

# Ironsmith QSM

---

Copyright (C) 2020 Valentinos Zachariou, University of Kentucky (see LICENSE file for more details).  
Third party software provided with Ironsmith are subject to their own licenses and copyrights (see section 7 for details).

**This software has been developed for research purposes only and is not a clinical tool.**

## Description:

Ironsmith is an automated pipeline for creating and processing Quantitative Susceptibility Maps (QSM), extracting QSM based iron concentrations from subcortical and cortical brain regions and evaluating the quality of QSM data using per ROI SNR measures.

Ironsmith can perform the following tasks:

- a) Create QSM maps from GRE DICOM images using MEDI Toolbox **(see section 7 for details)** .
- b) Register MPR or multi-echo MPR (MEMPR) T1 images to QSM maps and then segment these into 89 ROIs **(ROI list in section 8)** using FreeSurfer.
- c) Filter outlier voxels from these ROIs (default: QSM values larger than top 97th percentile of values), extract QSM based iron concentration, and format the output into CSV tables.
- d) Calculate SNR (GRE magnitude image based) for each ROI as a measure of quality control for QSM data and output SNR values in CSV tables.
- e) Non-linearly Warp QSM maps and aligned MPR/MEMPR to MNI152 1mm space for voxelwise QSM analyses.
- f) Process single or multiple participants in parallel (multiple instances and nohup supported).

## 1) Software requirements:

---

### a) Operating system

#### Unix

Any Linux distribution that supports Singularity (see point c).

Ironsmith tested on:

Red Hat Enterprise Linux Workstation release 7.8 (Maipo)

## **Windows 10 WSL2**

Ironsmith tested on:

Ubuntu 18.04 running on Windows 10 Subsystem for Linux V2 (WSL2)

Ubuntu 16.04 running on Windows 10 Subsystem for Linux V2 (WSL2)

## **b) MATLAB**

Ironsmith requires Matlab to run MEDI Toolbox and supports versions R2017b to R2019b.

Matlab is not needed if MEDI is not required.

## **c) Singularity**

Ironsmith tested on Singularity versions 3.5.2 and 3.5.3

Installation guide:

<https://sylabs.io/guides/3.5/admin-guide/installation.html>

## **d) Bash UNIX shell version 4.2.46(2) or later with GNU coreutils**

## **e) MEDI Toolbox version 01/15/2020**

Ironsmith requires MEDI Toolbox if QSM maps need to be generated

Download from:

<http://pre.weill.cornell.edu/mri/pages/qsm.html>

MEDI Toolbox is not required if QSM maps are already available

Currently only MEDI Toolbox version 01/15/2020 is supported

**NOTE: 8.8 GB of free space is required for each instance of IronSmith running (see section #5, parallel processing)**

# **2) Installation:**

---

## **a) Download Ironsmith QSM**

Option 1: download from github

Visit <https://github.com/vzachari/IronSmithQSM>

Click on tags

Click on desired IronsmithQSM version (release notes are displayed by clicking the three dots  )

Click on source code link (zip or tar.gz) to download

**NOTE:** Via this download option the IronSmithQSM folder will be **IronSmithQSM-version#** (Ex. *IronSmithQSM-1.00*).

Option 2: using git

```
git clone https://github.com/vzachari/IronSmithQSM.git && cd IronSmithQSM && git
checkout v1.0
```

**NOTE:** `git checkout v1.0` can be replaced with a different version number. Type `git tag -l` from within the IronSmithQSM folder for a list of available versions.

**b) Download QSM\_Container.simg (8.8GB)**

From: <https://drive.google.com/file/d/1wPdd2Xa0oLV2wwpHneXZ7nIIzB3XoKFb/view?usp=sharing>

Or

From: <https://tinyurl.com/QSMContainer>

**c) Place QSM\_Container.simg in IronSmithQSM/Functions**

**d) Download MEDI Toolbox version 01/15/2020 (~7MB)**

From: <http://pre.weill.cornell.edu/mri/pages/qsm.html>

**e) Unzip MEDI Toolbox (typically MEDI\_toolbox.zip)**

**f) Place MEDI\_toolbox folder (folder with README.m, UPDATES.m etc) into IronSmithQSM/Functions**

**NOTE:** Make sure the MEDI\_toolbox folder in IronSmithQSM/Functions does not have another MEDI\_toolbox folder in it (e.g MEDI\_toolbox/MEDI\_toolbox).

**g) Edit IronSmithQSM/Matlab\_Config.txt with the path to the matlab executable on your system**

(e.g. `/usr/local/MATLAB/R2019b/bin/matlab`)

Supported versions R2017b to R2019b.

**h) Add the IronSmithQSM directory to \$PATH**

Guide: <https://opensource.com/article/17/6/set-path-linux>

## 3) Syntax:

---

**Ironsmith [MyInputFile] [absolute path to output folder]**

*Example: "Ironsmith File.csv /home/data/MyAmazingExp/QSM\_Analysis"*

a) The output folder does not need to exist.

b) The output folder does not need to be empty but Ironsmith will skip any participant specified in MyInputFile that has a corresponding folder inside the output folder.

*Ex. if S0001 is specified in MyInputFile and folder S0001 exists in output folder, S0001 will be skipped.*

c) Absolute path to MyInputFile needs to be provided

(Ex. /home/data/MyAmazingExp/CSVFileVault/File.csv) if MyInputFile is not in current folder.

d) Freesurfer\_Skip is a reserved folder name under output folder and may be used by Ironsmith. See section #5 on optional features below.

## 4) MyInputFile format:

---

a) MyInputFile has to be CSV formatted (entries separated by commas ',').

b) MyInputFile can be created in Excel (MS Windows or MacOS) and saved as a CSV (Comma delimited) file or in a Unix text editor (e.g. Gedit, Atom, Emacs).

c) Each row in MyInputFile corresponds to a different participant.

*see Example\_File.csv in IronSmithQSM folder:*

## If MEDI Toolbox is required to create QSM images/maps:

**Column1** = Subj (nominal subject variable e.g. S0001 or 01 or Xanthar\_The\_Destroyer)

**Column2** = MEDI\_Yes <-- this is case sensitive

**Column3** =

**Either** absolute path to directory with MPR/MEMPR files **OR** absolute path to a single NIFTI (.nii or .nii.gz) MPR/MEMPR file

(e.g. /home/subjects/S01/MPR **OR** /home/subjects/S01/MPR/S01\_MPR.nii.gz)

**If a single NIFTI (.nii or .nii.gz) MPR/MEMPR file is provided:**

File can have any name.

File can have multiple volumes, each corresponding to a different echo (RMS will be calculated).

File can have a single echo/volume.

**If path to directory with MPR/MEMPR files is provided:**

*MPR/MEMPR files in provided folder can be:*

#### a) DICOMS

Only MPR/MEMPR DICOMS should be present in the MPR folder if you want Ironsmith to process DICOMS. If NIFTI files exist together with DICOMS, they will be selected instead (see "b" below).

b) Multiple .nii/.nii.gz files each corresponding to a different echo (will be catenated and RMS will be calculated).

To make sure the correct files are selected, each NIFTI file needs to have \_e# in the file name, where # is the echo number.

(e.g. *S0001\_MEMPR\_e1.nii.gz*, *S0001\_MEMPR\_e2.nii.gz*...)

This is the default **dcm2niix** output format for multiple echos.

c) Single .nii/.nii.gz file with multiple volumes, each corresponding to a different echo (RMS will be calculated).

This single NIFTI file can have any name.

d) Single .nii/.nii.gz file with a single echo/volume.

This can be rms/averaged across echos or just a single echo T1 MPAGE.

This single NIFTI file can have any name and will be used as is.

**Column4** = Absolute path to folder with QSM DICOM files

(e.g. */home/subjects/S01/QSM\_Dicom*)

The QSM DICOM folder must include DICOMS for both GRE magnitude and phase. T2\* DICOMS that are sometimes saved as part of a GRE sequence can be present in the QSM DICOM folder and will be ignored.

Preferably only DICOMS should be present in the QSM DICOM folder. However, Ironsmith can filter out the following filetypes .nii .json .txt .nii.gz .HEAD .BRIK .hdr .img

All 4 columns need to be provided, otherwise Ironsmith will exit with errors.

## If QSM maps and GRE magnitude images are already available:

**Column1** = Subj (nominal subject variable e.g. S0001 or 01 or Xanthar\_The\_Destroyer)

**Column2** = MEDI\_No <-- This is case sensitive

**Column3** =

**Either** absolute path to directory with MPR/MEMPR files **OR** absolute path to a single NIFTI (.nii or .nii.gz) MPR/MEMPR file

(e.g. */home/subjects/S01/MPR* **OR** */home/subjects/S01/MPR/S01\_MPR.nii.gz*)

### **If a single NIFTI (.nii or .nii.gz) MPR/MEMPR file is provided:**

File can have any name.

File can have multiple volumes, each corresponding to a different echo (RMS will be calculated).

File can have a single echo/volume.

### **If path to directory with MPR/MEMPR files is provided:**

*MPR/MEMPR files in provided folder can be:*

#### **a) DICOMS**

Only MPR/MEMPR DICOMS should be present in the MPR folder if you want Ironsmith to process DICOMS. If NIFTI files exist together with DICOMS, they will be selected instead (see "b" below).

b) Multiple .nii/.nii.gz files each corresponding to a different echo (will be catenated and RMS will be calculated).

To make sure the correct files are selected, each NIFTI file needs to have `_e#` in the file name, where `#` is the echo number.

(e.g. `S0001_MEMPR_e1.nii.gz`, `S0001_MEMPR_e2.nii.gz...`)

This is the default **dcm2niix** output format for multiple echos.

c) Single .nii/.nii.gz file with multiple volumes, each corresponding to a different echo (RMS will be calculated).

This single NIFTI file can have any name.

d) Single .nii/.nii.gz file with a single echo/volume.

This can be rms/averaged across echos or just a single echo T1 MPAGE.

This single NIFTI file can have any name and will be used as is.

**Column4** = Absolute path including filename to QSM magnitude image

(e.g. `/home/subjects/S01/QSM/QSM_Magnitude.nii.gz`)

**Column5** = Absolute path including filename to QSM map

(e.g. `/home/subjects/S01/QSM/QSM_Map.nii.gz`)

All 5 columns need to be provided, otherwise Ironsmith will exit with errors.

## **5) Optional features:**

---

### **Skipping FreeSurfer segmentation**

If FreeSurfer has already run and a participant has a completed FreeSurfer recon-all -all segmentation folder, Ironsmith can skip the FreeSurfer segmentation step by doing the following:

a) Copy the FreeSurfer recon-all folder (the one containing the *label*, *mri*, *scripts*, *stats*, *surf*... folders) into **/OutputFolder/Freesurfer\_Skip**, where **OutputFolder** is the one specified/to be specified in the Ironsmith command.

You can create the /OutputFolder/Freesurfer\_Skip folder or Ironsmith will create it for you if you have run it at least once previously for OutputFolder.

b) Rename the recon-all folder to **Subj\_FreeSurfSeg\_Skull**. Subj should match the one provided in MyInputFile and should correspond to the participant you want the segmentation step skipped.

**Note:** if Ironsmith runs FreeSurfer, it will create **Subj\_FreeSurfSeg\_Skull** and place it under **/OutputFolder/Subj/MPR**. This helps reduce processing time if for any reason one would like to repeat the analysis on a given participant (e.g. due to crash or errors). Just copy/move this folder over to **/OutputFolder/Freesurfer\_Skip**, delete the problematic participant folder (e.g. **/OutputFolder/Subj**) and re-run Ironsmith.

## Processing participants in parallel

Parallel processing can significantly increase the speed of analyses. Running the Ironsmith command with the same MyInputFile and output folder in multiple terminal windows allows for parallel processing of participants specified in MyInputFile.

*For example, running three instances of Ironsmith:*

Terminal 1:

```
Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis
```

Terminal 2:

```
Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis
```

Terminal 3:

```
Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis
```

Terminals 1-3 will each be running a different instance of Ironsmith (each working on a different set of participants) but all instances will be working on the same group/list of participants (from **File.csv**) and in the same output folder (**/home/data/MyAmazingExp/QSM\_Analysis**) and will only create a single set of group output files (see section #6 below).

**NOTE:** nohup can also be used with parallel processing:

For example, running three **nohup** instances of Ironsmith:

Terminal 1:

```
bash (press enter to switch to bash)
```

```
nohup Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis &> Ironsmith_Inst_1.txt &
```

```
nohup Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis &> Ironsmith_Inst_2.txt &
```

```
nohup Ironsmith File.csv /home/data/MyAmazingExp/QSM_Analysis &> Ironsmith_Inst_3.txt &
```

To monitor the nohup progress of an Ironsmith instance, locate the Ironsmith\_Inst\_#.txt file (in directory where the nohup command was executed) and use the following command:

```
tail -f Ironsmith_Inst_#.txt
```

ctrl+c exits the tail -f process.

## Viewing output NIFTI files

If you do not have a NIFTI viewer, AFNI can be launched from within the QSM\_Container.simg by using the *Ironsmith\_AFNI* command. Just type `Ironsmith_AFNI` from within the folder you would like to view NIFTI files from.

AFNI viewer documentation:

[https://afni.nimh.nih.gov/pub/dist/edu/latest/afni\\_handouts/afni03\\_interactive.pdf](https://afni.nimh.nih.gov/pub/dist/edu/latest/afni_handouts/afni03_interactive.pdf)

## 6) Outputs:

---

Each participant processed by Ironsmith will have a corresponding folder in **OutputFolder**. For example, if "S0001" is one of the participants processed, then **OutputFolder/S0001** will be created and populated with data.

a) All FreeSurfer based masks/ROIs are placed under:

**S0001/QSM/FreeSurf\_QSM\_Masks/Cort\_Mask\_AL\_QSM\_RS\_Erx1**

**S0001/QSM/FreeSurf\_QSM\_Masks/SubC\_Mask\_AL\_QSM\_RS\_Erx1**

b) All QSM maps/images created are placed under S0001/QSM/FreeSurf\_QSM\_Masks and are labelled as:

**Subj\_QSM\_Map\_FSL.nii.gz** <-- Whole brain CSF, segmented by MEDI Toolbox, as the QSM reference structure (default MEDI)

**Subj\_QSM\_Map\_New\_CSF\_FSL.nii.gz** <-- Lateral ventricles as the QSM reference structure

**Subj\_QSM\_Map\_New\_WM\_FSL.nii.gz** <-- White matter as the QSM reference structure



c) All QSM maps warped to MNI space are placed under

**S0001/QSM/FreeSurf\_QSM\_Masks/MNI152\_QSM**

d) QSM per ROI means (89 ROIs) are under **OutputFolder/Group** as follows:

**Group\_QSM\_Mean.csv**

**Group\_QSM\_Mean\_CSF.csv**

**Group\_QSM\_Mean\_WM.csv**

**Group\_QSM\_ADJ\_Mean.csv**

**Group\_QSM\_ADJ\_Mean\_CSF.csv**

**Group\_QSM\_ADJ\_Mean\_WM.csv**

**Group\_QSM\_SNR.csv**

**\_Mean** = Using only positive QSM voxels

**\_ADJ\_Mean** = Using only positive QSM voxels and adjusting for ROI size (*sum of all positive QSM voxels / Number of all voxels within an ROI*)

**\_CSF** = Lateral ventricles as the QSM reference structure

**\_WM** = White matter as the QSM reference structure

**NOTE:** For each ROI, only QSM voxels with values less than the 97th percentile of all positive QSM values are included in averages. This percentile cutoff point for outliers can be modified by manually editing the header of the **05\_Extract\_QSM.sh** script file (line 38) under the Ironsmith installation folder:

```
#Percentile cutoff for outlier removal. Edit Percnt variable to change outlier cutoff  
Percnt="97"
```

SNR is calculated as follows:

mean signal intensity of magnitude image within an ROI / standard deviation of magnitude signal outside the head (away from the frequency and phase axes).

Lastly, SNR is multiplied by the Rayleigh distribution correction factor  $\sqrt{2-\pi/2}$ .

The outside of the head mask used for SNR can be found here:

**/QSM\_Analysis/S0001/QSM/FreeSurf\_QSM\_Masks/Subj\_QSM\_Mag\_FSL\_rms\_OH\_Mask.nii.gz**

## 7) Ironsmith uses the following software, provided in the form of a Singularity image:

---

## AFNI

RW Cox. AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages. Computers and Biomedical Research, 29:162-173, 1996.

RW Cox and JS Hyde. Software tools for analysis and visualization of FMRI Data. NMR in Biomedicine, 10:171-178, 1997.

S Gold, B Christian, S Arndt, G Zeien, T Cizadlo, DL Johnson, M Flaum, and NC Andreasen. Functional MRI statistical software packages: a comparative analysis. Human Brain Mapping, 6:73-84, 1998.

## dcm2niix

Li, Xiangrui, et al. "The first step for neuroimaging data analysis: DICOM to NIFTI conversion." Journal of neuroscience methods 264 (2016): 47-56.

## FreeSurfer

Dale, A.M., Fischl, B., Sereno, M.I., 1999. Cortical surface-based analysis. I. Segmentation and surface reconstruction. Neuroimage 9, 179-194.

Dale, A.M., Sereno, M.I., 1993. Improved localization of cortical activity by combining EEG and MEG with MRI cortical surface reconstruction: a linear approach. J Cogn Neurosci 5, 162-176.

Desikan, R.S., Segonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., Albert, M.S., Killiany, R.J., 2006. An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. Neuroimage 31, 968-980.

## FSL

M.W. Woolrich, S. Jbabdi, B. Patenaude, M. Chappell, S. Makni, T. Behrens, C. Beckmann, M. Jenkinson, S.M. Smith. Bayesian analysis of neuroimaging data in FSL. NeuroImage, 45:S173-86, 2009

S.M. Smith, M. Jenkinson, M.W. Woolrich, C.F. Beckmann, T.E.J. Behrens, H. Johansen-Berg, P.R. Bannister, M. De Luca, I. Drobnjak, D.E. Flitney, R. Niazy, J. Saunders, J. Vickers, Y. Zhang, N. De

Stefano, J.M. Brady, and P.M. Matthews. Advances in functional and structural MR image analysis and implementation as FSL. NeuroImage, 23(S1):208-19, 2004

M. Jenkinson, C.F. Beckmann, T.E. Behrens, M.W. Woolrich, S.M. Smith. FSL. NeuroImage, 62:782-90, 2012

## 8) ROI List:

---

L\_ = Left hemisphere

R\_ = Right hemisphere

LR\_ = Bilateral

\_GM = Gray matter

LR\_Frontal\_Lobe\_GM

LR\_Parietal\_Lobe\_GM

LR\_Occipital\_Lobe\_GM

LR\_Temporal\_Lobe\_GM

L\_CaudalAnteriorCingulate\_GM

L\_CaudalMiddleFrontal\_GM

L\_Cuneus\_GM

L\_DLPFC\_GM

L\_Entorhinal\_GM

L\_Frontal\_GM

L\_Fusiform\_GM

L\_InferiorParietal\_GM

L\_AngularGyrus\_GM

L\_InferiorTemporal\_GM

L\_Insula\_GM

L\_IsthmusCingulate\_GM

L\_LateralOccipital\_GM

L\_LateralOrbitofrontal\_GM

L\_Lingual\_GM

L\_MedialOrbitofrontal\_GM

L\_MiddleTemporal\_GM

L\_Occipital\_GM\_Mask

L\_Parietal\_GM\_Mask

L\_Temporal\_GM\_Mask

L\_Parahippocampal\_GM

L\_Pericalcarine\_GM  
L\_Postcentral\_GM  
L\_PosteriorCingulate\_GM  
L\_Precentral\_GM  
L\_Precuneus\_GM  
L\_RostalMiddleFrontal\_GM  
L\_RostralAnteriorCingulate\_GM  
L\_SuperiorFrontal\_GM  
L\_SuperiorParietal\_GM  
L\_SuperiorTemporal\_GM  
L\_TransverseTemporal\_GM  
R\_CaudalAnteriorCingulate\_GM  
R\_CaudalMiddleFrontal\_GM  
R\_Cuneus\_GM  
R\_DLPFC\_GM  
R\_Entorhinal\_GM  
R\_Frontal\_GM\_Mask  
R\_Fusiform\_GM  
R\_InferiorParietal\_GM  
R\_AngularGyrus\_GM  
R\_InferiorTemporal\_GM  
R\_Insula\_GM  
R\_IsthmusCingulate\_GM  
R\_LateralOccipital\_GM  
R\_LateralOrbitofrontal\_GM  
R\_Lingual\_GM  
R\_MedialOrbitofrontal\_GM  
R\_MiddleTemporal\_GM  
R\_Occipital\_GM\_Mask  
R\_Parietal\_GM\_Mask  
R\_Temporal\_GM\_Mask  
R\_Parahippocampal\_GM  
R\_Pericalcarine\_GM  
R\_Postcentral\_GM  
R\_PosteriorCingulate\_GM  
R\_Precentral\_GM  
R\_Precuneus\_GM  
R\_RostalMiddleFrontal\_GM

R\_RostralAnteriorCingulate\_GM

R\_SuperiorFrontal\_GM

R\_SuperiorParietal\_GM

R\_SuperiorTemporal\_GM

R\_TransverseTemporal\_GM

LR\_Accumbens\_area

LR\_Amygdala

LR\_Caudate

LR\_Hipp

LR\_Pallidum

LR\_Putamen

LR\_Thalamus\_Proper

L\_Accumbens\_area

L\_Amygdala

L\_Caudate

L\_Hipp

L\_Pallidum

L\_Putamen

L\_Thalamus\_Proper

R\_Accumbens\_area

R\_Amygdala

R\_Caudate

R\_Hipp

R\_Pallidum

R\_Putamen

R\_Thalamus\_Proper