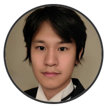
CODA (part 4): register the deep learning labelled images and Construct 3D tissue matrix | HuBMAP | JHU-TMC

Kyu Sang Han[1], Pei-Hsun Wu[1], Joel Sunshine[2], Ashley Kiemen[2], Sashank Reddy[2], Denis Wirtz[1,2]

[1]Johns Hopkins University; [2]Johns Hopkins Medicine

Human BioMolecular Atlas Program (HuBMAP) Method Development Community

TMC - Johns Hopkins University



Kyu Sang Han
Johns Hopkins University

APR 01, 2024

DOI:
dx.doi.org/10.17504/protocols.io.yxmvme7eog3p/v1

Protocol Citation: Kyu Sang Han, Pei-Hsun Wu, Joel Sunshine, Ashley Kiemen, Sashank Reddy, Denis Wirtz 2024. CODA (part 4): register the deep learning labelled images and Construct 3D tissue matrix | HuBMAP | JHU-TMC. **protocols.io** https://dx.doi.org/10.17504/protocols.io.yxmvme7eog3p/v1

MANUSCRIPT CITATION:
Kiemen, A.L., Braxton, A.M., Grahn, M.P. *et al.* CODA: quantitative 3D reconstruction of large tissues at cellular resolution. *Nat Methods* **19**, 1490–1499 (2022). https://doi.org/10.1038/s41592-022-01650-9

ABSTRACT

CODA (part 4): register the deep learning labelled images and Construct 3D tissue matrix

Protocol status: Working
We use this protocol and it's working

Created: Mar 28, 2024

Last Modified: Apr 01, 2024

PROTOCOL integer ID: 97512

Keywords: CODA, deeplearning, semanticsegmentation

Funders Acknowledgement:
Institute of Arthritis and Musculoskeletal and Skin Diseases
Grant ID: U54AR081774
National Cancer Institute
Grant ID: U54CA143868

Register the deep learning labelled images

1 Using the registration transformations calculated in low-resolution (as described in CODA-part2), register the segmentation masks of the high-resolution tif images generated in CODA-part3.

Function requirements

2 High-resolution images from CODA-part3 (**pthclassified**)

3 The path containing the image registration information from CODA-part2 (**pthdata**),

4 The scale between the high-resolution classified images and the low-resolution registration images (here for registration of 1x images and classification of 10x images, scale=10)

5 The pixel number of the background class in the segmentation model built in CODA-part3 (here, nwhite=3)

Execution

6

```
pthim=pthclassified;
pthdata=[pth,'1x\registered\elastic registration\save_warps'];
scale=10;
padnum=3;
apply_image_registration(pthim,pthdata,scale,padnum);
```

7 This function will create an subfolder named ‘registeredE’ inside the **pthclassified** folder that contains the registered, classified images. **pthclassifiedE=[pthclassified,'registeredE'];**

Construct 3D tissue matrix

8 Next, we will create a 3D quantifiable matrix from the tissue labels using the registered, segmented images we just created above. **vol** will be a 3D matrix containing the registered, labelled data.

9 To call this function, you need to locate the subfolder containing the registered classified images, define a location to save the output matrix, define the desired resolution of the volume matrix (relative to the current resolution of the images, and identify the RGB color map used in the deep learning model in CODA-part3

10 **pthclassifiedE** is the subfolder with the registered, classified images created in earlier steps in this protocol:

pthclassifiedE=[pthclassified,'registeredE'];

11 Define a subfolder where you would like to save the volumetric data:
pthvolume=[pth,'lung_data'];

12 Set **sk** to 4 so that 10x image (1 um / pixel resolution) are downsampled to 4 micron / pixel:
sk=4;

13 **nwhite** is the background class from the deep learning model. Here, let's use 6:
nwhite=6;

14 **cmap** is the matrix containing RGB triplets for each class in the deep learning model
cmap=[];

build_tissue_volume(pthclassifiedE,pthvolume,sk,nwhite,cmap);

15 First, the function will display to you a concatenated image containing the first, center, and last classified image in your image stack. Here, you will manually drag a rectangle to surround the tissue in the image, then double-click. This allows you to crop out excess whitespace from your 3D matrix to save RAM.

16 Next, the function will load all serial images, crop them, downsample them using the variable **sk**, and build them into the matrix **vol**. A z-projection will be created and displayed showing a combination of all non-background classes. This z-projection will be saved inside the folder **pthvolume**.

17 Additionally, this function will create a .mat file named **volume.mat** saved inside the subfolder **pthvolume**. Inside this file will be a variable named **vol** containing the volumetric tissue labels, **rr** containing the cropping information defined inside the function, **pthclassifiedE** the folder containing the registered classified images

used to create the volume, **imlist** containing the filenames of the images comprising the 3D matrix, and **sk** the downsample factor between the images in **imlist** and the data in **vol**.

Note: When determining what resolution to make your volumetric matrix, t…

18 You may want to make your 3D matrix isometric (same resolution in x, y, and z) to simplify 3D quantifications. In this case, define **sk** such that your classified images will be downsampled to the spacing between adjacent histological images

19 You may want to consider the size of structures you are trying to quantify. If you need to maintain classification of very fine structures (such as thin blood vessels), down sampling too much may eliminate the connectivity between these thin objects.