

# Software tools motivated by analysis of fMRI data

**Kendrick Kay**

<http://cvnlab.net>

Center for Magnetic Resonance Research (CMRR)

University of Minnesota, Twin Cities



UNIVERSITY  
OF MINNESOTA



# Introduction

- **Where I am**

- Center for Magnetic Resonance Research, University of Minnesota

- **What I work on**

- Computational models of visual processing
- Object and form vision

- **Approach**

- Cognitive neuroscience (experiments, fMRI)
- Theoretical neuroscience (modeling)
- Data analysis (stats, programming)

- **Resources**

- <http://cvnlab.net>



<http://www.cmrr.umn.edu>

# Computing approach

- Pull bits and pieces from:
  - FreeSurfer
  - SPM
  - FSL
- Integrate into MATLAB pipelines
  - Some standalone MATLAB toolboxes (GLMdenoise, analyzePRF, etc.)
- Analysis is done mostly on a large workstation, using cluster computing for parallel analysis of individual voxels

<http://github.com/kendrickkay/>

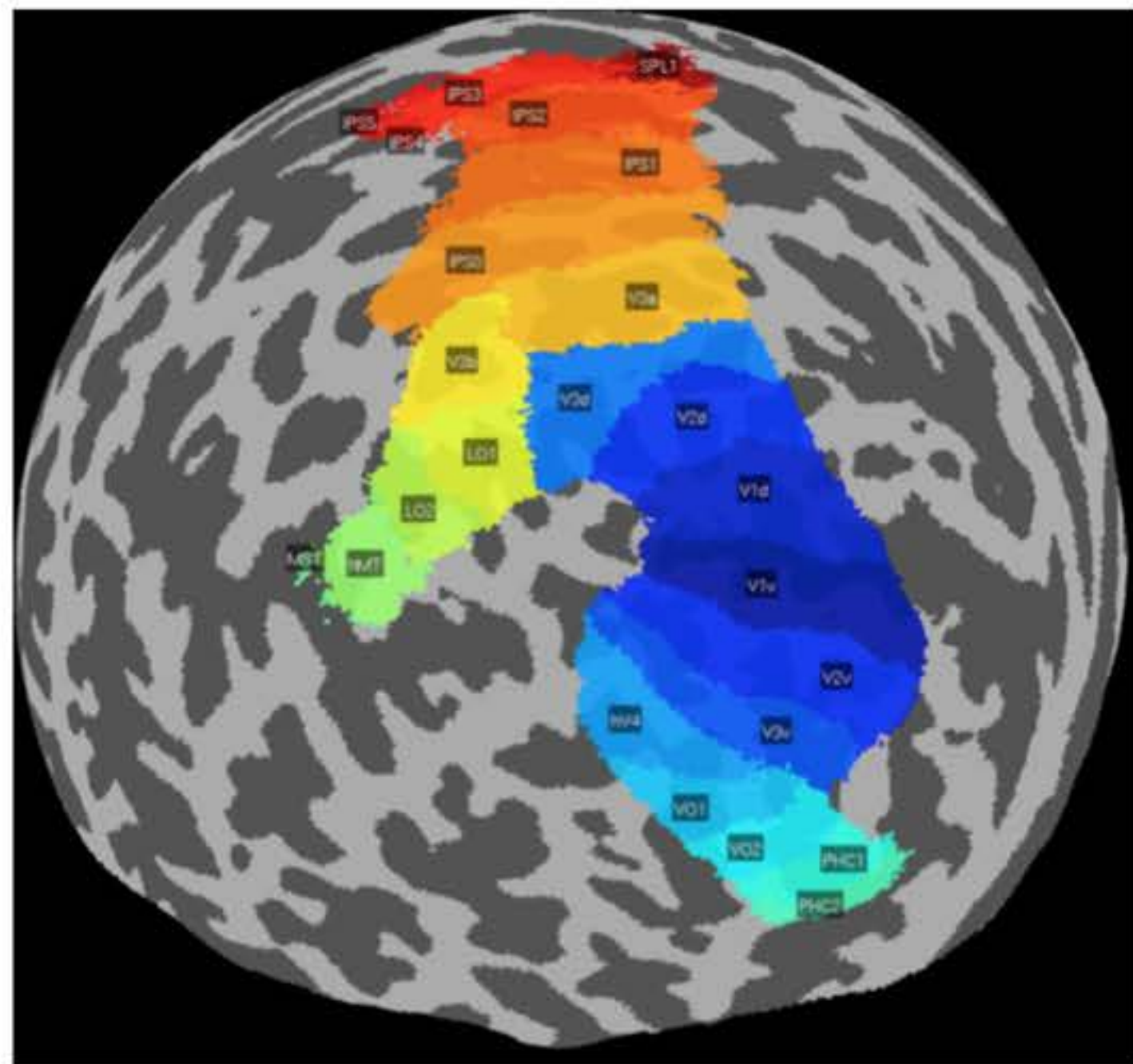


# Automated surface visualizations (1/7)

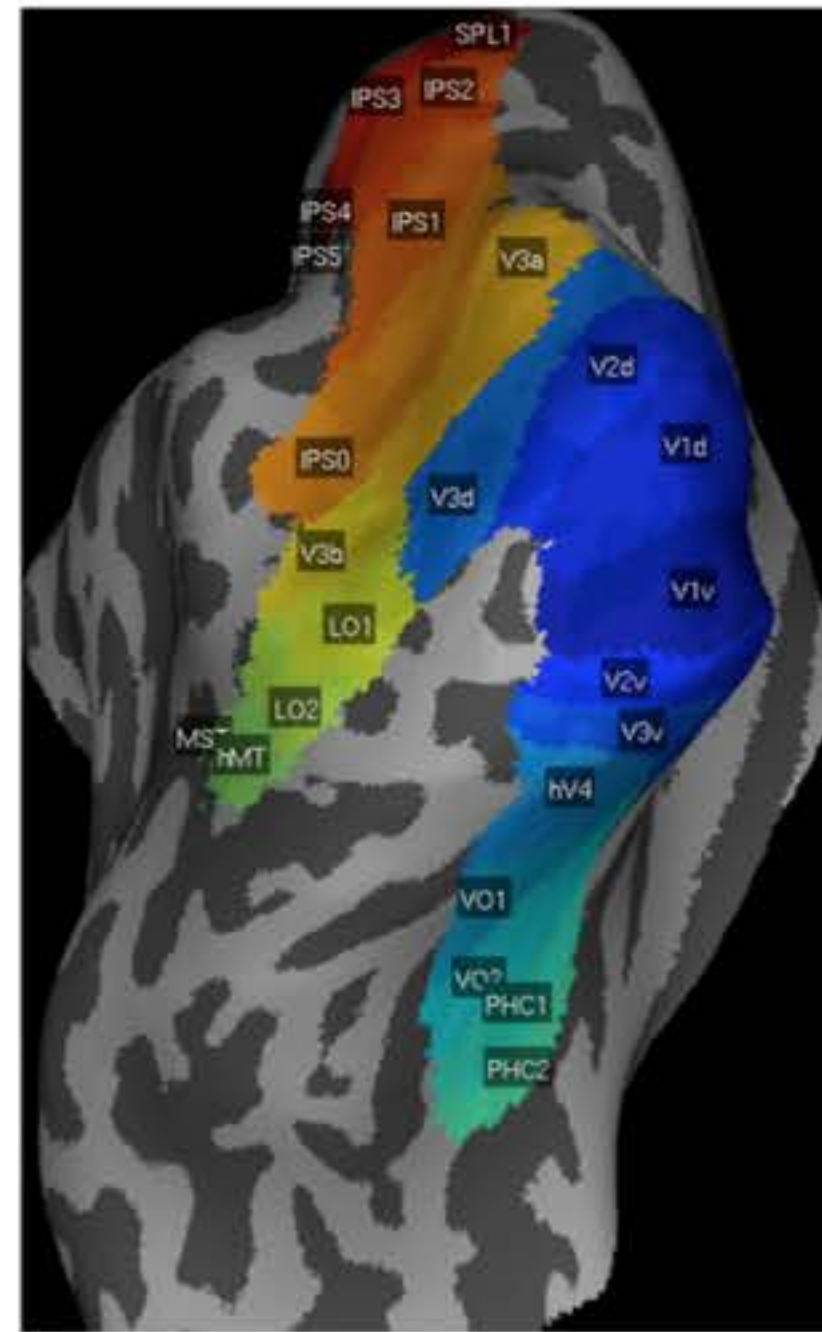
- High-throughput (avoid GUI, automated)
- Customizable (colormap, overlays, etc.)
- FreeSurfer-oriented but could be generalized
- Support for high-resolution surfaces
- Method: map pixels to vertices using nearest-neighbor interpolation, use caching mechanism for speed



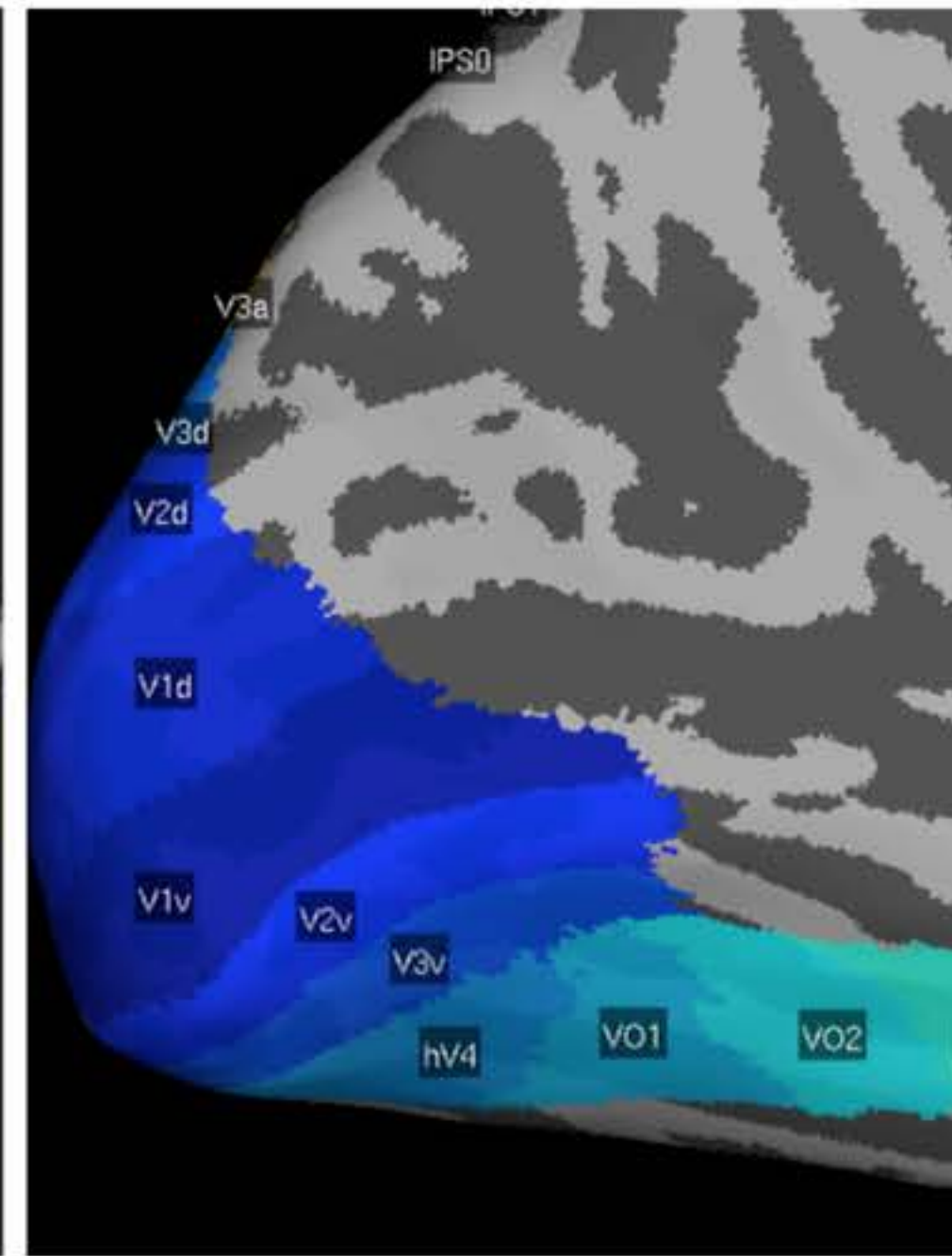
# Automated surface visualizations (1/7)



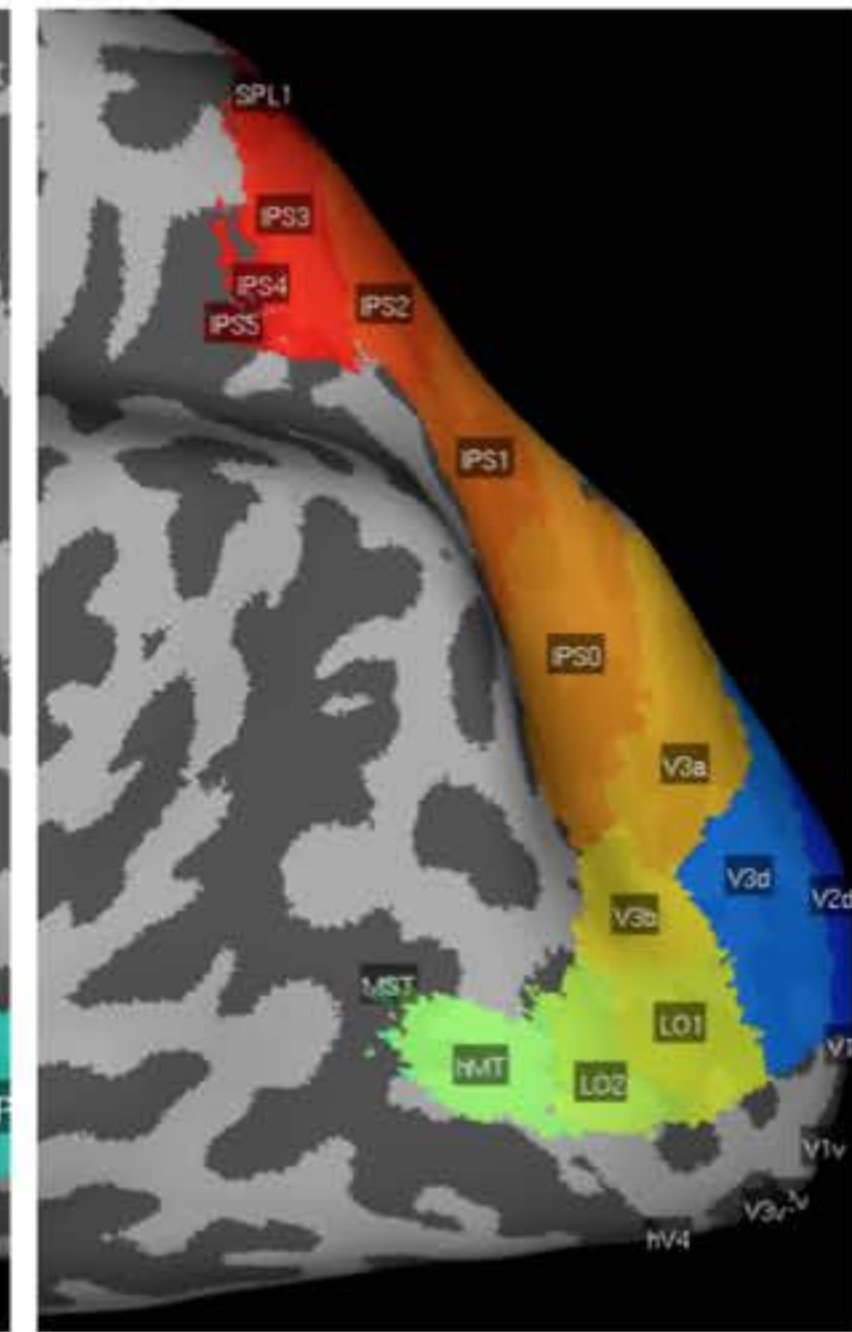
sphere



occipital



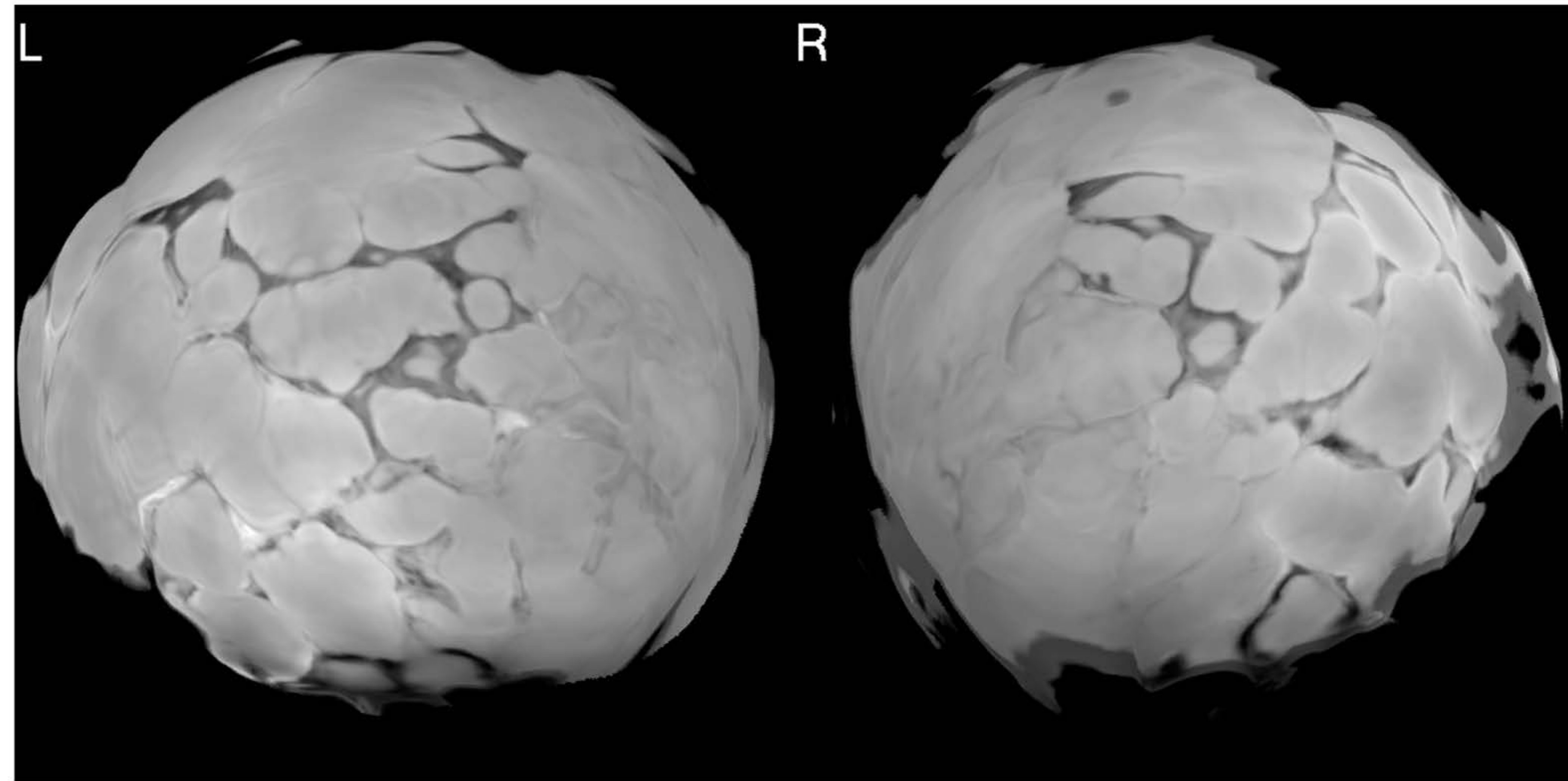
medial



lateral

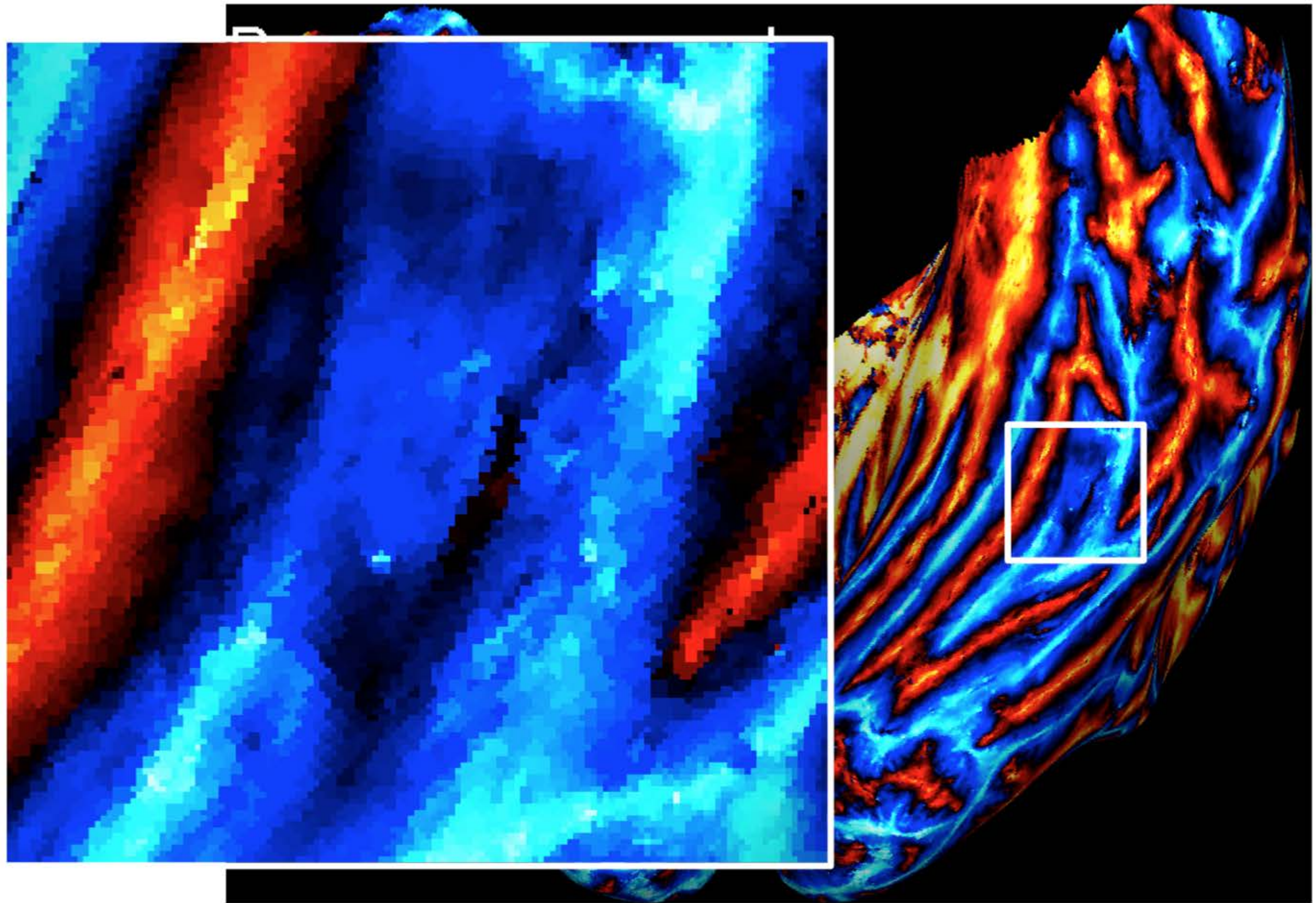


# Automated surface visualizations (1/7)



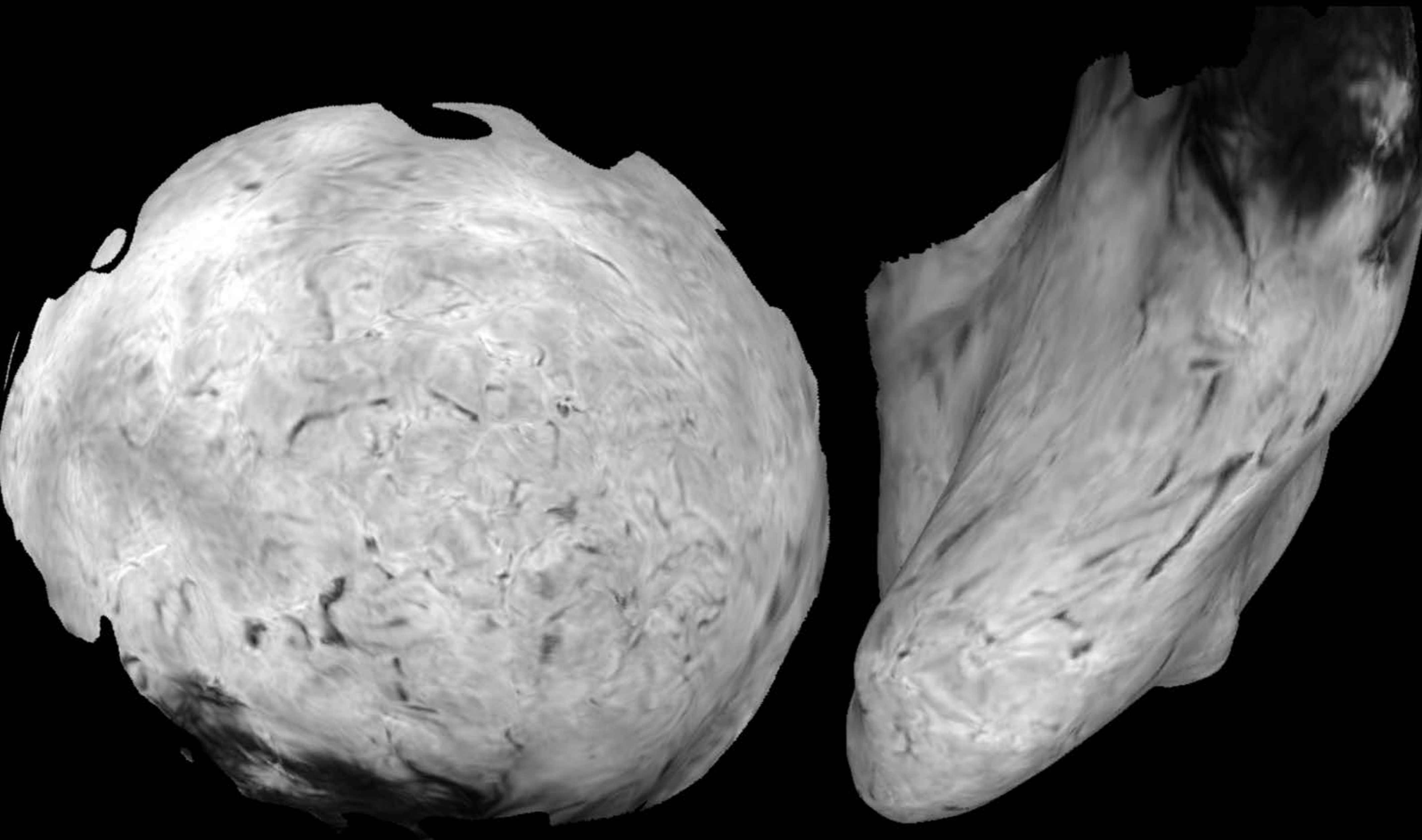


# Automated surface visualizations (1/7)



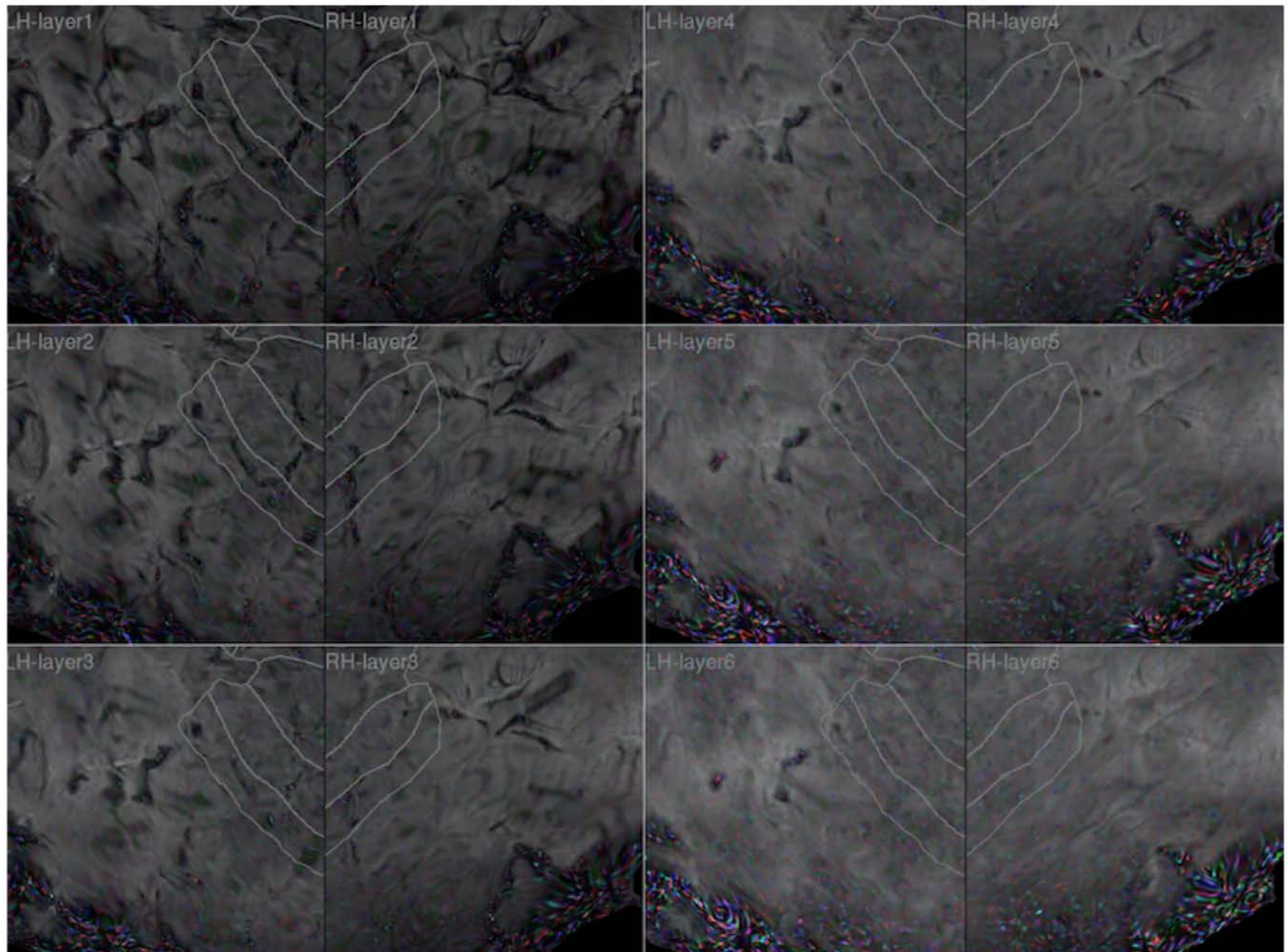


# Automated surface visualizations (1/7)





# Automated surface visualizations (1/7)



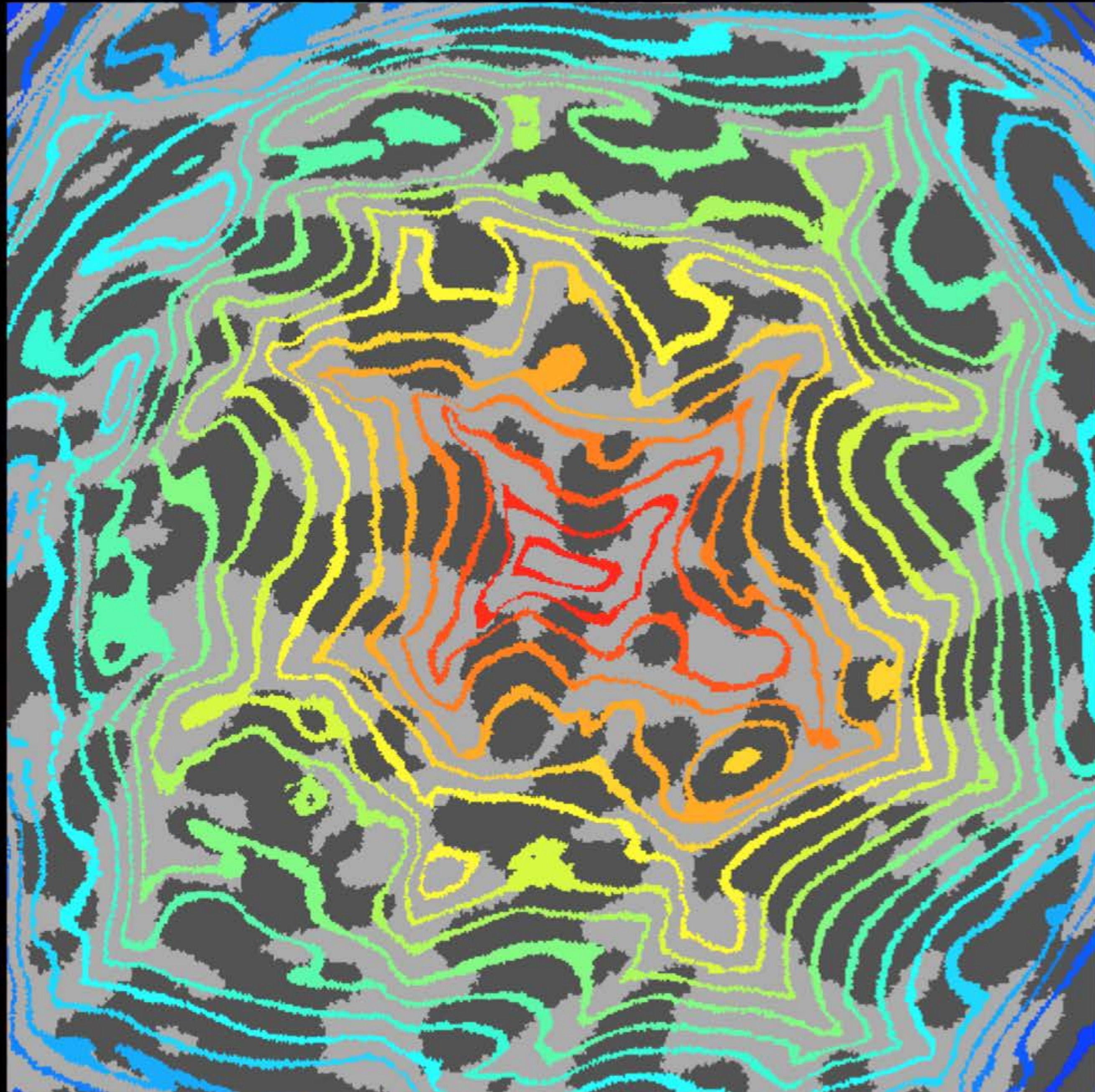


# Brain art (2/7)

- Real data masquerading as art!



# Brain art (2/7)





# Brain art (2/7)



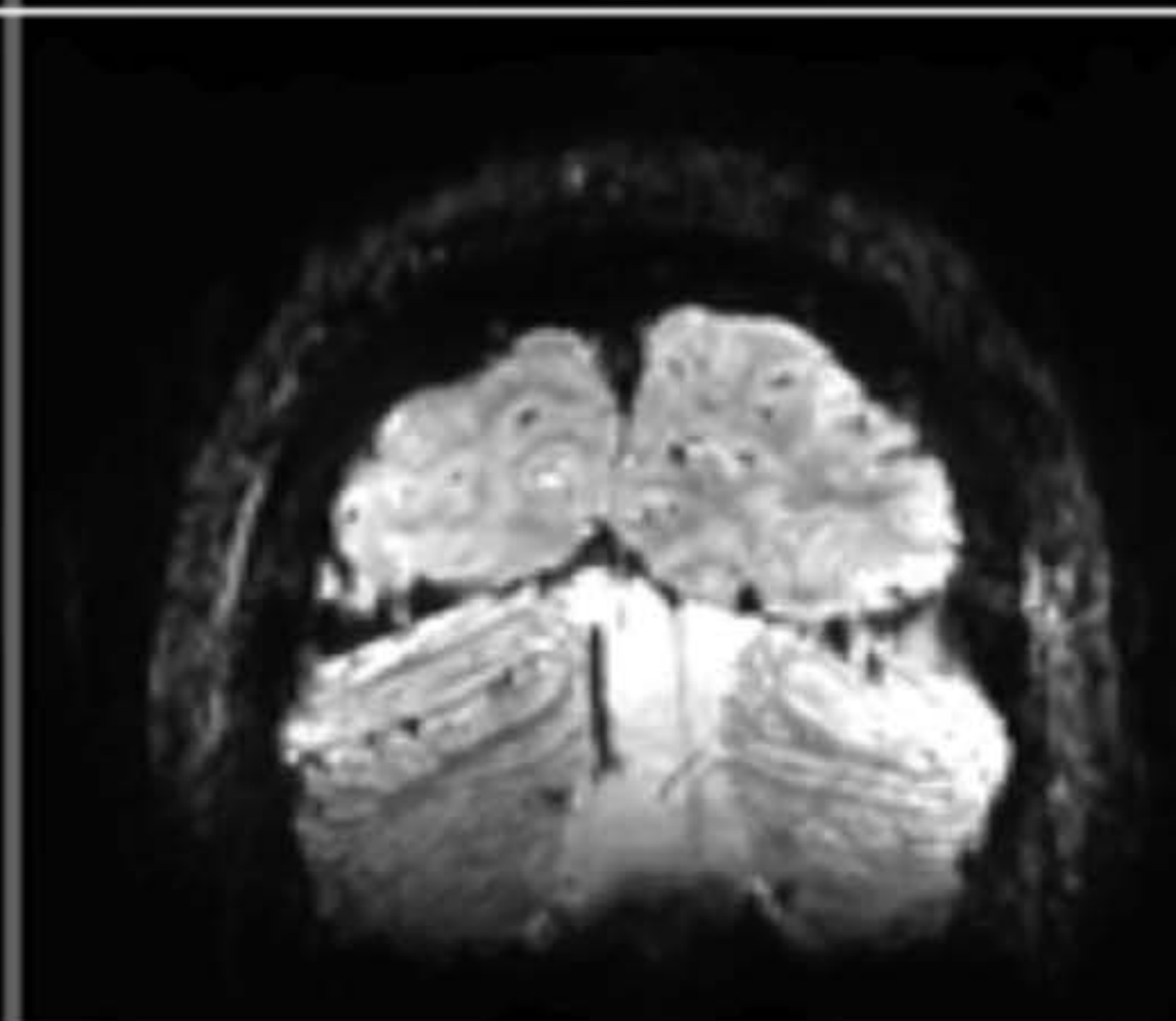
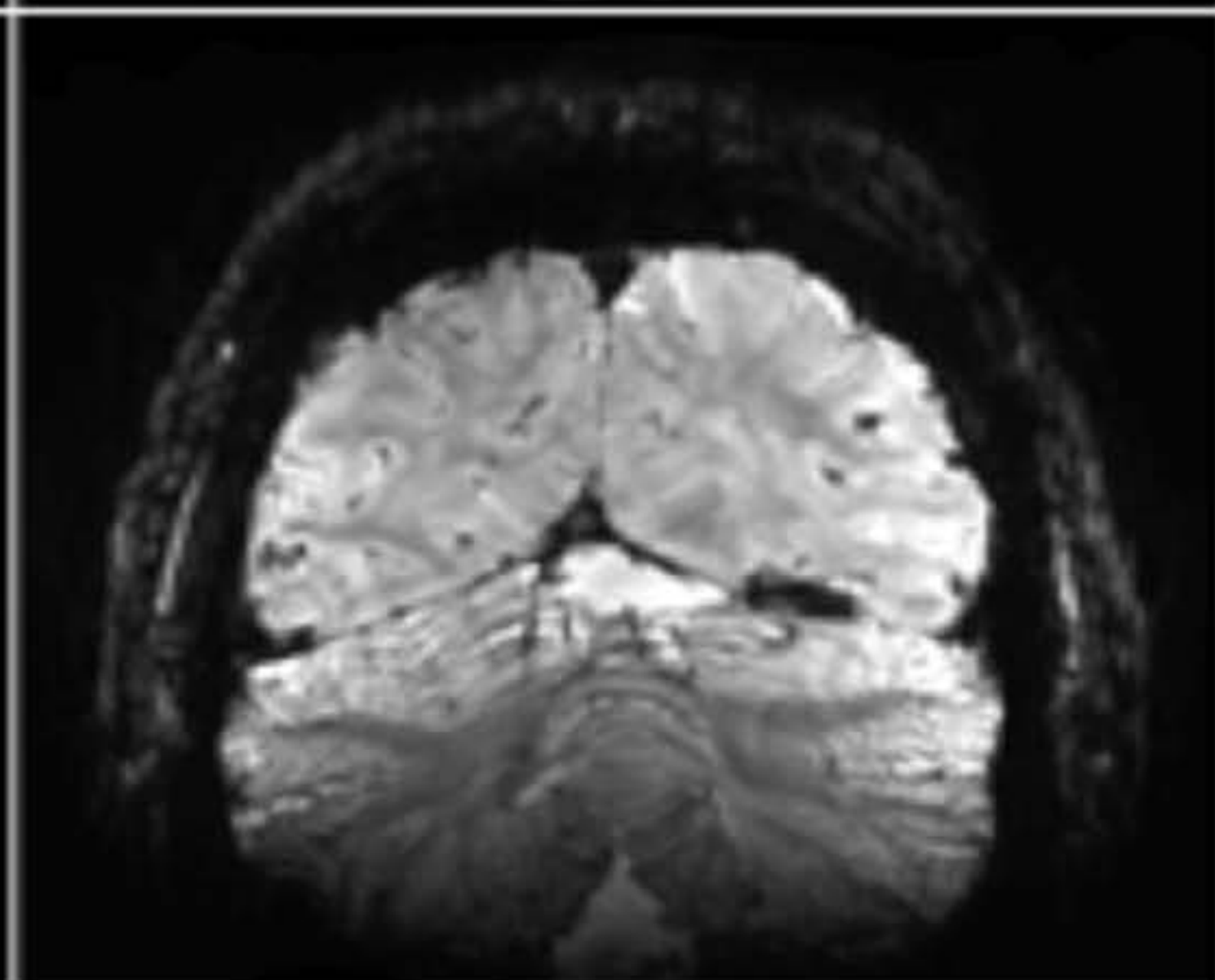
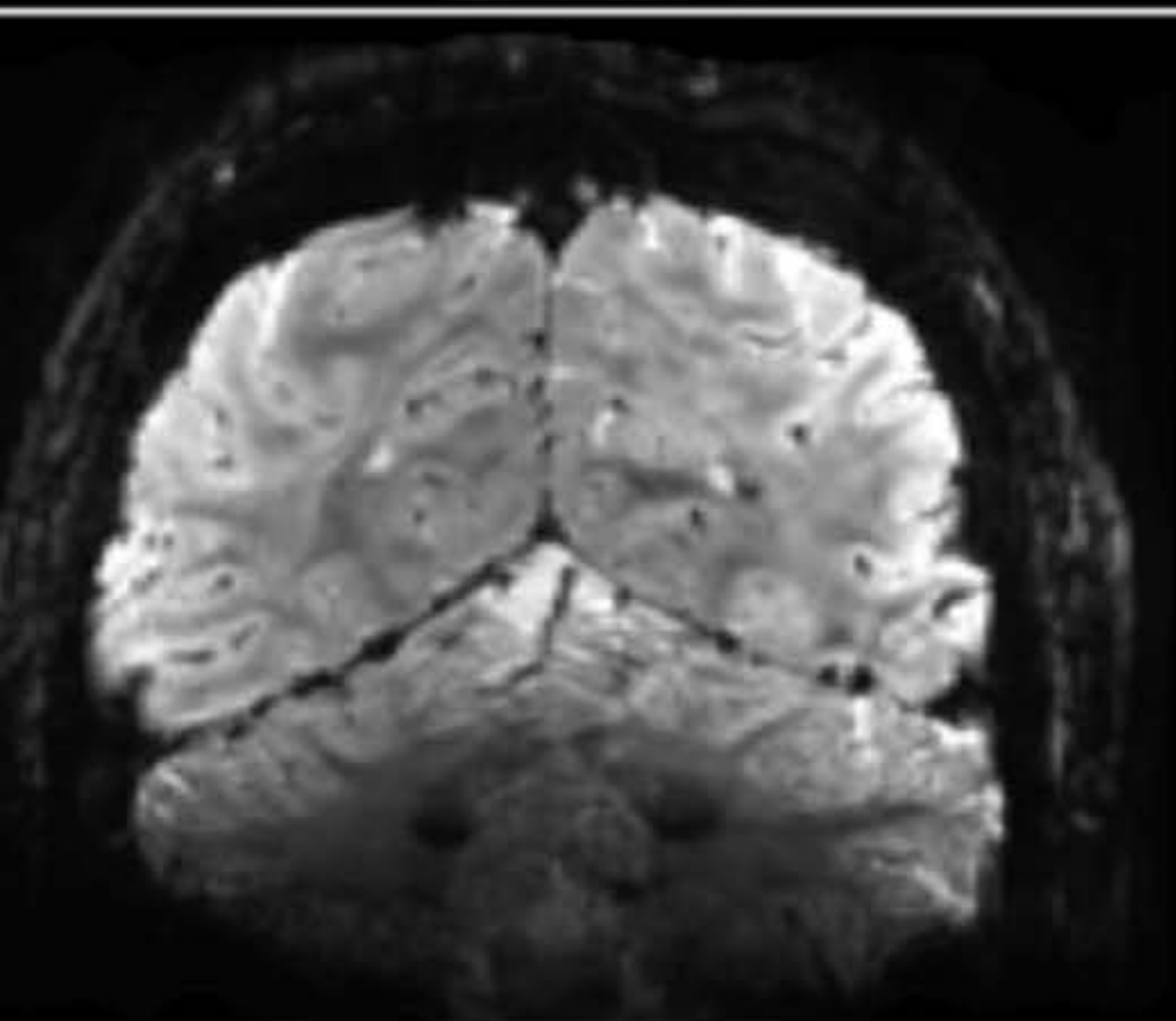
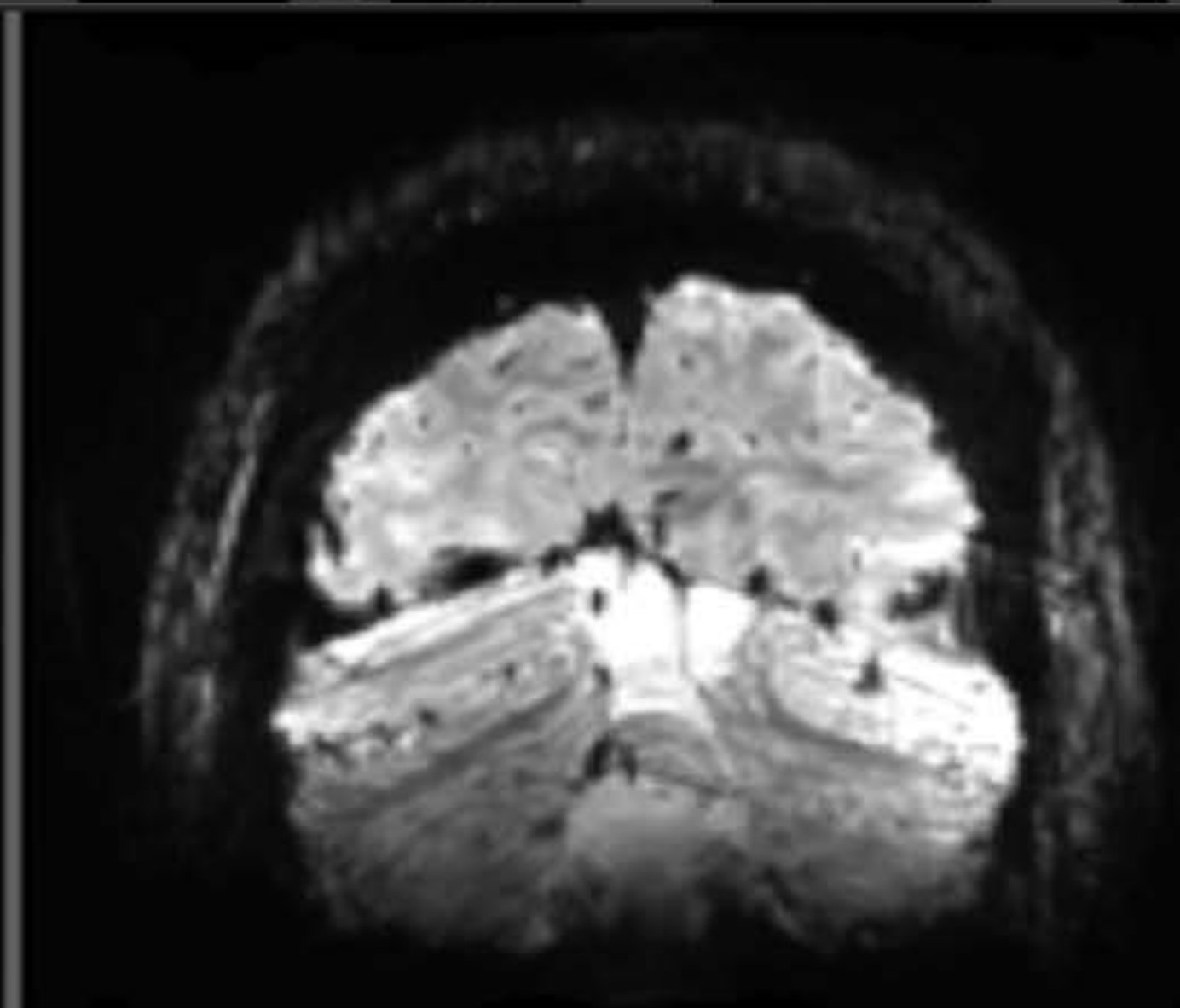
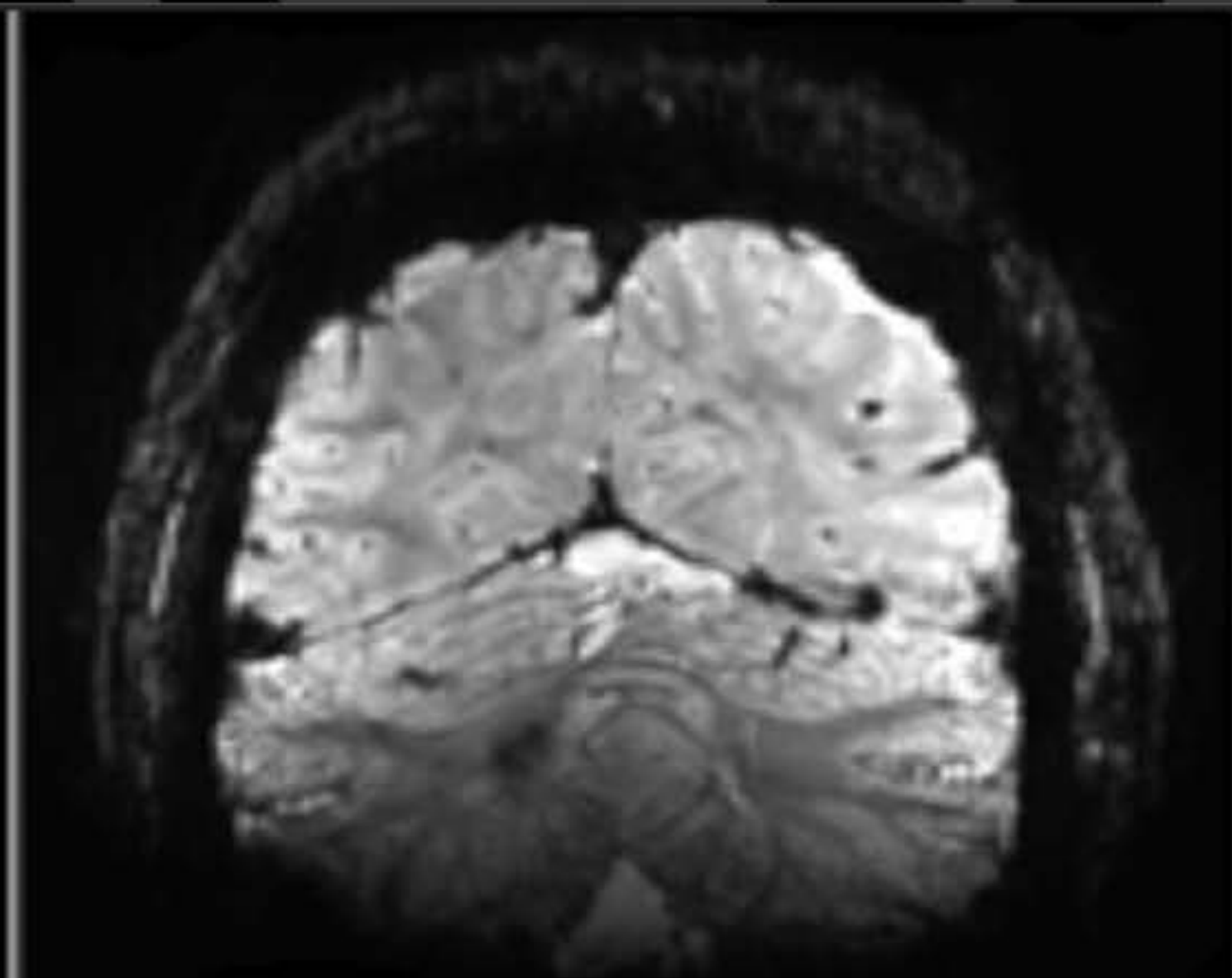
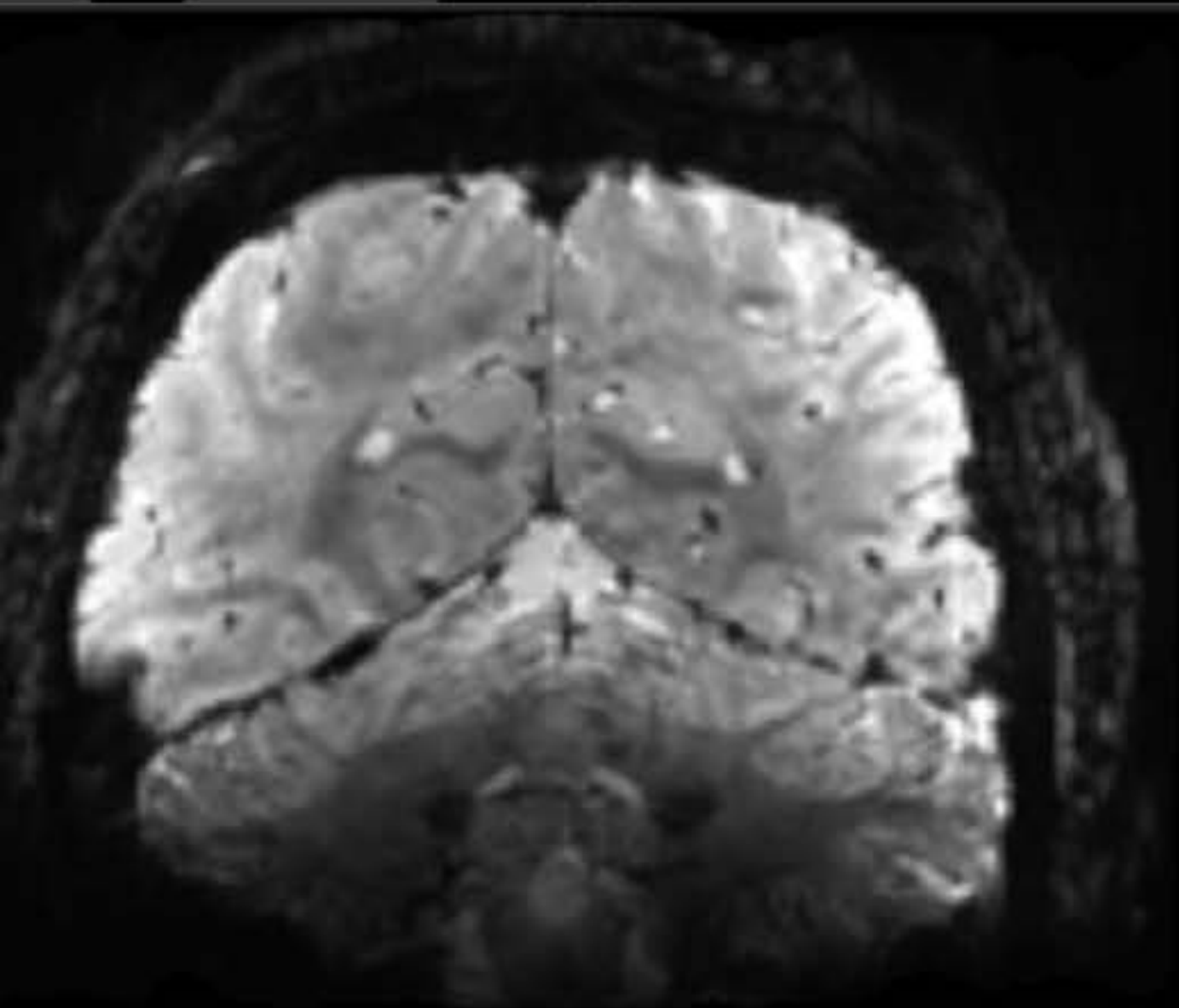


# High-res fMRI pre-processing (3/7)

- Average T1s to improve SNR <http://github.com/kendrickkay/cvncode/>
- Co-registration: T1 to T1, T2 to T2, T2 to T1, EPI to T2
- Generate FreeSurfer cortical surfaces  
(dense, equidistant layers, truncated)
- Fieldmaps:
  - Regularize using local linear smoothing
  - Interpolate over time
- EPI:
  - Slice time correction
  - Motion correction
  - Fieldmap undistortion
  - Interpolation onto cortical surfaces
  - Total: 1 temporal resampling, 1 spatial resampling
- Homogenization of EPI intensities  
(polynomial basis functions, surface-based)

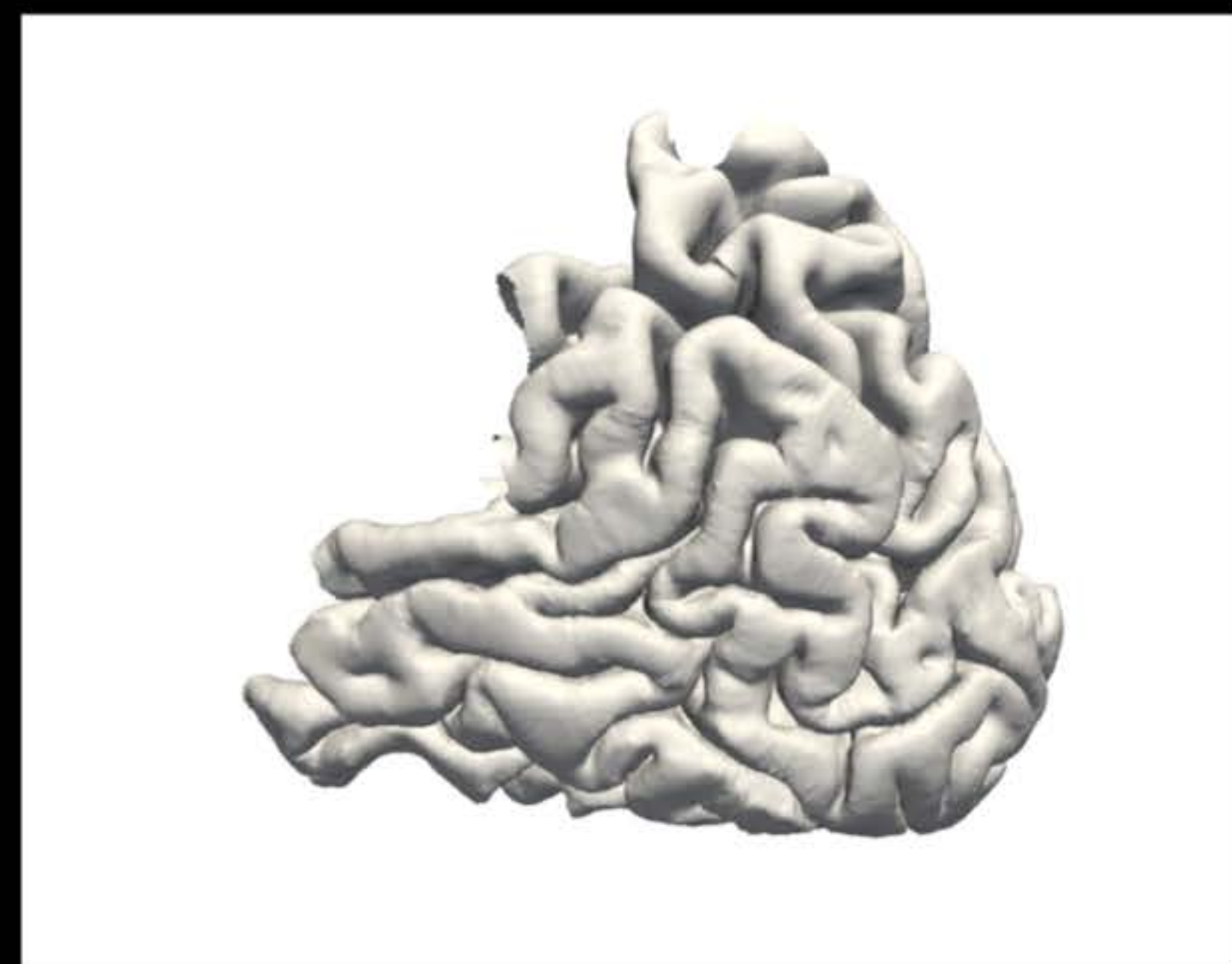
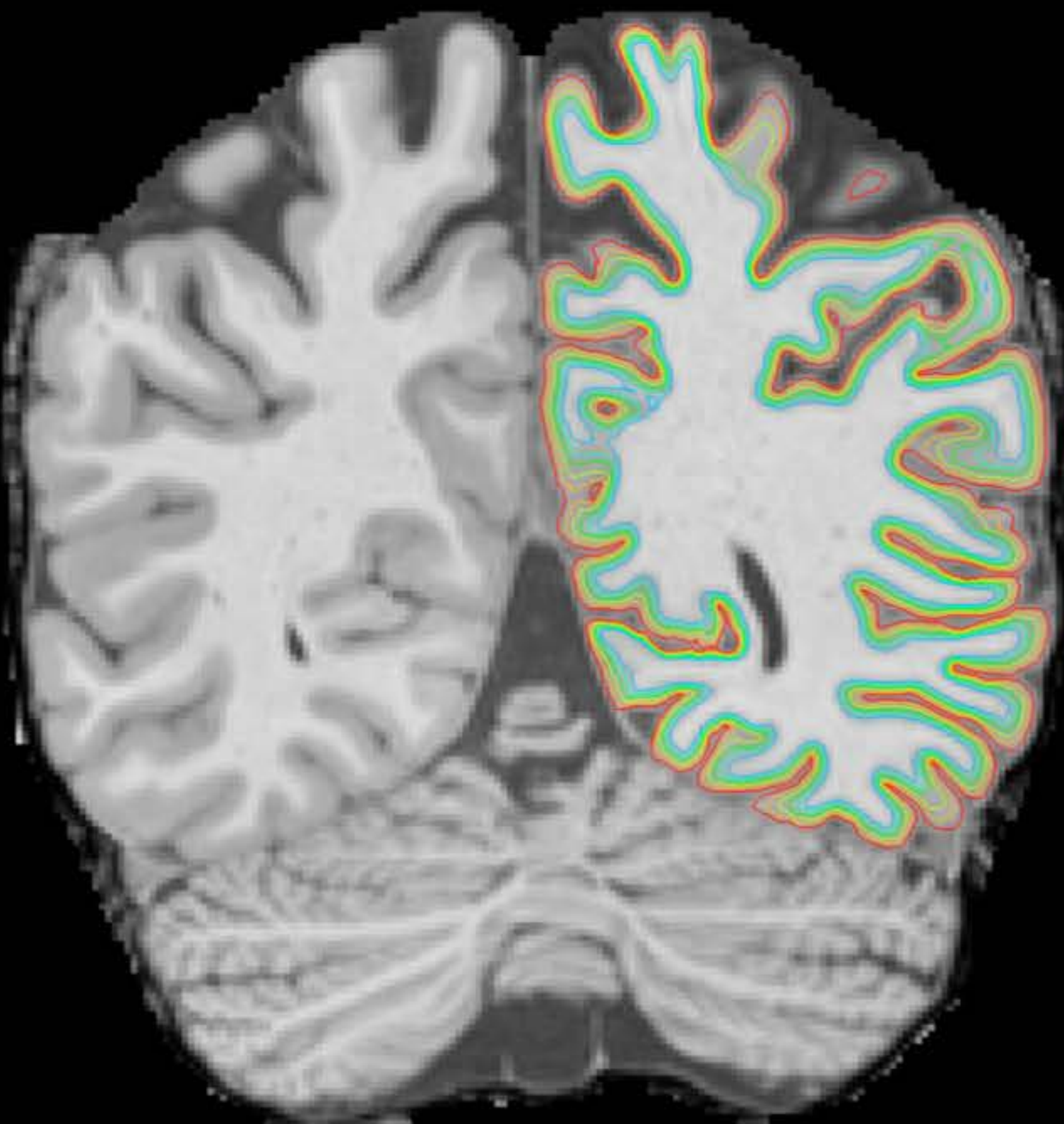


# High-res fMRI pre-processing (3/7)



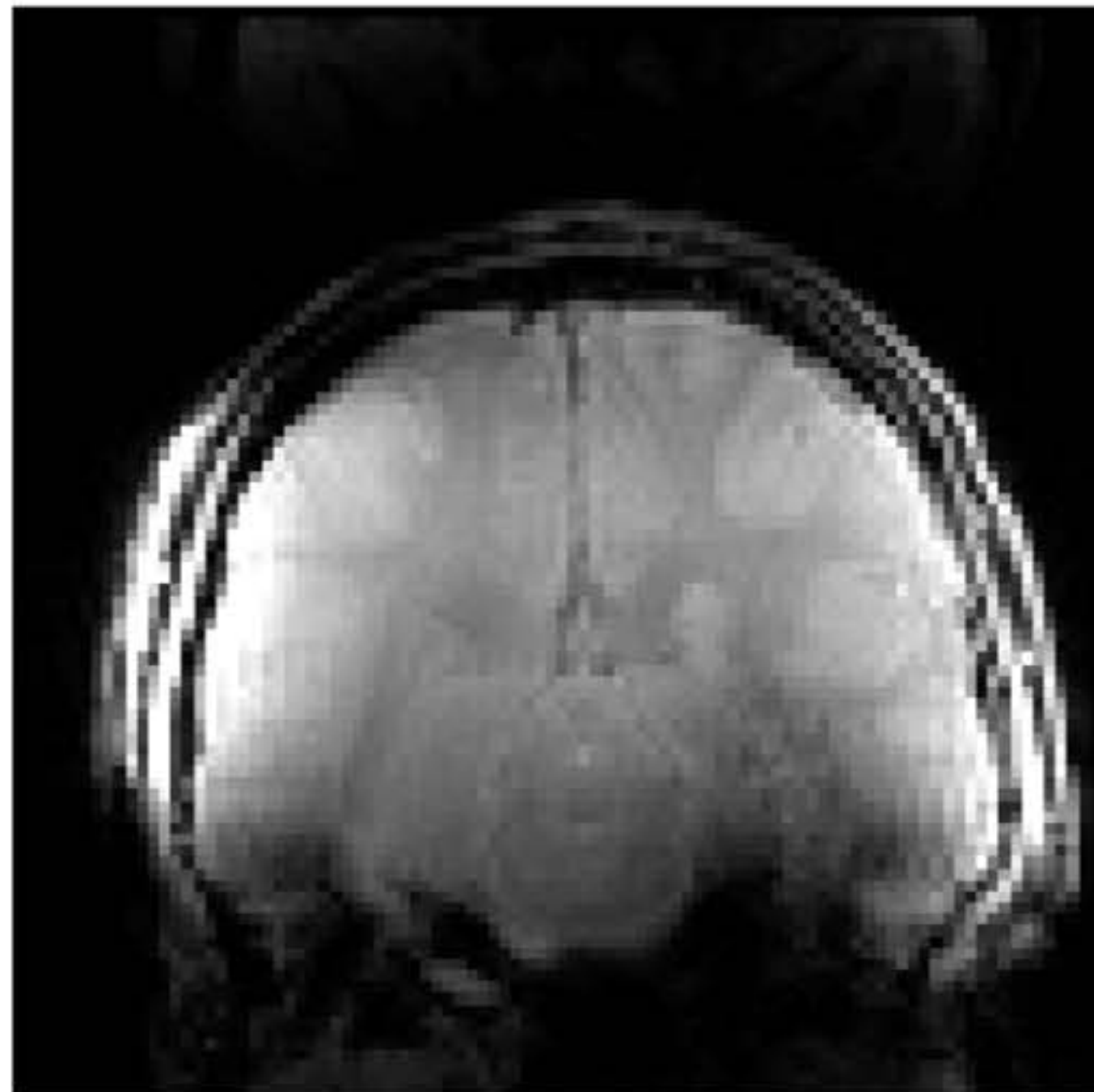


# High-res fMRI pre-processing (3/7)

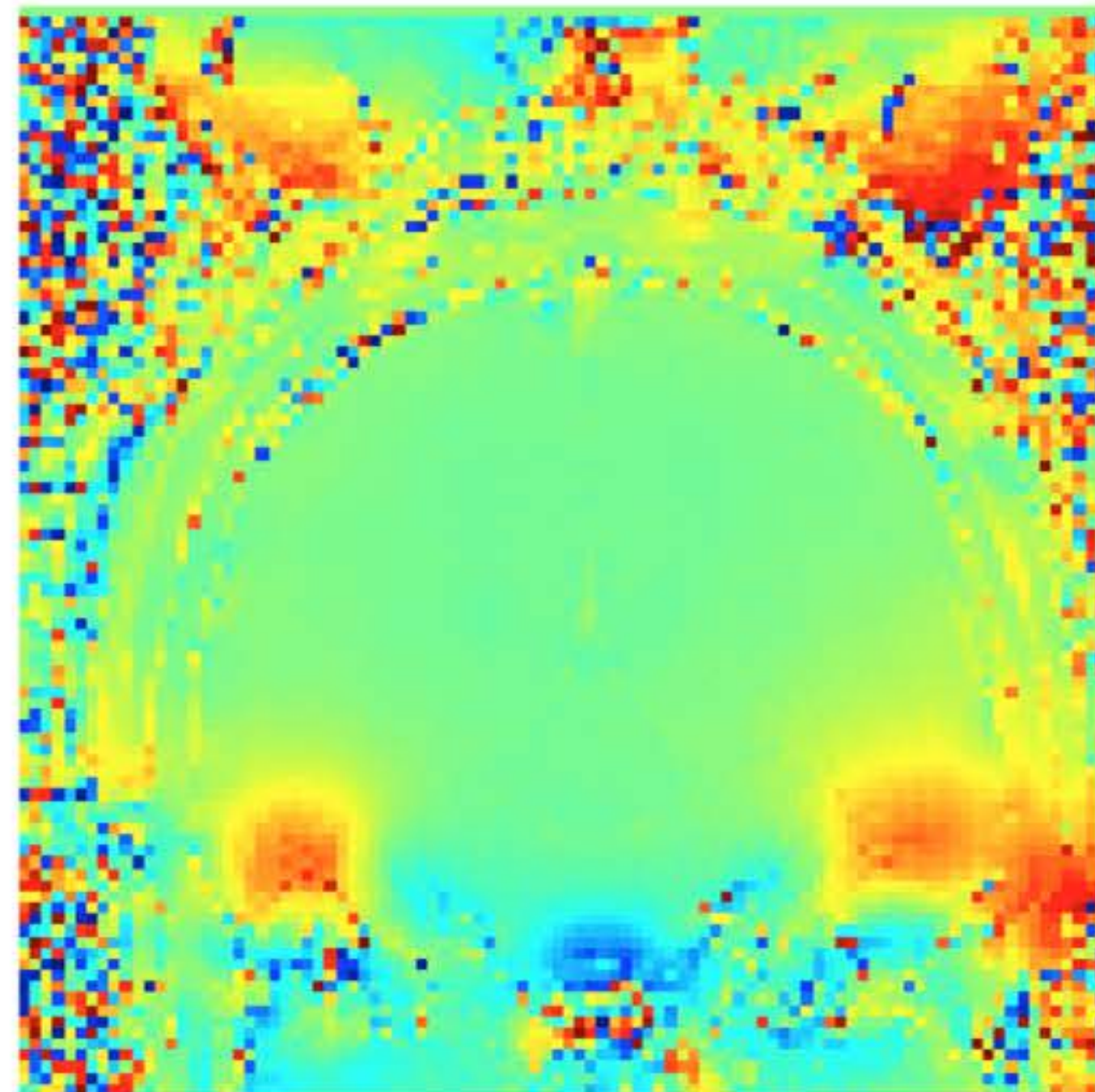




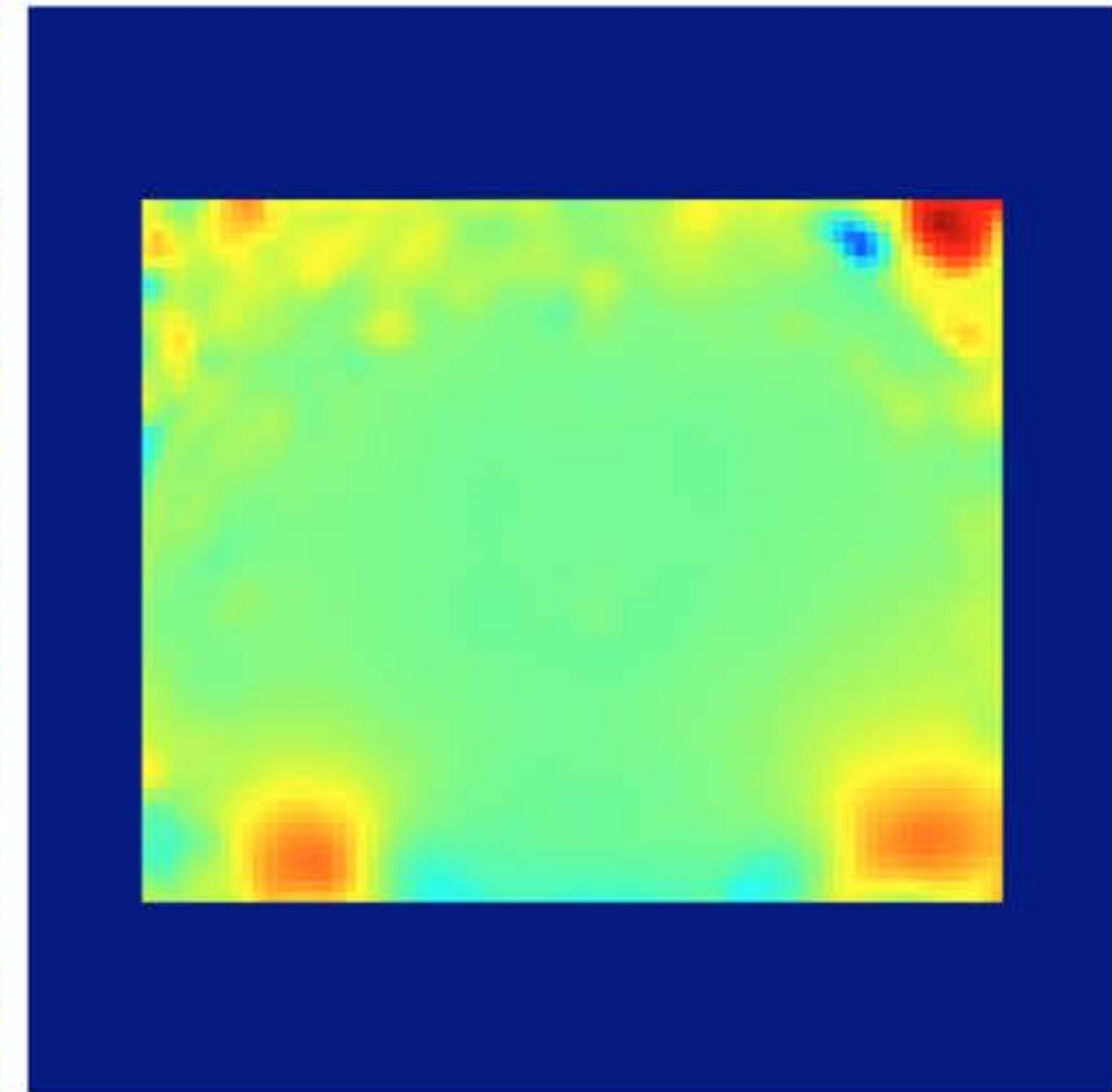
# High-res fMRI pre-processing (3/7)



Fieldmap magnitude



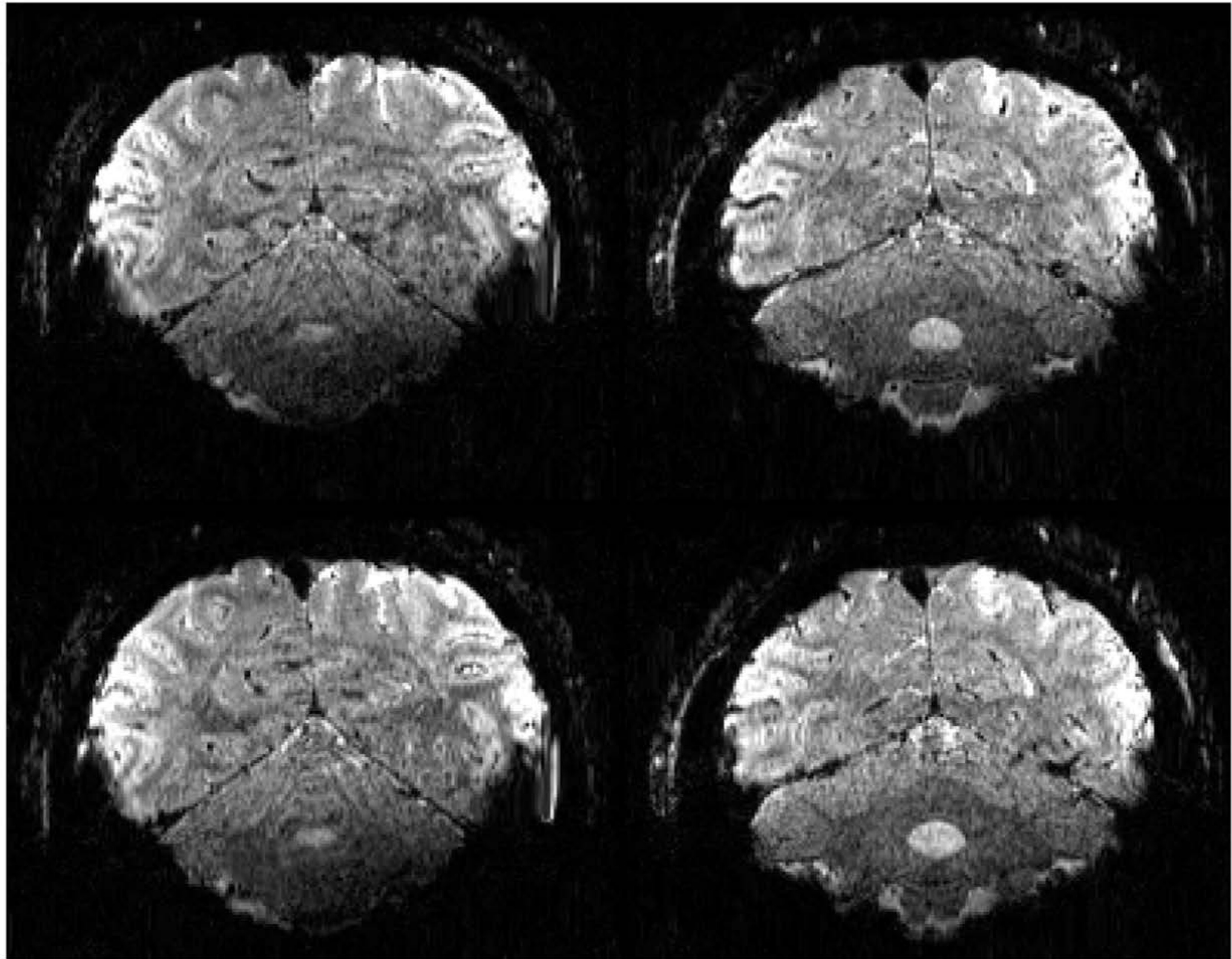
Fieldmap phase



Fieldmap phase  
(regularized)

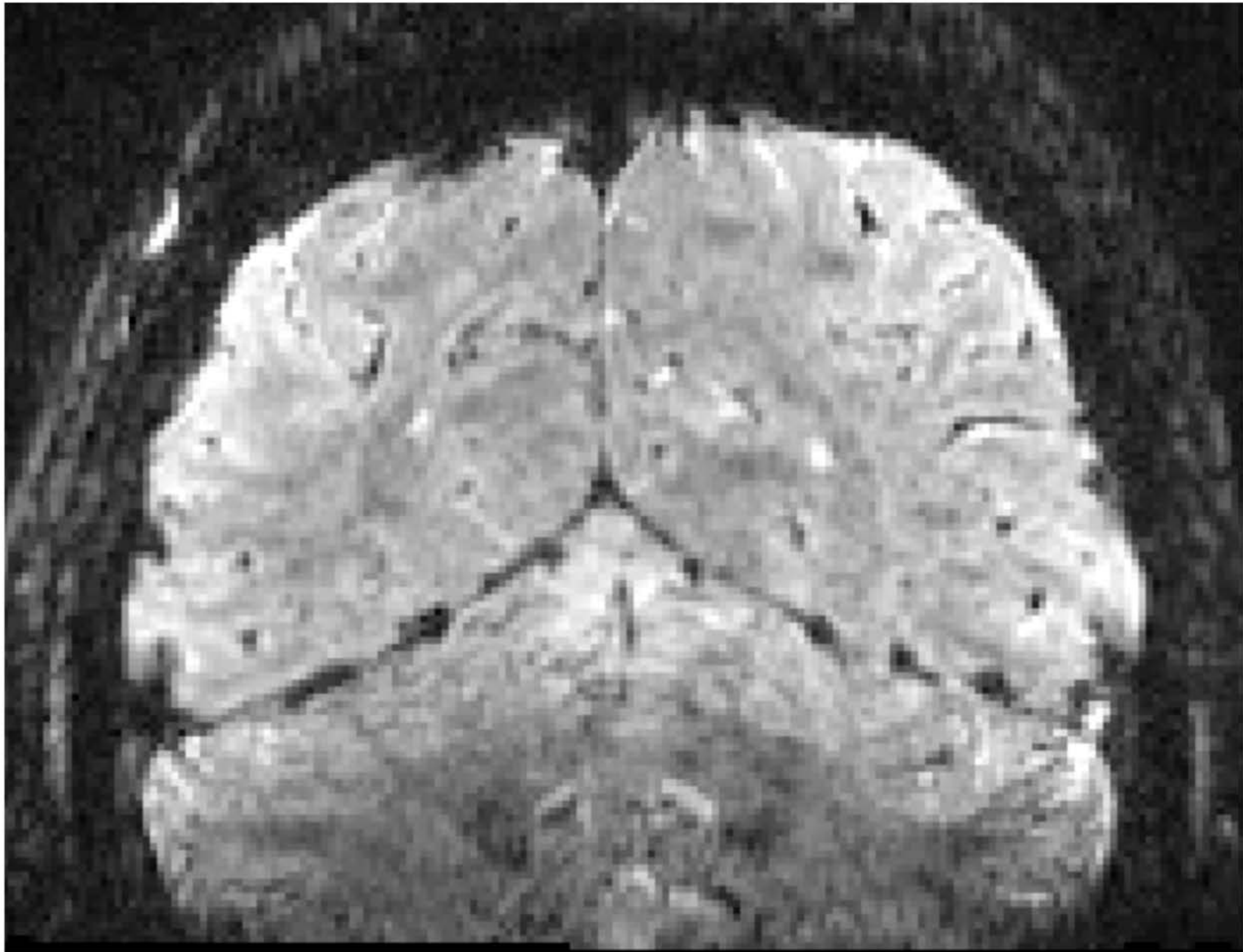


# High-res fMRI pre-processing (3/7)



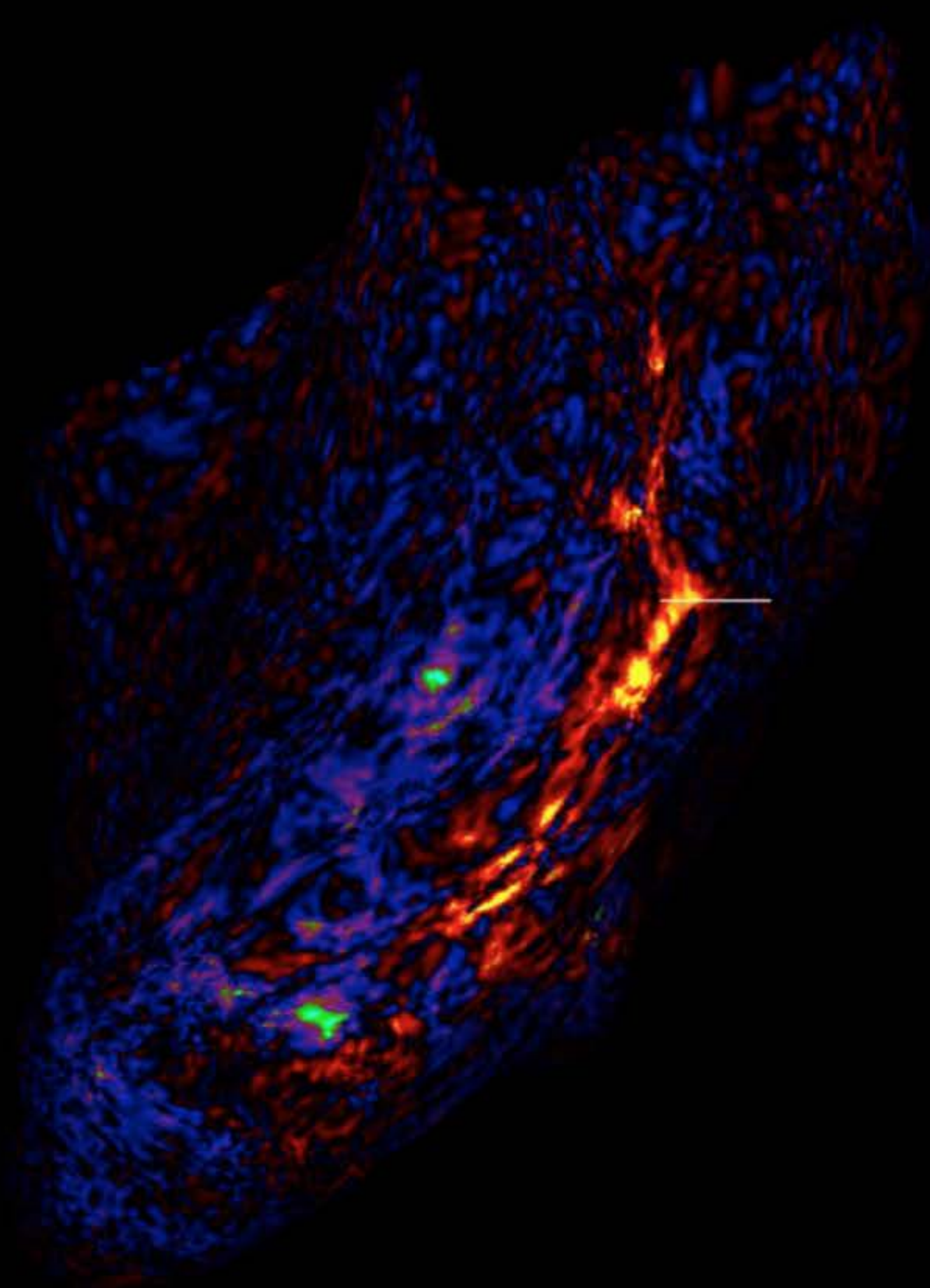
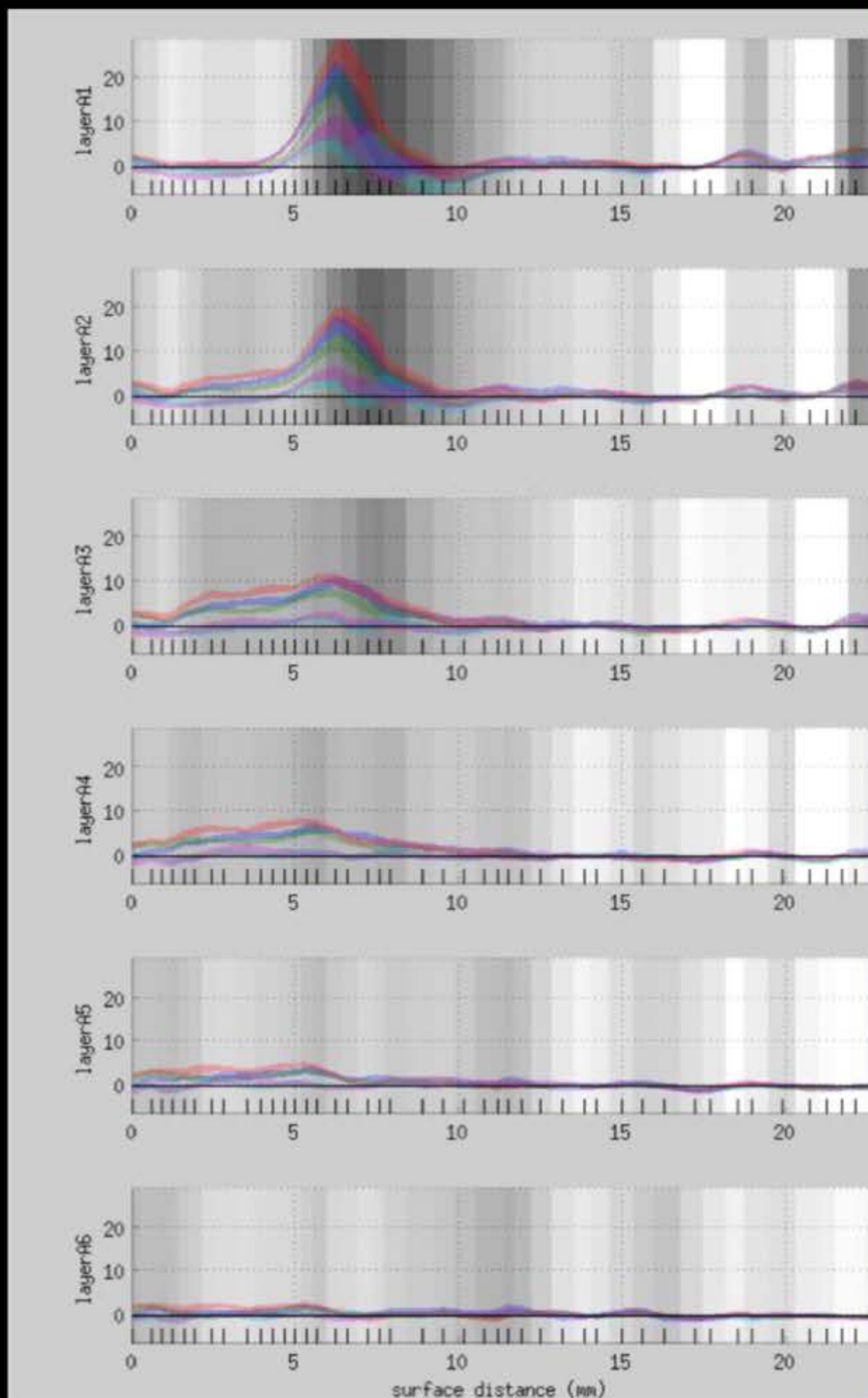


# High-res fMRI pre-processing (3/7)





# High-res fMRI pre-processing (3/7)





# Volume co-registration (4/7)

<http://github.com/kendrickkay/alignvolumedata/>

- Flexible inputs (any two volumes)
- Manual adjustment or automatic optimization
- Can use spatial mask
- Rigid-body or affine transformation



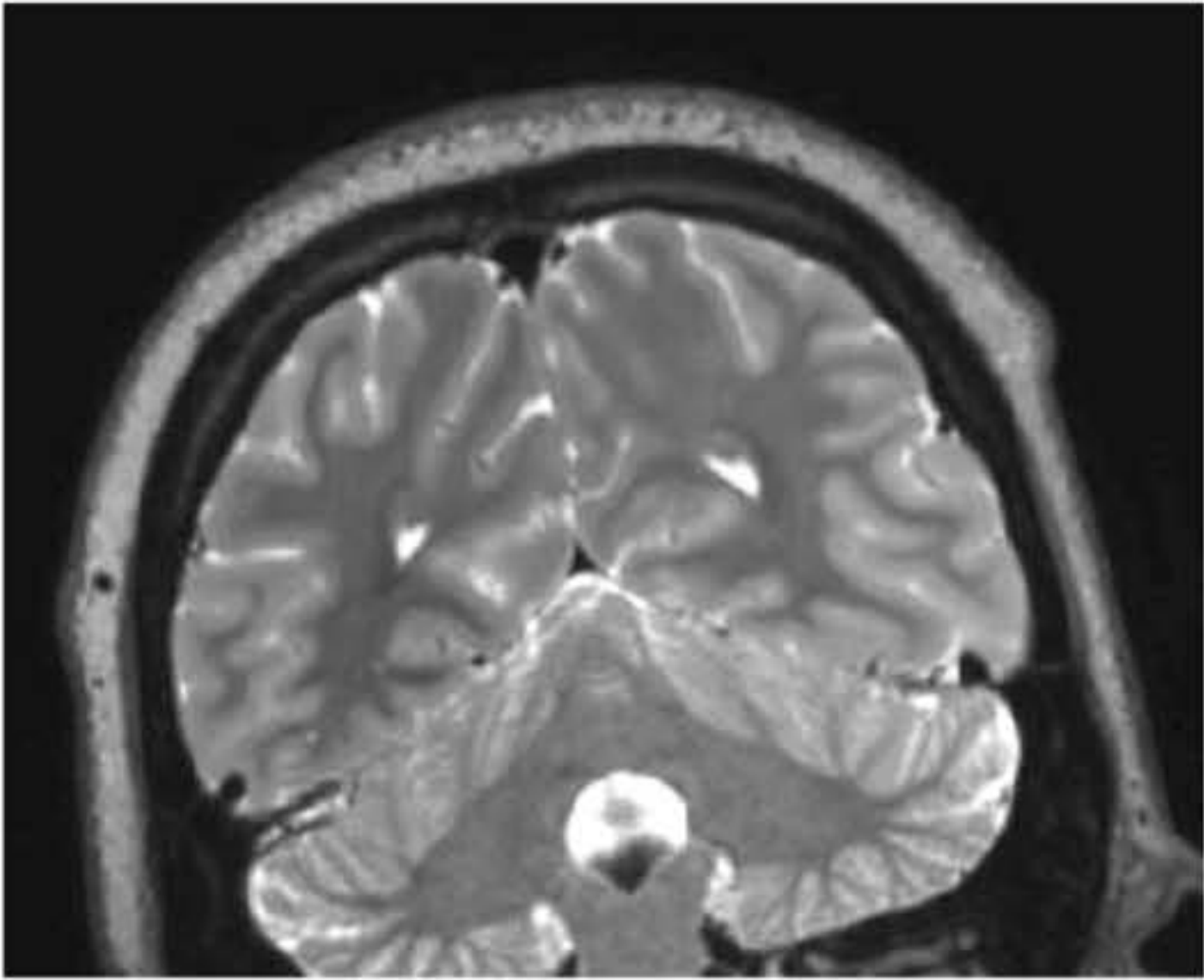
# Volume co-registration (4/7)

<http://github.com/kendrickkay/alignvolumedata/>

Figure 8

alignvolumedata

overlay (reference)



ALIGNMENT

ehx 0 < >

ehy 0 < >

ehz 0 < >

tx 123 < > esx 1 < >

ty 60 < > esy 1 < >

tz 137 < > esz -1 < >

rx -10 < > etx 0 < >

ry 43 < > ety 0 < >

rz 75 < > etz 0 < >

DISPLAY/OTHER

rotate ccw cw

interpolation linear

overlay mode single

overlay main ref

t/r step 1

s step 0.01

ref contrast

target contrast

render redraw

SLICE

slice 42 1

slice dimension z

slice average 1

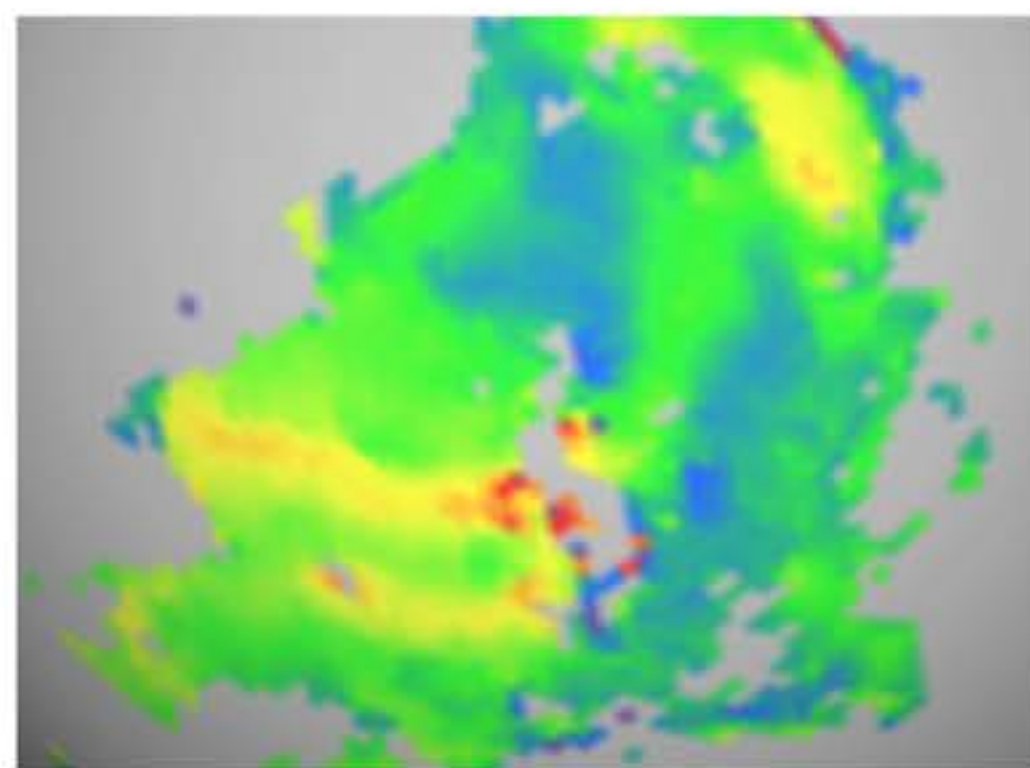
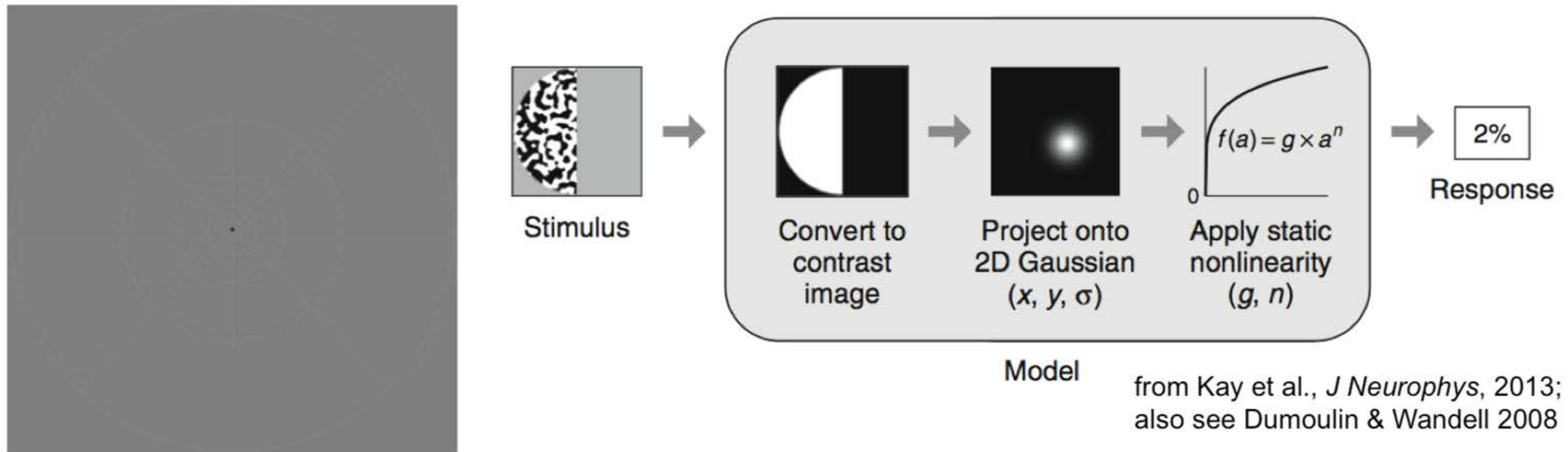
status: ready



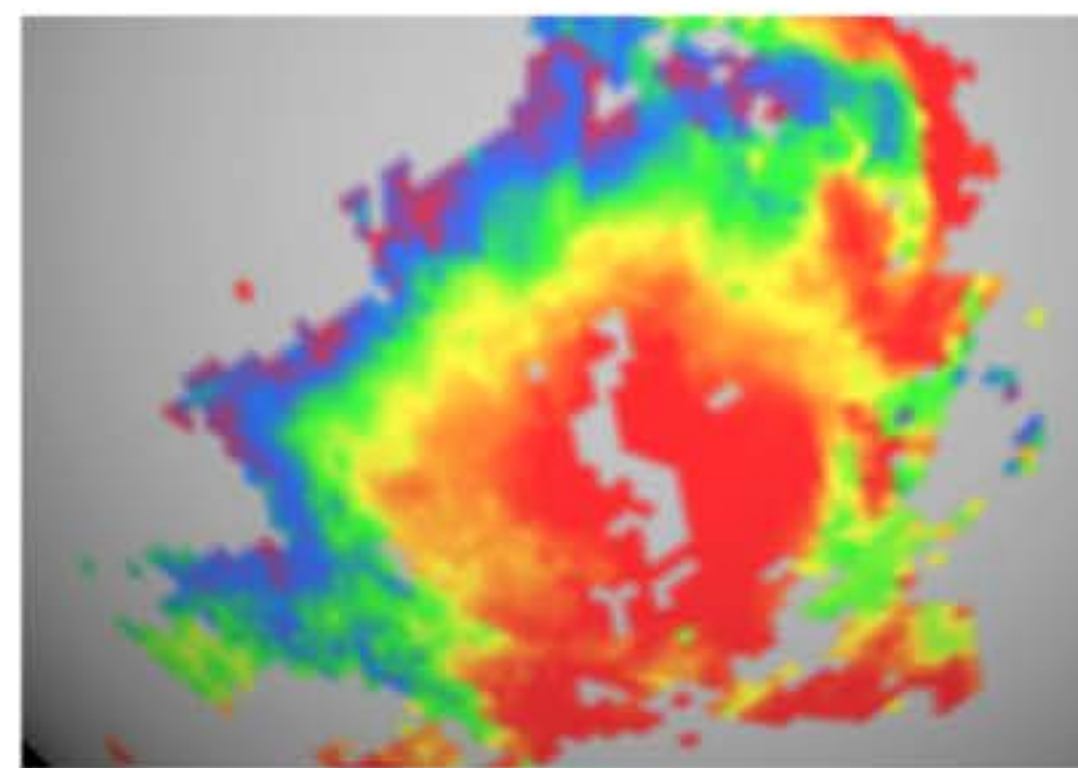
# analyzePRF (5/7)

<http://kendrickkay.net/analyzePRF/>

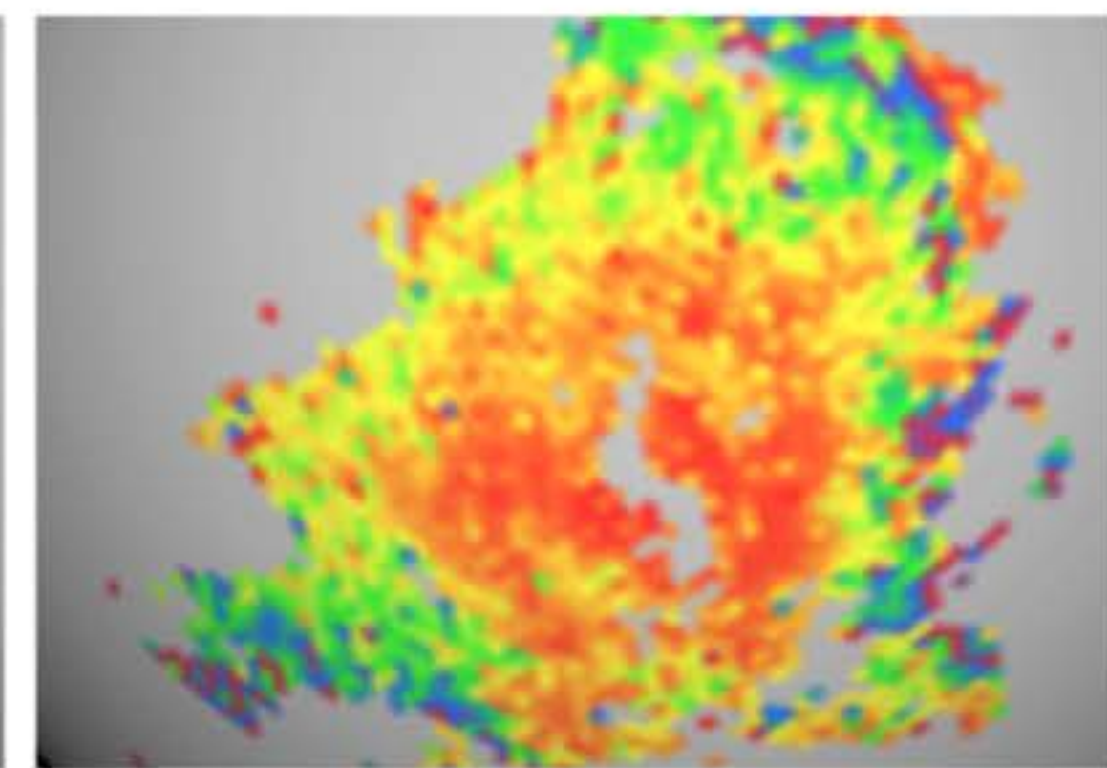
- Fit a parametric model that characterizes stimulus-response mapping



Angle



Eccentricity



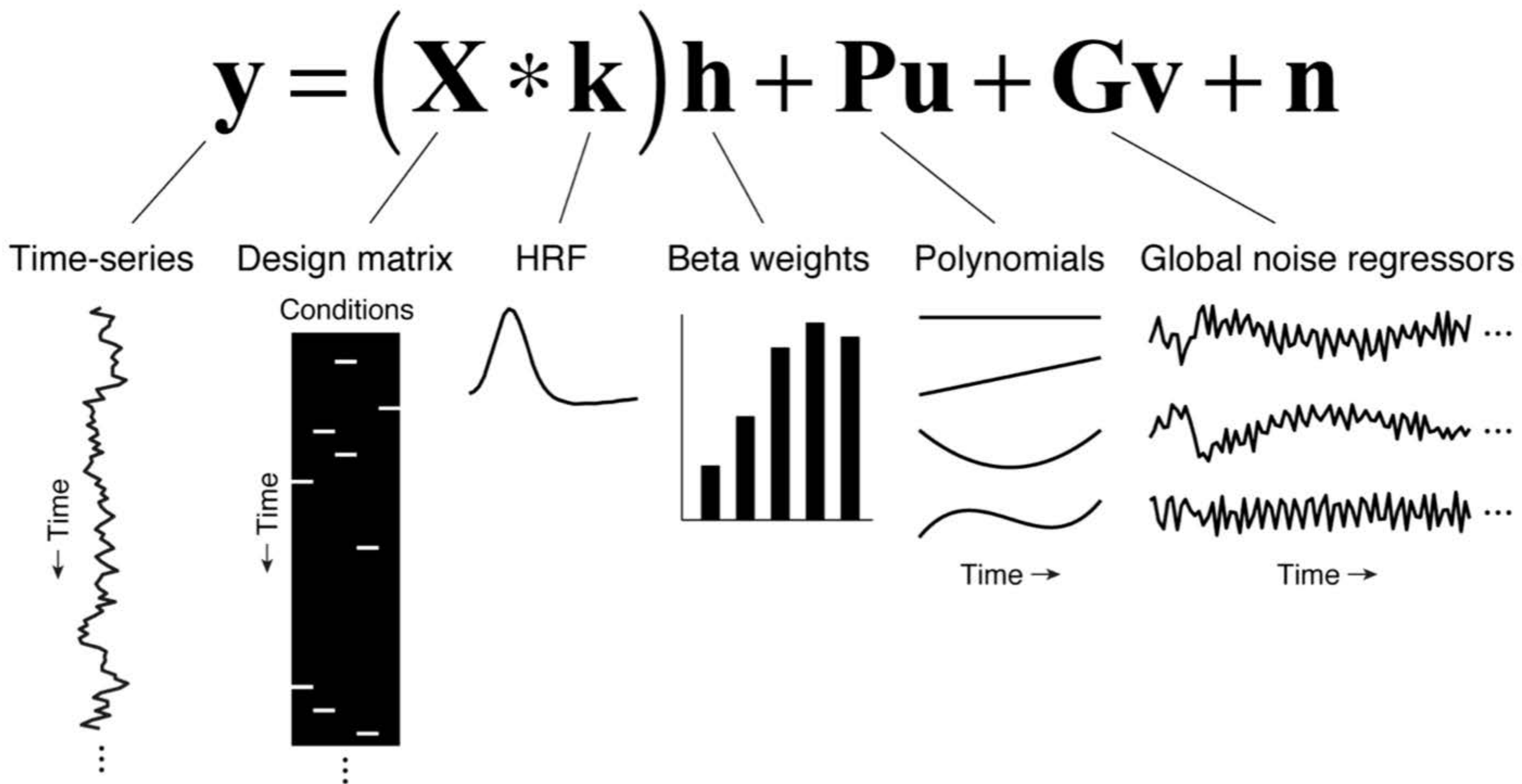
RF size



# GLMdenoise (6/7)

<http://kendrickkay.net/GLMdenoise/>

- Fit a GLM that derives noise regressors and produces denoised beta weights

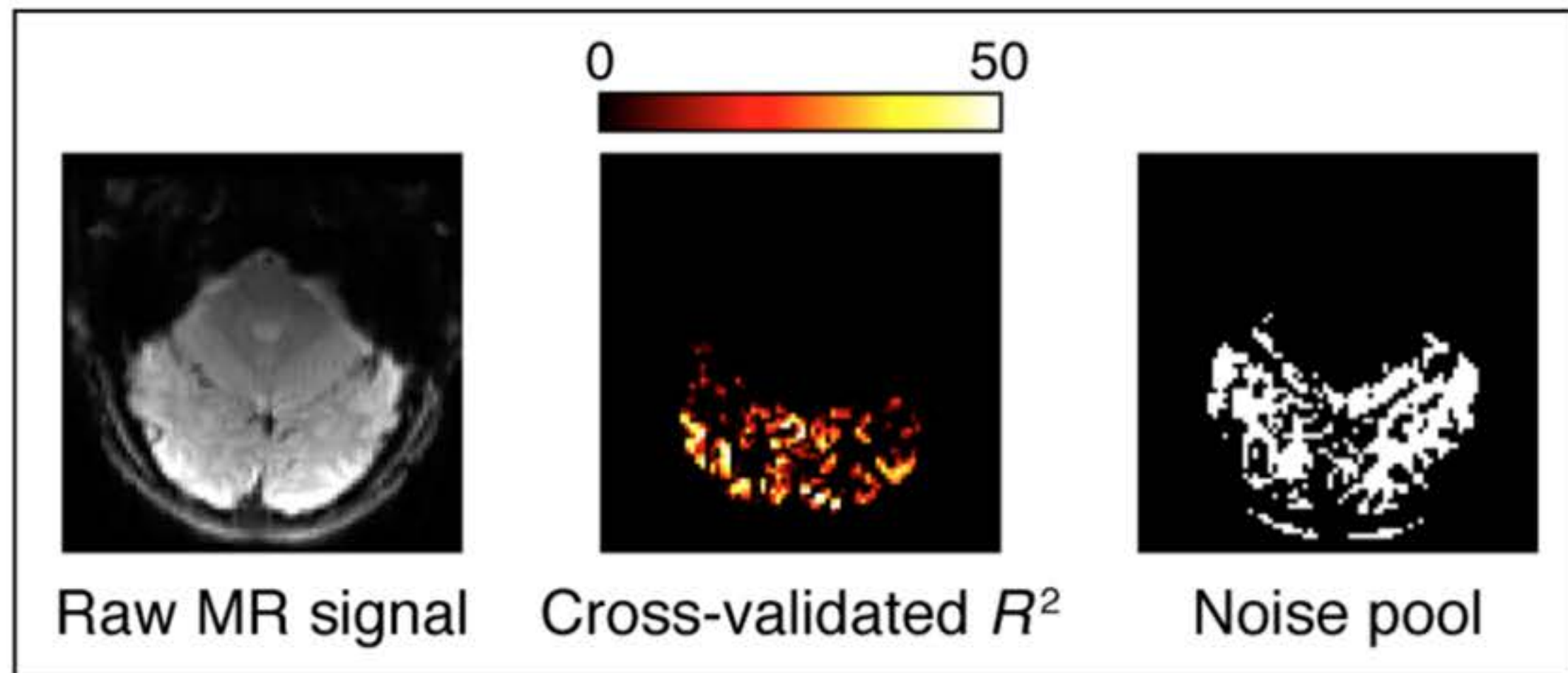


Kay et al., *Frontiers in Neuroscience*, 2013



# GLMdenoise (6/7)

<http://kendrickkay.net/GLMdenoise/>



1. Perform initial model fit

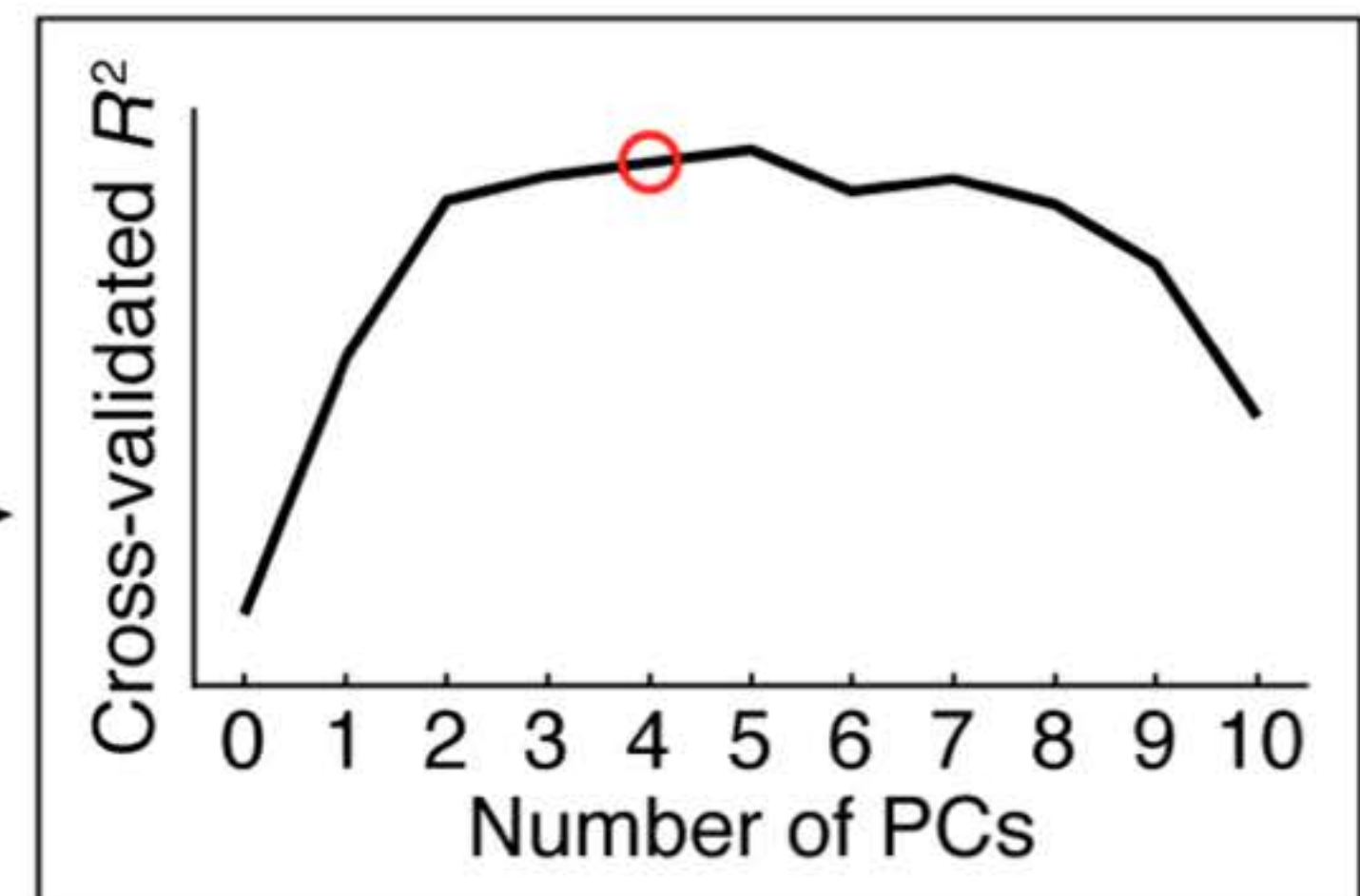
2. Determine noise pool (cross-validated  $R^2 < 0\%$ )

3. Perform PCA on noise pool

4. Add PCs into the model, one at a time

5. Select number of PCs using cross-validation

6. Fit final model to the full dataset



Kay et al., *Frontiers in Neuroscience*, 2013

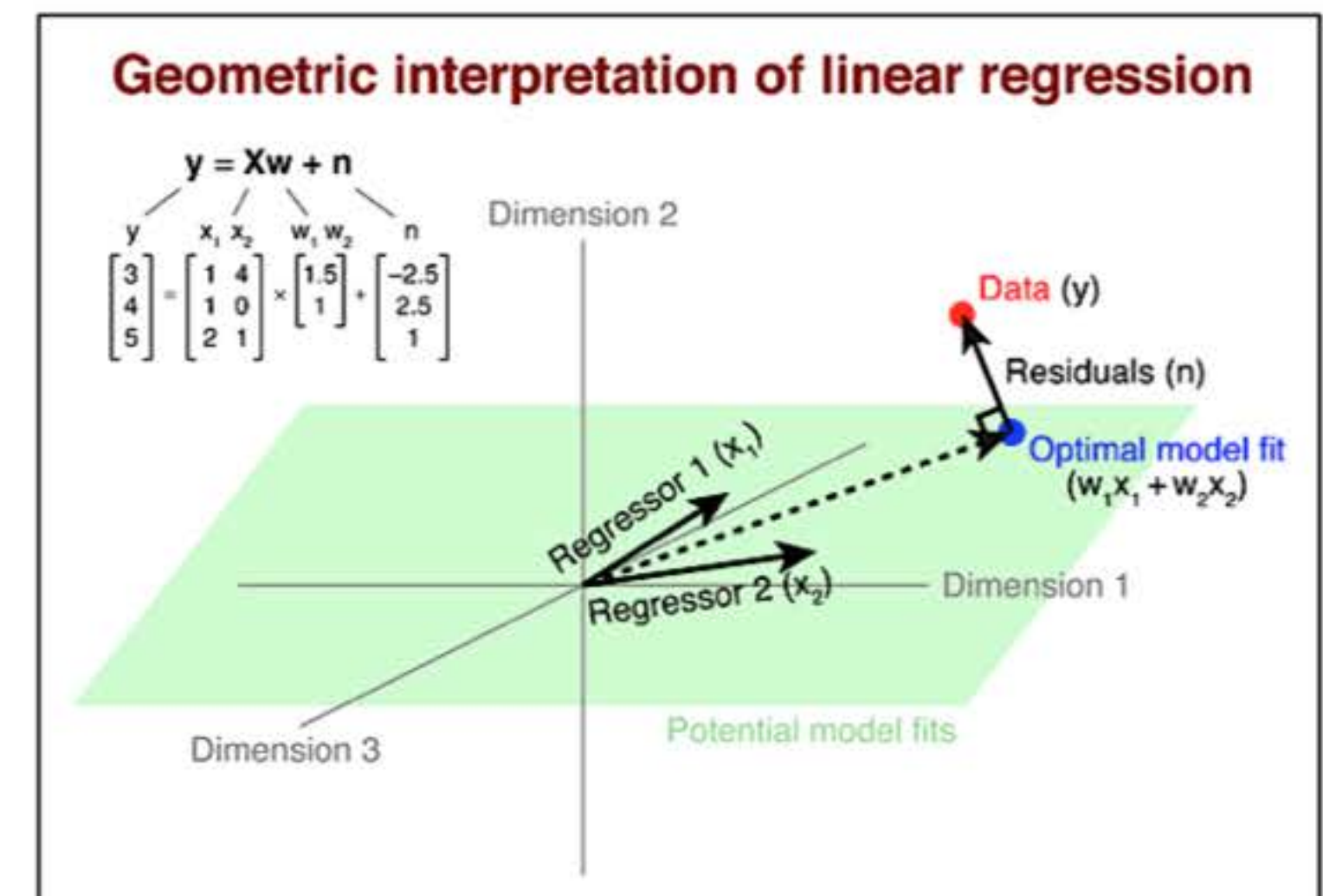
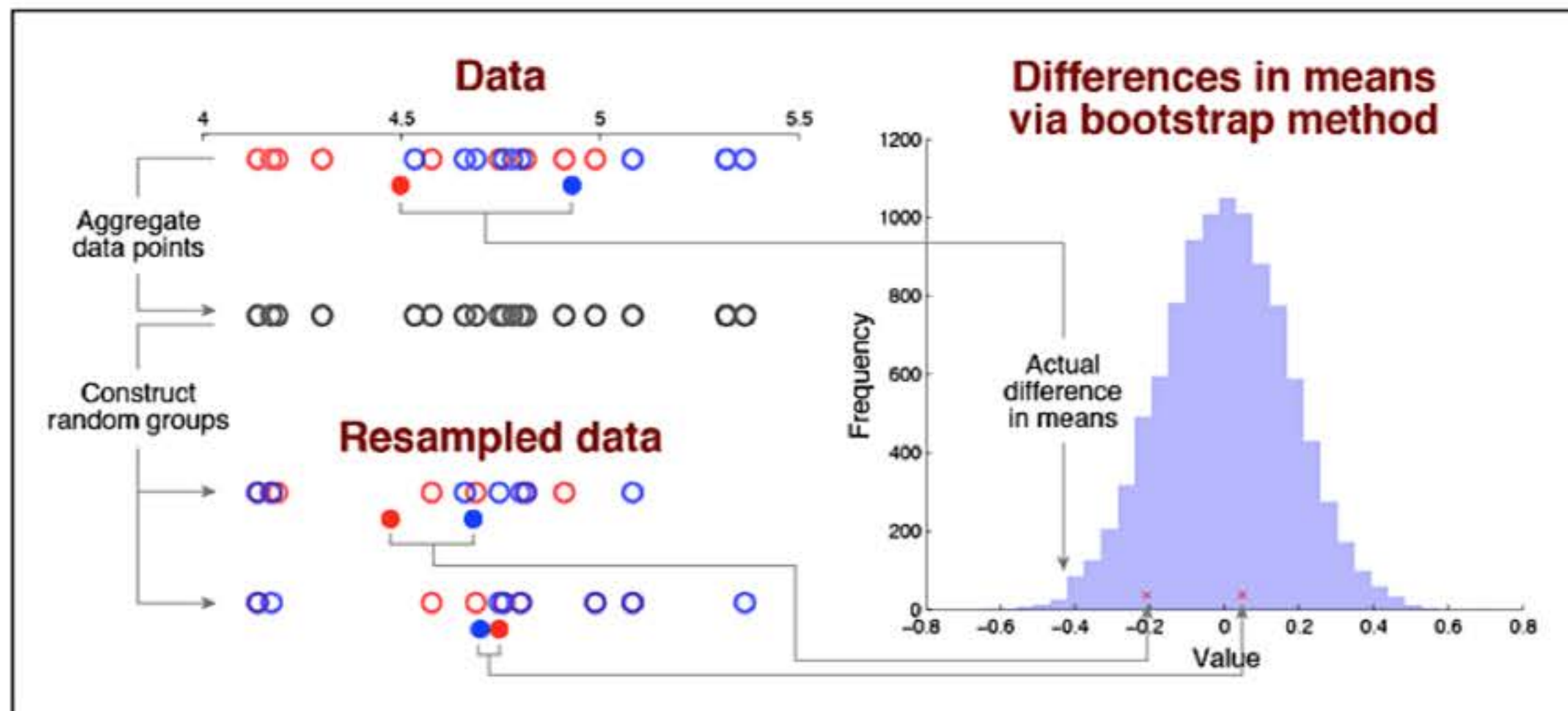
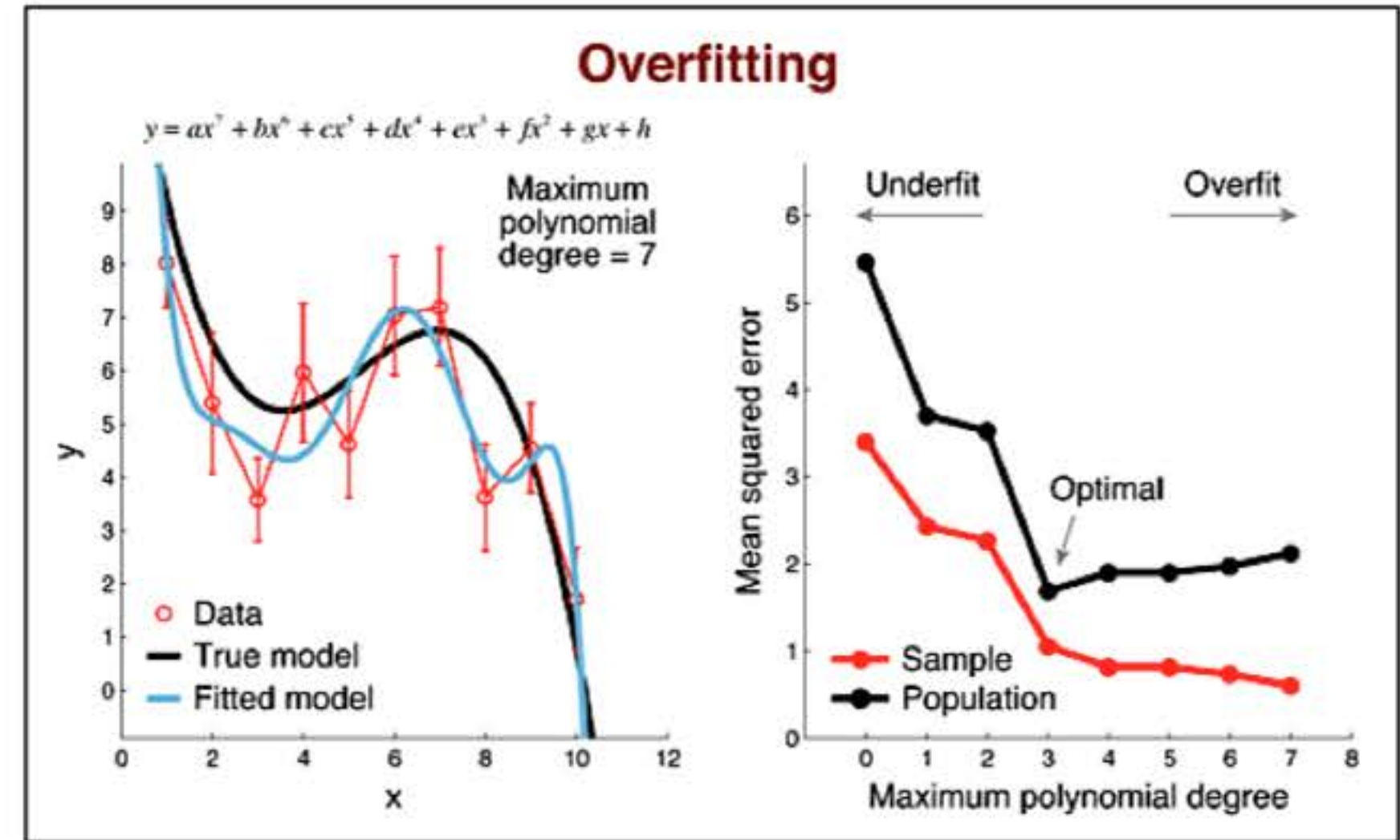
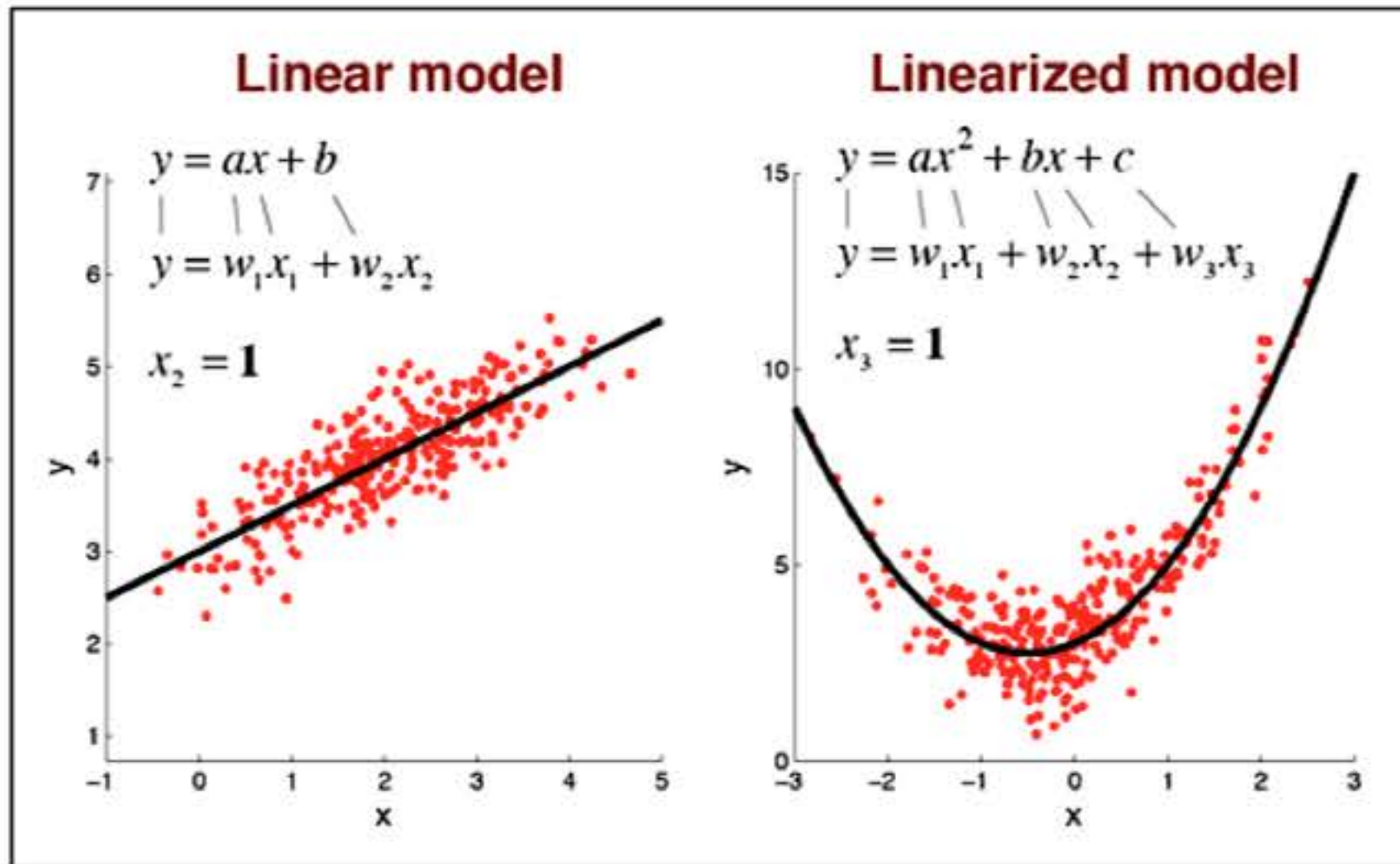


# Statistics, model fitting (7/7)

- Why statistical simulations?
  - They help teach concepts
  - They help check code correctness



# Statistics, model fitting (7/7)

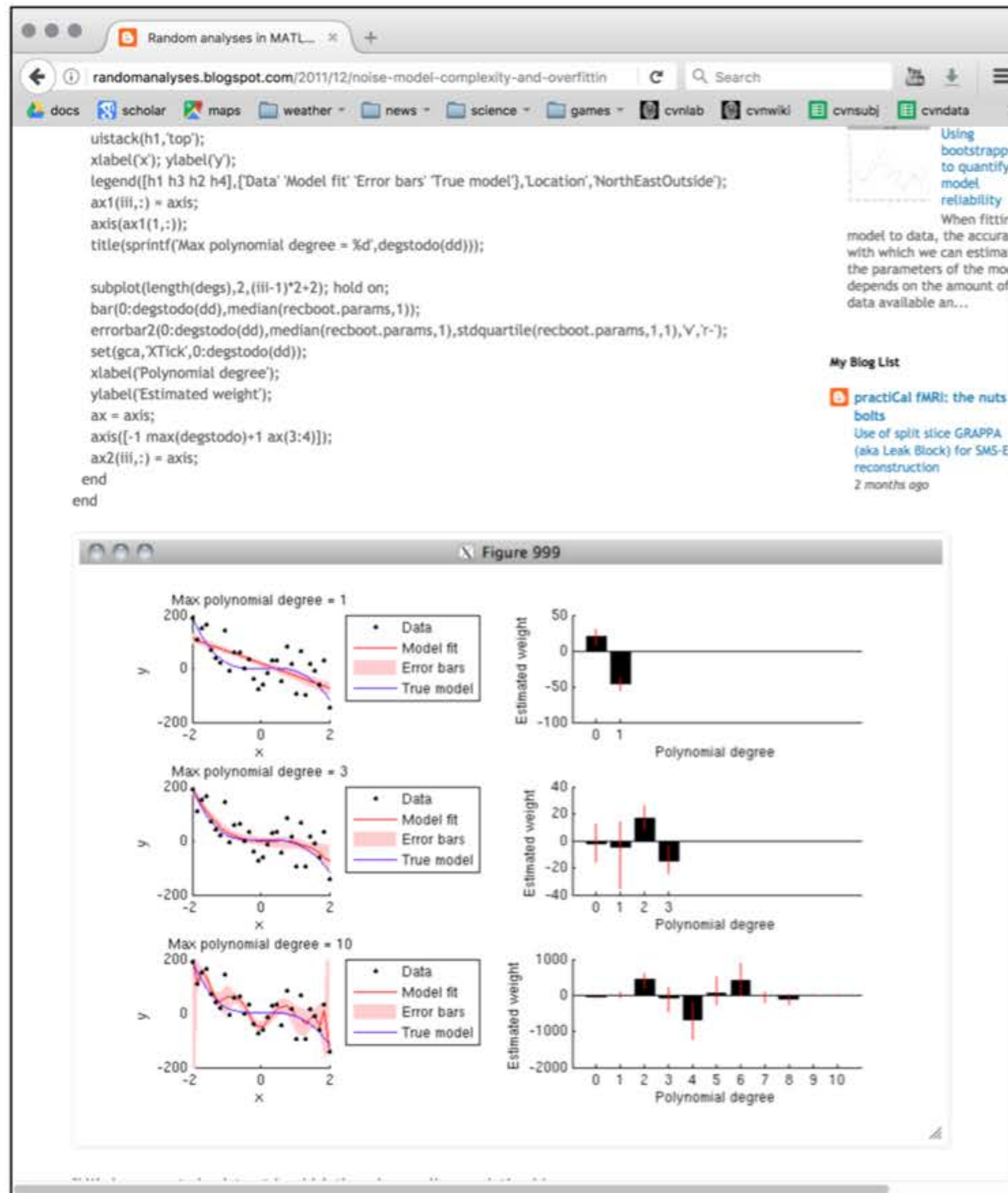


Materials at <http://kendrickkay.net/psych5007/>



# Statistics, model fitting (7/7)

<http://randomanalyses.blogspot.com>

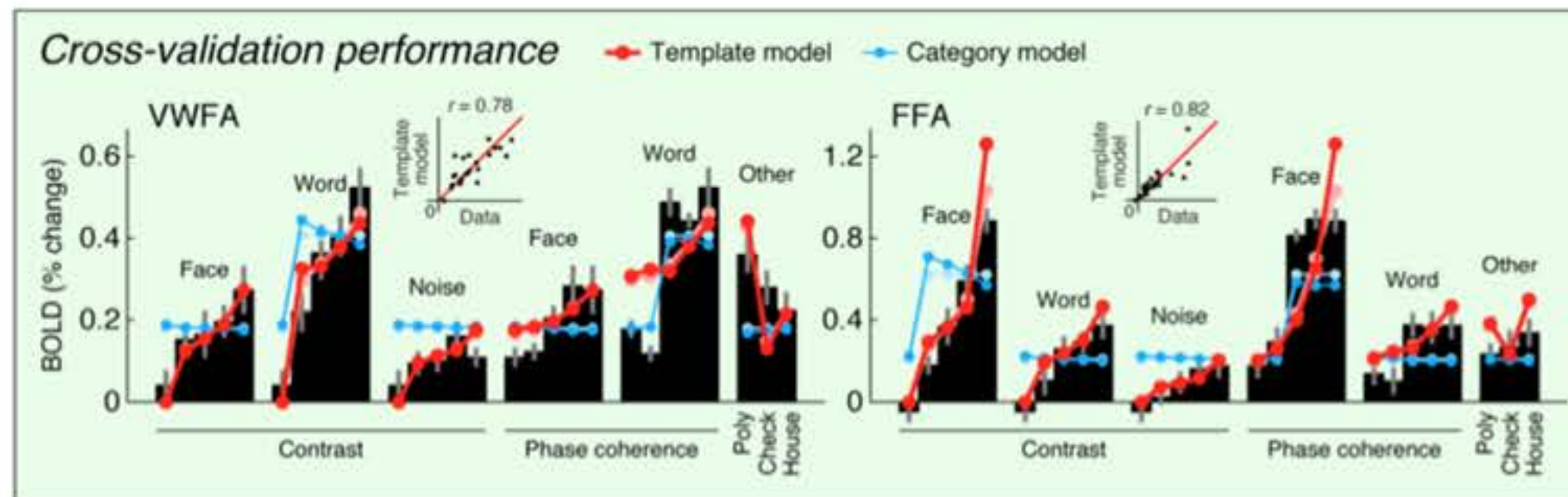
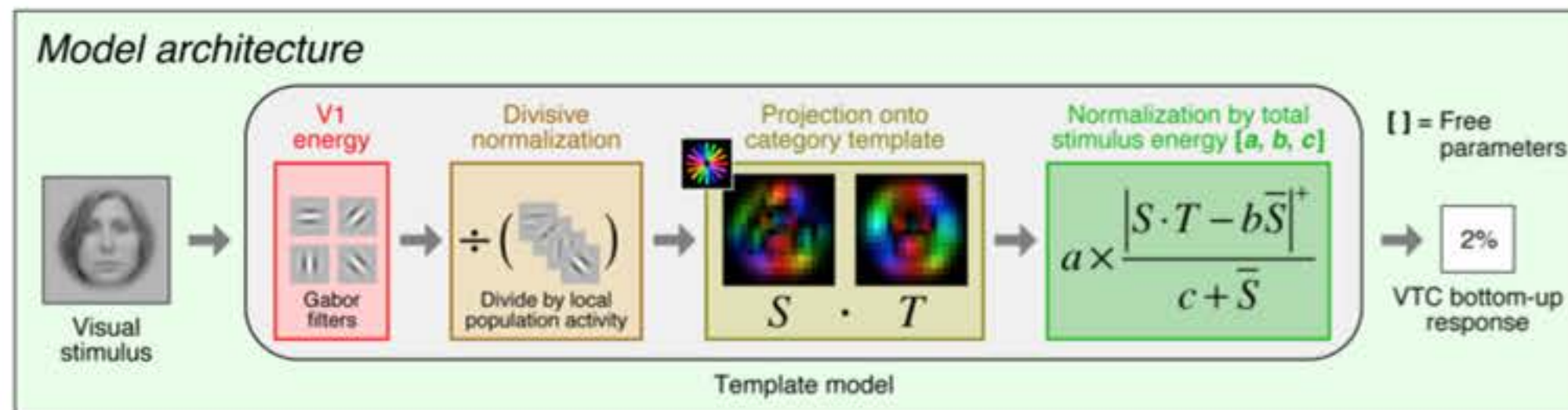




# Statistics, model fitting (7/7)

Kay & Yeatman, *eLife*, 2017

<http://cvnlab.net/vtcipsmodel/>



```

% IPS-scaling model
case 10
X = [repmat(eye(n),[3 1]) (1:3*n)'];
seed0 = @(ix) [data(1:n,ix)' datatopdown(:,ix)' 0 1];
opt1 = struct('stimulus',X,'data',@(ix) data(:,ix),'vxs',1:size(data,2), ...
    'model',{ ...
        {[NaN(1,n) NaN(1,3*n) -Inf -Inf; Inf(1,n+3*n+2)] ...
        @(p,x) x(:,1:n)*p(1:n)' .* nanreplace(p(n+3*n+1)*p(n+x(:,n+1))'+p(n+3*n+2),1)} ...
        {@(ss) ss [-Inf(1,n) NaN(1,3*n) -Inf -Inf; Inf(1,n+3*n+2)] ...
        @(p,x) x(:,1:n)*p(1:n)' .* nanreplace(p(n+3*n+1)*p(n+x(:,n+1))'+p(n+3*n+2),1)}}, ...
    'seed',seed0,'resampling',xvalscheme,'metric',metricfun, ...
    'optimoptions',{('Display','off')},extraopt{:});

end

% finally, fit the model
results = fitnonlinearmodel(opt1);

% take the results and store them
switch xx
case 1
    modelfit(:,mm) = squish(results.modelfit(1,:,:),2);
    modelparams(mm) = squish(results.params(1,:,:),2);
case 2
    modelpred(:,mm) = results.modelpred;
    modelperformance(:,mm) = results.aggregatedtestperformance(1,:);
end
    
```



# Summary

1. Automated surface visualizations
2. Brain art
3. High-res fMRI pre-processing
4. Volume co-registration
5. analyzePRF
6. GLMdenoise
7. Statistics, model fitting



<http://cvnlab.net>



# Acknowledgments

- Keith Jamison (UMN)



- Eshed Margalit (Stanford)

