


SynthSeg

This functionality is now available in FreeSurfer (v7.3.2 onwards)

Update September 2022: SynthSeg is also available as part of Matlab (R2022b and onwards); see details below.


Update December 2023: If you have images with white matter lesions and/or acquired at low-field (e.g., with a portable scanner), please try  WMH-SynthSeg.

Author: Benjamin Billot


E-mail: [benjamin.billot \[at\] inria.fr](mailto:benjamin.billot@inria.fr)

Rather than directly contacting the author, please post your questions on this module to the FreeSurfer mailing list at [freesurfer \[at\] nmr.mgh.harvard.edu](mailto:freesurfer@nmr.mgh.harvard.edu)

If you use SynthSeg in your analysis, please cite:

-  SynthSeg: Segmentation of brain MRI scans of any contrast and resolution without retraining. B Billot, DN Greve, O Puonti, A Thielscher, K Van Leemput, B Fischl, AV Dalca, JE Iglesias. Medical Image Analysis, 83, 102789 (2023).

For the robust mode (see below), please cite:

-  Robust machine learning segmentation for large-scale analysis of heterogeneous clinical brain MRI datasets. B Billot, C Magdamo, SE Arnold, S Das, JE Iglesias. PNAS, 120(9), e2216399120 (2023).

Contents

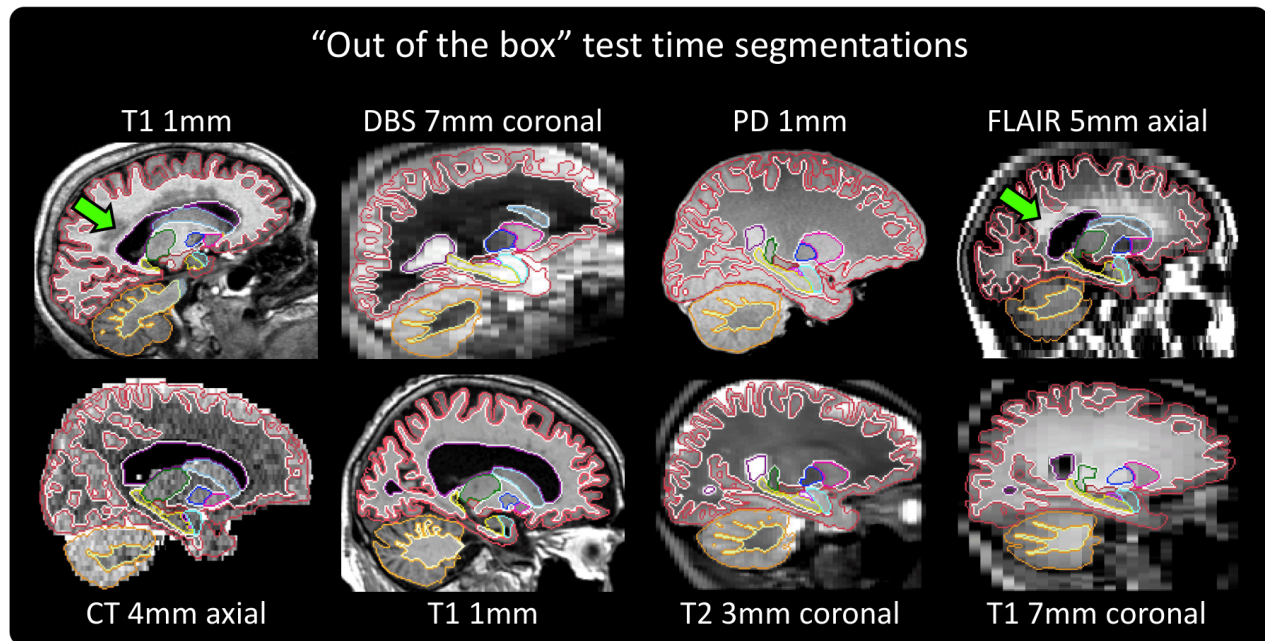
1. General Description
2. Installation
3. Usage
4. Processing CT scans
5. Frequently asked questions (FAQ)
6. Matlab implementation
7. List of segmented structures

1. General Description

This tool implements SynthSeg, the first convolutional neural network for segmentation of brain MRI scans of any contrast and resolution that works out-of-the-box, without retraining or fine-tuning. SynthSeg relies on a single model, which we distribute here. This model is robust to:

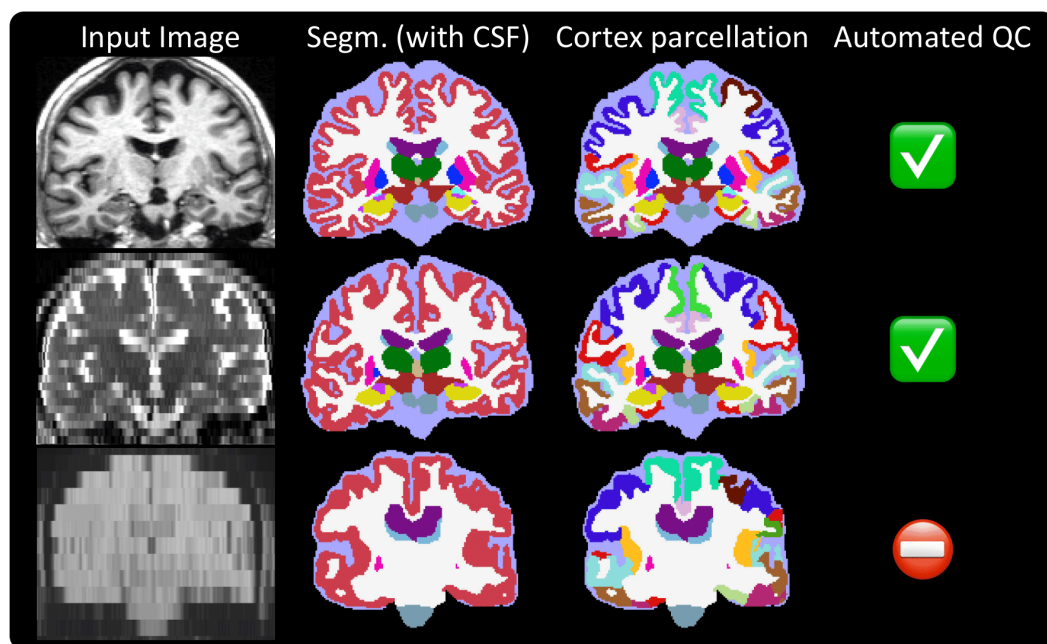
- a wide array of subject populations: from young and healthy to ageing and diseased subjects with strong atrophy,
- white matter lesions (see green arrows in image below),
- scans with or without preprocessing (bias field corruption, skull stripping, intensity normalisation, registration to template).

The output segmentations are returned at high resolution (1mm isotropic), regardless of the resolution of the input scans. The code can run on the GPU (6s per scan) as well as the CPU (2 minutes per scan). The list of segmented structures can be found at the bottom of this page.



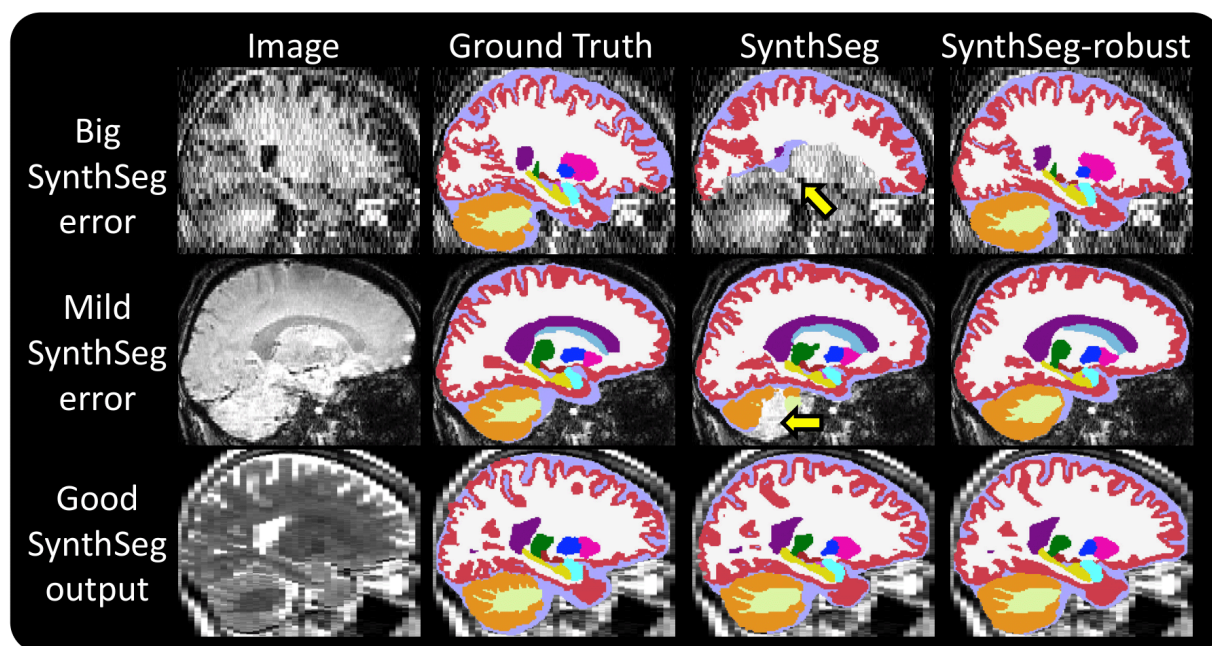
New features (28/06/2022)

SynthSeg 1.0 has now been replaced with SynthSeg 2.0, which offers more functionalities !! These new features are illustrated in the figure below. Specifically, in addition to whole-brain segmentation, SynthSeg is now also able to perform cortical parcellation, automated quality control (QC) of the produced segmentations, and intracranial volume (ICV) estimation computed by including the CSF to the list of segmented structures.



Robust version

While SynthSeg is quite robust, it sometimes falters on scans with low signal-to-noise ratio, or with low tissue contrast (see figure below). For this reason, we developed a new architecture for increased robustness, named "SynthSeg-robust", which can be selected with the `--robust` flag (see Section 3). You can use this mode when SynthSeg gives results like those in the third column (1st and 2nd rows) of the figure below:



See the table below for a summary of the functionalities supported by each version.

	Version 1.0	Version 2.0
Whole-brain segmentation...	✓	✓
...including CSF (for ICV estimation)	✗	✓
Cortical parcellation	✓	✓
Automated QC	✓	✓
Robust variant	✗	✓

2. Installation

The first time you run this module, it will prompt you to install Tensorflow. Simply follow the instructions in the screen to install the CPU or GPU version.

If you have a compatible GPU, you can install the GPU version for faster processing, but this requires installing libraries (GPU driver, Cuda, CuDNN). These libraries are generally required for a GPU, and are not specific for this tool. In fact you may have already installed them. In this case you can directly use this tool without taking any further actions, as the code will automatically run on your GPU.

3. Usage

You can use SynthSeg with the following command:

```
mri_synthseg --i <input> --o <output> [--parc --robust --fast --vol <vol> --qc <qc> --post <post> --resample <resample> --crop <crop> --threads <threads> --cpu --v1 --ct]
```

where:

- *<input>*: path to a scan to segment, or to a folder. This can also be the path to a text file, where each line is the path of an image to segment.
- *<output>*: path where the output segmentations will be saved. This must be the same type as *--i* (i.e., the path to a file, a folder, or a text file where each line is the path to an output segmentation).
- *--parc*: (optional) to perform cortical parcellation in addition to whole-brain segmentation.
- *--robust*: (optional) to use the variant for increased robustness (e.g., when analysing clinical data with large slice spacing). This can be slower than the other model.
- *--fast*: (optional) use this flag to disable some postprocessing operations for faster prediction (approximately twice as fast, but slightly less accurate). This doesn't apply when the *--robust* flag is used.
- *<vol>*: (optional) path to a CSV file where volumes for all segmented regions will be saved. If *--i* designates an image or a folder, *<vol>* must be the path to a single CSV file where the volumes of all subjects will be saved. Otherwise, *<vol>* must be a text file,

where each line is the path to a different CSV file where the volumes of the corresponding subject will be saved.

- `<qc>`: (optional) path to a CSV file where QC scores will be saved. The same formatting requirements apply as for the `--vol` flag.
- `<post>`: (optional) path where the output 3D posterior probabilities will be saved. This must be the same type as `--i` (i.e., the path to a file, a folder, or a text file).
- `<resample>`: (optional) in order to return segmentations at 1mm resolution, the input images are internally resampled (except if they already are at 1mm). Use this optional flag to save the resampled images. This must be the same type as `--i` (i.e., the path to a file, a folder, or a text file).
- `<crop>`: (optional) to crop the inputs to a given shape before segmentation. This must be divisible by 32. It can be given as a single (i.e., `--crop 160`) or several integers (i.e., `--crop 160 128 192`, ordered in RAS coordinates). By default the whole image is processed. Use this flag for faster analysis or to fit in your GPU.
- `<threads>`: (optional) number of threads to be used by Tensorflow (default uses one core). Increase it to decrease the runtime when using the CPU version.
- `--cpu`: (optional) to run on the CPU rather than the GPU.
- `--v1`: (optional) to run the first version of SynthSeg (SynthSeg 1.0).
- `--ct`: (optional) use this flag when processing CT scans (details below).
- `--photo`: (optional) use this flag when processing stacks of 3D reconstructed dissection photos (see PhotoTools). Use `--photo left` to segment left hemispheres, `--photo right` for right hemispheres, and `--photo both` for images with both hemispheres.

We note that `--parc`, `--robust`, `--fast`, `--qc`, `--v1`, as well as **input text files**, are only available on development versions from **June 28th 2022 onwards**.

Important: If you wish to process several scans, we highly recommend that you put them in a single folder, or use input text files, rather than calling SynthSeg individually on each scan. This will save the time required to set up the software for each scan.

4. Processing CT scans

Regarding CT scans: SynthSR does a decent job with CT ! The only caveat is that the dynamic range of CT is very different to that of MRI, so they need to be clipped to [0, 80] Hounsfield units. You can use the `--ct` flag to do this, as long as your image volume is in Hounsfield units. If not, you will have to clip to the Hounsfield equivalent yourself (and not use `--ct`).

5. Frequently asked questions (FAQ)

- **Does running this tool require preprocessing of the input scans?**

No! Because we applied aggressive augmentation during training (see paper), this tool is able to segment both processed and unprocessed data. So there is no need to apply bias field correction, skull stripping, or intensity normalisation.

- **The sum of the number of voxels of a given structure multiplied by the volume of a voxel is not equal to the volume reported in the output volume file.**

This is because the volumes are computed upon a soft segmentation, rather than the discrete segmentation. The same happens with the main recon-all stream: if you compute volumes by

counting voxels in aseg.mgz, you don't get the values reported in aseg.stats.

- **What formats are supported ?**

This tool can be run on Nifti (.nii/.nii.gz) and FreeSurfer (.mgz) scans.

- **How can I increase the speed of the CPU version without using a GPU?**

The first solution is to use the `--fast` flag, which will half the processing time if you're not using the `--robust` flag (gains in speed are much smaller if you are). Next, if you have a multi-core machine, you can increase the number of threads with the `--threads` flag (up to the number of cores). Additionally you can also try to decrease the cropping value, but this will also decrease the field of view of the image.

- **Why are the inputs automatically resampled to 1mm resolution ?**

Simply because, in order to output segmentations at 1mm resolution, the network needs the input images to be at this particular resolution! We actually do not resample images with resolution in the range $[0.95, 1.05]$, which is close enough. We highlight that the resampling is performed internally to avoid the dependence on any external tool.

- **Why aren't the segmentations perfectly aligned with their corresponding images?**

This may happens with viewers other than FreeSurfer's Freeview, if they do not handle headers properly. We recommend using Freeview but, if you want to use another viewer, you may have to use the `--resample` flag to save the resampled images, which any viewer will correctly align with the segmentations.

6. Matlab implementation

Matlab added the non-robust version of SynthSeg to their Medical Imaging Toolbox in version R2022b. They have a fully documented example on how to use it [here](#).

Alternatively, you can download our Matlab script, which you can call with a single line of code: [Download](#). Uncompress the code and type: "help SynthSeg" for instructions.

7. List of segmented structures

Please note that the label values follow the FreeSurfer classification. We emphasise that the structures are given in the same order as they appear in the posteriors, i.e. the first map of the posteriors corresponds to the background, then the second map is associated to the left cerebral white matter, etc.

Labels	Structures
0	Background
2	Left cerebral white matter
3	Left cerebral cortex
4	Left lateral ventricle
5	Left inferior lateral ventricle
7	Left cerebellum white matter
8	Left cerebellum cortex
10	Left thalamus
11	Left caudate
12	Left putamen
13	Left pallidum
14	3 rd ventricle
15	4 th ventricle
16	Brain-stem
17	Left hippocampus
18	Left amygdala
24	CSF (SynthSeg 2.0 only)

Labels	Structures
26	Left accumbens area
28	Left ventral DC
41	Right cerebral white matter
42	Right cerebral cortex
43	Right lateral ventricle
44	Right inferior lateral ventricle
46	Right cerebellum white matter
47	Right cerebellum cortex
49	Right thalamus
50	Right caudate
51	Right putamen
52	Right pallidum
53	Right hippocampus
54	Right amygdala
58	Right accumbens area
60	Right ventral DC

SynthSeg (last edited 2025-01-24 11:49:28 by JuanIglesias)