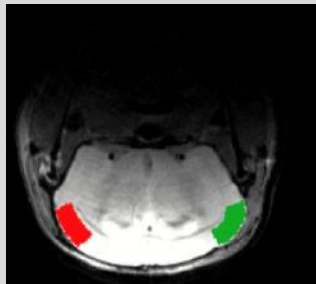# Tutorial mask-based parameter extraction
from images in native space

Scenario:

A mask was generated for each animal in native space. The mask contains two ROIs: ROI-1 with ID-1 (value is 1) and ROI-2 with ID-2 (value is 2).

For a specific image in native space, we want to extract basic parameters such as mean/min/max/number of voxels and volume for each ROI of the mask.

Example: t2.nii-image overlayed with the mask 'EMS.nii', both images in Native space. The mask contains two ROIs.

Content

1) Prerequisite

For all animals the mask file (here: 'EMS.nii') and the readout-file for parameter-extraction (here: 'rc_rCBV.nii') exist. Mask-file and readout-file should meet the following criteria:

- share same space (here: native space)
- In register to each other
- Ideally same voxel-size and identical header-matrix

2) Steps to perform the parameter extraction

-Select all animals from the left animal list box
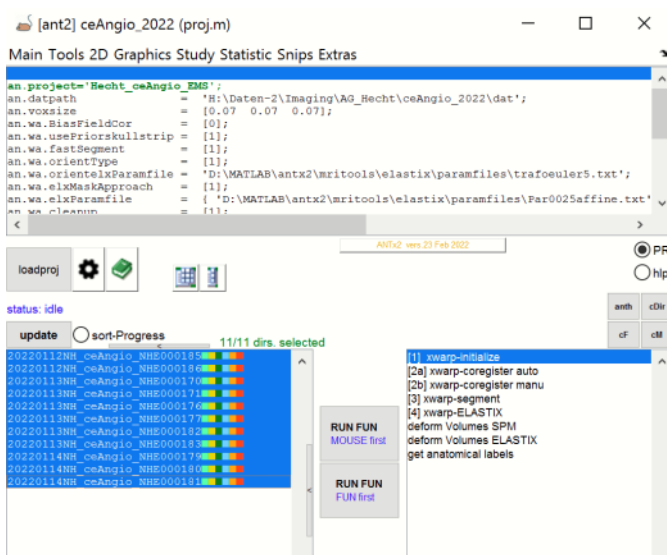
-go to MENU: **Statistic**/obtain parameter from masks
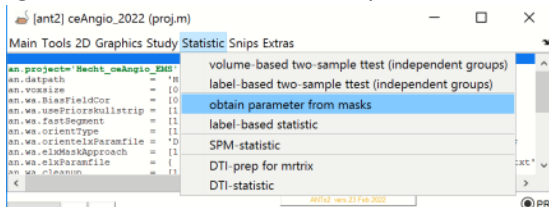
-this opens the parameter-GUI

Now do the following steps:

2.1) **Readout-file selection**

- select green icon left to the **x.file**-parameter
- from the file-section window select "**rc_rCBV.nii**" → hit [OK].

2.2) **Mask-file selection**

- select green icon left to the **x.masks**-parameter ...this opens a dialog...

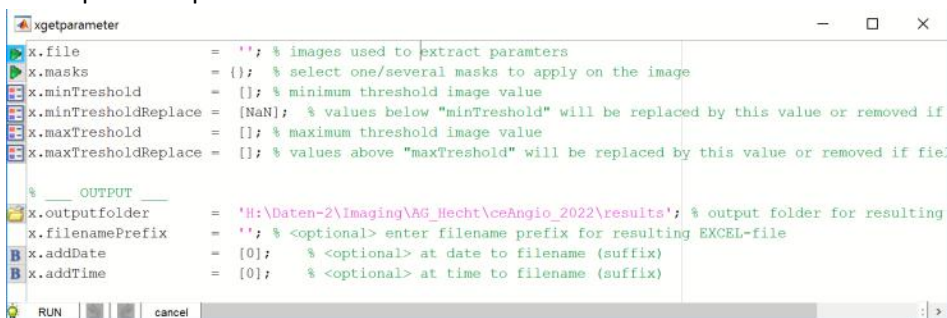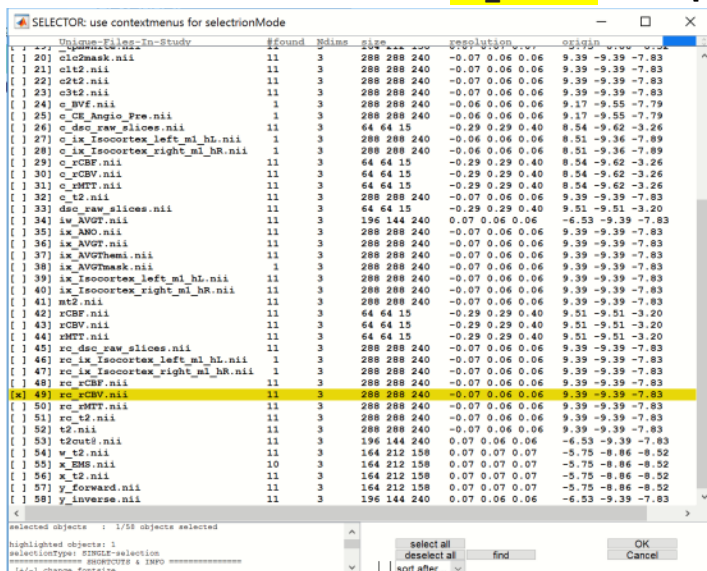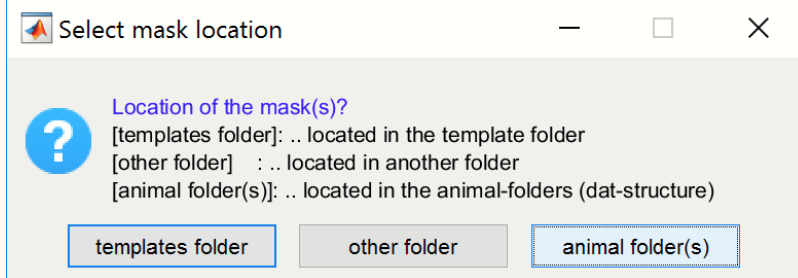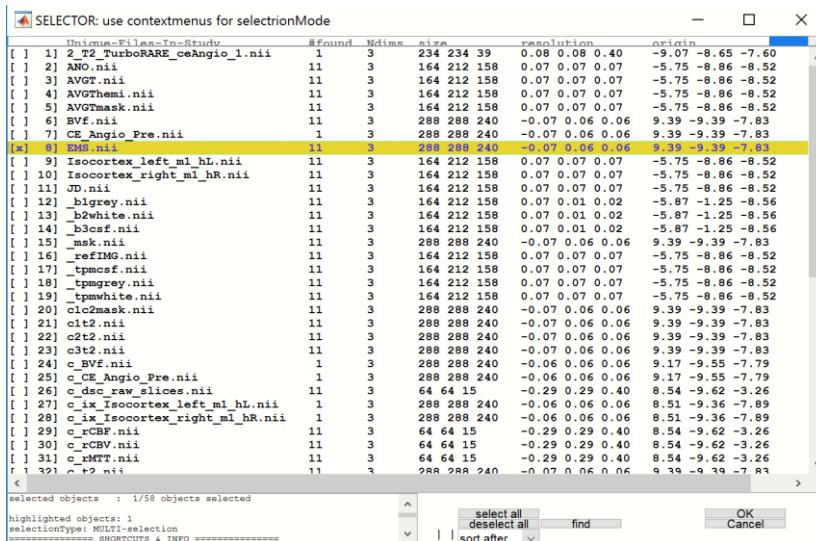-select [animal folder(s)] from the question-dialog

Reason: In this example each animal has it's own mask (native space), stored in the respective animal-folder (within the "dat"-folder).

- from the file-section window select the "**EMS.nii**"-file → hit [OK].

Remember: the "EMS.nii"-file is a mask-file containing 2 ROIs.

Important-1: **selection of multiple masks possible here**

You can also select more than one mask-image in this step. The numeric-Ids (values) in each mask-file do not interfere with the numeric ids of other selected masks during read-out and creation of the excel-file.

Example: Theoretical scenario: You have 2 masks:

-Mask1: "mask-1.nii" with 4 ROIs and respective ids: 1,2,3,4 and

-Mask2: "mask-2.nii" with 3 ROIs and respective ids: 1,2,3

→ You can select the two mask "mask-1.nii" and "mask-2.nii" in one step here

→ The resulting excel-file will contain the mask-file name and the respective ID of the two masks.

Example in the output-file:

mask-1__ID_1 .. with parameter extraction here

mask-1__ID_2 .. with parameter extraction here

mask-1__ID_3 .. with parameter extraction here

mask-1__ID_4 .. with parameter extraction here

mask-2__ID_1 .. with parameter extraction here

mask-2__ID_2 .. with parameter extraction here

mask-2__ID_3 .. with parameter extraction here

Important-2: **A text-file with specific labels can be used in combination with the mask**
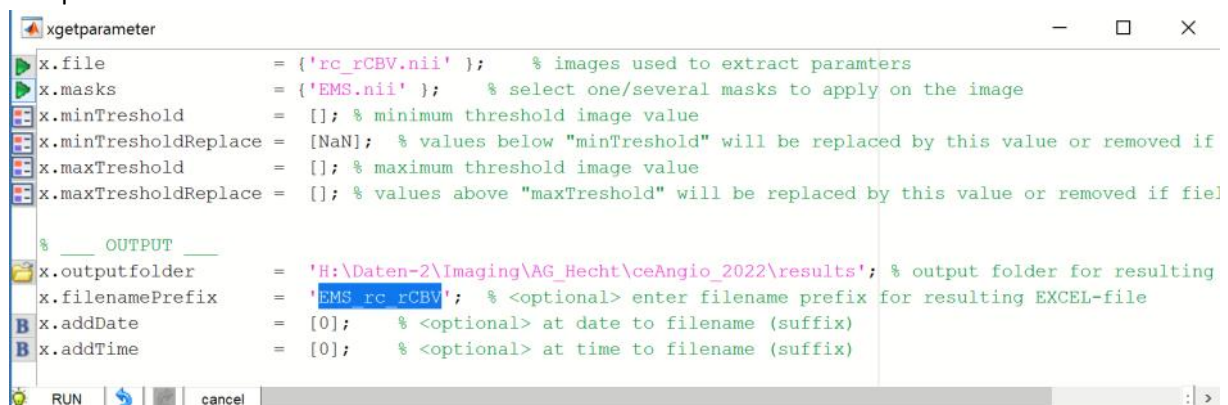
If a text-file with IDs and label is provided, the labels will be used in the resulting excel-file
To following is mandatory when using a text-file:
- text-file must have the same name as the mask-file
  → example: for "**EMS.nii**" the text-file would by "EMS.txt"
- text-file and mask-file must be in the same folder
  → in this example both files would be in the animal folders
- the text-file contains: the id (value) and the label for each ROI in the mask (row-wise)
  → for example, for this tutorial the text-file would contain:
        1 L_somat_Cortex
        2 R_somat_Cortex

2.3) **Select a filename-prefix for resulting excel-file**

- for the parameter: **filenamePrefix** type an arbitrary string that is used as prefix in the resulting filename. Here, I used 'EMS_rc_rCBV' (here a combination of mask- and readout-filename) as prefix

The parameter window should now look as follows:


```
xgetparameter                                                      —    □    ✕

x.file              = {'rc_rCBV.nii' };    % images used to extract paramters
x.masks             = {'EMS.nii' };    % select one/several masks to apply on the image
x.minTreshold       = []; % minimum threshold image value
x.minTresholdReplace = [NaN];  % values below "minTreshold" will be replaced by this value or removed if
x.maxTreshold       = []; % maximum threshold image value
x.maxTresholdReplace = []; % values above "maxTreshold" will be replaced by this value or removed if fiel

%  ____ OUTPUT ____
x.outputfolder      = 'H:\Daten-2\Imaging\AG_Hecht\ceAngio_2022\results'; % output folder for resulting
x.filenamePrefix    = 'EMS_rc_rCBV';  % <optional> enter filename prefix for resulting EXCEL-file
x.addDate           = [0];    % <optional> at date to filename (suffix)
x.addTime           = [0];    % <optional> at time to filename (suffix)

RUN  ↺ ▣  cancel                                                            : >
```

Optional Settings
1) minTreshold: set a min-threshold for the readout-image …, in this case: values below this values will by replace by the value defined by minTresholdReplace (here NAN-values).
The latter can be changed via "minTresholdReplace"-parameter.
→ example: minTreshold =3 and minTresholdReplace=3: In this case values below 3 will be replaced by the value 3
2) maxTreshold/TresholdReplace: …same logic …
3) outputfolder: change output-folder …default: "results"-folder in the study-folder
4) addDate   & addTime : add date & time to the filename of the new excel-file

---

-finally hit [RUN]

3. The output: Excel-file

-When the parameter "outputfolder" is not changed the resulting excel-file will be stored in the "**results**"-folder of the current study (aka project).
-Here the name of the **Excel-file** is "**res_EMS_rc_rCBV_maskParameter.xlsx**" ("res"+filenamePrefix + "_maskParameter.xlsx")

-The Excel-file contains the following sheets:

'info' basic information: date/mask-name/animals/ missing animals (due to missing files)

'frequencyMsk' : number of voxels for each ROI (real ROI-size , readout-image independent, true size of the ROI)

'volMsk' : volume of the ROI   (real ROI-size , readout-image independent, true size of the ROI)

'frequency' : number of voxels for each ROI (readout-image dependent, NaN/threshold dependent→thus can be smaller than 'frequencyMsk')

'vol' : volume of the ROI   (readout-image dependent (Nan-values!)) NaN/threshold dependent→thus can be smaller than 'volMsk')

'mean' : mean intensity value of readout-image within ROI

'std' : standard deviation over intensities of readout-image within ROI

'median' : median intensity value of readout-image within ROI

'integrDens' : product of number voxels and mean intensity of readout-image within ROI

'min' : min intensity value of readout-image within ROI

'max' : max intensity value of readout-image within ROI

- The parameter-sheets have the following structure:

-ROWS: …ROI-MASK , i.e., each row represents the parameter readout for a specific ROI

-COLUMNS:…ANIMALS, i.e. each column contains the data of one animal

Example: mean-parameter (sheet: "mean"): Here we have two ROIs from one mask-image (represented in rows) and a number of animals (columns).

	A	B	C	D	E	F	G
1	Region	20220112NH_ceAngio_NHE000185	20220112NH_ceAngio_NHE000186	20220113NH_ceAngio_NHE000170	20220113NH_ceAngio_NHE000171	20220113NH_ceAngio_NHE000176	20220113NH_ceAngio_NHE000177
2	EMS_1__ID_1	2721.263509	3225.162436	4042.199138	1800.53239	4863.803108	4852.100298
3	EMS_2__ID_2	3496.009106	4543.16769	4013.795715	2156.445782	4328.689987	4669.272042
4							
5							
6							
7							
8							
9							
10							

info | frequencyMsk | volMsk | frequency | vol | **mean** | std | median | integrDens | min | max

4. Batch

-type **anth** to obtain the code… alternatively you could type: **uhelp(anth)**   or hit the [anth]-button from the ANT-GUI.

- you can than copy and modify the code

```
% =========================================================
%   FUNCTION:         [xgetparameter.m]
%   info :                   [xgetparameter.m] obtain mask-based parameter from images (multi-masks & masks can be mulinary)
% =========================================================
z=[];
z.file              = { 'rc_rCBV.nii' };                          % images used to extract paramters
z.masks             = { 'EMS.nii' };                             % select one/several masks to apply on the image
z.minTreshold       = [];                                        % minimum threshold image value
z.minTresholdReplace = [NaN];                                    % values below "minTreshold" will be replaced by this
z.maxTreshold       = [];                                        % maximum threshold image value
z.maxTresholdReplace = [];                                       % values above "maxTreshold" will be replaced by this
z.outputfolder      = 'H:\Daten-2\Imaging\AG_Hecht\ceAngio_2022\results';   % output folder for resulting EXCEL-file
z.filenamePrefix    = 'EMS_rc_rCBV';                             % enter filename prefix for resulting EXCEL-file
z.addDate           = [0];                                       % at date to filename (suffix)
z.addTime           = [0];                                       % at time to filename (suffix)
xgetparameter(1,z);
```

EM   360/408 #1   ant.m

Here is the code to rerun the task:

```
% % =========================================================
% % #g FUNCTION:         [xgetparameter.m]
% % #b obtain mask-based parameter from images (multi-masks & masks can be mulinary)
% % =========================================================
```

```
z=[];
z.file               = { 'rc_rCBV.nii' };                          % % images used to extract paramters
z.masks              = { 'EMS.nii' };                              % % select one/several masks to apply on the
image
z.minTreshold        = [0];                                        % % minimum threshold image value
z.minTresholdReplace = [NaN];                                      % % values below "minTreshold" will be
replaced by this value or removed if field is empty
z.maxTreshold        = [];                                         % % maximum threshold image value
z.maxTresholdReplace = [];                                         % % values above "maxTreshold" will be
replaced by this value or removed if field is empty
z.outputfolder       = 'H:\Daten-2\Imaging\AG_Hecht\ceAngio_2022\results';   % % output folder for resulting EXCEL-file
z.filenamePrefix     = '_test';                                    % % <optional> enter filename prefix for
resulting EXCEL-file
z.addDate            = [1];                                        % % <optional> at date to filename (suffix)
z.addTime            = [1];                                        % % <optional> at time to filename (suffix)
xgetparameter(1,z);
```