**LOCATE – LOCALLY ADAPTIVE THRESHOLD ESTIMATION**

**User Manual**

**Contents:**

1. Introduction
2. Initial setup
3. Dataset preparation and input files required
4. Running LOCATE
   1. Leave-one-subject-out (LOSO) validation
   2. LOCATE training
   3. LOCATE testing
5. Viewing results with FSLeyes
6. **Introduction**

LOCATE (LOCally Adaptive Thresholds Estimation) is a supervised method to determine thresholds for binarising the subject-level lesion probability map (LPM) with specific applicability to BIANCA (https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/BIANCA). In principle, LOCATE can be applied to the LPM obtained by any method (not just BIANCA), provided there is some available training data with manual lesion masks.

LOCATE has the potential to improve the subject-level lesion segmentation, compared to the use of a global threshold, by making the threshold vary spatially across the brain and therefore also differing between subjects. Results can improve with respect to detecting more deep lesions, and being less sensitive to variability in lesion load without requiring additional training data, beyond that used in BIANCA classifier for LPM estimation. For the detailed information about the LOCATE method and validations, please refer our article <https://www.biorxiv.org/content/early/2018/10/08/437608>.

Thanks for trying our Beta version of LOCATE and we look forward to your feedback and comments to improve LOCATE.

1. **Initial setup**

LOCATE is currently available in MATLAB and the scripts are available in the following Github repository:

[**https://git.fmrib.ox.ac.uk/vaanathi/LOCATE-BIANCA**](https://git.fmrib.ox.ac.uk/vaanathi/LOCATE-BIANCA)

To run LOCATE in your machine, clone the git repository in your working folder. For doing this, open your terminal and type the following command in your working folder:

**git clone** [**https://git.fmrib.ox.ac.uk/vaanathi/LOCATE-BIANCA**](https://git.fmrib.ox.ac.uk/vaanathi/LOCATE-BIANCA)

Now change your working directory to LOCATE-BIANCA:

**cd LOCATE-BIANCA/**

Note: Henceforth, in this manual ‘LOCATE-BIANCA/’ is referred as the ‘present working directory’.

Also, please ensure that you have FSL installed in your machine and the following environment variables/paths are correctly set in your system by typing the following commands in your terminal and check if they provide similar output:

echo $FSLDIR

/usr/local/fsl

echo $FSLOUTPUTTYPE

NIFTI\_GZ

1. **Dataset preparation and input files required**

LOCATE is a supervised method and hence requires data for training. In the current version the user needs to provide data in a specific, standardised manner (which can typically be achieved by some moving and renaming of files).

LOCATE can be applied in various ways – you can train the LOCATE model on data (images of the subjects) present in a directory, test the LOCATE model (if you already have a LOCATE model) on data in a directory or perform leave-one-subject-out evaluation (LOSO) of LOCATE model on the data in a directory. As the first step, images of the subjects used for training and testing need to be grouped in separate directories, named ‘Training\_imgs’ and ‘Test\_imgs’ in a directory of your interest (eg. Myfolder).

LOCATE requires the following images for each subject (either belonging to the training or testing set). In the current version all the images need to have the standard name specified below:

1. The base image modality used in BIANCA.

**<subject\_name>\_feature\_<base\_modality\_name>.nii.gz**

It is essential that you provide at least one image modality (preferably the base image modality used in BIANCA). For example, if you used FLAIR as your base image modality in BIANCA, your base\_modality\_name will be ‘FLAIR’. So the image has to be named as <subject\_name>\_feature\_FLAIR.nii.gz

1. Any other additional images that were used as intensity features in BIANCA (--featuresubset) (optional)

**<subject\_name>\_feature\_<modality name>.nii.gz**

(eg. <subject\_name>\_feature\_T1.nii.gz, please note that it is mandatory to add \_feature\_ in the filename for it to be considered as a feature)

1. Lesion Manual mask (for training subjects only): binary mask (values of 0 and 1) indicating lesion voxels, based on manually segmentation

**<subject\_name>\_manualmask.nii.gz**

1. Ventricle distance map: image where each voxel intensity represents the distance from ventricles within the brain mask (this can be calculated using the FSL tool *distancemap* – see the wiki for more details, example call:distancemap -i <ventricle\_mask\_image\_in\_FLAIR\_space> -m <brain\_mask\_in\_FLAIR\_space> -o <subject\_name>\_ventdistmap.nii.gz)

**<subject\_name>\_ventdistmap.nii.gz**

1. BIANCA output: unthresholded lesion probability map (LPM) obtained from BIANCA

**<subject\_name>\_BIANCA\_LPM.nii.gz**

1. Brain mask: binary mask obtained from FSL-BET or any other method

**<subject\_name>\_brainmask.nii.gz**

1. BIANCA mask: binary mask, as obtained from make\_bianca\_mask (white matter mask excluding sub-cortical regions) - If you are not using BIANCA mask in your analysis, make a copy of the brain mask and rename it as BIANCA mask.

**<subject\_name>\_biancamask.nii.gz**

1. **Running LOCATE**

LOCATE can be run in three main ways for performing following operations: (1) Leave-one-subject-out validation; (2) train LOCATE model on a set of subjects in a directory and (3) test the LOCATE model on a set of subjects in a directory. You want to use LOSO testing evaluation of the LOCATE performance when the dataset has manual segmentations for all the subjects. On the other hand, if you do not have ground truth for all subjects, you can train using the ones with ground truth and test on the rest. Training and testing on separate sets of subjects is also logical if you have done the same for BIANCA.

The following three subsections outline the various options available within each of these main ways of running LOCATE.

Under each section, we have provided the example function call (for beginners), the main outputs that could be obtained by running LOACTE as specified in each section. Additionally, we have provided the list of optional inputs that could be provided to the LOCATE function call to make LOCATE more specific to the data/or feature sets. For providing the additional options. No additional input images are required unless specified in the section. Finally, we provide the complete list of outputs for running LOCATE in a specific way in each section.

* 1. **Leave-one-subject-out (LOSO) validation**

This function call is used to perform the leave-one-subject-out validation. For this function, images (both base and optional image modalities) need to be present in a single directory.

**Example function call**

LOCATE\_LOSO\_testing(train\_image\_directory\_name);

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located. If you had followed the instructions provided in section 2 and named the directory containing the training images as ‘Training\_imgs’, then train\_image\_directory\_name = ‘Myfolder/Training\_imgs’

Note: If you have your training image files in the same directory where LOCATE\_LOSO\_testing.m is located (i.e. your present working directory), you can leave the input argument empty and call the function as LOCATE\_LOSO\_testing();

**Main outputs:**

During LOSO testing, a new directory named ‘**LOCATE\_LOSO\_results\_directory**’ will be created in the ‘Training\_imgs’ directory. All the LOCATE output files will be saved in the ‘LOCATE\_LOSO\_results\_directory’. The main output of LOCATE LOSO testing is **<subjectname>\_BIANCA\_LOCATE\_binarylesionmap.nii.gz** – binary lesion map obtained as the final output of LOCATE.

**Additional/optional inputs to the function call**

1. LOCATE\_LOSO\_testing(train\_image\_directory\_name, feature\_select);

If you want to select specific features for training and testing

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located.

- feature\_select - vector with elements indicating if the feature has to be included or not (1 – to be included, 0 to be discarded). Current order is: distance from ventricles, size of individual lesions (for the pathologies where size of lesion matter – if you are unsure, put ‘1’) and other modalities (including the base\_image\_modality) in alphabetical naming order.

(e.g. If <base\_image\_modality> is the only modality provided and distance from ventricles is not needed then feature\_select = [0, 1, 1])

Note: Please note that you **do not** need to provide any additional input images if you choose to include the second feature (size of individual lesions). This feature will be calculated from the <subject\_name>\_BIANCA\_LPM.nii.gz image internally in the function. This option feature\_select is simply to indicate if the feature will be needed or not.

1. LOCATE\_LOSO\_testing(train\_image\_directory\_name, feat\_select, verbose);

* verbose (default = 0; 1 if the steps need to be displayed on the screen)

For the description of other additional options, refer to the point 1 above.

Any of the above commands performs LOSO validation for all the images (with <subjectname>\_BIANCA\_LPM.nii.gz) in the ‘Training\_imgs’ directory using the selected features.

**Complete list of outputs**

As mentioned before in the ‘main outputs’ section, during LOSO testing, a new directory named ‘**LOCATE\_LOSO\_results\_directory**’ will be created in the ‘Training\_imgs’ directory. All the LOCATE output files will be saved in the ‘LOCATE\_LOSO\_results\_directory’. The following text is a list of the various output files - the main output is highlighted by bold, red text.

The contents of LOCATE\_results\_directory are:

1. LOCATE\_test\_features.mat’ – an intermediate MATLAB data file containing test features for all the test images in a single file.
2. <subjectname>\_indexmap.nii.gz – image showing the Voronoi polygongs obtained from the Voronoi tessellation step
3. <subjectname>\_thresholdsmap.nii.gz – image showing the local thresholds within the Voronoi polygons
4. **<subjectname>\_BIANCA\_LOCATE\_binarylesionmap.nii.gz** – binary lesion map obtained as the final output of LOCATE
5. <subjectnames>\_LOCATE\_thresholds.mat– array of thresholds (basically, the same thresholds shown in the <subjectname>\_thresholdsmap.nii.gz in step 3). If you want the threshold values directly for any further analysis or plotting (e.g., a histogram), this output could be useful.
6. Consolidated\_LOCATE\_output.mat – The outputs from 2, 3, 4 and 5 for all the images available in the a single .mat file.
   1. **LOCATE\_training:**

This function call is used to train and save the LOCATE model using images in a given directory. Please note that running this function does not perform evaluation (or provide final output). You need to run LOCATE\_testing function (explained in section 3.3) for getting the final thresholded maps.

**Example function call**

LOCATE\_training(train\_image\_directory\_name);

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located. If you had followed the instructions provided in section 2 and named the directory containing the training images as ‘Training\_imgs’, then train\_image\_directory\_name = ‘Myfolder/Training\_imgs’

Note: If you have your training image files in the same directory where LOCATE\_training.m is located (i.e. your present working directory), you can leave the input argument empty and call the function as LOCATE\_training();

**Main outputs:**

During LOCATE training, a new directory called ‘**LOCATE\_training\_files’** will be created in the ‘Training\_imgs’ directory, and all the files including the model created during training will be saved in this folder. The main output of training is ‘**RF\_regression\_model\_LOCATE.mat’** - a regression model that is used in LOCATE testing. Please note that this file is a MATLAB data file and is different from the linear transformation matrix files used in FSL.

**Additional/optional inputs to the function call**

1. LOCATE\_training(train\_image\_directory\_name, feature\_select);

If you want to select specific features for training

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located.

- feature\_select - vector with elements indicating if the feature has to be included or not (1 – to be included, 0 to be discarded). Current order is: distance from ventricles, size of individual lesions (for the pathologies where size of lesion matter – if you are unsure, put ‘1’) and other modalities (including the base\_image\_modality) in alphabetical naming order.

(e.g. If <base\_image\_modality> is the only modality provided and distance from ventricles is not needed then feature\_select = [0, 1, 1])

Note 1: Please note that you **do not** need to provide any additional input images if you choose to include the second feature (size of individual lesions). This feature will be calculated from the <subject\_name>\_BIANCA\_LPM.nii.gz image internally in the function. This option feature\_select is simply to indicate if the feature will be needed or not.

Note 2: Unlike LOSO, here you will be performing training alone, and the created model can be evaluated on any set of images using LOCATE\_testing. It is essential that the image modalities and feature\_select values remains consistent for both training and testing. For example, if you run LOCATE\_training on T1 and PD images with feature\_select option [1,0,1,1], the same modalities and feature\_select options need to be provided for the test data while testing LOCATE (explained in section 3.3) as well.

2. LOCATE\_training(train\_image\_directory\_name, feature\_select, verbose);

- verbose (default = 0)

For the description of other additional options, refer to the point 1 above.

Any of the above commands performs training using all the images (with <subjectname>\_BIANCA\_LPM.nii.gz) in the ‘Training\_imgs’ directory using the selected features.

**Complete list of outputs**

As mentioned before, during LOCATE training, a new directory named ‘**LOCATE\_training\_files**’ will be created in the ‘Training\_imgs’ directory. All the LOCATE training files will be saved in the ‘LOCATE\_training\_files’. The following text is a list of the training files - the main output is highlighted by bold, red text.

1. LOCATE\_features\_<subjectname>.mat – containing features of individual training subjects
2. LOCATE\_features.mat – containing features of all the subjects in a single .mat file (this is the file that will be needed in the testing phase)
3. **RF\_regression\_model\_LOCATE.mat** – Trained Random Forest regression model for LOCATE in the training phase. This model could be applied to any test dataset (it need not be the same dataset or have the same degree or amount of lesions) with image dimensions matching the image training dataset.
   1. **LOCATE\_testing:**

This function call is used to evaluate the LOCATE model on images in a given directory. Please note that for running this function, you should have already run LOCATE training (explained in section 3.2) and have a trained model named ‘RF\_regression\_model\_LOCATE.mat’ available. This function call performs the testing and provides the final thresholded maps and the threshold values.

**Example function call**

LOCATE\_testing(test\_image\_directory\_name , train\_image\_directory\_name);

- test\_image\_directory\_name - Name of the directory where the test images are located. If you had followed the instructions provided in section 2 and named the directory containing the test images as ‘Test\_imgs’, then test\_image\_directory\_name = ‘Myfolder/Test\_imgs’

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located. If you had followed the instructions provided in section 2 and named the directory containing the training images as ‘Training\_imgs’, then train\_image\_directory\_name = ‘Myfolder/Training\_imgs’

Note: Again, if you have your training image files in the same directory where LOCATE\_testing.m is located (i.e. your present working directory), you can leave the second input argument empty and simply call the function as LOCATE\_testing(test\_image\_directory\_name);

**Main outputs:**

During LOCATE testing, a new folder named ‘**LOCATE\_results\_directory’** will be created in the ‘Test\_imgs’ directory, and all the LOCATE output files in ‘LOCATE\_results\_directory’. The main LOCATE output, the binary lesion map, **‘<subjectname>\_BIANCA\_LOCATE\_binarylesionmap.nii.gz’** will be saved in this folder.

**Additional/optional inputs to the function call**

1. LOCATE\_testing(test\_image\_directory\_name, train\_image\_directory\_name, feature\_select)

If you want to select specific features for testing

- test\_image\_directory\_name – Name of the directory where the test images are located

- train\_image\_directory\_name - Name of the directory where the training images for feature extraction are located.

- feature\_select - vector with elements indicating if the feature has to be included or not (1 – to be included, 0 to be discarded). Current order is: distance from ventricles, size of individual lesions (for the pathologies where size of lesion matter – if you are unsure, put ‘1’) and other modalities (including the base\_image\_modality) in alphabetical naming order.

(e.g. If <base\_image\_modality> is the only modality provided and distance from ventricles is not needed then feature\_select = [0, 1, 1])

Note 1: Please note that you **do not** need to provide any additional input images if you choose to include the second feature (size of individual lesions).

Note 2: The feature\_select option values must consistent with the values used for training LOCATE (explained in section 3.2).

2. LOCATE\_testing(test\_image\_directory\_name, train\_image\_directory\_name, feature\_select, verbose);

- verbose (default = 0)

For the description of other additional options, refer to the point 1 above.

Any of the above commands evaluates LOCATE on all the images (with <subjectname>\_BIANCA\_LPM.nii.gz) in the ‘Test\_imgs’ directory using the selected features, using the model trained using all the images from ‘Training\_imgs’ directory.

**Complete list of outputs**

As mentioned before, during LOCATE testing, a new directory named ‘**LOCATE\_results\_directory**’ will be created in the ‘Test\_imgs’ directory. All the LOCATE output files will be saved in the ‘LOCATE\_results\_directory’. The following text is a list of the training files - the main output is highlighted by bold, red text.

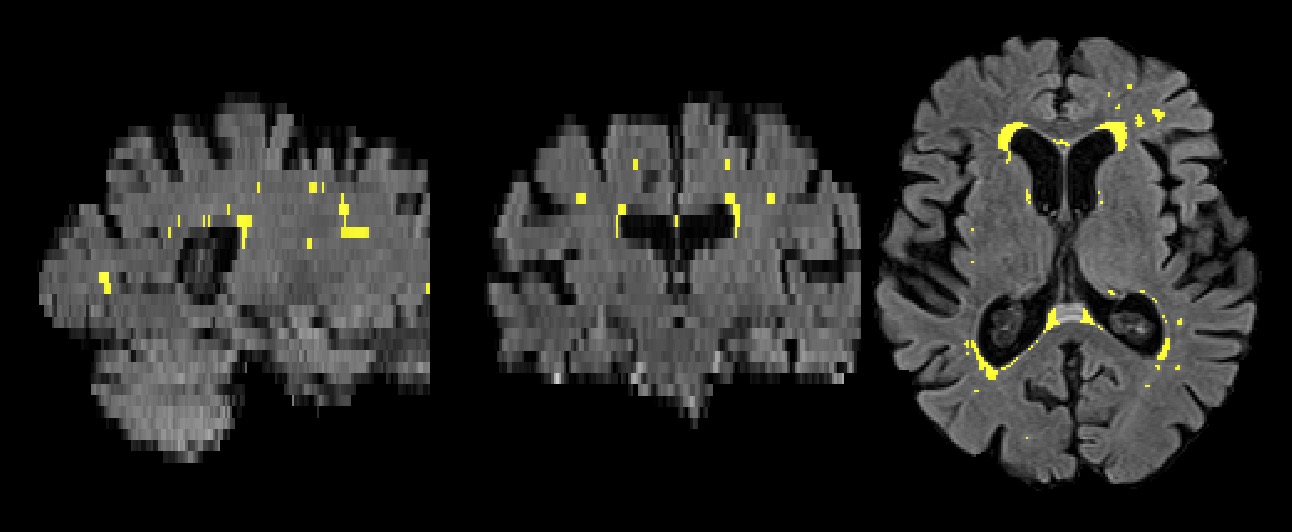
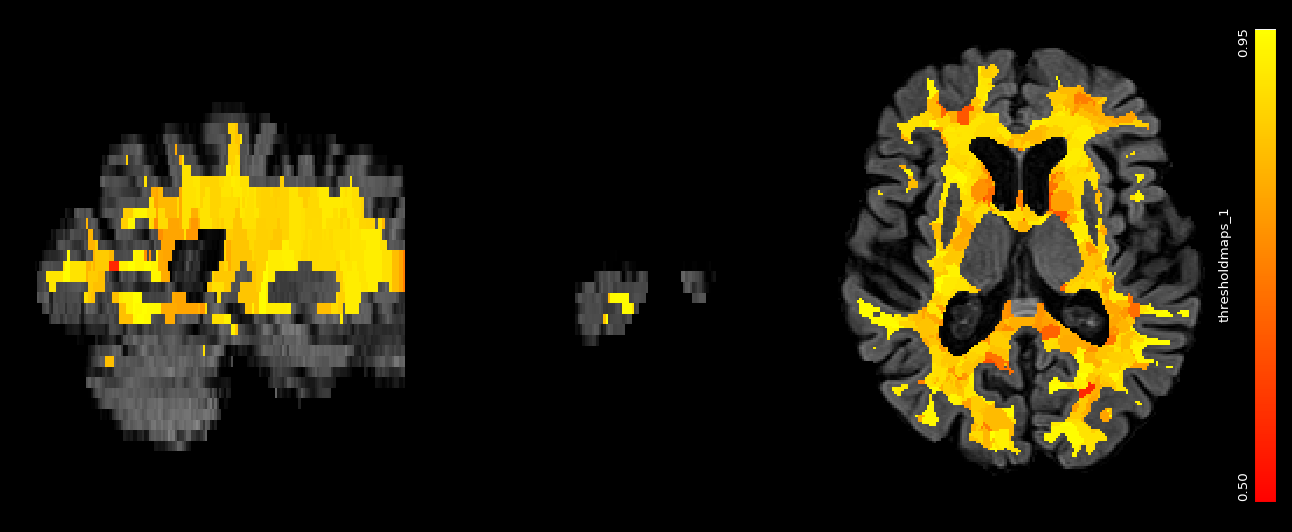
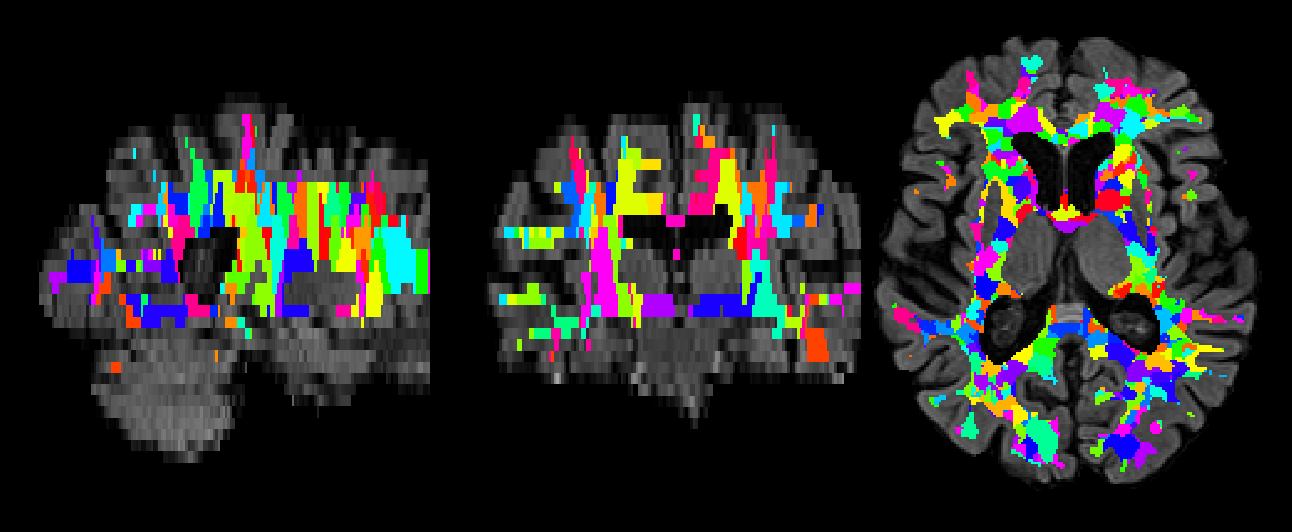
1. LOCATE\_test\_features.mat’ – an intermediate file containing test features for all the test images in a single file
2. <subjectname>\_indexmap.nii.gz – image showing the Voronoi polygons obtained from Voronoi tessellation step
3. <subjectname>\_thresholdsmap.nii.gz – image showing the local thresholds within the Voronoi polygons
4. **<subjectname>\_BIANCA\_LOCATE\_binarylesionmap.nii.gz** – binary lesion map obtained as the final output of LOCATE
5. <subjectnames>\_LOCATE\_thresholds.mat– array of thresholds (basically, the same thresholds shown in the <subjectname>\_thresholdsmap.nii.gz in step 3). If you want the threshold values directly for any further analysis or plotting (e.g. a histogram), this output could be useful.
6. Consolidated\_LOCATE\_output.mat – The outputs from 2, 3, 4 and 5 for all the images available in a single .mat file.

**5. Viewing results with FSLeyes**

The sample output maps and images are shown overlaid on the base modality image (e.g. the FLAIR image).

**Example command (one single line)**

fsleyes myfolder/Test\_imgs/LOCATE\_results\_directory/<subjectname>\_BIANCA\_LOCATE\_binarylesionmap.nii.gz --cmap yellow myfolder/Test\_imgs/LOCATE\_results\_directory**/**<subjectname>\_indexmap.nii.gz --cmap random /Users/vaanathi/Documents/Vaanu/DPhil\_work/Improving\_BIANCA\_Segmentation/ myfolder/Test\_imgs/LOCATE\_results\_directory/subjectname>\_thresholdsmap.nii.gz --cmap red-yellow myfolder/Test\_imgs/<subject\_name>\_feature\_FLAIR.nii.gz --cmap greyscale &



**Figure 2: Map showing local thresholds <subjectname\_thresholdsmap.nii.gz>**

**Figure 3: Output binary lesion map <subjectname\_BIANCA\_LOCATE\_binarylesionmap.nii.gz>**

**Figure 1: Voronoi tessellations <subjectname\_indexmap.nii.gz>**