

eeDAP

Evaluation Environment for Digital & Analog Pathology

User Manual

2015

Table of Contents

I. Introduction	4
Abbreviations	5
Windows	6
II. WorkFlow	7
Launch EEdap	8
eeDAP windows.....	9
eeDAP Linux.....	10
eeDAP Administrator Input Screen window(c).....	11
MicroRT Mode - Serial port configuration	13
Slide Registration window	14
Register eyepiece and camera window	16
GUI Welcome page window.....	17
Experiment Interface.....	18
Preview with cross hairs window	20
Thank You window	21
Stage and Camera Control Utility Functions.....	22
Camera_stage_review window.....	23
Run eeDAP***	25
III. Software Requirements and Set Up	26
Read and extract WSI image.....	27
BIO formats.....	28
error.....	29
Image file is too large.....	30
Out of memory.....	31
Image Scope by Aperio.....	32
Matlab compiler runtime libraries	33
IV. Hardware requirement and Set Up	34
Stage	35
Lude Stage	36
Set up	37
Windows 7	38
Linux.....	39
Test.....	40
Camera	41
Requirements	42
Point Grey Grasshopper color (GRAS-03K2C-C)	43
Point Grey Flea2 color (FL2G-50S5C-C)	44
Native pixel format.....	45
Default format	46
Alternative format.....	47
Setup	48
Windows 7	49
Linux.....	50
Test.....	51
Alignment	52
Scanners	53
Hamamatsu Nanozoomer 2.0HT (at NIH ATC).....	54
Aperio CS (at NIH ATC and FDA White Oak)	55
Aperio T2 (at NIH ATC).....	56
Aperio Scanscope XT	57

Microscopes	58
Axioplan 2 Imaging microscope with an Axiophot 2 head	59
Olympus BX43	60
Reticles	61
Klarmann Rulings KR-429	62
Klarmann Rulings KR-32536	63
V. FDA hardware specifications	64
Cameras	65
Displays	66
VI. Program Variables	67
Assigned in the Input File	68
myData.settings	70
List of Variables Assigned in the Input File	71
myData.tasks_in, myData.tasks_out	73
myData.wsi_files	74
Qtype: Question types	75
VII. eeDAP Developers	76
Creating the stand-alone application	77
Deployment project	78
Build the stand-alone application	79
Package the stand-alone application	80
VIII. MicroRT Registration	81
Semi-Automated Registration	82
Low Resolution Registration	83
High Resolution Registration	85
Manual Registration	86
IX. camera and stage review application	88
Camera_stage_review window	89
X. Input File	91
Input File Header	92
Input File Body	93
Input File Body: Opening and Editing	94
Input File Body: List of ROIs and Corresponding Tasks	96
Default task input format	97
Customized Input File Body: List of ROIs and Corresponding Task	98
Selecting ROIs???	99
XI. Output File	101
XII. Directory Structure	102

I. Introduction

This is a tool to design and execute correlated optical and digital pathology reader studies. This application supports two modes of reading, Digital and MicroRT. Digital mode uses a set of regions of interest (ROIs) identified and extracted from digital whole slide images (WSIs). MicroRT mode offers the reader a real-time view of the identical set of ROIs from a glass slide, through the optics of a microscope. In other words, the two modes are registered to one another. Furthermore this tool allows for the administrator of the study to have a high level of control over the format of the tasks presented to the readers through specifications in an input file. Several simple default tasks are available to the administrator: multiple choice, continuous sliders, or binary responses. These simple tasks can be modified through a graphical user interface in conjunction with moderate MATLAB programming skills. Furthermore, the administrator can also specify whether the reader is allowed to pan or zoom while confirming in real time the registration of MicroRT ROIs and Digital ROIs.

The ability to either present the readers with an analog image (MicroRT) or a digital image of the same specimen ROI allows for the reduction or elimination of a large source of variability in comparing these modalities: the cells being viewed. The registration of areas strengthens the evaluation of digital pathology and its comparison to the microscope.

Abbreviations

WSI, WSIs	Whole Slide Image(s)
ROI, ROIs	Region(s) of Interest
MicroRT	Microscope Real-Time Viewing
Administrator	The (task) administrator is responsible for task set up and monitoring.
Reader	The reader is person evaluating the ROI's.

Windows

MicroRT Mode
Digital Mode
ROI Image Presentation Screen
Video Preview

II. WorkFlow

A brief introduction of the program is provided in the following section

A brief introduction of the program
is provided in the following section.
The program is accessible through the
executable.



Download EEdap at <https://github.com/DIDSR/eeDAP/releases>

Launch EEdap

Launch EEdap

Download the eeDAP.zip file and extract it. There are two version inside. One is for windows system the other is for linux system

eeDAP windows

1. Double click the eeDAO_windows.exe file. It will install the eeDAP in current directory.
2. Double click eeDAP.exe file to launch eeDAP



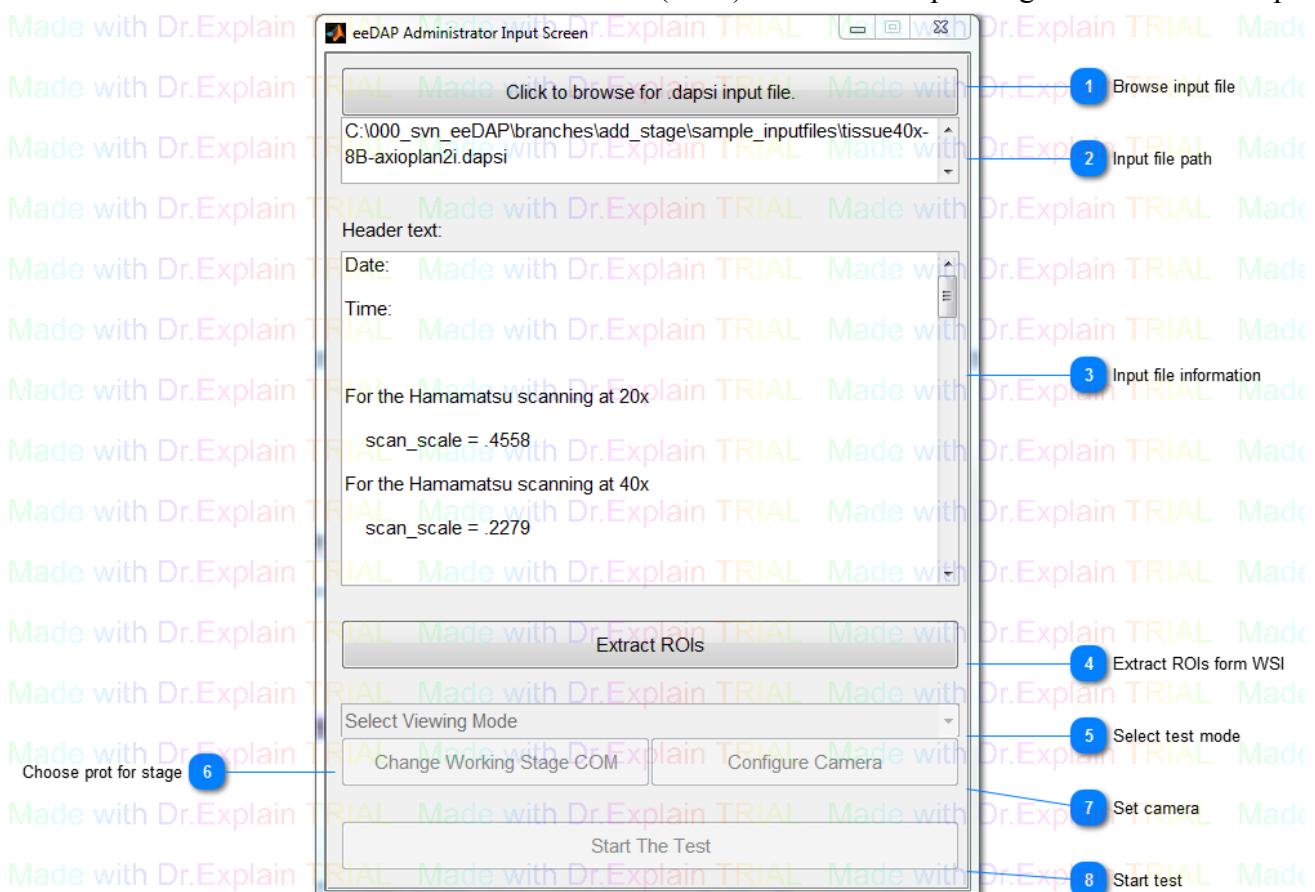
eeDAP Linux

1. Extract the eeDAP_linux.zip file
2. Open a terminal and visit the extracted folder
3. In terminal input "./run_eeDAP.sh" + " " + "matlab runtime libraries directory"

eeDAP Administrator Input Screen window(c)

In the first box the admin selects the input file containing the tasks; see the “Input File” section for formatting of this file. It is selected using the “Browse” button. Once chosen the header of the file is displayed in “Header text” box. Selecting “Extract ROIs” button will extract the experiment ROI’s from the WSI images. Extracting ROI’s may take a few minutes depending on the size of the experiment and the computer capabilities.

The drop down menu (Mode of Testing) allows for the administrator to choose either the Digital or MicroRT reading mode. The Digital mode allows for the evaluation of ROI’s identified and extracted from a whole slide Image. The MicroRT mode provides a real-time (RT) view of the same ROI’s through the optics of an adjoining microscope. Furthermore, the administrator is able to track a reader’s progress during MicroRT mode; a camera provides the administrator with the reader’s field of view (FOV) and the corresponding WSI ROI for comparison.



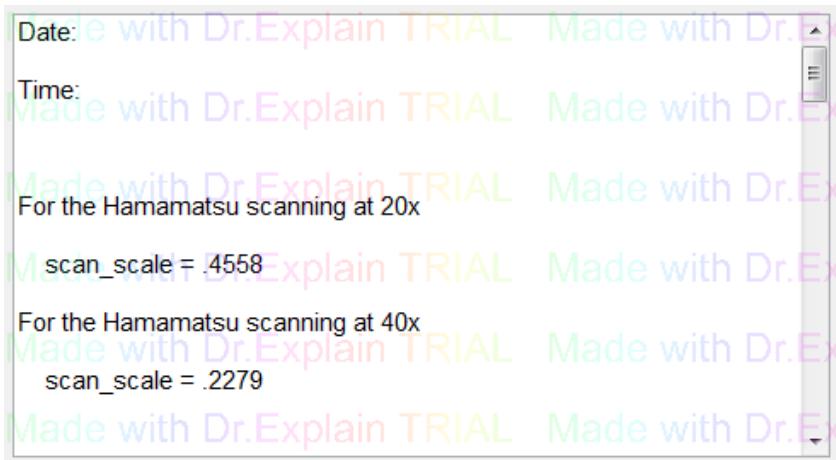
1 Browse input file

Browse input file from computer

2 Input file path

After browse input file, shows the path of the file

3 Input file information



After browse input file, shows the detail information of the file

4 Extract ROIs form WSI



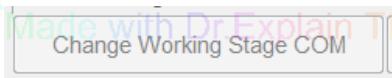
Following the input file, extract Region of interest from WSI

5 Select test mode



Choose Digital mode for just use WSI digital image in experiment. Choose microRT mode use both WSI digital and microscope live image in experiemnt

6 Choose prot for stage



In microRT mode, set the stage controller connect port

7 Set camera



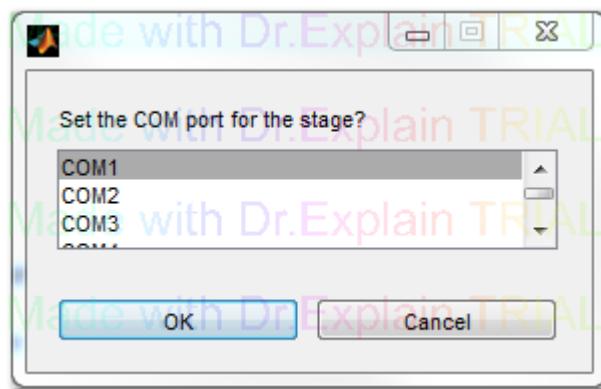
In microRT mode, set camera

8 Start test



Start test

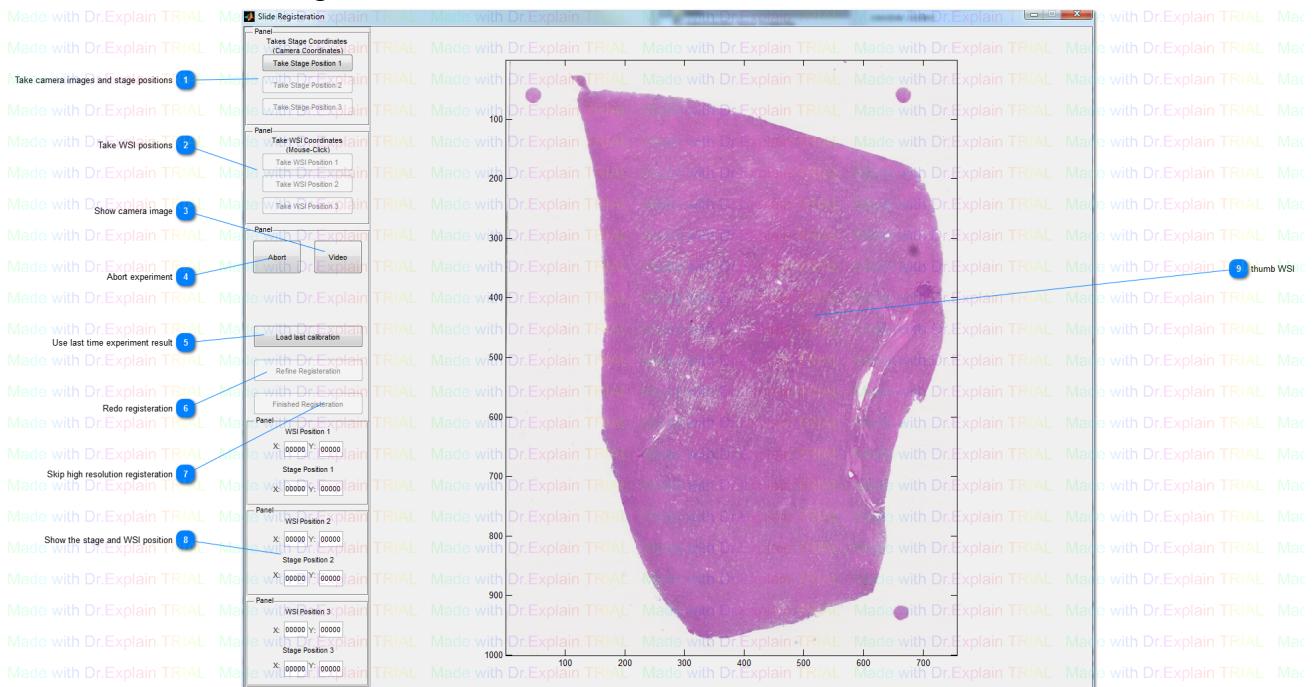
MicroRT Mode - Serial port configuration



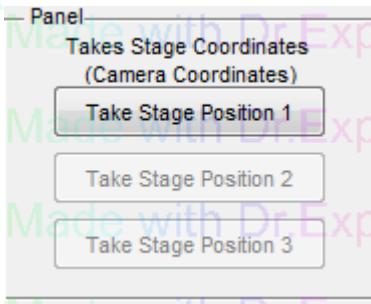
Setting the connection COM port for the microscope stage controller. To specify the correct COM port, open the PC's control panel then open Device Manager and expanded the Ports.

Slide Registration window

In the MicroRT mode the Slide Registration window will open. Also, the microscope camera will engage, and a video preview window will appear. In this step the Administrator verifies and registers the glass slides to the corresponding WSIs. This ensures that the reader's FOV in MicroRT will be the same as the ROI presented in the Digital mode. Refer to the "Registration" section for details.

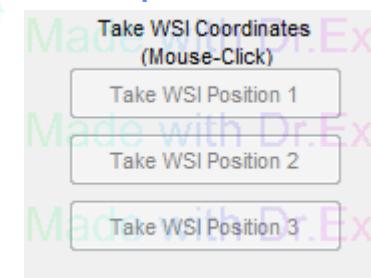


1 Take camera images and stage positions



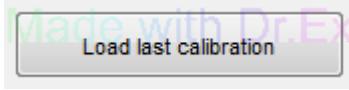
Take the microscope live image and stage position for 3 areas

2 Take WSI positions



Take the related WSI position for 3 areas

3 Show camera image

4 Abort experiment**5 Use last time experiment result**Load last calibration

Load last time 3 areas WSI and stage positions

6 Redo registrationRefine Registration

Redo registration

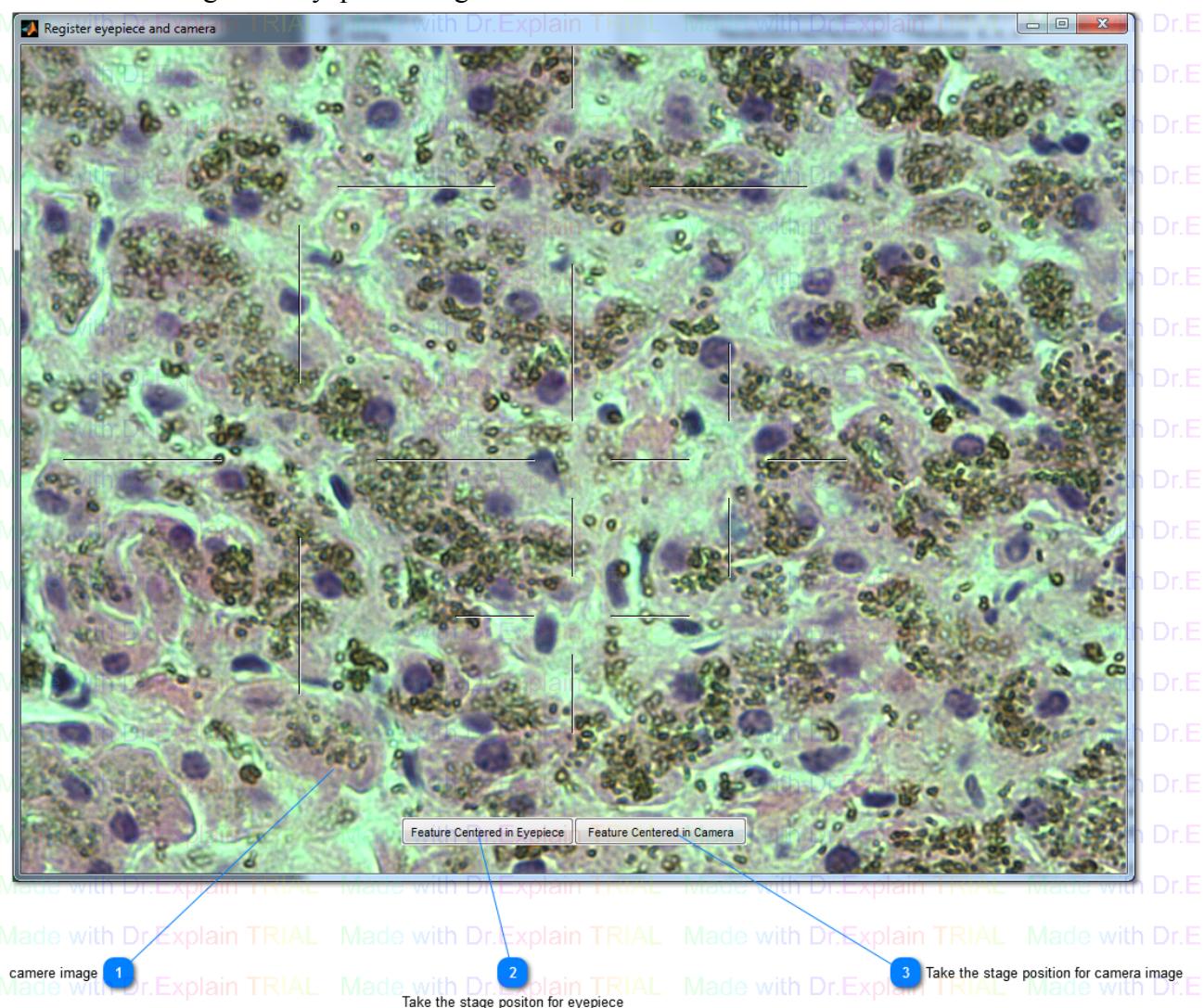
7 Skip high resolution registrationFinished Registration

Skip high-resolution registration, directly use low-resolution or last experiment postions

8 Show the stage and WSI position**9 thumb WSI**

Register eyepiece and camera window

Register the camear image with eyepiece image.



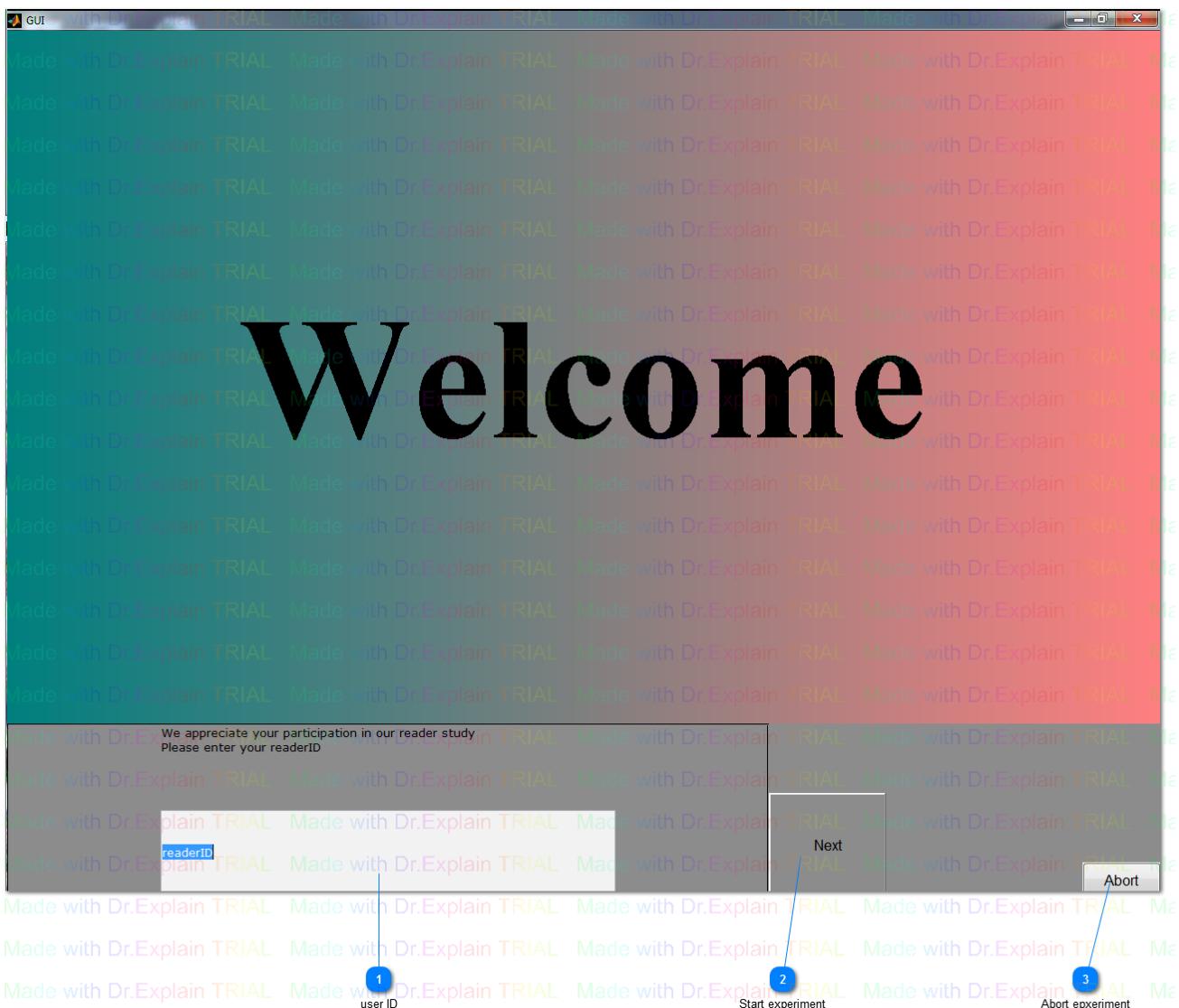
1 camere image

2 Take the stage positon for eyepiece

3 Take the stage position for camera image

GUI Welcome page window

The administrator will enter the reader's name or unique identifier and then proceed by clicking on the "Next" button.



1 user ID

In put user name

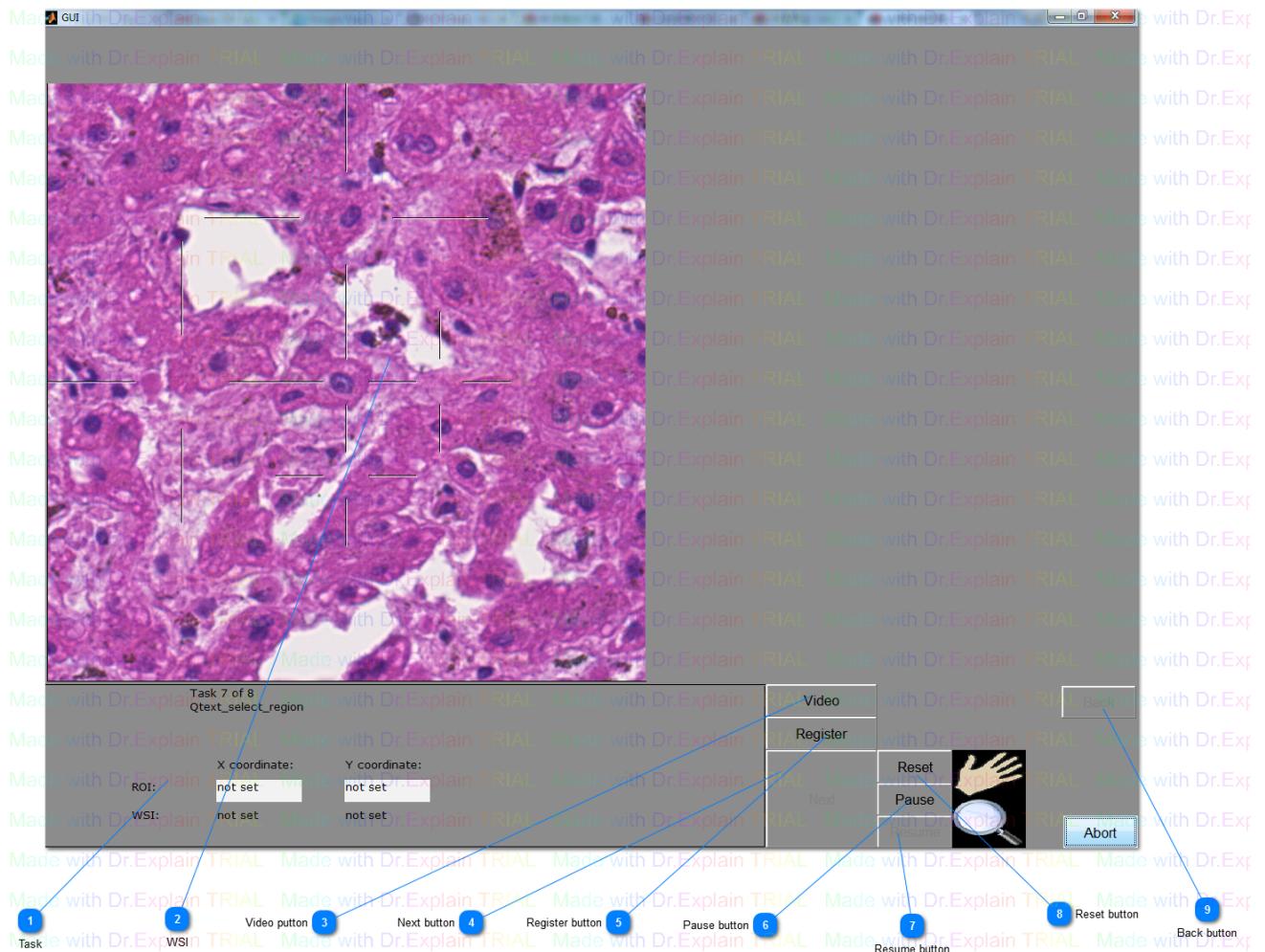
2 Start experiment

After input user name start experiment

3 Abort epxperiment

Experiment Interface

Once the input reader ID, the evaluation tasks begin when the ROI Image Presentation Screen opens. It is presented in a fashion identical to the Digital mode experiment with the addition of the Video Preview window. The Administrator is expected to read the tasks to the reader and record the responses of the Reader so that the Reader may stay engaged with the microscope. Please refer to the section on “Tasks”. The Administrator is also expected to monitor the Video Preview to confirm proper registration throughout the experiment.



1 Task

Shows the information and question for each task

2 WSI

ROI of WSI

3 Video button



show camera image, only available in MicroRT mode

4 Next button

go to next task

5 Register button



Register WSI and camare image and adjust the stage position, only available in MicroRT mode



Pause current task



Resume task from pause status



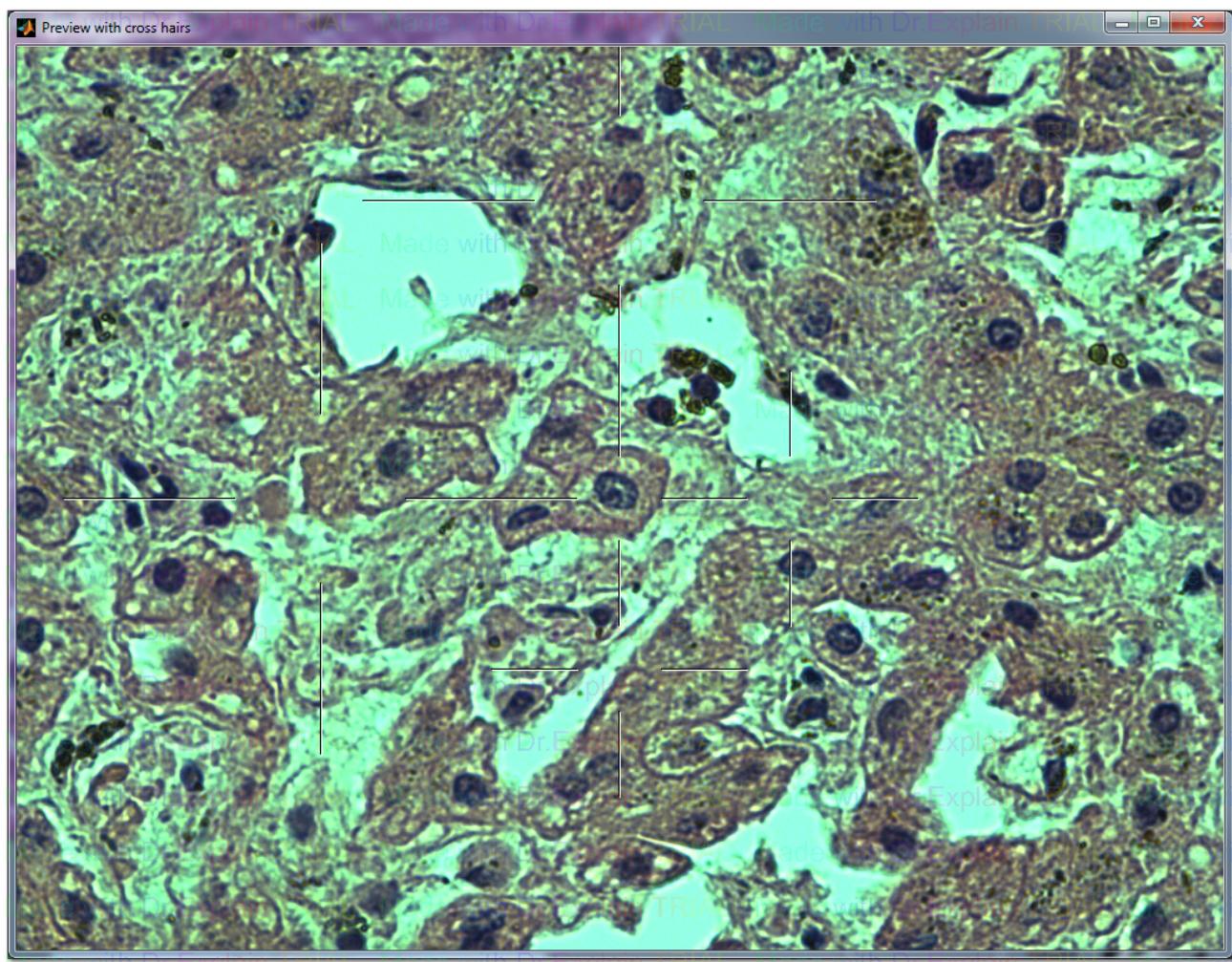
Reset all the result in this task and move stage to original position



Go back to previous task

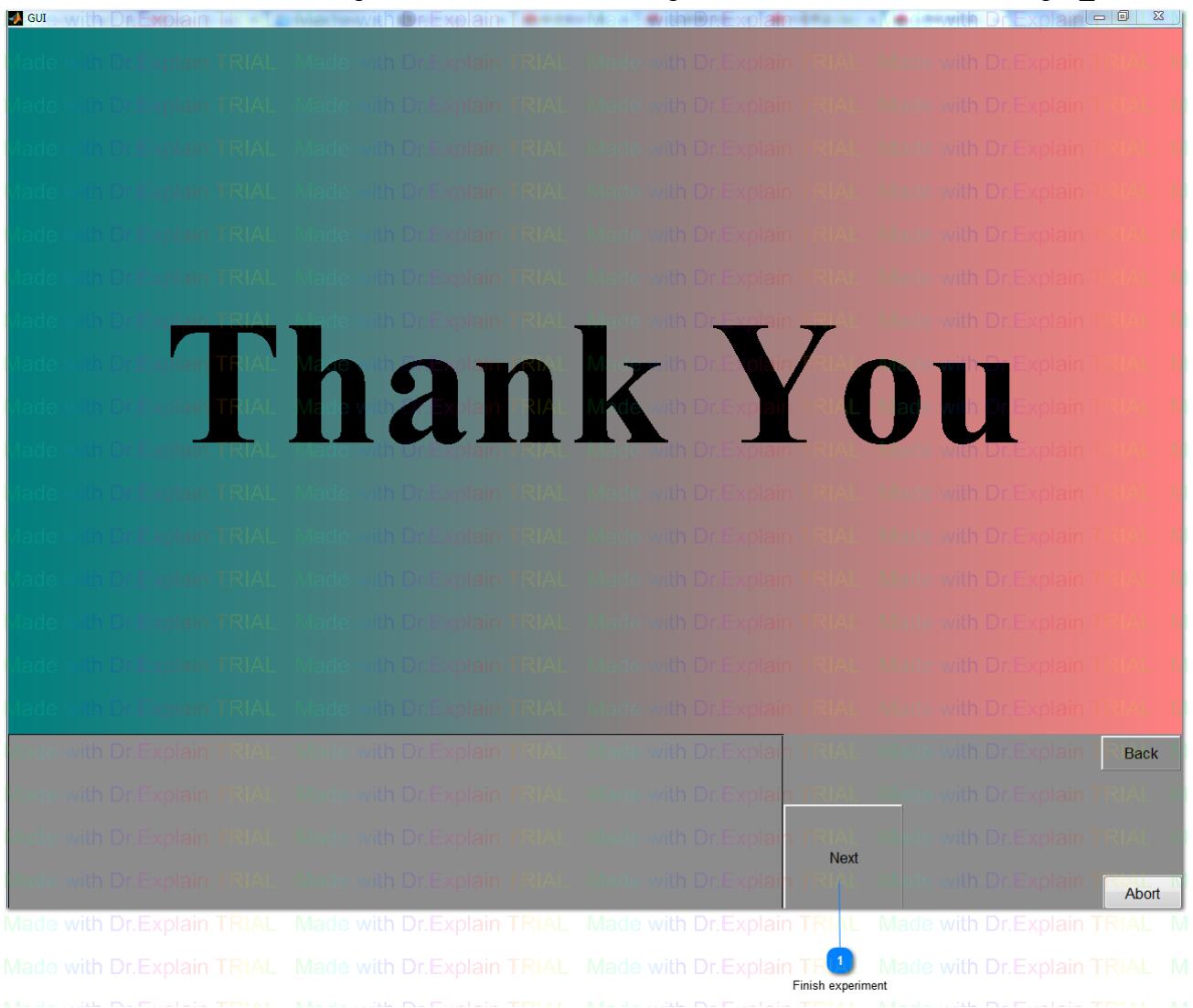
Preview with cross hairs window

In MicroRT mode, preview with cross hairs window show the camera live microscope image for administrator view and registration between WSI and camera image to move stage.



Thank You window

At the conclusion of the last task, responses are saved in the output file. Please refer to the “Output_File” section



1

Finish experiment

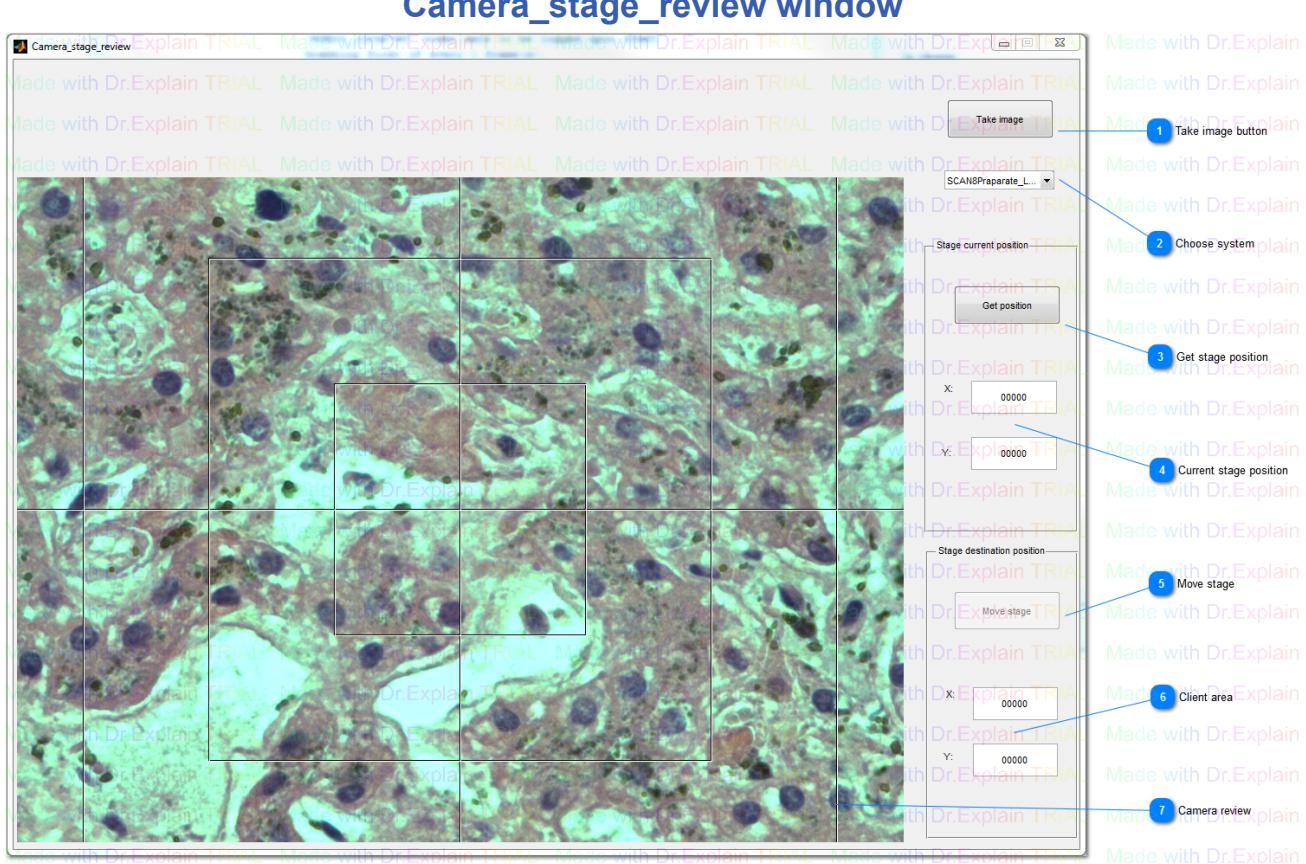
finish experiment and save result

Stage and Camera Control Utility Functions

This is an individual software. It has following features

1. Take a picture and save it to a file.
2. Get stage position coordinates.
3. Move to stage position coordinates.





1 Take image button

User could use this button to take image of the current camera review. If user chosen stage control system, the default image name has stage position information. Otherwise, the default name is in format "cam_+number". User could also edit the image name and saving directory.

2 Choose system

User could choose his working system at here. Without choose a system "Stage current position" and "Stage destination position" would be disabled. But user could still use Take image button to get camera image

3 Get stage position

Get current stage positions and display them in part 4

4 Current stage position

When user clicks "Get position", "Move stage" or "Take image" buttons, current stage position will update to here

5 Move stage

Move the stage to the position that user input in part 6. This button only enables when user inputs both X and Y position in part 6

6 Client area

Input the destination positions for stage



Camera review



Run eeDAP***

- How to run a sample eeDAP input file:

- Click the button "Click to browse for .dapsi input file."

- Navigate into the folder "sample_inputfiles"

Select one of the four input files: "sample_tissue-1.dapsi", "sample_tissue-olympus.dapsi", "ThorLabs.dapsi", and "ThorLabs-olympus.dapsi". • Click "Extract ROIs".

- This will create folders in the "sample_inputfiles" directory ("Output_Files", "Temporary_Registration_Images", and "Temporary_Task_Images")

- This will also create .tif files in the "Temporary_Task_Images" directory. One for each task listed in the .dapsi input file.

- Click "Continue". This will bring up the data collection GUI. •

- Enter your name where you see, "readerID" and click "Next". You are on your way. The text and labels for the questions are defaults and have no meaning. Hopefully you will figure out what to do. The current tasks are

1. pick a radio button
2. enter an integer/count
3. move the slider bar
4. click on the image.

III. Software Requirements and Set Up

Read and extract WSI image

There are two methods to extract ROI from WSI image "BIO formats" and "Image Scope".

BIO formats

Bio-Formats is currently being used in eeDAP.

Bio-Formats is a standalone Java library for reading and writing life sciences image file formats. It is capable of parsing both pixels and metadata for a large number of formats, as well as writing to several formats.

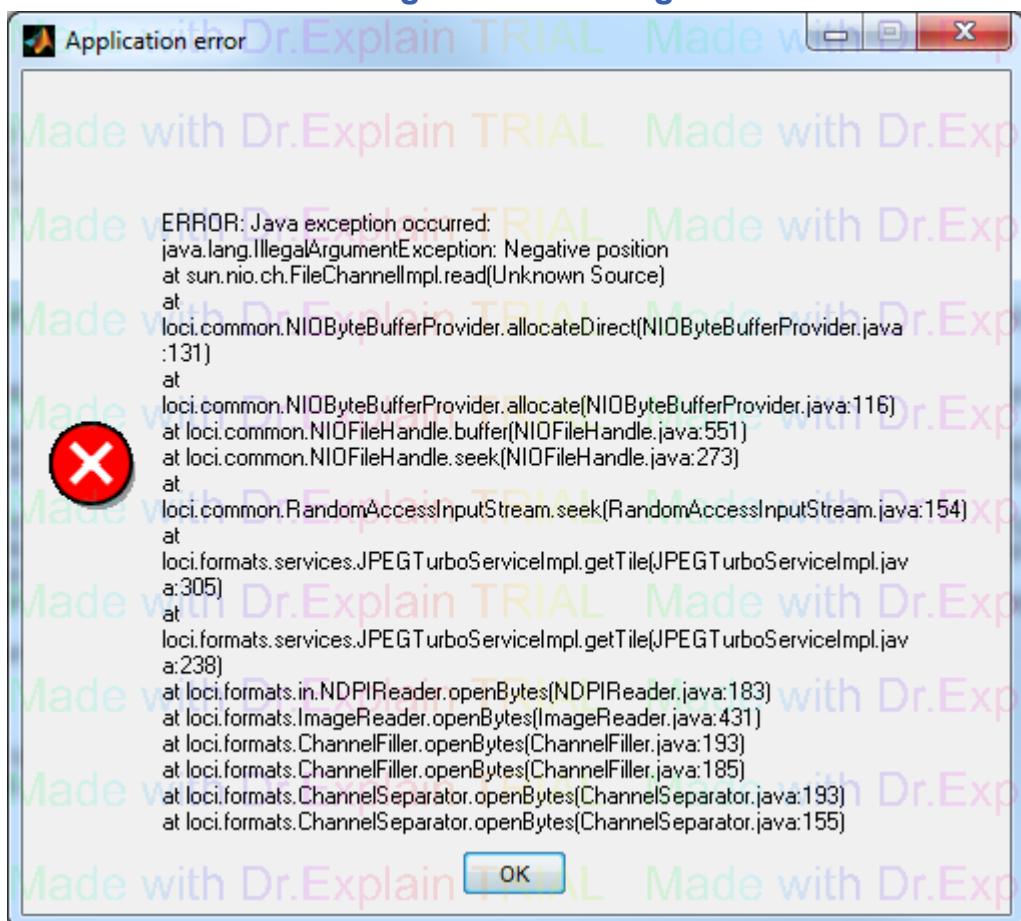
Bio-Formats is developed by the Open Microscopy Environment consortium, including development teams at LOCI at the University of Wisconsin-Madison, University of Dundee and Glencoe Software. Licensing and citing information is on the [OME licensing page](#).

The primary goal of Bio Formats is to facilitate the exchange of microscopy data between different software packages and organizations. It achieves this by converting proprietary microscopy data into an open standard called the OME data model, particularly into the OME-TIFF file format.

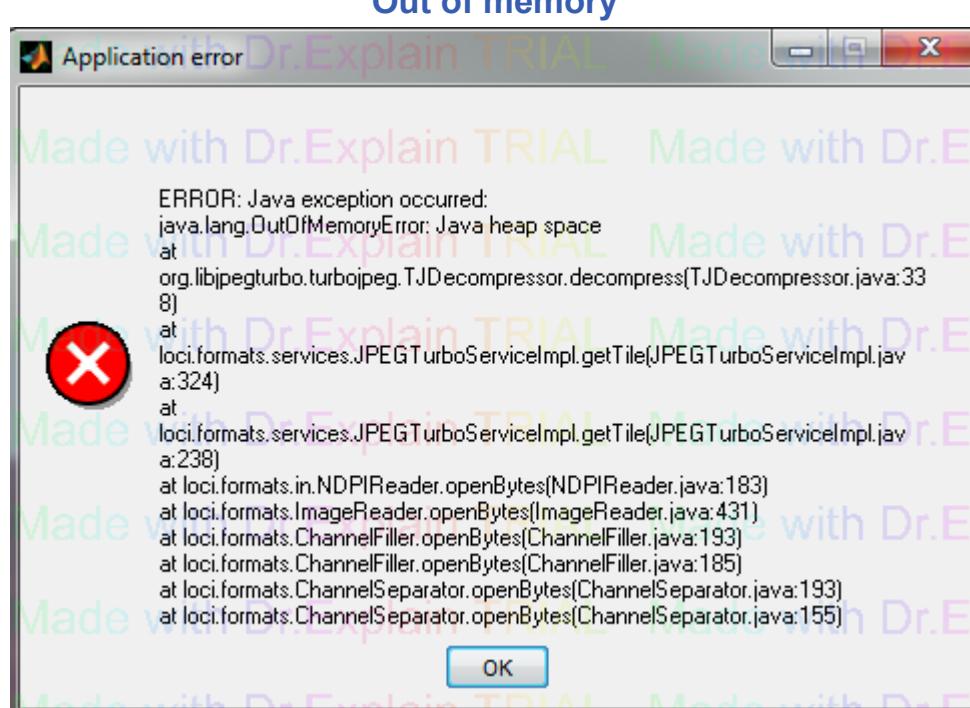
error

The BIO_formats might have some problem when extract WSI. In this section we show 2 normally errors and the solve methods



Image file is too large

This error is because the input WSI image file is too large. Now we are using 5.1.2 version Bio formats. It works for file smaller than 4 GB. If you want to use larger image, you might face this problem



When we process some large image, there might be a error about out of memroy. For large image a minimum of 512 MB is suggested.

Users could download the java.opts and paste it in the matlabroot directory, or follow this website : <http://www.mathworks.com/matlabcentral/answers/92813-how-do-i-increase-the-heap-space-for-the-java-vm-in-matlab-6-0-r12-and-later-versions>.

Image Scope by Aperio

Image Scanscope is our previous used method, which is a product of Aperio <http://www.aperio.com/healthcare/eslide/view>. This software contains an Active X controller name TIFFcomp which allows for manipulating of WSI images (panning, zooming, etc.). eeDAP uses TIFFcomp to extract the ROI's from the WSI images. The reason we choose new method is Image Scanscope could only works in 32 bit Matlab.

Matlab compiler runtime libraries

The precompiled stand-alone eeDAP application requires that the Matlab compiler runtime (MCR) library be installed. This works on Windows and Linux, but it does not require any Matlab license or libraries.

The MCR library must be the same version as was used to create the stand-alone application. The current versions of the MCR library that eeDAP is using are Windows 32-bit R2013a (8.1) and Linux 64-bit R2015a(8.5)

The installer can be found at:

- possible local folder: C:\Program Files (x86)\MATLAB\R2013a\toolbox\compiler\deploy\win
- internet: <http://www.mathworks.com/products/compiler/mcr/>
- internet: <http://medviso.com/download/mcrinstaller>

The installer will create a folder like

C:\Program Files (x86)\MATLAB\MATLAB Compiler Runtime\v81

IV. Hardware requirement and Set Up

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Stage

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Lude Stage

Now, FDA uses Lude MAC5000 and MAC6000 stage

Set up

The stage should connect with computer by RS232 serial to USB cable.

Windows 7

In windows 7 system, user could find the stage by Start -> Control Panel -> Device Manager -> Ports(COM & LPT).

The name of the Port should be something like Prolific USB-to-Serial Comm Port (COM*).

User should choose COM* in eeDAP to connect with stage.

Linux

In linux use could find the stage by command: "ls -l /dev/tty*".

The port of stage name could be "ttyUSB0". As the default setting of matlab only automatically recognizes serial port names of the form /dev/ttyS[0-255]. User should create a symbolic by command "ln -s /dev/ttyPS0 /dev/ttyS101" under root account. The S101 could be any integer port between S101 to S109. And please choose the same prot in eeDAP software. The following link provide detail information about build link <http://www.mathworks.com/matlabcentral/answers/95024-why-is-my-serial-port-not-recognized-with-matlab-on-linux-or-solaris>

If the users don't run the eeDAP under root account, users need provide permission to the account they want to use. They could log in to root account and use following command to provide permissions.

1. Serial prot permission:

"chmod 777 /dev/ttyXXX", where ttyXXX is the port name like ttyUSB0

2. Lock group permission:

- a. Command "nano /etc/group"
- b. Add linux account name to lock group
- c. logout and relogin linux account.

Test

When the user selects the mode to be "MicroRT", Microscope Real Time, the program will force the user to set the communication port for the stage. The stage must be plugged into one of the communication ports, and the correct communications port must be selected.

Camera

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Requirements

The code is written to work with a DCAM compatible camera.

eeDAP requires camera images to be RGB24 (8 bit for each channel).

eeDAP requires camera images to have width > 640 and height > 480.

FDA uses 2 Point Grey cameras for this eeDAP software

Point Grey Grasshopper color (GRAS-03K2C-C)

The default Matlab format code is RGB24_640x480, which has an aspect ratio of 1.333. The pixel size and format of the default format equals that of the native sensor specs.

- sensor size = 1/3"
- sensor size = 640x480 pixels (0.3MP)
- pixel size = 7.4um

When attached to the microscope, the scale factors are equal to the pixel size divided by the camera adapter magnification (if any) and the objective magnification. When there is no camera magnification:

- 0.185um/pixel at 40x
- 0.370um/pixel at 20x
- 0.740um/pixel at 10x
- 1.480um/pixel at 5x
- 1.850um/pixel at 4x
- 2.960um/pixel at 2.5x

Point Grey Flea2 color (FL2G-50S5C-C)

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Native pixel format

The Matlab format code that uses all of the native pixels is F7_RGB24_2448x2048. It is a non-standard format (Format_7) with aspect ratio 1.953 and the following specs:

- sensor size = 2/3"
- sensor size = 2448x2048 pixels (5.0MP)
- pixel size = 3.45um

Default format

The default Matlab format code is RGB24_1024x768, which has a standard aspect ratio of 1.333. The default pixels are 2x2 bins of the native pixels. Consequently, the pixel size is 6.9um.

When attached to the microscope, the scale factors are equal to the pixel size divided by the camera adapter magnification (if any) and the objective magnification.

When there is no camera magnification, the pixel size of the default Matlab format (6.9um) is divided by the objective magnification.

- 0.1725um/pixel at 40x
- 0.3450um/pixel at 20x
- 0.6900um/pixel at 10x
- 1.3800um/pixel at 5x
- 1.7250um/pixel at 5x
- 2.7600um/pixel at 2.5x
- 3.4500um/pizel at 2x

Alternative format

The Matlab format code F7_RGB24_1600x1200 uses the native pixels but a smaller standard format.

Setup

For Point Grey camera, users need install special diver to run it under eeDAP software.

Windows 7

In windows 7, after plugging in the camera, disable the default Windows driver and update the driver for the attached device to the industry standard DCAM compatible firewire (IEEE 1394 interface) driver. Carnegie Melon (CMU) has a general DCAM driver that may work (1394camera646.exe = version 6.46). It can be downloaded at <http://www.cs.cmu.edu/~iwan/1394/index.html>

We found out about the CMU driver at

<http://www.mathworks.com/products/imaq/supportedio.html>

1. Click on device manufacturer.

2. Click on industry standard DCAM IEEE 1394

There is also a demo application at the CMU web site (1394CameraDemo32.exe). It behaves similarly to the Matlab "imaqtool" mentioned above. I think it gets installed when the CMU driver is installed, location C://Program Files (x86)\CMU\1394Camera\bin. It is used to test the camera and different operating modes.

Linux

In linux, users should install two packages libdc1394-22 and libraw1394-11, as mentioned in the documentation link below, <http://www.mathworks.com/help/imaq/linux-dcam-ieee-1394-hardware.html>

If the users don't run the eeDAP under root account, users need provide permission to the account they want to use. They could log in to root account and use following command to provide permissions.

- a. Command "nano /etc/group"
- b. Add linux account name to video group
- c. logout and relogin linux account.

Test

When the user selects the mode to be "MicroRT", Microscope Real Time, there is a button that launches a Matlab utility called "imaqtool". This utility is an interactive GUI to allow you to explore, configure, and acquire data from your installed and supported image acquisition devices. If this doesn't work, Matlab can't find your camera.

Whenever you plug-and-play a camera, Windows enables the default Windows driver, which may actually be the camera manufacturer's driver. If the default Windows driver isn't DCAM compatible, Matlab can't find your camera.

Alignment

<TODO>: Need one more picture

Depending on hardware, you may have the ability to completely align the camera with the eyepiece by rotating and shifting the camera. This process is made possible by loosening and tightening one-three screws on the camera mount. This can be done at any time during the registration process. The registration should persist over time and need little adjustments unless the screws are ineffective or the system is bumped. A calibration slide, reticle, and virtual reticle are very useful for this alignment. A good camera mount also has a focus screw that can be adjusted so that the camera and the eyepiece are nearly in focus at the same time. User could use gird slide to do the alignment.

1. Use highest magnification optics
2. Look through the eyepiece and focus the microscope.
3. Move stage to let the reticle in eyepiece just touch the boundary of the gird corner.
4. Don't move the stage and focus camera image.
5. loose the screws around camera.
6. Use hand to adjust the camera to make the reticle in camera image just touch the same boundary of the gird corner.
7. Hold the camera and tight the screws.
8. look the camera image and keep adjustment during tightening

Note: 1. Small shift could be solved by eyepiece and camera register processing.

2. Large shift might influenct the fast registration during task. Because we fast registration use a small center aera of camera image, lager shift between eyepiece and camera will lead outstanding feature goes out of the camera registration area. Best registration is another method to slove the large shift problem.

Scanners

The scale factors of the scanner were measured by Neil O Flaherty
The scale factors can also be found embedded in the image files.
Currently, we use 4 difference scaners.

Hamamatsu Nanozoomer 2.0HT (at NIH ATC)

- scan_scale at 20x = .4558um/pixel
- scan_scale at 40x = .2279um/pixel

Aperio CS (at NIH ATC and FDA White Oak)

scan_scale at 20x = 0.50um/pixel

scan_scale at 40x = 0.25um/pixel



Aperio T2 (at NIH ATC)

We believe the specs are the same as Aperio CS.



Aperio Scanscope XT

We believe the specs are the same as Aperio CS.

Microscopes

The page that opened was Field of View Diameter at microscopyU. <http://www.microscopyu.com/tutorials/java/fielddiameter/index.html>. The first sentence is, "The diameter of the field in an optical microscope is termed the field number and represents the diameter of the field measured in millimeters at the intermediate image plane."

Microscopy from the very beginning - Carl Zeiss, Inc. <http://www.microscopy-news.com/news/carl-zeiss-microscopy-from-the-very-beginning.html>

On page 36 it says, "Eyepieces (or oculars, from the Latin "oculus" = the eye) are the magnifiers with which you view the intermediate image in the microscope, produced by the objective and the tube lens. In the Axiolab microscope, the intermediate image has a useful diameter of 20 mm. Eyepieces are not just simple lenses, but are corrected optical systems consisting of several lenses. It would be a pity if the intermediate image produced with such sophisticated optics were to be impaired just before it reaches the eye.

Normally, the additional magnification provided by the eyepiece is 10x. The intermediate image in this example then has a diameter of 20 cm at a reading distance of 25 cm to the eye. A comparison: this diameter is about as large as the width of this page."

Axioplan 2 Imaging microscope with an Axiophot 2 head

- Field Number (FN) of the oculars (eyepieces) = 23mm
- Magnification of eyepiece (mag_e) = 10x
- Magnification of objective (mag_o) = 2.5x, 5x, 10x, 20x, 40x
- FOV at Mag_o = FNmm/mag_o
- FOV at 2.5x = 23mm/2.25 = 9.200mm
- FOV at 5x = 23mm/5 = 4.600mm
- FOV at 10x = 23mm/10 = 2.300mm
- FOV at 20x = 23mm/20 = 1.150mm
- FOV at 40x = 23mm/40 = 0.575mm
- Apparent diameter of microscope image "at 25cm" = FN*mag_e = 23cm
- Diameter for ocular micrometer (reticle) = 26mm

Olympus BX43

- Field Number (FN) of the oculars (eyepieces) = 22mm
- Magnification of eyepiece (mag_e) = 10x
- Magnification of objective (mag_o) = 2x, 4x, 10x, 20x, 40x
- FOV at Mag_o = FNmm/mag_o
- FOV at 2x = 22mm/2 = 11.00mm
- FOV at 4x = 22mm/4 = 5.50mm
- FOV at 10x = 22mm/10 = 2.20mm
- FOV at 20x = 22mm/20 = 1.10mm
- FOV at 40x = 22mm/40 = 0.55mm
- Apparent diameter of microscope image "at 25cm" = FN*mag_e = 22cm
- Diameter for ocular micrometer (reticle) = 24mm

Reticles

Reticle is placed in eyepiece at Intermediate Ocular Plane (IOP). It could help use define the position of the tissue under microscopes

Klarmann Rulings KR-429

width and length of 10x10 grid at IOP = 12.5mm

apparent width and length of 10x10 grid "at 25cm" = 12.5cm

grid spacing at IOP = 1.25mm

grid spacing at stage at mag_o

$$5x = 1.25\text{mm}/5 = 0.25000\text{mm}$$

$$10x = 1.25\text{mm}/10 = 0.12500\text{mm}$$

$$20x = 1.25\text{mm}/20 = 0.06250\text{mm}$$

$$40x = 1.25\text{mm}/40 = 0.03125\text{mm}$$

apparent grid spacing "at 25cm" = 12.5mm

IMAGE

Klarmann Rulings KR-32536

This is a custom reticle with cross-hair-like fiducials pointing to gaps: 2 are 1mm x 1mm and 3 are 0.5mm x 0.5mm.

[IMAGE](#)



V. FDA hardware specifications

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Cameras

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Displays

HP L2335

Size: 49.6cm x 31.1cm

Size (pixels): 1920 x 1200

Pixel Pitch = 258um

Contrast: 500:1

Brightness: 250 cd/m²

Dell 1908 FPt

Size: 37.6cm x 30.1cm

Size (pixels): 1280 x 1024

Pixel Pitch = 294 um

Contrast = 800:1

Brightness = 300 cd/m²

Lenovo Thinkvision L220XWC

Size: 47.4cm x 29.6cm

Size (pixels): 1920x1200

Pixel Pitch: 247 um

Contrast = 1200:1

Brightness = 325 cd/m²

VI. Program Variables

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Assigned in the Input File

NUMBER_OF_WSI (int)

The number of wsi used in the study.

wsi_slot_n

directory and file name of n'th WSI image.

rgb_lut_slot

directory and file name of WIS image RGB look up table

label_pos

direction of glass slide

reticleID (string)

The model number of the reticle.

cam_format (string)

The Matlab format code for the camera, e.g. "RGB24_1024x768"

cam_pixel_size* (float)

The size of the camera pixels [um]. This depends on possible (de)magnification of the mounting adapter.

The camera pixel size may not be equal to that given in the camera specifications. The camera format may lead to binning which changes the camera pixel size.

mag_cam (float)

Magnification of the camera and to be used in the study.

mag_lres (float)

The low resolution magnification applied to the camera to be used for registration.

cam_hres_mag (float)

The high resolution magnification applied to the camrea to be used for registration and study.

scan_scale (float)

The width in specimen units that a scanner pixel represents [um/scan_pixel].

BackgroundColor_R (float)

The red value of the background color given as a fraction between zero and one.

BackgroundColor_G (float)

The green value of the background color given as a fraction between zero and one.

BackgroundColor_B (float)

The blue value of the background color given as a fraction between zero and one.

AxesBackgroundColor_* (RGB = float, float, float)

The red value of the background color of the axes area given as a fraction between zero and one.

ForegroundColor_* (RGB = float, float, float)

The red value of the foreground color given as a fraction between zero and one.

FontSize (int)

The font size.

taskorder (int)

The place in the order that each task will be executed. (See below).

—
—

myData.settings

<TODO>: Insert description text here... And don't forget to add keyword for this topic

List of Variables Assigned in the Input File

cam_w

int [determined from camera]
Width [pixels] of the camera image.

cam_h

int [determined from camera]
Height [pixels] of the camera image.

cam_roi_w

int [hardcoded]
used to determine width of camera patch for registration

cam_roi_h

int [hardcoded]
used to determine height of camera patch for registration

cam_scale_lres

float [derived]
The width in specimen units that a camera pixel represents at low magnification [um/cam_pixel].

cam_scale_hres

float [derived]
The width in specimen units that a camera pixel represents at high magnification [um/cam_pixel].

cam2scan_lres

float [derived]
The conversion factor to convert the width of a camera pixel at low mag to the width of a scanner pixel [scan_pixel/cam_pixel].

cam2scan_hres

float [derived]
The conversion factor to convert the width of a camera pixel at high mag to the width of a scanner pixel [scan_pixel/cam_pixel].

eye_cam_offset

(int, int) [determined]
This variable determines the offset caused by misalignment between the camera and the eyepiece. It is set in Administrator_Input_Function -> align_eye_cam.

scan2cam_lres

float [derived]
1.0/cam2scan_lres [cam_pixel/scan_pixel]

scan2cam_hres

float [derived]
1.0/cam2scan_hres [cam_pixel/scan_pixel]

Taskorder defines the place in the order that each task will be executed. After a study is executed, the place in the order that each task is executed will be saved in the field myData.tasks.order.

- 0: The order of tasks is randomized. The field myData.tasks.order is ignored on input.
- 1: The order of tasks follows the listed order. The field myData.tasks.order is ignored on input.

- 2: The order of tasks follows the order given by the field myData.tasks.order.

The field of view of the microscope (diameter of the field at the specimen) equals the field number divided by the magnification of the objective.

The apparent field of view of the microscope (diameter of the intermediate image assumed to be viewed at 25cm) equals the field number times the magnification of the eyepiece.

myData.tasks_in, myData.tasks_out

myData.tasks_in holds the tasks in the same order as they appear in the inputfile.
 myData.tasks_out holds the tasks in the order determined by myData.settings.taskorder

Field	Type	Desc	myData.tasks_in, myData.tasks_out
ID	string	String given to the task.	
order	int	If taskorder==2, then this field specifies the place in the order this task will be executed. Otherwise this field is ignored.	
slot	int	The slot number identifies the wsi file and image for the task.	
roi_x	int	The horizontal center of the ROI to be extracted from the wsi file.	
roi_y	int	The vertical center of the ROI to be extracted from the wsi file.	
roi_w	int	The width of the ROI to be extracted from the wsi file.	
roi_h	int	The height of the ROI to be extracted from the wsi file.	
img_w	int	The initial width of the displayed ROI. (Rotated with respect to wsi file s.t. it corresponds to the extracted ROI).	
img_h	int	The initial height of the displayed ROI. (Rotated with respect to the wsi file s.t. it corresponds to the extracted ROI).	
Qtype	string	Label identifying the task (See <K1.1/>).	
Qtext	string	The text displayed to the user providing instructions to complete the task.	
MoveFlag	int	Flag indicating whether or not moving is allowed (MoveFlag=1) or not (MoveFlag=0).	
ZoomFlag	int	Flag indicating whether or not zooming is allowed (ZoomFlag=1) or not (ZoomFlag=0).	
Op1	string	Label for task option #1 (See <K1.2/>).	
Op2	string	Label for task option #2 (See <K1.3/>).	
Op3	string	Label for task option #3 (See <K1.4/>).	
Op4	string	Label for task option #4 (See <K1.5/>).	

<K1.1 ilk="SPECIALNAME" >Sect-Qtype</K1.1>

<K1.2 ilk="SPECIALNAME" >Sect-Qtype</K1.2>

<K1.3 ilk="SPECIALNAME" >Sect-Qtype</K1.3>

<K1.4 ilk="SPECIALNAME" >Sect-Qtype</K1.4>

<K1.5 ilk="SPECIALNAME" >Sect-Qtype</K1.5>

myData.wsi_files

myData.wsi_files		
Field	Type	Desc
fullname	string	name of wsi file including full path
wsi_w	int	width in pixels of wsi file
wsi_h	int	height in pixels of wsi file

Qtype: Question types

<TODO>: Insert description text here... And don't forget to add keyword for this topic

VII. eeDAP Developers

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Creating the stand-alone application

<TODO>: Insert description text here... And don't forget to add keyword for this topic

Deployment project

Start a new "Deployment Project" named eeDAP. Matlab will create a project file and folder. The location for this project is not important. This content is not archived because it duplicates content already in the archive.

- 2013: Go to the "APPS" tab on the Matlab GUI.
- -OR-
- 2012: Go to File->New -> Deployment Project.

Build the stand-alone application

Move the main file into the build (Administrator_Input_Screen.m). Also add the "tasks" folder to the build. Click on the "Build" button. The build takes as much as 5 minutes! The build creates the "src" folder underneath the project folder.

Presumably, the build doesn't recognize the task functions as called from the main program. This might be because they are called by a function handle and not by name.

Package the stand-alone application

Add related files and folders to the package and click on the "Package" button. This creates the "distrib" folder underneath the project folder. Packaging also creates the package to be unwrapped by the client (eeDAP-1.0_pkg.exe). This package is what you share with the client. The related folders (and their contents) we are packing in the stand-alone package are:

- gui_graphics
- icc_profiles
- sample_inputfiles (first deleting backup files and temporary files and temporary folders)
- Maybe docs some day and this user manual.

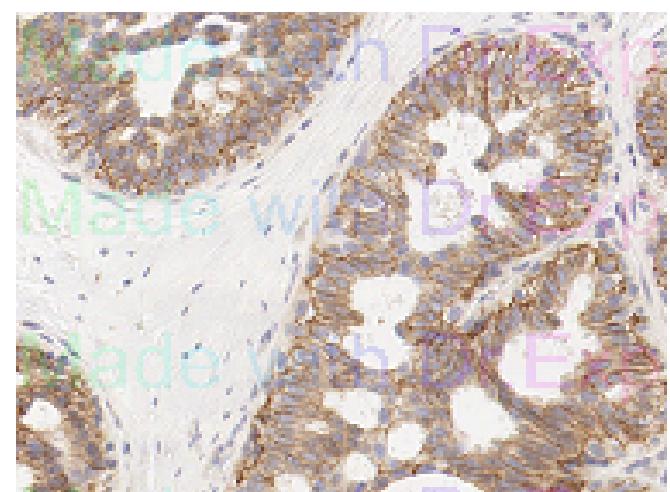
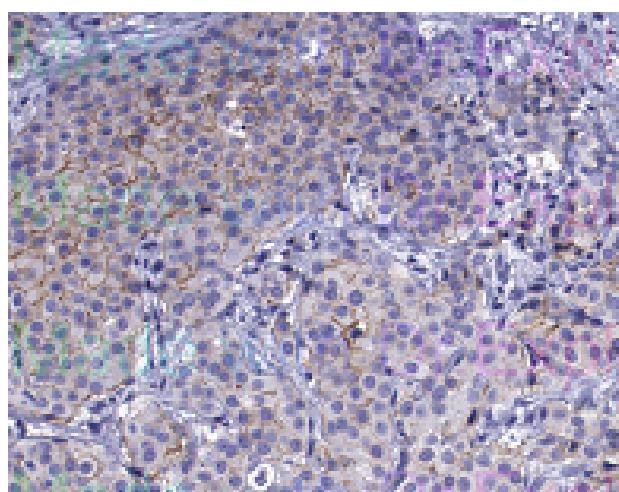
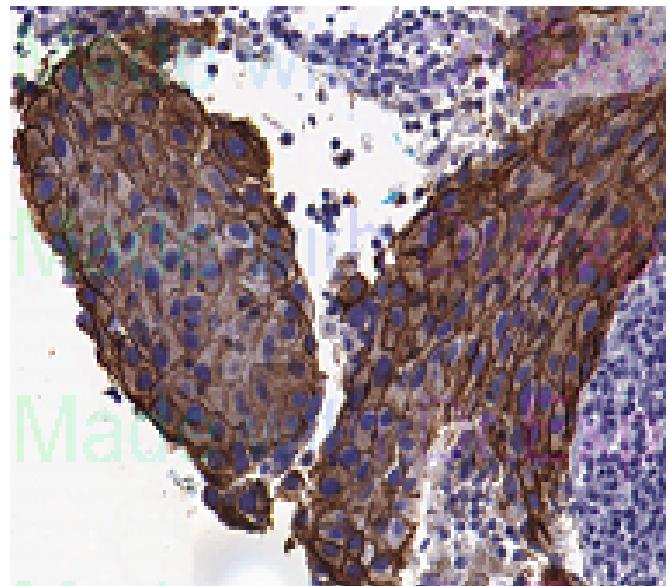
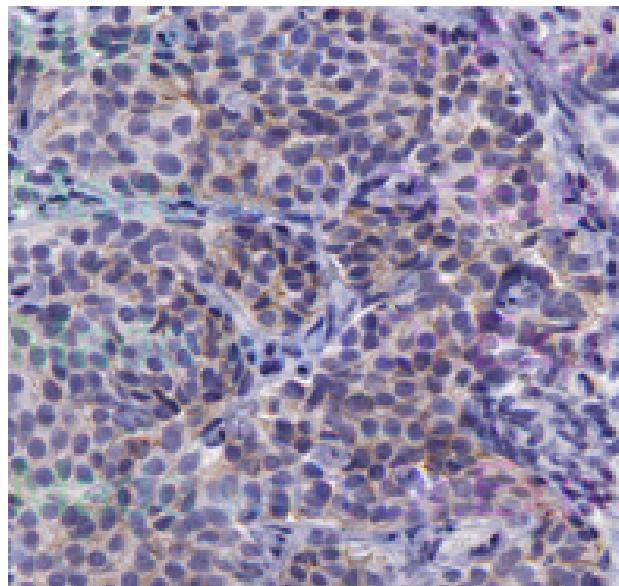
VIII. MicroRT Registration

Before data collection in MicroRT mode, the glass slides must be registered to the corresponding WSIs. After registration, the ROIs viewed through the microscope eyepiece in MicroRT mode will be the same as those extracted from the WSI in Digital mode.

Semi-Automated Registration

Verify the glass slide's label corresponds to the WSI presented in that slot of the motorized stage, also verify that the video preview window has been launched and len.

Use the joystick to control the microscope stage. Navigate to a non-homogenous area of the slide, such as the end of the tissue, or a unique feature in the tissue. Good and Bad features are displayed below.

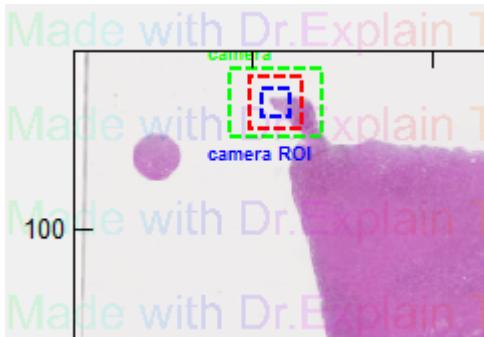


Low Resolution Registration

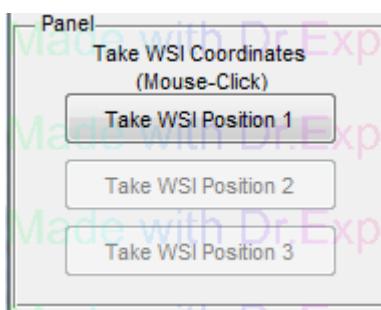
1. After find desire area, press "Take Stage Position 1"



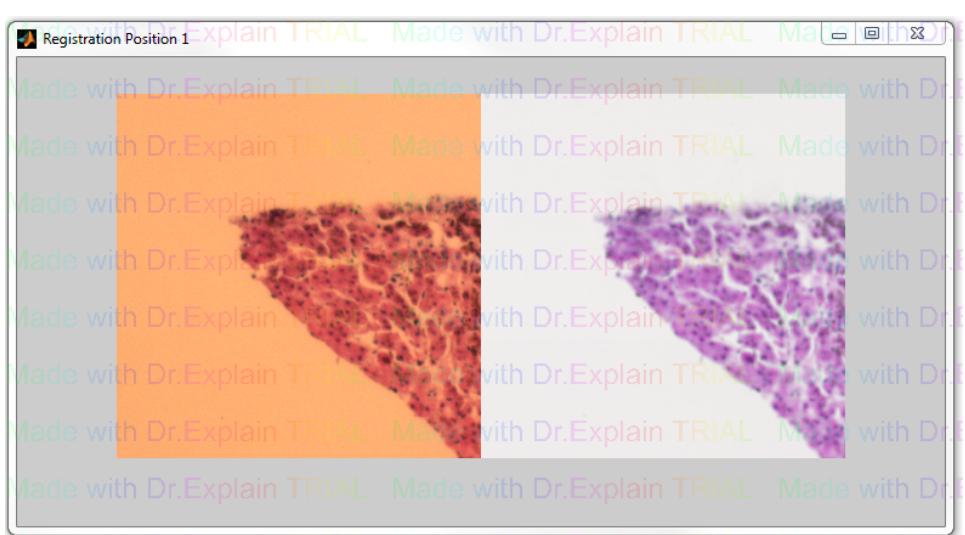
2. Click the related position in the WSI thumb image



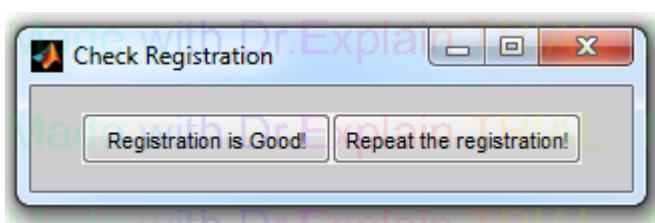
3. Press "Take WSI Position" to start registration



4. Check the registration result in Registration Position 1 window,



5. Decide whether satisfied the registration. If yes press "Registration is Good! to continue". If not, press "Repeat the registration" to redo registration for this position

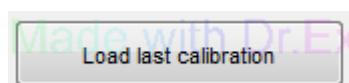


6. Repeat process twice for 2 other positions

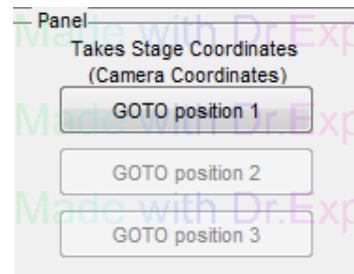
High Resolution Registration

1.change len to high resolution

2.After finish 3 positions low resolution registration. Press "Load last calibration" to get into high resolution registration, we can also use this button to skip low resolution registration, If we do experiment for previous input file with glass slide put at same position on the stage.



3.Press "GOTO position 1" to move stage to first low resolution registration position



4.Similar as low resolution registration part, click WSI image for related area for high resolution registration.

5 Reapeat steps 3, 4 twice for 2 other position

Note: User could use "Finished Registration" button to skip High Resolution Registration



Manual Registration

When automated registration fails for a slide, manual registration process may be necessary. This method requires the Administrator to identify three registration points and find them in the WSI and the glass slide. Remember to maximize the distance between the registration points and select points that have unique features in non-homogenous areas of the WSI. The Administrator needs to input the corresponding pairs of x,y coordinates in the Registration window: the x,y coordinates in the WSI and the x,y coordinates on the glass slide.

The screenshot shows a software interface for manual registration. It consists of three vertically stacked panels, each labeled "Panel" at the top. Each panel contains two sets of coordinates: "WSI Position" and "Stage Position". The first panel shows WSI Position 1 with X: 3391.1 and Y: 79729, and Stage Position 1 with X: 35272 and Y: 10262. The second panel shows WSI Position 2 with X: 6321.5 and Y: 10172, and Stage Position 2 with X: 51153 and Y: 10735. The third panel shows WSI Position 3 with X: 11333 and Y: 10353, and Stage Position 3 with X: 51414 and Y: 35133. All coordinates are displayed in text boxes.

Panel	WSI Position	Stage Position
Panel 1	X: 3391.1 Y: 79729	X: 35272 Y: 10262
Panel 2	X: 6321.5 Y: 10172	X: 51153 Y: 10735
Panel 3	X: 11333 Y: 10353	X: 51414 Y: 35133

To determine an x,y coordinate in the WSI,

- First, open the WSI file in Aperio's

ImageScope

- Select the “extract region”  tool in ImageScope

- Pan over the image and note that the current x,y coordinates of the cursor are displayed at the bottom of the window.

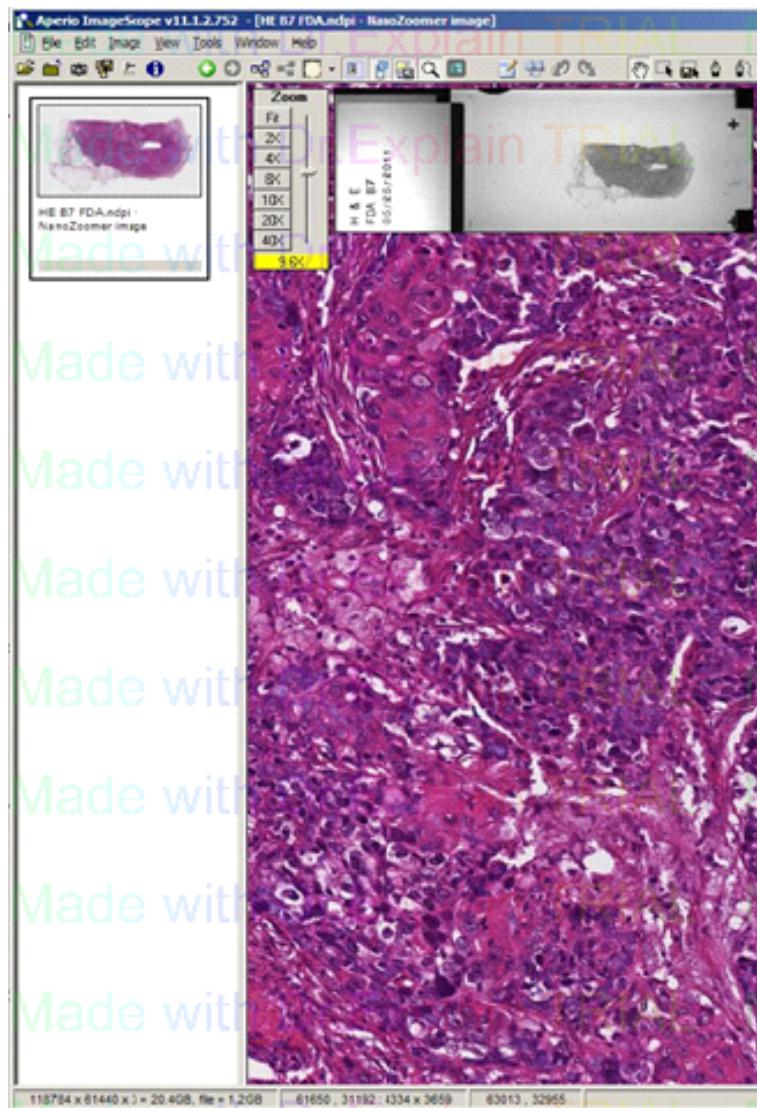
- Identify a registration point and record the x,y coordinate in the Registration window.

To determine the corresponding x,y coordinate on the glass slide.

- Using the joystick to control the microscope's motorized stage, locate the same point selected in the WSI. It is useful to have a reticle with cross hairs at the center.

- If you do not have a reticle with cross hairs, you can use the Video Preview image to center the registration point.

- Press “Take position 1”, this fills in the glass slide coordinates.



Once the x,y coordinates have been recorded for a registration points, press “Register ROI”. This may take a few minutes to execute. Repeat this process twice more locating and registering Points 2 and 3. Repeat this procedure for all slides loaded into slots 1-8 and hit “done” button.

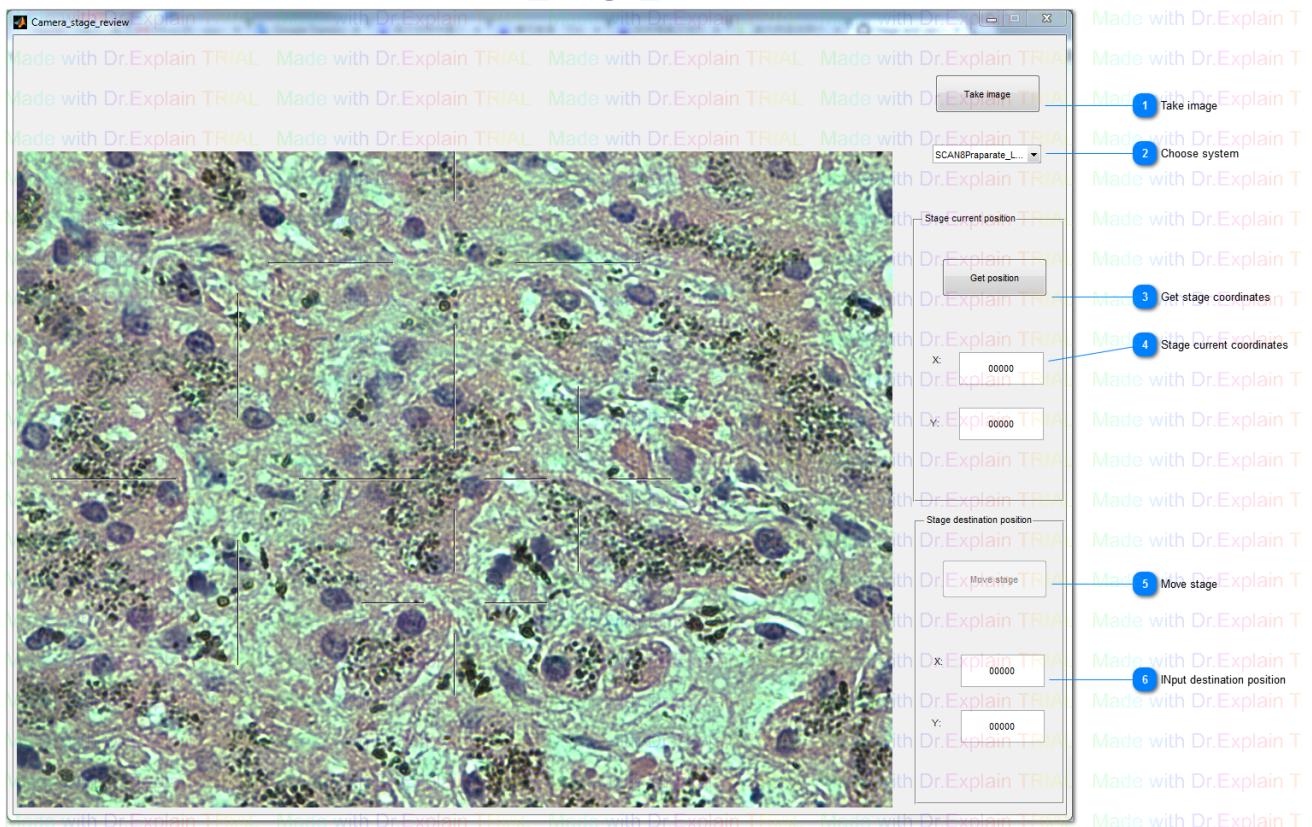
IX. camera and stage review application

There is another stand-alone application " Camera_stage_review.exe", which could run the test and control camera and microscope stage outside eeDAP file.

The application has following functions

- Take a picture and save it to a file.
- Get stage coordinates.
- Move to stage coordinates.

Camera_stage_review window



1 Take image

Take image from camera and save it to camera_images folder. If user choose the stage system, the image will named by stage coordinates. Otherwise, it will be named by number from 1.

2 Choose system

Choose system is used. Different system has different stage speed. The "Get stage coordinates" and "Move stage" functions are only enable after user choose system.

3 Get stage coordinates

Press to get stage current coordinates.

4 Stage current coordinates

X:	00000
Y:	00000

Displace stage current coordinates. It will update by press "Take image", "Get position" and "Move stage" button.

5 Move stage

Move stage to defined coordinates. The button is only enable when user input both x and y coordinates.

6

INput destination position

X:
00000

Made with Dr
Y:
00000

Made with Dr

Input stage destination coordinates.

X. Input File

All Input files for this program have the extension .dapsi. Two sample files have been provided for instructional purposes. They can be located under the “eeDAP→Sample_Input_Files”. In the following, we use Phantom_Test.dapsi to illustrate our discussion.

An input file has two sections: Header and Body. The Header includes free text that describes the experiment and formatted text that specifies global variables. The Body is a list of ROIs and corresponding tasks.

Input File Header

tissue40x-8B-axioplan2.dapi - Notepad

File Edit Format View Help

Author: Made with Dr.Explain TRIAL Date: Made with Dr.Explain TRIAL Time: Made with Dr.Explain TRIAL

For the Hamamatsu scanning at 20x scan_scale = .4558
For the Hamamatsu scanning at 40x scan_scale = .2279

Point Grey Grasshopper Color (GRAS-03K2C-C)
cam_format = RGB24_640x480
cam_pixel_size = 7.4um

Point Grey Flea2 Color (FL2G-505C-C): Full resolution, full format
cam_format = F7_RGB24_2448x2048
cam_pixel_size = 3.45

Point Grey Flea2 Color (FL2G-505C-C): Standard format, aspect ratio = 1.33
cam_format = RGB24_1024x768
cam_pixel_size = 6.9

taskorder=2 user specified order
taskorder=1 list order
taskorder=0 random order

saveimages = 1 save WSI and camera images
saveimages = 2 only save WSI image
saveimages = 3 only save camera image
saveimages = 4 don't save WSI and camera images

Input File Task Formats
task_checknof4, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, QText, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_count, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, QText, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_mark1, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, QText, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_mark1_out, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_kidroi0of4, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_kidSlider, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_mitotic_expert, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, MoveFlag, ZoomFlag, Q_Op1, Q_Op2, Q_Op3, Q_Op4
task_mitotic_train, TaskID, Taskorder, \$slot, ROI_X, ROI_Y, ROI_W, ROI_H, IMG_W, IMG_H, MoveFlag, ZoomFlag, ExpertCheckBoxes1-8, Expert_Score
task_mitotic_counts, TaskID, Taskorder, \$slot, MoveFlag, ZoomFlag

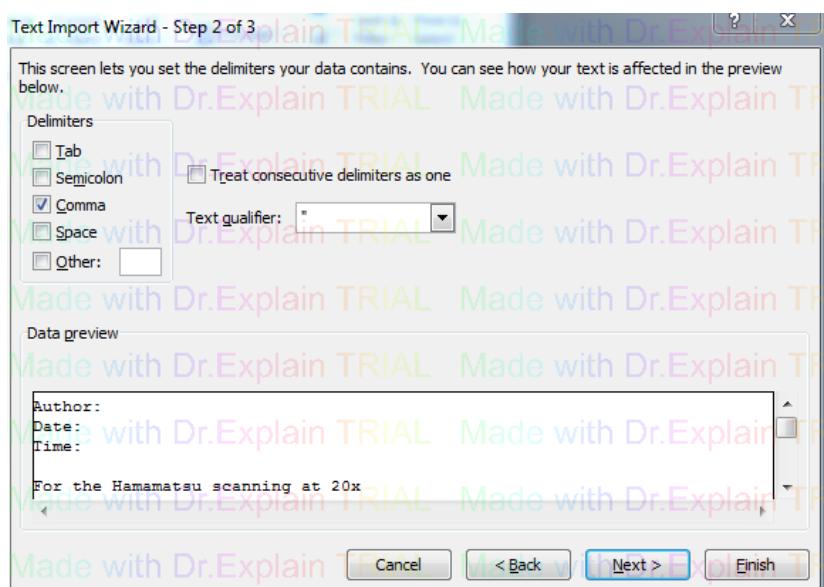
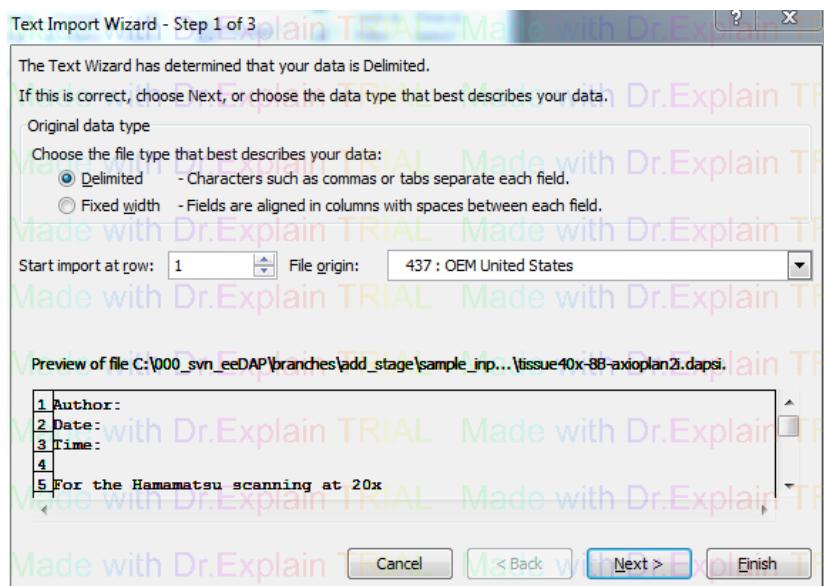
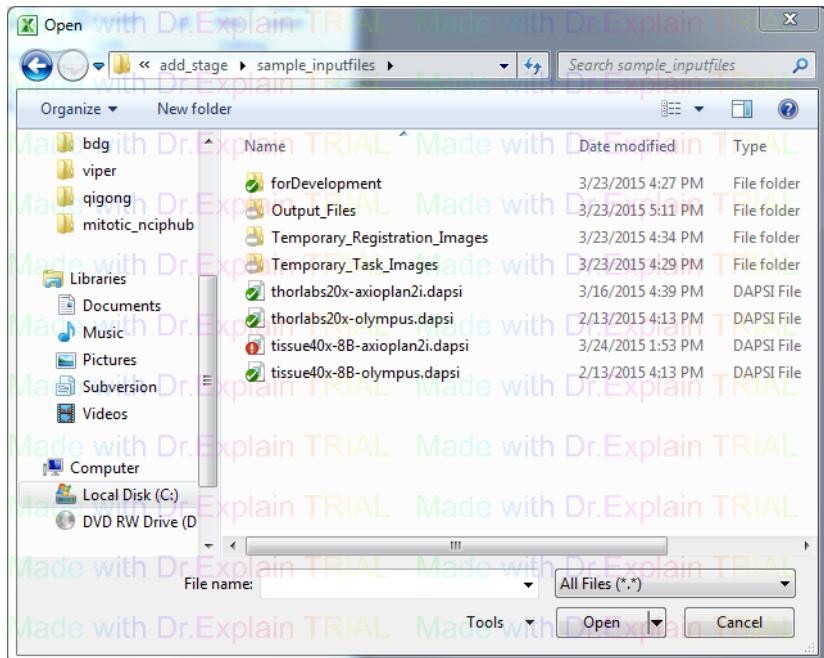
SETTINGS
NUMBER_OF_WSI = 1
ws1_slot_1 = C:\000_whole_slides\tissue40x-8B.ndpi
rgb_lut_slot_1 = 1cc_profiles\rgb_lut_gamma_inv1p8.txt
label_pos = 6
rec1_pos = KR-32536
cam_format = RGB24_1024x768
cam_pixel_size = 6.9
mag_cam = 1.0
mag_lres = 5
mag_hres = 40
scan_scale = 0.2279
stage_scale = 0.1
BG_Color_RGB = 0.55 = 0.55 = 0.55
Color_RGB = 0.00 = 0.00 = 0.00
AxesBG_Color_RGB = 0.10 = 0.20 = 0.10
FontSize = 13
SaveImages = 3
Taskorder = 0

BODY
start
finish
radio1nf4,1st0001,-1,-1,038459,16192,300,300,300,qtext_radio1of1,1,1,Q_Op1,Q_Op2,Q_Op3,Q_Op4
count,2nd0001,-1,1,027728,11381,300,300,300,qtext_count,1,1,description
slider,3rd0001,-1,1,19220,49879,300,300,300,qtext_slider,1,1,Q_Op1,Q_Op2,Q_Op3,Q_Op4
mark1,4th0001,-1,-1,041163,33208,300,300,300,qtext_select_region,1,1,Q_Op1,Q_Op2,Q_Op3,Q_Op4
mark1_out,5th0001,-1,-1,041163,33208,700,700,700,qtext_select_region,1,1,Q_Op1,Q_Op2,Q_Op3,Q_Op4
mitotic_train,6th0001,-1,-1,041163,33208,700,700,700,700,1,1,0,1,0,1,0,0,83
mitotic_expert,7th0001,-1,-1,041163,33208,700,700,700,700,1,1
mitotic_counts,8th0001,-1,-1,1

The Header is all the text before the line containing “BODY”. Please refer to the figure which shows the input file being edited in a plain text editor. The first section is free text and should be used to outline the author, experimental setup, date, time, and a short description of the experiment and hardware. The figure above shows what objectives are expected on the microscope. Following the line containing “SETTINGS”, global parameters are defined. First is the number of WSI and the figure shows how to specify the corresponding WSIs (full paths and filenames). Next, "label_pos" define the direction of the glass slide in microscope stage. And microscope settings that are under control are then defined: the objective, filter, and reflector. The GUI and screen characteristics are defined: foreground color, axes color, background color, font size and screen size. Colors are defined using simple RGB coordinates scaled within the range of zero to one. Then "saveimages" controls whether save experiment image. Finally, the last line of text “taskorder” controls the order the tasks are presented to the reader.

Input File Body

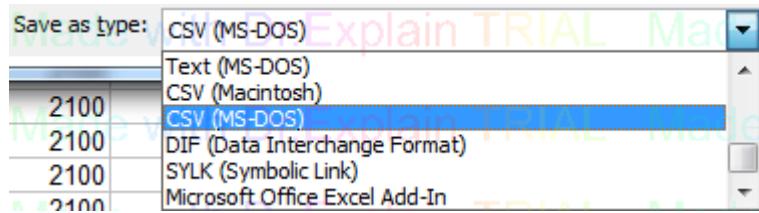
Input File Body: Opening and Editing



The Body of an input file is editable by a spreadsheet editor such as Excel. For example, open Excel and navigate to eeDAP→Sample_Input_Files→Phantom_Test and open Phantom_Test.dapsi. Make sure in Excel's open dialogue “all files is selected” (See Figure), otherwise you won't file eeDAP input files with the extension “.dapsi”.

After selecting Phantom_Test.dapsi, the “Text Import Wizard, Step 1 of 3” will open (See Figure). Select the “Delimited” option and click “Next”.

In the next window, “Text Import Wizard, Step 2 of 3”, select how the file is delimited, in this case highlight “Comma”.



In the final window, “Text Import Wizard, Step 2 of 3”, select “Finish”, this will open the file in Excel, aligning rows and columns, and allowing editing. Once editing is complete, “Save As” a “.CSV” file which is plain text with comma separated variables.

Input File Body: List of ROIs and Corresponding Tasks

The input file Body begins with the line containing “BODY” and follows with a line of column labels. Each line below the line of column labels represents a task, identifying an ROI and the task. There are 19 columns identifying the ROI and task. See “Selecting ROIs” in the “Utilities” section, and the “Tasks” section below. The 19 columns are explained here.

Default task input format

A.

Task Name: Describe name for each task

Task ID: Identification number of the task independent from the order that they are presented

Task Order: The order the slide is to be presented to the reader if the value of “taskorder” defined in the Header is set to “1”.

Slot: Specifies the stage slot for the task. This, consequently, specifies the glass slide for the task based on the header information.

B.

ROI_X: Defines center of the ROI; point of reference for the extraction process

ROI_Y : Defines center of the ROI; point of reference for the extraction process

ROI_W: Width of the area in pixels to be extracted for the ROI

ROI_H: Height of the area in pixels to be extracted for the ROI

IMG_W: This parameter defines the size of the image that will be presented to the reader.

IMG_H: This parameter defines the size of the image that will be presented to the reader.

Q_Text: This field contains the question (free text) displayed to the reader. Do not use commas here or the input file format will be corrupt.

MoveFlag: A value of 1 allows for the Reader to pan, value of 0 prohibits

ZoomFLag: A value of 1 allows for the Reader to zoom, value of 0 prohibits

Q_Op1: Displays text for this answer

Q_Op2: Displays text for this answer

Q_Op3: Displays text for this answer

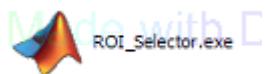
Q_Op4: Displays text for this answer

Customized Input File Body: List of ROIs and Corresponding Task

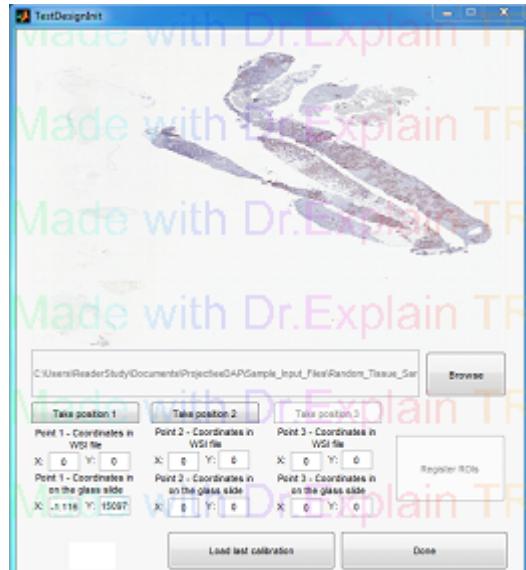
Customized Input File format should follow Task Name, Task ID, Task Order and Slot. The left parts could design for different task

Selecting ROIs???

Launch the ROI_Selector.exe . It is found in the “eeDAP→Utilities” directory.

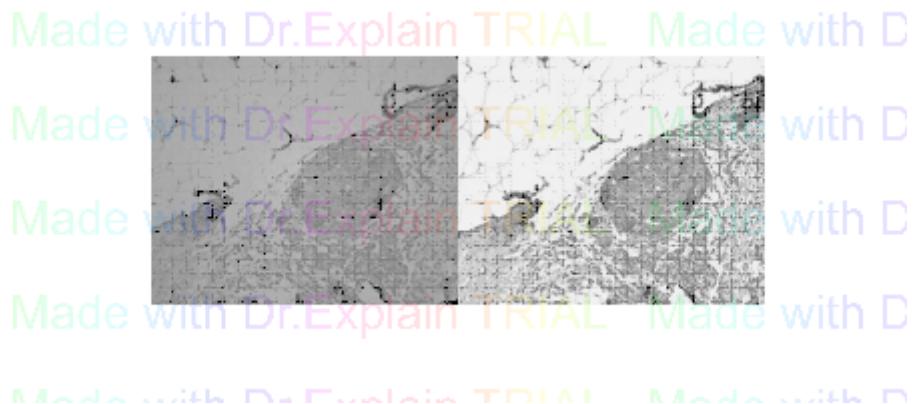


The ROI_Selector program will open the “??? ” window and the video preview window. Browse to the WSI from which the ROIs will be exacted from. Be sure the microscope is positioned over the corresponding glass slide.



The following steps describe how to first register the glass slide and the WSI.

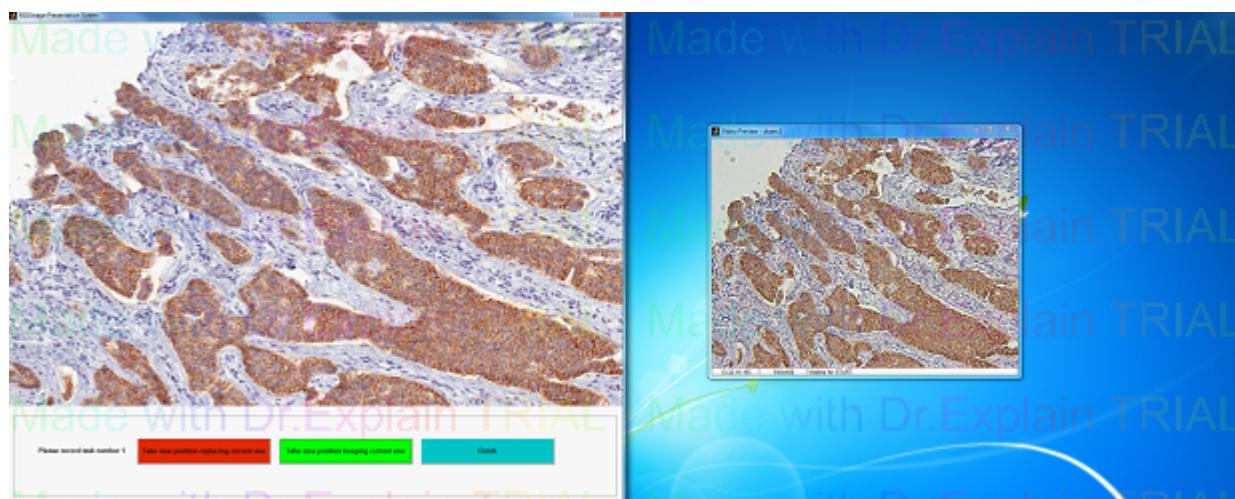
- # Use the joystick to control the microscope stage, navigate to a non-homogenous area of the slide, such as the end of the tissue, or a unique feature in the tissue.
 - # Once satisfied with the feature location press “Take position 1”.
 - # Now, repeat this process twice more using the “Take position 2” and “Take position 3”. For the best results maximize the distance between these three points
 - # Press “Register ROI”, , this may take up to a few minutes to execute.
 - # Compare the pairs of ROIs to ensure that the glass slide and WSI are properly registered. If executed properly the ROIs are identical, see image on right.
Note that illumination differences are possible.
 - # If satisfied, press the “Done” button



When using the same slides, and they have not been removed or adjusted in the microscope stage, the “Load Last Calibration” button will load the coordinates from the previous registration.

If the resulting image do no match, or are miss aligned refer to the “Manual Registration” section.

Once registration is complete the Input File Creator Window opens in addition to the desired FOV and presses the Green button. This selects the current FOV, extracts the File Creator Window. If satisfied with this selection navigate the microscope to the next field of view and repeat the process. Once all fields have been selected navigate the microscope to a superior FOV and use the Red button to replace the Input File Creator Window. Finally press the Blue “Finish button”. A file has been written to the current directory containing the extracted images.



Input File Creator

XI. Output File

An “Output_Files” folder will be created in the folder where the input file is. eeDAP writes to this folder the experiment output files (file extension .dapso). The naming convention is “Mode.ReaderName.Date .InputFileRootName.dapso”. eeDAP also writes to a subfolder (name similar to output file) images captured by the microscope’s camera and WSI depend on the setting of "saveimages" in input file . The images captured are taken when the task is completed, and the Administrator selects the “Next” button, advancing to the next task. The captured images provide further opportunities for verification of registration and other considerations.

eeDAP Output files are very similar in appearance to input files. They are comma delimited and have a Header (mostly free-text) and Body (comma-separated variables). Please refer to the “Input File” section for basics on the content of the Output file and instructions for opening and editing the Output file. Once this file is open the header will be displayed, figure to the right. Much of this information should be repetitive from the input file’s header.

The main differences between the Input file and the Output file are as follows:

- The Header begins with the Output file filename, which specifies the reading mode, reader name, date, time and the input file name and
- The Body includes additional column for task Duration, that is the amount of time the reader spent on each task in recorded in seconds, and some other columns related to the reader’s responses.

Slide Value: A discrete value between 1-100, if a slider was presented to the reader, a value of 0 indicates that there was no slider for this task

Option 1 Value: If the reader selected this option then the value will be 1

Option 2 Value: If the reader selected this option then the value will be 1

Option 3 Value: If the reader selected this option then the value will be 1

Option 4 Value: If the reader selected this option then the value will be 1

XII. Directory Structure

A typical eeDAP installation will have the following structure.

