General Information

News

- 2023/08/20 - All data and files associated with the last 3 NSD core scan sessions have now been publicly released.
- 2021/12/16 - The NSD data paper is now published in *Nature Neuroscience*.
- 2021/09/03 - The NSD dataset is now released, and version 1.1 of the NSD Data Manual is now complete. A video walkthrough of NSD data files is also now available (details below).
- 2021/02/15 - Version 1.0 of the NSD Data Manual is now complete.

Basic information

Welcome to the Natural Scenes Dataset (NSD) Data Manual. This web site provides a detailed, technical description of all NSD data files that are available. It will be updated as questions and issues arise. The information on this site is also available as a single downloadable PDF (last snapshot 2023/08/20 - version 1.3) (this may be convenient for performing "Find" queries).

If you want to browse or download the data, please see 🗎 How to get the data .

The official paper that formally describes the NSD dataset is available as:

> Allen, St-Yves, Wu, Breedlove, Prince, Dowdle, Nau, Caron, Pestilli, Charest, Hutchinson, Naselaris*, & Kay*. A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience* (2022).

We refer to this as the "NSD data paper". The data paper has some associated online resources at the NSD OSF site. The contents of this data manual assume familiarity with the data paper.

We have created a video walkthrough that gives an overview of the data files present in the NSD dataset.

Announcements and updates to the NSD dataset will be documented and logged on this page, so check back regularly.

If you have questions about the NSD dataset, please either (1) post your question to the nsd-users mailing list, (2) open an issue or discussion on the relevant github repository (e.g.

http://github.com/cvnlab/nsdcode/ or http://github.com/cvnlab/nsddatapaper/), (3) send queries directly to ✉ kay@umn.edu, or (4) submit anonymous feedback/suggestions via this Google form. Please let us know if there is missing documentation or if something is not clear.

Change history

Substantive changes to NSD data files are documented and logged here:

- 2023/08/20 - The files related to the last 3 NSD core scan sessions from each subject are now publicly released.
- 2023/05/27 - The files related to the final memory test (nsdmemory) are now publicly released. See Experiments and Behavioral data .
- 2022/08/15 - In nsddata/inspections/rois/prf-visualandecc/, a few visualizations were incorrect. Specifically, the files "subj02_prf-eccrois_on_eccentricity.png" and "subj02_prf-visualrois_on_angle.png" have now been corrected.
- 2022/01/26 - For user convenience, we now provide some additional versions of the nsddata/stimuli/prf stimulus files (description has been updated in Untitled).
- 2021/10/20 - Diffusion derivatives are now available (nsddata_diffusion/) and documented in the data manual (see Untitled). Summary b=0 diffusion files (called nsddata/ppdata/subj*/anat/DWI_*.nii.gz) and associated nsddata/inspections/coregistration/*DWI* files have been created to help visualize the quality of the pre-processed diffusion data and their registration to the T1+T2 anatomy. In addition, the "knowndataproblems.txt" file has been slightly updated/modified.
- 2021/09/03 - Initial public release of the NSD dataset.
- 2021/09/02 - actually add split-half ncsnr (noise ceiling) files (this was for some reason not completed on the previous iteration on 2021/08/07)
- 2021/08/07 - add additional files pertaining to BOLDscreen calibration; add information on race to nsddemographics.xlsx; include Phase component of the SWI scans to the raw BIDS data; add split-half ncsnr (noise ceiling) files; add pre-processed eyetracking data and inspection figures
- 2021/07/23 - design .tsv files for the nsdsynthetic experiment were incorrect; these have been fixed.
- 2021/05/16 - Added probmap .mgz files (see ROIs) and associated .png surfacevisualizations (see Data inspections)
- 2020/12/20 - Official version 1.0 release of nsd_mapdata (in the nsdcode repository).

Community-driven content

If you have NSD-related information, tools, resources, tutorials, or links that you would like to share with the community, please contact ✉ kay@umn.edu and the information can be listed here.

- **nsdexamples (http://github.com/kendrickkay/nsdexamples).** These example scripts, written by Kendrick Kay, were created to demonstrate some basic loading, analysis, and visualization of the NSD dataset.
- **nsd_access (https://github.com/tknapen/nsd_access).** This toolbox, written by Tomas Knapen, provides a convenient Python-based interface to the NSD dataset. There are also some examples of how to load data and perform basic visualization. The toolbox also enables easy access to COCO image annotation information, including category labels and bounding boxes.

Papers and pre-prints

Here are links to papers and pre-prints that use NSD data.

- **Fractional Ridge Regression: a Fast, Interpretable Reparameterization of Ridge Regression.**
  Rokem, A. & Kay, K.
  GigaScience (2020).
- **Extensive sampling for complete models of individual brains.**
  Naselaris, T., Allen, E., & Kay, K.
  Current Opinion in Behavioral Sciences (2021).
- **A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence.**
  Allen, St-Yves, Wu, Breedlove, Prince, Dowdle, Nau, Caron, Pestilli, Charest, Hutchinson, Naselaris*, & Kay*.
  Nature Neuroscience (2022).
- **NeuroGen: activation optimized image synthesis for discovery neuroscience.**
  Gu, Z., Jamison, K.W., Khosla, M., Allen, E.J., Wu, Y., Naselaris, T., Kay, K., Sabuncu, M.R., Kuceyeski, A.
  NeuroImage (2022).
- **Non-Neural Factors Influencing BOLD Response Magnitudes within Individual Subjects.**
  Kurzawski, J.W., Gulban, O.F., Jamison, K., Winawer, J.*, Kay, K.*
  Journal of Neuroscience (2022).
- **What can 5.17 billion regression fits tell us about artificial models of the human visual system?**
  Colin Conwell, Jacob S. Prince, George A. Alvarez, Talia Konkle
  NeurIPS SVRHM workshop (2021).
- **Improving the accuracy of single-trial fMRI response estimates using GLMsingle.**
  Prince, J.S., Charest, I., Kurzawski, J.W., Pyles, J.A., Tarr, M.J., Kay, K.N.

eLife (2022).

- **Personalized visual encoding model construction with small data.**
  Zijin Gu, Keith Jamison, Mert Sabuncu, and Amy Kuceyeski
  Communications Biology (2022).
- **Large-Scale Benchmarking of Diverse Artificial Vision Models in Prediction of 7T Human Neuroimaging Data.**
  Colin Conwell, Jacob S. Prince, George A. Alvarez, Talia Konkle
  bioRxiv (2022).
- **Selectivity for food in human ventral visual cortex.**
  Nidhi Jain, Aria Wang, Margaret M. Henderson, Ruogu Lin, Jacob S. Prince, Michael J. Tarr, and Leila Wehbe
  Communications Biology (2023).
- **High-level visual areas act like domain-general filters with strong selectivity and functional specialization.**
  Meenakshi Khosla, Leila Wehbe
  bioRxiv (2022).
- **Short-term plasticity in the human visual thalamus.**
  Jan W Kurzawski, Claudia Lunghi, Laura Biagi, Michela Tosetti, Maria Concetta Morrone, Paola Binda
  eLife (2022).
- **Color-biased regions in the ventral visual pathway are food selective.**
  Pennock, I.M.L., Racey, C., Allen, E.J., Wu, Y., Naselaris, T., Kay, K.N., Franklin, A., Bosten, J.M.
  Current Biology (2022).
- **Multiple Traces and Altered Signal-to-Noise in Systems Consolidation: Complementary Evidence from the 7T fMRI Natural Scenes Dataset.**
  Vanasse, T.J., Boly, M., Allen, E.J., Wu, Y., Naselaris, T., Kay, K., Cirelli, C., Tononi, G.
  PNAS (2022).
- **The risk of bias in data denoising methods: examples from neuroimaging.**
  Kay, K.
  PLoS One (2022).
- **A Highly Selective Response to Food in Human Visual Cortex Revealed by Hypothesis-Free Voxel Decomposition.**
  Meenakshi Khosla, N. Apurva Ratan Murty, Nancy G Kanwisher
  Current Biology (2022).
  - See commentary:
    Visual cortex: Big data analysis uncovers food specificity.
    Michael M. Bannert and Andreas Bartels
    Current Biology (2022).

- **Low-level tuning biases in higher visual cortex reflect the semantic informativeness of visual features.**
  Margaret Henderson, Michael J. Tarr, Leila Wehbe
  Journal of Vision (2023).
- **Re-expression of CA1 and entorhinal activity patterns preserves temporal context memory at long timescales.**
  Futing Zou, Wanjia Guo, Emily J. Allen, Yihan Wu, Ian Charest, Thomas Naselaris, Kendrick Kay, Brice A. Kuhl, J. Benjamin Hutchinson, Sarah DuBrow
  bioRxiv (2022).
- **A texture statistics encoding model reveals hierarchical feature selectivity across human visual cortex.**
  Margaret M. Henderson, Michael J. Tarr, Leila Wehbe
  Journal of Neuroscience (2023).
- **Semantic scene descriptions as an objective of human vision**
  Doerig, A., Kietzmann, T.C., Allen, E., Wu, Y., Naselaris, T., Kay, K., Charest, I.
  arXiv (2022).
- **Incorporating natural language into vision models improves prediction and understanding of higher visual cortex.**
  Wang, A.Y., Kay, K., Naselaris, T., Tarr, M.J., Wehbe, L.
  bioRxiv (2022).
- **Mind Reader: Reconstructing complex images from brain activities.**
  Sikun Lin, Thomas Sprague, Ambuj K Singh.
  arXiv (2022).
- **Natural scene sampling reveals reliable coarse-scale orientation tuning in human V1.**
  Roth, Z.N., Kay, K.*, Merriam, E.P.*
  Nature Communications (2022).
- **Representations in human primary visual cortex drift over time.**
  Roth, Z.N., Merriam, E.P.
  bioRxiv (2022).
- **High-resolution image reconstruction with latent diffusion models from human brain activity.**
  Takagi, Y., Nishimoto, S.
  bioRxiv (2022).
- **Decoding natural image stimuli from fMRI data with a surface-based convolutional network.**
  Zijin Gu, Keith Jamison, Amy Kuceyeski, Mert Sabuncu
  arXiv (2022).
- **The Algonauts Project 2023 Challenge: How the Human Brain Makes Sense of Natural Scenes.**

A.T. Gifford, B. Lahner, S. Saba-Sadiya, M.G. Vilas, A. Lascelles, A. Oliva, K. Kay, G. Roig, R.M. Cichy.
arXiv (2023).

- **Brain-Diffuser: Natural scene reconstruction from fMRI signals using generative latent diffusion.**
  Furkan Ozcelik and Rufin VanRullen.
  arXiv (2023).
- **Neural Selectivity for Real-World Object Size In Natural Images**
  Andrew F. Luo, Leila Wehbe, Michael J. Tarr, Margaret M. Henderson
  bioRxiv (2023)
- **MindDiffuser: Controlled Image Reconstruction from Human Brain Activity with Semantic and Structural Diffusion**
  Yizhuo Lu, Changde Du, Dianpeng Wang, Huiguang He
  arXiv (2023).
- **The transition from vision to language: distinct patterns of functional connectivity for sub-regions of the visual word form area**
  Maya Yablonski, Iliana I Karipidis, Emily Kubota, Jason D Yeatman
  bioRxiv (2023).
- **Modulating human brain responses via optimal natural image selection and synthetic image generation**
  Zijin Gu, Keith Jamison, Mert R. Sabuncu, and Amy Kuceyeski
  bioRxiv (2023).
- **Reconstructing seen images from human brain activity via guided stochastic search**
  Reese Kneeland, Jordyn Ojeda, Ghislain St-Yves, Thomas Naselaris
  arXiv (2023).
- **Sample Reweighting for Label Denoising of Neural Activity Data**
  Dongfang Xu, Rong Chen
  IEEE/EMBS Conference on Neural Engineering (2023)
- **BrainCLIP: Bridging Brain and Visual-Linguistic Representation Via CLIP for Generic Natural Visual Stimulus Decoding**
  Yulong Liu, Yongqiang Ma, Wei Zhou, Guibo Zhu, Nanning Zheng
  arXiv (2023).
- **Brain Captioning: Decoding human brain activity into images and text**
  Matteo Ferrante, Furkan Ozcelik, Tommaso Boccato, Rufin VanRullen, Nicola Toschi
  arXiv (2023).
- **A Unifying Principle for the Functional Organization of Visual Cortex**
  Eshed Margalit, Hyodong Lee, Dawn Finzi, James J. DiCarlo, Kalanit Grill-Spector, Daniel L. K. Yamins
  arXiv (2023).

- **Reconstructing the Mind's Eye: fMRI-to-Image with Contrastive Learning and Diffusion Priors**
  Paul S. Scotti*, Atmadeep Banerjee*, Jimmie Goode, Stepan Shabalin, Alex Nguyen, Ethan Cohen, Aidan J. Dempster, Nathalie Verlinde, Elad Yundler, David Weisberg, Kenneth A. Norman*, and Tanishq Mathew Abraham*
  arXiv (2023).
- **Brain Dissection: fMRI-trained Networks Reveal Spatial Selectivity in the Processing of Natural Images**
  Gabriel H. Sarch, Michael J. Tarr, Katerina Fragkiadaki, Leila Wehbe
  arXiv (2023).
- **Second Sight: Using brain-optimized encoding models to align image distributions with human brain activity**
  Reese Kneeland, Jordyn Ojeda, Ghislain St-Yves, Thomas Naselaris
  arXiv (2023).
- **Brain-optimized deep neural networks of human visual areas learn non-hierarchical representations.**
  St-Yves, G., Allen, E.J., Wu, Y., Kay, K.*, Naselaris, T.*
  Nature Communications (2023).
- **Brain Diffusion for Visual Exploration: Cortical Discovery using Large Scale Generative Models.**
  Andrew F. Luo, Margaret M. Henderson, Leila Wehbe, Michael J. Tarr
  arXiv (2023).
- **Improving visual image reconstruction from human brain activity using latent diffusion models via multiple decoded inputs.**
  Yu Takagi, Shinji Nishimoto
  arXiv (2023).
- **DreamCatcher: Revealing the Language of the Brain with fMRI using GPT Embedding**
  Subhrasankar Chatterjee, Debasis Samanta
  arXiv (2023).
- **What can 1.8 billion regressions tell us about the pressures shaping high-level visual representation in brains and machines?**
  Colin Conwell, Jacob S. Prince, Kendrick N. Kay, George A. Alvarez, Talia Konkle
  bioRxiv (2023)
- **THE ALGONAUTS PROJECT 2023 CHALLENGE: UARK-UALBANY TEAM SOLUTION**
  Xuan Bac Nguyen, Xudong Liu, Xin Li, Khoa Luu
  arXiv (2023)
- **Memory Encoding Model**
  Huzheng Yang, James Gee, Jianbo Shi

arXiv (2023)

- **Applicability of scaling laws to vision encoding models**
  Takuya Matsuyama, Kota S Sasaki, Shinji Nishimoto
  arXiv (2023).
- **A contrastive coding account of category selectivity in the ventral visual stream**
  Jacob S. Prince, George A. Alvarez, Talia Konkle
  bioRxiv (2023).
- **Predicting brain activity using Transformers**
  Hossein Adeli, Sun Minni, Nikolaus Kriegeskorte
  bioRxiv (2023).
- **A Parameter-efficient Multi-subject Model for Predicting fMRI Activity**
  Connor Lane, Gregory Kiar
  arXiv (2023).
- **Expansion of a frontostriatal salience network in individuals with depression**
  Charles J. Lynch, I. Elbau, Tommy Ng, Aliza Ayaz, Shasha Zhu, Nicola Manfredi, Megan A. Johnson, Daniel L Wolk, Jonathan D. Power, E. Gordon, Kendrick Norris Kay, A. Aloysi, Stefano Moia, C. Caballero-Gaudes, L. Victoria, N. Solomonov, E. Goldwaser, Benjamin Zebley, L. Grosenick, J. Downar, F. Vila-Rodriguez, Z. Daskalakis, D. Blumberger, N. Williams, F. Gunning, C. Liston
  bioRxiv (2023).
- **UniBrain: Unify Image Reconstruction and Captioning All in One Diffusion Model from Human Brain Activity**
  Weijian Mai, Zhijun Zhang
  arRxiv (2023).

Terms and Conditions

Before you download the NSD data, please read the data sharing and usage agreement below. You must agree to *all* terms and conditions before accessing the data.

Data Sharing and Usage Agreement

Before I download or process the NSD dataset, I agree to the following terms and conditions:

1. The Center for Magnetic Resonance Research (CMRR) grants me non-exclusive, royalty-free access to download and process this dataset.
2. I will utilize this dataset only for research and educational purposes.
3. I will not distribute this dataset or its components to any other individual or entity.
4. I will require anyone on my team who utilizes these data to comply with this data use agreement.
5. I will neither sell this dataset or its components nor monetize it.
6. I will comply with any rules and regulations imposed by my institution and its institutional review board in requesting these data.
7. The NSD dataset is collected from human subjects and has been de-identified. I will not retrieve or try to retrieve protected health information (PHI) of the human subjects in this dataset. If I incidentally discover PHI information, I will immediately inform the principal investigator.
8. I agree that all presentations and publications resulting from any use of this dataset must cite the relevant work using the suggested citation format listed below.
9. CMRR specifically disclaims any warranties including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The dataset and the encompassing software provided hereunder is on an “as is” basis, and CMRR has no obligation to provide maintenance, support, updates, enhancements, or modifications.
10. In no event shall CMRR be liable to any party for direct, indirect, special, incidental, or consequential damages arising out of the use of this dataset and the accompanying software, even if CMRR has been advised of the possibility of such damage.
11. In addition to the above-listed terms and conditions, I will also comply with federal, state, local, and institutional policies and regulations.

Citation Format

If you make use of the NSD dataset, please cite the NSD data paper:

- Allen, St-Yves, Wu, Breedlove, Prince, Dowdle, Nau, Caron, Pestilli, Charest, Hutchinson, Naselaris*, & Kay*. A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience* (2021).

In addition, please acknowledge the NSD funding sources using wording similar to:

- "Collection of the NSD dataset was supported by NSF IIS-1822683 and NSF IIS-1822929."

How to get the data

Before accessing the data, you must agree to the Terms and Conditions and fill out the NSD Data Access form. After doing so, you are granted full access to the NSD dataset.

AWS

The NSD data are available for download via Amazon Web Services (AWS)'s Simple Storage Service (S3). Thanks to the Public Dataset Program, access to files (request, egress, and transfer costs) is free of charge.

There are several ways to access the data:

- For a light-weight experience (no AWS account necessary), you can directly browse the NSD data files via a simple web interface at https://natural-scenes-dataset.s3.amazonaws.com/index.html
- Alternatively, you can use AWS and browse the NSD data files at https://s3.console.aws.amazon.com/s3/buckets/natural-scenes-dataset
  - Note that you can directly download individual files from AWS via a URL, like: https://natural-scenes-dataset.s3-us-east-2.amazonaws.com/nsddata/experiments/nsd/nsd_screencapture.mp4
- You can use a graphical S3 client (e.g. Cyberduck) to browse and download the data. If using a client, connect to natural-scenes-dataset.s3-us-east-2.amazonaws.com. (Note that in order to connect, you have to supply an access key ID and secret that is associated with your own personal AWS account.)
- For large-scale data downloading, the best bet is probably to use the AWS CLI (command-line interface) which is "rsync"-like.

Note that as an alternative to downloading the data and analyzing on local machines, AWS also provides access to cloud computing resources in the form of EC2 instances.

For your convenience, here is a text listing of all files in the AWS bucket (natural-scenes-dataset).

AWS CLI

The AWS CLI provides convenient programmatic access to the data.

Consider the following example:

```
aws s3 ls s3://natural-scenes-dataset
```

This command simply lists the buckets (folders) available.

As another example:

```
aws s3 cp s3://natural-scenes-
dataset/nsddata/experiments/nsd/nsd_screencapture.mp4
/path/to/local/dir/
```

This command downloads the .mp4 file and places it inside the local directory "dir".

As another example:

```
aws s3 sync --dryrun s3://natural-scenes-dataset/nsddata_betas
/path/to/local/nsddata_betas --exclude "*func1mm*" --exclude "*MNI*" --
exclude "*betas_assumehrf*" --exclude "*betas_fithrf_GLMdenoise_RR*" --
exclude "*betas*session*nii.gz"
```

This command synchronizes the "nsddata_betas" directory from the server to the local "nsddata_betas" directory (located under /path/to/local/). Note that we include the --dryrun flag for cautionary purposes; you should remove the --dryrun flag once you are ready to actually perform the download. Also, note that the command includes several --exclude flags in order to reduce the amount of data downloaded. Specifically, the command excludes the 1-mm preparation of the functional data, the MNI version of the data, beta version 1 ("betas_assumehrf") and beta version 3 ("betas_fithrf_GLMdenoise_RR"), and the NIFTI version of the very large beta files.

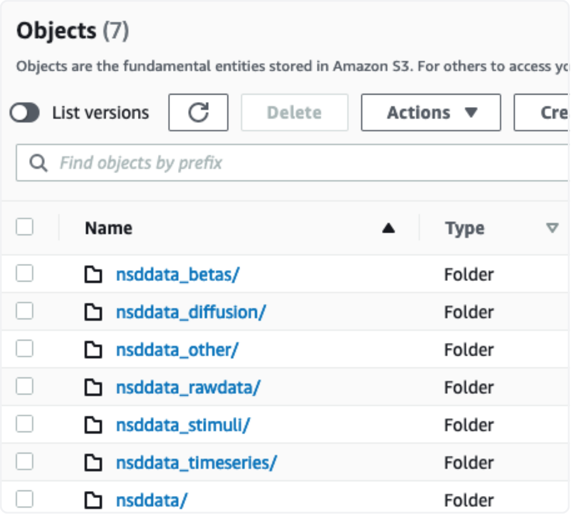
As another example:

```
aws s3 sync --dryrun --exclude "*" --include "*eyedata*" s3://natural-
scenes-dataset/nsddata_timeseries /path/to/local/nsddata_timeseries
```

This command synchronizes the "nsddata_timeseries" directory, excluding ALL files except for the "eyedata" files (using a wildcard mechanism). Remove the --dryrun flag if all looks good.

The AWS CLI includes many customizable flags. Some flags that may be useful include --size-only, --exact-timestamps, and --delete.

Overview of the data



Objects (7)

Objects are the fundamental entities stored in Amazon S3. For others to access yo

□	Name ▲	Type ▽
□	nsddata_betas/	Folder
□	nsddata_diffusion/	Folder
□	nsddata_other/	Folder
□	nsddata_rawdata/	Folder
□	nsddata_stimuli/	Folder
□	nsddata_timeseries/	Folder
□	nsddata/	Folder

Data listing from Amazon S3

Top-level directories

There are several top-level directories:

- **nsddata** (~49 GB) - This is the main directory containing essential data files, including (but not limited to) anatomical data, results of the prf and floc experiments, behavioral data, FreeSurfer subject directories, and ROIs.
- **nsddata_betas** (~8.3 TB) - This very large folder contains estimated fMRI single-trial responses ("betas") for the NSD experiment as well as associated results (e.g. noise ceiling estimates). There are multiple versions of the betas (e.g., betas_assumehrf (b1), betas_fithrf (b2), betas_fithrf_GLMdenoise_RR (b3)). Also, betas are prepared and available in different spaces (e.g., 1.8-mm volume (func1pt8mm), 1-mm volume (func1mm), subject-native surface (nativesurface), fsaverage, MNI).
- **nsddata_stimuli** (~40 GB) - This contains the color natural scene images used in the NSD experiment.
- **nsddata_timeseries** (~3.4 TB) - This very large folder contains the pre-processed fMRI time-series data from which the single-trial betas are estimated. Both 1.8-mm and 1-mm versions are available. In addition, this folder contains information associated with the time-series data, including physiological data (pulse and respiratory), experimental design information (i.e. which images were shown when), motion parameter estimates from the pre-processing of the fMRI data, and eyetracking data.
- **nsddata_other** (~25 GB) - This contains miscellaneous items, including (but not limited to) materials used to run the experiments and original unedited FreeSurfer outputs.

- **nsddata_diffusion** (~200 GB) - This contains derivatives from analyzing the diffusion data. *NOTE: We are currently preparing the final versions of the diffusion derivative files, and they will be made available within a few weeks.*
- **nsddata_rawdata** (~946 GB) - This contains raw data in BIDS format.

The NSD dataset is very large in size. Depending on your needs, you may not need all of the files. For example, if you wish to work from the single-trial betas that we have provided, there is no need to download the raw data nor the pre-processed time-series data. As another example, if you want only the standard-resolution (1.8-mm) preparation of the data, you can exclude the high-resolution (1-mm) preparation, which will result in major space savings (requirement of ~6 times less space). As a third example, if you want only beta version b3, there is no need to also download beta versions b1 and b2.

Held-out data

Some data collected as part of the NSD effort are not yet publicly available. These include the following:

- **nsdimagery (1 scan session)** - Data related to the nsdimagery 7T fMRI experiment are not yet available. These data will be described and released as part of a separate paper effort.
- **nsdsynthetic (1 scan session)** - Data related to the nsdsynthetic 7T fMRI experiment are not yet available. These data will be described and released as part of a separate paper effort.
- ~~**Last 3 NSD core sessions** - Due to the involvement of the NSD data in the Algonauts prediction challenge, the last 3 NSD core scan sessions from each of the 8 NSD subjects are being temporarily held out from public release. The held-out data will be released at a future date.~~ The data are now released (Aug 20 2023).
- **nsdmemory (behavioral experiment)** - Data from the final memory test conducted after completion of the NSD fMRI experiment are now available (released May 27 2023).

For the scan sessions mentioned above, the raw and pre-processed data are held out. ~~However, the behavioral data and experimental design information (including the actual stimuli shown) for the held-out scan sessions are still available. Note that the held-out scan sessions may include instances of images whose responses are available in some other scan session either from that subject or from other subjects.~~

Experiments

This section covers the various experiments conducted for the NSD dataset. This includes details on stimuli and experimental design (e.g. the order in which stimuli were presented).

Acquisition-related information

nsddata/experiments/scanningprotocols/3TB_cvnlab_standardcoil_structural0pt8mm.pdf

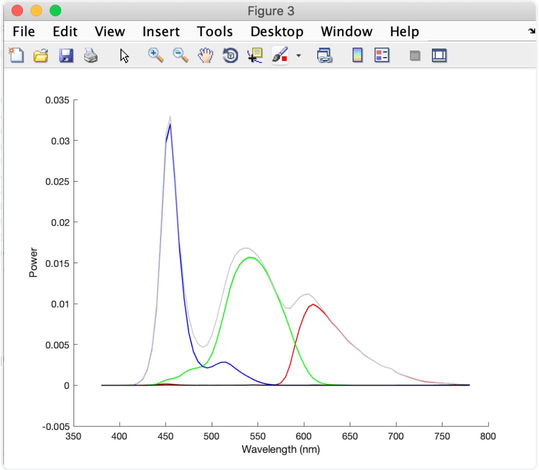
> This is a PDF report of the acquisition protocol for data collected at 3T. (Note: The diffusion scans are named dir98 and dir99, whereas the actually acquired data contain 99 and 100 volumes, respectively. This is because there is an additional b=0 volume at the beginning. Also, note that the actual b-values recorded in the .bval files deviate slightly from the "dialed-in" values of 0, 1500, and 3000.)

nsddata/experiments/scanningprotocols/7TPS_cvnlab_nova1x32_bold1pt8mm.pdf

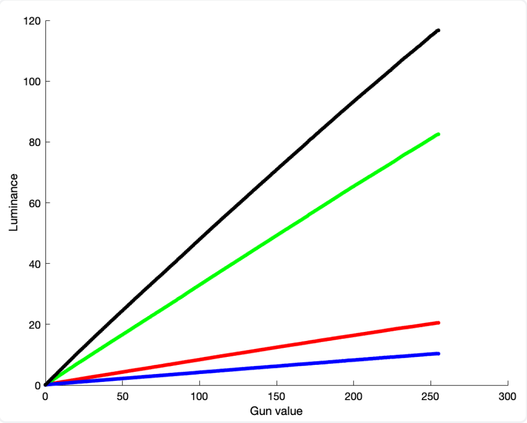
> This is a PDF report of the acquisition protocol for data collected at 7T.

nsddata/experiments/boldscreen/

> This contains code, data, and figures illustrating the spectral power density measurement of the BOLDscreen 32 LCD monitor.



boldscreen/boldscreen_spdmeasurement.p

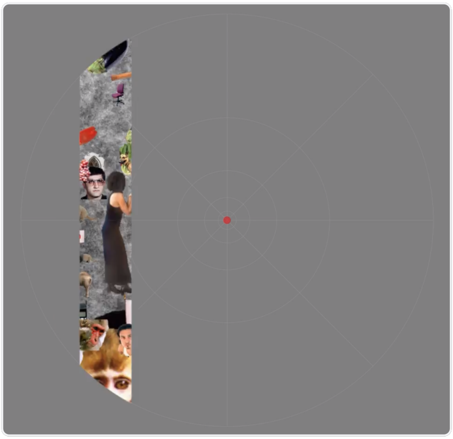


boldscreen/boldscreen_calibration.png

Information regarding the prf experiment

nsddata/experiments/prf/prf_screencapture.mp4

This movie is a screen capture of an example segment of the prf experiment.



screenshot from prf experiment

nsddata/stimuli/prf/RETBAR*

These are sequences of "aperture masks" that correspond to the multibar runs in the prf experiment. The files with "small" in the filename are resized versions of the masks. These resized versions have the aperture masks averaged across consecutive 1-s chunks of the spatiotemporal stimulus, with the exception of the file with "4div3" in the filename, which has been averaged across successive 4/3-s chunks. These aperture masks were used in analyzing the fMRI data from the prf experiment (1-s for the high-resolution preparation; 4/3-s for the standard-resolution preparation). The files without "small" in the filename are the original (unresized and unaveraged) versions of the masks — these masks update at a rate of 15 frames per second. Note that we provide .mp4 versions for convenience; however, the .mp4 files have some (very slight) compression artifacts in them, so be wary when using these files for actual analysis.

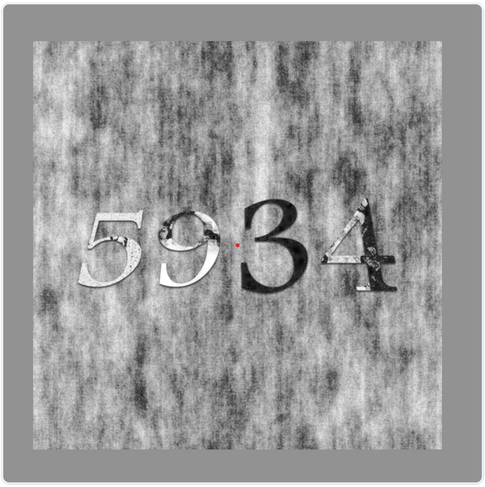
nsddata/stimuli/prf/RETWEDGERINGMASH*

Same information as RETBAR* except corresponding to the wedgering runs in the prf experiment.

Information regarding the floc experiment

nsddata/experiments/floc/floc_screencapture.mp4

This movie is a screen capture of an example segment of the floc experiment.



screenshot from floc experiment

nsddata/experiments/floc/categories.tsv

Names of the 10 categories used in the floc experiment. The order corresponds to the order in the analysis results.

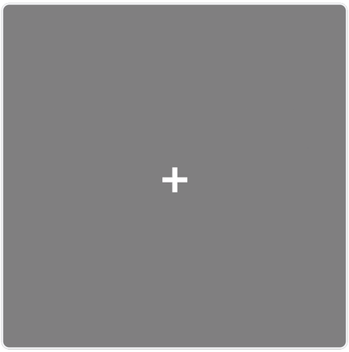
nsddata/experiments/floc/domains.tsv

Names of the 5 domains used in the floc experiment. The domains are in order and have a 1-to-2 relationship to the categories. For example, the first domain consists of the first two categories, the second domain consists of the third and fourth categories, and so on.

Information regarding the resting-state experiment

nsddata/experiments/resting/resting_screencapture.mp4

> This movie is a screen capture of the beginning of the resting-state experiment (type 2, instructed-breath). Notice that after 12 seconds, the cross turns red, which instructs the subject to take a deep breath.



screenshot from resting-

Information regarding the NSD experiment

The 73,000 images used in the NSD experiment are a subset of the COCO images, specifically the 2017 train/val split (see http://cocodataset.org for details). NSD images were selected from the COCO database such that all of the NSD images have “stuff”, “panoptic”, and “coco” annotations. In addition, since the NSD experiment involved square stimulus presentation, we cropped COCO images using a specific method that attempted to minimize loss of semantic information in the images (details provided here: Experiments).

COCO annotations can be accessed on the COCO web site. The following Python notebook is helpful for getting started: https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocoDemo.ipynb

nsddata/experiments/nsd/nsd_stim_info_merged.csv

> This is a comma-separated text file that contains information related to the selection and preparation of the NSD images. After a header row, what follows is one row for each of the 73,000 images used in the NSD experiment.

- Column 1 is the 0-based image number (0-72999).

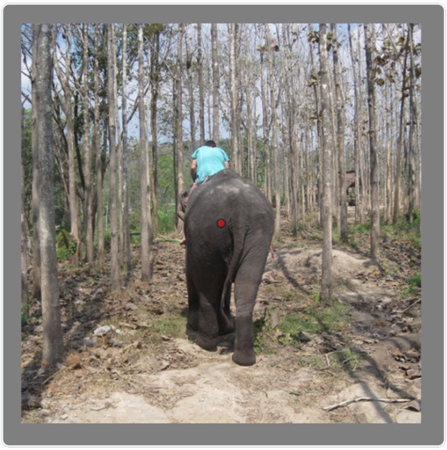
- Column 2 (cocoId) is the ID number assigned to this image in the COCO database.
- Column 3 (cocoSplit) is either “train2017” or “val2017”. The COCO web site designates different splits of images into training and validation sets. The NSD experiment does not involve any use of this designation (such as in the experimental design), but we provide this information just in case it is useful.
- Column 4 (cropBox) is a tuple of four numbers indicating how the original COCO image was cropped. The format is (top, bottom, left, right) in fractions of image size. Notice that cropping was always performed along only the largest dimension. Thus, there are always two 0’s in the cropBox.
- Column 5 (loss) is the object-loss score after cropping. See manuscript for more details, as well as the "Details on crop selection for COCO images" section below.
- Column 6 (nsdId) is the 0-based index of the image into the full set of 73k images used in the NSD experiment. Values are the same as column 1. (Note that in some other cases, 73k IDs are specified as 1-based. Here the IDs are specified as 0-based.)
- Column 7 (flagged) is True if the image has questionable content (e.g. violent or salacious content).
- Column 8 (BOLD5000) is True if the image is included in the BOLD5000 dataset (http://bold5000.github.io). Note that NSD images are square-cropped, so the images are not quite identical across the two datasets.
- Column 9 (shared1000) is True if the image is one of the special 1,000 images that are shown to all 8 subjects in the NSD experiment.
- Columns 10-17 (subjectX) is 0 or 1 indicating whether that image was shown to subjectX (X ranges from 1-8).
- Columns 18-41 (subjectX_repN) is 0 indicating that the image was not shown to subjectX, or a positive integer T indicating that the image was shown to subjectX on repetitionN (X ranges from 1-8; N ranges from 0-2 for a total of 3 trials). T provides the trialID associated with the image showing. The trialID is a 1-based index from 1 to 30000 corresponding to the chronological order of all 30,000 stimulus trials that a subject encounters over the course of the NSD experiment. Each of the 73k NSD images either has 3 trialIDs (if it was shown to only one subject) or 24 trialIDs (if it was shown to all 8 subjects).

nsddata/experiments/nsd/nsd_stim_info_merged.pkl

This contains the same information as the nsd_stim_info_merged.csv file, but is in Python-readable pickle file format (use pandas to read).

nsddata/experiments/nsd/nsd_screencapture.mp4

This movie is a screen capture of one entire run of the nsd experiment.



screenshot from nsd experiment

nsddata/experiments/nsd/nsd_expdesign.mat

Contents:

- <masterordering> is 1 x 30000 with the sequence of trials (indices relative to 10k)
- <basiccnt> is 3 x 40 where we calculate, for each scan session separately, the number of distinct images in that session that have a number of presentations equal to the row index.
- <sharedix> is 1 x 1000 with sorted indices of the shared images (relative to 73k)
- <subjectim> is 8 x 10000 with indices of images (relative to 73k). the first 1000 are the common shared 1000 images. it turns out that the indices for these 1000 are in sorted order. this is for simplicity, and there is no significance to the order (since the order in which the 1000 images are shown is randomly determined). the remaining 9000 for each subject are in a randomized non-sorted order.
- <stimpattern> is 40 sessions x 12 runs x 75 trials. elements are 0/1 indicating when stimulus trials actually occur. note that the same <stimpattern> is used for all subjects.

Note: subjectim(:,masterordering) is 8 x 30000 indicating the temporal sequence of 73k-ids shown to each subject. This sequence refers only to the stimulus trials (ignoring the blank trials and the rest periods at the beginning and end of each run).

Note: All of these indices (in the nsd_expdesign.mat file) are 1-based indices.

nsddata_stimuli/stimuli/nsd/nsd_stimuli.hdf5

This is a single .hdf5 file that contains all images used in the nsd experiment across all subjects. <imgBrick> is 3 channels x 425 pixels x 425 pixels x 73,000 images and is in uint8 format. These images are shown on a gray background with RGB value (127,127,127).

The images in the .hdf5 file constitute the official list of the 73k images. When we use the term ‘73k-ID’, this refers to an index into this list of 73k images (1-indexed).

There is a special common set of 1,000 images, which are a subset of the 73k. Each of the eight subjects sees the shared 1,000 images, as well as 9,000 unique images (with the caveat that some subjects did not complete all 40 NSD scan sessions).

Here is an example of how to use MATLAB to quickly load in the 10239th image.

```
im = permute(h5read('nsd_stimuli.hdf5','/imgBrick',[1 1 1 10239],[3 425 425 1]),[3 2 1]);
```

nsddata/stimuli/nsd/shared1000/

In this folder, there are 1,000 standard RGB .png files (uint8, 425 pixels x 425 pixels x 3 channels). Each file is named "sharedAAAA_nsdBBBBB.png" where AAAA ranges from 1 through 1000 and BBBBB indicates the 73k-ID (1-indexed). These are the 1,000 shared images common to all subjects. Note that the 73k-IDs are in sorted order.

nsddata/stimuli/nsd/special100/

This folder contains a subset of the files in the “shared1000” folder. Of the shared 1,000 images, there is a subset of 515 images that all 8 subjects saw for all 3 trials. From these 515 images, we chose a subset of size 100 in order to maximally span semantic space. These specially chosen 100 images are contained in this folder. These 100 images were used in the nsdmeadows experiment and in the nsdmemory experiment.

nsddata/stimuli/nsd/special3/

This folder contains a subset of the files in the “shared1000” folder. The valence/arousal component of the nsdmeadows experiment involved the special100 images as well as 3 additional images pulled from the subset of 515 images (as described above). These 3 additional images were selected on the criterion of having strong negative valence.

nsddata/stimuli/nsd/shared1000.mp4

> A movie that rapidly shows the shared 1,000 images.

nsddata/stimuli/nsd/shared1000.tsv
nsddata/stimuli/nsd/special100.tsv
nsddata/stimuli/nsd/special3.tsv
nsddata/stimuli/nsd/notshown.tsv

> Simple text files that contain the 73k IDs (1-indexed) that comprise the various sets of images. The "notshown" file indicates 73k IDs of images that were not shown to any NSD subject (due to the fact that not all 8 subjects completed all prescribed sessions).

Details on performance bonuses provided during NSD data acquisition

In each scan session from nsd11–20, the subject earned up to $15 extra bonus. The bonus consisted of $3 for achieving better than the mean performance attained by that subject in sessions nsd01–10 with respect to four metrics. These metrics included the general BOLD quality metric, the intentionally vague “performance metric” (which was actually the performance on easy trials), raw motion, and detrended motion (as described in the NSD data paper). The subject also received $3 for achieving a response rate higher than 99%.

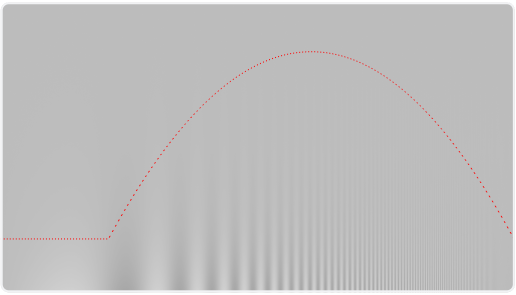
In each scan session from nsd21–30, the subject earned up to $25 extra bonus. The bonus consisted of $5 for agreeing to participate in the resting-state runs conducted in those sessions, $5 if the physiological recordings maintained stability throughout the session, $5 for staying awake and fixated during each resting-state run (thus, $10 in total was possible), and $5 for achieving the “performance metric” above the mean observed for that subject in sessions nsd01–20.

In each scan session from nsd31–40, the subject earned up to $35 extra bonus. The bonus consisted of $20 for participating in that scan session, $5 for achieving response rate higher than 99%, and $10 for agreeing to participate in 1–2 additional miscellaneous scanning runs unrelated to NSD.

Information regarding the nsdpostbehavior experiments

nsddata/experiments/csf/csf_screencapture.png

> This screenshot shows how contrast sensitivity functions were quickly measured.



csf_screencapture.png

nsddata/experiments/flicker/flicker_screencapture1.mp4
nsddata/experiments/flicker/flicker_screencapture2.mp4

> These video captures give a sense of the experiment that assessed the chromatic sensitivity of each subject.

screenshot from flicker

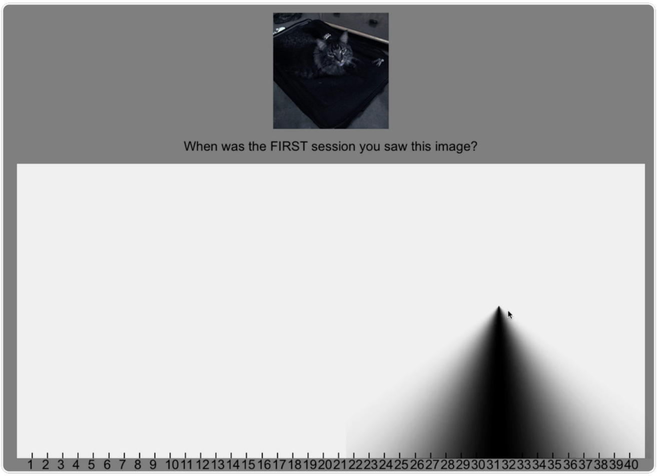
Information regarding the nsdmemory experiment

The experiment presentation code is available at https://github.com/hulacon/nsd-memory

Custom code to analyze the data is available at https://github.com/futingzou/nsdFinalMem

nsddata/experiments/nsdmemory/nsdmemory_screencapture.mp4

This video capture shows what the nsdmemory experiment is like.



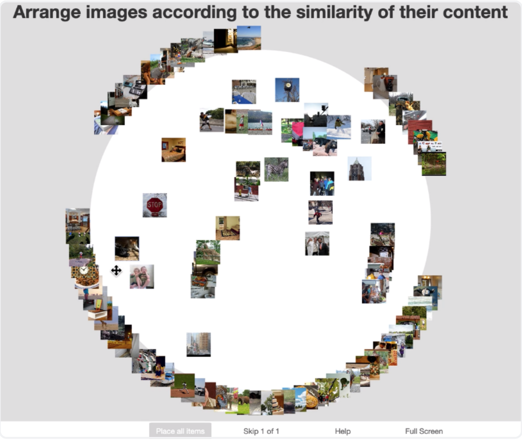
When was the FIRST session you saw this image?

Screenshot from nsdmemory experiment

Information regarding the nsdmeadows experiment

nsddata/experiments/meadows/meadows_screencapture.mp4

This movie shows an example of what subjects experienced during the nsdmeadows experiment which was conducted using the web-based Meadows platform.



Arrange images according to the similarity of their content

screenshot from meadows experiment

Presentation files for experiments

nsddata_other/experimentcode/

> This directory is an archive of materials used to conduct the various experiments in the NSD dataset.

Details on crop selection for COCO images

To select the optimal cropping box for each image, we computed an “object loss” score for each crop. Object loss was defined as the fraction of objects that are cropped by more than 50% of their total pixel count. We used only “thing” annotations to compute object loss. We did not use “stuff” annotations because these are often large and redundant, so that severely cropping them can often result in large object-loss scores but very little change to the semantic content of the image. When calculating object loss we did not include “things” that occupied less than 0.5% of the total pixels in the image. Finally, we imposed a bias toward center crops, selecting left, right, top, or bottom crops if object loss of the center crop exceeded the object loss of the left/right or top/bottom crops by more than 25%. For portrait-oriented images containing people, we always used the top crop, as these images almost always depicted human faces in the upper third of the image.

We examined all cropped images in the “val” portion of the train/val split and rejected any image, regardless of object loss score, for which cropping caused obvious “semantic loss”.

When examining the “val” images we observed the relationship between object loss and semantic loss, and noted several trends that guided our selection/rejection of “train” images.

First, we found that for landscape-oriented images an object-loss score of 0.0 was a reliable indication of negligible “semantic loss”. Thus, we automatically accepted all landscape-oriented images in the “train” set with an object-loss score of 0.0.

Second, we found that for landscape-oriented images crops resulting in 0.0 < object loss < 0.2 occasionally, but not often, induced appreciable semantic loss. Semantic loss occurred when small but key peripheral objects (i.e., a soccer ball) were cropped. We also noted that when images depicted a small number of salient objects, such as people, captions often indicated number of the objects (e.g., “four people sitting around a table”). In these cases crops sometimes made the picture inconsistent with the quantities stated in the captions. Thus we screened all landscape-oriented images in the training set with 0.0 < object-loss < 0.2 for special cases such as these, comparing images to their written captions where necessary.

Third, we found that for portrait-oriented images crops resulting in object loss = 0.0 occasionally, but not often, induced appreciable semantic loss. These images tended to contain a small number of objects with two distinct kinds of terrain in the bottom (e.g., sand, floor) and top (e.g., sky, ceiling) of the image. Cropping the bottom terrain often decontextualized images, for example by reducing “person running on a beach” to “person running”. Many portrait-oriented images depicted tall buildings towering over a semantically meaningful scene such as a flea-market or a street parade. Thus, we screened all portrait-oriented images in the “train” set with object-loss = 0.0 for special cases such as these, comparing images to their written captions where necessary.

After screening the “train” and “val” images, the 73,000 images selected for NSD had a maximum object loss of 0.167 and a median of 0.08.

Raw data

The NSD dataset can be conceptually divided into *raw data* (i.e. data with little or no additional processing) and *prepared data* (i.e. transformations of the raw data that have been performed in order to make the data more accessible and more convenient to use). The pre-processing methods that we used to create the prepared data are formally described in the NSD data paper, with technical details documented in this NSD Data Manual. Most of the remainder of the data manual after this page is specific to the prepared data and not the raw data. Note that a few of the scan sessions are currently held out from the public release (i.e. the raw data for this sessions are listable but not downloadable); see Overview of the data for more information.

nsddata_rawdata/

> This contains the raw data that were collected as part of the NSD effort. It is organized in BIDS format. Note that the data contained here are primarily the structural, functional, and diffusion MRI scans. Some of what might be considered raw data are contained elsewhere in the NSD directory structure. For example, various behavioral measures (see Behavioral data) are provided in nsddata; stimulus images and experimental information are provided in nsddata_stimuli and nsddata (see Experiments); and raw eyetracking data are provided in nsddata_timeseries (see Time-series data).

Naming of the different "tasks" in the raw BIDS data:

- The main NSD experiment is like "task-nsdcore_run-NN" where NN ranges from 01 to 12. The resting-state experiment is like "task-rest_run-N" where N ranges from 1 to 2. In the scan sessions involving resting-state, the chronological acquisition order was:
  - rest1
  - nsd01
  - nsd02
  - nsd03
  - nsd04
  - nsd05
  - nsd06
  - nsd07

  - nsd08
  - nsd09
  - nsd10
  - nsd11
  - nsd12
  - rest2
- The prf experiment is like "task-prfbar_runNN" where NN ranges from 01 to 03 and "task-prfwedge_runNN" where NN ranges from 01 to 03. The floc experiment is like "task-floc_runNN" where NN ranges from 01 to 06. The chronological acquisition order in the prffloc scan session was:
  - prfbar01
  - prfwedge01
  - floc01
  - floc02
  - prfbar02
  - prfwedge02
  - floc03
  - floc04
  - prfbar03
  - prfwedge03
  - floc05
  - floc06
- The nsdsynthetic experiment is like "task-fixation_runNN" where NN ranges from 01 to 04 and "task-memory_runNN" where NN ranges from 01 to 04. Note the fixation and memory were interleaved and acquired in the following chronological order:
  - fixation01
  - memory01
  - fixation02
  - memory02
  - fixation03
  - memory03
  - fixation04
  - memory04
- The nsdimagery experiment is like "task-vis[A-C]", "task-att[A-C]", and "task-img[A-C]_runNN" where NN ranges from 01 to 02. The chronological order was:
  - visA
  - attA
  - imgA01
  - visB
  - attB

- imgB01
- visC
- attC
- imgC01
- imgA02
- imgB02
- imgC02

Time-series data

This section covers pre-processed fMRI time-series data and other measures that exist at the level of the time-series data, which include motion parameter estimates, design matrix information (i.e. which stimulus was shown when), physiological data, and eyetracking data.

Pre-processing of the functional data involved two operations. First, a temporal resampling was performed using a cubic interpolation. The time-series for each voxel was upsampled to either 1 s (high-resolution version) or 1.333 s (standard-resolution version) and in doing so, slice-time differences were corrected. Note that the first time point (after pre-processing) is coincident with the start of the acquisition of the very first volume (i.e. the time of the first RF pulse). Second, a spatial resampling was performed using a cubic interpolation. Each volume was sampled at either 1 mm (high-resolution version) or 1.8 mm (standard-resolution version). This operation corrects for head motion, EPI distortion, gradient nonlinearities, and across-scan-session alignment. Note that no high-pass filtering, nuisance regression, nor units conversion are performed for the pre-processed functional data.

Pre-processed time-series data

nsddata_timeseries/ppdata/subjAA/func*/timeseries/timeseries_BB_runCC.nii.gz

> These are the pre-processed fMRI volumes. The only processing that has been performed for these data is a temporal resampling and a spatial resampling. To save space, a liberal brain mask has been used to zero-out the data for non-brain voxels (same mask for all data from a given subject). "BB" is either prffloc (referring to the scan session in which the prf and floc experiments were conducted) or sessionNN (where NN is the number of the core NSD scan session). Note that scan sessions involving resting-state acquisition consist of 14 runs (as opposed to the typical 12 runs), so in these cases CC ranges from 01 to 14.
>
> For the high-resolution (1-mm) preparation, the data are sampled at 1-s and contain 301 volumes in each run (for the core NSD experiment). For the standard-resolution (1.8-mm) preparation, the data are sampled at 1.333-s and contain 226 volumes in each run (for the core NSD experiment). In both cases, the time associated with the first volume corresponds to the start of the acquisition of the first volume (first RF pulse).

For the prffloc scan session, there are 12 runs in the following order: prfbar, prfwedge, floc, floc, prfbar, prfwedge, floc, floc, prfbar, prfwedge, floc, floc.

Motion parameter estimates

nsddata_timeseries/ppdata/subjAA/func*/motion/motion_BB_runCC.tsv

Motion parameter estimates (SPM style). These reflect rigid-body transformations that indicate how each given fMRI volume is aligned to the reference fMRI volume (which is taken to be the first volume acquired in each scan session).

Note that each fMRI volume is spatially undistorted before estimating the rigid-body motion. Also, note that the motion parameter estimation is done with the first volume as the reference. However, the full pre-processing involves also estimating an affine transformation that aligns the data from each given scan session to the master space defined for each subject; this affine transformation is concatenated with the rigid-body transformations in order to generate the final pre-processed fMRI data.

In the .tsv files, the first 3 columns correspond to translation parameters (mm) and the second 3 columns correspond to rotation parameters (radians). The number of rows matches the number of volumes in the pre-processed time-series data. Positive on the first column means the brain is displaced towards the posterior direction; positive on the second column means the brain is displaced towards the subject's right; positive on the third column means the brain is displaced towards the inferior direction; positive on the fourth column (roll) means the head is twisted such that the nose is fixed and the top of the head goes towards the subject's right; positive on the fifth column (pitch) means the ears are fixed and the head nods up; positive on the sixth column (yaw) means the top of the head is fixed and the head twists such that the nose goes to the subject's left.

Design matrix information

Below, we document design matrix files for the NSD and floc experiments. Note that the pre-preprocessed fMRI data (and motion files) extend for one volume beyond the number of elements contained in the .tsv design files. This is expected behavior (due to how the pre-processing is performed); to achieve correspondence to the .tsv design files, one can simply trim (drop) the trailing volumes of the fMRI (and motion) data.

nsddata_timeseries/ppdata/subjAA/func*/design/design_sessionBB_runCC.tsv

> This is a specification of the design of the NSD experiment. Each file is a column vector of integers, and the number of elements corresponds to the number of volumes in the functional data preparation for a given run. Each element is either N where N is a 73k ID (1-indexed), marking the onset of a presentation of that image, or 0 for all other elements. Note that in order to achieve correspondence to the motion and fMRI time-series data files, the run number CC is 1-12 for scan sessions that contained only NSD runs but is 1-14 for scan sessions that included resting-state runs (in this case, the first (1) and last (14) runs are resting-state runs and the middle 12 runs are the NSD runs). Also, note that in the case of resting-state runs, the .tsv file consists simply of all 0s. Finally, note that the information contained in these .tsv files is redundant with respect to the nsd_expdesign.mat file (see Experiments), but is provided in this .tsv format for your convenience.

nsddata_timeseries/ppdata/subjAA/func*/design/design_floc_runCC.tsv

> This is a specification of the design of the floc experiment. Each file is a column vector of integers, and the number of elements corresponds to the number of volumes in the functional data preparation for a given run. Each element is either N where N is between 1 and 10 marking the onset of one of the 10 categories in the floc experiment, or 0 for all other elements. Note that CC ranges from 1 through 6 (even though the 6 floc runs were acquired chronologically as runs 3, 4, 7, 8, 11, and 12 in the prffloc scan session).

Physiological data

Pulse and respiratory data were collected in NSD scan sessions 21-30 (same as when the primary set of resting-state data are acquired).

nsddata_timeseries/ppdata/subjAA/physio/physio_BB_runCC_DDDD.tsv

> CC ranges from 1 to 14 (chronological acquisition order), and DDDD is either ‘puls’ or ‘resp’, indicating pulse and respiratory data, respectively. Each file consists of a column of numbers (typically numbering 15040 or 15041). The numbers in the .tsv file contain the actual physiological data samples extracted from the Siemens files. It appears that they can be interpreted as close to exactly 50-Hz sampling (more on this below). The data samples start immediately after the AcquisitionTime of the first DICOM volume and end

immediately after the completion of the last DICOM volume. Note that no actual analysis of the physiological data has been performed (aside from the timing extraction).

Notes on how we handled the synchronization of the physiological data and the fMRI data: Our strategy was to assume the accuracy of the LogStartMDHTime and LogStopMDHTime values stored in the Siemens files. We assume that these times correspond to the absolute time of the first and last physiological data samples. We also assume that the data samples come in equally spaced in time. In order to synchronize with the fMRI data, we extracted the AcquisitionTime stored in the DICOM headers of the first volume of each run, and used that time accordingly along with an empirical measurement of the average DICOM duration as recorded by the scanner internal clock.

To interpret the timing of a .tsv file, the following is suggested. Since the TR is 1600 ms and since we acquired 188 volumes in each run, we expect the fMRI acquisition to last from time 0 s through time 300.8 s. Thus, if there are say, 15040 samples in a given .tsv file, we can assume that the time points corresponding to these samples is something like linspace(0,300.8,15040). Moreover, the acquisition times for each of the raw 188 fMRI volumes would correspond to 0, 1.6, 3.2, and so on. In pre-processing, we correct for slice time differences and also upsample the data to either 0.999878 s (for the func1mm preparation) or 0.999878*(4/3) = 1.333171 s (for the func1pt8mm preparation). Thus, the times corresponding to the pre-processed fMRI time-series volumes would be 0, 0.999878, 1.999756, and so on (for func1mm) or 0, 1.333171, 2.666342, and so on (for func1pt8mm).

Occasionally, a physio .tsv file will have a different number of samples (e.g. 15000). It is not clear what the cause of this is (perhaps dropped frames?). We suggest to proceed as described above and assume that the first and last frames still correspond to 0 s and 300.8 s.

Eyetracking data

Note that only the NSD runs (and not the resting-state runs) have associated eyetracking video and data. For this reason, the files from a given scan session may start with run02 and this is correct behavior (since sessions with resting-state data have resting-state runs as run01 and run14).

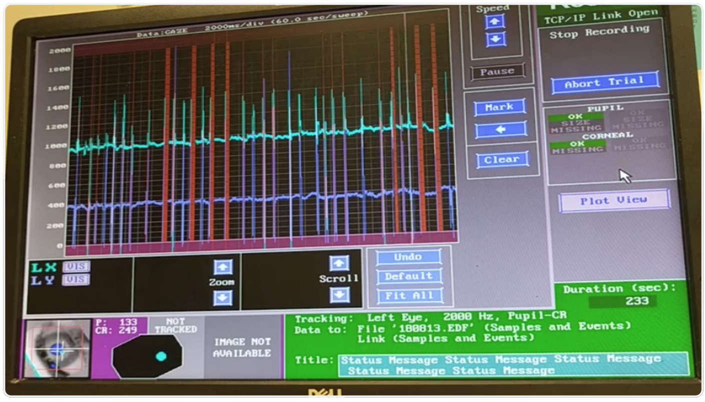
Note that the "CC" in the runCC filename is in chronological acquisition order. If you are matching these to the raw BIDS data, please see the Raw data page for how the naming scheme is designed.

nsddata_timeseries/ppdata/subjAA/eyevideo/eyevideo_BB_runCC.mp4

This is a video capture of the eyetracker computer's display (via a cell phone). This may be a useful complement to the actual eyetracking data (e.g., for informal inspection of the subject's eye and/or for when the eyetracker failed to lock onto the subject's pupil).

All of the .mp4 clips have been cropped to exactly match the fMRI data acquisition duration (i.e., from the start of the very first fMRI volume through the acquisition of the very last fMRI volume in a given run). This cropping was done manually by a human on basis of the audio cues from the video recording; the approximate accuracy of this manual procedure is estimated to be about +/- 1 s. For example, the expected duration of an .mp4 file corresponding to 1 NSD run is 188*1.6 s = 300.8 s.

To protect privacy, the .mp4 files have had the audio stripped (only video is present). The .mp4 files often begin with a few seconds of a black screen — this is correct behavior and is due to video codec issues. When interpreting the timecodes from these video files, be careful to ensure that whatever software you are using is using precise timecodes as opposed to approximate estimates.



Duration (sec):
233

Tracking: Left Eye, 2000 Hz, Pupil-CR
Data to: File '180813.EDF' (Samples and Events)
Link (Samples and Events)

Title: Status Message Status Message Status Message Status Message Status Message

sample frame from one of the videos

nsddata_timeseries/ppdata/subjAA/eyedata/eyedata_BB_runCC.edf

This is the raw eyetracking data file obtained from the EyeLink device. The eyetracker was run at 2000 Hz. The BOLDscreen was run at 1920 x 1080. Note that 8.4° of visual angle (the size of the NSD stimuli) corresponds to 714 pixels on the BOLDscreen.

The utility edf2asc can be used to convert the .edf file to ASCII format. The edf2asc utility is available from SR Research.

Keep in mind that eyetracking data acquisition starts well before actual fMRI data acquisition (approximately 30-90 seconds before). To determine precise synchronization between the eyetracking data and other measures (e.g. the fMRI data), the stimulus computer issues a synchronization message (using PsychToolbox) to the eyetracker computer:

```
Eyelink('Message','SYNCTIME')
```

and this is done right before the actual experiment starts (i.e. right before the display of the very first stimulus frame) and right after the experiment ends (i.e. right after the display of the very last stimulus frame). For example, in a sample .edf file for an NSD run, we find that there is a SYNCTIME message that occurs at timestamp 12829505 and timestamp 13129473. Notice that 13129473-12829505 = 299968, which is interpreted as 299.968 s. The experiment conducted in NSD runs is indeed intended to be 300 s long. If we use the precise time estimates (see ⊟ Technical notes), we find that 0.999878 s * 300 = 299.9634 s, which is quite close to the duration indicated by the eyetracking timestamps. (However, keep in mind that the fMRI data acquisition extends a little bit longer than the actual experiment duration (e.g., 1 NSD run consists of 188 volumes * 1.6 s = 300.8 s). See ⊟ Technical notes for more details.)

nsddata/inspections/eyetrackinginspections/pupil_subjAA_BB_runCC.jpg

This shows the pupil area over time before (top panel) and after preprocessing (bottom panel). Detected blinks and noise shown in orange. Each file shows the data of a single scanning run and subject.

nsddata/inspections/eyetrackinginspections/XY_subjAA_BB_runCC.jpg

This shows the preprocessed gaze positions as 2D scatter plot (top left) and as line plots for horizontal (X, top right) and vertical gaze coordinates (Y, bottom right panel). It further shows a histogram of the Euclidean distances between each recorded gaze position and the median gaze position (bottom left panel). Removed blinks and noise marked in orange.

nsddata_timeseries/ppdata/subjAA/eyedata_preprocessed.mat

This contains the pre-processed eyetracking data. The data is stored in a cell array named “data”. Each cell represents one scanning run. Following fields are included.

- **samples**: Raw data cut to imaging session
- **samples_clean**: Preprocessed data (no blinks & noise)
- **samples_blinks**: Blinks & noise removed from samples_clean
- **filename**: Name of the imported raw data file (after edf-to-ascii conversion)
- **euclDist**: Euclidean distance to median gaze position over time
- **messages**: EDF-file header and recorded messages
- **saccs**: Saccade on-/offsets detected by the Eyelink
- **blinks**: Blink on-/offsets detected by the Eyelink
- **valid_ratio**: Percent valid samples after preprocessing

Note that “samples”, “samples_clean” and “samples_blinks” all contain 2D matrices with time stamps (column 1), horizontal gaze position (column 2), vertical gaze position (column 3) as well as pupil area (column 4) over time (rows).

Informational files

This section covers various informational files and other files relevant to how the NSD data were pre-processed.

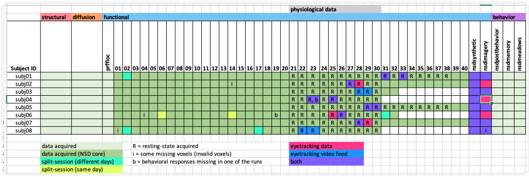
Informational files

nsddata/information/knowndataproblems.txt

This is a detailed, comprehensive list of all known data problems. Most of these problems are very minor, but we are deliberately comprehensive so that the user understands what is in the data.

nsddata/information/nsddatacollection.xlsx

A table that provides an overview of all of the NSD data collected.



																								physiological data																								
	structural	diffusion	functional																																											behavior		
Subject ID			prffloc	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	nsdsynthetic	nsdimagery	nsdpostbehavior	nsdmemory	nsdmeadows
subj01																								R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R							
subj02																	i							R	R	R	R	R	R	R	R	R	R															
subj03																								R	R	R	R	R	R	R	R	R	R															
subj04																								R	R	R, b	R	R	R	R	R	R	R															
subj05																								R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R							
subj06							i														b		R	R	R	R	R	R	R	R	R	R															
subj07																								R	R	R	R	R	R	R	R	R	R															
subj08				i																				R	R	R	R	R	R	R	R	R	R												i			

data acquired
data acquired (NSD core)
split-session (different days)
split-session (same day)

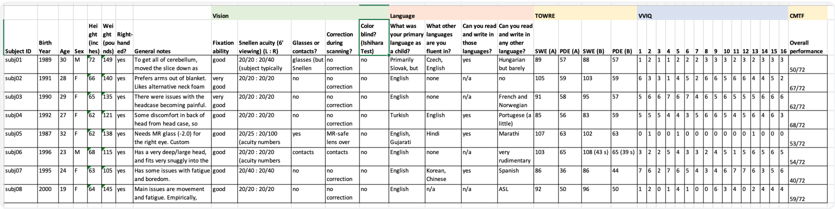
R = resting-state acquired
i = some missing voxels (invalid voxels)
b = behavioral responses missing in one of the runs

eyetracking data
eyetracking video feed
both

nsddatacollection.xlsx

nsddata/information/nsddemographics.xlsx

A table that provides demographic information (age, sex) on the NSD subjects as well as basic information concerning vision- and language-related abilities. The table also provides behavioral data for TOWRE and VVIQ.

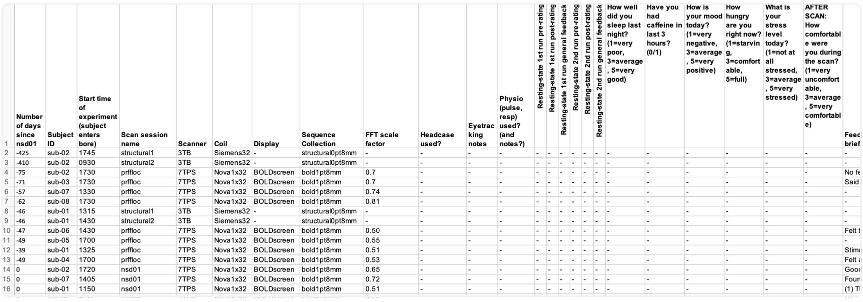


								Vision					Language				TOWRE				VVIQ																CMTF
Subject ID	Birth Year	Age	Sex	Height (inches)	Weight (pounds)	Right-handed?	General notes	Fixation ability	Snellen acuity (6' viewing) (L : R)	Glasses or contacts?	Correction during scanning?	Color blind? (Ishihara Test)	What was your primary language as a child?	What other languages are you fluent in?	Can you read and write in those languages?	Can you read and write in any other language?	SWE (A)	PDE (A)	SWE (B)	PDE (B)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Overall performance
subj01	1989	30	M	72	149	yes	To get all of cerebellum, moved the slice down as	good	20/20 : 20/40 (subject typically	glasses (but Snellen	no correction	no	Primarily Slovak, but	Czech, English	yes	Hungarian but barely	89	57	88	57	1	2	1	1	2	2	2	3	3	3	2	3	2	3	3	3	50/72
subj02	1991	28	F	66	140	yes	Prefers arms out of blanket. Likes alternative neck foam	very good	20/20 : 20/20	no	no correction	no	English	none	n/a	no	105	59	103	59	6	3	3	1	4	5	2	6	6	5	6	6	4	4	5	2	67/72
subj03	1990	29	F	65	135	yes	There were issues with the headcase becoming painful.	very good	20/20 : 20/20	no	no correction	no	English	none	n/a	French and Norwegian	91	58	95	57	5	6	6	7	6	7	4	6	5	6	5	5	5	6	6	6	62/72
subj04	1992	27	F	62	121	yes	Some discomfort in back of head from head case, so	good	20/20 : 20/20	no	no correction	no	Turkish	English	yes	Portugese (a little)	85	56	83	59	5	5	5	4	4	3	5	6	6	5	4	5	6	4	6	3	68/72
subj05	1987	32	F	62	138	yes	Needs MR glass (-2.0) for the right eye. Custom	good	20/25 : 20/100 (acuity numbers	yes	MR-safe lens over	no	English, Gujarati	Hindi	yes	Marathi	107	63	102	63	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	53/72
subj06	1996	23	M	68	115	yes	Has a very deep/large head, and fits very snuggly into the	good	20/20 : 20/20 (acuity numbers	contacts	contacts	no	English	none	n/a	very rudimentary	103	65	108 (43 s)	65 (39 s)	3	2	2	5	4	3	3	2	4	5	1	5	6	5	6	5	54/72
subj07	1995	24	F	63	105	yes	Has some issues with fatigue and boredom.	good	20/40 : 20/40	no	no correction	no	English	Korean, Chinese	yes	Spanish	86	36	86	44	7	6	2	7	6	5	4	3	4	6	7	7	6	3	5	6	40/72
subj08	2000	19	F	64	145	yes	Main issues are movement and fatigue. Empirically,	good	20/20 : 20/20	no	no correction	no	English	n/a	n/a	ASL	92	50	96	50	1	2	0	1	4	1	0	0	6	3	4	0	2	4	4	4	59/72

nsddemographics.xlsx

nsddata/information/nsdsessionlog.xlsx

A table that provides information at the level of individual scanning sessions. Includes information such as time of session, notes on eyetracking and physiology, sleepiness ratings, mood, stress, hunger, general notes on scanning, and subject feedback.



1	Number of days since nsd01	Subject ID	Start time of experiment (subject enters bore)	Scan session name	Scanner	Coil	Display	Sequence Collection	FFT scale factor	Headcase used?	Eyetracking notes	Physio (pulse, resp) used? (and notes?)	Resting-state 1st run pre-rating	Resting-state 1st run post-rating	Resting-state 1st run general feedback	Resting-state 2nd run pre-rating	Resting-state 2nd run post-rating	Resting-state 2nd run general feedback	How well did you sleep last night? (1=very poor, 3=average, 5=very good)	Have you had caffeine in last 3 hours? (0/1)	How is your mood today? (1=very negative, 3=average, 5=very positive)	How hungry are you right now? (1=starving, 3=comfortable, 5=full)	What is your stress level today? (1=not at all stressed, 3=average, 5=very stressed)	AFTER SCAN: How comfortable were you during the scan? (1=very uncomfortable, 3=average, 5=very comfortable)	Feed brief
2	-425	sub-02	1745	structural1	3TB	Siemens32	-	structural0pt8mm	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-410	sub-02	0930	structural2	3TB	Siemens32	-	structural0pt8mm	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-75	sub-02	1730	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	No fe
5	-71	sub-03	1730	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Said
6	-57	sub-07	1330	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.74	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	-62	sub-08	1730	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.81	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
8	-46	sub-01	1315	structural1	3TB	Siemens32	-	structural0pt8mm	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9	-46	sub-01	1430	structural2	3TB	Siemens32	-	structural0pt8mm	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10	-47	sub-06	1430	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.50	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Felt t
11	-49	sub-05	1700	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.55	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12	-39	sub-01	1325	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.51	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Stimu
13	-49	sub-04	1700	prffloc	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.53	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Felt a
14	0	sub-02	1720	nsd01	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.65	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Good
15	0	sub-07	1405	nsd01	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.72	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Foun
16	0	sub-01	1150	nsd01	7TPS	Nova1x32	BOLDscreen	bold1pt8mm	0.51	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	(1) Th

nsdsessionlog.xlsx

nsddata/information/runmetrics.mat

This file contains some data quality metrics that are computed at the level of individual NSD runs. There are two variables:

- ‘runmetrics’ is 8 subjects x 40 sessions x 12 runs x 7 metrics. The seven metrics, in order, are (1) tSNR (this was computed by taking the raw functional volumes, computing the voxel-wise mean of each voxel divided by the voxel-wise standard deviation after quadratic detrending of each voxel’s time-series, and the calculating the median value observed across a liberal whole-brain mask), (2) FD (this was computed on the 1.8-mm version of the pre-processed data by computing the absolute value of the temporal derivative of each of the six motion parameters in each run, computing a weighted sum according to weights [1 1 1 50 50 50], and calculating the mean FD across the volumes in the run), (3) ON-OFF R^2 (for the 1.8-mm version of the data, we fit a simple ON-OFF GLM to the voxel time-series, and we extract the variance explained for each run and compute the median variance explained across voxels in the nsdgeneral ROI), (4) response rate (percentage of trials on which a button was pressed), (5) percent correct (percentage of trials for which the subject pressed the correct response), (6) easy trials (percentage of easy trials (trials that are memory events for an image seen earlier in the scan session) for which the subject pressed the correct response; can be NaN for cases where there are zero easy trials), and (7) number of easy trials (the number of easy trials that actually occurred; this is useful because some runs might have zero or very few easy trials).
- ‘runmetricsRS’ is 8 subjects x 40 sessions x 2 runs x 2 metrics. The two metrics, in order, are tSNR and FD, as described above. When acquired, the resting-state runs were acquired as the very first and very last run in a given session.

Note that because not all subjects participated in all 40 sessions, some of the values in ‘runmetrics’ are NaN. Also, note that because resting-state data were acquired in only certain sessions, some of the values in ‘runmetricsRS’ are NaN. Also, note that for subject 8’s second NSD session, the fourth run was actually split across two distinct scan sessions (on different days); when computing FD, we compensated for this discontinuity (by dropping the appropriate volume), and when computing tSNR, we considered only the first segment of the fourth run. Also, note that for subject 1, session 2, run 2, there was complete MR signal loss for a few volumes in the middle of the run, and for this reason the tSNR values are abnormally low for that run (in the pre-processing of the data, compensation was applied to appropriately deal with this issue).

nsddata/information/b3pcnum_*.tsv

This text file contains a 2D matrix of dimensionality 40 sessions x 8 subjects. The entries indicate the number of nuisance regressors chosen by GLMdenoise for each NSD scan session. NaNs indicate scan sessions that subjects did not participate in.

Files related to pre-processing

nsddata/templates/expert.opts

FreeSurfer configuration file that was used.

nsddata/templates/FreeSurferColorLUT.txt

Information file copied from the FreeSurfer software package.

nsddata/templates/hrfs_*.mat

Each of these files contains a variable ‘hrfs’ that has dimensions time-points x 20 HRFs. The first time point is coincident with trial onset. There are 20 different HRFs comprising the library of HRFs used to estimate voxel-specific HRFs. The ‘func1mm’ version has a sampling rate of 1-s whereas the ‘func1pt8mm’ version has a sampling rate of 1.333-s.

nsddata/templates/hrfparams.mat

Contains HRF parameters (using the parametric function implemented in spm_hrf.m) that were determined by fitting each of the HRFs in the library of HRFs (as described above). The variable ‘params’ is 20 different HRFs x 7 parameters.

nsddata/inspections/hrfparams_example.*

An example MATLAB script that generates an figure illustrating the contents of the hrfparams.mat file.

nsddata/templates/MNI152*

MNI template files copied from fsl-5.0.7/fsl/data/standard. These were used in the pre-processing of the NSD data.

nsddata/templates/T1_2_MNI152_2mm.cnf

Configuration file used in the T1-to-MNI alignment procedure.

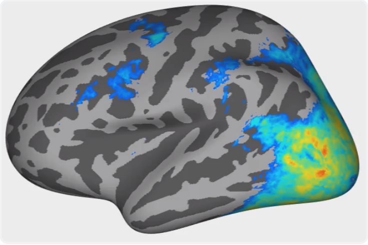
Data inspections

We generated a variety of images and movies that provide a comprehensive look at the quality of the NSD data and pre-processing results.

In the various inspections, note that "sess00" corresponds to the prffloc scan session. Also, note that inspections are included even for the held-out data (now released). For example, for subj01, sess38-sess40 are the 3 held-out NSD scan sessions (now released), sess41 is the nsdsynthetic scan session, and sess42 is the nsdimagery scan session. As another example, for subj08, sess28-sess30 are the 3 held-out NSD scan sessions (now released).

nsddata/inspections/b3noiseceiling.mp4

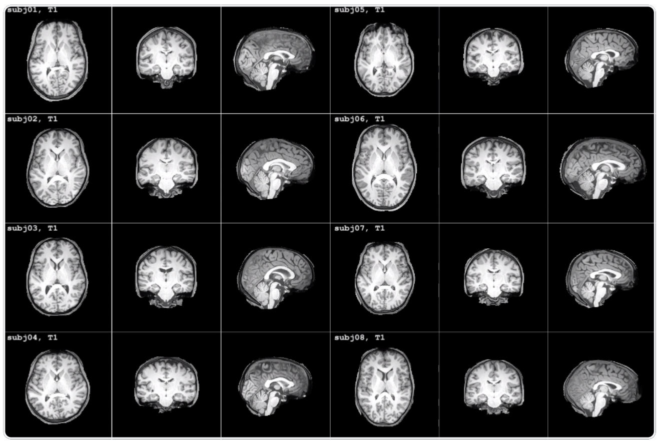
Same as Supplementary Video 10 of the data paper. This shows the group-average b3 noise ceiling results on a rotating brain.



screenshot from

nsddata/inspections/coregistration/T1-T2-EPI.mp4
nsddata/inspections/coregistration/T1-TOF.mp4
nsddata/inspections/coregistration/T2-SWI.mp4

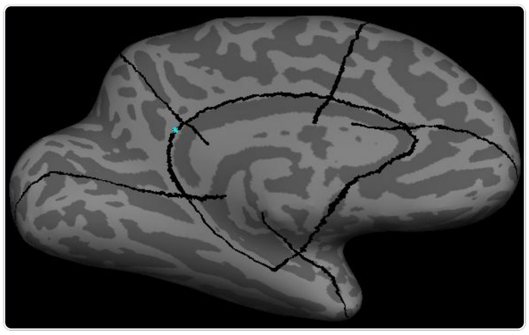
Same as Supplementary Video 1 of the data paper. These show the various modalities collected on the NSD subjects (T1, T2, EPI, TOF, SWI). The figures show the end-result of pre-processing and are all in the common anatomical space set by the T1 volume.



screenshot from T1-T2-EPI.mp4

nsddata/inspections/flattenedsurfaces/

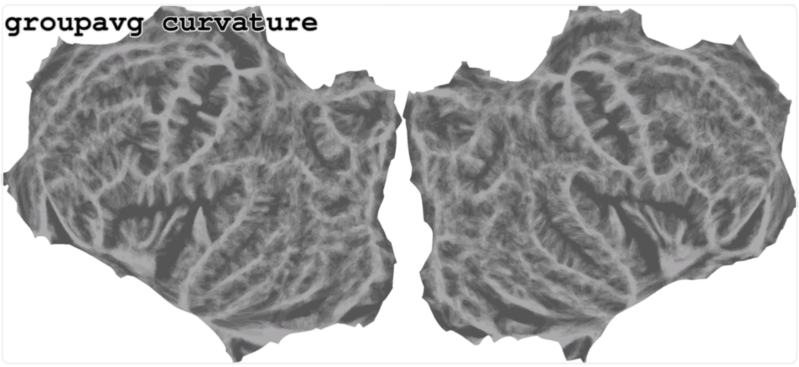
Screenshots showing where cuts were made to the fsaverage surface and each individual NSD subject's surface in order to allow flattening of the cortical surfaces. Cuts were placed in approximately consistent locations across subjects.



subj01_lh_cut.png

nsddata/inspections/fsaveragecheck.mp4

Same as Supplementary Video 3 of the data paper. This movie shows the results of curvature-based fsaverage alignment for the NSD subjects.



screenshot from fsaveragecheck.mp4

nsddata/inspections/functionaltostructural/

These images show, for each NSD subject, the alignment achieved between the EPI data and the T2 anatomical volume. One result is shown for an affine transformation, and another result is shown for the nonlinear ANTS transformation. The ANTS transformation is used for the prepared NSD data.

nsddata/inspections/gradunwarp/

Sample figures illustrating the size of the gradient nonlinearity effect at the two different scanners used (3T and 7T).

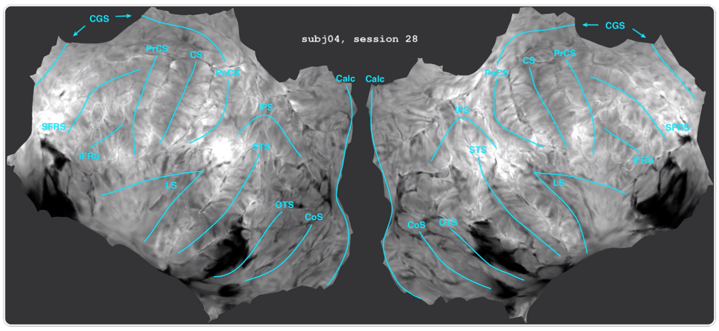
nsddata/inspections/grandmean.mp4

nsddata/inspections/grandmeansurface.mp4

nsddata/inspections/grandR2.mp4

nsddata/inspections/grandR2surface.mp4

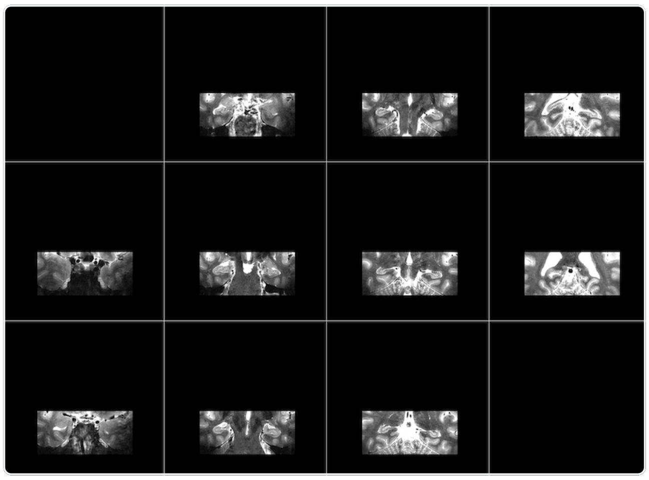
Same as Supplementary Videos 6-9 in the data paper. These movies show the stability of the mean EPI and of BOLD signal strength across all scan sessions for all subjects.



screenshot from grandmeansurface.mp4

nsddata/inspections/HRT2/

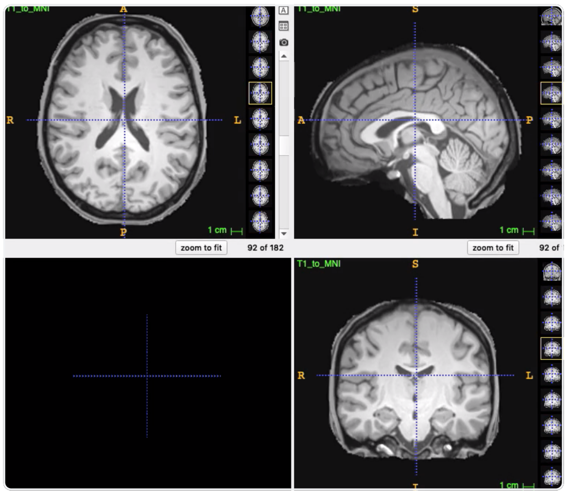
Figures illustrating the alignment achieved for the high-resolution T2 volume acquired for each NSD subject. The figures include the small box used for alignment (mask); the high-res T2 volume masked by this box (masked); the full high-res T2 volume (raw); manually defined MTL labels (rawlabels); resliced volume from the T2 anatomy masked by the box (T2matched_masked); and the full resliced volume from the T2 anatomy (T2matched).



subj01_masked.png

nsddata/inspections/MNIcheck.mp4

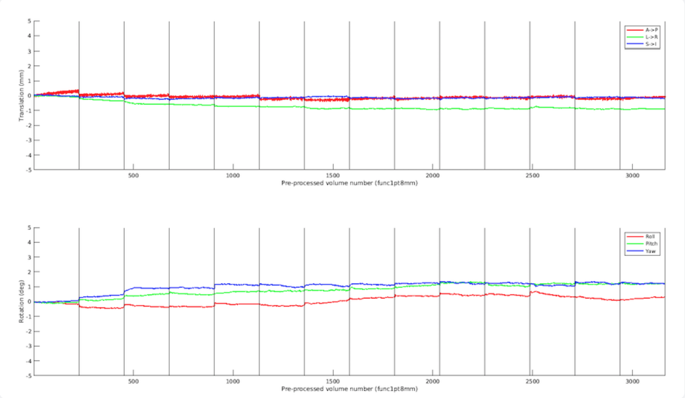
Same as Supplementary Video 4. This shows the quality of the volume-based nonlinear MNI alignment.



screenshot from MNIcheck.mp4

nsddata/inspections/motioninspections*

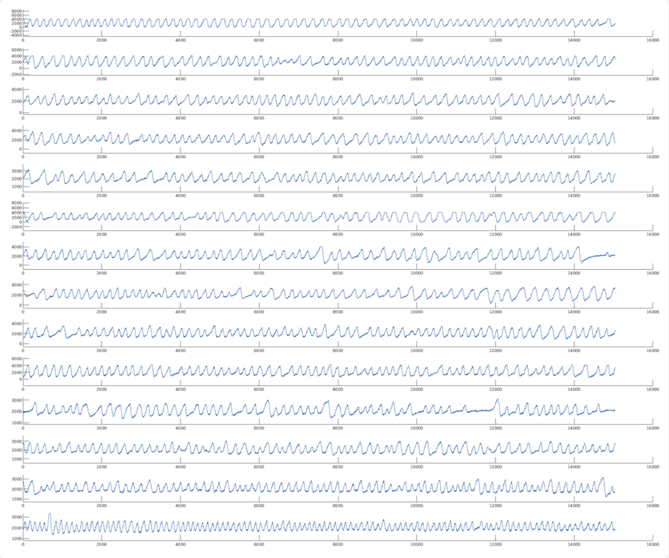
At-a-glance inspection of all motion parameter estimates for all subjects in all sessions.



subj01_sess32.png

nsddata/inspections/physioinspections*

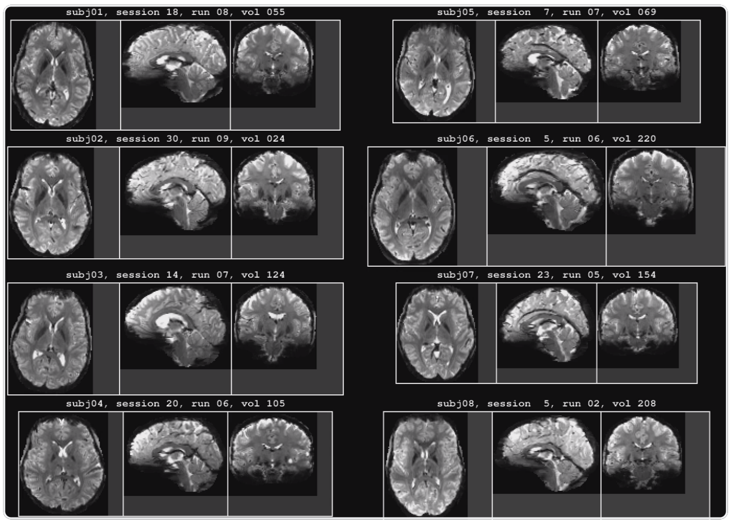
At-a-glance inspection of physiological data for all subjects in all sessions.



physioinspections_resp/subj01_sess25.png

nsddata/inspections/randomscrubbing.mp4

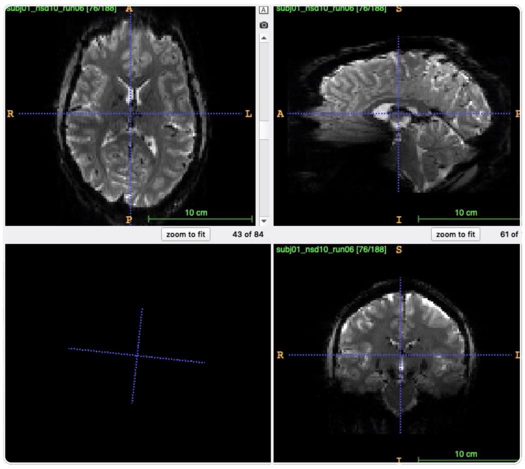
This movie shows an inspection of the overall stability of the pre-processed fMRI data. For each subject, we show 100 volumes randomly picked over time (all runs, all scan sessions).


subj01, session 18, run 08, vol 055
subj05, session 7, run 07, vol 069
subj02, session 30, run 09, vol 024
subj06, session 5, run 06, vol 220
subj03, session 14, run 07, vol 124
subj07, session 23, run 05, vol 154
subj04, session 20, run 06, vol 105
subj08, session 5, run 02, vol 208

screenshot from randomscrubbing.mp4

nsddata/inspections/rawdatamovies/

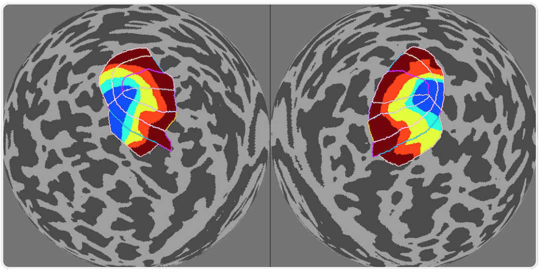
Same as Supplementary Video 5. For one example run (NSD session 10, run 6), we show movies of both the raw fMRI volumes and the pre-processed fMRI volumes for each NSD subject.


subj01_nsd10_run06 [76/188]
A
R
L
P
10 cm
zoom to fit
43 of 84
subj01_nsd10_run06 [76/188]
S
A
P
I
10 cm
zoom to fit
61 of
subj01_nsd10_run06 [76/188]
S
R
L
I
10 cm

screenshot from

nsddata/inspections/rois/

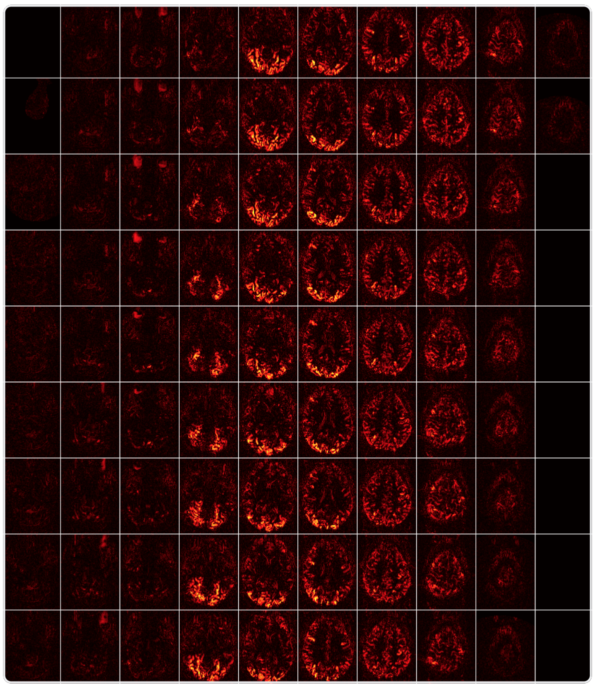
A variety of visualizations of the ROIs provided with the NSD dataset.



prf-visualandecc/subj01_prf-

nsddata/inspections/sessionwise/

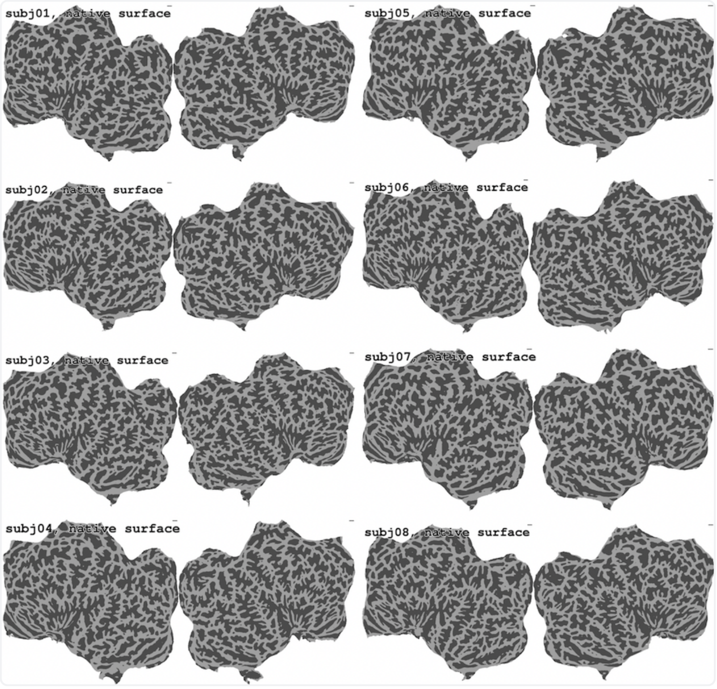
For each NSD subject, we inspect the mean EPI and ON-OFF R^2 results in each scan session (1.8-mm preparation).



subj01/R2_sess09.png

nsddata/inspections/subjectmontages/

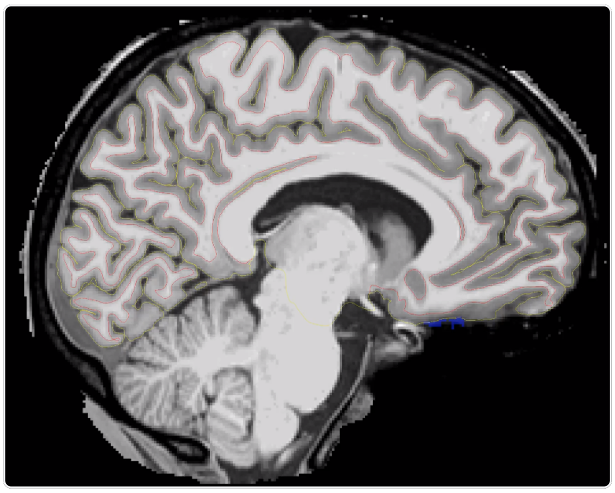
At-a-glance inspection of all NSD subjects' cortical surface reconstructions, including how well they align to the fsaverage template.



native.png

nsddata/inspections/surfaceinspections/

Same as Supplementary Video 2 from the data paper. These movies show 3 views of each subjects' T1 anatomy. The obtained FreeSurfer cortical surface reconstructions are indicated.



screenshot from subj03_sagittal.mp4

nsddata/inspections/surfacevisualizations/

A variety of different surface maps visualizing different aspects of the NSD dataset.

- **bNnc** - These are noise ceilings computed for the different beta versions (b1, b2, b3). The noise ceilings reflect the case of 3 trials being averaged together. The color range is [0 75] with a jet colormap.

- **b3R2** - these are GLM R^2 values for the b3 GLM model. The values range from 0% to 100%. For display, the values are divided by 100, square-rooted, and then visualized using a range of [0 1] with a hot colormap.
- **[corticalsulc,HCP_MMP1,Kastner2015,nsdgeneral,streams]** - These are visualizations of various ROI collections defined on fsaverage.
- **curvature** - These are visualizations of binarized curvature values.
- **mean** - These are visualizations of the mean fMRI signal using a range of [0 2000] with a gray colormap.
- **probmap** - These are visualizations of the fraction of subjects that have a given ROI at each fsaverage vertex. The range is [0 1] with a copper colormap. Values that are equal to 0 are thresholded away.
- **R2** - These are R^2 values from the simple ON-OFF GLM model fitted to the NSD data. Values range from 0 to 100%. For display, values are divided by 100, square-rooted, and then visualized using a range of [0 1] with a hot colormap.
- **signaldropout** - These are regions deemed to suffer from signal dropout (using methods described in the data paper). The maps are binary for each subject. The group result is the average of binary values across subjects. All are visualized using [0 1] with a winter colormap.
- **surfaceimperfections** - These are regions deemed to suffer from cortical surface reconstruction errors (as described in the data paper). The maps are binary for each subject. The group result is the average of binary values across subjects. All are visualized using [0 1] with a winter colormap.
- **valid** - These are indications of which vertices have valid data during the NSD experiment. The binary values are averaged across sessions conducted for a given subject. The color range is [0 1] with a jet colormap.

Note that "fsaverageflat" refers to a flattened version of the fsaverage surface, whereas "subjNNflat" refers to a flattened version of subject NN's native surface.

Behavioral data

This section covers behavioral data acquired for the NSD dataset. Note that some behavioral data is provided in nsddemographics.xlsx (as documented in Informational files).

NSD experiment

nsddata/ppdata/subjAA/behav/responses.tsv

This is a tab-separated text file that contains all behavioral data from the NSD experiment for subject AA. After a header row, what follows is one row for every stimulus trial encountered by the subject. Stimulus trials from different runs and scan sessions are concatenated together.

- Column 1 (SUBJECT) is the subject number (1-8).
- Column 2 (SESSION) is the session number (1-40).
- Column 3 (RUN) is the run number (1-12).
- Column 4 (TRIAL) is the stimulus trial number (1-63 (odd runs) or 1-62 (even runs)). Note that the numbering of stimulus trials ignores (skips over) blank trials.
- Column 5 (73KID) is the 73k ID of the presented image. (Note that here, the 73k IDs are provided as 1-based indices.)
- Column 6 (10KID) is the 10k ID of the presented image. (Note that here, the 10k IDs are provided as 1-based indices.)
- Column 7 (TIME) is the trial start time (i.e. time that the image comes on) as a MATLAB serial date number. The units are days. Time 0 is defined as the beginning (midnight) of the day that the subject's first NSD core scan session took place.
- Column 8 (ISOLD) is 0 (the image is novel) or 1 (the image is old).
- Column 9 (ISCORRECT) is 0 (subject's response was incorrect) or 1 (subject's response was correct).
- Column 10 (RT) is the reaction time in milliseconds (time between trial start time and button-press time).
- Column 11 (CHANGEMIND) is whether this is a trial that involved more than one button press (0 = no, 1 = yes, NaN = no buttons pressed). We score only the final button pressed by the subject.
- Column 12 (MEMORYRECENT) is the number of stimulus trials in between current and most recent presentation. 0 means the current and most recent presentation followed one another (no stimulus trials in between).

- Column 13 (MEMORYFIRST) is the number of stimulus trials in between current and second most recent presentation. If there has been only one previous presentation, this is NaN.
- Column 14 (ISOLDCURRENT) is 0 (the image is novel) or 1 (the image is old) with respect to acting as if the experiment included only the current session.
- Column 15 (ISCORRECTCURRENT) is 0 (subject's response was incorrect) or 1 (subject's response was correct) with respect to acting as if the experiment included only the current session.
- Column 16 (TOTAL1) is the total number of 1s ("novel") pressed during this trial. Will be a non-negative integer.
- Column 17 (TOTAL2) is the total number of 2s ("old") pressed during this trial. Will be a non-negative integer.
- Column 18 (BUTTON) is the button pressed by the subject (1 = button 1, 2 = button 2, NaN = no buttons pressed). Note that there might be multiple buttons pressed during a trial; we score only the final button pressed (and consider the very first of a series of repeated presses of the same button).
- Column 19 (MISSINGDATA) is 0 (button presses were recorded) or 1 (buttons failed to be recorded). This is very rare (it happened in two runs (see knowndataproblems.txt)), and if it happens, it happens at the level of entire runs. In the case that buttons failed to be recorded, note that columns 9-11 and 15-18 are necessarily NaN.

Note that columns 12-13 are NaN for the case of novel images. Note that columns 9-11, 15, and 18 are NaN if no button is pressed on a given trial.

prf experiment

nsddata/bdata/prf/

Results from the prf experiment. For each subject, <results> is [A,B,C] x 6 runs, where A is the total number of color changes, B is the number of hits, and C is the number of false alarms (extra button presses).

floc experiment

nsddata/bdata/floc/

Results from the floc experiment. For each subject, <results> is [A,B,C] x 6 runs, where A is the total number of trials, B is the number of hits, and C is the number of false alarms.

nsdpostbehavior

nsddata/bdata/cmtf/

Results from the Cambridge Memory Test for Faces experiment.

nsddata/bdata/flicker/

Results from the flicker-based assessment of chromatic sensitivity. While maintaining fixation, participants adjusted intensities of red, green, and blue channels on the BOLDscreen display until minimal luminance flicker was perceived. The basic presentation setup was to rapidly switch between two colors (A and B), performing this 15 times in 1 second. Three different trial types were conducted: (1) fix the green channel to 26, ignore blue, and vary the red channel, (2) fix the red channel to 77, ignore green, and vary the blue channel, and (3) fix the green channel to 26, ignore red, and vary the blue channel.

nsdmeadows

nsddata/bdata/meadows/

Results from the nsdmeadows experiment. A set of 100 images were chosen on the basis of their position in a semantic representational space. Participants performed three different behavioural tasks with these chosen stimuli. First, participants were asked to perform a multiple-arrangements task, arranging images according to their similarity with mouse drag and drop operations. Following this, participants performed additional arrangements along a valence scale, and along an arousal scale.

The data is stored in a .json file. The json dictionary has a key for each subject, and in each subject's subdict, there are 11 tasks. The first task is the multiple arrangements task, and this is followed by five separate blocks for valence and five separate blocks for arousal.

Some example code in Python can be found as part of the **nsdcode** repository here:

https://github.com/kendrickkay/nsdcode/blob/master/examples/examples_meadowsdata.py

nsdmemory

nsddata/bdata/nsdmemory/nsdmemory_subj??.[mat,tsv]

These contain the raw data for the nsdmemory experiment.

Post-scanning questionnaires

nsddata/bdata/postnsd/

Results from the questionnaire given to NSD subjects after completion of the NSD experiment and final memory test.

nsddata/bdata/postrestingstate/

Results from the questionnaire given to NSD subjects after completion of resting-state data collection.

Spaces for imaging data

This section describes the spaces used in the prepared NSD data. Understanding this information is important for appropriate handling of the imaging data.

Spaces for the pre-processed data

Each subject has two functional data preparations: “**func1mm**” and “**func1pt8mm**”. This refers to either preparing the data at 1-mm spacing (i.e. upsampling the data) or at 1.8-mm spacing. The 1-mm data has additional fine-scale detail, but is very large in size (approximately 6 times larger than the 1.8-mm data). There is also a difference in temporal resolution in the pre-processed data: the temporal sampling rate (TR) for the two preparations is 1 s and 1.333 s, respectively.

The two functional data preparations are in the same physical space. For example, the two preparations share a common first “corner” voxel (located at anterior, right, inferior) and the data from this voxel are identical across the two preparations. However, the two preparations have slightly different fields-of-view (since the voxel sizes do not evenly divide).

Each subject has three anatomical data preparations: “**anat0pt5**”, “**anat0pt8**”, “**anat1pt0**”. This refers to preparing the anatomical data (e.g. T1, T2) at 0.5-mm, 0.8-mm, and 1.0-mm resolution. All three versions share exactly the same field-of-view and are centered at exactly the same location in space.

The functional and anatomical data are not in register; rather, we have estimated a nonlinear warping for each subject that specifies how the functional data can be registered to the anatomical data (and vice versa). In some cases, we provide convenient versions of the data that have already been mapped (e.g. a version of the T1 that is warped and matched to the functional data).

There are three other spaces of note:

- Some analysis results are prepared in FreeSurfer's surface space, and they are either contained within FreeSurfer directories (e.g. "label") or in directories named **nativesurface**.
- Some analysis results are prepared in **MNI** space. This is achieved based on a nonlinear warp estimated for each subject that takes their 1.0-mm T1 and matches a 1-mm MNI

template.
- Some analysis results are prepared in FreeSurfer's **fsaverage** space. This is achieved based on the curvature-based alignment provided by FreeSurfer.

Note that for the diffusion data, the pre-processed volumes are matched to the anat0pt8 space (0.8-mm).

Basic handling of NSD data files

All NIFTI files in the prepared NSD data are in LPI ordering (the first voxel is Left, Posterior, and Inferior). In addition, all NIFTI files have their origin set to the exact center of the image slab, with one exception being NIFTI files in MNI space (for details, see Technical notes).

We have pre-computed transformations that map between the various spaces, and these transformations are incorporated into the lightweight utility **nsd_mapdata**. This utility transforms user-supplied data from one space to another using interpolation (see Code for details).

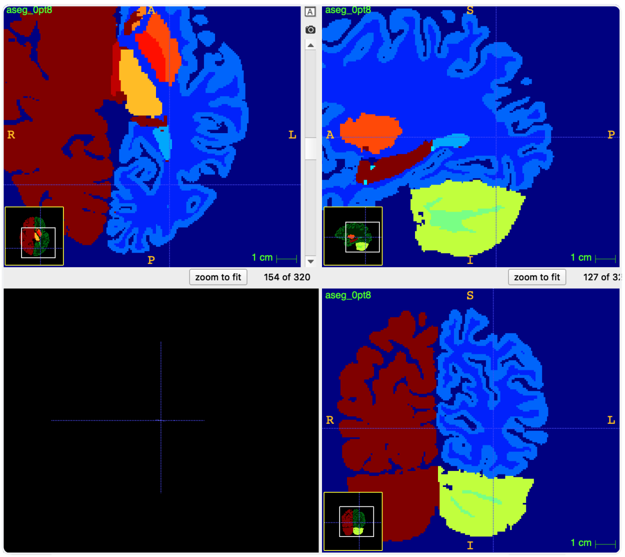
Structural data

This covers anatomical data prepared for the NSD dataset (e.g. T1 and T2 volumes) as well as FreeSurfer outputs.

Anatomical files

nsddata/ppdata/subjAA/anat/aseg_RRRR.nii.gz
nsddata/ppdata/subjAA/func*/aseg.nii.gz

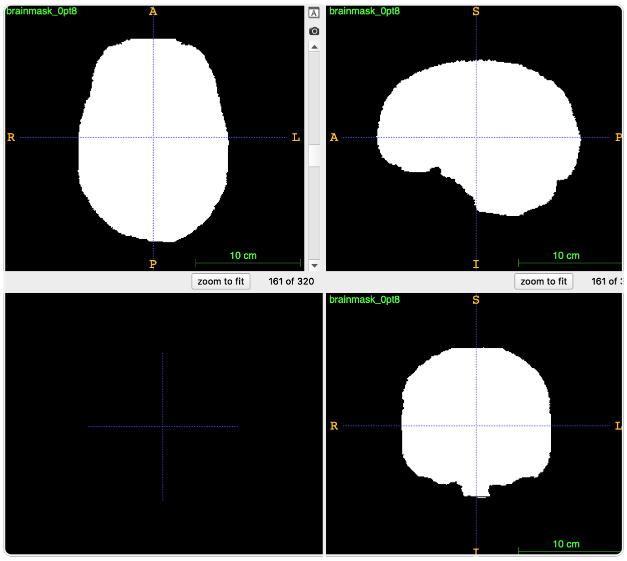
> This is the aseg.mgz (anatomical segmentation) file that is created by FreeSurfer but transformed (using winner-take-all) to the official NSD anatomical spaces and functional spaces. See FreeSurfer's FreeSurferColorLUT.txt file (a copy is in nsddata/templates/FreeSurferColorLUT.txt) for interpretation of what the integer values mean. This information allows you to select white matter, CSF, ventricles, subcortical regions, etc.



subj01/anat/aseg_0pt8.nii.gz

nsddata/ppdata/subjAA/anat/brainmask_RRRR.nii.gz

> The binary brain mask that was used to mask the anatomical volumes (e.g. T1, T2) for de-identification purposes. Note that this brain mask is intentionally liberal so as to not lose brain voxels.

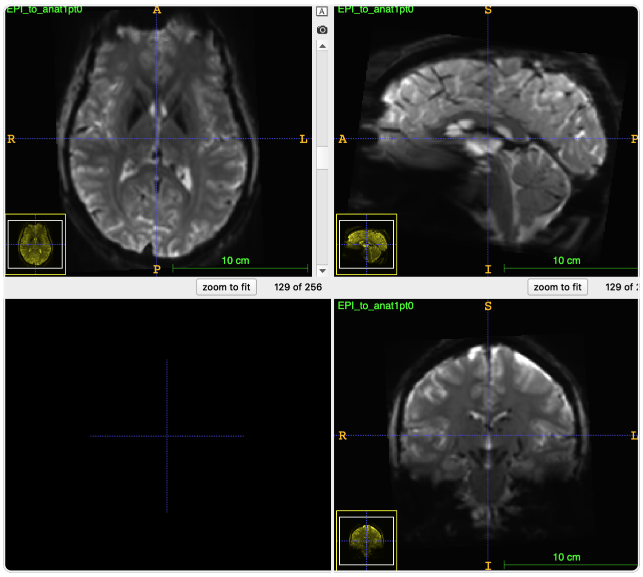


subj01/anat/brainmask_0pt8.nii.gz

nsddata/ppdata/subjAA/anat/EPI_to_anat1pt0.nii.gz

nsddata/ppdata/subjAA/anat/EPI_to_MNI.nii.gz

A version of the mean EPI volume that has been warped to the 1.0-mm anatomical space for that subject, as well as a version that has been warped to MNI space.

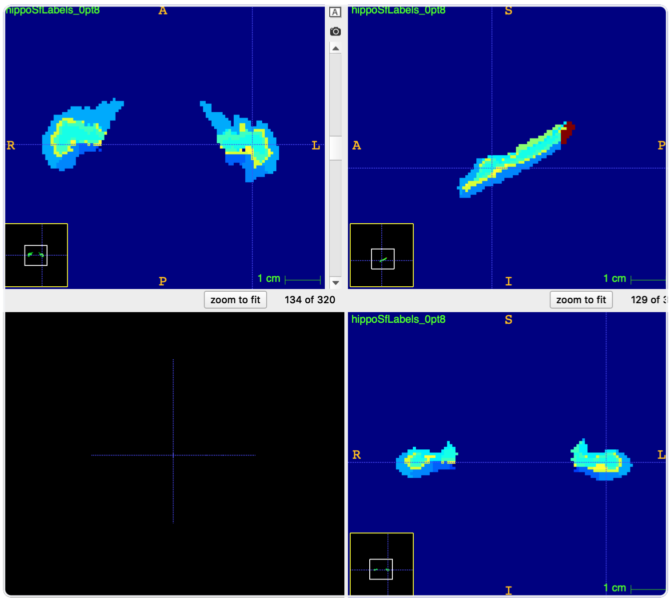


subj01/anat/EPI_to_anat1pt0.nii.gz

nsddata/ppdata/subjAA/anat/hippoSfLabels_RRRR.nii.gz

nsddata/ppdata/subjAA/func*/hippoSfLabels.nii.gz

This is the automated FreeSurfer hippocampal segmentation that has been transformed (using winner-take-all) to the official NSD anatomical spaces and functional spaces.

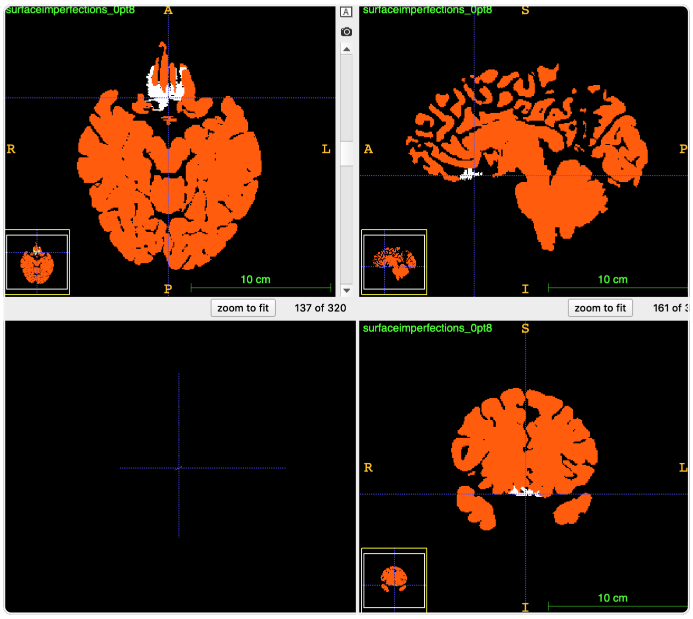


subj01/anat/hippoSfLabels_0pt8.nii.gz

nsddata/ppdata/subjAA/anat/surfaceimperfections_RRRR.nii.gz

nsddata/freesurfer/subjAA/label/[lh,rh].surfaceimperfections.mgz

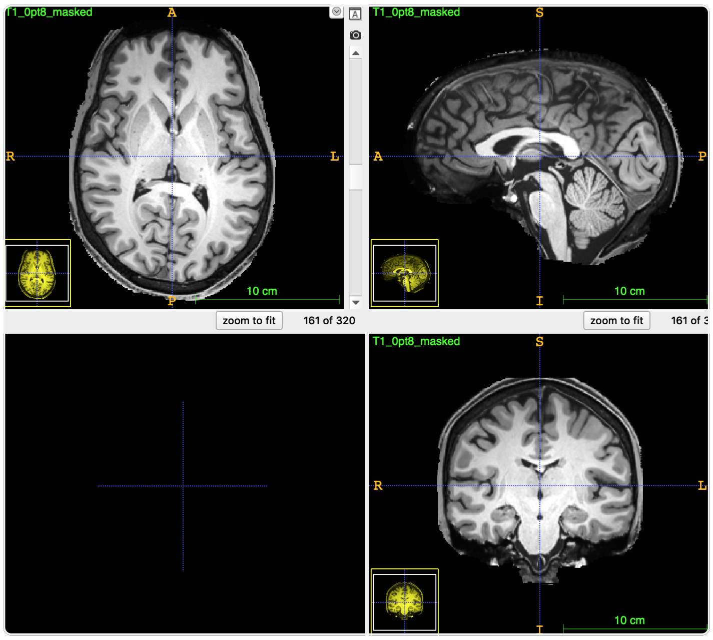
This shows locations of errors in FreeSurfer cortical surface reconstructions, as determined by visual inspection. There are generally few errors, and these errors occur in stereotypical locations (see NSD data paper).


surfaceimperfections_0pt8
A
R
L
P
10 cm
zoom to fit
137 of 320
surfaceimperfections_0pt8
S
A
P
I
10 cm
zoom to fit
161 of 3
surfaceimperfections_0pt8
S
R
L
I
10 cm

subj01/anat/surfaceimperfections_0pt8.nii.gz

nsddata/ppdata/subjAA/anat/[T1,T2,SWI,TOF]_RRRR_masked.nii.gz

The official T1, T2, SWI, and TOF volumes for a given subject. These volumes have been masked. The different resolutions of the volumes all share the exact same field-of-view and exact same center.



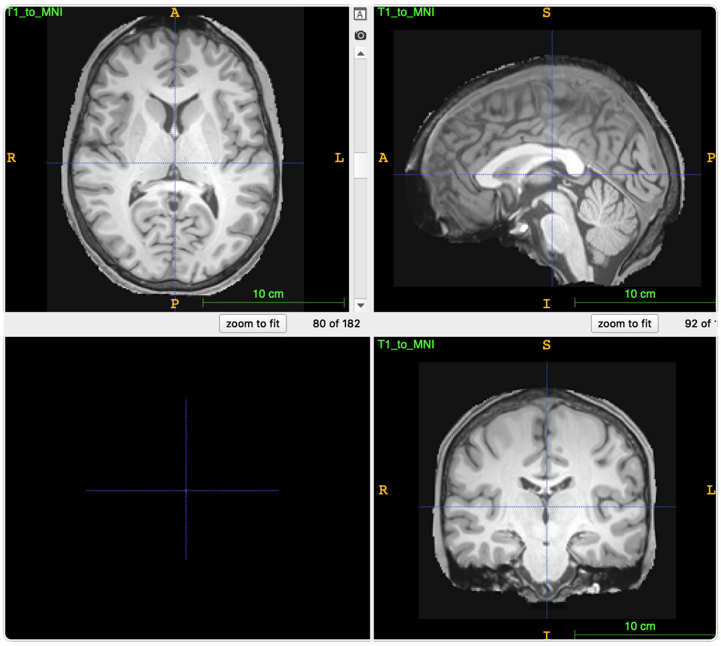
subj01/anat/T1_0pt8_masked.nii.gz

nsddata/ppdata/subjAA/anat/DWI_RRRR.nii.gz

Here, we took the pre-processed diffusion data, extracted the b0 volumes, averaged the b0 volumes within Run 1 and within Run 2, and then averaged the two averages together, producing a single volume (at 0.8-mm resolution). This volume was then resampled to different resolutions (in the same manner as the other anatomical volumes).

nsddata/ppdata/subjAA/anat/[T1,T2,SWI,TOF]_to_MNI.nii.gz

Versions of the volumes that have been warped to MNI space.



subj01/anat/T1_to_MNI.nii.gz

FreeSurfer files

nsddata/freesurfer/subjAA

This is the final FreeSurfer directory for subject AA, reflecting the result of manual edits to the tissue segmentation.

In running FreeSurfer, a 0.8-mm T1 volume was provided to FreeSurfer and the '-hires' flag was used. Also, we have performed additional FreeSurfer-related processing, which created additional files. The changes include (1) creating layerB1, layerB2, and layerB3 surfaces which correspond to 25%, 50%, and 75% of the distance from the pial surface to the white-matter surface; (2) creating semi-inflated surfaces (e.g. ?h.semiinflated + ?h.sulcsemiinflated); and (3) creating flattened cortical surfaces (e.g. ?h.full.flat.patch.3d)). Also, note that the manually edited subject directory has modified files: for example, the brainmask.mgz file has had "holes" put into it (to aid in the surface reconstruction process).

Note that FreeSurfer has a built-in fsaverage flattened surface called [lh,rh].cortex.patch.flat. This is distinct from the flattened cortical surfaces described above. Note that the two flattened surfaces are rotated differently, so one may need to rotate the surfaces to a more canonical orientation for visualization purposes. Also, note that the full-cortex flattened surfaces remove substantial cortex near the midline (e.g. cingulate cortex), so be careful when interpreting results.

Besides the typical FreeSurfer outputs, the subject directories also contain a number of NSD-specific data files. These include ROI files and results from the prf, floc, and NSD experiments.

nsddata/freesurfer/fsaverage

The FreeSurfer special "fsaverage" subject. Again, additional files are present in this directory, reflecting additional FreeSurfer-related processing that we have performed.

nsddata/freesurfer/fsaverage[_sym,3,4,5,6]

These are standard FreeSurfer directories. No additional files are present in these directories.

nsddata_other/freesurferoriginals/subjAA_original

This is the original, non-edited FreeSurfer output for subject AA. Note that the surfaces in the edited and original versions are not compatible with one another, given that they have different numbers of vertices.

nsddata_other/freesurferoriginals/subjAA_repBB

This the raw FreeSurfer output produced when run on an individual T1 acquisition (the BBth one) for subject AA. (Please note that the individual T1 acquisitions processed here are **after** the co-registration procedure; hence, all of the results should be directly spatially comparable.) The call to FreeSurfer was the same as the original FreeSurfer call, except that the -hippocampal-subfields option was run with -T1 not -T1T2. Furthermore, no additional FreeSurfer-related processing was run for these directories.

These individual T1 FreeSurfer directories may be useful for assessing the reliability of FreeSurfer outputs for individual subjects. However, note that the final FreeSurfer directory (freesurfer/subjAA) reflects manual edits to the segmentation. Thus, a more appropriate comparison may be to use the freesurferoriginals/subjAA_original directory.

Functional data (general)

This covers general files that pertain to the preparation of the fMRI data.

nsddata/ppdata/subjAA/func1mm

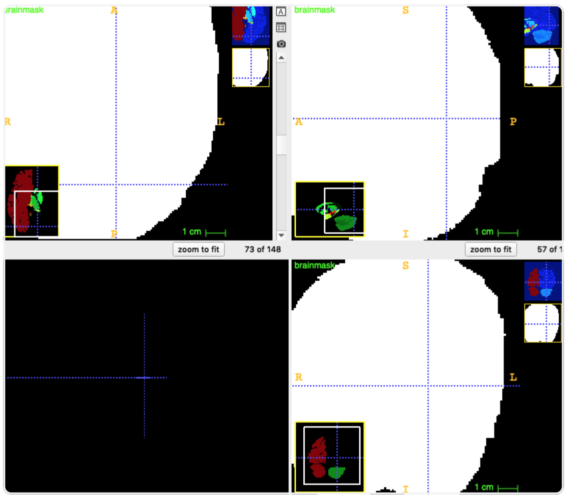
This contains the high-resolution 1-mm preparation of the fMRI data.

nsddata/ppdata/subjAA/func1pt8mm

This contains the standard-resolution 1.8-mm preparation of the fMRI data.

nsddata/ppdata/subjAA/func*/brainmask.nii.gz

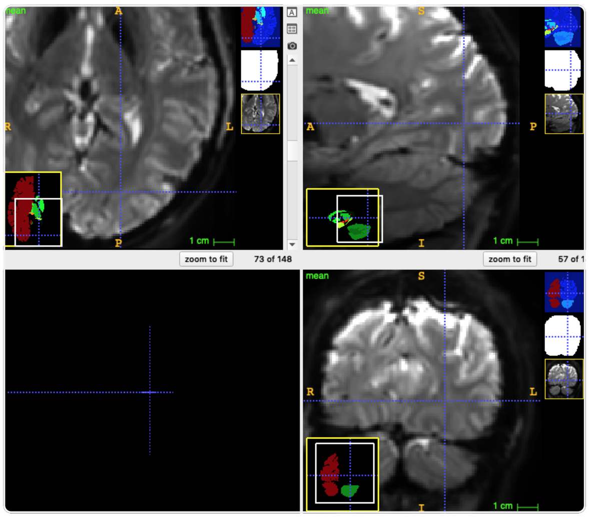
The binary brain mask that was used to mask the betas (in order to save disk space). Note that this brain mask is intentionally liberal so as to not lose brain voxels.



subj01/func1mm/brainmask.nii.gz

nsddata/ppdata/subjAA/func*/meanBBBB.nii.gz
nsddata/freesurfer/subjAA/label/[lh,rh].mean.mgz

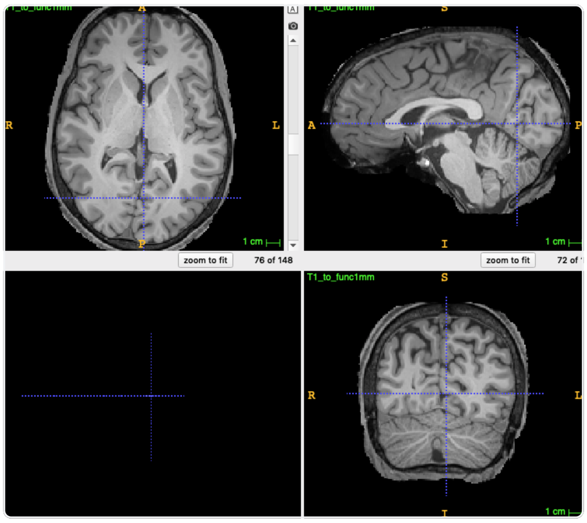
This is the mean of all of the pre-processed volumes in BBBB for subject AA. BBBB can be '' (i.e. mean.nii.gz) which means averaged across all of the NSD core scan sessions; or 'FIRST5' which means averaged across the first 5 NSD core scan sessions (this version was used in various co-registration procedures); or '_sessionNN' which means the Nth NSD core scan session; or '_prffloc' which means the prffloc scan session.



subj01/func1mm/mean.nii.gz

nsddata/ppdata/subjAA/func*/[T1,T2,SWI,TOF]_to_func*.nii.gz

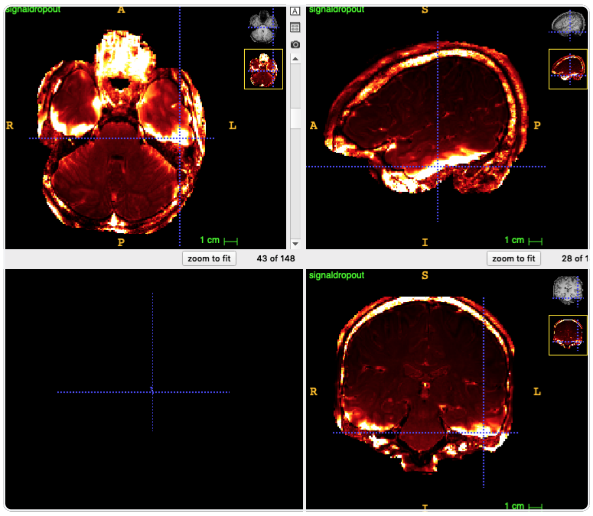
This is a version of the subject's anatomical volumes that has been matched to the functional data space.



subj01/func1mm/T1_to_func1mm.nii.gz

nsddata/ppdata/subjAA/func*/signaldropout.nii.gz
nsddata/ppdata/subjAA/func*/signaldropout_masked.nii.gz
nsddata/freesurfer/subjAA/label/[lh,rh].signaldropout.mgz

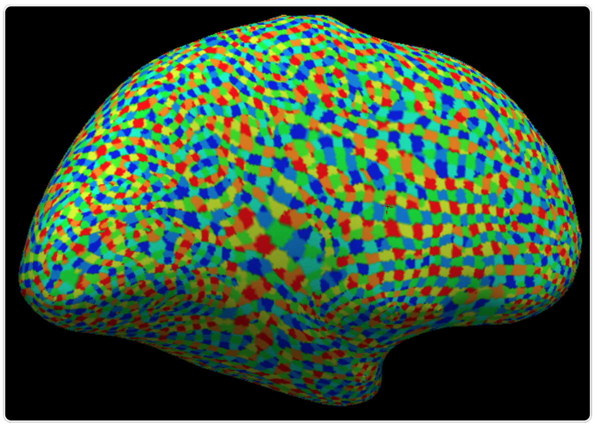
These are volumes that indicate areas of EPI signal dropout. They are computed by dividing the T2 (T2_to_func*.nii.gz) volume by the mean pre-processed EPI volume (mean.nii.gz) and then scaling the resulting volume such that 1 corresponds to a reasonable threshold that divides "good" EPI voxels from "bad" ones (see paper for details). The former volume (signaldropout.nii.gz) is not masked, whereas the latter volume (signaldropout.nii.gz) is masked according to the aseg.nii.gz file (any voxel that is zero in aseg is set to zero). The masked volume can be useful for ignoring voxels outside of the brain.



subj01/func1mm/signaldropout.nii.gz

nsddata/freesurfer/subjAA/label/[lh,rh].surfacevoxels*.mgz

Results of the ‘surface voxels’ visualization technique (Kay et al., *NeuroImage*, 2019). We sampled 1-, 2-, and 3-mm volumetric test patterns onto surface vertices using nearest-neighbor interpolation.



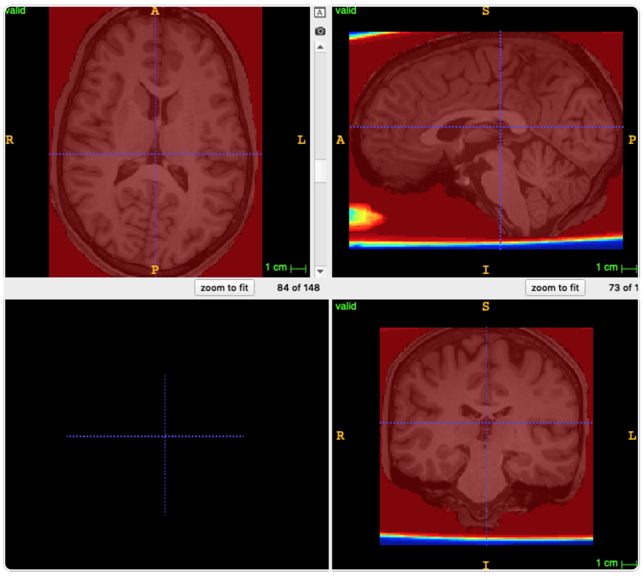
subj05/label/lh.surfacevoxels_layerB3.mgz (3-

nsddata/ppdata/subjAA/func*/validBBBB.nii.gz
nsddata/freesurfer/subjAA/label/[lh,rh].valid.mgz

This is a binary mask indicating which voxels contain valid data in BBBB for subject AA. (Invalid data occurs when motion or spatial distortion cause missing data for voxels.)

BBBB can be '' (i.e. valid.nii.gz) which means average of valid mask across all NSD core scan sessions; or '_sessionNN' which means the Nth NSD core scan session; or '_prffloc' which means the prffloc scan session.

In valid.nii.gz, the values consists of fractions between 0 and 1. For the most part, data were acquired for the entire brain in every session. However, there are a few sessions in which a small amount of brain was cut off. These cases can be detected by finding voxels in valid.nii.gz with values less than 1.0.



subj01/func1mm/valid.nii.gz (superimposed on

Functional data (pRF, fLoc)

This covers analysis results for the pRF and fLoc experiments conducted in the initial 7T prffloc scan session.

Results from the prf experiment

The pre-processed fMRI time-series data from the prf experiment (6 runs, 300-s each) was fit with a pRF model using nonlinear optimization (the CSS model; see Kay et al., *J Neurophys*, 2013). Note that the model was constrained to have non-negative gain.

The results of the fitting are provided in the following files. Note that in each of the files, NaN values are possible and indicate either missing data or voxels outside of the brain mask.

Both volume-based and surface-based versions of the results are available. Volume-based results are located at

nsddata/ppdata/subjAA/func*/prf_BBB.nii.gz

and surface-based results are located at

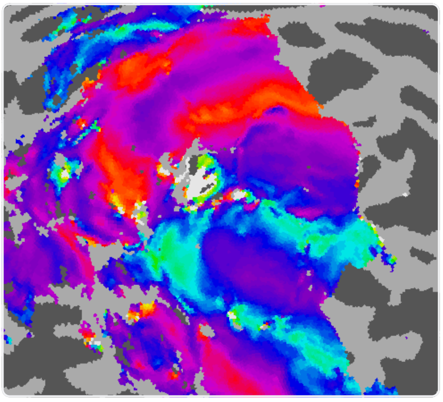
nsddata/freesurfer/subjAA/label/[lh,rh].prfBBB.mgz

where BBB refers to different quantities. To create surface-based versions, we take the 1-mm volume-based prf results and map them to the left and right hemisphere cortical surfaces (linear interpolation onto the 3 depth surfaces, average across depth).

Below, we document each of the BBB quantities.

angle

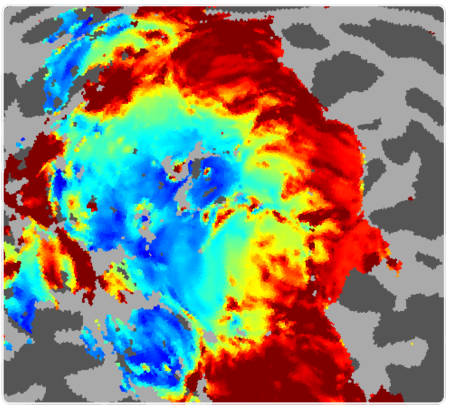
This contains, for each voxel, the polar angle of the pRF center. Values are between 0 and 360 (Cartesian coordinate system where 0 corresponds to the right horizontal meridian, 90 corresponds to the upper vertical meridian, etc.) and are in units of degrees. NaNs exist in the case that pRF eccentricity is exactly 0.



subj01/label/lh.prfangle.mgz

eccentricity

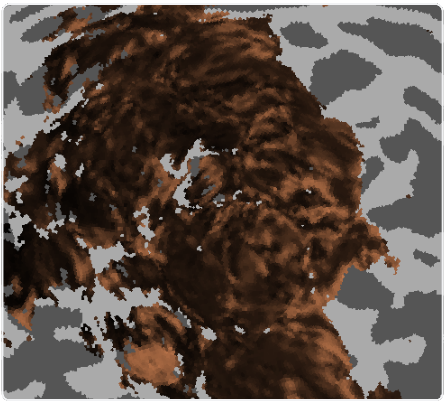
This contains, for each voxel, the eccentricity of the pRF center. Values are non-negative and are in units of degrees of visual angle. Values are capped at 1000.



subj01/label/lh.prfeccentricity.mgz

exponent

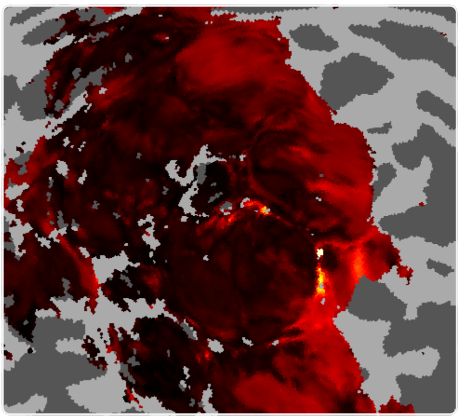
This contains, for each voxel, the fitted pRF exponent. Values are non-negative and are capped at 1000.



subj01/label/lh.prfexponent.mgz

gain

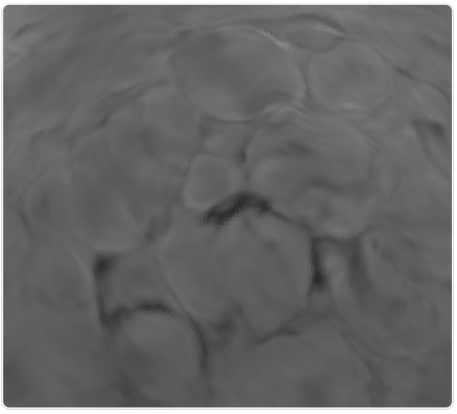
This contains, for each voxel, the gain of the pRF model. The interpretation is that this is the amplitude reached for a stimulus that completely covers the full extent of the pRF. Values are non-negative, in percent signal change, and are capped at 1000%.



subj01/label/lh.prfgain.mgz

meanvol

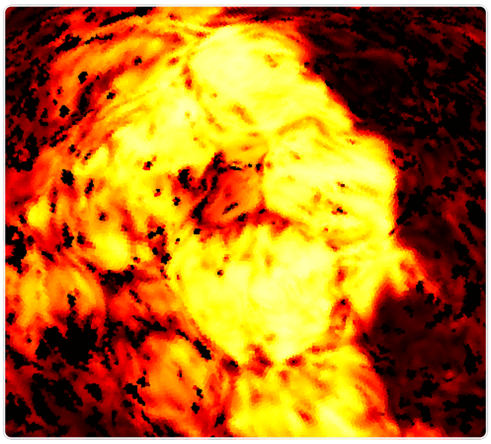
This contains, for each voxel, the mean EPI intensity (in the prf data). Values are in raw scanner units and generally fall in the range 0 to 4095.



subj01/label/lh.prfmeanvol.mgz

R2

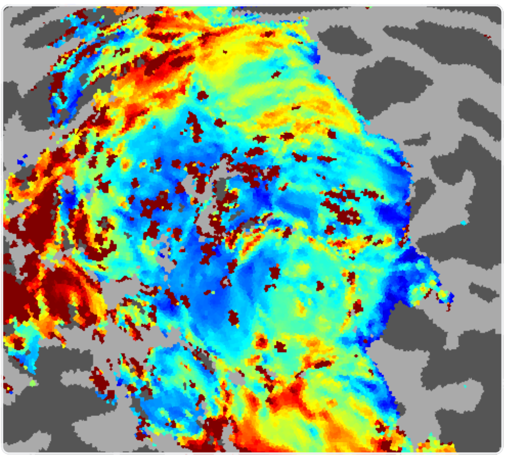
This contains, for each voxel, the variance explained by the pRF model. Values generally lie between 0% and 100%, but other values are possible. Values are capped at the low end at -1000%.



subj01/label/lh.prfR2.mgz

size

This contains, for each voxel, the estimated pRF size. Values are non-negative and are in units of degrees of visual angle. The definition of pRF size is one standard deviation of a Gaussian that describes the response of the model to point stimuli. Note that this definition of pRF size takes into account the nonlinear summation behavior of the pRF model and is not the same as the “sigma” parameter used in the pRF model (see additional notes below). Values are capped at 1000 deg.



subj01/label/lh.prfsize.mgz

Technical notes on pRF size, sigma, and exponent

- The stimulus apertures were prepared at 200 pixels x 200 pixels (corresponding to 8.4° x 8.4° visual extent) and were subsequently used in the model fitting.
- The underlying pRF model used to fit the data is given as follows:

```
modelfun = @(params,stim) ...
  posrect(params(4)) * ...
  (stim * vflatten(makegaussian2d(200,params(1),params(2),abs(params(3)),abs(params(3))) / (2*pi*abs(params(3))^2))) .^ posrect(params(5));

```

  - In the above code, params(4) is the gain parameter, stim refers to the apertures (formatted as volumes x pixels*pixels), params(1) is the position of the pRF center expressed in terms of row indices (1-200 corresponds to the middle of the 1st row (top) to the middle of the 200th row (bottom)), params(2) is the position of the pRF center expressed in terms of column indices (1-200 corresponds to the middle of the 1st column (left) to the middle of the 200th column (right)), params(3) is the standard deviation of the Gaussian (expressed in pixel units), and params(5) is the exponent parameter.

- Note that the results written to the .nii.gz files are **not the raw parameters** mentioned above, but instead are the parameters that have been transformed to more meaningful units (e.g. degrees of visual angle).
- Note that the prf_size.nii.gz file that is provided **does not reflect** the sigma parameter as used in the model code above, but rather sigma/sqrt(exponent). The motivation behind dividing sigma by sqrt(exponent) is to produce a measure of pRF size that takes into account the nonlinear behavior induced by the compressive power-law nonlinearity. Specifically, sigma/sqrt(exponent) is one standard deviation of a Gaussian that describes how the model would respond to a point-like stimulus that is moved around in the visual field.
- Here is some example code that starts with the prf_XXX.nii.gz outputs that are provided with NSD and transforms these into a format that follows the model implementation:

```
prfangle = 15;    % degrees
prfecc = 2;       % degrees visual angle
prfexpt = 0.2;
prfsize = 4;      % degrees visual angle

sigma = prfsize*sqrt(expt);      % sigma parameter in degrees visual angle
sigmapx = sigma * (200/8.4);     % sigma parameter in pixel units
rindex = (1+200)/2 - (prfecc*sin(prfangle/180*pi) * (200/8.4)); % pRF y-position in row pixel units
cindex = (1+200)/2 + (prfecc*cos(prfangle/180*pi) * (200/8.4)); % pRF x-position in column pixel units
```

```
gau = makegaussian2d(200,rindex,cindex,sigmapx,sigmapx);  % Gaussian image that peaks at 1. This Gaussian corresponds to the Gaussian used in the modeling function (prior to the scale normalization, dot-product with the stimulus, exponentiation, and the gain).
```

- For all the gory details, you can find the code that performed the pRF analysis in analysis_prf.m (provided in the nsddatapaper github repository).

Results from the floc experiment

The pre-processed fMRI time-series data from the floc experiment (6 runs, 312-s each) was analyzed using a GLM. The results are provided in the files detailed below. Note that in each of the files, NaN values are possible and indicate either missing data or voxels outside of the brain mask.

For convenience, we already compute various contrasts. Keep in mind that what we call "domains" is a higher hierarchical organizational scheme of the "categories". For example, the domain of 'faces' includes both the 'adult' and 'child' categories. We compute contrasts for each of the 5 domains (see nsddata/experiments/floc/domains.tsv), yielding values that quantify how large the response is to stimuli from a given domain compared to all other stimuli. We also compute contrasts for each of the 10 categories (see nsddata/experiments/floc/categories.tsv), yielding values that quantify how large the response is to stimuli from a given category compared to all other stimuli EXCLUDING stimuli in the category that is paired with the given category. For example, “facestval” contrasts responses to the domain of faces (adult and child faces aggregated) against responses to all other stimuli; “adulttval” contrasts responses to the category of adult faces against responses to all other stimuli excluding child faces; “childtval” contrasts responses to the category of child faces against responses to all other stimuli excluding adult faces.

Each contrast is expressed using two different metrics. “tval” is a conventional t-statistic that results from performing a two-sample t-test. “anglemetric” is a metric that, in contrast to “tval”, does not depend on the amount of data collected, and is simply the angle in the Cartesian coordinate plane made by the mean of the two groups being compared. For example, the point (A,B) plots the response to A along the x-axis and the response to B along the y-axis. Values for “anglemetric” range between -180 and 180 degrees and the zero point corresponds to the situation where A==B and A and B are positive. Thus, 0° indicates equal response to A and B;

positive values going up to 180° proceed clockwise and indicate a preference for A; negative values going down to -180° proceed counterclockwise and indicate a preference for B.

Both volume-based and surface-based versions of the results are available. Volume-based results are located at

nsddata/ppdata/subjAA/func*/floc_BBB.nii.gz

and surface-based results are located at

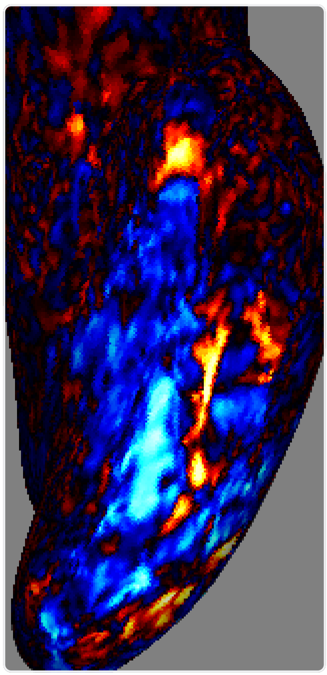
nsddata/freesurfer/subjAA/label/[lh,rh].flocBBB.mgz

where BBB refers to different quantities. To create surface-based versions, we take the 1-mm volume-based floc results and map them to the left and right hemisphere cortical surfaces (linear interpolation onto the 3 depth surfaces, average across depth).

Below, we document each of the BBB quantities.

DDDtval

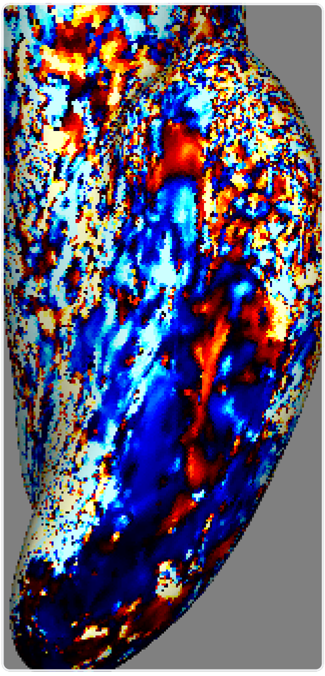
This contains, for each voxel, the t-value corresponding to contrasting domain DDD (or category DDD) against other stimuli.



subj02/label/lh.flocfacestv

DDDanglemetric

This contains, for each voxel, the angle metric corresponding to contrasting domain DDD (or category DDD) against other stimuli.



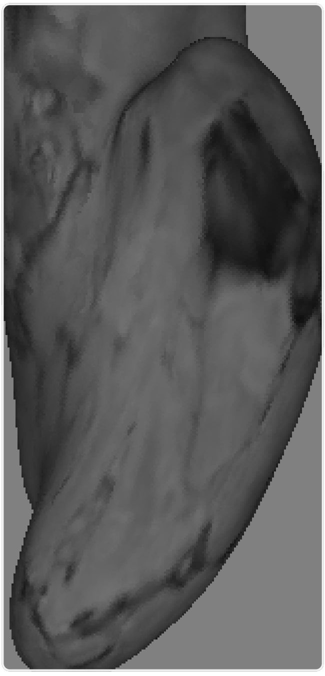
subj02/label/lh.flocfacesa

betas

This contains, for each voxel, 6 trials x 10 categories = 60 beta weights. The GLM incorporates six separate regressors for each category (coding distinct trials), producing six separate beta weights for each category. These distinct beta weights are used to compute the *t*-values and the angle metrics.

meanvol

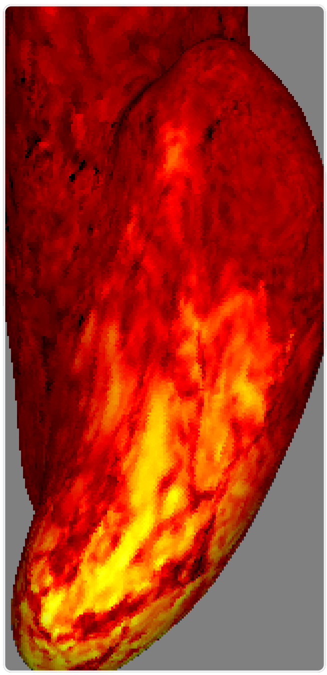
This contains, for each voxel, the mean EPI intensity (in the floc data). Values are in raw scanner units and generally fall in the range 0 to 4095.



subj02/label/lh.flocmeanv

R2

This contains, for each voxel, the variance explained by the GLM model. Values generally lie between 0% and 100%, but other values are possible.



subj02/label/lh.flocR2.mgz

Functional data (NSD)

This covers GLM results for the NSD experiment. The goal of the main GLM analysis of the NSD data was to estimate single-trial betas (BOLD response amplitudes) for each voxel.

File format issues for betas

The files that contain NSD betas are very large. The units of the prepared betas are percent signal change. However, for some of ther NSD data files that we have prepared, the betas have been multiplied by 300 and converted to int16 format to reduce space usage. Upon loading the beta files, the values should be immediately converted back to percent signal change by casting to decimal format (e.g. single or double) and dividing by 300.

For volume-based format of the betas, two versions have been prepared:

- **NIFTI (.nii.gz)**. These data are in int16 format. A liberal brain mask has been applied such that non-brain voxels have been zeroed-out in order to save disk space. The .gz indicates that the files are compressed (to save disk space). The advantage of .nii.gz format is that it is standard and easy-to-use, but the disadvantage is that the files must be uncompressed when loading and must be completely loaded into memory.
- **HDF5 (.hdf5).** These data are in int16 format. '/betas' is X voxels x Y voxels x Z voxels x 750 trials. A liberal brain mask has been applied such that non-brain voxels have been zeroed-out. This file is in HDF5 format (with a specific chunk size of [1 1 1 750]) in order to enable very fast random access to small parts of the data file. A disadvantage of this format is that the file is uncompressed and therefore large in size.

  Here is an example of how to use MATLAB to quickly load all 750 single-trial betas associated with 5 voxels from a single scan session, using h5read.m.

  ```
  data = h5read('betas_session01.hdf5','/betas',[10 10 10 1],[1 1 5 750]);
  ```

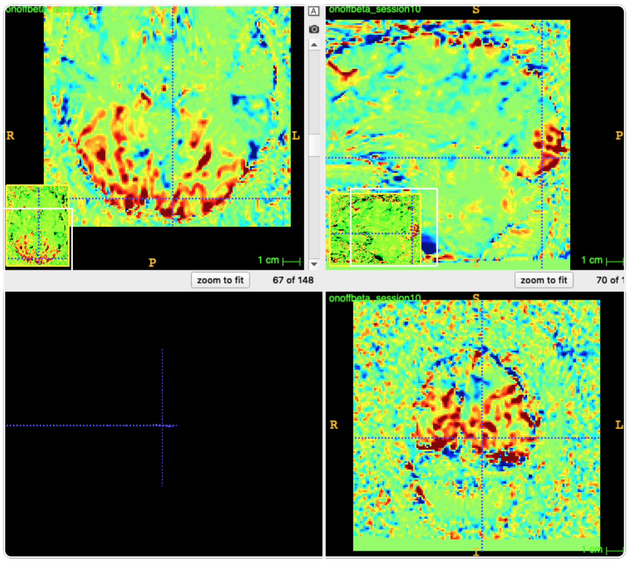
  Note that these are 1-indexed (due to MATLAB’s convention), and hence we are loading the 10th, 11th, 12th, 13th, and 14th voxels along the third dimension.

Results of a simple ON-OFF GLM

Besides the single-trial GLM, the NSD were also analyzed with a simple ON-OFF GLM in order to derive some useful quantities.

nsddata/ppdata/subjAA/func*/onoffbeta_sessionBB.nii.gz

> This is the beta (in percent signal change units) obtained, for session BB, for a simple GLM model that describes experiment-related variance with a simple ON-OFF predictor (one condition, canonical HRF).



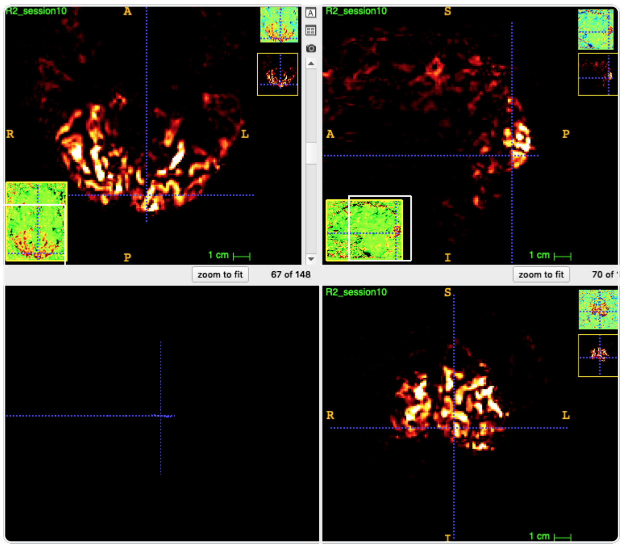
subj01/func1mm/onoffbeta_session10.nii.gz

nsddata/ppdata/subjAA/func*/onoffbeta.nii.gz

> This is the average (using nanmean.m) of the onoffbeta across all sessions.

nsddata/ppdata/subjAA/func*/R2_sessionBB.nii.gz

> This is the voxel-wise variance explained (0-100) for the simple ON-OFF GLM model for session BB.



subj01/func1mm/R2_session10.nii.gz

nsddata/ppdata/subjAA/func*/R2.nii.gz
nsddata/freesurfer/subjAA/label/[lh,rh].R2.mgz

> This is the voxel-wise variance explained, averaged across all sessions (using nanmean.m).

Results of single-trial GLM

For single-trial GLM, we analyzed the time-series data from the NSD experiment using 3 different GLM models. The identifiers for these models are:

- **betas_assumehrf (beta version 1; b1)** - GLM in which a canonical HRF is used
- **betas_fithrf (beta version 2; b2)** - GLM in which the HRF is estimated for each voxel
- **betas_fithrf_GLMdenoise_RR (beta version 3; b3)** – GLM in which the HRF is estimated for each voxel, the GLMdenoise technique is used for denoising, and ridge regression is used to better estimate the single-trial betas.

The interpretation of the betas obtained from these GLMs is that they are the BOLD response amplitudes evoked by each stimulus trial relative to the baseline signal level present during the absence of a stimulus (“gray screen”). Note that betas are expressed in percent signal change by dividing the full set of amplitudes obtained for a voxel by the grand mean intensity observed for that voxel in a given scan session and then multiplying by 100.

Betas are provided both in the subject-native volume spaces (func1mm and func1pt8mm), a subject-native surface space (nativesurface) as well as in group spaces (fsaverage and MNI).

Details on the nativesurface and group spaces are provided later.

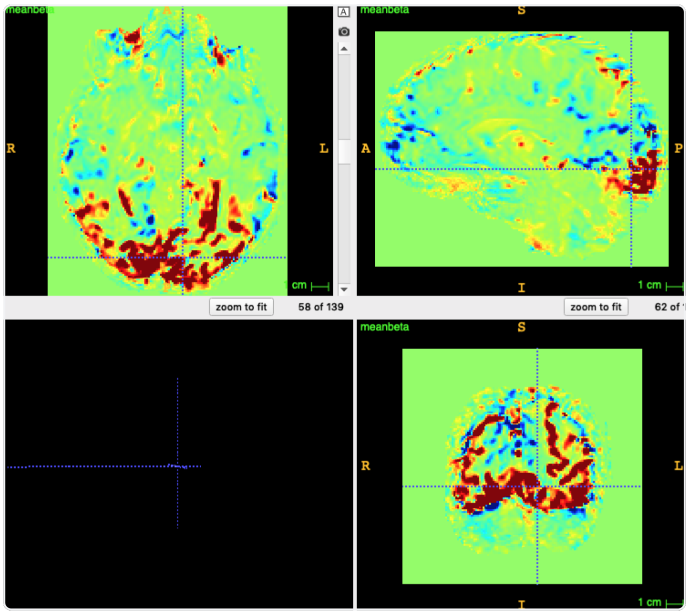
Note that to save disk space, the ‘betas_assumehrf’ version is provided for the func1pt8mm space but not for the func1mm space.

nsddata_betas/ppdata/subjAA/func*/betas_*/betas_sessionBB.[nii.gz,hdf5]

These are single-trial betas (that have been multiplied by 300 and converted to integer format). The betas are in chronological order. There are 750 betas since there are 750 stimulus trials in each scan session (after concatenating all 12 runs). The betas correspond to the data acquired in session BB for subject AA.

nsddata_betas/ppdata/subjAA/func*/betas_*/meanbeta.nii.gz
nsddata_betas/ppdata/subjAA/func*/betas_*/meanbeta_sessionBB.nii.gz

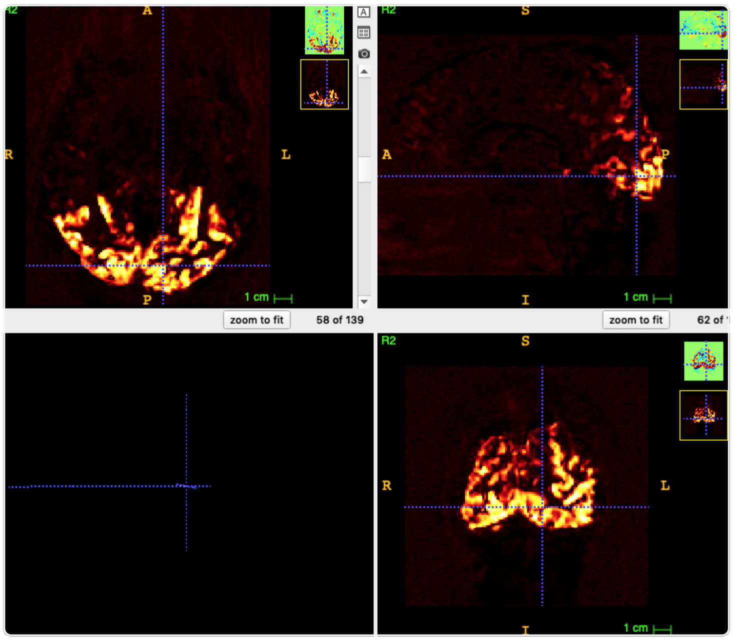
For each session, the mean of all single-trial betas is calculated (meanbeta_sessionBB); then, this mean is averaged across all scan sessions (meanbeta). The result is a volume that indicates the voxel-wise average single-trial beta obtained for subject AA. (Please note that although the file format is single, the values must still be divided by 300 in order to return to percent signal change units.)



subj05/func1mm/betas_fithrf/meanbeta.nii.gz

nsddata_betas/ppdata/subjAA/func*/betas_*/R2.nii.gz
nsddata_betas/ppdata/subjAA/func*/betas_*/R2_sessionBB.nii.gz

This contains the variance explained by the GLM model in each session (R2_sessionBB), and the average of this quantity across all sessions (R2). Please note that the R2 values for the 'betas_assumehrf' and 'betas_fithrf' models are probably not very useful given that these models are very flexible and can essentially fit nearly all of the variance in a given time-series (even if the time-series has no reliable stimulus-evoked responses). In contrast, the R2 for the 'betas_fithrf_GLMdenoise_RR' may be useful given that the ridge-regression regularization does shrink the model according to the response reliability that appears to be in the data for each given voxel. NaNs are possible in R2_sessionBB.nii.gz for invalid voxels. For R2.nii.gz, we compute the mean using nanmean.



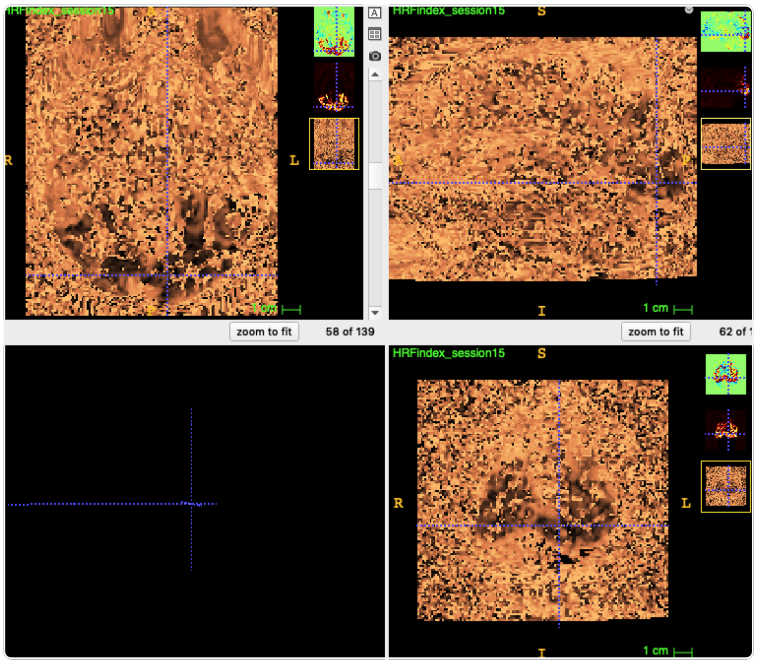
subj05/func1mm/betas_fithrf_GLMdenoise_RR/R2.nii.gz

nsddata_betas/ppdata/subjAA/func*/betas_*/R2run_sessionBB.nii.gz

This contains the variance explained by the GLM model calculated separately for each run in a given session.

nsddata_betas/ppdata/subjAA/func*/betas_*/HRFindex_sessionBB.nii.gz
nsddata_betas/ppdata/subjAA/func*/betas_*/HRFindexrun_sessionBB.nii.gz

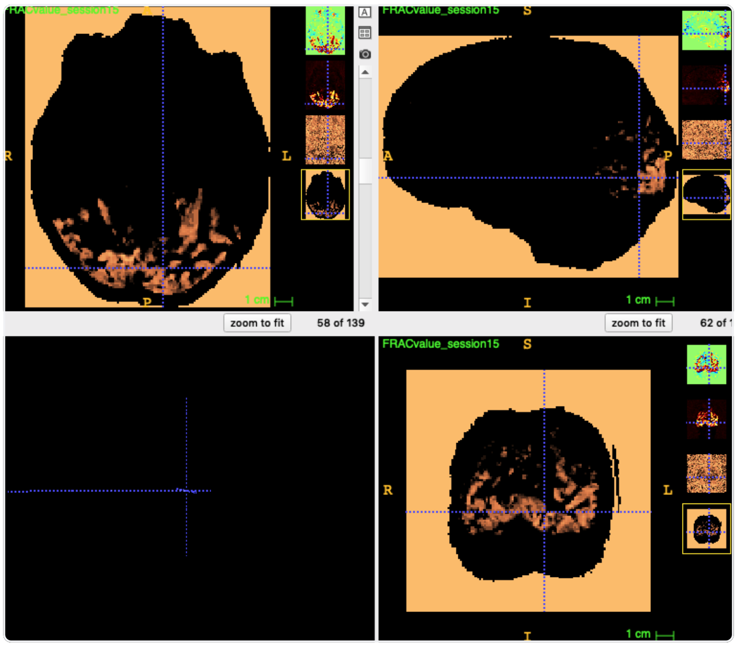
Index of the chosen HRF for each voxel (integer between 1 and 20). This is estimated for each run in a session (HRFindexrun_sessionBB). The final HRF used to analyze the entire session of data is determined by combining results across runs (HRFindex_sessionBB).



subj05/func1mm/betas_fithrf_GLMdenoise_RR/HRFindex_se

nsddata_betas/ppdata/subjAA/func*/betas_*/FRACvalue_sessionBB.nii.gz

> The fractional regularization level chosen for each voxel. Note that invalid voxels (e.g. outside of brain) are given a fraction of 1.



subj05/func1mm/betas_fithrf_GLMdenoise_RR/FRACvalue

Single-trial GLM results in nativesurface format

nsddata_betas/ppdata/subjAA/nativesurface/betas_*/[lh,rh].betas_sessionBB.hdf5

These files contain betas in the native FreeSurfer surface space for a given subject. They are saved in .hdf5 format to allow for very rapid access to subsets of the available vertices.

To generate these betas, we take the 1-mm subject-native volume betas, resample via cubic interpolation onto the subject-native cortical surfaces (which exist at 3 different depths), and average the resulting betas across depths. The resulting matrices have dimensions vertices x trials (and are separated by hemisphere).

Note that the betas are saved in int16 format and are multiplied by 300. In the case of missing data in a given scan session (i.e., due to head motion, a spatial location is moved out of the imaging field-of-view), it is possible that vertices have their betas set to all zeros. (There are very few instances where data are missing for cortical surface vertices; see nsddata/information/knowndataproblems.txt for more information. To detect such cases, one can simply check in each scan session whether all betas for a given vertex are equal to 0.) The ‘ChunkSize’ for the .hdf5 files is [1 T] where T is the total number of trials; this makes loading of all of the trials for single vertex (or small group of vertices) very fast.

Here is an example of how to use MATLAB to quickly load all 750 single-trial betas associated with the first 100 vertices from a single scan session.

```
data = h5read(‘lh.betas_session01.hdf5','/betas',[1 1],[100 750]);
```

Note that the indices in MATLAB are 1-based.

Single-trial GLM results in group spaces (fsaverage, MNI)

The primary advantage of the subject-native spaces is that they provide the highest-resolution version of the NSD data. However, group analyses may be of interest, and one may want to transform the NSD data to group spaces prior to analysis. (Note that in theory, one can perform analyses of subject-native data and then transform to group spaces at the very end of the analysis process; this will likely give similar but not identical results.)

The group space versions of the betas are obtained by taking the betas in the subject-native 1-mm volume space and then resampling the betas to the group spaces (more details on the resampling procedures for fsaverage and MNI is provided below). Thus, there is some additional interpolation (and loss of resolution) inherent in the group-space betas.

nsddata_betas/ppdata/subjAA/fsaverage/betas_*/[lh,rh].betas_sessionBB.mgh

These files contain betas in the FreeSurfer fsaverage space. To generate these betas, we start with the subject-native surface format (i.e. take the 1-mm subject-native volume betas, resample via cubic interpolation onto the subject-native cortical surfaces (which exist at 3 different depths), average the resulting betas across depths), but then we additionally map via nearest-neighbor interpolation to the fsaverage surface. Note that the betas are saved in decimal format and are in percent signal change units (i.e. they are not multiplied by 300). In the case of missing data, it is possible that betas will have NaNs. Here is a simple example:

```
a1 = load_mgh('lh.betas_session04.mgh');
>> size(a1)
ans =
       163842            1            1          750
```

nsddata_betas/ppdata/subjAA/MNI/betas_fithrf/betas_sessionBB.nii.gz

These files contain betas in MNI space. To generate these betas, we take the 1-mm subject-native volume betas and resample them via cubic interpolation into MNI space. Note that values are in int16 format and are multiplied by 300. Note that voxels with invalid data for a given scan session (either because data was missing from the subject-native volume or because the location is outside of the subject-native brain mask) will have their betas set to all zeros. Finally, note that to save disk space, we provide only the 'betas_fithrf' version of the betas in MNI space (we do not include 'betas_assumehrf' nor 'betas_fithrf_GLMdenoise_RR').

nsddata_betas/ppdata/subjAA/MNI/betas_fithrf/valid_sessionBB.nii.gz

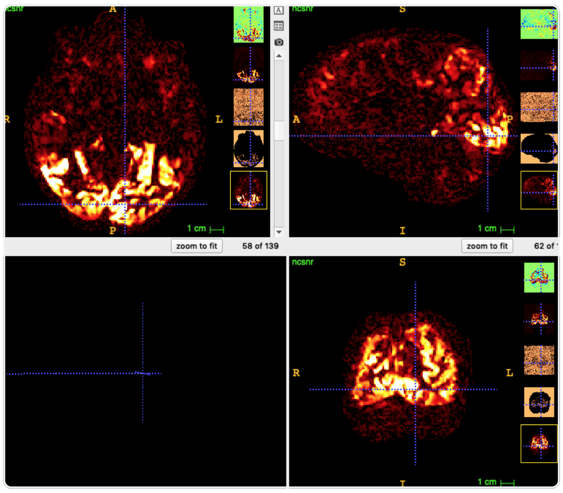
These files correspond to betas_sessionBB.nii.gz and indicate which voxels contain valid data for each given scan session.

Noise ceiling

Noise ceiling estimates have been computed based on the trial-to-trial reliability of the beta weights. In essence, the more repeatable the response across repeated presentations of an image, the more variance in the response can be attributed to a stimulus-related signal. These noise ceiling estimates are useful for putting an upper bound on the amount of variance that can be explained/predicted in a given voxel's (or vertex's) beta weights. Formal description of the statistical theory behind the noise ceiling calculation can be found in the NSD data paper.

nsddata_betas/ppdata/subjAA/func*/betas_*/ncsnr.nii.gz
nsddata_betas/ppdata/subjAA/fsaverage/betas_*/[lh,rh].ncsnr.mgh
nsddata_betas/ppdata/subjAA/nativesurface/betas_*/[lh,rh].ncsnr.mgh

> These files provide the noise ceiling signal-to-noise ratio (*ncsnr*) for each voxel (or vertex). These *ncsnr* values are computed on basis of all of the beta weights obtained in all NSD scan sessions. Values are generally between 0 and 0.6 but can go higher (a subset of voxels/vertices will be exactly 0, and this is expected behavior given the nature of the procedure). Invalid voxels (e.g. outside the brain) are given a value of NaN. The *ncsnr* can be easily converted into noise ceilings (see below). The "ncsnr_split1" and "ncsnr_split2" files reflect calculations of the ncsnr value from two independent splits of the images available for each given subject.



subj05/func1mm/betas_fithrf_GLMdenoise_R

Conversion of ncsnr to noise ceiling percentages

In the NSD data paper, we explain that the noise ceiling (*NC*) can be expressed as:

$$NC = 100 \times \frac{{\sigma_{signal}}^2}{{\sigma_{signal}}^2 + {\sigma_{noise}}^2}$$

where *sigma_signal* is the standard deviation of the signal and *sigma_noise* is the standard deviation of the noise. But how can this be computed based on knowledge of the noise ceiling signal-to-noise ratio (*ncsnr*)? Before deriving that result, consider the fact that the user may wish to average together responses across several trials conducted for each image. By averaging, the user is effectively reducing the variance of the noise. Since we are assuming that the noise is Gaussian-distributed, the effective noise variance becomes:

$$\frac{{\sigma_{noise}}^2}{n}$$

where *n* is the number of trials that are averaged together. We can now re-write the noise ceiling as:

$$NC = 100 \times \frac{{\sigma_{signal}}^2}{{\sigma_{signal}}^2 + \frac{{\sigma_{noise}}^2}{n}}$$

Dividing the numerator and denominator by sigma_noise^2, we obtain

$$NC = 100 \times \frac{\frac{{\sigma_{signal}}^2}{{\sigma_{noise}}^2}}{\frac{{\sigma_{signal}}^2}{{\sigma_{noise}}^2} + \frac{1}{n}}$$

which further reduces to

$$NC = 100 \times \frac{ncsnr^2}{ncsnr^2 + \frac{1}{n}}$$

This shows how the noise ceiling for a given voxel can be computed from its *ncsnr* value.

One complication is that one might be using a preparation of the data in which different images have different numbers of trials that are averaged together. To flexibly deal with any potential scenario, we can use a weighted average to pool variance estimates across different images and re-write the noise ceiling equation as:

$$NC = 100 \times \frac{{\sigma_{signal}}^2}{{\sigma_{signal}}^2 + \left(\frac{A\left(\frac{\sigma_{noise}}{\sqrt{3}}\right)^2 + B\left(\frac{\sigma_{noise}}{\sqrt{2}}\right)^2 + C\left(\frac{\sigma_{noise}}{\sqrt{1}}\right)^2}{A + B + C} \right)}$$

where *A* is the number of data points that reflect 3 trials, *B* is the number of data points that reflect 2 trials, and *C* is the number of data points that reflect 1 trial. With some algebra, we can then re-write the noise ceiling equation as follows:

$$NC = 100 \times \frac{ncsnr^2}{ncsnr^2 + \frac{\frac{A}{3} + \frac{B}{2} + \frac{C}{1}}{A + B + C}}$$

Notice that this equation is simply a more general version of the earlier noise ceiling equation.

Technical notes

- The b3 betas are appropriate only for brain regions where there is some expectation that the BOLD response will be consistent across the repetitions of a given image. This is because the regularization level is based on cross-validation of responses to image repetitions.
- Ridge regression tends to shrink betas and therefore induces bias for percent signal change to be closer to 0. We apply a post-hoc scale and offset to the b3 betas to approximately match what is observed for unregularized betas (see NSD data paper for details). If absolute units of percent signal change are of specific interest, the betas_fithrf (b2) preparation is more straightforward to interpret and is therefore recommended for use instead.
- If one seeks to perform connectivity-based analyses that look for correlations in betas across voxels (or regions), there may be large differences in results comparing b1 and b2 against b3. The general expectation is that the GLMdenoise procedure (which is incorporated as part of b3) will tend to remove global signal correlations that may exist in the fMRI data.
- In the ncsnr values that are provided, there are occasionally high values outside the brain; this is likely an artifact due to an interaction between the fact that imaging artifacts tend to have low temporal frequencies and the specific temporal distribution of repeated trials in the NSD experiment.

Functional data (resting-state)

Most users will likely want to start with the pre-processed time-series data for the resting-state data (see Time-series data). However, as described in the NSD data paper, we have used a GLM to analyze the resting-state data, and the results may be of interest to some users. To obtain betas, we simply analyzed the resting-state data as if they were data acquired for the first NSD run and last NSD run in each given scan session.

Results of single-trial GLM

nsddata_betas/ppdata/subjAA/*/restingbetas_fithrf/

> This contains results of the GLM analysis of the resting-state runs. Note that only the '_fithrf' GLM version of the betas are provided. The format is the same as for the NSD runs.

Diffusion data

This section covers the measurements and pre-processing of diffusion-weighted magnetic resonance imaging data (dMRI) prepared for the NSD dataset.

Data were preprocessed using publicly available processing pipelines available on brainlife.io. Preprocessing pipelines were used to remove artifacts as well as possible; see note at the end. After artifact removal/minimization, a series of additional brainlife.io pipelines were used to generate and share data derivatives, including minimally preprocessed dMRI data, tractography, and network outputs.

Diffusion (dMRI) data collection

The four diffusion-weighted acquisitions were combined into two runs of diffusion data (referred to as 'run_1', 'run_2'). The two diffusion runs were combined (stacked in the 4th dimension) before being processed. Data preprocessing included susceptibility-weighted, motion, and eddy correction.

Cloud processing via *brainlife.io*

All processing was performed on the reproducible, open cloud-based service known as *brainlife.io*. *Brainlife.io* orchestrates large-data storage, processing via open-service code applications (apps), and high-speed large computing resources to quickly and reproducibly process neuroimaging data.

All of the code and pipelines used for processing the data described below can be found on brainlife.io and from there on GitHub.com. A table at the end of this document provides all references to the pipeline used for data processing and generation.

The output files generated are further described below.

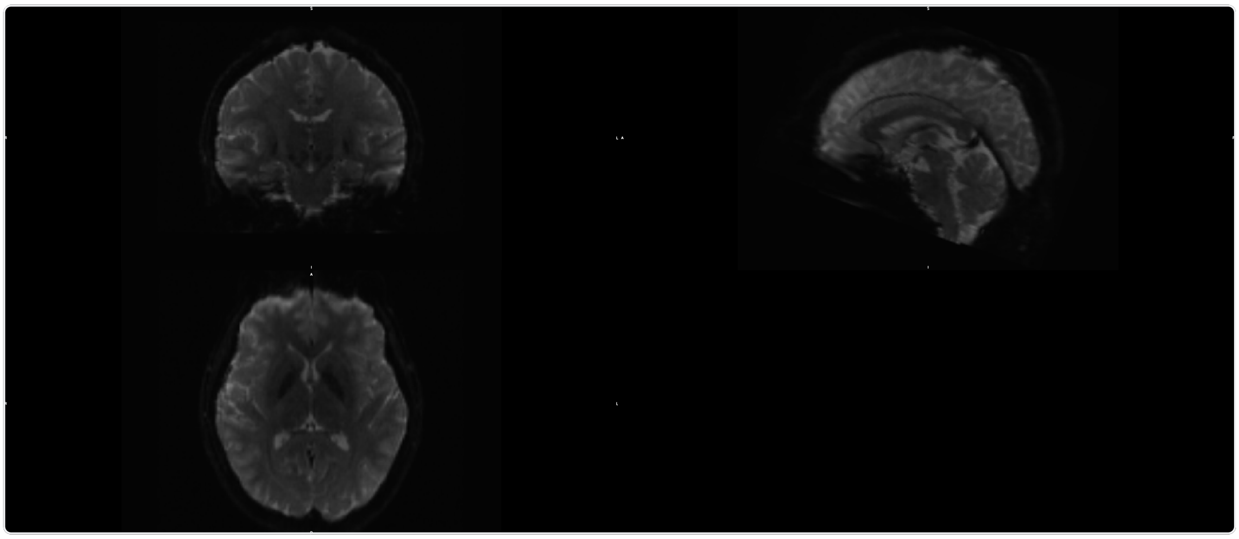
Diffusion-weighted imaging (dMRI).

The preprocessed dMRI data were used as the basis for all further modeling and analyses. This includes NIFTI images and the corrected b-values (bvals) and b-vectors (bvecs) in FSL format. These NIFTIs are in alignment with and have the same slice dimensions and voxel size as the official 0.8-mm T1w images provided with NSD (see Untitled). All NIFTI-based volume derivatives from the dMRI data maintain the same properties in regards to slice and voxel sizes. (Note that in our preprocessing, we drop the very last acquired volume; hence there is a one-volume mismatch between the number of volumes in the raw data (99, 99, 100, 100 for the four raw diffusion acquisitions) and the number of volumes in the preprocessed data (98 for 'Run 1' (which combines the first two acquisitions) and 99 for 'Run 2' (which combines the second two acquisitions).)

`nsddata_diffusion/ppdata/subjAA/run_*/dwi.nii.gz`

`nsddata_diffusion/ppdata/subjAA/run_*/dwi.bvecs`

`nsddata_diffusion/ppdata/subjAA/run_*/dwi.bvals`



subj07/run_1/dwi/dwi.nii.gz

Signal-to-noise ratio (SNR) quantification.

Following preprocessing and separation of the dMRI data into its component runs, the signal-to-noise ratio (SNR) was computed using a brainlife.io App implementing methods available on the scientific library DIPy.org. The output of this process is a .csv file describing the SNR found across the x-, y-, or z-directions in diffusion-weighted volumes and the SNR across the non-diffusion weighted volumes:

`nsddata_diffusion/ppdata/subjAA/run_*/snr/snr.csv`

dMRI brain mask.

A brain mask was generated with an App implementing FSL BET and used for all dMRI signal modeling and analyses purposes. The brain mask was generated using the preprocessed and combined dMRI data following preprocessing. The same mask was used for all subsequent processing steps:

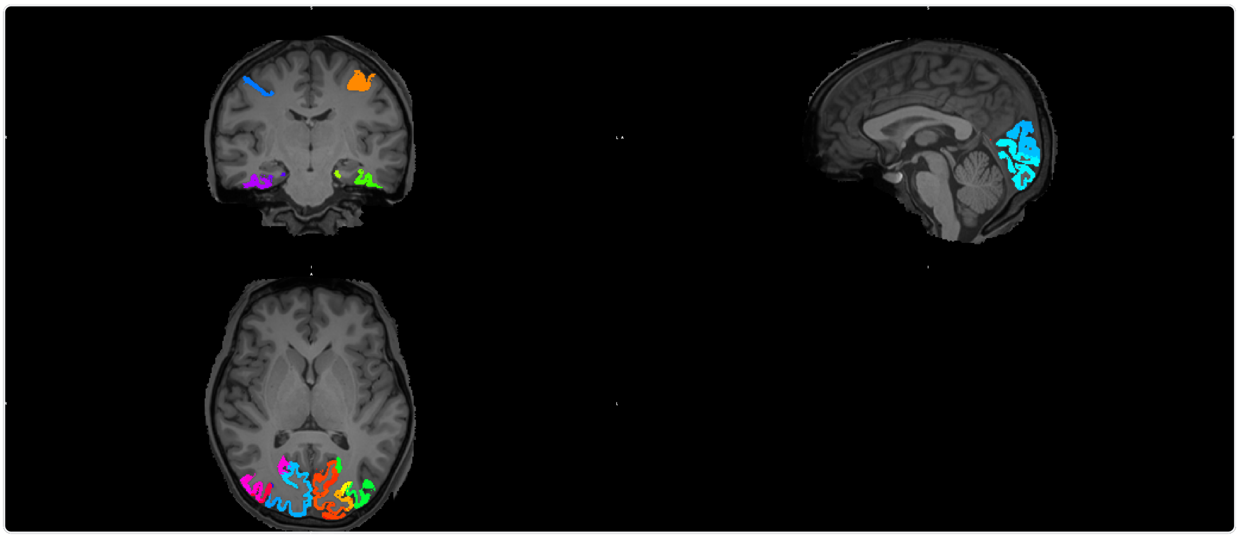
`nsddata_diffusion/ppdata/subjAA/brainmask/mask.nii.gz`

subj07/brainmask/mask.nii.gz

Visual area parcellation.

A parcellation of the visual areas was implemented using the 180 multi-modal cortical Atlas (Glasser et al, 2016). The Atlas and areas were imported into dMRI volume space. The areas were used to segment the optic radiation and to generate area-to-area connectivity matrices. A `key.txt` file is provided also. The file includes the assignment of the voxels into the NIFTI files to the indices of the areas in the parcellation. A `label.json` file is also provided to includes important information for the parcellation nifti.

`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-parcellation/parcellation.nii.gz`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-parcellation/key.txt`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-parcellation/label.json`



subj07/run_1/visual-area-parcellation/parc.nii.gz

Diffusion signal modeling and data derivatives

The Diffusion-Tensor Model (DTI; Le Bihan et al., Journal of Magnetic Resonance Imaging, 2001), Diffusion Kurtosis Imaging (DKI; Rosenkrantz et al. Journal of Magnetic Resonance Imaging, 2015), and Neurite Orientation Dispersion Diffusion Imaging (NODDI; Zhang et al. Neuroimaging 2012) models were fit to the dMRI data.

Diffusion Tensor Imaging (DTI).

The fractional anisotropy, mean diffusivity, axial diffusivity, and radial diffusivity maps from the DTI model were generated using methods implementing in MRTrix3 (JD Tournier et al. Neuroimage 2019) as implemented in a brainlife.io App.

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/ad.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/fa.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/md.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/rd.nii.gz
```

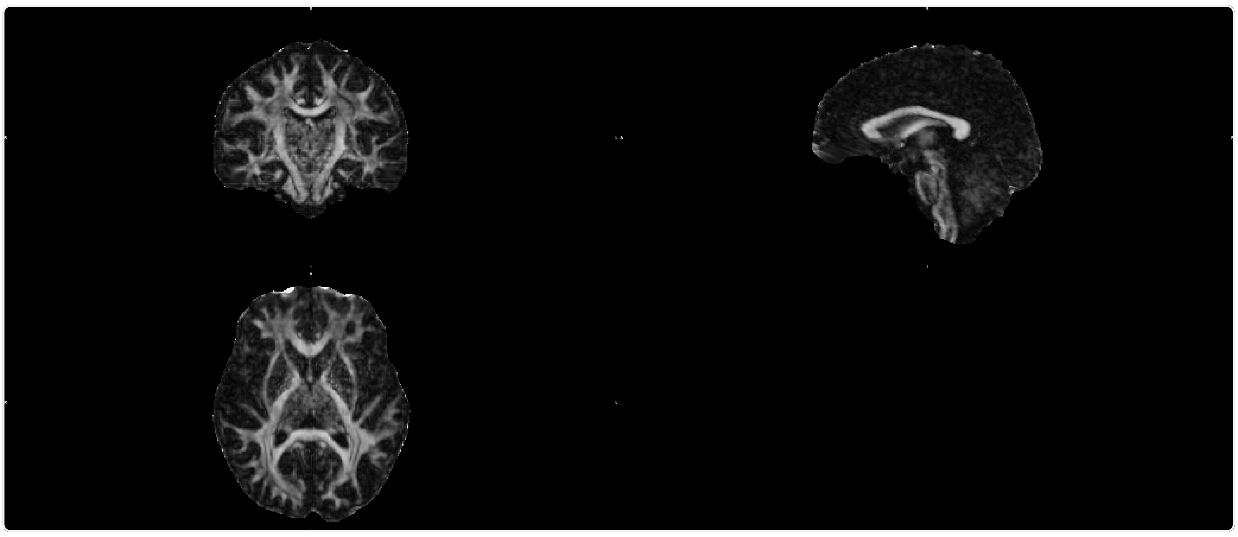
Additional parameters were also returned byMRTrix3 given the multi-shell nature of the data.

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/cs.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/cl.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/cp.nii.gz
```

```
nsddata_diffusion/ppdata/subjAA/run_*/dti/kurtosis.nii.gz
```

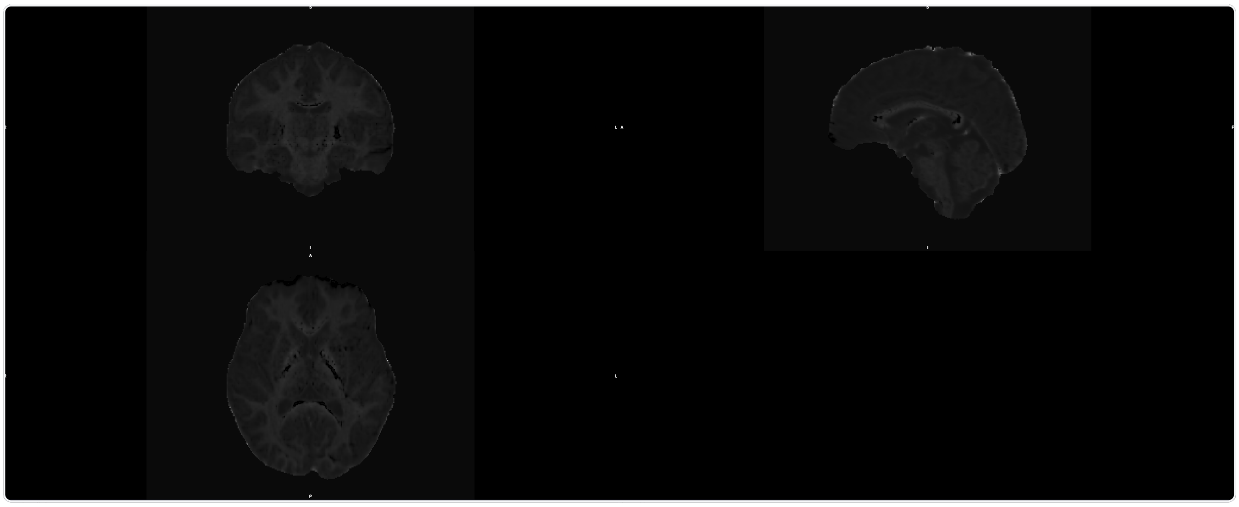


subj07/run_1/dti/fa.nii.gz

Diffusion Kurtosis Imaging (DKI).

The implementation of DKI provided by the library DIPy.org was used via a brainlife.io App to generate DKI model parameter estimates. Both DTI measures (fractional anisotropy, mean diffusivity, axial diffusivity, radial diffusivity), as well as proper DKI measures (axial kurtosis, geodesic anisotropy, mean kurtosis, radial kurtosis), maps were generated.

```
nsddata_diffusion/ppdata/subjAA/run_*/dki/ad.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/fa.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/md.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/rd.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/ak.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/ga.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/mk.nii.gz
nsddata_diffusion/ppdata/subjAA/run_*/dki/rk.nii.gz
```



subj07/run_1/dki/mk.nii.gz

Neurite Orientation Dispersion Density Imaging (NODDI).

The NODDI implementation available in the library AMICO was used via a brainlife.io App to generate all parameter estimates. The neurite density, orientation dispersion, and isotropic volume fraction maps were generated. Two fits of the NODDI model were applied per dMRI run. The parallel diffusivity parameter (d//) was changed by run/fit.

The **first model fitting** was performed with d// = 1.7 x 10^{-3}mm^{2}/s, which is designed for fitting in deep white matter. In the data, this is marked as *noddi-wm* directory.

The **second model fitting** was performed with d// = 1.7 x 10^{-3}mm^{2}/s which was found to be the optimal value for gray matter mapping as identified in Fukutomi et al, 2018. This is designated with a *noddi-cortex* directory. The files within each directory have the same name, and thus we describe one set of directories below.

```
nsddata_diffusion/ppdata/subjAA/run_*/noddi-{}/ndi.nii.gz
```

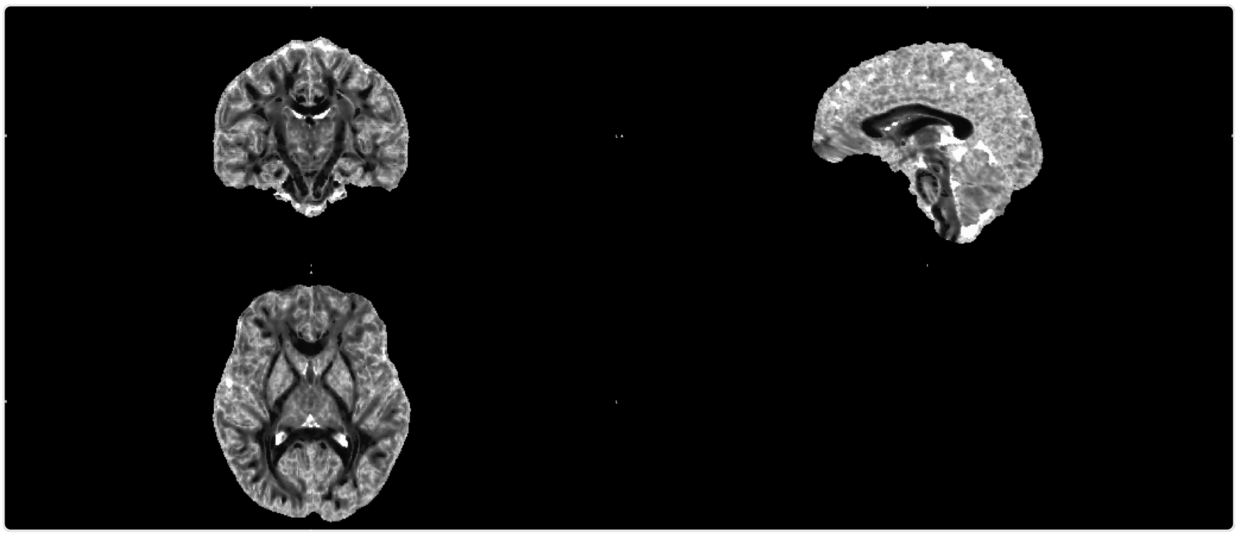
# neurite density index map for either the white matter (wm) or cortex fits

`nsddata_diffusion/ppdata/subjAA/run_*/noddi-{}/odi.nii.gz`

# orientation dispersion index map for either the white matter (wm) or cortex fits

`nsddata_diffusion/ppdata/subjAA/run_*/noddi-{}/isovf.nii.gz`

# isotropic volume fraction map for either the white matter (wm) or cortex fits



subj07/run_1/noddi-wm/odi.nii.gz

Constrained Spherical Deconvolution (CSD).

CSD model fits for diffusion tractography across multiple spherical harmonic orders (L_{max}=2, 4, 6, and 8) using MRTrix3.

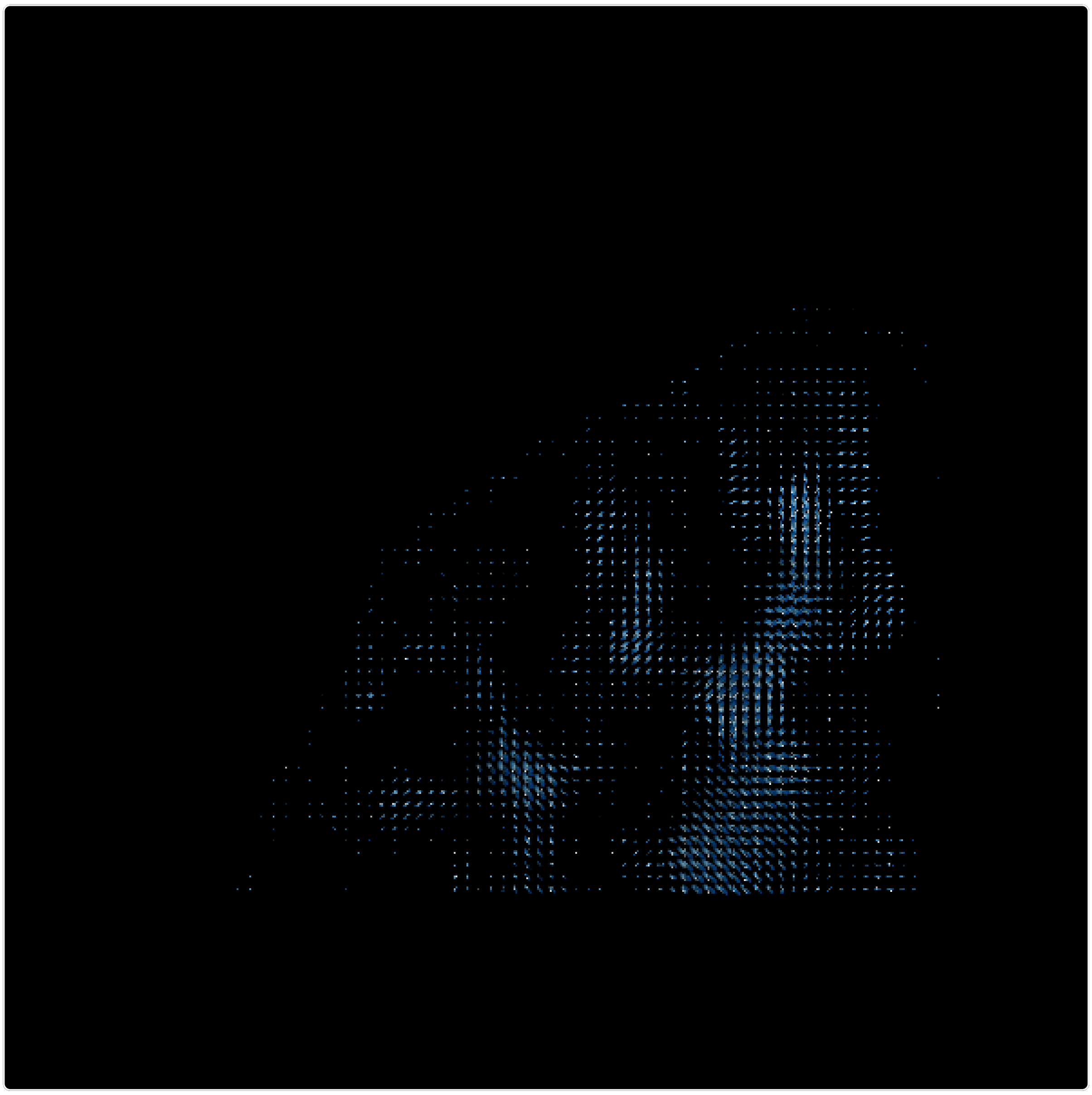
`nsddata_diffusion/ppdata/subjAA/run_*/csd/lmax2.nii.gz`

`nsddata_diffusion/ppdata/subjAA/run_*/csd/lmax4.nii.gz`

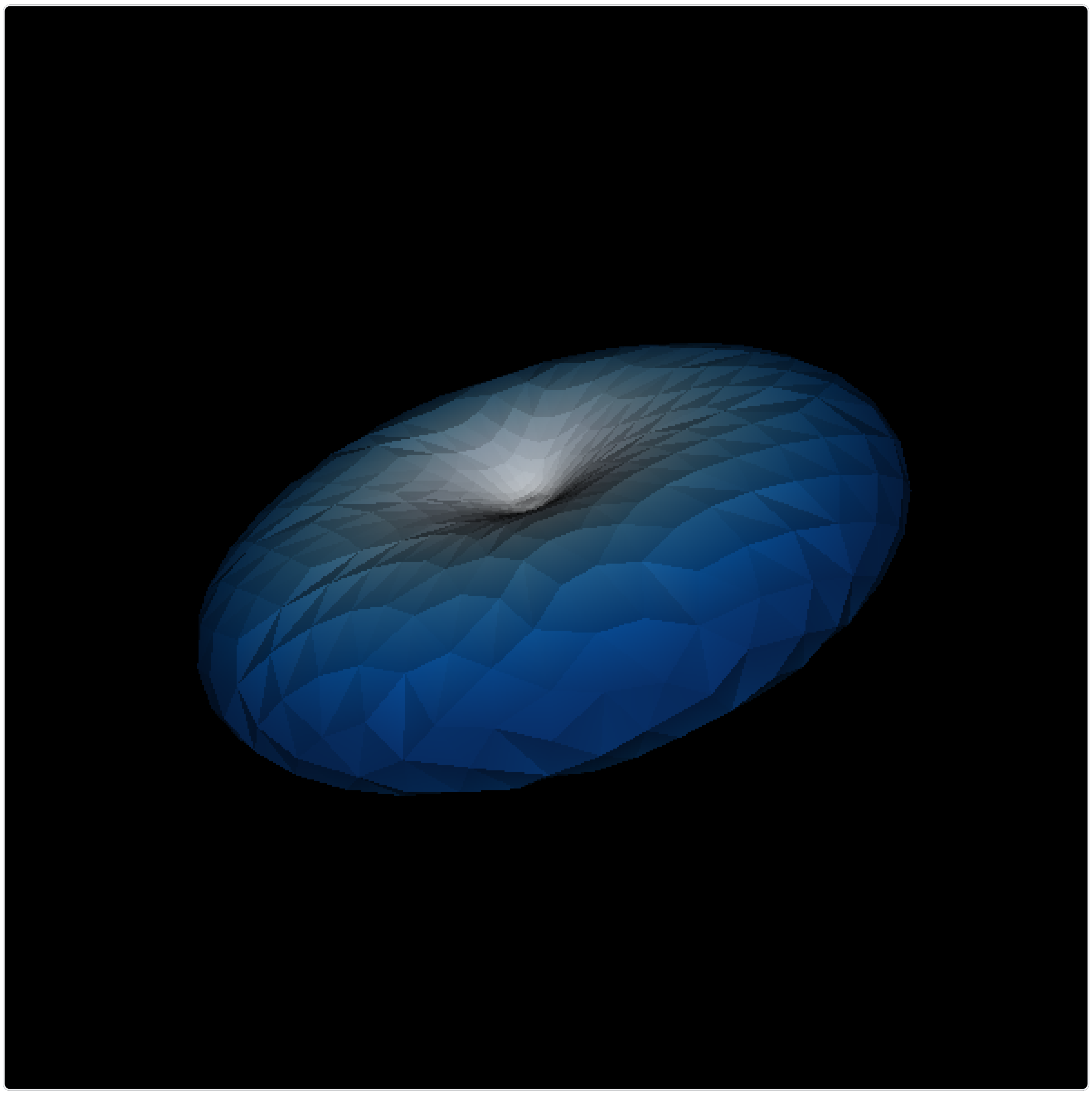
`nsddata_diffusion/ppdata/subjAA/run_*/csd/lmax6.nii.gz`

`nsddata_diffusion/ppdata/subjAA/run_*/csd/lmax8.nii.gz`

`nsddata_diffusion/ppdata/subjAA/run_*/csd/response.txt`



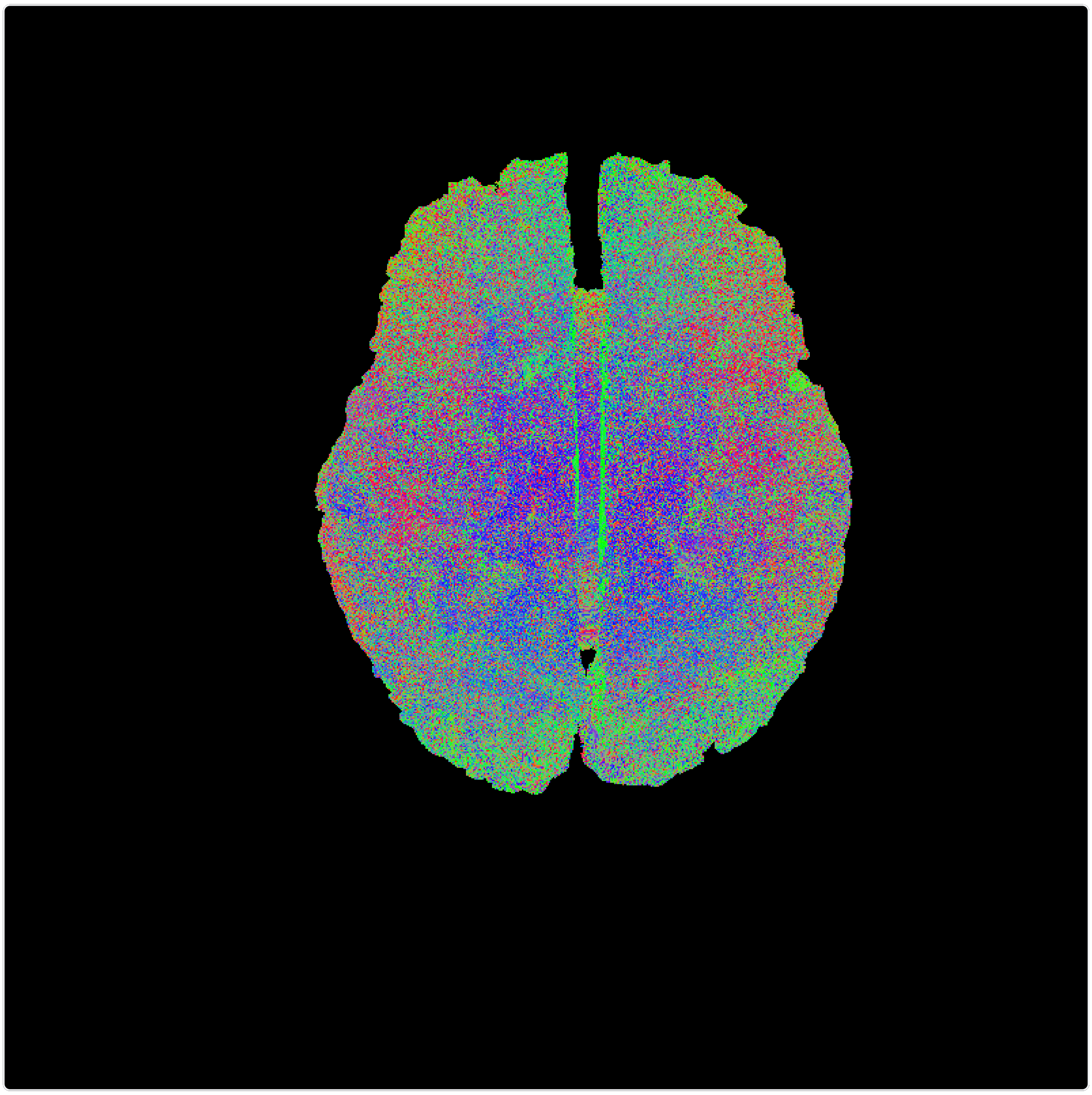
subj07/run_1/csd/lmax8.nii.gz



subj07/run_1/csd/response.txt

Tractography.

Whole-brain diffusion tractography was performed using a brainlife.io App implementing an advanced version of MRTrix3's anatomically-constrained tractography (ACT) methodology (McPherson and Pestilli, Communications Biology, 2021). The multi-shell constrained spherical deconvolution (CSD) model was used to identify fiber orientation distributions. Multiple CSD model orders (L_{max}) were run, namely 6 and 8, and used to separately generate tractograms. Each tractogram was generated with 1.5 million streamlines. The two tractograms were merged into a single tractogram containing 3 million streamlines implementing a simplified version of Ensemble Tractography (Takemura et al., PloS Computational Biology, 2018).



subj07/run_1/track/track-merged.nii.gz

The optic radiations were identified using a novel brainlife.io App (L_{max} 8) using parallel transport tractography implemented in the software library Trekker (Aydogan et al., IEE TMI, 2021). To identify the termination of the Optic Radiation, the LGN as identified with Freesurfer and V1 as identified by the multimodal parcellation were used. 5,000 streamlines were generated for each hemispheric and optic radiation. Left and right Optic Radiations were then merged to generate a single tractogram containing 10,000 streamlines.

```
nsddata_diffusion/ppdata/subjAA/run_*/track/track-lmax6.tck
```

```
nsddata_diffusion/ppdata/subjAA/run_*/track/track-lmax8.tck
```

```
nsddata_diffusion/ppdata/subjAA/run_*/track/track-merged.tck
```

```
nsddata_diffusion/ppdata/subjAA/run_*/track/track-optic-radiation.tck
```

Major white matter tracts segmentation.

The 61 major white matter tracts were segmented using the 3,000,000 whole-brain tractograms. The segmentation was performed using a brainlife.io App implementing an improved version of rules provided by the White Matter Query Language (WMQL; Wassermann et al., Brain Structure and Function, 2016). The segmentation outputs are organized into MatLab files (.mat) containing two cell structures:

1. White Matter Tract Name: the name of each white matter tract (1 x 61 tracts),
2. White matter Tract-streamline Index: the integer index of each tract for every streamline in the whole-brain, merged, tractogram (1 x 3,000,000 streamlines).

Following the tracts segmentation, a brainlife.io App was used to remove outlier streamlines from each tract. Outliers streamlines were defined as those with at least one node x,y,z coordinates more than 3 standard deviations away from the median white matter tract trajectory (i.e., median x,y,z tract coordinates). The resulting outliers' removed white matter tracts classification structure was returned (*`classification-cleaned.mat`*). Finally, a classification structure was generated for the optic radiation tractogram (*`classification-optic-radiation.mat`*), along with a version with outliers removed (*`classification-optic-radiation-cleaned.mat`*).

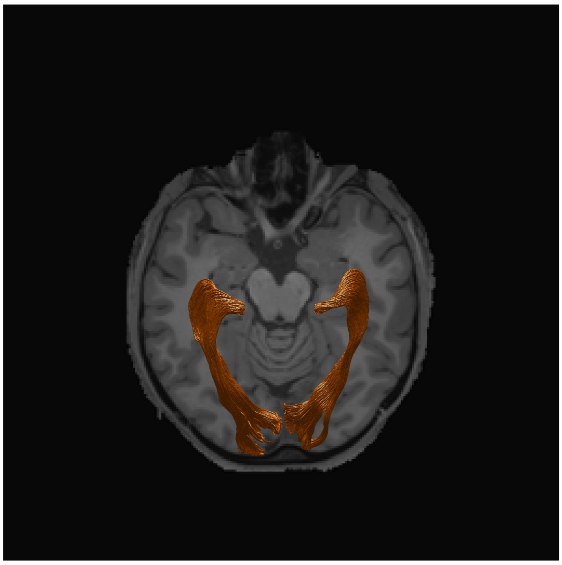
> Note that poor segmentations of the cinguli were returned in both the classification-wholebrain and `classification-wholebrain-cleaned.mat` files for subj02, subj03, subj07, and subj08.

`nsddata_diffusion/ppdata/subjAA/run_*/tract-segmentation/classification-wholebrain.mat`

`nsddata_diffusion/ppdata/subjAA/run_*/tract-segmentation/classification-wholebrain-cleaned.mat`

`nsddata_diffusion/ppdata/subjAA/run_*/tract-segmentation/classification-optic-radiation.mat`

`nsddata_diffusion/ppdata/subjAA/run_*/tract-segmentation/classification-optic-radiation-cleaned.mat`



subj07/run_1/tract-segmentation/classification-optic-radiation-clean.mat

Tract Profiles and macrostructural statistics.

Mapping of DTI, DKI, and NODDI metrics along the core of the segmented whole-brain white matter tracts and the optic radiation using Tract Profiles (Yeatman et al, 2012), and quantitative statistics of macrostructure including tract volume, length, and streamline count provided in a single .csv file following format of AFQ-Browser (Yeatman/Rokem). As brainlife.io treats DTI and DKI as the same datatypes (with differentiating datatype tags), profilometry was performed separately on DTI and DKI measures, but NODDI values were computed in both. These two are designated with a specific directory, specifically *tract-statistics/dti* and *tract-statistics/dki*. Within each directory includes the profiles for the whole-brain segmentation following streamline outlier removal and the optic radiation segmentation following streamline outlier removal.

`nsddata_diffusion/ppdata/subjAA/run_*/tract-statistics/*/tractmeasures-wholebrain.csv` *# whole-brain segmentation statistics derived from either DTI or DKI models and NODDI*

`nsddata_diffusion/ppdata/subjAA/run_*/tract-statistics/*/tractmeasures-optic-radation.csv` *# optic radiation segmentation statistics derived from either DTI or DKI models and NODDI*

Visual area networks.

The merged 3,000,000 whole-brain tractogram was used in combination with the visual areas defined by the multi-modal cortical atlas to build a connectivity matrix of the visual system using a brainlife.io App implementing MRTrix3's method to build networks.

Multiple network measures were generated. Both standard network measures such as fiber count, density, and length as well as more advanced measures derived from the DTI, DKI, and NODDI model were generated.

> Note that the DTI and DKI matrices have been seperated into distinct directories (i.e. visual-area-networks/dti and visual-area-networks/dki). Both directories contain the NODDI matrices generated during the generation of the DTI and DKI matrices. The same networks were then normalized by density. A final network of density normalized by length was also computed. The streamline weights defined by SIFT2 and node assignments are also provided.

`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/density.csv`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/length.csv`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/count.csv`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/{}_mean.csv` # DTI, DKI, NODDI measures
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/{}_mean_density.csv` # DTI, DKI, NODDI measures normalized by density
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/weights.csv`
`nsddata_diffusion/ppdata/subjAA/run_*/visual-area-networks/*/assignments.csv`

Measures of cortical white matter properties.

Diffusion measures derived from DTI, DKI, and NODDI models were mapped to the 'midthickness' surface derived from FreeSurfer following procedures outlined in Fukutomi et al, 2018. Each diffusion model mapping is designated by a `cortexmap-{}` directory. Within each model directory contains a main directory titled *cortexmap*. Within this directory are three sub-directories containing various surface gifti (gii) files: `func, label, surf`.

- *Func* contains the diffusion measures for each model mapped to the cortical midthickness surface, including temporal signal-to-noise ratio (tSNR).
- *Label* contains the Desikan-Killiany (aparc.a2009s) atlas converted to GIFTI.
- *Surf* contains all of the surfaces generated during the procedures, including (but not limited to) the midthickness surface and inflated versions of the midthickness surface. The remaining surfaces are surfaces derived from Freesurfer converted to gifti that were

necessary for generating the midthickness surface and for mapping the diffusion model data to the midthickness surface.

ⓘ Note, the `func.gii` metric surface files, and the GIFTI derivatives, may not load well into FreeSurfer but will load into Connectome Workbench. To ease the burden on users who are more accustomed to FreeSurfer's outputs, .mgh versions of the metric files are also provided. The GIFTI versions of the *pial, white,* and *.label* files are simple conversions of the FreeSurfer outputs using *mris_convert*. The midthickness surface GIFTI surface, to which the dMRI measures of microstructure were mapped, is nearly identical, although derived slightly differently, to the LayerB2 files described in Untitled . However, this only matters if a user wants to replicate the cortex mapping analysis, as the number of vertices between the *func.gii files and the Freesurfer surfaces are the same.

`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap/func/*/*h.{}.func.gii` or `*.mgh` *# hemispheric diffusion measure mapped to midthickness surface in gifti and Freesurfer datatypes*

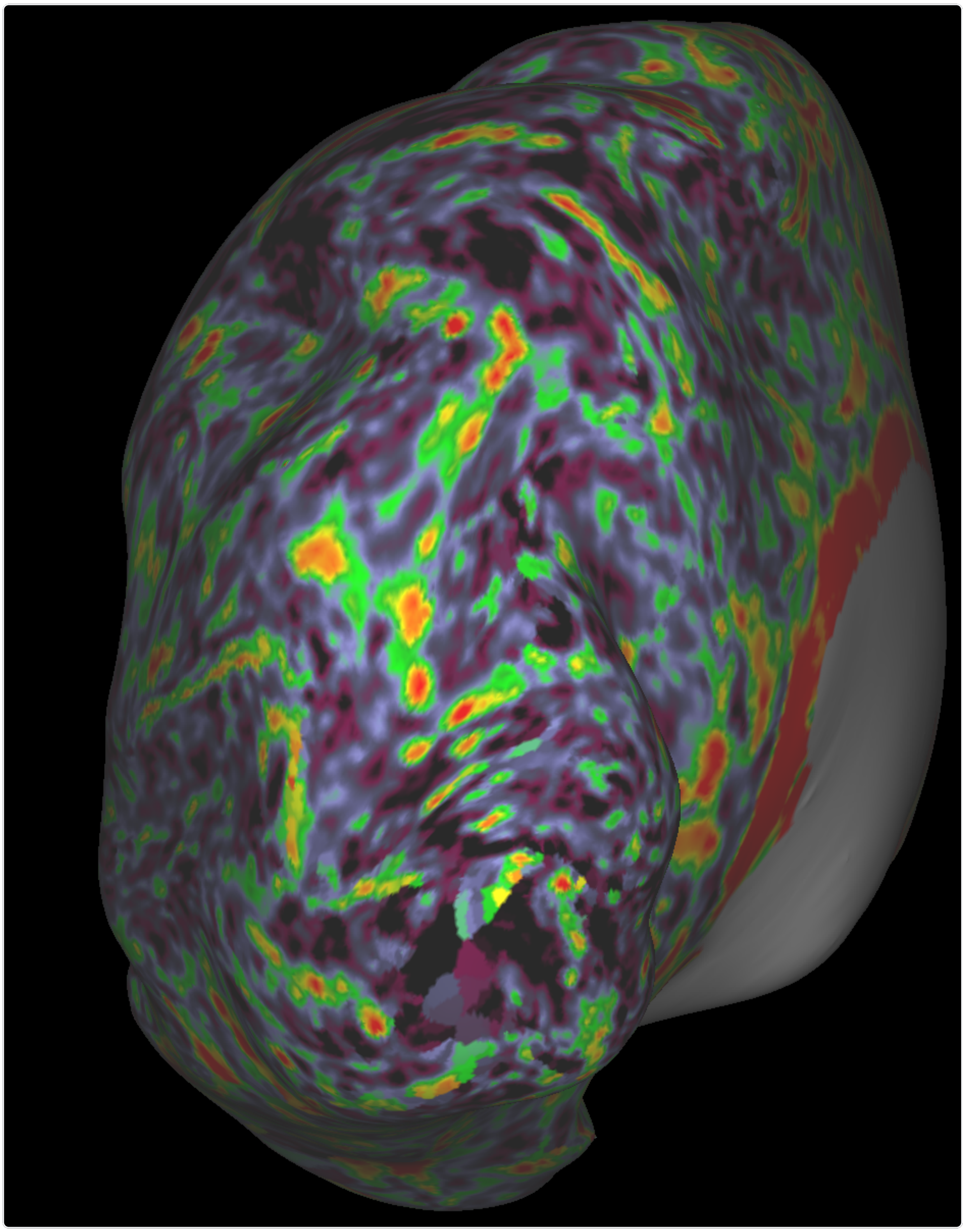
`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap/label/*h.aparc.a2009s.native.label.gii` *# hemispheric Desikan-Killiany (aparc.a2009s) atlas in gifti*

`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap/surf/*h.midthickness.native.surf.gii` *# hemispheric midthickness surface in gifti*

`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap/surf/*h.midthickness.inflated.surf.gii`

# hemispheric inflated midthickness surface in gifti

`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap/surf/*h.midthickness.very_inflated.surf.gii` *# hemispheric inflated midthickness surface in gifti*



subj07/run_1/cortexmap (FA mapped)

Statistics of cortical midthickness mapped diffusion measures.

Mapping of DTI, DKI, and NODDI metrics to the cortical mid thickness surface within both the Desikan-Killiany (aparc.a2009s) and 180 multi-modal cortical node atlases outputted to .csv files is compatible with the format proposed by AFQ-Browser (Yeatman et al., Nature Communications 2017). As brainlife.io treats DTI and DKI as the same datatypes (with differentiating datatype tags), profilometry was performed separately on DTI and DKI measures, but NODDI values were computed in both. These two are designated with a specific directory, specifically `cortexmap-statistics/func/dti` and `cortexmap-statistics/func/dki.` Within each directory includes the number of non-zero vertices (COUNT_NONZERO), minimum (MIN), maximum (MAX), average (MEAN), median (MEDIAN), mode (MODE), and standard deviation (STDEV) of each diffusion-based measure within each parcel found in the Desikan-Killiany (aparc.a2009s; *aparc*) and 180 multi-modal cortical node (hcp-mmp; *parc*) atlases.

`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap-statistics/*/aparc_{}.csv` *# summary statistic for each DTI or DKI, and every NODDI, measure in every parcel in the aparc.a2009s atlas*
`nsddata_diffusion/ppdata/subjAA/run_*/cortexmap-statistics/*/parc_{}.csv` *# summary statistic for each each DTI or DKI, and every NODDI, measure in every parcel in the aparc.a2009s atlas*

Colormap for visual-area parcellation

Below is a table of the ROI parcellations and colormap used to generate the visual area networks and images found in the NSD data paper. Note these are not the exact colors as the colors from the HCP_MMP parcellation.

HCP-MMP Parcel	Color (HEX)	HCP-MMP Parcel	Color (HEX)
lh.v1	#000000	rh.v1	#1CE6FF
lh.vmv1	#FFFF00	rh.vmv1	#FF34FF
lh.mst	#FF4A46	rh.mst	#008941
lh.v6	#006FA6	rh.v6	#A30059
lh.v2	#FFDBE5	rh.v2	#0000A6
lh.vmv2	#7A4900	rh.vmv2	#63FFAC
lh.v3	#B79762	rh.v3	#8FB0FF
lh.vmv3	#004D43	rh.vmv3	#997D87
lh.v4	#5A0007	rh.v4	#809693
lh.v8	#FEFFE6	rh.v8	#1B4400

lh.fef	#4FC601	rh.fef	#3B5DFF
lh.pef	#4A3B53	rh.pef	#FF2F80
lh.v3a	#61615A	rh.v3a	#BA0900
lh.v7	#6B7900	rh.v7	#00C2A0
lh.ips1	#FFAA92	rh.ips1	#FF90C9
lh.ffc	#B903AA	rh.ffc	#D16100
lh.v3b	#DDEEFFFF	rh.v3b	#000035
lh.lo1	#7B4F4B	rh.lo1	#A1C299
lh.lo2	#3000018	rh.lo2	#0AA6D8
lh.pit	#013349	rh.pit	#00846F
lh.mt	#372101	rh.mt	#FFB500
lh.mip	#C2FFED	rh.mip	#A079BF
lh.pres	#CC0744	rh.pres	#C0B9B2
lh.pros	#C2FF99	rh.pros	#001E09
lh.pha1	#00489C	rh.pha1	#6F0062
lh.pha3	#0CBD66	rh.pha3	#EEC3FF
lh.te1p	#456D75	rh.te1p	#B77B68
lh.tf	#7A87A1	rh.tf	#788D66
lh.te2p	#885578	rh.te2p	#FAD09F
lh.pht	#FF8A9A	rh.pht	#D157A0
lh.ph	#BEC459	rh.ph	#456648
lh.tpoj2	#0086ED	rh.tpoj2	#886F4C
lh.tpoj3	#34362D	rh.tpoj3	#B4A8BD

lh.dvt	#00A6AA	rh.dvt	#452C2C
lh.pgp	#636375	rh.pgp	#A3C8C9
lh.ip0	#FF913F	rh.ip0	#938A81
lh.v6a	#575329	rh.v6a	#00FECF
lh.pha2	#B05B6F	rh.pha2	#8CD0FF
lh.v4t	#3B9700	rh.v4t	#04F757
lh.fst	#C8A1A1	rh.fst	#1E6E00
lh.v3cd	#7900D7	rh.v3cd	#A77500
lh.lo3	#6367A9	rh.lo3	#A05837
lh.vvc	#6B002C	rh.vvc	#772600

Visual white matter parcel-color correspondence for visual white matter network analyses. *HCP-MMP parcel ID and Color (hex) correspondence for scatterplots in* ***Results Figure 5b,c.*** *This is also the order of the nodes found in the network matrices in* ***Results Figure 5b.***

Preprocessing applications implemented via *brainlife.io*

Application	Github repository	Open Service DOI	Git branch
Tissue type segmentation	https://github.com/brainlife/app-mrtrix3-5tt	https://doi.org/10.25663/brainlife.app.239	binarize-v1.0
Visual area parcellation	https://github.com/brainlife/app-roiGenerator/	https://doi.org/10.25663/brainlife.app.411	visual-white-matter-glasser-dwi-v1.0
dMRI preprocessing	https://github.com/brainlife/app-FSLTopupEddy	https://doi.org/10.25663/bl.app.287	cuda-v1.0
dMRI-T1 Registration	https://github.com/brainlife/app-epi-t1-	https://doi.org/10.25663/brainlife.app.	v1.0

	registration	286	
SNR Calculation	https://github.com/davhunt/app-snr_in_cc/tree/plot	https://doi.org/10.25663/bl.app.120	plot
Brain mask Generation	https://github.com/brainlife/app-FSLBET	https://doi.org/10.25663/brainlife.app.163	dwi
NODDI model fit	https://github.com/brain-life/app-noddi-amico	https://doi.org/10.25663/brainlife.app.365	1.3
Diffusion Kurtosis Fit	https://github.com/dipy/bl_apps_dipy_fit_dki	https://doi.org/10.25663/bl.app.9	1.1.1
Constrained Spherical Deconvolution Fit	https://github.com/bacaron/app-mrtrix3-act	https://doi.org/10.25663/brainlife.app.238	csd_generation-v1.0
Whole-brain Tractography	https://github.com/bacaron/app-mrtrix3-act	https://doi.org/10.25663/brainlife.app.297	1.3
Merging Tractography Files	https://github.com/bacaron/app-mergeTCK	https://doi.org/10.25663/brainlife.app.305	two-tck
Optic radiation Tractography	https://github.com/brainlife/app-trekker-roi-tracking	https://doi.org/10.25663/brainlife.app.226	optic-radiation-v1.2
Structural Connectome	https://github.com/brainlife/app-sift2-connectome-generation	https://doi.org/10.25663/brainlife.app.394	sift2_v1.2_centers_netneuro
White Matter Anatomy Segmentation	https://github.com/brainlife/app-wmaSeg	https://doi.org/10.25663/brainlife.app.188	3.9
Remove Tract	https://github.com/brainlife/app-	https://doi.org/10.25663/brainlife.app.1	1.3

Outliers	removeTractOutliers	95	
Tract Profiles	https://github.com/brain-life/app-tractanalysisprofiles	https://doi.org/10.25663/brainlife.app.361	1.13
Cortex Tissue Mapping	https://github.com/brainlife/app-cortex-tissue-mapping	https://doi.org/10.25663/brainlife.app.379	v1.2-snr-input
Cortical Summary Statistics	https://github.com/brainlife/app-cortex-tissue-mapping-stats	https://doi.org/10.25663/brainlife.app.383	v1.1

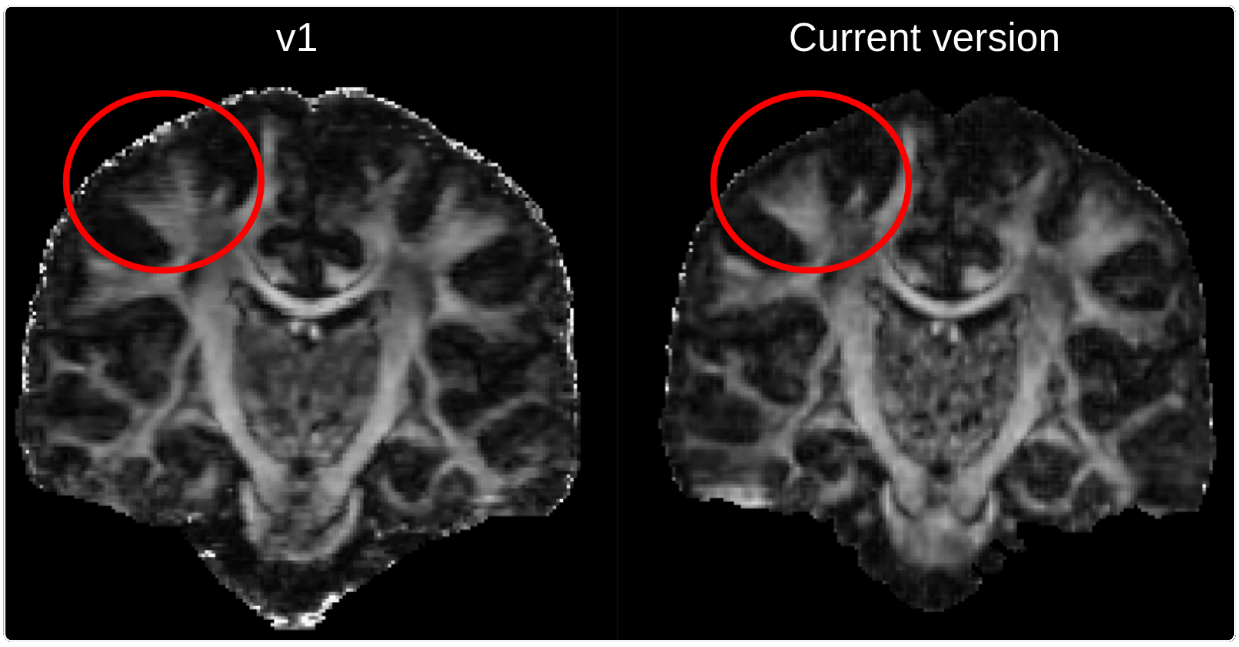
Description and web-links to the open-source code and open cloud services used in the processing of this dataset.

Additional dMRI data preprocessing and data limitations.

The version of the diffusion derivatives that we provide online have some changes with respect to pre-processing compared to what is demonstrated in the NSD data paper. This was done to improve the quality of the diffusion derivatives with respect to strong slice-motion-eddy interactions in the raw dMRI data.

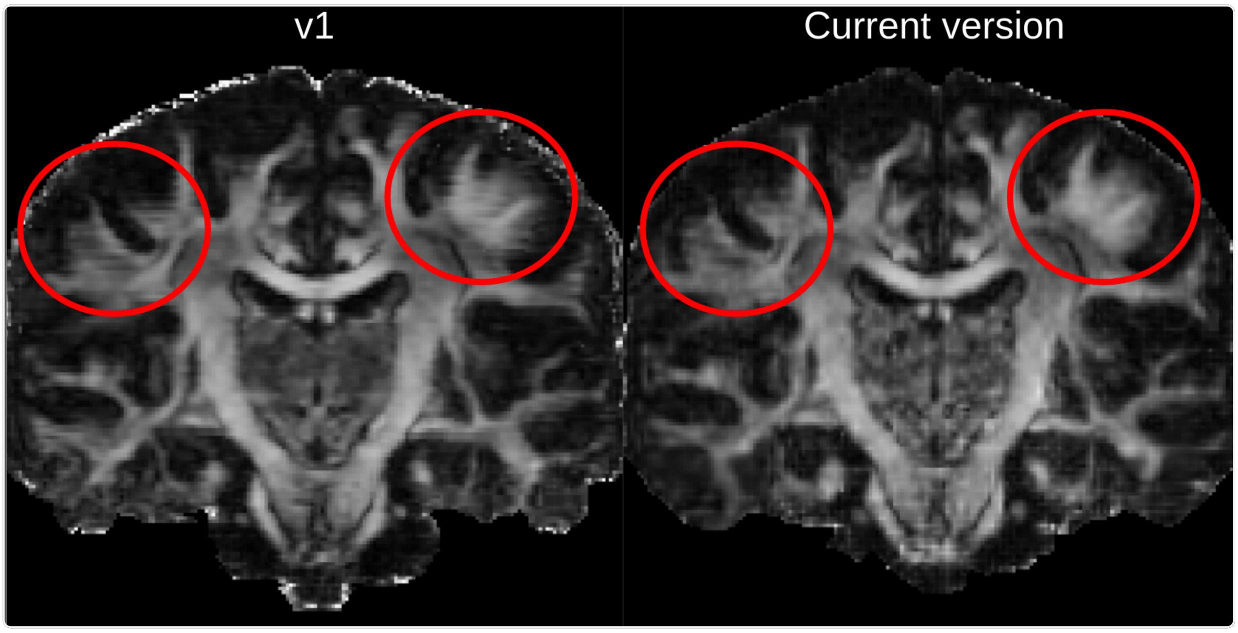
The preprocessing changes involved using only FSL's Topup and Eddy for preprocessing. It is important to note that although this change in the preprocessing corrected a significant amount of the artifact, it may have completely rid the data of the artifact. See screenshots for examples. Following preprocessing, the preprocessed combined dMRI data were aligned to the anatomical (T1w) image and split into the subsequent runs, and all further processing was performed individually on each run separately.

Example of regions where updated preprocessing improved artifact correction.



Example of reduced artifact following updated preprocessing. FA map of subj05 from first version of

Example of regions where updated preprocessing did not completely correct artifact.



Example of subject where preprocessing did not completely alleviate artifact. FA map of subj05 from

ROIs

The NSD dataset comes with a variety of regions of interest (ROIs). Some ROIs are derived from atlases and are automatically determined, whereas other ROIs reflect manual definition based on data from each subject.

Surface-derived ROIs

Some ROIs are generated from surface-based representations of the data. These ROIs include:

- **HCP_MMP1** is the Glasser et al., *Nature*, 2016 atlas.
- **Kastner2015** is the Wang et al., *Cerebral Cortex*, 2015 atlas.
- **nsdgeneral** is a general ROI that was manually drawn on fsaverage covering voxels responsive to the NSD experiment in the posterior aspect of cortex.
- **corticalsulc** is a folding-based atlas defined based on the curvature of fsaverage (sulci, gyri). It labels major sulci and some gyri throughout the whole cortex.
- **streams** is an anatomical atlas that labels various “streams” in visual cortex. It is largely based on fsaverage folding but also takes into account the b3 noise ceiling results to ensure that the regions generally cover where there are stimulus-related signals. More details are provided below.
- **prf-visualrois** is a collection of manually drawn ROIs based on results of the prf experiment. These ROIs consist of V1v, V1d, V2v, V2d, V3v, V3d, and hV4. These ROIs extend from the fovea (0° eccentricity) to peripheral cortical regions that still exhibit sensible signals in the prf experiment given the limited stimulus size (this means up to about ~5-6° eccentricity).
- **prf-eccrois** is a collection of manually drawn ROIs that cover the exact same cortical extent as the prf-visualrois ROIs. These ROIs consist of ecc0pt5, ecc1, ecc2, ecc4, and ecc4+, and indicate increasing “concentric” ROIs that cover up to 0.5°, 1°, 2°, 4°, and >4° eccentricity.
- **floc-faces** is a collection of manually drawn ROIs based on results of the floc experiment. These ROIs consist of OFA, FFA-1, FFA-2, mTL-faces ("mid temporal lobe faces"), and aTL-faces ("anterior temporal lobe faces"). These ROIs were the result of (liberal) thresholding at $t > 0$ (flocfacestval).
- **floc-words** is a collection of manually drawn ROIs based on results of the floc experiment. These ROIs consist of OWFA, VWFA-1, VWFA-2, mfs-words ("mid fusiform sulcus words"), and mTL-words ("mid temporal lobe words"). These ROIs were the result of (liberal) thresholding at $t > 0$ (flocwordtval).

- **floc-places** is a collection of manually drawn ROIs based on results of the floc experiment. These ROIs consist of OPA, PPA, and RSC. These ROIs were the result of (liberal) thresholding at $t > 0$ (flocplacestval).
- **floc-bodies** is a collection of manually drawn ROIs based on results of the floc experiment. These ROIs consist of EBA, FBA-1, FBA-2, and mTL-bodies ("mid temporal lobe bodies"). These ROIs were the result of (liberal) thresholding at $t > 0$ (flocbodiestval).

Note that for the floc-faces, floc-words, and floc-bodies ROIs, not all subjects have all of these ROIs in every hemisphere.

Please note that the floc-related ROIs are quite liberal (given the threshold of t > 0) and will look quite "large" relative to what one may be typically used to. It is a good idea to carefully visualize the ROIs; you can easily whittle down the ROIs using a more stringent threshold if you desire.

The ROIs listed above are initially defined in surface space. For convenience, we have also created volumetric versions of the ROIs. Values of -1 indicate non-cortical voxels in the case of ROIs in volume format. Values of 0 indicate non-labeled vertices/voxels. Positive integers indicate labelings for vertices/voxels.

When surface-based ROIs are converted to volume format, there is an implicit parameter that controls the spatial extent of the volume version. We attempted to create volume ROIs that are not too liberal and not too conservative.

Note that although surface-defined ROIs in floc-faces, floc-words, floc-places, and floc-bodies are guaranteed to be $t > 0$, after conversion to volume space, this constraint may not be entirely still true. If you use the volume versions, you may want to consider further shrinking down these ROIs.

Volume-derived ROIs

Some ROIs are generated from volume-based representations of the data. These ROIs include:

- **thalamus** provides manual segmentation of thalamic regions: LGN, SC, and pulvinar (several subdivisions). Regions were defined in each hemisphere by an expert. Definition was based mostly on T1 anatomical data, but for the pulvinar, MNI-based results from other datasets were projected to each subject to aid ROI definition. Note that as a matter of definition, the ventral pulvinar is most correlated with early visual cortex; the dorsal lateral pulvinar is most correlated with the attention network; and the dorsal medial pulvinar is most correlated with the default-mode network. *Additional information:* LGN and SC were defined based on T1 and T2 image contrast. For the ventral pulvinar, the extent of the pulvinar was defined based on T1 and T2 contrast and then constrained to the ventral lateral portion based on the extent of the two ventral pulvinar maps reported in Arcaro et al., *Journal of Neuroscience,* 2015. The dorsolateral pulvinar was based on the average correlation with IPS maps; and the dorsomedial pulvinar was based on average correlation with precuneus (as reported in Arcaro et al. Nature Communications 2018).

- **MTL** provides manual segmentation of various regions in the medial temporal lobe, including hippocampal subfields. A expert human annotator used the raw high-resolution T2 volumes and manually segmented regions according to Berron et al., *NeuroImage Clinical*, 2017 for each of the 8 NSD subjects. These ROI labelings were then co-registered to the official isotropic T2 volume space and processed.

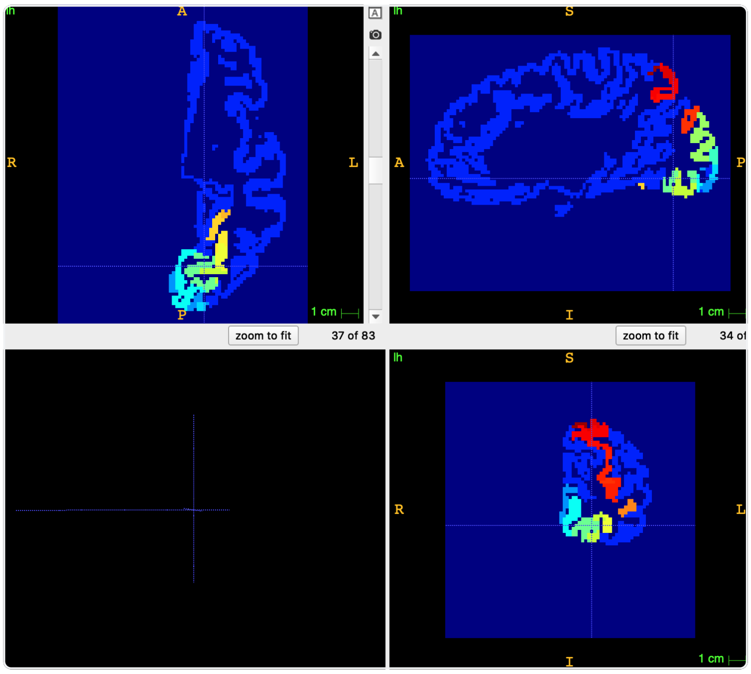
Also, note that ROI labels are mutually exclusive across hemispheres (i.e. every voxel is either assigned to the left hemisphere, right hemisphere, or neither).

ROI files

For convenience, ROI files have been prepared in multiple spaces. ROI files are available in functional spaces (func1pt8mm, func1mm) as well as anatomical spaces (anat). For ROIs in anatomical space, we provide ROIs at 0.8-mm anatomical resolution. ROI files are also available in surface space (FreeSurfer .mgz).

nsddata/ppdata/subjAA/*/roi/[lh,rh].EEE.nii.gz

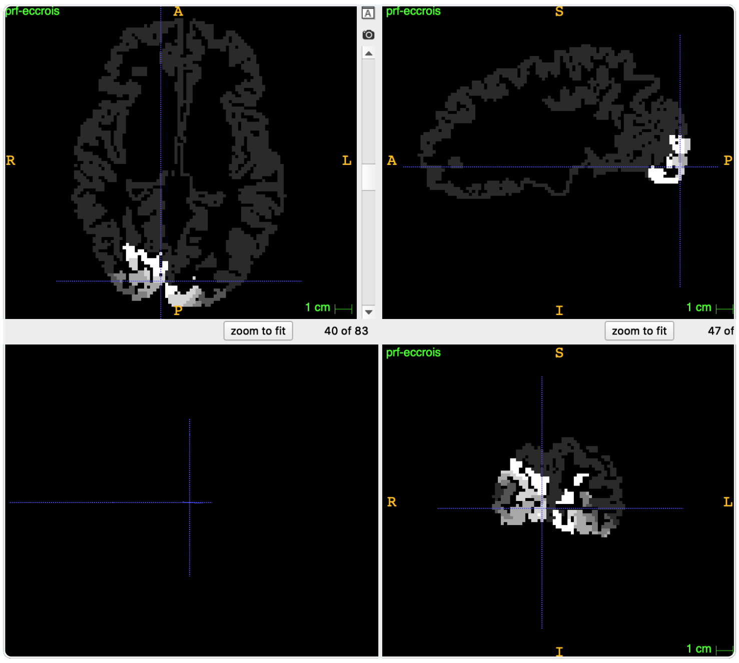
> These are volumes providing integer labels for ROI EEE, generated separately for each hemisphere.



subj01/func1pt8mm/roi/lh.Kastner2015.nii.gz

nsddata/ppdata/subjAA/*/roi/EEE.nii.gz

These are volumes providing integer labels for ROI EEE, combining across hemispheres.



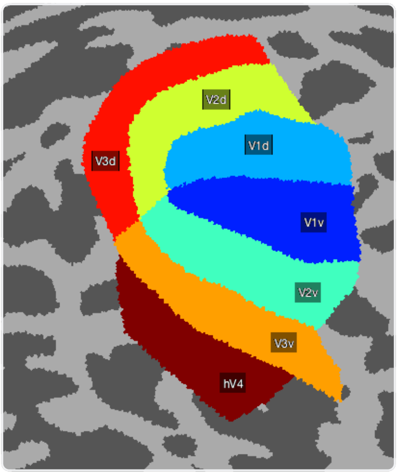
subj01/func1pt8mm/roi/prf-eccrois.nii.gz

nsddata/ppdata/subjAA/anat/roi/other/*.nii.gz

The thalamus and MTL segmentations are originally drawn at 0.5-mm; for completeness, we provide here the original anat0pt5 version of these segmentations, as well as an anat1pt0 version of these segmentations.

nsddata/freesurfer/*/label/[lh,rh].EEE.mgz

These are surface files providing integer labels for ROI EEE.



subj01/label/lh.prf-

nsddata/freesurfer/*/label/EEE.mgz.{ctab,txt}

This is a text file specifying the meaning of the labels in surface-based ROI EEE. (The same labels apply to volume versions of the ROI.) *In other words, this is the critical text file that tells you what each of the integer labels means in terms of ROI names.*

- **prf-visualrois:**
  - **V1d,V1v,V2d,V2v,V3d,V3v** - dorsal and ventral subdivisions of V1, V2, and V3
  - **hV4** - human V4
- **prf-eccrois:**
  - **ecc0pt5,ecc1,ecc2,ecc4,ecc4+** - eccentricity-restricted regions within early visual areas V1, V2, and V3
- **floc-faces:**
  - **OFA** - occipital face area
  - **FFA-1** - posterior section of fusiform face area
  - **FFA-2** - anterior section of fusiform face area
  - **mTL-faces** - face-selective region in middle portion of temporal lobe
  - **aTL-faces** - face-selective region in anterior portion of temporal lobe
- **floc-words:**

        - **OVWFA** - occipital visual word form area
        - **VWFA-1** - posterior section of visual word form area
        - **VWFA-2** - anterior section of visual word form area
        - **mfs-words** - word-selective region located near the mid-fusiform sulcus
        - **mTL-words** - word-selective region in middle portion of temporal lobe
    - **floc-places:**
        - **OPA** - occipital place area
        - **PPA** - parahippocampal place area
        - **RSC** - retrospenial cortex (place-selective)
    - **floc-bodies:**
        - **EBA** - extrastriate body area (can also be referred to as LOTC-bodies (lateral occipitotemporal cortex))
        - **FBA-1** - posterior section of fusiform body area (can also be referred to as VOTC-bodies-1 (ventral occipitotemporal cortex))
        - **FBA-2** - anterior section of fusiform body area (can also be referred to as VOTC-bodies-2)
        - **mTL-bodies** - body-selective region in middle portion of temporal lobe

nsddata/templates/EEE.ctab

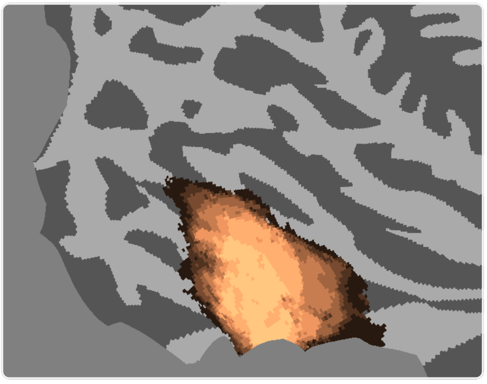
This is a text file specifying the meaning of the labels in volume-based ROI EEE. For example, EEE can be "thalamus" or "MTL".

Probmap files

For convenience, we also create "probmap" (probabilistic map) results. Specifically, we take each manually defined cortical ROI and map these via nearest-neighbor interpolation to fsaverage and then compute the fraction of subjects at each vertex that has each individual ROI present.

nsddata/freesurfer/fsaverage/label/[lh.rh].RRR.mgz

This file consists of fractions between 0 and 1. The value indicates the fraction of subjects that have ROI RRR present at a given fsaverage vertex.



fsaverage/label/rh.PPA.mgz

Other files

The following files are intermediate files created in the process of the manual segmentation of the MTL ROI collection.

nsddata/ppdata/subjAA/anat/HRT2/HRT2_raw.nii.gz

> The raw high-resolution T2 volume used for MTL segmentation.

nsddata/ppdata/subjAA/anat/HRT2/HRT2_mask.nii.gz

> The binary mask within which an affine transformation was optimized to match the official 0.5-mm T2 volume.

nsddata/ppdata/subjAA/anat/HRT2/MTL_rawlabels.nii.gz

> The manually defined MTL labels (same space as the HRT2_raw.nii.gz volume).

nsddata/ppdata/subjAA/anat/HRT2/T2matched.nii.gz

> Given the affine transformation determined (within the HRT2_mask), this volume is the result of reslicing through the 0.5-mm T2 volume to match the HRT2_raw volume (cubic interpolation).

Additional information on the *streams* ROIs

Early visual cortex ROI:

- The early visual cortex ROI was drawn as the union of the V1v, V1d, V2v, V2d, V3v and V3d ROIs from the Wang 2015 retinotopic atlas. Additionally, V2v and V2d were connected such that the part of the occipital pole typically containing foveal representations was also included. The same was repeated for V3v and V3d.

Intermediate ROIs:

- Three intermediate ROIs were drawn corresponding to each of the three streams: ventral, lateral and parietal. All three ROIs border the early visual cortex ROI on the posterior side.
- The intermediate ventral ROI was drawn to reflect the inferior boundary of hV4 from the Wang atlas and to include the inferior occipital gyrus (IOG), with the anterior border of the ROI drawn based on the anterior edge of the inferior occipital sulcus (IOS).
- The intermediate lateral ROI was drawn directly superior to the intermediate ventral ROI, with the superior and anterior borders determined as the LO1 and LO2 boundaries from the Wang atlas.
- The intermediate parietal ROI was drawn directly superior to that, reflecting exactly the borders of the union of V3A and V3B from the Wang atlas.

Higher-level ROIs:

- Three higher-level ROIs were drawn for each of the ventral, lateral and parietal streams, bordering their respective intermediate ROIs on their posterior edges.
- The ventral ROI was drawn to follow the anterior lingual sulcus (ALS), including the anterior lingual gyrus (ALG) on its inferior border and to follow the inferior lip of the inferior temporal sulcus (ITS) on its superior border. The anterior border was drawn based on the midpoint of the occipital temporal sulcus (OTS).
- The lateral ROI was drawn such that the higher-level ventral ROI was its inferior border and the superior lip of the superior temporal sulcus (STS) was used to mark the anterior/superior boundary. The rest of the superior boundary traced the edge of angular gyrus, up to the tip of the posterior STS (pSTS).
- The parietal ROI was drawn to reflect the boundary of the lateral ROI on its inferior edge and to otherwise trace the borders of and include the union of IPS0, IPS1, IPS2, IPS3, IPS4, IPS5 and SPL1 from the Wang atlas.

Technical notes

This section contains a number of technical details that help document the NSD dataset.

Final numbers

Some of the NSD subjects did not complete all 40 planned NSD core scan sessions. Here we provide some useful summary statistics on what is present in the NSD dataset. Note that the numbers are calculated with respect to the full dataset).

- How many core NSD scan sessions did each of the 8 NSD subjects complete?
  [40 40 32 30 40 32 40 30]
- How many distinct images were shown at least once to each subject?
  [10,000 10,000 9,411 9,209 10,000 9,411 10,000 9,209]
- How many distinct images were shown at least twice to each subject?
  [10,000 10,000 8,355 7,846 10,000 8,355 10,000 7,846]
- How many distinct images were shown all three times to each subject?
  [10,000 10,000 6,234 5,445 10,000 6,234 10,000 5,445]
- How many trials did each subject perform?
  [30,000 30,000 24,000 22,500 30,000 24,000 30,000 22,500]
- How many of the shared 1,000 images were shown at least once to each subject?
  [1,000 1,000 930 907 1,000 930 1,000 907]
- How many of the shared 1,000 images were shown all 3 times to every subject?
  515
- How many of the shared 1,000 images were shown at least 2 times to every subject?
  766
- How many of the shared 1,000 images were shown at least once to every subject?
  907
- What is the total number of distinct images, aggregated across all subjects?
  70,566
- What is the total number of trials, aggregated across all subjects?
  213,000

Data sizes

The following are the matrix dimensions for the high-res (1.0-mm) functional data preparation, matrix dimensions for the standard-res (1.8-mm) functional data preparation, the vertex number in the left-hemisphere cortical surfaces, and the vertex number in the right-hemisphere cortical surfaces.

```
Subject 1       [145 186 148]   [81 104 83]    227021 226601
Subject 2       [146 190 150]   [82 106 84]    239633 239309
Subject 3       [145 190 146]   [81 106 82]    240830 243023
Subject 4       [152 177 143]   [85 99 80]     228495 227262
Subject 5       [141 173 139]   [79 97 78]     197594 198908
Subject 6       [152 202 148]   [85 113 83]    253634 259406
Subject 7       [139 170 145]   [78 95 81]     198770 200392
Subject 8       [143 184 139]   [80 103 78]    224364 224398
```

On the issue of valid voxels

Due to spatial distortion and/or head displacement over the course of a scan session, voxels on the edges of the imaged volume may not obtain a full set of data for that session. In pre-processing, such voxels are detected, deemed “invalid”, and are essentially set to 0 for the whole scan session. For the most part, brain voxels of interest are almost always valid.

The files named valid*.nii.gz provide information regarding which voxels contain valid data. Invalid voxels exhibit the following behavior:

- timeseries*.nii.gz – Invalid voxels have pre-processed time-series data values that are all zeroes over the course of the entire scan session.
- mean*.nii.gz – Invalid voxels have a mean intensity of 0.
- R2*.nii.gz – Invalid voxels have a GLM variance explained value of NaN.
- betas*.[nii.gz,hdf5] – Invalid voxels have betas that are all zeroes. (This is the result of the data being saved in int16 format, which converts NaNs to 0.)
- meanbeta*.nii.gz – Invalid voxels have mean betas equal to 0.
- onoffbeta*.nii.gz – Invalid voxels have onoffbeta weights equal to NaN.

Note that voxels outside of the brain mask are also set to 0 in the time-series data and in the beta weights; thus, they appear similar to invalid voxels.

Computational tips

The massive scale of the NSD dataset poses some computational challenges. Here we comment on some issues related to computational efficiency.

- File format choices are important. HDF5 provides fast access because it is uncompressed.
- Pre-allocation of variables when loading data into memory is important (otherwise, unnecessary time costs are incurred).
- Consider using 'single' or 'float' format to save memory usage.
- For huge data, breaking up the analysis into chunks may be necessary in order to reduce memory usage (e.g., analyze one subject at a time).
- In general, when loading in chunks from an HDF5 file, it is fastest to load chunks from the last dimension. However, the HDF5 files used for the NSD betas were saved with ChunkSize [1 1 1 750], which means that the trials were deliberately chunked together when saved. This was done because in theory, one will probably want to always get all of the trials (from a given set of voxels). Speed benefits for the NSD betas would be obtained when loading chunks from the third dimension (as opposed to the first or second dimensions).
- Vectorization of code is important (avoid for-loops if possible).
- If averaging across trials for the same image, one can do this efficiently through a single indexing operation (e.g. an indexing matrix that is 3 trials x N images), as opposed to using a for-loop.

Timing issues

Here is how timing issues are dealt with in the NSD dataset:

- An empirical audio check of a typical fMRI scanning run (i.e. an NSD run involving 188 volumes at a TR of 1.6 s) indicates the following breakdown: There is 31.8 s from the start of scanner calibration noises to the start of the EPI noises; then, there is 8 s from the start of EPI noises until the start of the first actual recorded fMRI volume (the 8 s is due to dummy fMRI volumes); and, finally, there is 300.8 s (i.e. 188*1.6) from the start of the first recorded fMRI volume until the end of the EPI noises (indicating that data collection is complete). Thus, the dummy fMRI volumes are already dropped and do not show up in the NSD dataset. We consider the start of the first recorded fMRI volume to be time = 0.
- The fMRI volumes are acquired at 1600 ms TR, and this is assumed to be exactly accurate. Empirical measurements of scanner triggers, as detected by the stimulus computer, indicate that the difference between successive triggers is consistently between 1599.95 and 1600.12 ms. Some of this variability is due to polling uncertainty. We believe this is good validation that the 1600 ms number can be trusted.

- The stimulus computer controls the experiment presentation. The presentation code locks to the display rate of the BOLDscreen monitor, and empirical measurements of the duration of each 5-min (300 s) run come out to consistently between 299.955 s and 299.97 s. Thus, we are confident that the timing of the experimental presentation is highly reliable. Because these values are not exactly 300.000 s, in the pre-processing of the fMRI data, we resample the fMRI data to a sampling rate of 0.999878 s. (Note that 0.999878*300 = 299.9634 s.) Specifically, the high-resolution (func1mm) preparation of the data uses a new sampling rate of 0.999878 s, while the low-resolution (func1pt8mm) preparation of the data uses a new sampling rate of (0.999878)*(4/3) = 1.3331707 s. These numbers are quite close to 1 s and 4/3 s, respectively, and we often abbreviate using those numbers for simplicity.
- Note that the fMRI acquisition extends slightly longer than the experiment duration. For example, for a typical NSD run, the experiment lasts 299.9634 s, while the fMRI acquisition lasts 188 * 1.6 = 300.8 s. This is intentional and no cause for concern.
- With respect to the pre-processing of the fMRI data, the total duration of the func1mm preparation of each fMRI run is 0.999878 * 301 volumes = 300.96 s. The total duration of the func1pt8mm preparation of each fMRI run is (0.999878)*(4/3) * 226 volumes = 301.29 s. Notice that the two numbers are slightly different, and extend slightly beyond the original extent of the acquisition (1600 ms * 188 volumes = 300.8 s). This is all expected behavior, and is due to how the pre-processing code decides to place the final time points.
- After the pre-processing of the fMRI data, it is convenient to simply interpret the fMRI data as being sampled at a rate of 1 s (or 4/3 s), even though that is not exactly accurate.
- Slice acquisition order was determined from the DICOM header of the fMRI volumes. In the temporal pre-processing of the fMRI data, all slices were sampled to be coincident with the first (temporally) acquired slices. (Note that multiple slices were “first” because of the multiband acquisition.)
- The experimental design comes in 4-s trials; thus, fMRI volumes after pre-processing land exactly on the onset of each trial (4 s is divisible by 1 s and by 4/3 s).
- At the beginning of each run, the stimulus computer waits for a trigger to be sent by the MRI scanner, and once the trigger is detected, the computer starts the experiment. Note that there is a brief and somewhat variable (about 5-20 ms) delay that persists between the detection of the trigger and the first stimulus frame shown (e.g. due to the fixed refresh rate of the monitor). Thus, there may be a small (and more or less fixed) delay between the fMRI data and the stimulus frames. This seems like a relatively minor issue: the readout of the first slice in the EPI sequence itself takes some time, so there is already a delay (e.g. half of the readout window) that is essentially being ignored here.
- The internal MR scanner clock shows some odd behavior. According to the stored AcquisitionTime header of the EPI DICOMs, we extracted the average duration of each TR volume and that number comes out to 1606.425 ms. This is surprising since the empirical measurements from the stimulus computer indicate that the TR (as reflected in

the triggers that are sent by the scanner) is essentially exactly 1600 ms. Checks that we performed strongly suggest that, for the purposes of internal times recorded by the scanner in the DICOMs and in the physiological data, it does seem that the MR scanner believes the DICOMs come at a rate of 1606.425 ms. We found that under the assumptions we make when extracting the physiological data, the physiological data and the DICOM times are very nicely consistent with one another. Moreover, the number of samples that we extract corresponding to the actual fMRI acquisition does empirically turn out to be around 15040-15041, which is essentially exactly 50 Hz for a run duration of 188*1.6=300.8 s. Thus, our working interpretation is that (i) the correct time is being recorded by the stimulus computer; (ii) the MR scanner in fact achieves exactly the time requested (1600 ms TR); (iii) the MR scanner has some strange internal timing system that is internally consistent but which does not match the stimulus computer’s timing, and (iv) the user need not worry about the strange MR scanner timing.

FreeSurfer notes

- FreeSurfer includes an internal T1 volume (e.g. mri/T1.mgz). Beware that although this volume contains basically the same image data as the original 0.8-mm anatomical volume that we provided to FreeSurfer, it has some header differences. Thus, if you were to load in the raw image data from the two volumes, in order to get them to match up, you may have to apply a specific set of flips, rotations, and shifts. This is because the orientation and exact positioning of the two volumes are different. A NIFTI-header-aware application that knows how to properly interpret the orientation and origin information will reveal that the two volumes are identical, in the sense that both volumes, when properly interpreted, are in the same position (e.g. (0,0,0) in millimeters corresponds to the same location in the two volumes). The following shows how the image data (ignoring headers) can be matched between the two volumes.

```
% load aseg
sourcedata = '~/nsd/nsddata/freesurfer/subj01/mri/aseg.mgz';
vol = cvnloadmgz(sourcedata);

% bring it to our anat0pt8 space
vol = flipdim(flipdim(permute(vol,[1 3 2]),3),1);
volB = zeros(size(vol));
volB(2:end,:,2:end) = vol(1:end-1,:,1:end-1);
```

  Note that we have converted some of the standard FreeSurfer output volumes to conform to the formats used for the NSD data. For example: nsddata/ppdata/subj01/func1pt8mm/aseg.nii.gz

- The FreeSurfer surfaces (e.g. lh.white) have coordinates that must be interpreted with respect to the FreeSurfer headers. This is quite tricky, and requires using the FreeSurfer vox2ras and vox2ras-tkr information. Here is the basic idea (see preprocess_nsd_calculatetransformations.m) for how we map FreeSurfer's surface coordinates to a 1-based coordinate system that corresponds to the official T1 0.8-mm anatomical volume:

  newcoord = inv(M)*Norig*inv(Torig)*[tkrR tkrA tkrS 1]' + 1

  where [tkrR tkrA tkrS] are coordinates stored in the surface file, Torig is the output from vox2ras-tkr, Norig is the output from vox2ras, and M is the voxel-to-world transformation from the official T1 0.8-mm anatomical volume. The idea is that we first map from surface coordinates to 0-based pixel (CRS) space (i.e. inv(Torig)), then we map from FreeSurfer's 0-based pixel space to physical RAS space (i.e. Norig), and then we map from physical RAS space to 0-based pixel space associated with the official T1 0.8-mm anatomical volume. Finally, we add 1 to the coordinates in order to convert to 1-based pixel space (i.e. 1 means the center of the first voxel).
- In the diffusion files (nsddata_diffusion), various cortical surfaces are provided in GIFTI format. The coordinates contained in these GIFTI files are "world coordinates" and they are identical to the surface coordinates contained in the usual FreeSurfer surface files after making sure to convert the surface coordinates to physical RAS space.

MNI notes

- All NIFTI files that we write are in LPI ordering (the first voxel is Left, Posterior, and Inferior). This applies even to files written by nsd_mapdata in the MNI space. Note that this is the same as what FreeSurfer calls "RAS" ordering, since that nomenclature refers to which directions have increasing voxel indices.
- The MNI template (1mm) (borrowed from FSL) has matrix dimensions [182 218 182] and is in RPI ordering (first voxel is right, posterior, inferior). The origin lies at 1-based image coordinates (91,127,73).
- NSD files provided in MNI (1mm) space have the same matrix dimensions [182 218 182] and are in LPI ordering. The origin lies at 1-based image coordinates (92,127,73). Note that while the MNI template is in RPI ordering, NSD files that are provided in MNI space are in LPI ordering. When comparing these two types of files in an application that understands and respects the NIFTI header information, everything should be correct and in correspondence.
- When using nsd_mapdata to map from MNI to some other space, **note that the source data is expected to be in RPI ordering (since that is what the MNI template uses)**. This means that if one performs analyses of, for example, the NSD beta weights prepared

in MNI space (which have LPI ordering), the results need to be flipped along the first dimension before being passed to nsd_mapdata.m.

  - Furthermore, when trying to map MNI source data, the data should be EXACTLY in the same resolution, matrix size, etc. as the MNI 1mm template. (For example, if your MNI source data is 2-mm, you need to bring it to 1-mm resolution.) There are many ways to do this; one option is resliceniftitomatch.m as provided in https://github.com/cvnlab/knkutils/

- When using nsd_mapdata to map to MNI space, **note that the output variable is returned to the workspace in RPI ordering. But notice that if you ask nsd_mapdata to write out a NIFTI file, that file has data stored in LPI ordering.**
- All NIFTI files that we write have the origin set to the exact center of the image slab. The only exception to this is when nsd_mapdata writes out MNI space files: in this case, we set the origin to match that used in the MNI template files.

Other notes

- Recorded reaction times in the behavioral data have some rounding error due to the presentation of images at a 10 Hz rate. That is, the stimulus computer both controls image presentation and tries to record button presses. Approximately every 100 ms, the stimulus computer has to do work to present the image, and at these points in time, if there is a button that is pressed, it will be logged a few milliseconds late. (You will see this weird effect if you plot a histogram of a large number of RTs in bin widths of 1 ms.)
- Note that the func1pt8mm and func1pt0mm have origins that are in slightly different places. This is because the field of view of the two preparations are different and because we set the origin to be the center of the image slab in both cases.

Transform files

Various coregistration procedures were performed in the pre-processing of NSD data, and the results of these procedures have been written out to a collection of files. Essentially, we have pre-computed a large number of possible mappings that the user might want to perform. These pre-computed transform files are used by the nsd_mapdata utility in order to map data from one space to another, and ordinary users should not need to worry about the contents of these files.

nsddata/ppdata/subjAA/transforms/

This directory contains the set of pre-computed transform files for subject AA.

Note that file format conventions vary across different software packages. Thus, these files are not necessarily "standard" and not necessarily compatible "off the shelf" with a given software package!

The basic form of a filename is "X-to-Y", indicating that this file contains information on how to access data from X for each location in Y. For example, "func1pt0-to-MNI.nii.gz" is a NIFTI file with the dimensionality of the 1-mm MNI space; there are three volumes in this file, corresponding to three spatial dimensions; and each value indicates how to pull from the 1.0-mm functional space. Intuitively, this file provides func1pt0 coordinates in an MNI-like volume.

Our convention is to use image coordinates for volume data. For example, 1 is the center of the first voxel; 2 is the center of the second voxel; and 1.5 is exactly in between the centers of the first and second voxels. Furthermore, our convention is to use 1-based indexing for surface data. For example, 1 indicates the first surface vertex.

To conform to FreeSurfer conventions, files named like "lh.X-to-Y.mgz" indicate how to access data from X for each location in the left hemisphere Y surface. For example, "lh.func1pt8-to-layerB2.mgz" indicates, for each surface vertex in the mid-gray left hemisphere cortical surface, how to pull data from the 1.8-mm functional volume.

For transform files involving fsaverage, all values are indices and not spatial locations (since our convention is to use nearest-neighbor interpolation for fsaverage-related transformations).

Additional documentation can be found in preprocess_nsd_calculatetransformations.m.

Code

Several code resources are provided with the NSD dataset. See the nsddatapaper github repository for an archive of code used in the NSD data paper. Below, we document other resources.

nsd_mapdata

We provide a lightweight github repository:

http://github.com/kendrickkay/nsdcode/

This repository contains the utility nsd_mapdata.{m,py}, which helps map data between the various spaces used in the NSD dataset (see Spaces for imaging data). In brief, transformations between various spaces (e.g. functional, anatomical, MNI, fsaverage) have been pre-computed, and the utility simply loads in these transformations and applies them to user-supplied data.

Example scripts demonstrating usage of nsd_mapdata are provided: examples_nsdmapdata.{m,py}. In addition, we provide a video that walks through the example script: https://www.youtube.com/watch?v=XeiyFEr29gA

nsd_mapdata
https://www.youtube.com/watch?v=XeiyFEr29gA

Some notes on using nsd_mapdata:

- Three types of interpolation are available: nearest-neighbor, linear, or cubic.
- Be careful about the choice of interpolation. In particular, when mapping volume data to the cortical surface, it is easy for "holes" to occur, depending on the extent to which valid values exist in the volume data and depending on the type of interpolation used.
- In general, transformation between volume and surface spaces is lossy, in the sense that information loss and discretization errors are inevitable. One strategy is to perform analysis of the functional data fully in volume format and then transform to surface space at the very end (e.g. for visualization). A different strategy is to simply start up front with the "nativesurface" preparation (in which we have already transformed/interpolated the NSD betas to FreeSurfer's surface space) and then conduct analyses.
- The conversion of surface data to volume format is a tricky procedure that involves certain assumptions. One particular method is implemented by nsd_mapdata (and is described in

the NSD data paper), and this method was used in order to create volumetric versions of surface-oriented ROI labels (e.g. prf-visualrois). Other methods are possible.

Example text for papers

The following is a text template that may be useful for briefly describing the NSD dataset in a paper that uses the NSD data. Of course, you may need to modify or expand as necessary.

> *Natural Scenes Dataset*
>
> A detailed description of the Natural Scenes Dataset (NSD; http://naturalscenesdataset.org) is provided elsewhere {cite Allen et al., Nature Neuroscience, 2021}. The NSD dataset contains measurements of fMRI responses from 8 participants who each viewed 9,000–10,000 distinct color natural scenes (22,000–30,000 trials) over the course of 30–40 scan sessions. Scanning was conducted at 7T using whole-brain gradient-echo EPI at 1.8-mm resolution and 1.6-s repetition time. Images were taken from the Microsoft Common Objects in Context (COCO) database {cite Lin 2014}, square cropped, and presented at a size of 8.4° x 8.4°. A special set of 1,000 images were shared across subjects; the remaining images were mutually exclusive across subjects. Images were presented for 3 s with 1-s gaps in between images. Subjects fixated centrally and performed a long-term continuous recognition task on the images. The fMRI data were pre-processed by performing one temporal interpolation (to correct for slice time differences) and one spatial interpolation (to correct for head motion). A general linear model was then used to estimate single-trial beta weights. Cortical surface reconstructions were generated using FreeSurfer, and both volume- and surface-based versions of the beta weights were created.
>
> *Natural Scenes Dataset (extremely abbreviated)*
>
> The NSD dataset contains measurements of 7T fMRI responses (1.8 mm, 1.6 s) from 8 participants who each viewed 9,000–10,000 distinct color natural scenes (22,000–30,000 trials). Subjects fixated centrally and performed a long-term continuous recognition task on the images.

Other snippets of text that might be useful as a template:

> The dataset includes additional measures including structural (T1, T2), diffusion, and resting-state data.

In this paper, we used the 1.8-mm volume preparation of the NSD data and version 3 of the NSD single-trial betas (betas_fithrf_GLMdenoise_RR).

We used the ‘nativesurface’ preparation of the NSD betas.

We used the nsd01–nsd10 scan sessions from all 8 NSD subjects.

If you make use of the NSD dataset, please cite the NSD data paper:

Allen, St-Yves, Wu, Breedlove, Prince, Dowdle, Nau, Caron, Pestilli, Charest, Hutchinson, Naselaris*, & Kay*. A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience* (2021).

In addition, please acknowledge the NSD funding sources using wording similar to:

Collection of the NSD dataset was supported by NSF IIS-1822683 and NSF IIS-1822929.

FAQ

This page lists frequently asked questions about the NSD dataset. If you have questions that are not answered on this page, please post to the nsd-users mailing list. This FAQ will be updated as questions arise.

- *"I need help double-checking the indexing of the images (e.g. figuring out which images were seen by all subjects). Do you have any pointers?"*

  The following script might be helpful to see some examples of how to handle tricky indexing. Note that this is MATLAB, so the indices are generally 1-based in this script: https://github.com/cvnlab/nsddatapaper/blob/main/mainfigures/FINALNUMBERS/FINALNUMBERSnotes.m

  Note that the full set of 40 NSD scan sessions were collected for four of the eight subjects but that only the first 30 or 32 NSD scan sessions were collected for the other four subjects. Hence, for exact numbers you must take this into account. Also, note that the numbers of images for which responses are available depend on whether you have access to all collected NSD scan sessions or not. ~~(Remember that the last 3 NSD scan sessions from each subject are held-out from public release due to the Algonauts challenge. For example, there were actually 40 NSD scan sessions collected for subject 1, but only the first 37 scan sessions are publicly downloadable.)~~

  Here is a simple example showing how to determine which of the shared 1,000 images were seen all 3 times by all 8 subjects (assuming that the full dataset is downloaded). Note the subjects who had the fewest scan sessions had 30 NSD scan sessions. Hence, we use "30" in the code snippet below. (To figure out which of the shared 1,000 images were seen all 3 times by all 8 subjects within the currently downloadable data, you would simply substitute "27" for "30" in the code snippet below.)

```
temp = masterordering(1:750*30);  % 750 trials per session; all 8 subjects participated in at least the first 30 NSD scan sessions
shared515 = [];  % 1 x 515 vector of 1-indices. these indices are between 1-1000.
for q=1:1000
  if sum(temp==q)==3
    shared515 = [shared515 q];
  end
end
```

- *"Can I average prf time-series data across runs? Do they have different stimulation protocols?"*

  As described in the NSD data paper, the prf experiment involved six runs acquired as BWBWBW (where B and W refer to multibar and wedgering run types). The spatial aperture pattern was identical within each run type, and hence averaging is reasonable (e.g. average the Bs together; average the Ws together). (However, note that the specific colorful texture shown at a given point in time and the precise fixation dot behavior are stochastic across runs and are therefore not exactly identical across runs.)

- *"I am getting "access denied" errors and/or I am getting 403 errors from AWS ("fatal error: An error occurred (403) when calling the HeadObject operation: Forbidden"). Can you help?"*

  A subset of the NSD data files (e.g. nsdsynthetic, nsdimagery) is forbidden from being downloaded at this point (see 'Held-out data' on the Untitled page).

- *"I want to transform the surface-defined ROI masks provided with NSD into a format that works with pycortex. How do I do this?"*

  Pycortex involves creating .svg files for ROI masks. Please see https://github.com/gallantlab/pycortex/issues/312 for more information.

- "What is the breakdown of image databases NSD pulls from and what resources/annotation already exist for those?"

  All NSD images come from the Microsoft COCO image database. As for resources, there are a number of online 'computer vision' resources that provide a wealth of annotations on the COCO images (see http://cocodataset.org/#external). In addition, note that the externally contributed nsd_access toolbox (see General Information) provides a convenient Python interface for understanding how the images selected for use in NSD are mapped onto the COCO images.

- "How do I load in the NSD-generated ROI files, like lh.floc-faces.mgz, into FreeSurfer's freeview? It won't load in freeview as a surface annotation."

If you use the "Overlay → Load generic..." option, freeview should be able to load and interpret the surface data in the .mgz files.

- "How do I map from MNI to fsaverage and/or vice versa?"

  Since MNI and fsaverage are fundamentally different in nature (volume vs. surface-based), the mapping is, in general, a bit ill-defined. But given that we have lots of information on the 8 NSD subjects, you could use nsd_mapdata (volume-to-nativesurface option) to go from MNI to the native subject surfaces (e.g. [lh,rh].layerB2), and then use nsd_mapdata (nativesurface-to-fsaverage option) to go from the native subject surface space (e.g. [lh,rh].white) to fsaverage. You could repeat this process for each of the NSD subjects and then you could average the results in fsaverage space. For additional ideas and background, see here.

- "How do I map a specific MNI coordinate using nsd_mapdata?"

  The easiest approach would be to copy the MNI 1mm NIFTI template (MNI152_T1_1mm.nii.gz), modify the image data inside the template to specifically label the MNI coordinate that is desired (ITK-SNAP reports the MNI coordinate as "World units (NIFTI)") (e.g., create a binary volume with a "1" at the location of interest), load the image data, and then use nsd_mapdata to map the image data to some other space. Note that the motivation for building off of the MNI template is to ensure that the headers and the RPI ordering is all preserved and handled correctly.

- "I want to use some of the FreeSurfer outputs, but I am having trouble getting the outputs to work well with nsd_mapdata."

  There are tricky issues in terms of how the volume data stored in, e.g., the .mgz files are oriented (e.g., NIFTI header issues). The best bet is to see how we handled this in the code:
  https://github.com/cvnlab/nsddatapaper/blob/main/main/analysis_transforms.m

- "How can I access the gradient nonlinearity information?"

  The pre-processed files provided with NSD involved correcting for gradient nonlinearities (these are fairly negligible for the 3T data, but are somewhat substantial for the 7T data). We cannot publicly supply the gradient coefficient files. The following information was taken from the Human Connectome Project, and it is assumed to apply equally to NSD: "The gradient field coefficients are considered proprietary and need to be obtained from your institution's Siemens collaboration manager. Your institution must have a research agreement or be willing to sign a non-disclosure agreement with Siemens. Contact Yulin Chang (✉ yulin.chang@siemens-healthineers.com) (USA) or Martin Stoltnow (✉ martin.stoltnow@siemens-healthineers.com) (rest of world)."

Common pitfalls and things to watch out for

- Please note that the noise ceiling metrics provided with NSD assume that voxel-wise beta weights are z-scored within each session and then aggregated across sessions. Analyses that wish to use the noise ceiling metrics must mimic these operations. It is possible to apply the general theory of noise ceiling to the case where z-scoring is not performed, but this is up to the user and is not currently provided with NSD.

- Some data files and/or results can involve NaN values (for example, NaN might indicate when data are missing), and this may cause problems with various software tools. When processing files, it is recommended to check for these cases and resolve them appropriately (e.g., possibly setting NaNs to 0).

- In the nsd_mapdata utility (and associated transform files), the 'native surface' to 'fsaverage' transform is accomplished using the arbitrary naming convention of 'lh.white' and 'rh.white', even though the concept of the fsaverage transform is not actually specific to any cortical depth (i.e. it would be equally applicable to the mid-gray or pial surfaces). Thus, do not let the naming convention cause any confusion. For example, using nsd_mapdata, it would be reasonable to map data from the 'func1pt0' space to 'lh.layerB2' (mid-gray surface), and then map from 'lh.white' (which is just the arbitrary naming convention for data on the subject's surface) to 'fsaverage'.

- Note that no intensity normalization, detrending, or noise removal has been applied to the pre-processed fMRI time-series data that are provided with NSD. In particular, note that the mean signal intensity present at a given voxel may drift to some degree over the course of a run, and might be somewhat variable across runs and scan sessions. One should keep these observations in mind when designing an approach that starts with the time-series data.

- Some of the files that contain betas in percent signal change units are actually multiplied by 300 and stored as integer format (to save space), and thus need to be casted to decimal format and divided by 300 upon loading. Be careful.

- The category-selective ROIs that are provided are intentionally defined with a liberal threshold (*t* > 0). This has the consequence that some ROIs may overlap with other ROIs (e.g., the floc-faces ROI collection may label some of the same voxels/vertices as the floc-bodies ROI collection). If you require more stringent ROIs, you can further whittle them down based on the provided t-values and/or winner-take-all operations, etc.

- In the floc experiment, the 'body' category is **distinct** from the 'bodies' domain. The latter pools over body the 'body' and 'limb' categories.

- The MNI formatting conventions (especially regarding left vs. right) are tricky. Please see Technical notes for details.

- If you plan to try to use the transform files provided with NSD, keep in mind that these files have very specific meanings and conventions, so do not necessarily assume that they will work "out of the box" with some specific software. If possible, we recommend using nsd_mapdata.

- Note that the number of volumes in the pre-processed time-series data may be slightly larger than expected. This is correct behavior (some "excess" volumes are at the end) and has to do with how the pre-processing is performed. For example, the duration of the experiment conducted in each prf run is 300 s (or more precisely, 300 x 0.999878 s = 299.9634 s). The TR is 1.6 s. We acquired 188 volumes for a given prf run. Notice that 188 x 1.6 = 300.8 s, which therefore extends a little beyond the end of the actual experiment duration. In pre-processing for the standard resolution version, we resample to a rate of 4/3 s (or more precisely, 4/3 * 0.999878 = 1.33317 s). To ensure that we accommodate the full duration of the acquired data, the pre-processing is designed to produce 300.8/(1.33317) = 225.63 volumes. But of course, fractional volumes are non-sensical; hence, what we actually do is to round up to produce 226 volumes. (Note that there is a little bit of extrapolation involved to compute the very final volume.) Thus, 226 volumes is obtained in the pre-processing, even though there is a sense in which 225 volumes should have been obtained (since 300 s / (4/3 s) = 225). Nonetheless, everything is correct, and you can simply strip the 226th volume from the end of the pre-processed data, and you can interpret the first 225 volumes as coinciding with, say, the prf stimulus design information that we have provided.

- The flattened surfaces provided with NSD may have a rotation that may be unexpected in certain software packages. If that is the case, you may wish to read in the flattened surfaces (e.g. read_patch.m) and apply your own rotation to the vertices and then re-save the files.

- NSD provides a manually flattened version of the fsaverage surface (?h.full.flat.patch.3d) that is distinct from the one that comes with FreeSurfer (?h.cortex.patch.flat). The former is a bit less jaggedy than the latter. Also, the same general cutting strategy used for the manually flattened fsaverage was used to generate the manually flattened version of each native-subject surface.