCSF waves have been calculated for the data the user is ready to splice. The first step is to load the splice tie data into either a folder named Site_Splice_Ties (e.g. U1338_Splice_Ties) or Site_Splice_Intervals (e.g. U1338_Splice_Intervals). Column names need to follow the naming protocol shown in the examples above.

Both of these tables are useful and can be chosen according to what the user is doing at the time. The Splice_Interval table has each spliced section listed on a single line. It is easy to read the limits of the spliced section from a single core and is handy when sampling. The Splice_Tie table lists each tie point on a single line and is useful for examining the points chosen in different holes to make the splice.

Run Splice_Images. A new image will be formed in the root: directory. This is the spliced image. To view it, run Append_Spliced_Image.
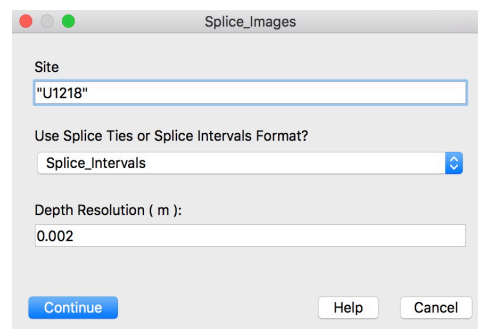
Run Splice_Data . New spliced waves will be formed in the appropriate data folders. Older experiments used the "_Splice" designator for the spliced waves but it is now recommended that the "S" option is used. Spliced data names then have the same protocol as individual hole wave names with "S" as the hole designator (e.g. U849S_GRA_GRA).
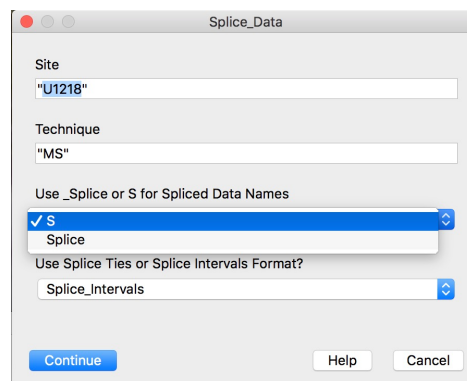
Append_Splice_Data will plot the old version of spliced data waves. Use Append_Data for the "S" version. Note that while the data from individual holes may be shared from another experiment and the binary waves reside elsewhere, the newly created Splice waves will reside in the Splice experiment folder. They will not automatically be shared and will need to be loaded into new experiments independently of the individual Hole GRA data waves.
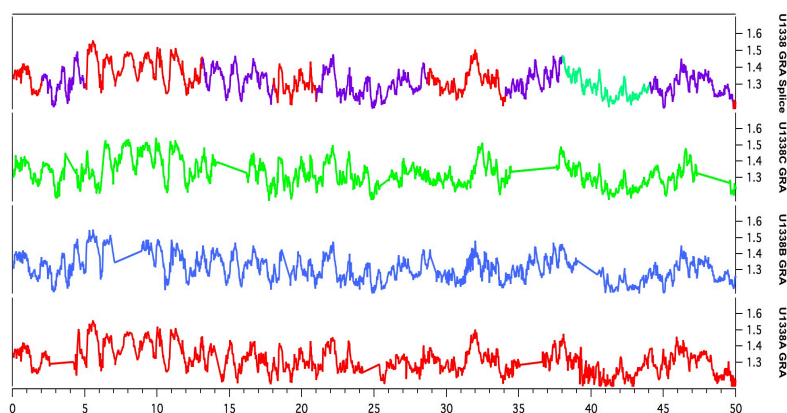
In addition to Splice waves of all of the input data, like Core, Section, Interval, Type etc. a new hole Index wave named will also be created. The Hole_Index waves are a plotting device that allows the user to take advantage of IGOR's ability to vary the colors of the plot of a wave depending on the value of a second wave.

Another way to view the position of splices is by running Calculate_Splice_Lines. This macro forms (or overwrites) a folder in the splice experiment named "Site"_Splice_Lines(e.g. U1338_Splice_Lines). Waves within the folder are used to plot splice interval limits on an image plot when a second macro, Plot_Splice_Lines is called. The tops of splice intervals are indicated by yellow vertical lines; the bottoms by red lines.
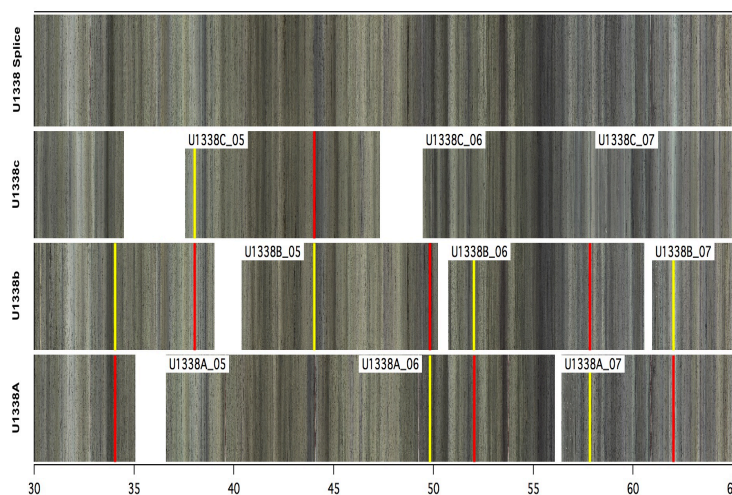
Depth To Age
Fit a Depth to Age Curve

In order to convert a depth wave to an age wave an age model is required. Age data such as paleomagnetic reversals or biostratigraphic datums gathered from multiple holes often do not fulfill this function because there can be inconsistencies in depth-age progression, that is, due to sparse sampling and core stretching, the same age will occur at different CCSF depths. One way to overcome this problem is to fit a curve to the depth-age data using AgeDepthFit. This function fits a polynomial (order 2 for straight line) through age and depth data. Data and the fit line are plotted and updated each time the function is run. Users may try different order fits to the data simply by changing the order number variable in the command line version of AgeDepthFit. The function also adds the sedimentation rate based on the age model (curve fit) to the graph. Output is in a data folder named AgeDepthFit and consists of the copies of the data waves used for the fit (with 0 added at the top of each if specified), age and depth fit waves, and a wave holding the fit coefficients. The fit waves should be used for DataDepth2Age and ImageDepth2Age (next section).
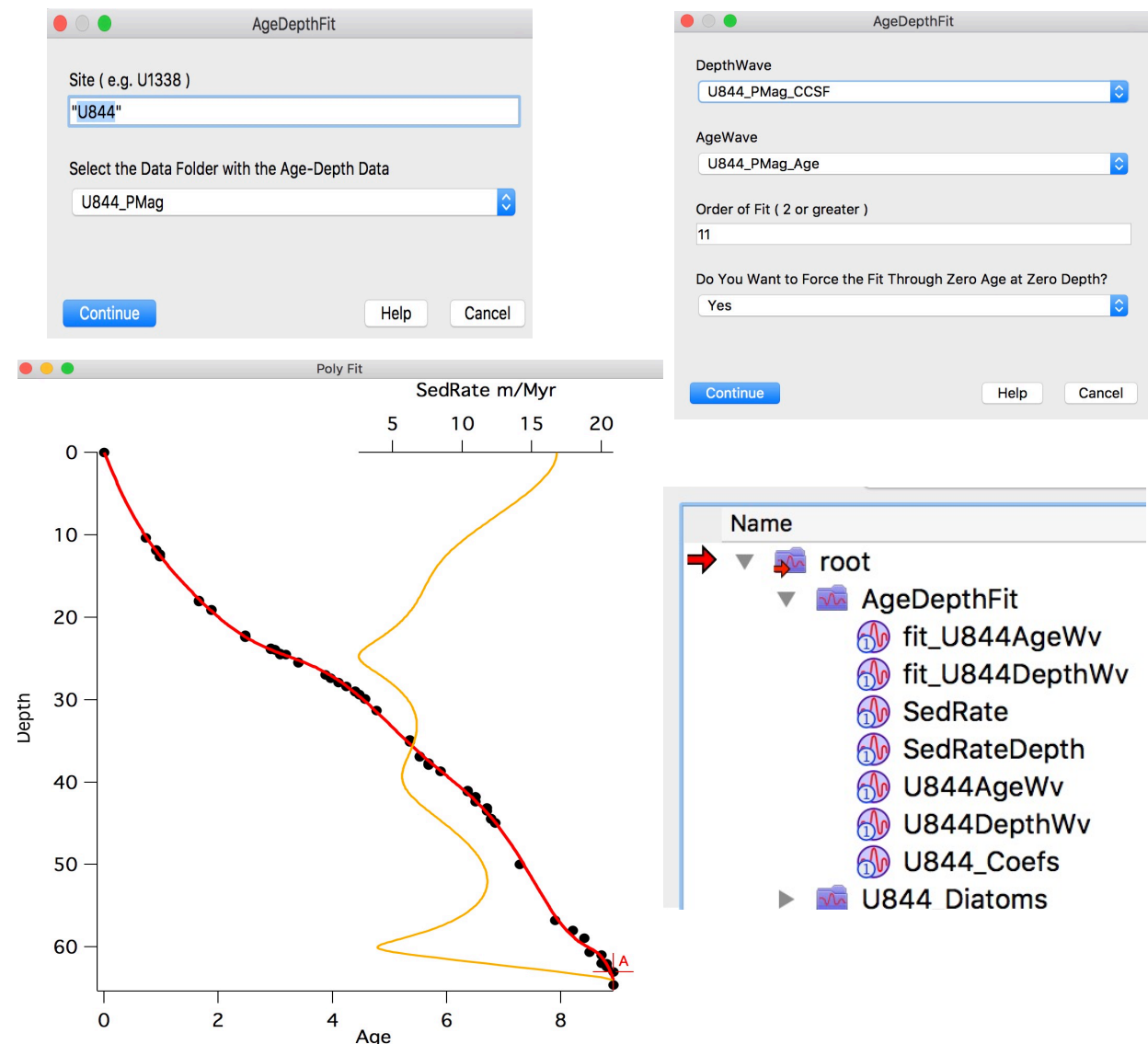
Depth to Age for Data

Run DataDepth2Age to generate an age wave corresponding to a selected depth wave based on an age model (use millions of years) specified by the user. The age wave is created in the same folder as the depth wave. The function checks the age model to be sure that there are no reversals in age or depth. For data such as biostratigraphy where there may be some confusion, run AgeDepthFit to fit the data and then use the fit waves for the age model.

There are four different prompts involved in calculating ages for depth waves. The first two identify the folder holding the age model and then the depth and age waves that constitute the model. The third and fourth prompts select the folder holding the data and then the depth wave to be converted to age. The user enters a new name for the output wave - which will appear in the same data folder as the depth wave.
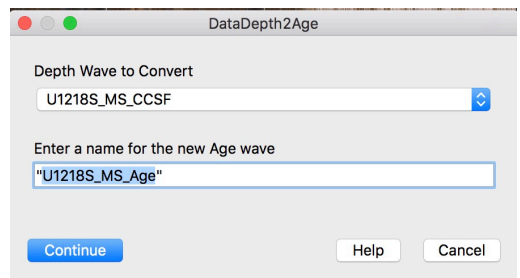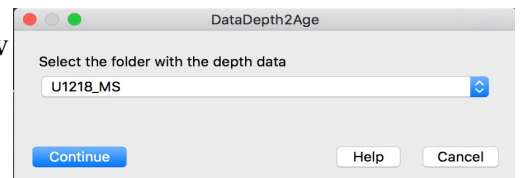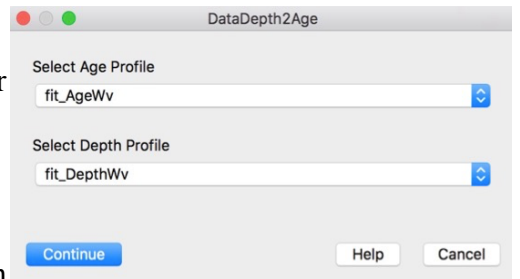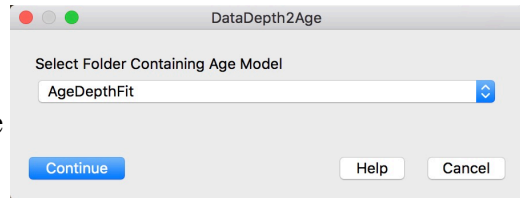
Depth to Age for Images

Run ImageDepth2Age to generate an age-scaled core image corresponding to a selected depth-scaled image based on an age model (use millions of years) specified by the user. The age-scaled image is created in the same folder as the depth-scaled image with the same name with "_Age" appended. User selects the age scaling increment. The function checks the age model to be sure that there are no reversals in age or depth. For data such as biostratigraphy where there may be some confusion, run AgeDepthFit to fit the data and then use the fit waves for the age model.
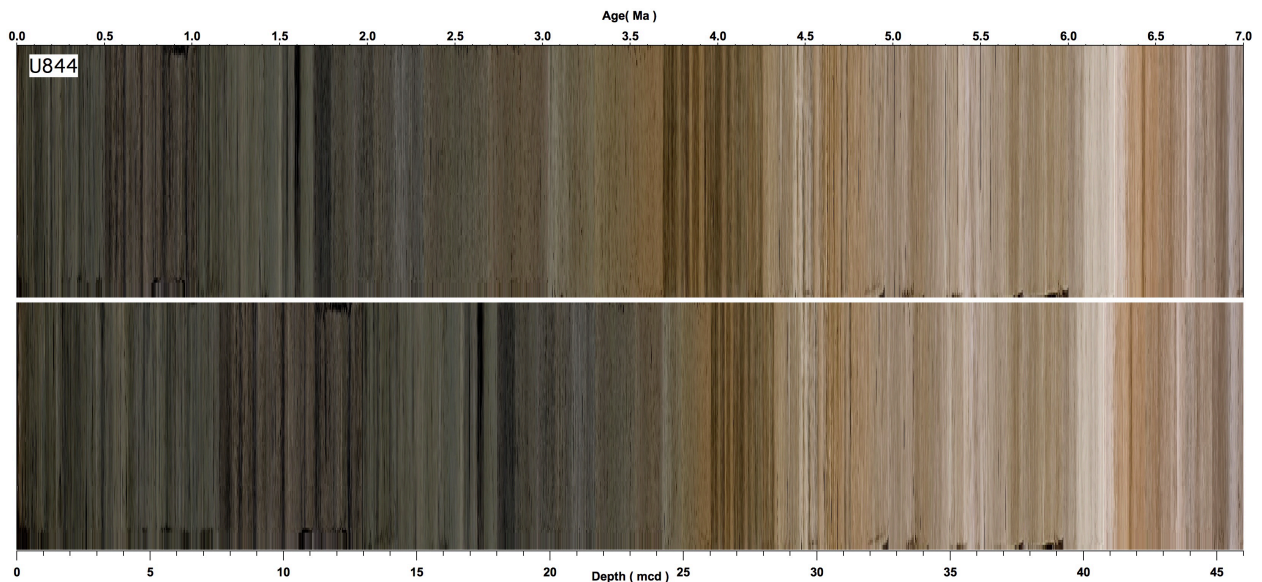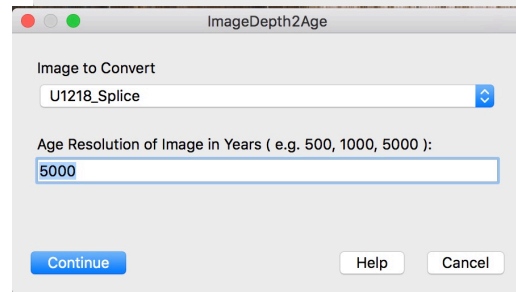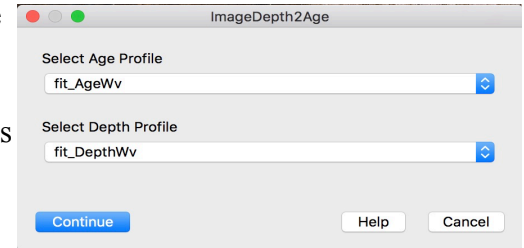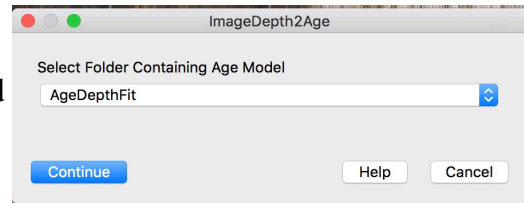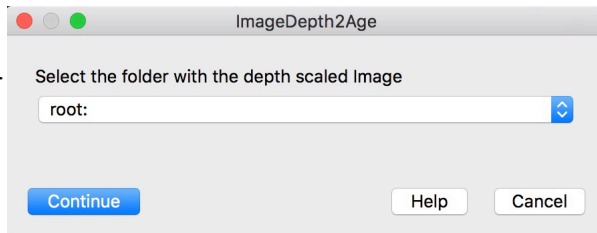
Age-scaled image (top) and depth-scaled image (bottom).

Lighting Correction

Prior to implementation of core scanners aboard the JOIDES Resolution the only core images collected were in the form of the core table photographs. In an earlier section instructions were detailed on the method of "cutting out" section images from core table photos and their subsequent combination into complete core images. However, there is a problem with the raw section images as a result of the fact that the cores were illuminated by a single light source from above when the pictures were taken (or images captured). The intensity levels of the colors in the image are lower at the ends of the sections than in the middle. Although subtle when viewed as individual sections, once the sections have been concatenated, the uneven lighting shows up as dark/light banding at regular 1.5 m intervals. This is particularly vexing in sediment cores, which tend to have natural banding related to compositional changes. The lighting correction, while not perfect, does alleviate the worst of the problem by fitting curves to R, G, and B intensities for each core and using the fits to smooth the intensity profiles.
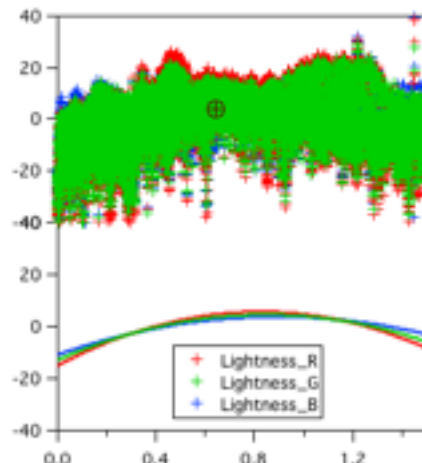
Intensities offset by median values of R, G, and B profiles through 6 sections of a single core from Hole U1143A. Upper - all values, lower - Fits to data.

Applying the Lighting Correction

Lighting correction is a complex process, but it can be automated so that it is easy to apply. To correct a single core image simply load open the Includes_Core_Table_Photos file and run Assemble_Core_Image. The correction process also adds a WaveNote to the image that can be seen at the bottom of the information panel of theData Browser that indicates that the image has been corrected.
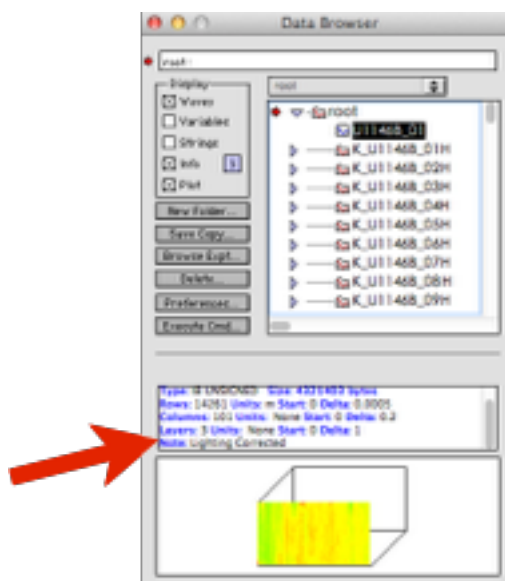
Batch Processing of the Lighting Correction

The macro Batch_Lighting_Correction will step through all of the "K_" folders in the root directory. It has no inputs. It will automatically save the experiment after processing cores that are an even multiple of 5 - i.e. every 5 cores, just in case the process is interrupted and IGOR quits.

Data Editing

Data acquired using scanning systems such as the core tracks aboard the GLOMAR Challenger and the JOIDES Resolution are indiscriminate in the sense that they do not have a way to evaluate the quality of the core at the point of measurement, and so are prone to collect "bad" data points. Bad data points make data analysis and interpretation more difficult than it need be and thus, it is left to the investigator to clean bad data from raw data output.
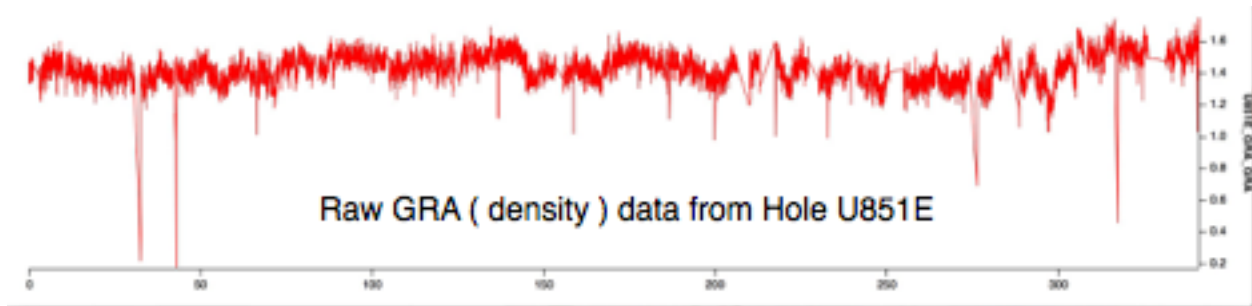

The Data_Editing menu appears when the file "Includes_Data_Editing" is opened. There are 3 functions that de-spike entire waves at one time and 2 that allow the user to replace individual points with either a NaN or an average value. The 3 whole wave editors create a new wave whose name is the same as the original data wave with "_DeSpk " appended.

Auto_Edit is the simplest of the 3 editing macros. The user specifies a range of acceptable data values and all data outside of that range are replaced with NaN's (blanks) in the new de-spiked wave.
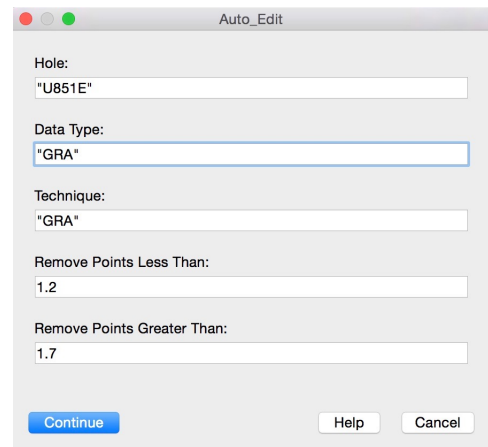
Median_DeSpike

Median de-spiking involves defining how much a data point deviates from a local median value. Points exceeding the limit are replaced by the median value. In the function the limit is set as a multiple of the standard deviation of the wave. The multiple may be any number and is not limited to integers. The width parameter defines the number of points used to define the local median value. Note that you need to plot the new " _DeSpk " wave to see the results.

U851E GRA Density after running Median_DeSpike (1.5σ, 51pts)
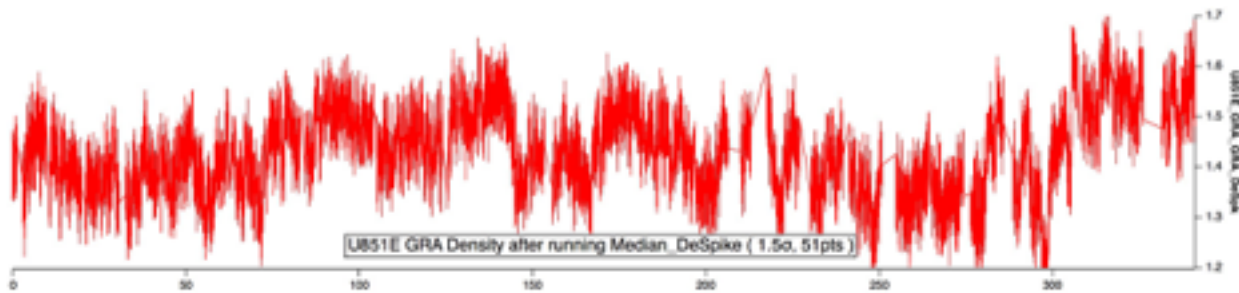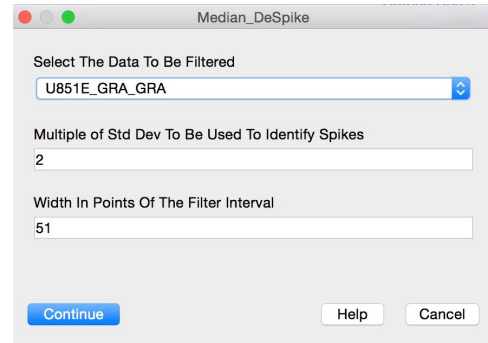
DeSpike_Via_Smoothing

DeSpike_Via_Smoothing uses a smoothed data curve as a reference for eliminating outliers in the data. The process requires several steps:

Step 1. Use the **Smooth** function under the **Analysis** window to create a continuous smooth curve of the data. Plot the smooth curve on top of the data using the depth wave of the data as the X axis.

GRA data with smoothed curve superimposed.
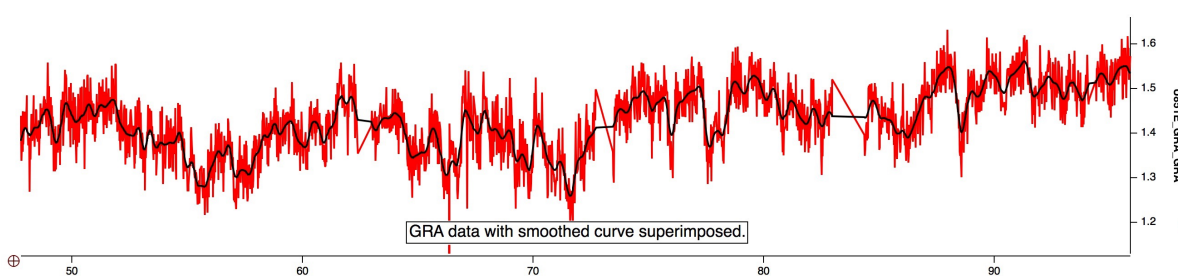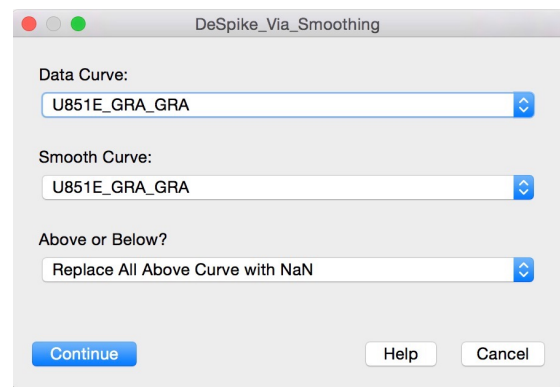
Step 2. From the command line, shift the smooth curve up or down so that it defines a boundary between good and bad data.

Step 3. Run DeSpike_Via_Smoothing.
Inputs for the macro allow the user to identify the Technique, the smooth curve, and whether the user wants to replace all data above or below the curve. Data points that fall above/below the smooth curve will be replaced by blanks (NaNs). Note that you will need to plot the new "_DeSpk" wave to see the results.
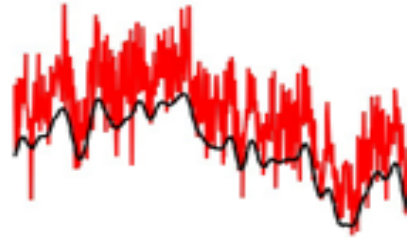
Step 4. Shift smooth curve, using the command line, to the opposite side of the data and run DeSpike_Via_Smoothing again. The data should now be rid of the most obvious spikes and outliers. The smoothed curve can be discarded.

NaN_Replace/Avg_Replace
        At this point the data are reasonably clean. However, the user may want to perform a final data edit of the "_DeSpk" wave on a point-by-point basis. NaN_Replace and Avg_Replace are functions that allow the user to replace individual data points with blank values interactively. I like to use the following procedure:

Step 1. Plot the data versus depth and change from the default wave display mode of lines connecting points to **Lines and markers** in the **Modify Trace Appearance** dialog. It is further useful to make the colors of the markers dependent on their Section number using **Set as f(z)...**, since often bad data are collected at section ends due to end effects of the measuring system. Finally, by right-clicking somewhere within the graph window (but not on the data wave) you will be able to change the background color of the plot to make the colors of the sections easier to see.
        It can also be very instructive to plot data in the same manner on top of core images. In practice I like to have both versions showing while editing. Note that you may shift the depth ranges of both plots simultaneously using the Change_Depth_Range macro (shortcut command-5).
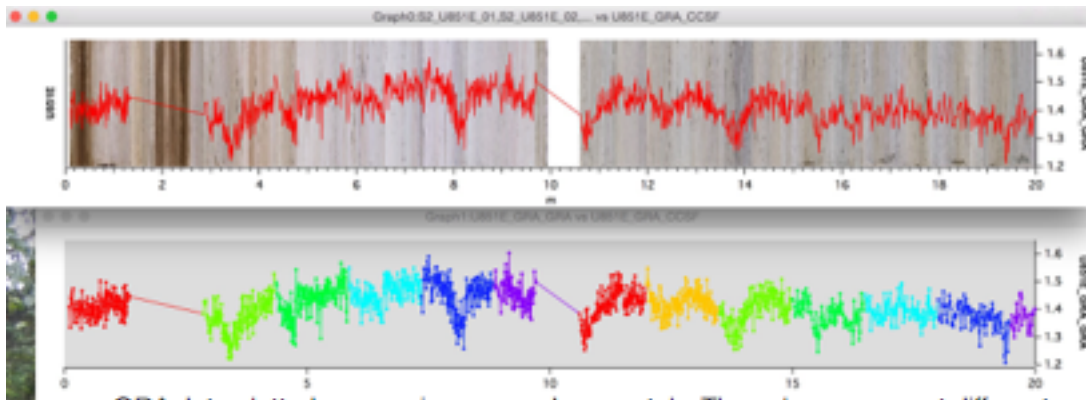
GRA data plotted on core images and separately. The colors represent different sections in each core. Data from Core 1, Section 2 is missing. No data for CC's

Step 2. Place **Cursor A** on a bad point and run Nan_Replace or Avg_Replace. The point will be replaced in the data wave by a blank/average value and the cursor will advance to the next data point..

Shortcuts (Command-8/9) have been installed through the Includes file so that the user need not bother with using the macro menu for each data replacement. BEWARE though, that continuing to hold down the shortcut keys will continue to edit points.

There is no "**UnDo**" for Nan_Replace or Avg_Replace. Users are advised to save often so that if mistakes are made the experiment can be **Reverted** to a previous state that will not cost much in lost time. Additionally, I like to save a copy of the raw data wave so that I can, if desired, recover single data points or sections of data if I suspect later that what I thought were bad points were actually valid.

Core & Data Stretching

Once core offsets and a splice have been set for a Site most sampling is initially done in the splice - that is - from sections of holes that are included within the splice. Parts of cores that are not in the splice are what we consider to be "**Off-Splice**". It is not unusual that features in off-splice sections do not totally align with those inside the splice. It is not unusual for the mis-alignment to be 50 cm or more. Mis-alignment may be the result of coring disturbance or natural variability of the sediment structure. In either case it is useful to re-align depths of off-splice features with those in the splice so that measurements made outside the splice, like paleomagnetic or biostratigraphic datums, can be better integrated with those inside. This is especially important for heavily sampled sites, where investigators may have to sample off-splice when there is no longer sufficient material in the spliced intervals for their needs.

In the following we describe a set of functions to create stretched images and data depth scales as well as a set of tie points between off-splice and splice depths that can be used to interpolate between them.

Stretching Setup

Stretching can be done with either data or images and in practice the user commonly will use both at the same time. A common set of ties between splice and off-splice depths can be generated using any of the open stretching graphs and quickly applied to all data and images.

Assemble the data - Before running the macros the user needs to assemble (1) the spliced core image, (2) the individual core images, (3) any data desired, including both Hole data and Spliced data, and (4) the Spliced Interval data for the Site. Remember that if the data were spliced in a separate experiment that shared data from individual holes, the user will need to share waves from both the splice and measurement technique experiments.
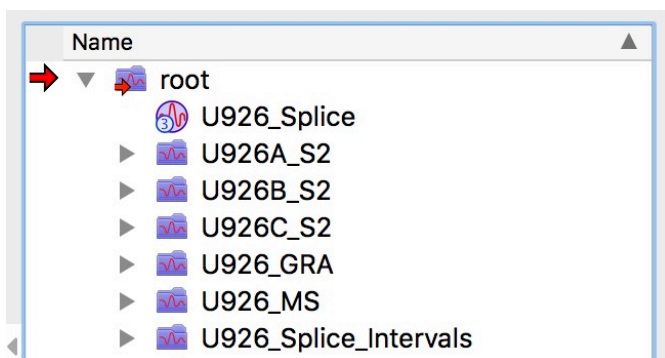
With the data in place, run Core_Stretch_Image_SetUp. The user is prompted for the Hole and then the Core image to stretch.
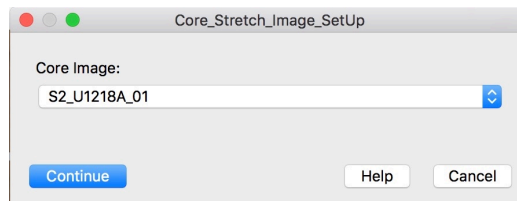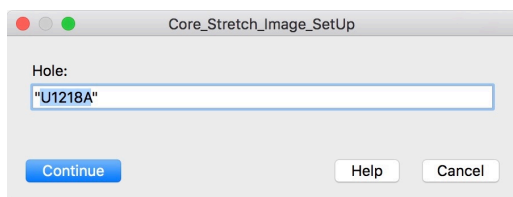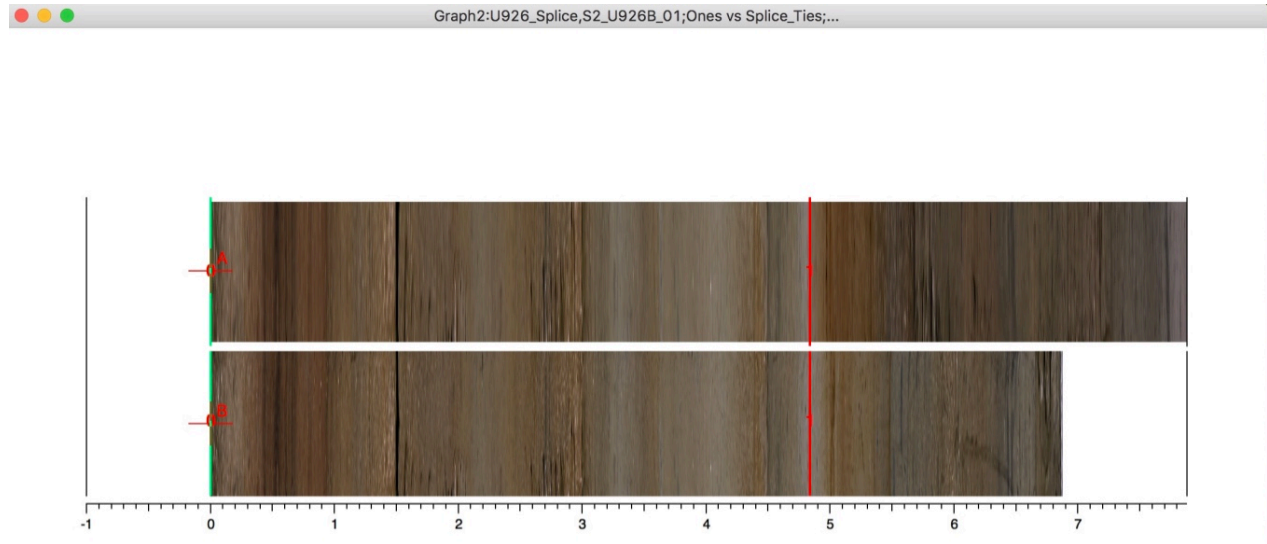
Stretch image setup. Hole image is on the bottom, spliced image in the middle.

The spliced Site image will be plotted above a core image and a Core Tie Table will be opened. If there is a splice interval within the core image the depths of the splice interval will be entered into the tie table. Run AppendSpliceLines4Stretching to place vertical bars on the plot indicating the top (green) and the bottom (red) of the splice interval. Cursor A is attached to the spliced image and Cursor B to the Hole image. A new folder is created for results of stretching for each hole (e.g. U926B_Tie_Points).

Point	Core_Ties	Splice_Ties
0	0.00	0.00
1	4.84	4.84
2		

Data may be plotted on top of the image plot or in a separate graph by running AppendData4Stretching. Add splice lines to the data plots by running AppendSpliceLines4Stretching.
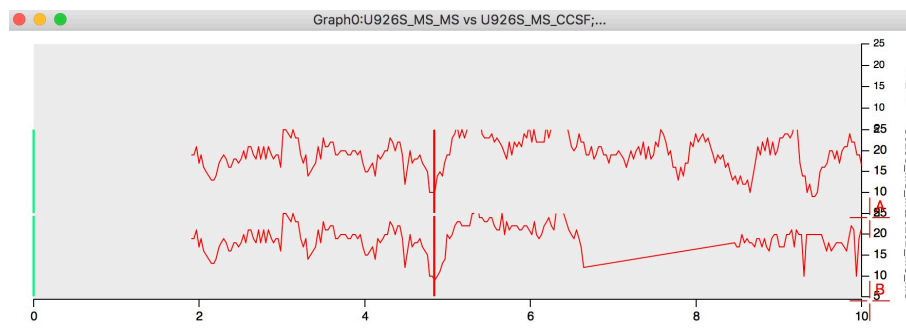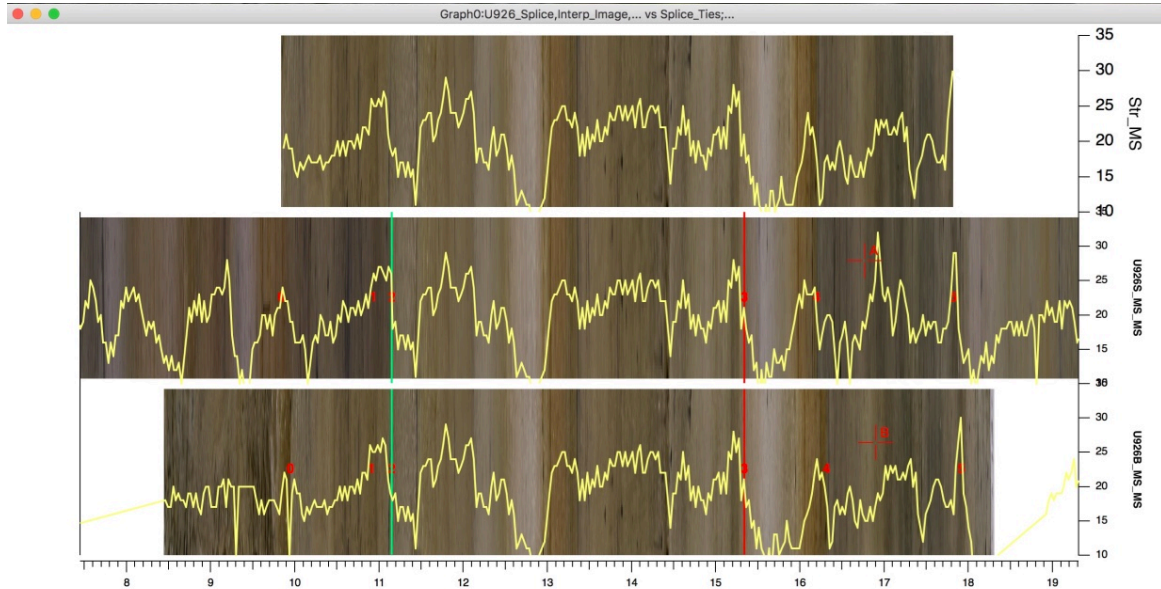
Recording Ties

Once the data and image graphs have been set up, ties are identified by placing Cursor A on the Spliced data or image and Cursor B on the corresponding feature on the Core data or image. No ties are needed within spliced intervals. Run Record_Stretch_Ties (⌘-1). The new ties will be added to the core tie table. Ties may be recorded from multiple windows and data sets and will appear in all graphs simultaneously. Ties are removed by cutting out the appropriate row from the Core Tie Table.

Point	Core_Ties	Splice_Ties
0	9.95	9.84
1	10.91	10.93
2	11.15	11.15
3	15.34	15.34
4	16.32	16.20
5	16.95	16.83
6	17.92	17.82
7		

Core_Tie_Table — 9.9461517

Stretching

At any point after registering the first pair of ties, or a single tie if splice ties are already in the Core Tie Table, the user may stretch the core image by running Core_Stretch (⌘-2) and data by running StretchOffSpliceData (⌘-3). There is no prompt for the core image stretch command, but the data stretching function must be run for each data type.

StretchOffSpliceData
Hole (e.g. U925B): "U926B"
Technique: "MS"
Data: "MS"
Continue   Help   Cancel

In practice, it is useful to open a notebook and record the stretching commands. All of the commands can then be run repeatedly by selecting them and choose "Execute Selection" in the right-click menu, e.g.

```
Core_Stretch( )
StretchOffSpliceData_F( "U926B", "MS", "MS" )
StretchOffSpliceData_F( "U926B", GRA, "GRA" )
StretchOffSpliceData_F( "U926B", "RSC", "AStar" )
```

Move on to the next core by running Next_Core (⌘-4).

Stretching Strategy

Stretching of data and images is a relatively quick process and may be repeated many times in the course of stretching a single core. Don't be afraid to try a tie point and then reject it. After having stretched many cores, we have found that it is useful, if possible, to work from outside-in. That is, find a tie point at each end of a core and do an initial stretch. Then work in towards the middle of the core, matching finer features with each iteration. Of course, there are times when identifiable features are simply not present, in which case we opt for no stretching, simply matching the depth of the ends of the core to the same depths in the spliced data or image. Fortunately, with the amount of data produced for more recent coring it is rare that there is nothing to tie together off-splice and spliced data.

Finishing Up Stretching

Each time core images or data are stretched, duplicate values from the Core Tie Table and a copy of the stretched image are stored in a Tie Points data folder for each Hole. Once the user completes core stretching for an entire Hole, the final step is to run Concatenate_Ties. This function strings together tie pairs from all cores in the data folder into a master Core and Splice tie table that can be used to interpolate original CCSF (mcd) depths into stretched depths that are better aligned with Splice depths. Besides the Core and Splice tie depths, a third wave records the core number responsible for each tie pair.

- ▼ 🗀 U926A_Tie_Points
  - S2_U926A_01_Str
  - S2_U926A_02_Str
  - S2_U926A_03_Str
  - S2_U926A_04_Str
  - U926A_Core_Ties
  - U926A_Splice_Ties
  - U926A_Ties_CoreNum
  - U926A_01_Core_Ties
  - U926A_01_Splice_Ties
  - U926A_02_Core_Ties
  - U926A_02_Splice_Ties
  - U926A_03_Core_Ties
  - U926A_03_Splice_Ties
  - U926A_04_Core_Ties
  - U926A_04_Splice_Ties
  - U926A_05_Core_Ties
  - U926A_05_Splice_Ties

Off-splice stretched depths can be calculated by running CCSF_2_OffSplice.

CCSF_2_OffSplice

Hole
"U926A"

Technique
"MS"

Depth Scale (e.g. CCSF)
"CCSF"

Continue   Help   Cancel
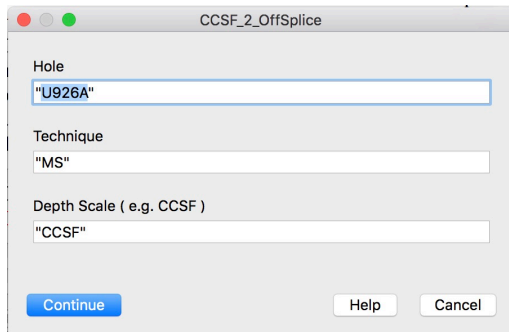
U926A_Core_Ties	U926A_Splice_Ties	U926A_Ties_CoreNum
0.06	0	1
0.946226	0.942137	1
2.11052	2.29433	1
3.2208	3.51451	1
3.75694	4.1286	1
7.01011	7.06903	2
7.85224	7.73913	2
8.18029	8.08918	2
10.4827	10.6235	2
12.1209	12.1518	2
12.947	12.9559	2
15.2374	15.2882	2
15.9435	15.9123	2
16.1743	16.5264	3
16.6263	16.9365	3
16.9564	17.2085	3
17.6105	17.8506	3
18.5226	19.4568	3
19.9889	20.541	3
20.851	21.0011	3
21.4831	21.5131	3
22.0452	22.0352	3
22.7993	23.1394	3
23.2694	23.8215	3
26.2681	26.2478	4

A new wave is created in the selected data folder.

- ▼ 🗀 U926_MS
  - U926A_MS_CCSF
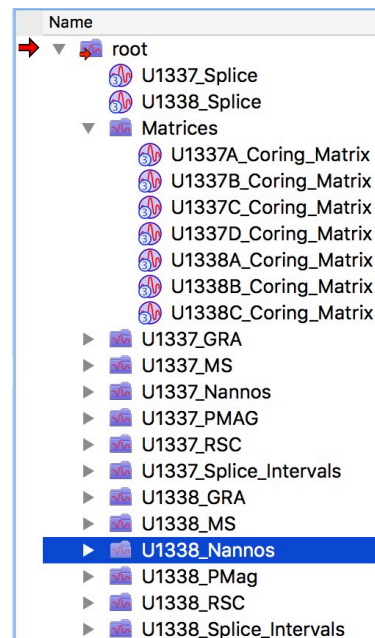 - U926A_MS_CCSFOffspl
  - U926A_MS_Core
  - U926A_MS_CSF
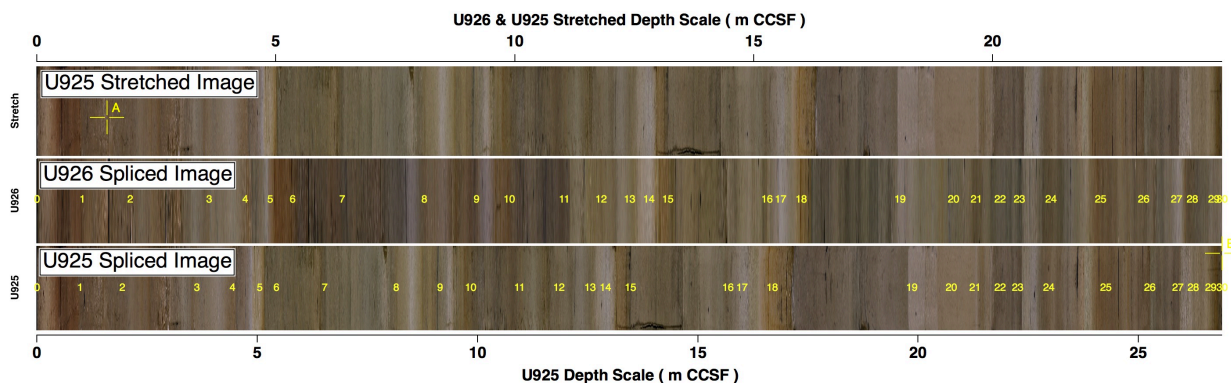
U925 spliced core image stretched to match U926. Numbers are tie points between the two sites. Note the different depth scale top and bottom.

Site to Site Correlation

Although different sites may have age models based on biostratigraphic markers or magnetic reversals, the best way to correlate from site to site is to identify common features in the depth domain and form one common "pseudo-depth" scale. In practice this is typically accomplished by picking a single site as a Base Site against which all other Stretch Sites are compared. Once a common pseudo-depth scale has been produced, a single age model can be applied to all of the correlated sites. The following set of functions allows the user to identify tie points using core images and data plots and to efficiently stretch images so that site to site correlations can be established in great detail.

Site to Site Correlation Setup

Before starting to identify site to site tie points the user must first assemble the proper data from the Base Site and a Stretch Site, including spliced images, spliced data such as GRA and MS, Splice Interval folders for each site, folders with biostratigraphic or paleomagnetic datums, and Coring Matrices for all of the Holes of each site (placed in a data folder named "Matrices"). Discrete age models, if available, may also be included although if age waves have already been calculated for any of the spliced data they can be used.

Once all of the data have been loaded, open Includes_Site_2_Site_Menu.

A new CODD_Site_2_Site menu will open.
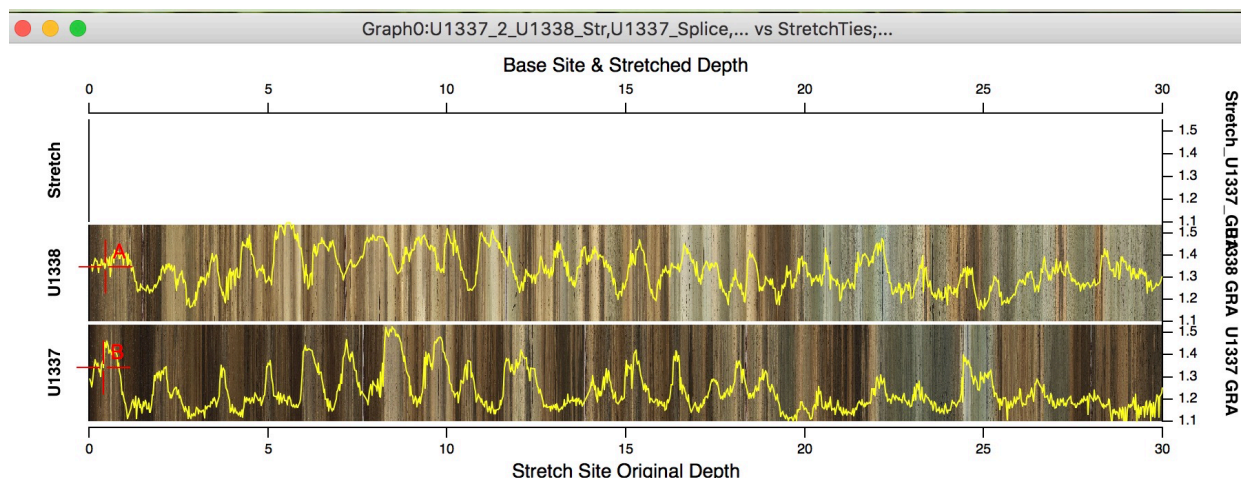
Run Site_2_Site_Image_SetUp. The user will be
prompted for the Base Site and the Stretch Site. A new
graph will open with the Base Site image plotted in the
middle third of the frame and the Stretch Site plotted at
the bottom. The top and bottom of the graph will have
different depth scales, the bottom for the Stretch Site and

the top for the Base Site and the Stretched image. Five new waves will also be created in root:
and opened in a new table. Base and Stretch ties are for tie depths. Tie type identifies which data
were used to pick a tie (Images, GRA, MS, etc.). DeltaTies is the absolute difference in depth
between tie pairs. TieStr is the calculated stretching between adjacent pairs of tie points - think
of it as a relative sedimentation rate based on DeltaTies. Data may also be plotted on top of the
images by running Append_Site_2_Site_Data.

After the image graph has been set up, the next step is to run Site_2_Site_Data_SetUp. The user
is prompted for the Base Site, Stretch Site and the data type. A new graph with the Base Site data
plotted in the middle third and the Stretch Site data plotted in the bottom third. If not already
formed by the image setup, the tie table waves will be created and the Tie Table opened as above.
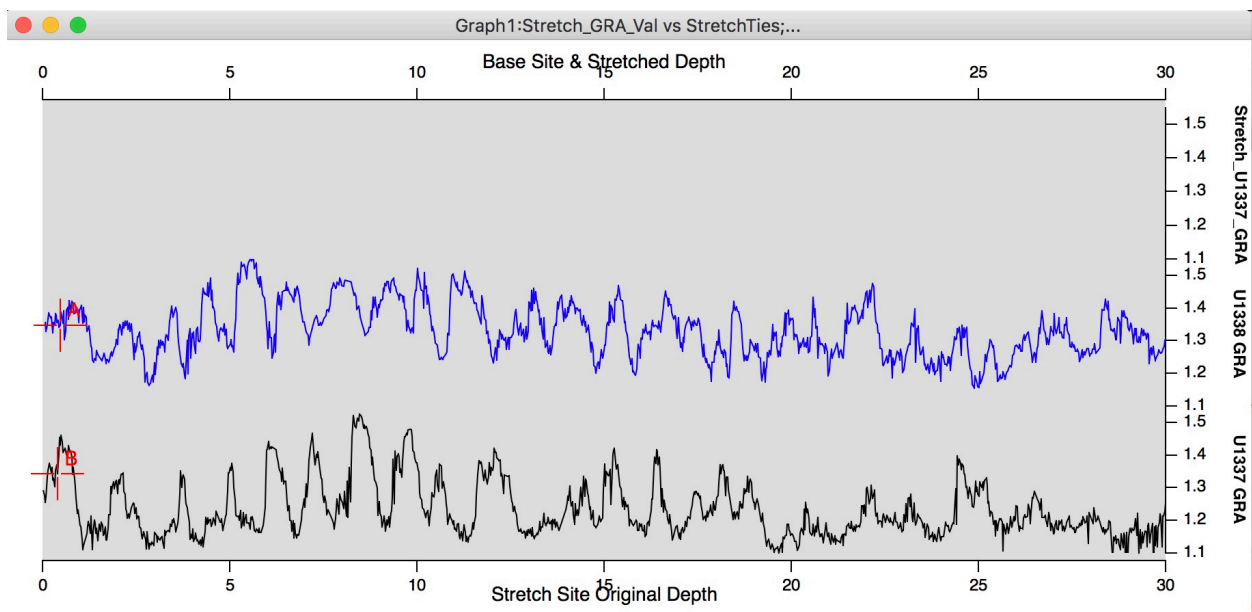
Adding Datums

      The more data types that can be set up, the greater the confidence the user will have in tie point selection. There are also other aids available to help with site to site correlation, such as biostratigraphic and paleomagnetic datums and existing age models for each site. With a data graph active, run BioPMag_Plot. The user will be prompted for the Site, the datum folder, and finally the datum Hole, depth, description (e.g.species or reversal) uncertainty (if any) and the tag angle and font size. To change the angle or font size simply re-run BioPMag_Plot. To remove datums use Delete Annotations from the regular Igor Graph menu.
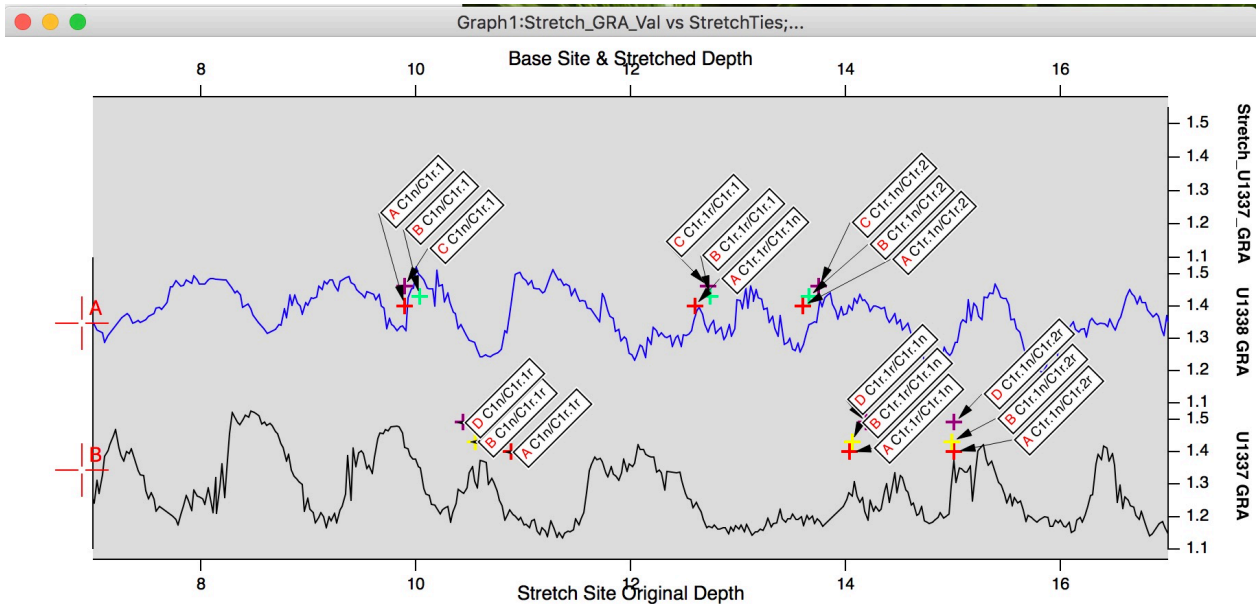
The first letter (in red) is the Hole that is the source of the datum. The rest of the text tag identifies the datum. In the example above the paleomagnetic reversals are tightly grouped, but the example below illustrates how biostratigraphic datums from different holes can be more dispersed due primarily to the sparse nature of sampling, particularly aboard the drillship where sample are examined in core catchers only - one determination per 9.5 m. Still, these datums can be useful in identifying general intervals of correlation.
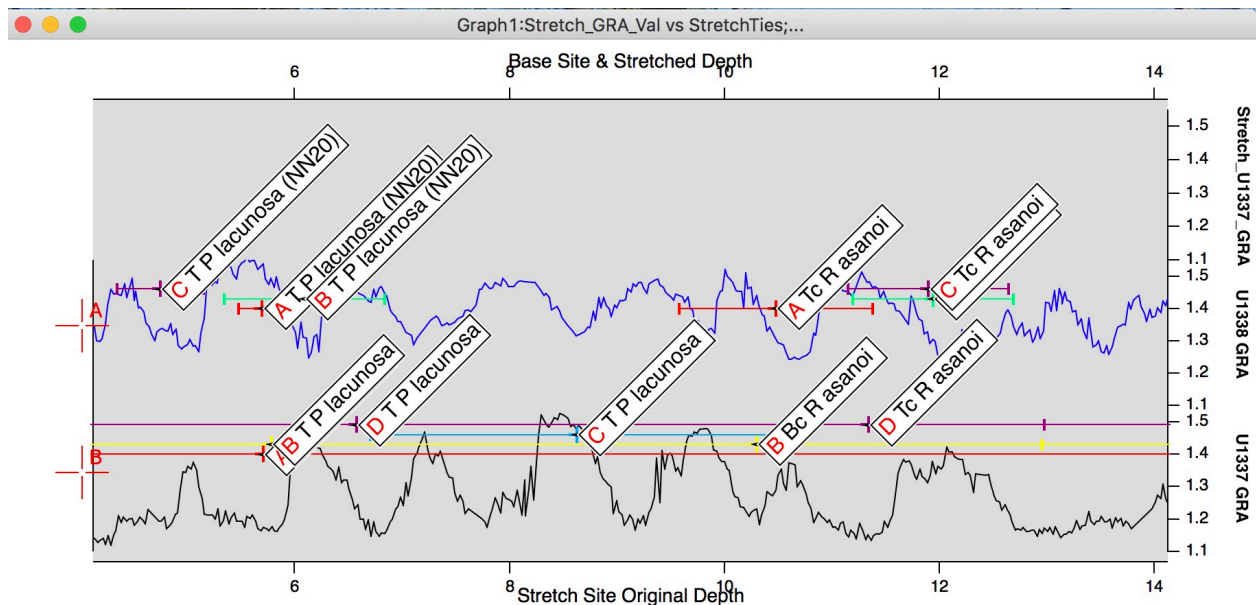
Adding Ages

A second guide to help with site to site correlation is the use of existing age models for the Base and Stretch Sites. The way CODD does this is to plot age values on data or image plots versus depth at user specified age intervals.

The first step is to run AgeModelTies. The user will be prompted for a folder containing age models for both Base and Stretch Sites. These may be pairs of any depth and age waves. In the example at right we use results from running AgeDepthFit (see the **Depth to Age** section of this Guide). However, if ages have been calculated for one of the data types, such as GRA or MS, the depth and age waves in those data folders may be used as well After the age model folders have been identified the user will be prompted for age and depth waves for the Base and Stretch site individually.
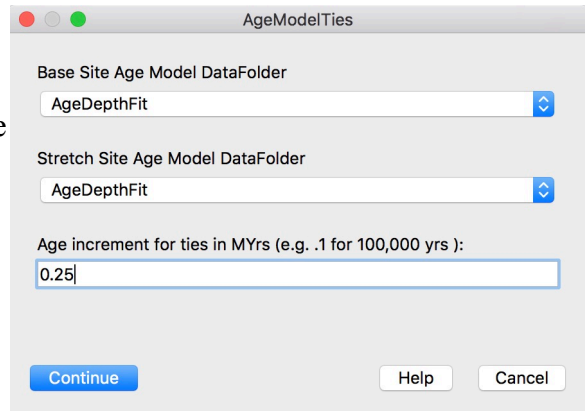
The final step is to run AppendAgeModelTies. Age values are printed on the graph in the lower portion of the fields for the Base and Stretch data or images.

43
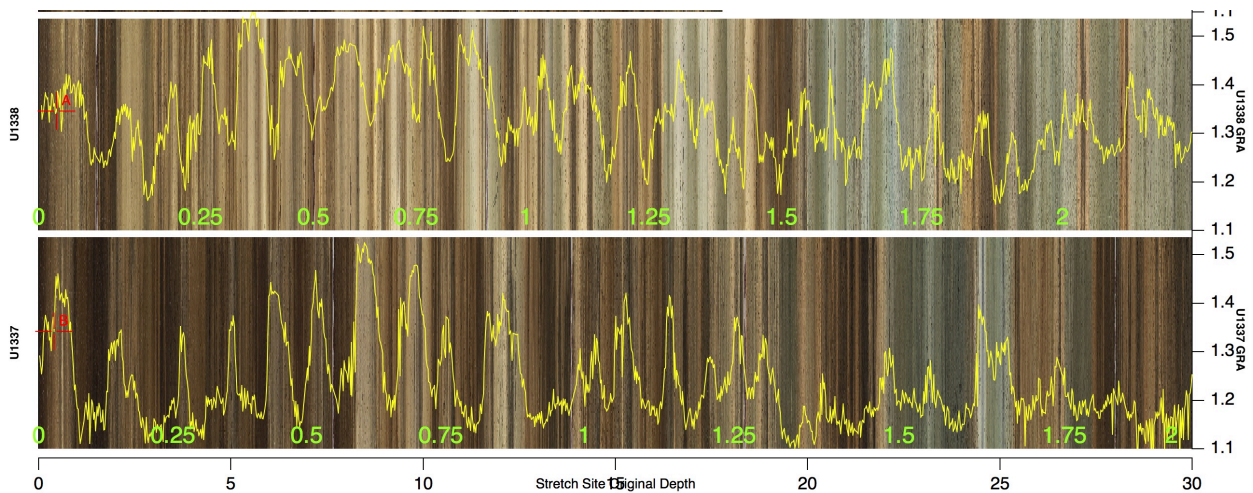
Picking Ties

        Once all of the setup is finished it is time to pick tie points. Cursor A is placed on the Base Site data or image and Cursor B goes on the Stretch Site data or image. Running Site_2_Site_Ties (⌘-1) records the ties in the TieTable. As the new ties are added a green dot will appear on all open data graphs at the placement of the tie pair and a tie number will be shown above the dot. Only the numbers appear on the images graph. The data wave type that the user used to select the tie is recorded in the TieType wave. The difference between successive ties is calculated as is the stretching between pairs of ties.
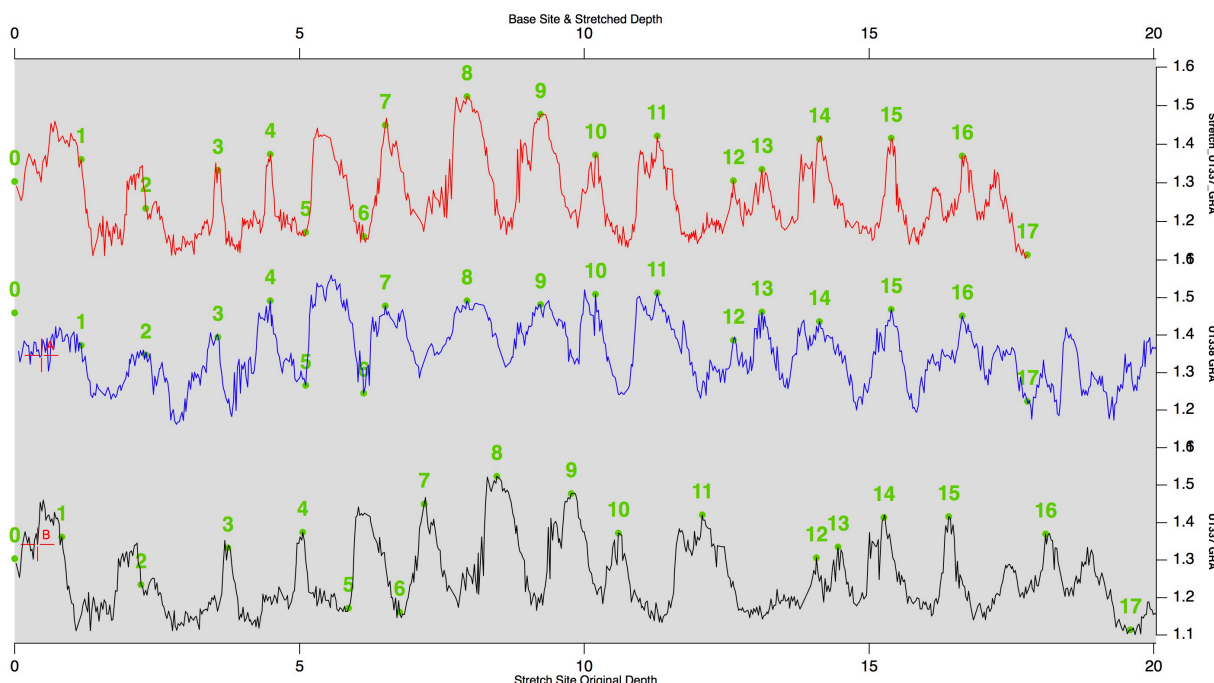
After identifying a few ties run Site_2_Site_Image_Stretch to see the stretched image within the limits of the ties and run Site_2_Site_Data_Stretch for each type of data in use to see the same for data waves. Although tie pairs can be removed by simply cutting them from the table, it is best to use Delete_Ties to keep all of the values coordinated.

BaseTies	StretchTies	TieType	TieStr	DeltaTies
0	0	Images		0
1.17	0.83	Images	0.709402	-0.34
2.3	2.22	Images	0.965217	-0.0799999
3.57	3.74	Images	1.2125	0.17
4.49	5.06	GRA	1.2968	0.57
5.11	5.86	GRA	1.37662	0.75
6.13	6.76	GRA	1.03658	0.63
6.51	7.19	GRA	0.95	0.68
7.94	8.4685	GRA	0.943923	0.5285
9.23	9.7685	GRA	0.947978	0.538501
10.2	10.5985	GRA	0.942477	0.398499
11.28	12.07	GRA	1.12268	0.789999
12.62	14.08	GRA	1.43864	1.46
13.12	14.46	GRA	1.29891	1.34
14.13	15.26	GRA	0.781458	1.13

Although there is no dedicated function at this time to make graphs for TieStr and DeltaTies it is useful to plot them via regular Igor tools as bar graphs (sticks to zero mode). large fluctuations may indicate a mis-placed tie point.
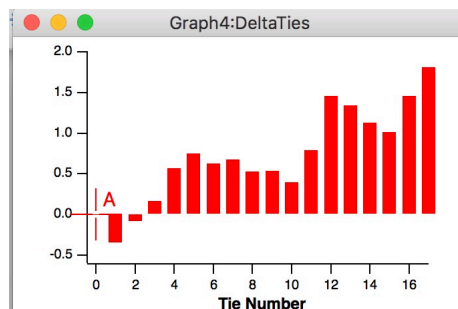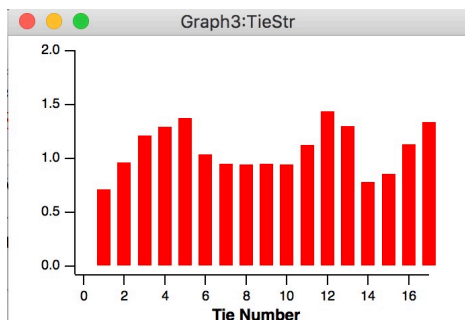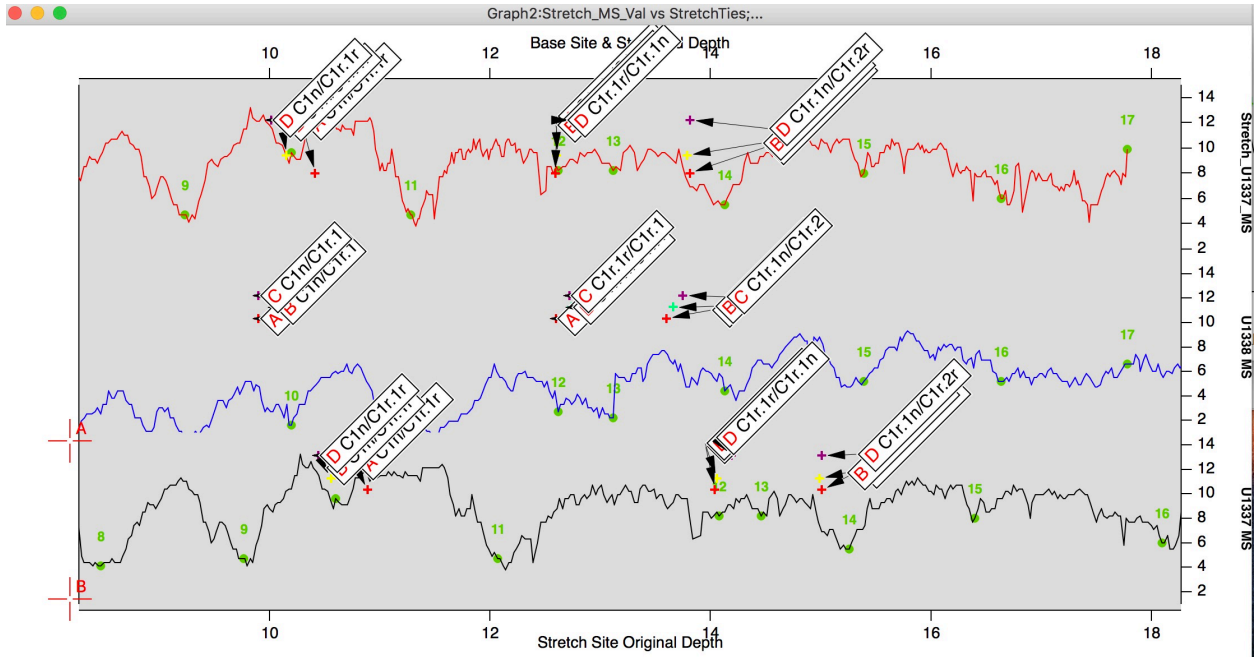
Datum positions may also be added to the stretched data plots by running
Stretched_BioPMag_Plot.

Site to Site Correlation Strategy

    When the data and image setup plots initially open the top and bottom axes are the same. The default is 0 - 30 m. This view can become cumbersome quickly in the case where sedimentation rates between the sites is markedly different. Our preferred strategy is to first select fairly widely spaced ties based on datums or other distinct features seen in data or images in both sites.
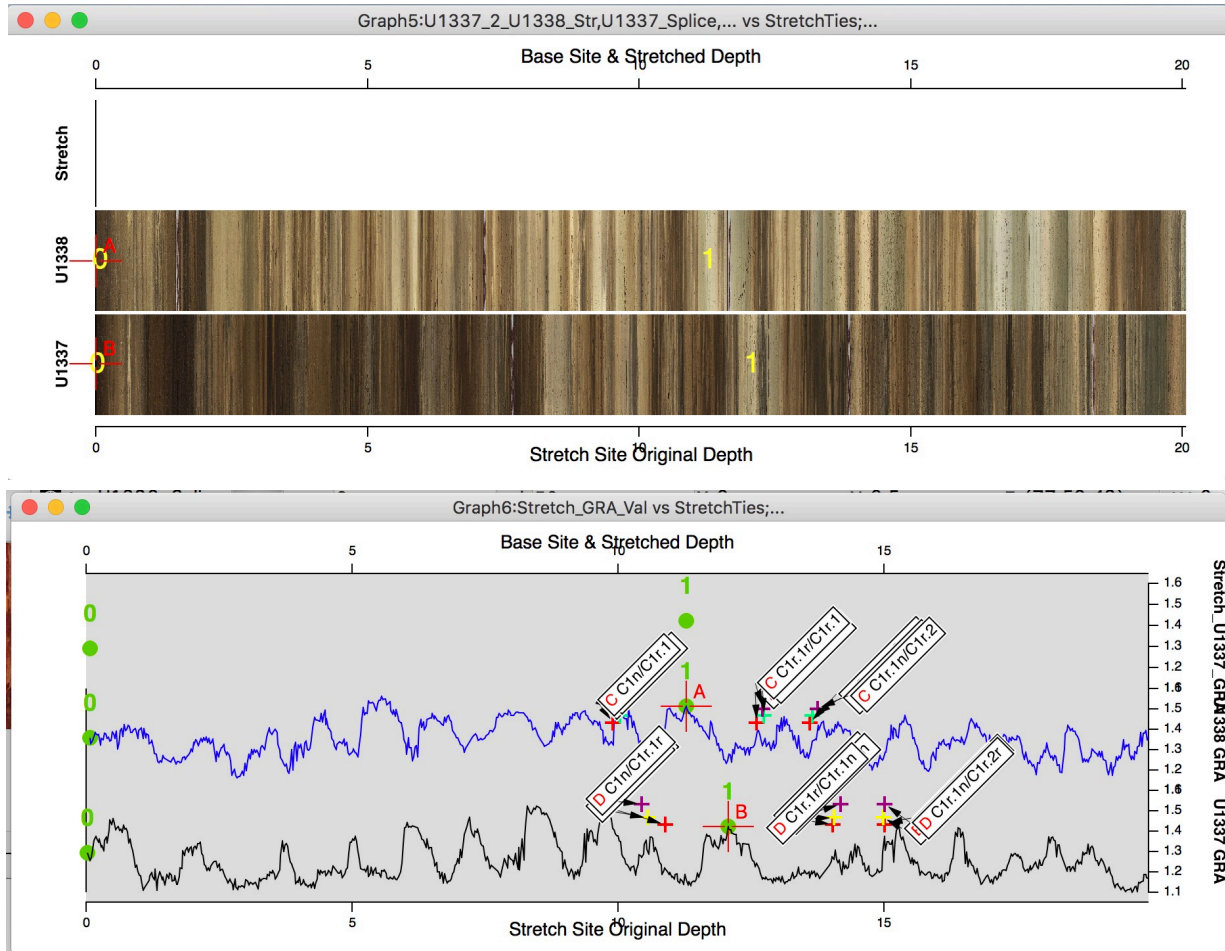

Next, run AlignTiePointRange (⌘-4). This function resets the top and bottom axes to reflect the depth intervals between selected tie points. The effect is "stretch" the Stretch Site data and image to roughly match the Base Site.

Note the different depth scales on the top and bottom of the example. With the intervals displayed roughly equivalent it becomes a simpler task to identify features common to both sites. Select more tie points between the initial choices and then run Site_2_Site_Image_Stretch and Site_2_Site_Data_Stretch. Then add or adjust as necessary. Continuing to run AlignTiePointRange (⌘-4) over increasingly narrow depth ranges can sometimes allow for correlation at remarkably fine scales.
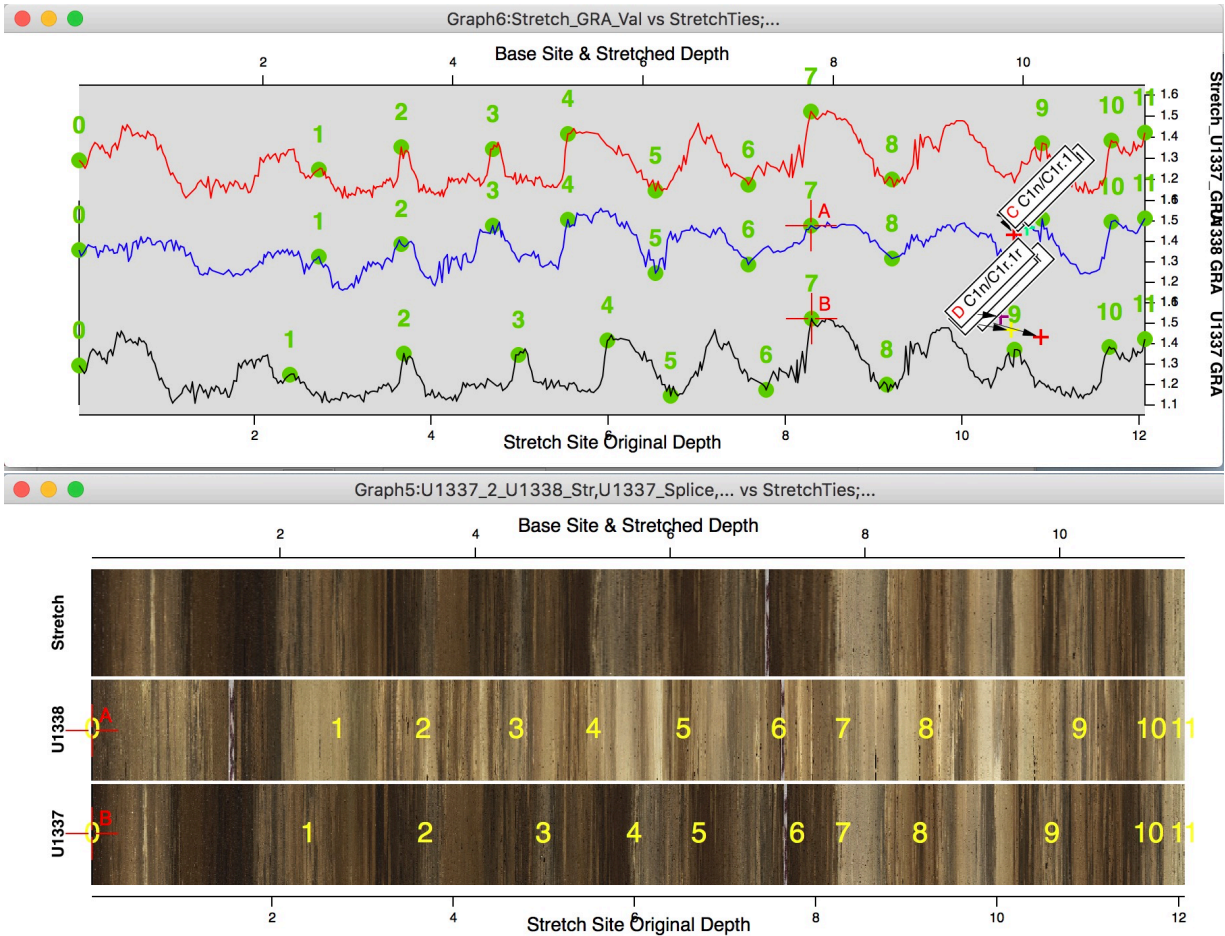
Interpolation Using Site to Site Ties

Ties between sites can be used to calculate pseudo-depths for any selected depth waves by running Site_2_Site_DepthInterp. The user can specify any output name for the resultant wave. The preferred name is the depth wave name with the name of the Base Site appended - the same naming protocol as the waves formed when using running Site_2_Site_Data_Stretch.
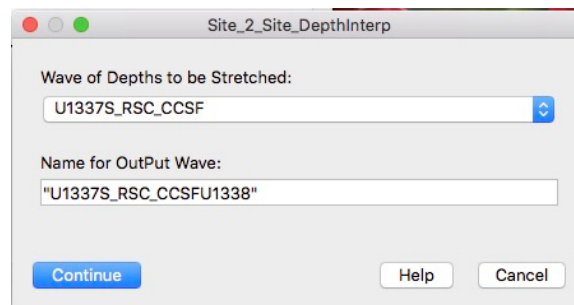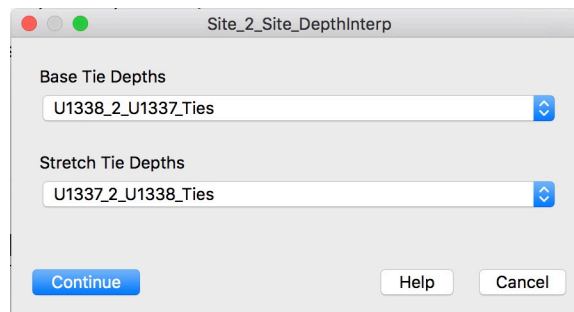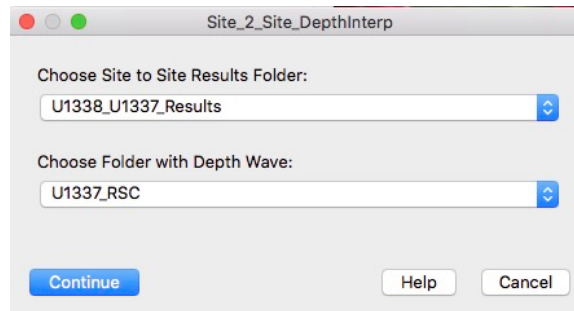
The Site to Site Tie List

Once the task of tying 2 sites is complete, the final step is to produce a list of ties that includes the complete identification of each tie point. It is entirely possible that CCSF depths for one or both sites may change as offsets and splices are reconsidered in the light of new data. However, it is rare that features that have been positively identified as correlating between sites will also change. By forming a Tie List with Core, Section, and Intervals, CSF and CCSF depths can quickly be recalculated and remain valid. The new depths can then be used to interpolate new data pseudo-depths.

Run Site_2_Site_CSITieList. A new data folder is formed named using the names of the 2 sites and a Tie Table is opened.
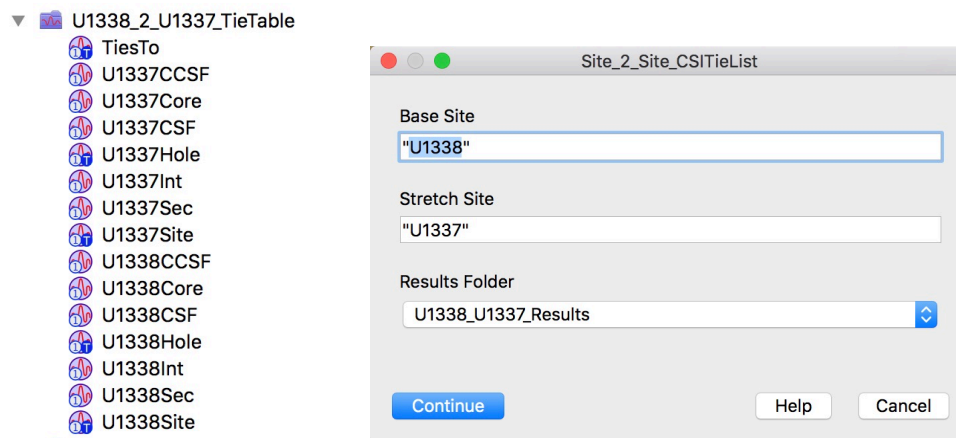
Point	U1338Site	U1338Hole	U1338Core	U1338Sec	U1338Int	U1338CSF	U1338CCSF	TiesTo	U1337Site	U1337Hole	U1337Core	U1337Sec	U1337Int	U1337CSF	U1337CCSF
0	U1338	A	1	1	0	0.00	0.00	Ties To	U1337	B	1	1	0	1.00	0.00
1	U1338	A	1	1	117	1.17	1.17	Ties To	U1337	B	1	1	83	1.83	0.83
2	U1338	B	1	2	80	2.30	2.30	Ties To	U1337	B	1	2	72	3.22	2.22
3	U1338	B	1	3	57	3.57	3.57	Ties To	U1337	B	1	3	74	4.74	3.74
4	U1338	B	1	3	149	4.49	4.49	Ties To	U1337	A	2	4	37	10.37	5.06
5	U1338	A	2	1	93	3.63	5.11	Ties To	U1337	A	2	4	117	11.17	5.86
6	U1338	A	2	2	45	4.65	6.13	Ties To	U1337	A	2	5	57	12.07	6.76
7	U1338	A	2	2	83	5.03	6.51	Ties To	U1337	A	2	5	100	12.50	7.19
8	U1338	A	2	3	76	6.46	7.94	Ties To	U1337	C	2	4	20	6.60	8.47
9	U1338	A	2	4	55	7.75	9.23	Ties To	U1337	C	2	5	0	7.90	9.77
10	U1338	A	2	5	2	8.72	10.20	Ties To	U1337	C	2	5	83	8.73	10.60
11	U1338	A	2	5	110	9.80	11.28	Ties To	U1337	A	3	1	118	16.18	12.07
12	U1338	A	2	6	94	11.14	12.62	Ties To	U1337	A	3	3	19	18.19	14.08
13	U1338	A	2	6	144	11.64	13.12	Ties To	U1337	A	3	3	57	18.57	14.46
14	U1338	B	2	4	74	12.84	14.13	Ties To	U1337	A	3	3	137	19.37	15.26
15	U1338	B	2	5	50	14.10	15.39	Ties To	U1337	A	3	4	101	20.51	16.40

Index