DeepRF: Ultrafast population receptive field mapping with deep learning

Jordy Thielen[*][†]
Radboud University
Donders Centre for Cognition
jordy.thielen@donders.ru.nl

Umut Güçlü[†]
Radboud University
Donders Centre for Cognition
u.guclu@donders.ru.nl

Yagmur Güçlütürk
Radboud University
Donders Centre for Cognition
y.gucluturk@donders.ru.nl

Luca Ambrogioni
Radboud University
Donders Centre for Cognition
l.ambrogioni@donders.ru.nl

Sander E. Bosch
Radboud University
Donders Centre for Cognition
s.bosch@donders.ru.nl

Marcel A. J. van Gerven
Radboud University
Donders Centre for Cognition
m.vangerven@donders.ru.nl

Abstract

Population receptive field (pRF) mapping is an important asset for cognitive neuroscience. The pRF model is used for estimating retinotopy, defining functional localizers and to study a vast amount of cognitive tasks. In a classic pRF, the cartesian location and receptive field size are modeled as a 2D Gaussian kernel in visual space and are estimated by optimizing the fit between observed responses and predicted responses. In the standard framework this is achieved using an iterative gradient descent algorithm. This optimization is time consuming because the number of pRFs to fit (e.g., fMRI voxels) is typically large. This computation time increases further with the complexity of the pRF model (e.g., adding HRF parameters, surround suppression and uncertainty measures). Here, we introduce DeepRF, which uses deep convolutional neural networks to estimate pRFs. We compare the performance of DeepRF with that of the conventional method using a synthetic dataset for which the ground truth is known and an empirical dataset. We show that DeepRF achieves state-of-the-art performance while being more than 3 orders of magnitude faster than the conventional method. This enables easier and faster modeling of more complex pRF models, resolving an important limitation of the conventional approach.

1   Introduction

In visual neuroscience the investigation of receptive field properties has been one of the major advancements in understanding the visual system. Receptive field properties (i.e., location and size of the receptive field) provide a way to construct the preferred positions in the visual field of each recording site, from which visual field maps can be constructed. Furthermore, by creating a model for the receptive field one can predict neural responses to different stimuli. Recording techniques that are used in investigating human receptive field properties often take the responses of many neurons.

---

[*] Corresponding author.
[†] The authors contributed equally.

In creating receptive field models at this scale, one refers to the model as a population receptive field (pRF) model as it measures the pooled response from several million neurons [29].

In humans, functional magnetic resonance imaging (FMRI) provides an excellent instrument to measure brain activity as a response to visual stimuli and is often used in visual field mapping. The conventional pRF mapping approach proceeds by specifying a parameterized model (and estimating the parameters per voxel using gradient descent [30]. Often, this receptive field model is taken to be a symmetric two-dimensional Gaussian [31]. A wide variety of extensions to this basic approach has been developed such as incorporating surround suppression with a difference of Gaussians (DoG) model [36], a method without any explicit assumption on the shape of the receptive field [18], compressive spatial summation [13], Bayesian extensions of the conventional method [21, 35] and connective field modeling [7].

The pRF approach has been used to map visual cortex using retinotopy, specifically to draw regions of interest (ROIs) like dorsal and ventral V1, V2 and V3 [25], to map color throughout visual cortex [32], to map numerosity in parietal areas [10, 9], to map auditory properties throughout temporal cortex [27] and to investigate a wide range of phenomena like scotomas [6], attention [28, 16], shape perception [17], autism spectrum disorder [23], Alzheimer's disease [2] and blindness [19].

A major bottleneck with conventional pRF mapping is the time required to compute a pRF map, which easily takes multiple hours per subject. This not only wastes computational resources and valuable time, but also precludes the estimation of more complex receptive field models that have more free parameters.

We aim to improve the methods to fit pRF models to allow faster whole-brain model fitting. We show that our proposed method is indeed faster than the baseline method while remaining on par in terms of performance. The proposed method enables faster model fitting as well as experimentation with and comparison between more complex models.

2 Methods

2.1 Forward model

We use a forward model that characterizes BOLD activity over time of an arbitrary MRI voxel. The forward model generates a voxel time series $y(t)$ at time t as the sum of the signal of interest $p(t)$ and the additive noise $\xi(t)$:

$$y(t) = p(t) + \xi(t) .$$

The forward model was implemented in Python (version 3.7.3).

2.1.1 Signal component

The signal component is defined by two sources: the convolution of the heamodynamic response function (HRF) $h(t)$ with the population response $r(t)$:

$$p = (h \star r)(t) .$$

We use the double gamma HRF defined as:

$$h(t) = \left(\frac{t^{\alpha_1} \beta_1^{\alpha_1} e^{-\beta_1 t}}{\Gamma(\alpha_1)} - c \frac{t^{\alpha_2} \beta_2^{\alpha_2} e^{-\beta_2 t}}{\Gamma(\alpha_2)} \right) ,$$

with $\alpha_1 = 5 + \delta$, $\alpha_2 = 15 + \delta$, $\beta_1 = \beta_2 = 1$ and $c = 0.1$. Here, α_1 and α_2 represent the time of the peak and undershoot of the HRF with a delay δ. The population response $r(t)$ is the response between the stimulus $s(x, y, t)$ and the static population receptive field $g(x, y)$:

$$r(t) = \sum_{x,y} s(x, y, t) g(x, y) .$$

The stimulus function $s(x, y, t)$ is a binary indicator function and denotes the presence of the stimulus at location (x, y) at time t. The population receptive field is modelled as a 2D isometric Gaussian:

$$g(x, y) = \exp \left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2} \right) .$$

The goal is to estimate the receptive field location (x_0, y_0), receptive field size σ and HRF delay δ.

2

2.1.2 Noise component

The noise component $\xi(t)$ is based on other simulation packages (e.g., NeuRosim [33] or SimTB [5]). Hyperparameter settings are taken from [34]. For an extensive review on MRI noise sources see [20]. The noise component $\xi(t)$ contains five different sources: low frequency noise, physiological noise, system noise, task-related noise, and temporal noise, with an amplitude of ν.

$$\xi(t) = \nu(\xi_1(t) + \xi_2(t) + \xi_3(t) + \xi_4(t) + \xi_5(t)) \ .$$

Low frequency noise models the drifts in the signal that are caused by slow fluctuations in the scanner hardware. This noise source is modeled on the basis of discrete cosine waves θ from the TR up to 128 seconds:

$$\xi_1(t) = \sum_\theta \cos(2\pi\theta t) \ .$$

The physiological noise comprises cardiac and respiratory artefacts, which are typically at 1.17 Hz and 0.2 Hz respectively:

$$\xi_2(t) = \cos(2\pi 1.17t) + \sin(2\pi 0.2t) \ .$$

The system noise models the measurement noise, which is typically modeled as white noise:

$$\xi_3(t) \sim \mathcal{N}(0,1) \ .$$

The task-related noise adds noise relative to neural activity evoked by the task as well as residual noise from head motion. It is also modeled as white noise, but only at those time points where the task is 'active':

$$z(t) \sim \mathcal{N}(0,1) \ ,$$
$$\xi_4(t) = z(t)f(t) \ ,$$

where $f(t)$ is an indicator function that is 1 when the task is active at time t and 0 otherwise. Finally, the temporal noise contains the auto correlation of the signal as we are dealing with repeated measures. This is modeled as a first-order autoregressive process:

$$\xi_5(t) \sim \mathcal{N}(0.2\xi_5(t-1),1) \ .$$

2.2 Conventional approach

The baseline method is the coarse-to-fine method originally proposed by [4]. Under this framework, the observed response $y(t)$ at time t is modeled as:

$$y(t) = \beta p(t) + e \ ,$$

where $p(t)$ is the predicted response, β is a scaling factor to account for the unknown units of the observed response and e is the measurement noise. The predicted response is the convolution of the heamodynamic response function (HRF) $h(t)$ and the modeled population response $r(t)$:

$$p(t) = (h \star r)(t) \ ,$$

where $h(t)$ is modeled as the double-gamma response function as defined above with delay δ. The modeled population response $r(t)$ is the combination of the static receptive field properties of the population $g(x,y)$ and the stimulus $s(x,y,t)$ at time t:

$$r(t) = \sum_{x,y} s(x,y,t)g(x,y) \ .$$

The $s(x,y,t)$ is an indicator function that is 1 when there is a stimulus at coordinates (x,y) at time t and 0 otherwise. Here, $g(x,y)$ is modeled as a 2D isotropic Gaussian:

$$g(x,y) = \exp\left(-\frac{(x-x_0)^2 + (y-y_0)^2}{2\sigma^2}\right) \ ,$$

where x_0, y_0 and σ are the parameters of the pRF model describing its receptive field location (x_0, y_0) and receptive field size σ. Then, the quality of the predicted response can be measured as the sum of squared errors:

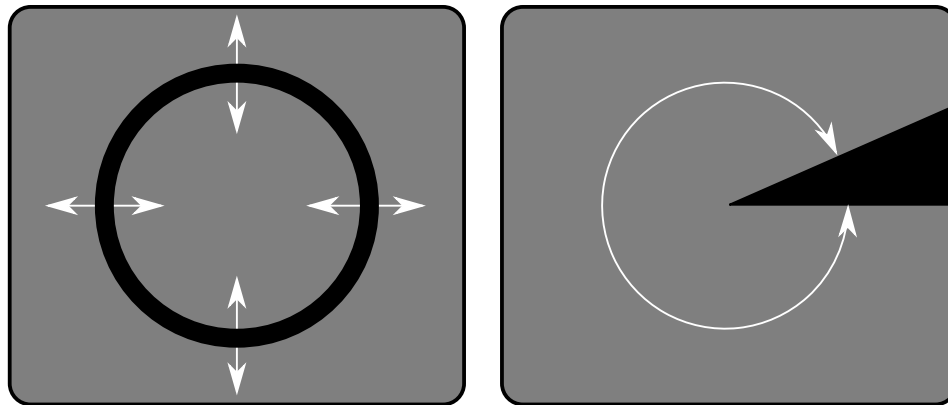$$L = \sum_t (y(t) - \beta p(t))^2 \ .$$

3

Figure 1: Stimuli were expanding and contracting rings (left panel) and clockwise and counter-clockwise rotating wedges (right panel). Both stimuli in both directions traveled in 16 uniform steps over the entire screen. Image adapted from [24].

The optimal parameters x_0, y_0, σ and δ are estimated by a coarse-to-fine search that minimizes L. In the coarse stage, a grid search is performed with several combinations of plausible parameters. The combination of parameters with the smallest L in the coarse fit is used as the initialization for the fine stage. In the fine stage, the optimization fine-tunes the parameters further using a gradient descent approach. In the current study we used the Popeye library, which is a Python implementation of the baseline method [3]. Note, that Popeye also fits β, which should be 1 as we normalize the observed response $y(t)$ before subjecting it to any model. Note that the computationally costly coarse-to-fine approach needs to be performed for every voxel.

2.3 DeepRF approach

Our proposed method employs deep learning to optimize the parameters of the pRF, therefore conveniently called DeepRF. DeepRF makes use of a deep convolutional neural network (CNN) that receives the observed response as input and predicts the parameters of the pRF so that the fit between observed parameters and predicted parameters is optimized. Theoretically, this can be interpreted as a forward amortized inference procedure [1].

As an architecture, we use the 50-layer preactivation residual neural network (ResNet) [11]. Since we are dealing with 1D time series as inputs instead of 2D images, we used 1D convolutions instead of the original 2D convolutions. The ResNet was implemented in Python (version 3.7.3) using the Chainer library (version 6.0.0) for artificial neural networks.

As an objective function, the network minimized the mean squared error (MSE) loss function between predicted pRF parameters (x_0, y_0, σ) and HRF parameter (δ) and the ground truth.

For optimization, we used Adam with hyperparameters $\alpha = 0.0001$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$ [15]. The ResNet was trained in 500 epochs, each of which presented 1024 newly drawn samples from the synthetic data generator in batches of 128 samples to the model (i.e., the model saw a total of 512.000 samples).

Note that, in contrast to the conventional approach, we only need to run gradient descent once during model training. That is, DeepRF model parameters are estimated uisng simulated fMRI timeseries generated under randomly chosen pRF parameters. Subsequently, the trained model can be used to recover pRF parameters of any voxel using a rapid forward pass. The only requirement is that the experimental design with which the simulated data were generated to train the model is the same as the design that was used to collect the empirical data.

2.4 Experimental design

The performance of the DeepRF approach was compared with that of the conventional approach using both simulated and empirical data. To this end, we used a classical pRF mapping protocol.

4

Stimuli were expanding and contracting rings and clockwise and counter-clockwise wedges displayed on a mean-luminance gray background (see Figure 1). Both rings and wedges were filled with a radial colored checkerboard texture and changed colors to its complementary luminance at 5 Hz.

The four stimuli were presented in four isolated runs. Each run lasted 180 s, starting with 4 s and ending with 16 s of pure fixation periods (i.e., no stimulus). In between, five stimulus cycles of 32 s were presented. The expanding and contracting ring had a width of 0.95 visual degree and were not scaled with the cortical magnification factor. The clockwise and counter-clockwise moving wedge had an angle of 22.5 degrees. All stimuli traveled across visual field in 16 equal steps of 2 s, making up cycles of 32 s. Each of five cycles repeated the same procedure for the specific stimulus.

This protocol was used to train the DeepRF model as well as to collect empirical data.

2.5   Empirical data

For empirical validation, we used the Study Forrest dataset [8]; specifically its extension to retinotopic mapping [24].

2.5.1   Participants

Retinotopic mapping data was acquired for 15 right-handed participants (age 21-39 years, 6 female). For each of the participants, normal visual function was verified by means of individual eye visual acuity and visual field sensitivity tests. In this study, we used Participant 2 only.

2.5.2   Experimental setup

Stimuli were presented on a 26.5 by 21.2 cm rear-projection screen inside the bore of the magnet using a LCD projector at a resolution of 1280 by 1024 pixels at a frame rate of 60 Hz. Participants perceived the stimuli via a mirror mounted on the head coil at a viewing distance of 63 cm. Stimuli were presented using PsychoPy version 1.79. During the runs, participants maintained fixation at a circle with a radius of 0.4 degrees of visual angle. To keep participants attentive, individual characters of 0.5 visual degree were presented at 1.5 Hz in the fixation circle. These characters were excerpts from song lyrics which participants had to read. After a run, participants were asked two-option forced choice questions about these lyrics to verify attention.

2.5.3   MRI acquisition

Anatomical scans were T1-weighted images with 274 sagittal slices (field-of-view (FoV) of 191.8 by 256 by 256 mm) with voxel size of 0.7 mm with an 384 by 384 in-plane reconstruction matrix (0.67 isotropic resolution) recorded with a 3D turbo field echo (TFE) sequence (repetition time (TR) of 2.5 seconds, inversion time (TI) of 900 ms, flip angle of 8 degrees, echo time (TE) of 5.7 ms, bandwidth of 144.4 Hz/pix, Sense reduction AP 1.2, RL 2.0) [8].

Functional scans were T2*-weighted echo-plannar images (gradient-echo, repetition time (TR) of 2 s, echo time (TE) of 30 ms, flip angle of 90 degree, bandwidth of 1943 Hz/pix, parallel acquisition with sensitivity encoding (SENSE) reduction factor 2 acquired with a 3 Tesla Philips Achieva dStream MRI scanner and a 32 channel head-coil [24]. Whole-brain volumes were recorded in 35 axial slices (thickness 3.0 mm) with 80 by 80 voxels (3.0 by 3.0 mm) of in-plane resolution, 240 mm field-of-view (FoV), anterior to posterior phase encoding direction, 10% inter-slice gap, recorded in ascending order.

2.5.4   Data preprocessing

Functional scans were motion corrected and coregistered to the anatomical scan using the SPM toolbox. Individual runs were further detrended, spectrally filtered and converted to percent of signal change by division of the pre-detrended mean. Finally, we extracted a visual ROI as defined by the Study Forrest dataset.

2.6   Analyses

Model performance was analysed as follows. First, we trained the DeepRF model using synthetic data generated under the described experimental protocol. Next, the conventional and DeepRF approaches

were evaluated using another set of synthetic data. Model performance was compared by comparing predicted parameters with ground truth parameters. Finally, parameters estimated from empirical data were compared with each other. In addition to the DeepRF and Popeye models, we also tested a combined version, where DeepRF was used as the coarse estimation to be fine-tuned by Popeye.

3 Results

3.1 Model estimation

Training of the DeepRF model was performed on 512.000 unique samples. Training was done over 500 epochs each presenting 1024 new samples in batches of 128 samples. Training was performed on a GPU and took 3.1 hours to complete. Training loss consistently decreased over epochs (Figure 2).
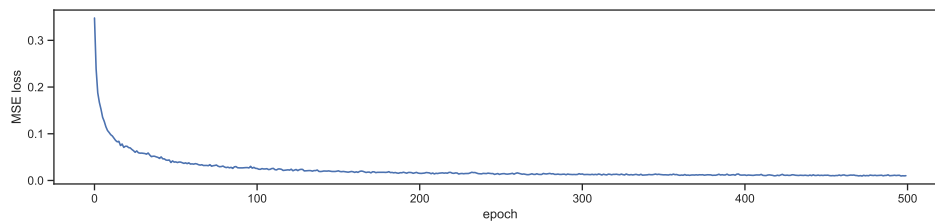

Figure 2: Mean squared error loss of DeepRF over training epochs.

We validated DeepRF and Popeye on newly generated syntetic data as well as empirical data. The results are outlined below. In addition to the two frameworks, we ran DeepRF as the coarse estimation to be fine-tuned by Popeye. In terms of computational speed, DeepRF was 1868 times faster than Popeye on the synthetic dataset and even 6178 times faster on the empirical dataset. Using DeepRF as initialization for Popeye resulted in a factor of two faster estimations (Figure 3). Note, both DeepRF as well as Popeye were ran on a CPU.

3.2 Validation on simulated data

The parameters as predicted by DeepRF, Popeye and their combination are very close to the target values (see Figure 4). The Pearson correlations of predicted parameters with target values for DeepRF were 0.985, 0.987, 0.991 and 0.986 for δ, σ, x_0 and y_0 respectively. The same correlations for Popeye were 0.991, 0.988, 0.991 and 0.982 and for the combination of DeepRF and Popeye they were 0.953, 0.975, 0.971 and 0.960. From these values we can observe that all models captured the target values almost perfectly.

Given the parameters, one can generate the modeled responses (see Figure 5). We can then compute the explained variance as the squared Pearson correlation between the predicted responses and the observed responses (see Figure 6). The median explained variance was 0.79 for DeepRF, 0.80 for Popeye and 0.80 for their combination. As a comparison, we also computed the median explained variance of the target values, which was 0.80, which denotes the upper bound on the explained variance.
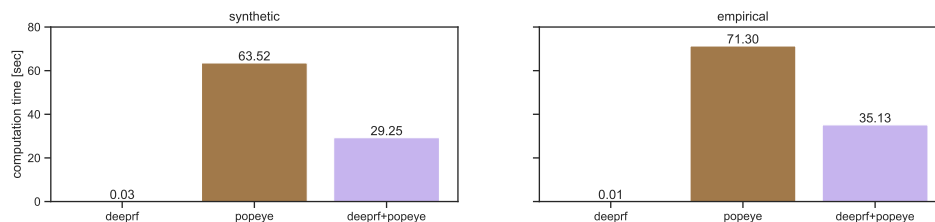

Figure 3: The average computation time in seconds needed per voxel for DeepRF, Popeye and using DeepRF as the coarse fit for Popeye. Note, all models were ran on a CPU.
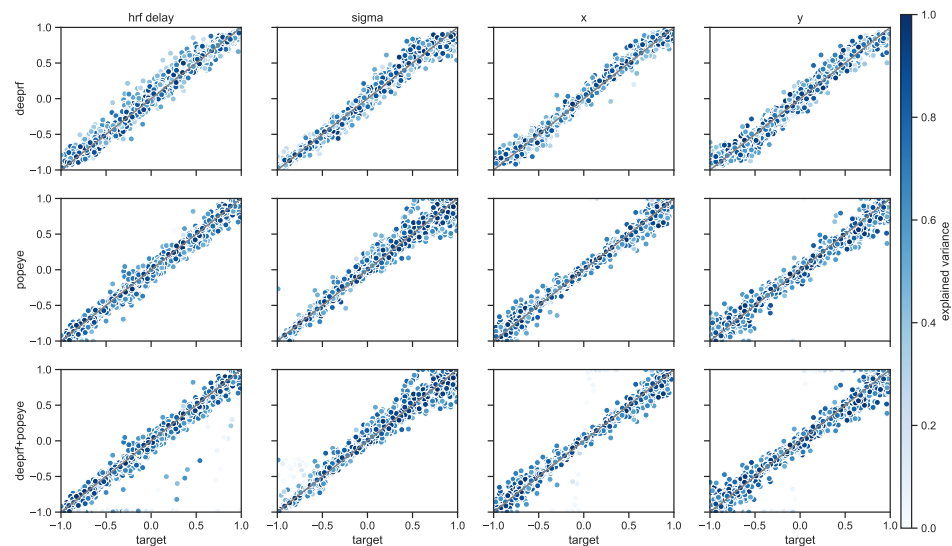
6

Figure 4: Predicted parameters versus target parameters of the synthetic dataset. The top row shows the predicted parameters of DeepRF versus target, the middle row shows the predicted parameters of Popeye versus target and the last row shows the predicted parameters of DeepRF combined with Popeye versus target. The gray lines indicate the values as they should be given a perfect model. The datapoints represent individual voxels colored by their explained variance.
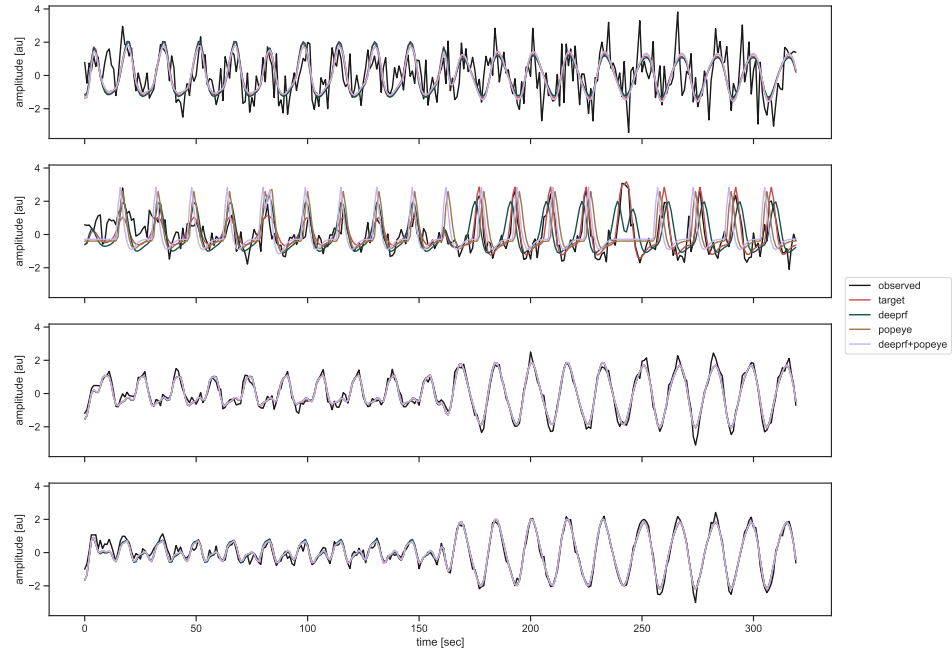

Figure 5: Observed and predicted responses of the synthetic dataset. Four examples of responses are shown where the top two have low explained variance and the bottom two have high explained variance. The black line shows the observed responses which act as input to the prediction models. The target response is the underlying modeled response of the observed data (i.e., the observed data without noise).
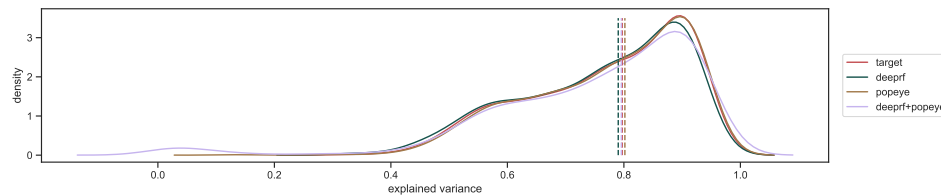
7

Figure 6: Histograms of explained variances of the synthetic dataset. Solid lines show the distribution of explained variances for the target (i.e., showing the upper limit), DeepRF, Popeye and the combination of DeepRF and Popeye. The dashed vertical lines mark the median values of these distibutions.

To measure the overlap between DeepRF and Popeye we measured the error of both models with the target values as the absolute difference between the predicted parameters and the target parameters, as well as for the difference in explained variance. These were subjected to a paired t-test to test for differences in the error distributions. We found significant effects for the δ, x_0 parameters and explained variances ($p < 0.001$), where the the error was always on average larger for DeepRF than for Popeye. The differences were 0.02, 0.00, 0.01 and 0.01 for δ, σ, x_0 and y_0 respectively and 0.01 for the explained variance. Note that these differences are very small, specifically only 1% of explained variance, which is also reflected in the almost equal response predictions as shown in Figure 5.

3.3 Empirical data

As an empirical validation, we applied the models to the Study Forrest dataset. For this dataset, of course, we do not know the ground truth labels. Instead, we take the conventional model, Popeye, as the target model. The parameters as predicted by DeepRF and the combination of Deeprf and Popeye, are very close to the target values of Popeye (see Figure 7). The Pearson correlations of predicted parameters with target values for DeepRF were 0.889, 0.721, 0.975 and 0.985 for δ, σ, x_0 and y_0 respectively. The same correlations for the combination of DeepRF and Popeye were 0.779, 0.916, 0.972 and 0.988. From these values we can observe that there is a close correspondence between DeepRF and Popeye on empirical data.
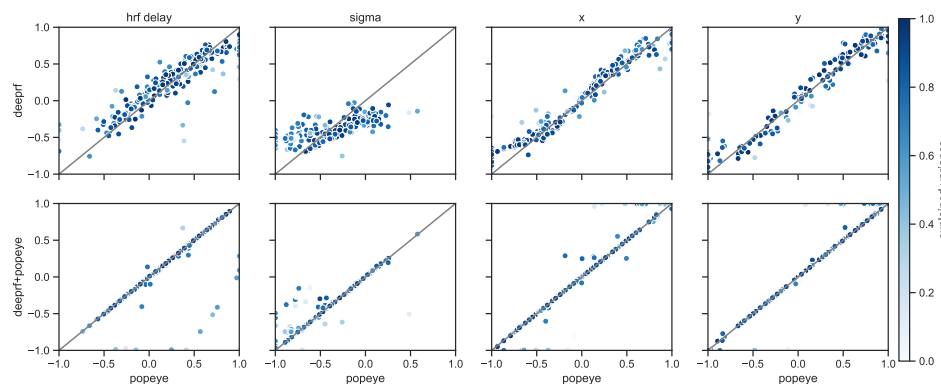

Figure 7: Predicted parameters of DeepRF and DeepRF combined with Popeye versus Popeye parameters on the empirical dataset. The top row shows the predicted parameters of DeepRF, while the bottom row shows the predicted parameters of DeepRF combined with Popeye. The gray lines indicate the values when a model predicts exactly like Popeye. The datapoints represent individual voxels colored by their explained variance.

Given the parameters, one can generate the modeled responses (see Figure 8). We can then compute the explained variance as the squared Pearson correlation between the predicted responses and the observed responses (see Figure 9). The median explained variance was 0.72 for DeepRF, 0.77 for Popeye and 0.74 for their combination.
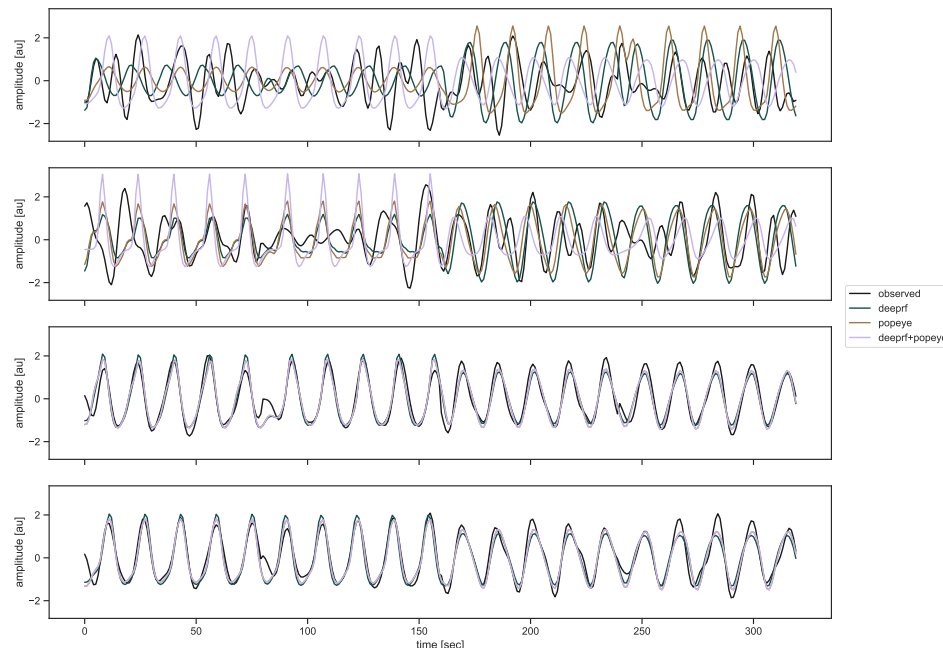
8

Figure 8: Observed and predicted responses on the empirical dataset. Four examples of responses are shown where the top two have low explained variance and the bottom two have high explained variance. The black line shows the observed responses which act as input to the DeepRF, Popeye and combined models.
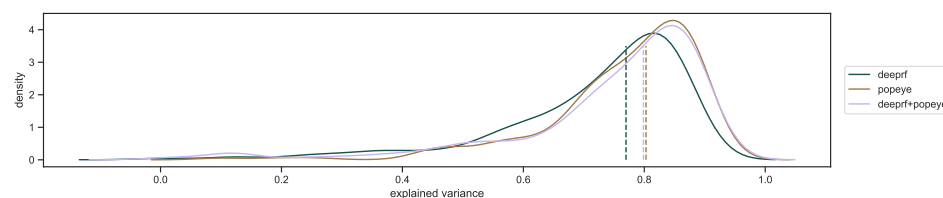

Figure 9: Histograms of explained variances on the empirical dataset. Solid lines show the distribution of explained variances for DeepRF, Popeye and the combination. The dashed vertical lines mark the median values of these distributions.

4 Discussion

We introduced DeepRF as a new approach to pRF mapping. Compared to conventional pRF mapping, DeepRF is more than three orders of magnitude faster with only 1% performance loss on average on simulated data and about 5% reduced performance on average on empirical data.

In practice, this entails that we can obtain whole-brain receptive field maps in seconds rather than hours. Note also, that despite the reduced overlap between the models' δ and σ parameters, the x_0 and y_0 parameters are very well captured. Hence, DeepRF allows retinotopy, the delineation of visual ROIs (e.g., V1, V2, V3) to be done within seconds, instead of hours. Additionally, our results also imply that pRF models with many more free parameters that could previously not be fitted in a realistic time frame come within reach.

We not only massively improve time efficiency but also greaty reduce the number of computational cycles needed to estimate pRF models, thereby reducing the carbon footprint of these resource-hungry analysis approaches [22, 26]. Note further that the underlying theoretical motivation of our work [1] is fully general. This suggests that a wide range of other demanding analysis techniques within and beyond neuroscience can likewise be improved both in terms of speed and energy demands.

9

We have to note here, that we performed model comparison on the level of explained variances. One has to be cautious here, as a model that captures noise will gain higher explained variance. Popeye has the power to fit any response well by optimizing its explained variance directly, whereas DeepRF might fail doing so if the signal to noise ratio drops largely or is different between training and testing. In Figure 5 and Figure 8 we show two voxels with high explained variance and two with low explained variance for all models. Note, that the ones with low explained variance typically have higher noise levels.

DeepRF can further be extended to predict posterior distributions over parameters, affording a Bayesian approach to pRF mapping [21]. For example, this can be achieved by adding dropout [12] to the model and using test-time dropout to predict a distribution over parameter values [14]. This may then also capture the noise-related fits discussed above as reduced certainty.

DeepRF might be further improved by allowing the model fitting to be done on predicted responses, as is done in the conventional approach. Currently, DeepRF minimizes the mean squared error between the predicted and target parameters. The loss function might be more informed if with the predicted parameters the error is computed at the level of differences between predicted and observed responses, directly optimizing the explained variance.

Concluding, we have shown that receptive field mapping using DeepRF can be achieved at a speed that is more than three orders of magnitude above the conventional pRF mapping approach. This opens up new possibilities for furthering our understanding of the functional properties of the human brain.

Acknowledgements

References

[1] Luca Ambrogioni, Umut Güçlü, Julia Berezutskaya, Eva W. P. van den Borne, Yağmur Güçlütürk, Max Hinne, Eric Maris, and Marcel A. J. van Gerven. Forward amortized inference for likelihood-free variational marginalization. In *AIStats*, pages 1–11, 2019.

[2] Alyssa A Brewer and Brian Barton. Visual cortex in aging and Alzheimer's disease: changes in visual field maps and population receptive fields. *Frontiers in Psychology*, 5:74, 2014.

[3] Kevin DeSimone, Ariel Rokem, and Keith Schneider. popeye: a population receptive field estimation tool. *The Journal of Open Source Software*, 1, 2016.

[4] Serge O Dumoulin and Brian A Wandell. Population receptive field estimates in human visual cortex. *NeuroImage*, 39(2):647–660, 2008.

[5] Erik B Erhardt, Elena A Allen, Yonghua Wei, Tom Eichele, and Vince D Calhoun. SimTB, a simulation toolbox for fMRI data under a model of spatiotemporal separability. *NeuroImage*, 59(4):4160–4167, 2012.

[6] Koen V Haak, Frans W Cornelissen, and Antony B Morland. Population receptive field dynamics in human visual cortex. *PLoS One*, 7(5):e37686, 2012.

[7] Koen V Haak, Jonathan Winawer, Ben M Harvey, Remco Renken, Serge O Dumoulin, Brian A Wandell, and Frans W Cornelissen. Connective field modeling. *NeuroImage*, 66:376–384, 2013.

[8] Michael Hanke, Florian J Baumgartner, Pierre Ibe, Falko R Kaule, Stefan Pollmann, Oliver Speck, Wolf Zinke, and Jörg Stadler. A high-resolution 7-Tesla fMRI dataset from complex natural stimulation with an audio movie. *Scientific Data*, 1:140003, 2014.

[9] Ben M Harvey and Serge O Dumoulin. A network of topographic numerosity maps in human association cortex. *Nature Human Behaviour*, 1(2):0036, 2017.

[10] Ben M Harvey, Barrie P Klein, Natalia Petridou, and Serge O Dumoulin. Topographic representation of numerosity in the human parietal cortex. *Science*, 341(6150):1123–1126, 2013.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[12] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[13] Kendrick Kay, Jonathan Winawer, Aviv Mezer, and Brian A Wandell. Compressive spatial summation in human visual cortex. *American Journal of Physiology-Heart and Circulatory Physiology*, 2013.

[14] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] Barrie P Klein, Ben M Harvey, and Serge O Dumoulin. Attraction of position preference by spatial attention throughout human visual cortex. *Neuron*, 84(1):227–237, 2014.

[17] Peter Kok and Floris P de Lange. Shape perception simultaneously up-and downregulates neural activity in the primary visual cortex. *Current Biology*, 24(13):1531–1535, 2014.

[18] Sangkyun Lee, Amalia Papanikolaou, Nikos K Logothetis, Stelios M Smirnakis, and Georgios A Keliris. A new method for estimating population receptive field topography in visual cortex. *NeuroImage*, 81:144–157, 2013.

[19] Netta Levin, Serge O Dumoulin, Jonathan Winawer, Robert F Dougherty, and Brian A Wandell. Cortical maps and white matter tracts following long period of visual deprivation and retinal image restoration. *Neuron*, 65(1):21–31, 2010.

[20] Thomas T Liu. Noise contributions to the fMRI signal: an overview. *NeuroImage*, 143:141–151, 2016.

[21] Silvan C Quax, Thomas C Van Koppen, Pasi Jylänki, Serge O Dumoulin, and Marcel Aj Van Gerven. Slice-sampled Bayesian PRF mapping. *BioRxiv*, page 093724, 2016.

[22] Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. Green AI. *arXiv preprint arXiv:1907.10597*, 2019.

[23] D Samuel Schwarzkopf, Elaine J Anderson, Benjamin de Haas, Sarah J White, and Geraint Rees. Larger extrastriate population receptive fields in autism spectrum disorders. *Journal of Neuroscience*, 34(7):2713–2724, 2014.

[24] Ayan Sengupta, Falko R Kaule, J Swaroop Guntupalli, Michael B Hoffmann, Christian Häusler, Jörg Stadler, and Michael Hanke. A studyforrest extension, retinotopic mapping and localization of higher visual areas. *Scientific Data*, 3:160093, 2016.

[25] Martin I Sereno, AM Dale, JB Reppas, KK Kwong, JW Belliveau, TJ Brady, BR Rosen, and RB Tootell. Borders of multiple visual areas in humans revealed by functional magnetic resonance imaging. *Science*, 268(5212):889–893, 1995.

[26] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and Policy Considerations for Deep Learning in NLP. *arXiv preprint arXiv:1906.02243*, 2019.

[27] Jessica M Thomas, Elizabeth Huber, G Christopher Stecker, Geoffrey M Boynton, Melissa Saenz, and Ione Fine. Population receptive field estimates of human auditory cortex. *NeuroImage*, 105:428–439, 2015.

[28] Ashika Verghese, Scott C Kolbe, Andrew J Anderson, Gary F Egan, and Trichur R Vidyasagar. Functional size of human visual area V1: a neural correlate of top–down attention. *NeuroImage*, 93:47–52, 2014.

[29] Jonathan D Victor, Keith Purpura, Ephraim Katz, and Buqing Mao. Population encoding of spatial frequency, orientation, and color in macaque V1. *Journal of Neurophysiology*, 72(5):2151–2166, 1994.

[30] Brian A Wandell and Jonathan Winawer. Imaging retinotopic maps in the human brain. *Vision Research*, 51(7):718–737, 2011.

[31] Brian A Wandell and Jonathan Winawer. Computational neuroimaging and population receptive fields. *Trends in Cognitive Sciences*, 19(6):349–357, 2015.

[32] Lauren E Welbourne, Antony B Morland, and Alex R Wade. Population receptive field (pRF) measurements of chromatic responses in human visual cortex using fMRI. *NeuroImage*, 167:84–94, 2018.

[33] Marijke Welvaert, Joke Durnez, Beatrijs Moerkerke, Geert Verdoolaege, and Yves Rosseel. neuRosim: An R package for generating fMRI data. *Journal of Statistical Software*, 44(10):1–18, 2011.

[34] Marijke Welvaert and Yves Rosseel. A review of fMRI simulation studies. *PLoS One*, 9(7):e101953, 2014.

[35] Peter Zeidman, Edward Harry Silson, Dietrich Samuel Schwarzkopf, Chris Ian Baker, and Will Penny. Bayesian population receptive field modelling. *NeuroImage*, 180:173–187, 2018.

[36] Wietske Zuiderbaan, Ben M Harvey, and Serge O Dumoulin. Modeling center–surround configurations in population receptive fields using fMRI. *Journal of Vision*, 12(3):10–10, 2012.