Virtual Staining: Instructions

Disclaimer

This self-contained software package allows to test the virtual staining framework on a wide range of machines, without requiring GPU or any software. The software is built upon ImageJ platform. Since the software package is general an does not run on GPUs it performance is substantially slower than reported timings in the referring manuscript: “Deep learning-based virtual histology staining using auto-fluorescence of label-free tissue”.

General instruction

After downloading our file and unzipping it, please:

1. Double-click the application: ImageJ-win64.exe file in Fiji.app folder.
2. An automated Updater option might appear. Please do not update as new ImageJ core has  
   API bugs.
3. On the application, go to Process Menu and select Virtual Staining at the bottom, and start testing various image data that are included in this package by browsing  
   our data.

For example, you can:

* Select 'Test Single' option or 'Test Folder’ option. (Test Single: open and virtually stain a single image; Test Folder: run the model on all images inside a folder)
* Select the Model folder by clicking 'Browse'. For example, go to the ...Data/Models/ folder in this package to select a Folder for a specific Model. For example,  
  select ...Data/Models/LiverMT3 or other models. All the folders are within the same unzipped folder. Once the Model folder is selected, click OK. *Please select the model folder instead of selecting any files inside.*
* If the 'Test Single' Option is selected, then pick a single image to be used as Input image to the Model. For example: go to …/Data/Example\_images /LiverMT3/input folder or …/Data/Example\_images /LiverMT3/input folder and select a tif image which will be the Input to the Network. Depending on the Model you have selected, please use the Input images from the corresponding Input folder. For example, if your model is: ...Data/Models/LiverMT3 then select an image from …/Data/Example\_images/LiverMT3/input folder; same idea applies for all our models.
* After the Input image is selected, then the network output will be calculated. You will see  
  Input vs. Network images once the network is done. To better visualized the auto-fluorescence images, use ctrl+shift+C in ImagJ/Fiji to adjust contrast.
* Ground truth images are also included in the same package, see for example the  
  folder: …/Data/Example\_images/LiverMT3/target

Software version

The Fiji.app(with\_Plugin) in this package is the official Fiji v1.5.2e integrated with our plugin.

If an Updater option appears. Please do not update as new ImageJ core have API bugs.

This plugin is developed with Fiji/ImageJ version 1.5.2e. You can check it by clicking Help-  
>About ImageJ.

Prepare testing data

The testing images must be 16-bit grayscale tif images of size 1224×1224.

Example images are provided in [This package folder]/Data/Example\_images.

(To better visualized the auto-fluorescence images, use ctrl+shift+C in ImagJ/Fiji to adjust contrast.)

Please use the correct pretrained model with corresponding test images for the best network  
inference results.

Test a pre-trained model

1. In ImageJ, Go to Process->Virtual Staining at the bottom
2. Select 'Test Single' option or 'Test Folder’ option

o Test Single: open and enhance a single image

o Test Folder: run the model on all images inside a folder

1. Select Model Directory (For example, [This package folder]/Data/Models/Kidney  
   folder in this package). Please select the model folder instead of selecting any files inside.
2. If the Test Single Option is selected:
   * Select a single image. For example: [This package folder]/Data/Example\_images  
     /LiverMT3/input/[imagename].tif
   * Then the network output will show up
3. If the Test Folder Option is selected:
   * Select an input image directory. (For example: [This package  
     folder]/Data/Example\_images /LiverMT3/input)
   * Select or type in an output image directory name.
   * The plugin will create a folder with network inferred images with the selected  
     model.