



Real-time and Recursive Estimators for Functional MRI Quality Assessment

Nikita Davydov^{1,2,3} · Lucas Peek⁴ · Tibor Auer⁵ · Evgeny Prilepin¹ · Nicolas Gninenko^{6,7} · Dimitri Van De Ville^{6,7} · Artem Nikonorov^{2,3} · Yury Koush⁸

Accepted: 3 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Real-time quality assessment (rtQA) of functional magnetic resonance imaging (fMRI) based on blood oxygen level-dependent (BOLD) signal changes is critical for neuroimaging research and clinical applications. The losses of BOLD sensitivity because of different types of technical and physiological noise remain major sources of fMRI artifacts. Due to difficulty of subjective visual perception of image distortions during data acquisitions, a comprehensive automatic rtQA is needed. To facilitate rapid rtQA of fMRI data, we applied real-time and recursive quality assessment methods to whole-brain fMRI volumes, as well as time-series of target brain areas and resting-state networks. We estimated recursive temporal signal-to-noise ratio (rtSNR) and contrast-to-noise ratio (rtCNR), and real-time head motion parameters by a framewise rigid-body transformation (translations and rotations) using the conventional current to template volume registration. In addition, we derived real-time framewise (FD) and micro (MD) displacements based on head motion parameters and evaluated the temporal derivative of root mean squared variance over voxels (DVARs). For monitoring time-series of target regions and networks, we estimated the number of spikes and amount of filtered noise by means of a modified Kalman filter. Finally, we applied the incremental general linear modeling (GLM) to evaluate real-time contributions of nuisance regressors (linear trend and head motion). Proposed rtQA was demonstrated in real-time fMRI neurofeedback runs without and with excessive head motion and real-time simulations of neurofeedback and resting-state fMRI data. The rtQA was implemented as an extension of the open-source OpenNFT software written in Python, MATLAB and C++ for neurofeedback, task-based, and resting-state paradigms. We also developed a general Python library to unify real-time fMRI data processing and neurofeedback applications. Flexible estimation and visualization of rtQA facilitates efficient rtQA of fMRI data and helps the robustness of fMRI acquisitions by means of substantiating decisions about the necessity of the interruption and re-start of the experiment and increasing the confidence in neural estimates.

Keywords Real-time quality assessment · Recursive · Functional MRI · Task · Rest · Neurofeedback paradigms · OpenNFT · rtspm Python library

Introduction

Real-time quality assessment (rtQA) of functional magnetic resonance imaging (fMRI) is essential to improve research in cognitive and clinical neurosciences, including

Highlights

- Recursive methods for real-time fMRI quality assessment (rtQA)
- Implementation of the rtQA as an extension in the open-source OpenNFT software
- Interactive visualization of rtQA parameters for time-series and whole-brain data
- OpenNFT supports task, rest, and neurofeedback experimental paradigms

the developments of therapeutic approaches based on neuroimaging. There are different types of noise affecting fMRI data quality, including the system noise originating from the inhomogeneity and instability of the magnetic field, background noise and, typically the strongest, physiological noise (Diedrichsen & Shadmehr, 2005; Friedman & Glover, 2006; Goto et al., 2016; Greve et al., 2011; Kasper et al., 2017; Triantafyllou et al., 2011). The fMRI quality assurance was suggested to facilitate the consistency of fMRI experiments across different sites, explore stability of MR scanners, classify data quality and detect artifacts (Friedman & Glover, 2006; Lu et al., 2019; Stöcker et al., 2005). While quality assurance typically suggests an improvement of consistency of data acquisitions and identification of noise sources,

Extended author information available on the last page of the article

quality control suggests a post-hoc evaluation of the fMRI data acquisition and processing workflows and exclusion of corrupted acquisitions (Alfaro-Almagro et al., 2018; Esteban et al., 2017). Most recent developments on quality assurance and control employ data-driven and machine learning methods to separate noise sources and evaluate the quality of (f)MRI data and data processing based on trained classifiers (Alfaro-Almagro et al., 2018; Astrakas et al., 2016; Esteban et al., 2017).

Several estimates have been proposed for MRI quality assurance and control, including signal-to-noise ratio (SNR), contrast-to-noise ratio (CNR), percent signal change (PSC), correlation and quantification of specific artifacts depending on the applied MRI contrast mechanisms (Alfaro-Almagro et al., 2018; Esteban et al., 2017; Friedman & Glover, 2006; Lu et al., 2019; Stöcker et al., 2005). SNR remains the fundamental and most widely used parameter for assessment of the fMRI data quality, various noise affecting data quality, and data processing approaches (Friedman & Glover, 2006; Koush et al., 2012; Maziero et al., 2020; Triantafyllou et al., 2011; van der Zwaag et al., 2012; Zilverstand et al., 2017). CNR is preferred if noise is inflated by fMRI activation during the physiological stimulation (Geissler et al., 2007; Koush et al., 2012) and in case of referential phantom-based quality evaluations when the contrast between the signal of interest and the noise is defined (Lu et al., 2019; Simmons et al., 1999; Stöcker et al., 2005). Alternatively, SNR could be also used in task-related studies when paradigm-related activation is removed after regression (Murphy et al., 2007; Zilverstand et al., 2017). When applied to fMRI time-series, temporal SNR (tSNR) is often implied (Murphy et al., 2007). For consistency, the term temporal CNR (tCNR) is also used here.

Head motion remains one of the major sources of artifacts in fMRI data, which is hard to eliminate completely (Bolton et al., 2020; Parkes et al., 2018; Satterthwaite et al., 2013; Scheinost et al., 2014). The sensitivity and specificity of fMRI acquisitions can be diminished due to susceptibility-induced T2* signal dropouts, head motion, and motion-by-susceptibility interaction during head motion (Fair et al., 2020; Koush et al., 2012; Wu et al., 1997). Head motion can be associated with heterogeneity of fMRI data smoothness leading to inter-subject head motion confounds (Scheinost et al., 2014) and with reduction of statistical sensitivity leading to false negative results (Kasper et al., 2017). Besides, retained head motion in the pre-processed fMRI data can predict anthropomorphic, behavioral and clinical factors (Bolton et al., 2020). Larger head motion is also associated with reduced SNR in resting-state fMRI runs (Van Dijk et al., 2012). Most commonly, the direct and indirect effects of head motion are suppressed along with those of other noise sources using data preprocessing pipelines based on multiple linear regression and/or principal/independent component analysis (Glover et al., 2000; Parkes et al., 2018). In particular,

framewise (FD) and micro (MD) displacements based on combination of head motion parameters, as well as volume censoring based on scrubbing and spike motion identification are used to improve the robustness of fMRI activity and connectivity estimates (Parkes et al., 2018; Power et al., 2012, 2014; Van Dijk et al., 2012).

Real-time fMRI data processing is used to reliably correct fMRI data contaminations in neurofeedback studies (Heunis et al., 2020; Koush et al., 2017a). Recursive multiple linear regression, i.e., incremental general linear model (iGLM), is proposed to perform fMRI data regression in real-time to evaluate whole-brain activation maps and to reduce physiological noise (Bagarinao et al., 2003; Nakai et al., 2006). Incremental GLM for processing whole-brain data and its less methodologically demanding cumulative GLM for processing time-series are implemented in OpenNFT (Koush et al., 2017a). Of note, iGLM also outperforms exponential moving average and sliding-window algorithms for linear detrending (Kopel et al., 2019), which highlights the potential of iGLM solutions for rtQA.

The fMRI data quality is typically evaluated after all experimental data is acquired and data-processing is performed, which imposes the risk of losing data for entire participants if it is of insufficient quality or substantially reduced after artifact removal (Dosenbach et al., 2017). Manual identification of artifacts (e.g. image distortions and spatial inconsistencies) is tedious and often not consistent; therefore, automatic comprehensive quality assessment methods are necessary to efficiently maintain an awareness of acquired fMRI (Alfaro-Almagro et al., 2018; Astrakas et al., 2016) and real-time fMRI (Dosenbach et al., 2017; Heunis et al., 2020; Ros et al., 2020; Weiskopf et al., 2007) data quality. For real-time fMRI, currently available quality assurance/control tools include, e.g., the Functional Real-time Interactive Endogenous Neuro modulation and Decoding (FRIEND) software (Basilio et al., 2015; Sato et al., 2013), Framewise Integrated Real-time MRI Monitoring software (FIRMM) (Dosenbach et al., 2017; Fair et al., 2020), Turbo-Brain Voyager (TBV) and Pyneal (MacInnes et al., 2020) software, and real-time Analysis of Functional NeuroImages (AFNI) plugin (Cox & Jesmanowicz, 1999).

To facilitate rapid analyses of key fMRI quality parameters, we applied real-time and recursive methods of rtQA. The real-time aspect refers to handling all processing steps before the next acquisition (i.e., fMRI volume) becomes available. Recursive methods update an estimate of interest using the current acquisition only, without revisiting the previous measurements. Therefore, the computational load is fixed per acquisition and does not increase during the scanning session. Specifically, we implemented recursive mean and variance estimates (Welford, 1962) to derive recursive tSNR and tCNR for time-series of whole-brain volumes and averaged within single regions of interest (ROIs) and

resting-state networks (RSNs). We also derived FD (Power et al., 2012) and MD (Van Dijk et al., 2012) real-time estimates using the conventional current to template volume registration (Koush et al., 2017a). We estimated the amount of filtered high-frequency noise and the number of spikes in time-series of interest using a recursive method based on modified Kalman filter (Koush et al., 2012). In addition, we provided the real-time estimate of the rate of intensity change across the entire brain using temporal derivative of root mean squared variance over voxels (DVARs) (Power et al., 2012). Finally, for modelled whole-brain and time-series regressors of interest and confounds, we extended a conventional application of iGLM by calculating GLM estimates as quality parameters in real-time. The estimation and visualization of all rtQA parameters were implemented as an extension of OpenNFT – an open-source Python, MATLAB and C++ framework developed for real-time fMRI data processing and neurofeedback training (Koush et al., 2017a, b). The proposed rtQA was demonstrated in real-time fMRI neurofeedback runs of a single participant without and with excessive head motion and in a group of participants using real-time data export simulations of neurofeedback and resting-state fMRI data (Koush, Meskaldji, et al., 2017c; Krylova et al., 2021).

Methods

For neurofeedback and resting-state fMRI runs, we evaluated the similar sets of rtQA parameters: (i) whole-brain and time-series recursive tSNR and tCNR, (ii) real-time head motion translations, rotations, framewise and micro displacements, (iii) DVARs, (iv) whole-brain and time-series parameters based on iGLM estimates for nuisance regressors, and (v) time-series number of spikes and the amount of filtered high-frequency noise. Task-related variance was regressed out from tSNR using iGLM (Murphy et al., 2007), and tCNR was not evaluated for resting-state runs due to the lack of pre-specified conditions. Because the purpose of the present work was to demonstrate the feasibility of recursive rtQA estimation, we refrained from linking between quality estimates and experimental findings reported elsewhere (Koush, Meskaldji, et al., 2017c; Krylova et al., 2021), as well as between evaluated neurofeedback and resting-state rtQA estimates. Aggregated estimates are expressed as mean ± std.

Recursive Mean, Variance, tSNR and tCNR

The recursive estimation implies exploiting temporal recursion that only requires the current acquisition for updating the estimator. In contrast, the cumulative estimation implies that all data acquired up to the current time point is used

for estimation based on conventional equations. For recursive estimations of mean and variance, we used the Welford online algorithm (Welford, 1962). Recursive and cumulative estimations were applied to time-series and whole-brain data for both neurofeedback and resting-state runs,

$$\bar{x}_{1..t} = \frac{1}{t} \sum_{i=1}^t x_i$$

$$\bar{v}_{1..t} = \bar{\sigma}_{1..t}^2 = \frac{1}{t} \sum_{i=1}^t (x_i - \bar{x}_{1..t})^2$$

$$\bar{x}_t = \bar{x}_{t-1} + \frac{x_t - \bar{x}_{t-1}}{t}$$

$$\bar{\sigma}_t^2 = \bar{\sigma}_{t-1}^2 + \frac{(x_t - \bar{x}_{t-1})(x_t - \bar{x}_t) - \bar{\sigma}_{t-1}^2}{t} = \frac{t-1}{t} \bar{\sigma}_{t-1}^2 + \frac{(x_t - \bar{x}_{t-1})(x_t - \bar{x}_t)}{t}$$

$$M_{2,t} = M_{2,t-1} + (x_t - \bar{x}_{t-1}) \cdot (x_t - \bar{x}_t)$$

$$\bar{v}_t = \bar{\sigma}_t^2 = \frac{M_{2,t}}{t-1}$$

where $\bar{x}_{1..t}$ —cumulative mean, x_t —signal value, $\bar{v}_{1..t}$ —cumulative variance, $\bar{\sigma}_{1..t}$ —cumulative standard deviation, \bar{x}_t —recursive mean, $M_{2,t}$ —recursive sum of squares, \bar{v}_t —recursive variance, $\bar{\sigma}_t$ —recursive standard deviation. Recursive estimations were compared to cumulative estimators (Koush et al., 2012), using mean squared error (MSE). For neurofeedback runs, we also computed separate recursive mean and variance for the baseline and regulation blocks.

As rtQA estimates, we computed recursive tSNR (rtSNR) for neurofeedback and resting-state runs, as well as recursive tCNR (rtCNR) for neurofeedback runs using recursive temporal mean and variance estimates given condition (*cond*) and baseline (*bas*) indices (Koush et al., 2012):

$$ctSNR_t = \frac{\bar{x}_{1..t}}{\sqrt{\bar{v}_{1..t}}}, \quad ctCNR_t = \frac{\bar{x}_{1..t}(cond) - \bar{x}_{1..t}(bas)}{\sqrt{\bar{v}_{1..t}(cond) + \bar{v}_{1..t}(bas)}}$$

$$rtSNR_t = \frac{\bar{x}_t}{\sqrt{\bar{v}_t}}, \quad rtCNR_t = \frac{\bar{x}_t(cond) - \bar{x}_t(bas)}{\sqrt{\bar{v}_t(cond) + \bar{v}_t(bas)}}$$

Notably, rtCNR was estimated when baseline and regulation variances existed and/or were not equal to zero. For neurofeedback runs, rtSNR was estimated for raw time-series corrected for serial correlations using autoregressive model of the first order in real-time (Koush et al., 2017a), and the task-related activity was regressed out using iGLM.

Head motion parameters

The three translation (X, Y, Z) and rotation (pitch, roll, yaw) head motion parameters were provided using the realignment routines of SPM12 adapted for real-time applications (Koush et al., 2017a). The FD was estimated based on the six head motion parameters (Power et al., 2012), and the MD estimation was based on 3 translations (Van Dijk et al., 2012):

$$FD_t = \sum_{i=1}^3 |T_{t,i} - T_{t-1,i}| + \frac{r \cdot \pi}{180} \cdot \sum_{i=1}^3 |R_{t,i} - R_{t-1,i}|$$

$$MD_t = \left| \sqrt{\sum_{i=1}^3 T_{t,i}^2} - \sqrt{\sum_{i=1}^3 T_{t-1,i}^2} \right|$$

where $T_{t,i}$ —X, Y, Z translations, $R_{t,i}$ —pitch, roll, yaw rotations, r —radius of the sphere. Rotations for FD estimation were converted from radians to millimeters given the approximate 50 mm radius of the sphere defined as the distance from the cerebral cortex to the center of the head for a healthy adult participant (Power et al., 2012). As rtQA parameters, head motion parameters, FD and MD were calculated in real-time. In addition, we estimated recursive temporal average of FD_t and MD_t using recursive mean equations. As rtQA parameters, we also estimated thresholded FD (at 0.2 mm and 0.5 mm) and MD (at 0.1 mm) and counted the number of volumes exceeding corresponding thresholds.

DVARS

In addition, distortions in real-time fMRI data due to other sources of movements, e.g. chest movements, can be controlled using temporal DVARS (Fair et al., 2020; Power et al., 2012), which represents the change of the whole-brain signal intensity between current and previous timepoints:

$$DVARS_t = \sqrt{\langle [I_t(\bar{x}) - I_{t-1}(\bar{x})]^2 \rangle}$$

where $I_t(\bar{x})$ —volume intensity at locus x , $\langle \rangle$ —spatial average over whole-brain mask. Real-time DVARS implementation in OpenNFT is based on the whole-brain mask defined at the fixed template brain volume used for real-time realignment and reslicing of the acquired real-time fMRI volumes (Koush et al., 2017a). This ensures real-time DVARS estimation across completely sampled voxels. The whole-brain mask is defined automatically based on the least-squares histogram fitting of the non-scaled voxel intensities (thresholded at < 30 to exclude zeros and very small intensity values). For fitting, we modeled exponential (low intensity values primarily outside the brain) and Gaussian (primarily

brain intensity values) functions, as implemented in MATLAB. Voxels above the half of the fitted Gaussian peak center intensity fall well within the whole-brain mask. The intensity of the real-time realigned, resliced and smoothed volumes was scaled to the median of the voxel intensities within the mask, and resultant DVARS estimates were multiplied by 100. We counted the number of volumes exceeding the 5a.u. DVARS threshold (Power et al., 2012). Note that real-time implementation of DVARS could deviate from offline estimates due to the scaling to median intensity based on the single template volume and relatively coarse automatic definition of the whole-brain mask as compared to masking based on the segmentation of structural volumes.

Kalman Filter Denoising Parameters

In fMRI time-series, spike-like artifacts are not always related to physiological noise. Thereby, alternative non-linear recursive approaches, e.g., modified Kalman filter, can be efficiently applied in real-time to simultaneously remove high-frequency noise and identify/correct spike-like artifacts after iGLM denoising (Koush et al., 2012, 2017a; Koush, Meskaldji, et al., 2017c). The linear Kalman filter is an adaptive estimation algorithm that allows extraction of the desired signal from the input through a filtering operation. It was extended for detection of the spike-like fluctuations by thresholding a discrepancy between an a priori predicted and an a posteriori estimate (Koush et al., 2012):

$$\left| K_t \cdot (y_t - H \cdot x_t) \right| < 0.9 \cdot \tilde{\sigma}_{1..t}$$

$$y_t^{corr} = y_{t-1}$$

$$R/Q = 4, H = 1$$

where y_t —observation value, x_t —state value, K —Kalman gain factor, R/Q —update rate, R, Q —noise covariance matrices, y_t^{corr} —corrected value. The Kalman filter update rate controls for the reduction of the high-frequency noise and implicitly defines the cutoff frequency. For time-series, we used the update rate $R/Q=4$ and threshold $0.9 \cdot \tilde{\sigma}_{1..t}$ as the difference between predicted and posterior estimates.

In addition, we evaluated the amount of high-frequency noise filtered by the Kalman filter using recursive MSE (rMSE) between unfiltered and filtered time-series:

$$cMSE_t = \frac{1}{t} \sum_{i=1}^t (y_i - \bar{y}_i)^2$$

$$rMSE_t = \frac{t-1}{t} rMSE_{t-1} + \frac{(y_t - \bar{y}_t)^2}{t}$$

where y_i – input value, \bar{y}_i – filtered value. Cumulative cMSE estimation implies that all data acquired up to the current time point were used for the MSE estimation. Lower rMSE indicates lower high-frequency contamination of the signal given low-pass cutoff.

Incremental and Cumulative GLM

Linear regression remains one of the most common approaches used to estimate the weights of regressors in fMRI processing pipelines (Parkes et al., 2018). The recursive linear regression, i.e., iGLM, has been proposed to perform fMRI data regression in real-time (Bagarinao et al., 2003, 2006; Nakai et al., 2006). It is based on the orthogonalization procedure to recursively estimate the coefficients of the regressors. For whole-brain data, we extended the iGLM application by means of using weights of modelled regressors as rtQA parameters. To evaluate the magnitude of target brain (de)activation and artifacts in specific brain areas, we visualized the whole-brain activation and deactivation maps associated with the experimental design as well as activation maps associated with the nuisance head motion and linear trend regressors. We also visualized beta coefficients estimated for time-series as plots. We performed similar real-time estimations for time-series using cumulative GLM, i.e. when data acquired up to the current time point was used for the GLM estimation.

Regions of Interest and Resting-State Networks

Automatic rtQA can be applied to time-series extracted from regions of interest (ROIs). For task-related and neurofeedback paradigms, ROIs are typically known and could be used for rtQA. For resting-state paradigms, users can check time-series quality from specific resting-state networks and/or combinations of ROIs. Since real-time fMRI volumes are realigned to the fixed template volume, all supported paradigms in OpenNFT are complemented with time-series rtQA of the automatically defined whole-brain ROI (see DVARS).

During neurofeedback training runs, iGLM was used to dynamically re-define ROIs for neurofeedback estimation as the most active voxels within preselected anatomical MNI templates for bilateral amygdala and dmPFC (Koush, Meskaldji, et al., 2017c). The bilateral amygdala template was based on the Talairach Daemon atlas (Lancaster et al., 2000), and the dmPFC template was defined based on pilot data as a 14 mm radius sphere around the activation peak in dmPFC excluding voxels outside the brain (Koush, Meskaldji, et al., 2017c). For resting-state runs, we used ten resting-state network (RSN) masks created from the atlas of 90 functional ROIs (Shirer et al., 2012): anterior salience, auditory, basal ganglia, dorsal default mode (dorsal DMN), higher visual, precuneus, primary visual, sensorimotor, ventral default

mode (ventral DMN), and visuospatial RSNs. Regions of interest and RSN masks were transformed from the MNI space to the native space using individual structural volumes and DARTEL tools (Ashburner, 2007) and resliced to the first volume of the corresponding fMRI run as implemented in SPM12.

Real-time fMRI Data Processing Using OpenNFT

OpenNFT is a GUI-based multi-processing open-source software framework and library originally designed for real-time fMRI neurofeedback training (Koush et al., 2017a). The software's GUI, synchronization module and multi-processing core are implemented in Python, whilst modules for real-time data processing including the computation of the neural signal are implemented in MATLAB. OpenNFT supports a broad functionality asset for real-time fMRI studies including real-time fMRI data watchdog, conventional whole-brain and time-series data processing (e.g., realignment, reslicing, smoothing, incremental GLM, filtering, despiking), computation and presentation of neural signal. OpenNFT can provide the neural signal either as activity, functional connectivity (correlation) based on single or multiple areas, effective connectivity based on dynamic causal modelling (DCM), or classification based on pre-trained support vector machine (SVM) classifier. For non-neurofeedback task-related experiments, OpenNFT functionality can be branched off and simplified by means of disabling the feedback estimation and visualization routines.

OpenNFT Design and Implementation of rtQA Extension

OpenNFT is based on parallel architecture and contains multiple processes implemented in Python and MATLAB (Fig. 1). Parallel processes are maintained using the Python multiprocessing package and MATLAB Engine, which could be called (a)synchronously, i.e., with(out) waiting for launched procedures. Python Core Process controls all the other processes, the inter-process communication, and watches the data from the MR scanner. When the new MR export data file arrives, the Python watchdog module catches the corresponding file-system notification generated by the operation system and waits until the file is completely written, which implies the indirect real-time fMRI-based triggering on new file arrivals. The MR scanner trigger pulse could be also monitored and applied using the Python synchronization process, and potentially take over the indirect real-time triggering. After the volume data is read, processed, denoised, the feedback signal could be estimated and displayed or sent to external software via UDP/COM ports. The Python GUI subprocess maintains GUI interactions and data visualization for the experimenter. The Python

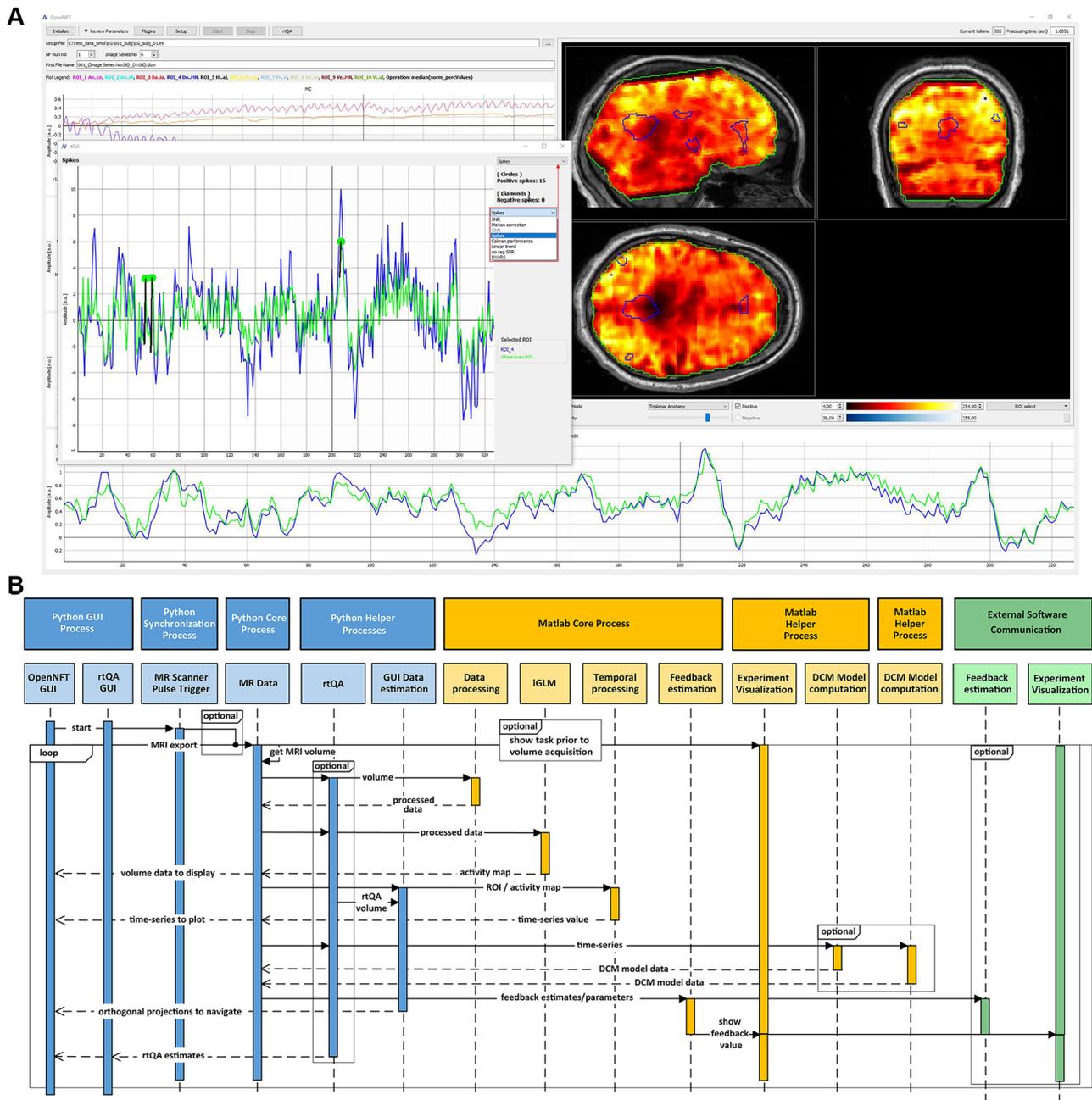


Fig. 1 The rtQA extension as part of the OpenNFT framework architecture. **A** The rtQA extension contains navigable rtQA features as illustrated using an exemplary resting-state run, spike detection menu and whole-brain rsSNR with the overlaid dorsal DMN (blue) and

whole-brain ROI (green). **B** The OpenNFT architecture and workflow. The rtQA extension is implemented using parallel Python and MATLAB subprocesses and can be completely disabled. It operates with real-time fMRI volumes and time-series of target ROIs/RSNs

Helper process is now used to facilitate the GUI whole-brain navigation. As compared to the initially Matlab Helper process for whole-brain navigation (Koush et al., 2017a), the transition to Python Helper process was accomplished with the help of our new *python-rtspm* library (see *OpenNFT Python library*) based on open-source SPM C scripts. The Matlab Core process performs fMRI data (pre)processing,

time-series denoising, and feedback signal estimation. Matlab Helper processes are reserved for experimental task and feedback visualization, as well as for heavy estimations. Further details about general OpenNFT functionality and architecture are described elsewhere (Koush et al., 2017a).

The rtQA extension supports neurofeedback, task-related, and resting-state fMRI paradigms, significantly extending

OpenNFT functionality (Fig. 1B). Implemented modes of rtQA allow flexible access to time-series and whole-brain fMRI data at different stages of (pre)processing using core Python and MATLAB processes. For instance, rtSNR and rtCNR could be estimated for raw, realigned, and smoothed volumes, as well as for raw, iGLM-filtered and despiked time-series. The rtQA extension can be used in a fully automated mode or configured for a particular paradigm. Fully automated rtQA mode implies no parameter *Settings* to facilitate rapid setup (for parameter *Settings*, see (Koush et al., 2017a)). This mode is limited to the selection of the export data folder, run number and series number to define the file names, and number of expected volumes, which is suitable for resting-state paradigms and could be used for task-related studies with certain limitations (e.g., no filtering of the regulation/task-related activity in rtSNR and no rtCNR). A semi-automatic rtQA mode is also available to support more detailed parameter *Settings*, e.g. using OpenNFT configuration files and modeling of regressors of interest in time-series and volume processing for neurofeedback and task-based paradigms (Koush et al., 2017a). Assessed rtQA parameters are saved at the end of the fMRI run. The rtQA extension is implemented independently from the (pre)processing, neural signal estimation and visualization and could be disabled when computational speed is at the edge of the repetition time of fMRI acquisitions (see *Performance of rtQA extension*). This independent implementation enables further extensions of rtQA modes and visualizations and implementations of rtQA estimators at different levels of volume and time-series (pre)processing.

OpenNFT Python Library

To unify real-time fMRI data processing and neurofeedback applications, we developed a general Python library based on key open-source SPM functions written in C, termed *rtspm* (github.com/OpenNFT/python-rtspm). We transferred original SPM functions to C++ and adapted them for using the pybind11 binding package and NumPy arrays. The binding implies interface functions for calling the compiled code (pybind11.readthedocs.io/en/stable). We created the automatic workflow for compiling, building and publishing distribution packages for Linux, MacOS, and Windows platforms, compatible with Python (3.6–3.9). The *rtspm* library could be installed using the Python dependency manager (python-poetry.org) and run under Python. The *rtspm* library performs the real-time realignment (`spm_realign_rt`) and reslicing (`spm_reslice_rt`) of real-time fMRI volumes, orthogonal volume slicing (`spm_slice_vol`), calculation of the affine transformation matrix (`spm_matrix`) and its parameters (`spm_imatrix`), and Gaussian smoothing of volumes (`spm_smooth`). Python library contributes to extension and unification of the OpenNFT parallel architecture based on

further optimization of the inter-process communication and data flows. Availability of sophisticated estimation operations both in MATLAB and Python allows balancing between educational and performance aspects of the open-source software, as well as paves the way for the development of Python-based OpenNFT. For instance, the *rtspm* library was used to optimize the orthogonal volume slicing by transferring the GUI Helper Process from MATLAB to Python. Its `realign`, `reslice` and `smooth` modules can replace corresponding MATLAB counterparts for developing Python-based real-time fMRI applications.

Participants and Experimental Design

We demonstrated the rtQA performance in a single female participant (age 31 years) using neurofeedback runs without and with excessive head motion, and in a group of 15 participants (7 male, 8 female, age 26 ± 1 years) using real-time fMRI data export simulations for neurofeedback and resting-state runs. All participants were without prior history of neurological or psychiatric diseases and with normal or corrected-to-normal vision. All participants gave written informed consent to participate in the experiment, and all methods used in this study were performed in accordance with the relevant guidelines and regulations of the University Hospital of Geneva approved by the Ethics Committee of the University Hospital of Geneva.

To demonstrate the rtQA performance during excessive head motion, a single participant performed a 2-run neurofeedback experiment. Specifically, we tested the ability of the participant to control the feedback signal by covertly shifting the visual-spatial attention. Each neurofeedback run consisted of nine regulation blocks interleaved with nine baseline blocks (20 s block duration). During regulation blocks, the participant was asked to covertly attend to the right side of the screen while fixating the eyes at the center of the screen. During baseline blocks, the participant was instructed to fixate at the central fixation cross and count backward from a random number that was briefly displayed at the onset of the block. The intermittent feedback signal was provided at the end of each regulation block as a difference between left and right primary visual cortex activations (4 s neurofeedback block duration). The visual cortex ROIs were localized using retinotopic flashing-checkerboard functional localizer (Koush et al., 2013). During the first neurofeedback run, the participant was asked to remain as still as possible. During the second neurofeedback run, however, the participant was asked to specifically move three times when prompted (via headphones). During the first instruction, the participant was asked to moderately slide out of the coil. During the second instruction, the participant was asked to slide back into the coil and strongly tilt the head at least in two directions. During the last instruction, the participant

was asked to move to the opposite directions as compared to the previous instruction, i.e. to try to return to the initial head location.

To demonstrate rtQA in a group of participants that performed a regular neurofeedback experiment complemented with the resting-state acquisition, we simulated real-time data export using previously acquired real-time fMRI data. Specifically, we selected the first neurofeedback training run (17.5 min run duration) and a pre-training eyes-closed resting-state run (6.1 min run duration) from our previous neurofeedback study targeting positive-social emotion regulation (Koush, Meskaldji, et al., 2017c; Krylova et al., 2021). The neurofeedback run consisted of seven trials. Each trial was composed of four regulation blocks interleaved with five baseline blocks of 12 s duration (2.5 min trial duration). During baseline blocks, participants were instructed to passively observe images of neutral objects. During regulation blocks, images with moderately positive-social content were presented, and participants were asked to control their positive emotions to maximize the feedback signal. At the end of each neurofeedback trial participants rested with their eyes open for 38 s, followed by a 4 s display of a feedback value and a monetary reward. The feedback signal was based on a comparison of how well two alternative effective connectivity models fitted the data acquired during the trial. The two models were modeled as top-down and bottom-up interactions between dorsomedial prefrontal cortex (dmpFC) and bilateral amygdala, estimated using dynamic causal modeling (DCM) (Friston et al., 2003) and compared using Bayesian model comparison (Koush et al., 2013; Koush, Meskaldji, et al., 2017c; Penny et al., 2004). During the resting-state run, participants were instructed to remain as still as possible, breathe steadily, avoid specific thinking and falling asleep, which was verified during debriefing. Comprehensive experimental details and findings are reported in the original publications (Koush, Meskaldji, et al., 2017c; Krylova et al., 2021).

Real fMRI Data and Real-time Data (pre)Processing

For neurofeedback experiment also testing the effect of excessive head motion, real-time fMRI data were acquired on a 3 T whole-body MRI system equipped with a 16-channel head receive coil (Trio Tim, Siemens Medical Solutions, Erlangen, Germany). For both neurofeedback runs, 290 whole-brain fMRI volumes were acquired using a single-shot gradient-echo T_2^* -weighted EPI sequence (TR = 1.76 s, TE = 30 ms, 32 slices with 25% distance factor, 64×64 matrix, 3mm^3 isotropic voxels, flip angle $\alpha = 90^\circ$, bandwidth = 2.004 kHz/pixel, GRAPPA, iPAT = 2).

For the simulations, the data has been previously acquired on the same scanner equipped with a 32-channel head

receive coil. For neurofeedback training runs, 1050 fMRI volumes with partial brain coverage were acquired using a single-shot gradient-echo T_2^* -weighted EPI sequence (TR = 1.1 s, TE = 30 ms, 18 slices with 25% distance factor, 120×120 matrix, 1.8mm^3 isotropic voxels, flip angle $\alpha = 70^\circ$, bandwidth = 1.54 kHz/pixel, GRAPPA, iPAT = 3). Resting-state runs were acquired using a multi-band gradient-echo T_2^* -weighted EPI sequence (333 volumes, TR = 1.1 s, TE = 30 ms, 45 slices with 25% distance factor, 120×120 matrix, 1.8mm^3 voxels, flip angle $\alpha = 70^\circ$, bandwidth = 1.49 kHz/pixel, multi-band acceleration factor = 3, GRAPPA with iPAT = 3). EPI protocols were designed to ensure a precise subdivision of the target prefrontal and limbic brain areas, and a short TR. For each scanning session, a T_1 -weighted structural volume was acquired (3D MPRAGE, voxel size = 1mm^3 isotropic, flip angle $\alpha = 9^\circ$, TR = 1.9 s, TI = 900 ms, TE = 2.27 ms).

The real-time fMRI data export was simulated using triggered data copy at a rate of a single volume per 1.1 s. Real-time data (pre)processing included registration and spatial filtering using a 5 mm full width at half maximum (FWHM) smoothing as implemented in OpenNFT. Various data enter rtQA estimations during real-time fMRI (pre)processing (Fig. 1). Head motion parameters and derived head motion estimates (FD, MD) were calculated from the raw fMRI volumes. Volume rtSNR and rtCNR were estimated based on smoothed volumes that were realigned to the fixed template volume and resliced. DVARS, whole-brain iGLM, and time-series rtQA estimates were generated based on the smoothed volumes. Time-series GLM and Kalman filter were applied during temporal processing. The same GLM was used for real-time fMRI data filtering, rtQA, and neurofeedback signal estimation (Koush et al., 2017a). For neurofeedback data per trial, GLM included regressors for the experimental design (i.e., neurofeedback regulation blocks convolved with hemodynamic response function), six head motion, linear trend, high-pass filter, and constant. For resting-state data, the model included the same set of regressors except the experimental condition. We used the same set of regressors for analyzing the whole-brain and time-series data extracted from ROIs and RSNs.

Results

Recursive and Cumulative Estimates

Although the proposed recursive estimators have been validated in original publications, we demonstrate the numerical similarity between recursive and cumulative estimates for neurofeedback runs without and with excessive head motion using time-series from the retinotopically localized right visual cortex. We found negligible difference MSEs $< 1e^{-10}$

between cumulative and recursive estimates for mean, variance, tSNR and tCNR (Fig. 2). During the second neurofeedback run with excessive head motion, we observed increases of mean and variance and decreases of tSNR and absolute tCNR. Note that visual cortex in our neurofeedback runs was suppressed by the attention task, which resulted in a more negative tCNR for the run with low head motion, and a less negative tCNR in the run with excessive head motion.

For neurofeedback runs with different resolutions, we found gradual increase of cumulative tSNR estimation time as compared to recursive tSNR (Fig. 3). Since FD, MD and DVARS are based on the current and the previous fMRI volumes preprocessed in real-time, their estimations require fixed estimation time. In addition to detecting fMRI volumes with FD, MD and DVARS above the pre-selected thresholds, we also estimated non-thresholded recursive temporal average of FD, MD and DVARS as rtQA parameters. The amount of high-frequency noise filtered using Kalman filter

was also estimated recursively in terms of the MSE between unfiltered and filtered time-series. For average FD, MD, DVARS, and MSE of the filtered noise, we found negligible differences ($MSEs < 1e^{-10}$) between recursive and cumulative estimates (Fig. 4).

Real-time Head Motion Estimates and DVARS

We also report head motion parameters, FD and DVARS for neurofeedback runs without and with excessive head motion (Fig. 5). The three distinct large head movements were evident in all head motion parameters, FD and DVARS. For neurofeedback runs, we evaluated translations/rotations both individually and at the group level (Fig. 6A,C; group average; translations: $X = 0.02 \pm 0.1$ mm, $Y = 0.18 \pm 0.32$ mm, $Z = 0.26 \pm 0.45$ mm; rotations: pitch = 0.01 ± 0.00 mm, roll = -0.04 ± 0.19 mm, yaw = -0.06 ± 0.18 mm). Framewise (FD) and micro (MD) displacements were also calculated

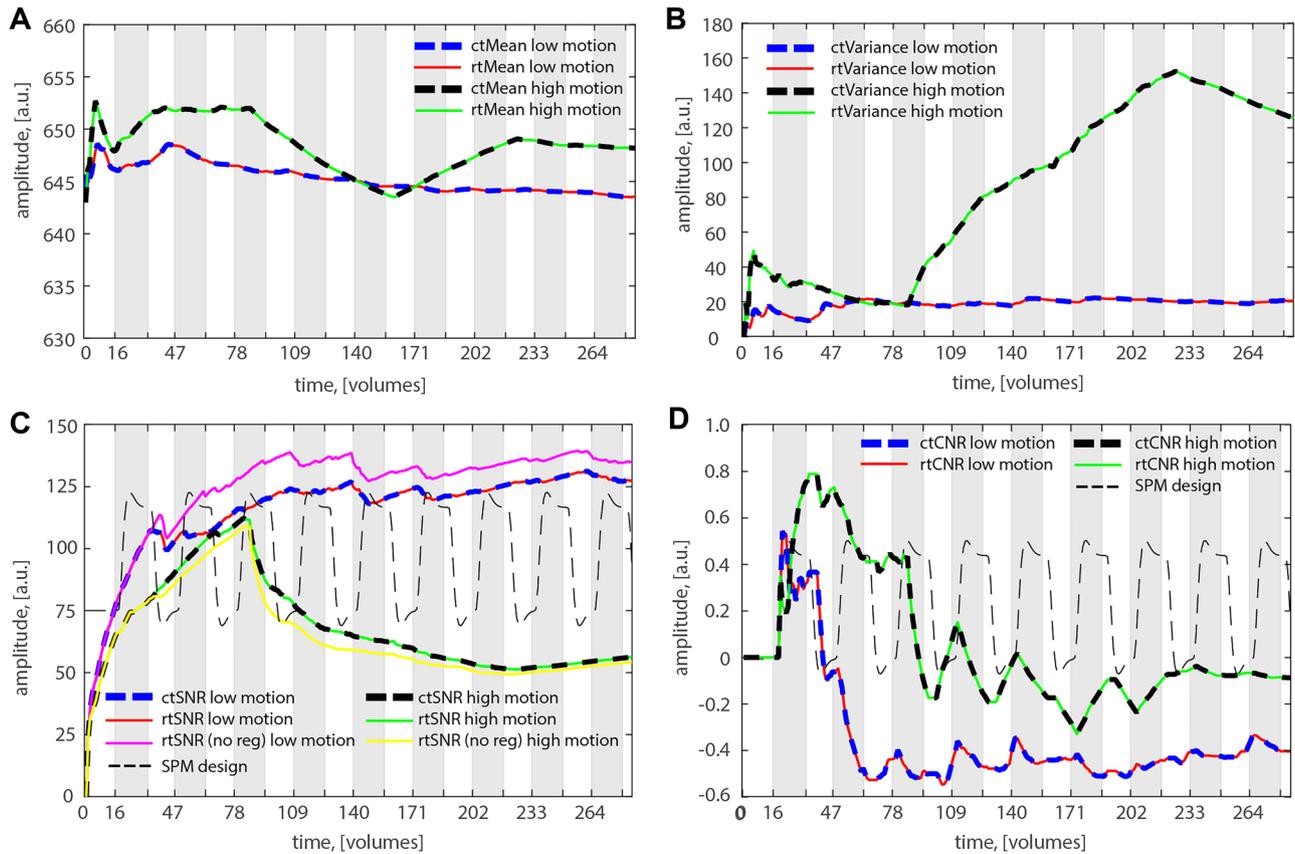
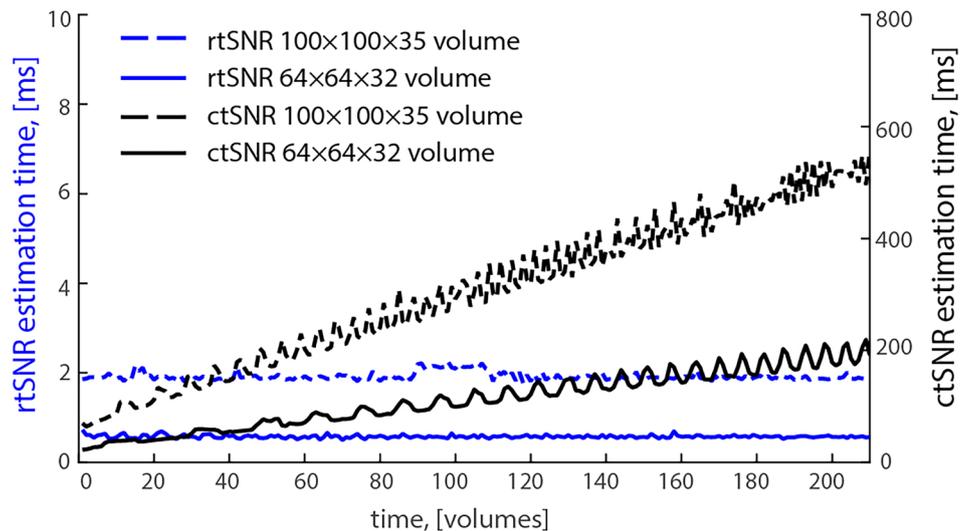


Fig. 2 Recursive and cumulative **A** tMean, **B** tVariance, **C** tSNR and **D** tCNR for neurofeedback runs without (“low motion”) and with (“high motion”) excessive head motion of the same participant. Recursive and cumulative temporal mean and variance were computationally the same with a negligible $MSEs < 1e^{-10}$. tCNR has not been estimated during the first rest phase (i.e., zero on the plots) due to the lack of variance in the real-time design. Black dashed lines represent

the regressor of the experimental condition (grey blocks) modeled as a boxcar function and convolved with the canonical hemodynamic response function as implemented in SPM12 (thin dashed lines). r – recursive (solid lines), c – cumulative (thick dashed lines), “no reg” denotes time-series where activity associated with regulation conditions was regressed out to compute unbiased rtSNR (pink and yellow solid lines, respectively)

Fig. 3 Recursive and cumulative tSNR estimation times for neurofeedback runs with different volume dimensions ($64 \times 64 \times 32$ and $100 \times 100 \times 35$ voxels, 210 volumes). Average time of recursive tSNR estimations was 0.58 ± 0.04 ms and 1.92 ± 0.09 ms, respectively. However, for cumulative tSNR, estimation time was substantially higher and gradually increased from 24 to 217 ms and from 64 to 561 ms, respectively



in real-time. Based on these values, we calculated group average FD (0.10 ± 0.18 mm) and MD (0.04 ± 0.09 mm). We also calculated the individual and group average number of displacements above selected thresholds expressed in percentage from the total number of volumes (Fig. 7A,C; group

average; FD, threshold 0.2 mm: $7.8 \pm 9.2\%$; FD, threshold 0.5 mm: $1.2 \pm 1.9\%$; MD, threshold 0.1 mm: $5.1 \pm 6.5\%$).

For resting-state runs, we evaluated translations/rotations both individually and at the group level (Fig. 6B,D; group average; translations: X: 0.00 ± 0.11 mm, Y: 0.06 ± 0.10 mm,

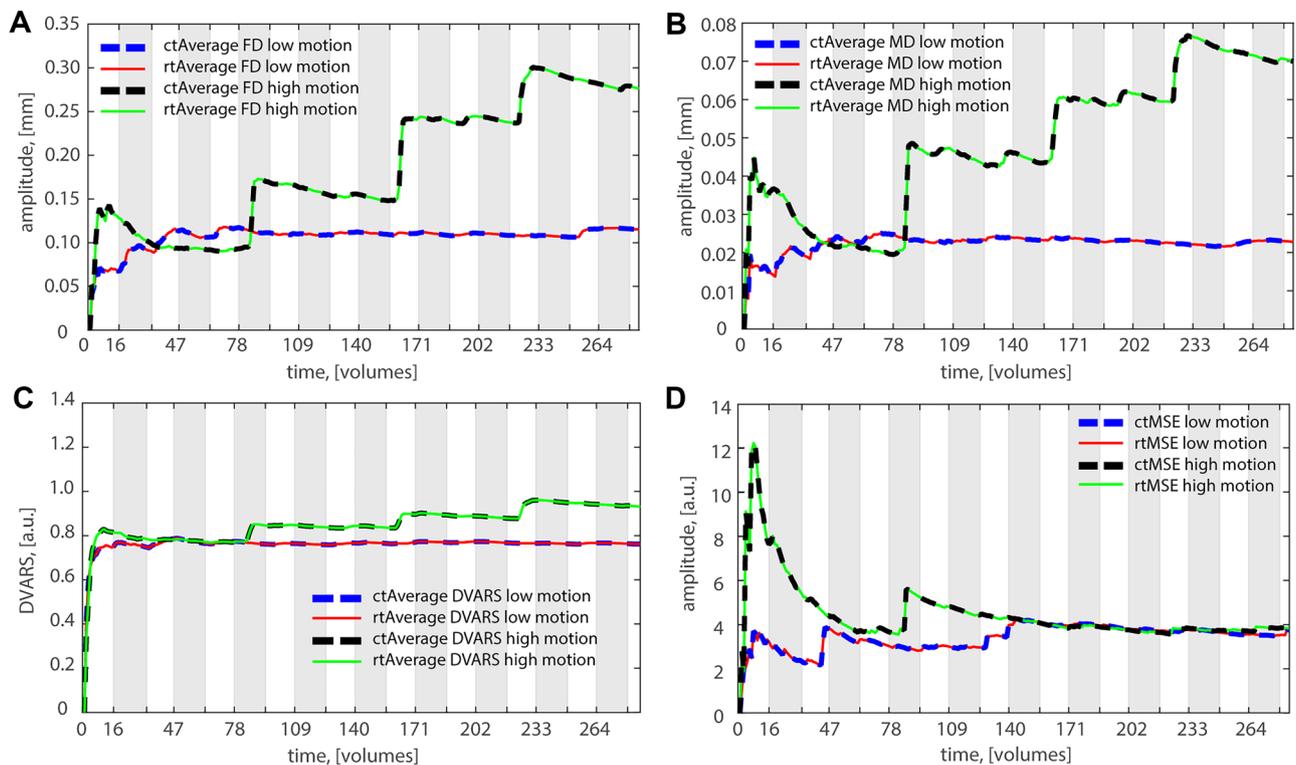


Fig. 4 Recursive and cumulative temporal average **A** FD, **B** MD, **C** DVARS and **D** the amount of high-frequency noise by means of MSE for neurofeedback runs without ("low motion") and with ("high motion") excessive head motion of the same participant. Conven-

tional six head motion parameters were provided by SPM12 realignment routines adapted for real-time application (Koush et al., 2017a). MSEs between all cumulative and recursive estimates were negligible ($< 1e^{-10}$)

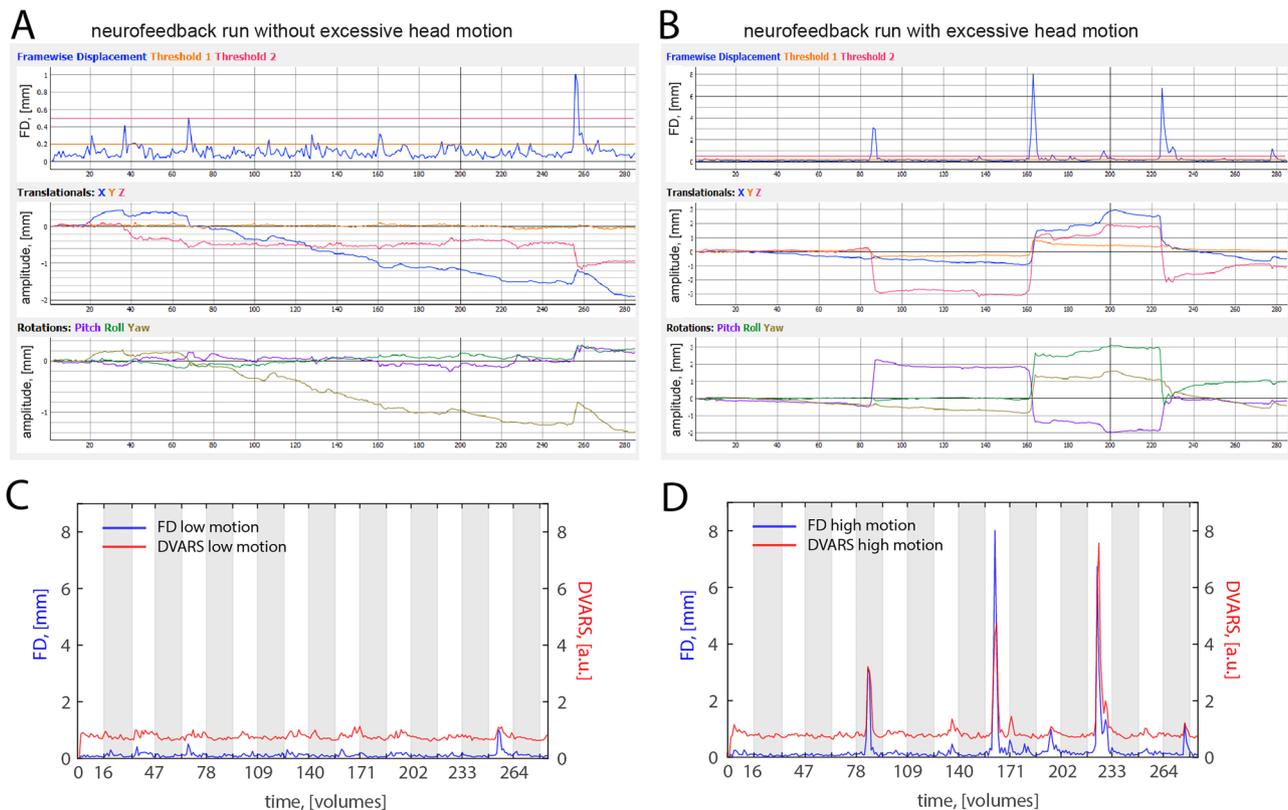


Fig. 5 Real-time fMRI neurofeedback runs without (“low motion”) and with (“high motion”) excessive head motion. **A, B** Head motion parameters as they appear in OpenNFT. **C, D** FD and DVARS. **B, D**

Three head movements are seen as consistent step-like displacements in translations and rotations, as well as spike-like displacements in FD and DVARS

Z: -0.04 ± 0.18 mm; rotations: pitch: 0.02 ± 0.12 mm, roll: -0.03 ± 0.10 mm, yaw: -0.06 ± 0.11 mm). We estimated group average FD (0.12 ± 0.08 mm) and MD (0.03 ± 0.03 mm), as well as the number of these displacements above selected thresholds (Fig. 7B,D; group average; FD, threshold 0.2 mm: $9.2 \pm 11.4\%$; FD, threshold 0.5 mm: $0.6 \pm 0.5\%$; MD, threshold 0.1 mm: $1.4 \pm 2.2\%$).

We evaluated DVARS both individually and at the group level for time-series of the whole-brain ROI in neurofeedback runs (Fig. 7E; group average 1.6 ± 0.6) and resting-state runs (Fig. 7F; group average 1.2 ± 0.4). We also calculated individual and group average of number of volumes with DVARS values above selected threshold (5a.u.) expressed in percentage of the total number of volumes ($< 3\%$ in all runs).

Recursive tSNR and tCNR

For time-series of the target ROIs, we evaluated group average rtSNR in neurofeedback runs (Fig. 8A; left amygdala: 92.2 ± 36.4 , right amygdala: 108.2 ± 45.7 , dmPFC: 186.5 ± 118.7) and in resting-state runs (Fig. 8B; Table 1, RSNs). For an exemplary real-time fMRI resting-state run, we also illustrated voxel-wise rtSNR (Fig. 1A). We

evaluated group average rtCNR for time-series of three target ROIs (Fig. 8C; left amygdala: 0.10 ± 0.32 , right amygdala: 0.16 ± 0.35 , dmPFC: 0.25 ± 0.54). Of note, activity associated with regulation condition was regressed out from rtSNR for neurofeedback runs (Fig. 2C).

Kalman Filter Denoising and iGLM Confounds

We used Kalman filter to identify and count positive and negative spikes in time-series of ROIs and RSNs. For neurofeedback runs, the group average number of positive and negative spikes was 10.4 ± 4.3 and 11.3 ± 4.2 for left amygdala, 9.1 ± 2.2 and 10.7 ± 3.0 for right amygdala, 7.1 ± 4.1 and 5.4 ± 3.0 for dmPFC, respectively (Fig. 9A,B). For resting-state runs, the group average number of identified spikes is also shortlisted (Fig. 9C,D; Table 1). Note that the sensitivity to spike identification was controlled by the discrepancy threshold of the Kalman filter, which could be set so that less spikes are identified (Koush et al., 2012). For neurofeedback time-series, we evaluated the amount of high-frequency noise filtered by the Kalman filter using rMSE (Fig. 10A; group average, left amygdala: 11.0 ± 7.4 , right amygdala: 8.7 ± 5.7 , and dmPFC: 4.0 ± 8.4). For resting-state

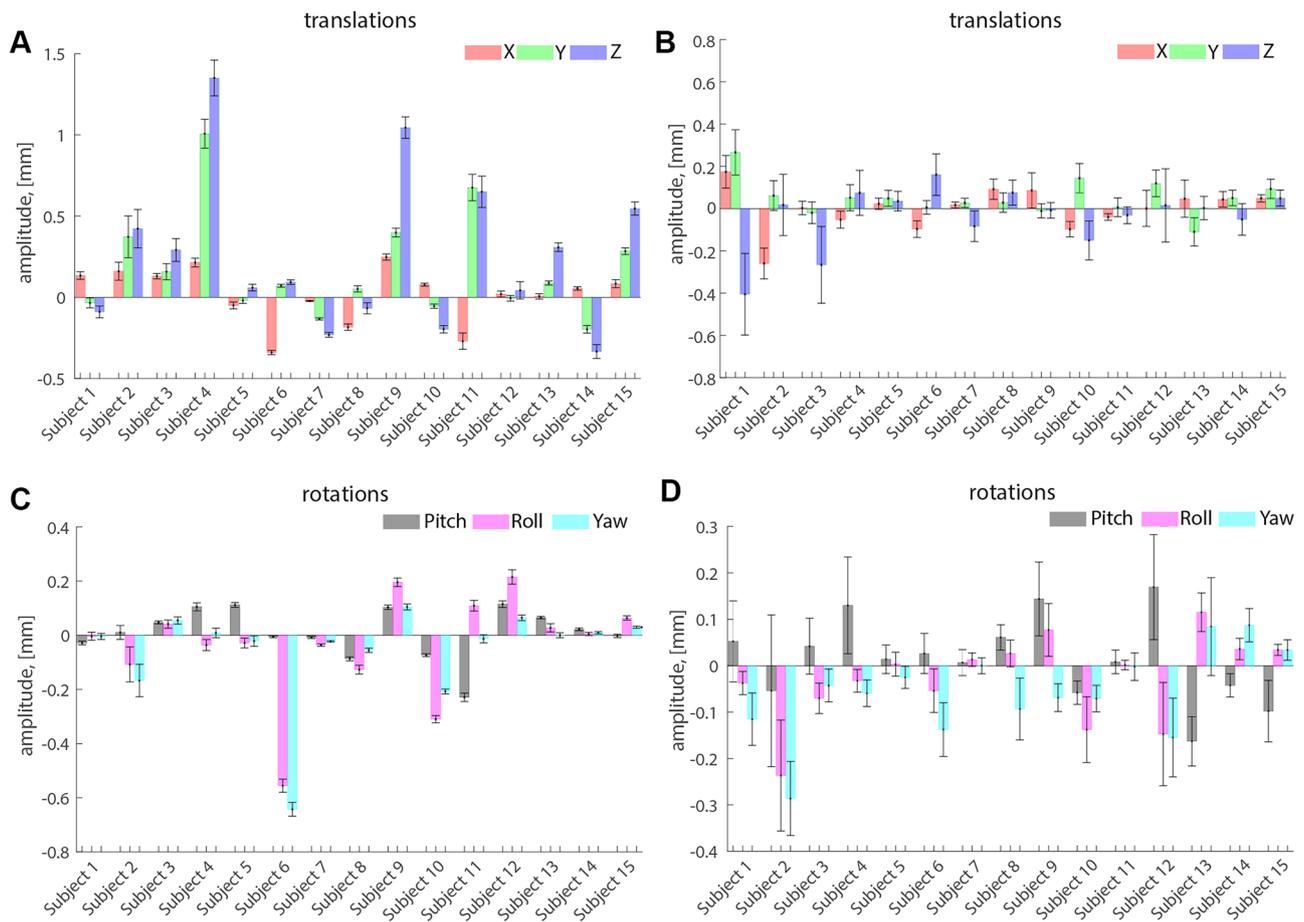


Fig. 6 Individual head motion parameters. Translations and rotations averaged per neurofeedback **A, C** and resting-state **B, D** runs. Head motion parameters were estimated using the realignment routine of SPM12 as implemented in OpenNFT. Error bars denote standard deviation

time-series of RSNs, group average rMSE is also shortlisted (Fig. 10C; Table 1).

For time-series and whole-brain iGLM analysis, head motion parameters and linear trend beta coefficients were estimated and visualized using contrasts for nuisance regressors. For neurofeedback and resting-state time-series, we evaluated group average linear trend beta coefficients (Fig. 10B,D; linear trend beta per neurofeedback trial, left amygdala: -0.2 ± 1.6 , right amygdala: -0.4 ± 1.6 , dmPFC: 0.1 ± 2.3 ; for linear trend betas of RSNs, see Table 1).

Performance of rtQA Extension

The performance of rtQA extension was assessed using real-time data export simulations on a desktop PC with Intel Core i7-8700 (3.2 GHz, 6 cores, 16 GB RAM with 2666 MHz), SSD disk (writing speed 550 Mb/s, reading speed 525 Mb/s), Windows 10, MATLAB R2021b and Python 3.9.7. The rtQA extension moderately increased group average data processing time by 88.2 ± 20.9 ms

for neurofeedback runs ($120 \times 120 \times 18$ volumes) and by 209.2 ± 24.4 ms for resting-state runs ($120 \times 120 \times 45$ volumes) (Table 2). Data processing time represents the time in core Python process for processing a single fMRI volume, updating the OpenNFT GUI, and transferring data between the concurrent processes.

Discussion

To facilitate an efficient fMRI data collection and decide about the quality of fMRI volumes in real-time, we applied recursive and real-time methods of rtQA and developed the rtQA extension of the OpenNFT software. Specifically, we implemented recursive mean, variance, tSNR, tCNR, quality parameters based on GLM estimates and denoising of time-series, as well as real-time DVARS, head motion parameters and derived estimates. The feasibility of applied rtQA was demonstrated in real-time fMRI neurofeedback and resting-state runs.

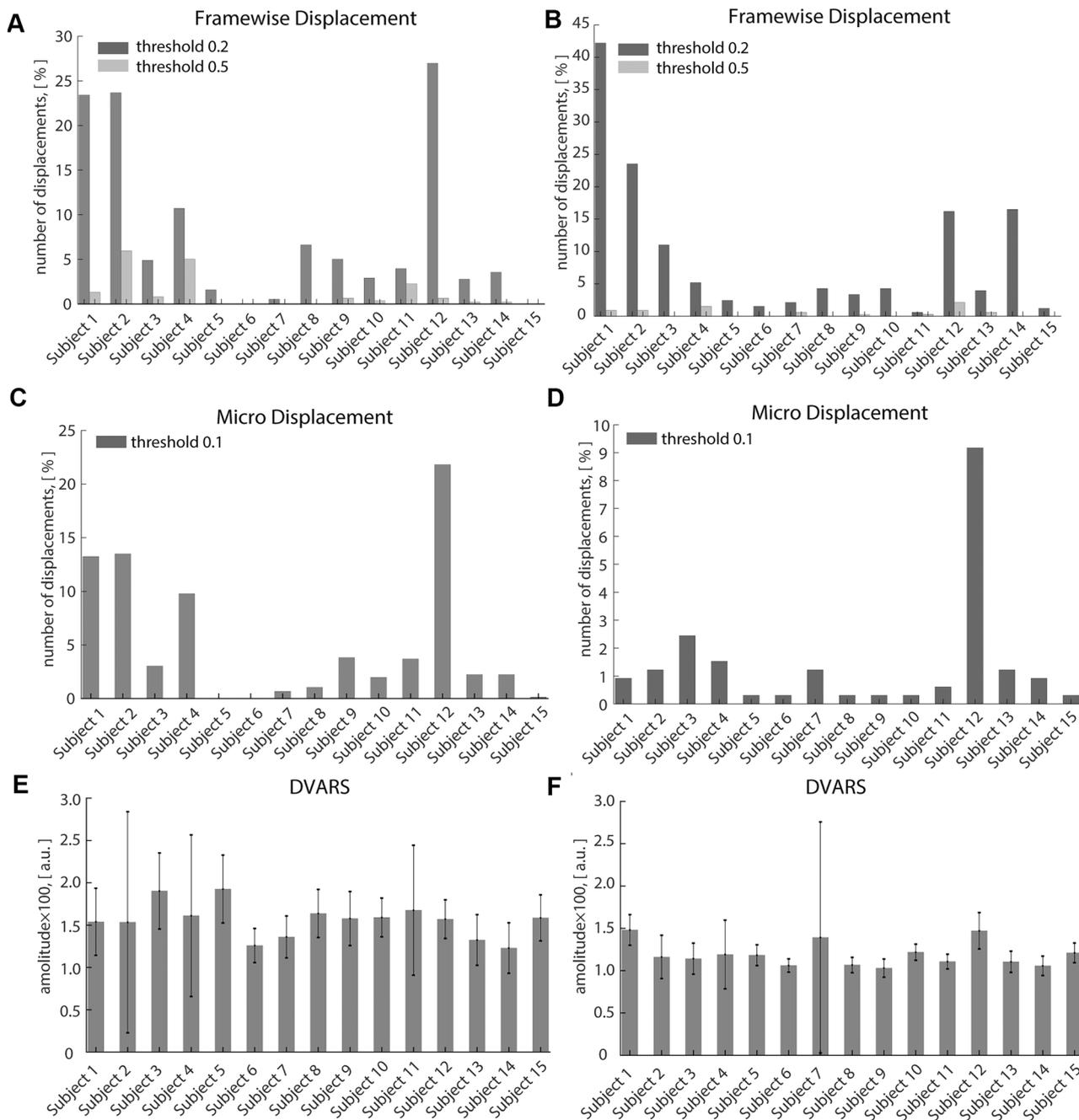


Fig. 7 Individual head motion parameters. For neurofeedback **A, C, E** and resting-state **B, D, F** runs, we illustrated thresholded FD and MD evaluated per run, as well as DVARS averaged per run, respectively. Error bars denote standard deviation

Recursive tSNR and tCNR

We found only negligible difference between recursive and cumulative mean, variance, tSNR and tCNR. These differences are also negligible in the presence of excessive head movements because these methods are independent of it. The low errors between recursive and cumulative estimates confirm the feasibility of recursive methods for

rtQA and confirm that they can preserve the precision and informativeness of the original cumulative methods with a much lower computational cost. Specifically, the estimation time of cumulative tSNR progressively increased as compared to recursive tSNR, because cumulative estimations were applied to all data up to each time point. In contrast, recursive estimations facilitate fixed memory and computation time because the contribution of new

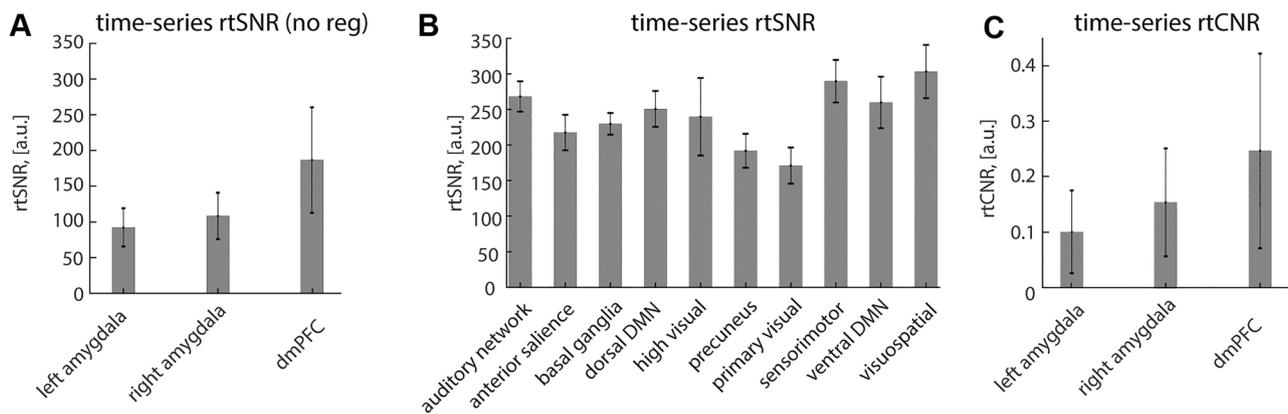


Fig. 8 Time-series rtCNR and rtSNR. Group average rtSNR for **A** neurofeedback runs in bilateral amygdala and dmPFC, and **B** for resting-state runs in target networks. For neurofeedback runs, activity

associated with regulation condition was regressed out from rtSNR using real-time GLM. **C** rtCNR in bilateral amygdala and dmPFC. Error bars denote standard deviation

data was directly incorporated into the estimated values at each time point (Bagarinao et al., 2003; Welford, 1962). Thereby, recursion allows computationally heavy estimations for real-time fMRI applications, such as whole-brain tSNR, tCNR, and GLM (see *Incremental GLM*). Of note, rtSNR and rtCNR could fluctuate during the initial volumes and stabilize towards the end. These intrinsic instabilities could be due to the shortcoming of the recursive estimation in small data samples.

It has been shown that tSNR could be applied to assess the quality of task-related fMRI runs when considering the residual of the GLM fit (Murphy et al., 2007), which was used to investigate quality differences between control and neurofeedback groups (Zilverstand et al., 2017). Specifically, tSNR in dorsal ACC was substantially higher in the

neurofeedback group (145) as compared to the control group (98) due to more extensive head motion in the control group. Consistently, activity associated with regulation condition was regressed out from tSNR for neurofeedback time-series, and the resulted rtSNR varied from 92.2 ± 36.4 in left amygdala to 186.5 ± 118.7 in dmPFC.

Average rtCNR in bilateral amygdala (0.10 ± 0.32 , 0.16 ± 0.35) and dmPFC (0.25 ± 0.54) during emotion regulation were also consistent with previously reported rtCNR in medial frontal and middle temporal gyrus during auditory/visual imagery, and anterior cingulate cortex (ACC) during emotion regulation (0.1–0.5) (Koush et al., 2012). Of note, tCNR is strongly individual and highly sensitive to task complexity (Welvaert & Rosseel, 2013), e.g., it varied from -0.1 to 1.8 in a few participants during motor imagery in the supplementary motor area (Koush et al., 2012). Whole-brain rtCNR maps provide very similar information as whole-brain activation maps, which is not surprising considering that its estimation is also based on GLM but with a simple model. The whole-brain rtCNR maps, however, are confounded by noise typically explained away by means of nuisance regressors; therefore, these maps can inform about the magnitude of signal relative to the noise. For this comparison, whole-brain rtCNR estimation is provided in our rtQA extension.

The rtSNR in the resting-state time-series varied from 170.8 ± 88.7 in the primary visual network to 303.2 ± 124.7 in the visuospatial network. This is overly consistent with previously reported resting-state tSNR values averaged across all voxels (ca. 100 – 280, TR = 3 s) (Van Dijk et al., 2012), as well as in the default mode network, the subcortical areas, and the global grey matter (ca. 130 – 200, TR = 2 s; ca. 60 – 160, TR = 1.1 s) (DeDora et al., 2016), and with tSNR across non-activated voxels in frontal cortex (ca. 132 – 203,

Table 1 Group average rtQA parameters for time-series of the RSNs in the resting-state runs, (mean \pm std)

network	rtSNR	+ spikes	- spikes	rMSE	trend
auditory	268.0 \pm 99.6	4.2 \pm 3.3	4.4 \pm 2.8	0.7 \pm 0.6	1.1 \pm 2.7
anterior salience	217.3 \pm 92.2	4.0 \pm 3.6	3.3 \pm 2.5	1.0 \pm 0.6	1.0 \pm 3.1
basal ganglia	229.6 \pm 72.6	4.8 \pm 2.4	5.3 \pm 3.6	1.2 \pm 0.6	0.6 \pm 1.9
dorsal DMN	250.5 \pm 90.4	3.3 \pm 3.9	3.0 \pm 1.8	0.7 \pm 0.5	0.6 \pm 2.0
higher visual	239.6 \pm 149.5	3.0 \pm 3.4	3.5 \pm 2.9	1.0 \pm 1.2	0.8 \pm 2.3
precuneus	191.6 \pm 70.1	3.7 \pm 3.7	3.1 \pm 2.4	1.2 \pm 0.8	0.4 \pm 2.9
primary visual	170.8 \pm 88.7	3.4 \pm 4.1	3.8 \pm 2.2	1.8 \pm 1.9	1.1 \pm 3.5
sensorimotor	289.6 \pm 123.9	4.5 \pm 3.8	4.1 \pm 3.0	0.6 \pm 0.5	0.7 \pm 2.0
ventral DMN	259.6 \pm 113.3	4.3 \pm 4.0	2.5 \pm 1.8	0.6 \pm 0.4	0.5 \pm 2.0
visuospatial	303.2 \pm 124.7	4.1 \pm 4.4	3.5 \pm 3.1	0.5 \pm 0.3	0.9 \pm 2.1

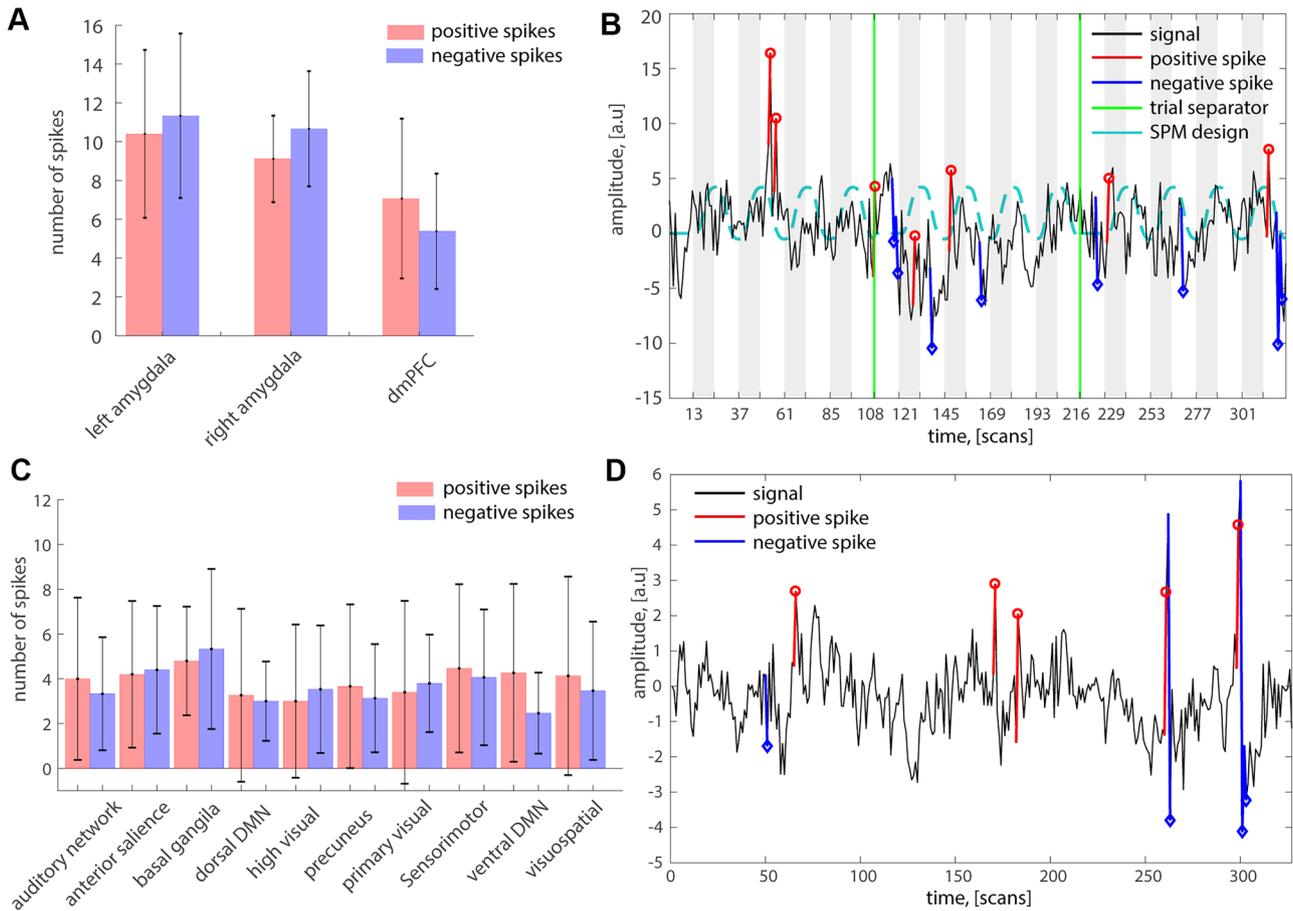


Fig. 9 Detection of spikes using Kalman filter. For neurofeedback runs, we **A** estimated the group average number of positive and negative spikes for three ROIs and **B** illustrated an exemplary spike detection for the dmPFC time-series. Similarly, for resting-state runs, we **C** estimated the group average number of spikes for the target RSNs and **D** illustrated an exemplary spike detection for the dorsal DMN time-

series. **B, D** Red circles and blue rhombs mark positive and negative spikes, respectively. **B** Green lines demark separations between three exemplary trials, the blue dashed line illustrates the regulation regressor, and grey bars indicate condition blocks. Error bars denote standard deviation

TR = 2 s) (Posse et al., 2012), while considering different acquisition parameters and head coils (Triantafyllou et al., 2011; Welvaert & Rosseel, 2013).

Head Motion Parameters

Head motion parameters are commonly estimated post-hoc when fMRI data is completely acquired, imposing the risk of losing data of entire participants. Real-time head motion analysis monitors fMRI data quality and reduces experimental risks and costs by means of reducing the amount of overscanning required to collect sufficient data under low-movement criterion, e.g. 20 min resting-state runs given $FD < 0.2$ mm (Dosenbach et al., 2017). Along with preventing MRI data distortions by head motion (e.g., by optimizing head and body fixations and experimental instructions to the participants), real-time head motion analysis allows substantiated interruptions and timely re-scans of experimental runs

instead of complete exclusion of individual data with large head displacements.

The post-hoc FD estimate was proposed to detect the motion-related effects in fMRI data (Power et al., 2012, 2014). Its real-time characterization is based on real-time calculation of head motion parameters during fMRI data acquisition (Dosenbach et al., 2017). In our real-time fMRI data, low-movement criterion ($FD < 0.2$ mm) was exceeded in 7.8% of the neurofeedback runs with a duration of 17.5 min and in 9.2% of the resting-state runs with a duration of 6.1 min, which is consistent with above 90% data surveillance under the same criterion (Dosenbach et al., 2017). As compared to FD, MD estimation is based only on the three translation parameters (Van Dijk et al., 2012), however, its real-time estimation could facilitate disentangling artifacts from head translations and rotations.

OpenNFT implements accurate real-time realignment based on the recommended rigid body spatial transformation

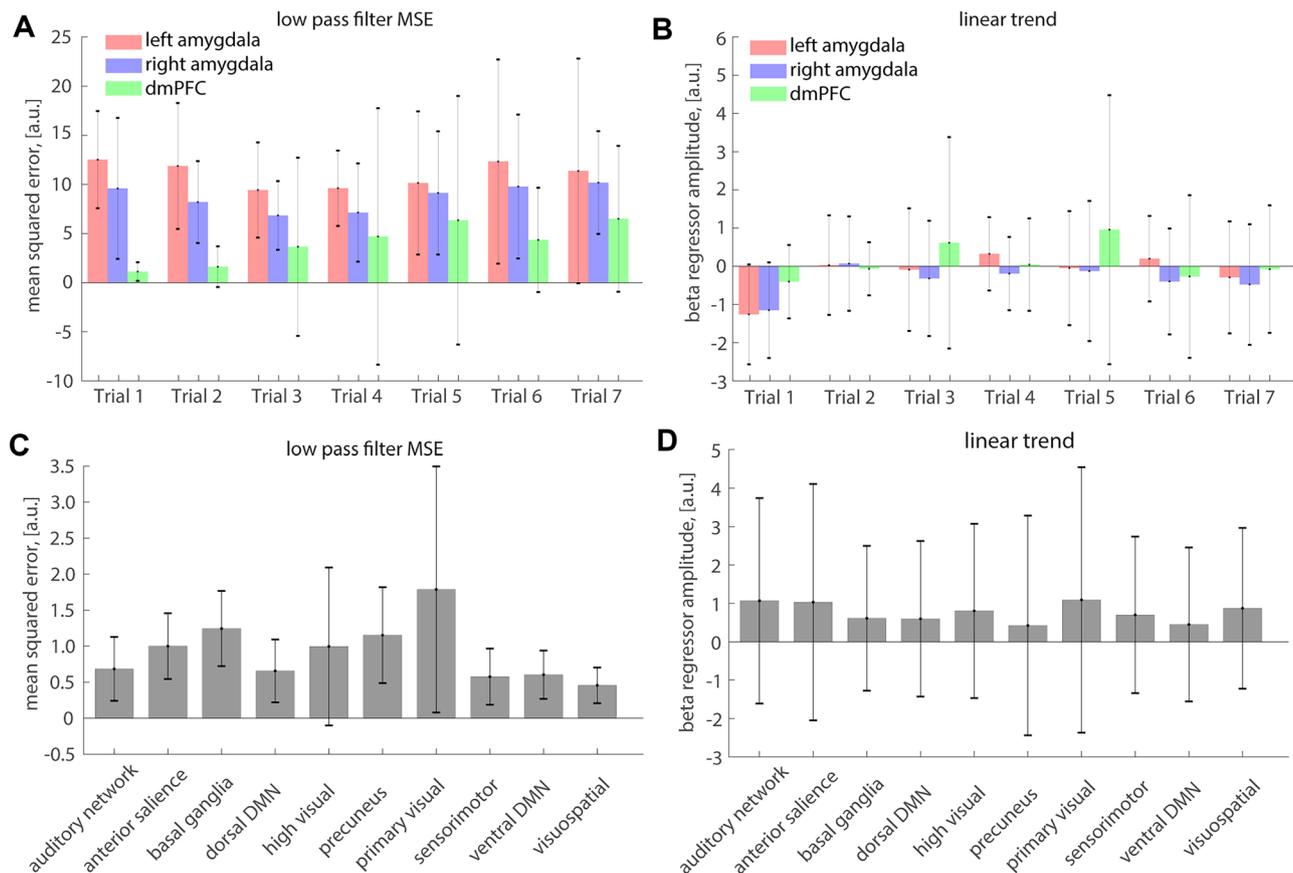


Fig. 10 Kalman filter recursive MSE and linear trend beta values. For neurofeedback runs, we illustrated group average **A** rMSE between raw and filtered time-series and **B** linear trend iGLM beta values for

three ROIs. For resting-state runs, we illustrated group average **C** rMSE between raw and filtered time-series and **D** linear trend beta values for target RSNs. Error bars denote standard deviation

and interpolation using B-splines of the 4th order, and the quality of the real-time implementation comes close to that of the conventional offline realignment in SPM (Koush et al., 2012, 2017a). Although optimized for real-time, these estimations are not recursive and may require 300–600 ms depending on the algorithm complexity and data size (Koush et al., 2017a). Due to the high precision level, rigid body preprocessing of fMRI volumes takes most of the processing time in OpenNFT. To minimize the number of iterations and reduce the computation time, the template volume should have the same dimensions as the real-time

fMRI volumes. This is typically accomplished by acquiring the template volume with the same fMRI sequence and acquisition parameters. Computationally less extensive (e.g., cubic) interpolations may be substantially faster, however, at the potential expense of accuracy.

DVARS

DVARS is a framewise data quality index that reflects the rate of volume intensity change across whole-brain. DVARS is similar to FD and can be used for data scrubbing, although it has no explicit relation to the head motion parameters (Power et al., 2012). It can capture distortions in fMRI data also due to other sources of motion, e.g. chest motion (Fair et al., 2020). We applied DVARS as rtQA parameter using (i) intensity average within the whole-brain mask defined based on the fixed template volume used for the realignment, (ii) realignment and reslicing procedures to ensure that voxels within the fixed mask are identically sampled, and (iii) whole-brain data scaling based on the median of voxel intensities within the mask. The threshold definition for

Table 2 Group average data processing time per acquired volume, (mean \pm std) ms

OpenNFT mode	neurofeedback (120 \times 120 \times 18)	resting-state (120 \times 120 \times 45)
rtQA ON	446.3 \pm 55.5	1091.3 \pm 51.3
rtQA OFF	357.7 \pm 41.4	842.6 \pm 21.2
difference	88.2 \pm 20.9	209.2 \pm 24.4

DVARS-based fMRI volume scrubbing is usually arbitrary as changes in signal intensity could vary across scanners and sequences. Given that DVARS estimates were multiplied by 100 after scaling, we applied coarse 5a.u. threshold for an exemplary volume censoring. This threshold could be set based on local site settings for offline data processing or box-plot right-outliers (1.5 interquartile range above the 75% percentile; as implemented in FSL, the FMRIB Software Library) (Jenkinson et al., 2012). Recently, a more formal approach for DVARS has been proposed that showed it to be a part of the sum of squares decomposition of the 4D fMRI data along with the thresholding based on DVARS inference testing (Afyouni & Nichols, 2018). These methods may require additional adaptation for real-time data processing.

Incremental GLM

We extended iGLM (Bagarinao et al., 2003) for rtQA using weights of modelled regressors as quality estimators. For time-series and whole-brain data, results of iGLM are shown to the experimenter in real-time as plots and whole-brain maps to evaluate the strength of the specific contamination and to identify brain regions with significant contamination from specific sources. Multiple nuisance GLM regressors typically include at least six head motion parameters, white matter and cerebrospinal fluid regressors, high-pass filter and linear trend regressors, regressors based on volume censoring, physiological noise, as well as their derived estimates, principal components, and various combinations (Parkes et al., 2018). Although iGLM is a flexible tool for real-time fMRI data processing, the necessity to process data during its acquisition, as compared to post-hoc processing, substantially limits the number of regressors, because iGLM precision takes a substantial time to stabilize if the number of regressors is large (Misaki et al., 2015). Nevertheless, six head motion, high-pass filter and linear trend regressors are often used as nuisance regressors for processing time-series and whole-brain volumes in real-time as implemented in OpenNFT (Koush et al., 2017a). A certain extension of the nuisance regressors based on recommendations can be feasible given the lower number of regressors of the interest, however, it requires a more systematic evaluation (Misaki et al., 2015; Parkes et al., 2018).

Kalman Filter Denoising

A non-linear modified Kalman filter is feasible to filter high-frequency noise and to correct spike-like artifacts in real-time fMRI time-series (Koush et al., 2012; Koush, Meskaldji, et al., 2017c; Lorenz et al., 2016). Here, we demonstrated that recursive MSE between raw and Kalman-filtered time-series could be used as the real-time estimate of the amount of filtered high-frequency noise. In addition,

identified spikes could be counted and highlighted in real-time. Notably, for real-time fMRI analysis, modified Kalman filter provides more effective spikes detection and correction in comparison to conventional filtering methods, such as EMA and Butterworth filter (Koush et al., 2012). The modified Kalman filter is feasible for different event- and block-related designs and is efficient if applied before signal averaging, however, its parameters need to be justified based on simulations. This includes the approximate cutoff frequency and threshold between predicted and posterior estimates to control for the identification of outliers given the repetition time (Koush et al., 2012). Stability of the Kalman filter depends on the steady state of model parameters and may require about 5–10 iterations to provide the reliable filtered output or rtQA estimate (Koush et al., 2012). Data scrubbing approaches also inspired further real-time spike detection and correction techniques, such as that using various statistical scores (Heunis et al., 2020) and that modeling identified spikes as regressors of no interest in iGLM. More advanced methods, such as despiking based on wavelets (Patel et al., 2014) and Schrödinger filtering (Benigno et al., 2021) are also promising but may require additional adaptation for real-time data processing either based on recursion or sliding-window.

Practical Considerations

Recursive and real-time methods could be jointly applied for fMRI data processing to increase informativeness and optimize scanning time. For instance, iGLM could be used for brain activity estimation and visualization, for data filtering, and for rtQA (e.g., to assess contributions of nuisance regressors). Versatile rtQA tools are particularly important to assess consequences of image distortions during data acquisitions due to technical and physiological noise. It provides a quick evaluation of the quality of fMRI data and allows informed decision to interrupt and/or restart data acquisitions if needed. In our excessive head motion example, an operator could have interrupted the scanning session already after the first large motion of the participant.

Some QA parameters are related directly or indirectly. For instance, head motion is directly captured in the six head motion parameters, the derived FD and MD parameters, and changes in the whole-brain intensity values in terms of DVARS. Rather indirectly, excessive head motion may result in decreases of tSNR, tCNR, and (de)activation statistics, as well as in increases of the amount of noise to be filtered and weights of the corresponding nuisance regressors. Therefore, the effect of head motion on fMRI data quality could be also assessed using whole-brain tSNR (Van Dijk et al., 2012) and iGLM. Specifically, whole-brain (de)activation maps associated with the experimental design as well as activation maps associated with the nuisance head motion and linear

trend regressors should be cross-checked in addition to head motion parameters to evaluate the magnitude of target brain (de)activation and artifactual activity in target brain areas.

The instantiation of the proposed recursive and real-time methods does not require additional adjustments in OpenNFT. However, these methods could be further optimized through a pilot fMRI run to reach the best trade-off between the imaging parameters including the repetition time, data complexity, accuracy, and computational needs. The rtQA parameters could also be used to evaluate different preprocessing methods and fMRI volumes after different preprocessing stages (Heunis et al., 2020; Koush et al., 2012).

Although temporal dynamics of model parameters and intermediate estimates is largely neglected in neurofeedback studies, it often leads to underestimation of the feedback signal in the beginning of neurofeedback runs (Koush et al., 2017a; Misaki et al., 2015). Time-series of mean, variance, rtSNR, rtCNR, and iGLM estimates provide insights into the temporal dynamic of the data and the derived estimates (e.g., percent signal change feedback signal based on cumulative average) and filtered feedback signal (e.g., based on cumulative/recursive GLM) and inform about the reliability of the feedback signal and the filtering method. While systematic

research is needed to improve the initial period of instability of recursive estimates, longer initial baseline blocks could partially compensate for some instability. Since iGLM and corresponding algorithms are implemented in the OpenNFT pipeline, it allows the exploration of the model space to define, extend, and justify models based on pilot data.

OpenNFT rtQA and other Real-time fMRI Quality Assessment Software

To the best of our knowledge, we shortlisted the available real-time fMRI quality assessment software (Table 3). We provided general details on the software availability, type, programming language, availability of parallel/multithreading computing, support of the GPU computations, provided rtQA estimates and some other available key real-time estimates. It must be noted that despite strongly developed quality control/assurance/assessment software in neuroimaging and vast availability of quality control methods, real-time quality assessment is generally underrepresented in the literature. This is also true for neurofeedback and brain-computer-interface (BCI) literature, which historically pioneered the real-time data processing, yet did not really

Table 3 rtQA software overview (alphabetically) focusing on their key rtQA and real-time estimates

software	availability	type	language	ParC	GPU	rtQA estimates	rt estimates	source
FRIEND	open-source	framework	C++	yes	no	MP, RMS of MP	PSC, SVM, corr	https://nitrc.org/projects/friend (Basilio et al., 2015; Sato et al., 2013)
FIRMM	commercial	standalone	Matlab, Python	na	no	MP, FD	na	https://nousimaging.com/ (Dosenbach et al., 2017)
OpenNFT	open-source	framework, library	Python, Matlab, C++	yes	no	ROI _m , ROI _v , tSNR, tSNRT, tCNR, MP, FD, MD, Spikes, FN, WB ROI, DVARs	VOL/ROI iGLM, PSC, SVM, corr	https://opennft.org/ (Koush et al., 2017a)
Pyneal	open-source	framework	Python	yes	no	AD, RD	PSC, corr	https://github.com/jeffmacinnes/pyneal (MacInnes et al., 2020)
rt AFNI	open-source	plugin	C, Python	na	yes	MP	PSC, SVM, corr	https://pypi.org/project/afniRTI
TBV	commercial	standalone	C++	yes	yes	MP	VOL/ROI rLS, GLM, PSC, ICA, SVM, corr	https://brainvoyager.com/products/turbobrainvoyager.html

Framework software – a software that provides tools (workflows, modules, libraries, functions) for developing user-defined modifications and extensions, standalone software – a software that does not require other software tools (workflows, modules, libraries, functions) to operate, library – a suite of generic tools (workflows, modules, functions) that is used to develop software programs and applications

ParC parallel computing, *GPU* graphics processing unit, *na* not available/unknown, *rt* real-time, *VOL* whole-brain volume, *ROI* region of interest, *MP* motion parameters (3 translations and 3 rotations), *FD* framewise displacement based on MP, *MD* micro displacement, *ROI_m* ROI mean, *ROI_v* ROI variance/std, *ROI PSC* ROI percent signal change, *VOL_v* VOL variance/std, *tSNR* temporal SNR for rest paradigm, *tSNRT* tSNR for task paradigm with task regressor regressed out, *tCNR* temporal CNR for task paradigm, *Spikes* detection of spikes/outliers, *AD* absolute displacement (compared to the first VOL), *RD* relative displacement (compared to the previous VOL), *SVM* support vector machine, *FN* amount of the filtered high-frequency noise evaluated using recursive MSE, *RMS* root mean square, *WB ROI* whole-brain fMRI ROI time-series and its rtQA estimates, *DVARs* temporal derivative of RMS variance over WB ROI voxels, *LT* linear trend, *GLM* general linear modeling, *rLS* recursive least squares, *ICA* independent component analysis, *iGLM* incremental GLM, *corr* (sliding-window) correlation

get use of rtQA. Thus, just a few neurofeedback software developers noted that their tools could be used for rtQA as well, e.g. in FRIEND (Basilio et al., 2015; Sato et al., 2013), real-time AFNI plugin, TBV, and Pyneal (MacInnes et al., 2020). Thereby, the present rtQA extension using OpenNFT appears as the tool that significantly extends the number of feasible rtQA estimates by means of various real-time and recursive methods. Further research and developments is needed to transfer the power of offline quality control methods to real-time applications (Alfaro-Almagro et al., 2018) and to improve the efficiency of real-time algorithms.

Conclusion

We developed an automatic rtQA extension of the OpenNFT software to facilitate real-time analyses of key fMRI quality parameters. Specifically, we implemented the recursive tSNR and tCNR for the whole brain and time-series extracted from regions of interest, iGLM nuisance regressors for whole-brain and time-series data, number of spikes and amount of high-frequency noise in time-series, as well as real-time head motion translations, rotations, framewise and micro displacements, and DVARS. The rtQA extension was implemented in the open-source OpenNFT software written in Python, MATLAB and C++. We demonstrated our rtQA developments using real-time fMRI neurofeedback and resting-state runs. Our GUI-based multi-thread software implementation allows for parallel estimation and real-time monitoring of both time-series and volumetric quality parameters at negligible time costs. User-friendly automatized rtQA of fMRI is of particular importance for an efficient data acquisition in brain research and clinical applications.

Information Sharing Statement

The most recent OpenNFT version is freely available under the GNU GPL license at GitHub (github.com/OpenNFT). This software framework is based on practices of the platform-independent interpreted programming languages Python (python.org) and MATLAB (MathWorks, MA, US), as well as pre-compiled programming language C++ (isocpp.org) to facilitate concurrent functionality, high modularity, and extensibility. OpenNFT includes, but is not limited to, the functionality of SPM (fil.ion.ucl.ac.uk/spm), PsychoPy (psychopy.org), and Psychtoolbox (psychtoolbox.org) software suites. OpenNFT supporting materials include the opennft.org website linking to the up-to-date software framework and library, thorough installation instructions for each supported platform, test

routines, educational courses, video tutorials, real-time fMRI demo data (Koush et al., 2017b), and recently added task-based and resting-state demo data. OpenNFT GitHub repositories provide the communication channel on potential problems and feature requests, as well as contributions from developers. The experimental data and OpenNFT performance were analyzed in MATLAB using custom scripts and available functions. The figures were prepared in MATLAB and Adobe Illustrator (Adobe Inc., CA, US).

Acknowledgements ND, EP, AN were supported by RFBR 20-31-90113A and 19-29-01235MK grants. TA was supported by the Biotechnology and Biological Sciences Research Council, London (BB/S008314/1).

Author contributions YK – conceived and coordinated the study; all authors contributed to data analyses and/or software tools; ND, YK – wrote the manuscript; all authors critically edited the manuscript.

Data Availability The software repositories and test datasets are openly available at <https://github.com/OpenNFT>. All datasets are available from the corresponding author on reasonable request.

Declarations

Conflict of Interests The authors declare no competing interests.

References

- Afyouni, S., & Nichols, T. E. (2018). Insight and inference for DVARS. *NeuroImage*, *172*, 291–312.
- Alfaro-Almagro, F., Jenkinson, M., Bangerter, N. K., Andersson, J. L. R., Griffanti, L., Douaud, G., Sotiropoulos, S. N., Jbabdi, S., Hernandez-Fernandez, M., Vallee, E., Vidaurre, D., Webster, M., McCarthy, P., Rorden, C., Daducci, A., Alexander, D. C., Zhang, H., Dragonu, I., Matthews, P. M., ... Smith, S. M. (2018). Image processing and Quality Control for the first 10,000 brain imaging datasets from UK Biobank. *NeuroImage*, *166*, 400–424.
- Ashburner, J. (2007). A fast diffeomorphic image registration algorithm. *NeuroImage*, *38*, 95–113.
- Astrakas, L. G., Kallistis, N. S., & Kalef-Ezra, J. A. (2016). Technical Note: Independent component analysis for quality assurance in functional MRI. *Medical Physics*, *43*, 983–992.
- Bagarinao, E., Matsuo, K., Nakai, T., & Sato, S. (2003). Estimation of general linear model coefficients for real-time application. *NeuroImage*, *19*, 422–429.
- Bagarinao, E., Nakai, T., & Tanaka, Y. (2006). Real-time functional MRI: Development and emerging applications. *Magnetic Resonance in Medical Sciences*, *5*, 157–165.
- Basilio, R., Garrido, G. J., Sato, J. R., Hoefle, S., Melo, B. R., Pamplona, F. A., Zahn, R., & Moll, J. (2015). FRIEND Engine Framework: A real time neurofeedback client-server system for neuroimaging studies. *Frontiers in Behavioral Neuroscience*, *9*, 3.
- Benigno, G.B., Menon, R.S., Serrai, H. (2021). Schrodinger filtering: a precise EEG despiking technique for EEG-fMRI gradient artifact. *NeuroImage* *226*, 117525.
- Bolton, T.A.W., Kebets, V., Glerean, E., Zöllner, D., Li, J., Yeo, B.T.T., Caballero-Gaudes, C., Van De Ville, D. (2020). Agito ergo sum:

- Correlates of spatio-temporal motion characteristics during fMRI. *NeuroImage*, 209, 116433.
- Cox, R. W., & Jesmanowicz, A. (1999). Real-time 3D image registration for functional MRI. *Magnetic Resonance in Medicine*, 42, 1014–1018.
- DeDora, D.J., Nedic, S., Katti, P., Arnab, S., Wald, L.L., Takahashi, A., Van Dijk, K.R.A., Strey, H.H., Mujica-Parodi, L.R. (2016). Signal Fluctuation Sensitivity: An Improved Metric for Optimizing Detection of Resting-State fMRI Networks. *Front Neuroscience* 10
- Diedrichsen, J., & Shadmehr, R. (2005). Detecting and adjusting for artifacts in fMRI time series data. *NeuroImage*, 27, 624–634.
- Dosenbach, N. U. F., Koller, J. M., Earl, E. A., Miranda-Dominguez, O., Klein, R. L., Van, A. N., Snyder, A. Z., Nagel, B. J., Nigg, J. T., Nguyen, A. L., Wesevich, V., Greene, D. J., & Fair, D. A. (2017). Real-time motion analytics during brain MRI improve data quality and reduce costs. *NeuroImage*, 161, 80–93.
- Esteban, O., Birman, D., Schaer, M., Koyejo, O.O., Poldrack, R.A., Gorgolewski, K.J. (2017). MRIQC: Advancing the automatic prediction of image quality in MRI from unseen sites. *PLoS One* 12, e0184661.
- Fair, D.A., Miranda-Dominguez, O., Snyder, A.Z., Perrone, A., Earl, E.A., Van, A.N., Koller, J.M., Feczko, E., Tisdall, M.D., van der Kouwe, A., Klein, R.L., Mirro, A.E., Hampton, J.M., Adeyemo, B., Laumann, T.O., Gratton, C., Greene, D.J., Schlaggar, B.L., Hagler, D.J., Jr., Watts, R., Garavan, H., Barch, D.M., Nigg, J.T., Petersen, S.E., Dale, A.M., Feldstein-Ewing, S.W., Nagel, B.J., Dosenbach, N.U.F. (2020). Correction of respiratory artifacts in MRI head motion estimates. *NeuroImage* 208, 116400.
- Friedman, L., & Glover, G. H. (2006). Report on a multicenter fMRI quality assurance protocol. *Journal of Magnetic Resonance Imaging*, 23, 827–839.
- Friston, K. J., Harrison, L., & Penny, W. (2003). Dynamic causal modelling. *NeuroImage*, 19, 1273–1302.
- Geissler, A., Gartus, A., Foki, T., Tahamtan, A. R., Beisteiner, R., & Barth, M. (2007). Contrast-to-noise ratio (CNR) as a quality parameter in fMRI. *Journal of Magnetic Resonance Imaging*, 25, 1263–1270.
- Glover, G. H., Li, T. Q., & Ress, D. (2000). Image-based method for retrospective correction of physiological motion effects in fMRI: RETROICOR. *Magnetic Resonance in Medicine*, 44, 162–167.
- Goto, M., Abe, O., Miyati, T., Yamasue, H., Gomi, T., & Takeda, T. (2016). Head Motion and Correction Methods in Resting-state Functional MRI. *Magnetic Resonance in Medical Sciences*, 15, 178–186.
- Greve, D. N., Mueller, B. A., Liu, T., Turner, J. A., Voyvodic, J., Yetter, E., Diaz, M., McCarthy, G., Wallace, S., Roach, B. J., Ford, J. M., Mathalon, D. H., Calhoun, V. D., Wible, C. G., Brown, G. G., Potkin, S. G., & Glover, G. (2011). A novel method for quantifying scanner instability in fMRI. *Magnetic Resonance in Medicine*, 65, 1053–1061.
- Heunis, S., Lamerichs, R., Zinger, S., Caballero-Gaudes, C., Jansen, J. F. A., Aldenkamp, B., & Breeuwer, M. (2020). Quality and denoising in real-time functional magnetic resonance imaging neurofeedback: A methods review. *Human Brain Mapping*, 41, 3439–3467.
- Jenkinson, M., Beckmann, C. F., Behrens, T. E. J., Woolrich, M. W., & Smith, S. M. (2012). *FSL*. *Neuroimage*, 62, 782–790.
- Kasper, L., Bollmann, S., Diaconescu, A. O., Hutton, C., Heinzle, J., Iglesias, S., Hauser, T. U., Sebold, M., Manjaly, Z. M., Pruessmann, K. P., & Stephan, K. E. (2017). The PhysIO Toolbox for Modeling Physiological Noise in fMRI Data. *Journal of Neuroscience Methods*, 276, 56–72.
- Kopel, R., Sladky, R., Laub, P., Koush, Y., Robineau, F., Hutton, C., Weiskopf, N., Vuilleumier, P., Van De Ville, D., & Scharnowski, F. (2019). No time for drifting: Comparing performance and applicability of signal detrending algorithms for real-time fMRI. *NeuroImage*, 191, 421–429.
- Koush, Y., Ashburner, J., Prilepin, E., Sladky, R., Zeidman, P., Bibikov, S., Scharnowski, F., Nikonorov, A., & De Ville, D. V. (2017a). OpenNFT: An open-source Python/Matlab framework for real-time fMRI neurofeedback training based on activity, connectivity and multivariate pattern analysis. *NeuroImage*, 156, 489–503.
- Koush, Y., Ashburner, J., Prilepin, E., Sladky, R., Zeidman, P., Bibikov, S., Scharnowski, F., Nikonorov, A., & Van De Ville, D. (2017b). Real-time fMRI data for testing OpenNFT functionality. *Data in Brief*, 14, 344–347.
- Koush, Y., Meskaldji, D.-E., Pichon, S., Rey, G., Rieger, S. W., Linden, D. E., Van De Ville, D., Vuilleumier, P., & Scharnowski, F. (2017c). Learning control over emotion networks through connectivity-based neurofeedback. *Cerebral Cortex*, 27, 1193–1202.
- Koush, Y., Rosa, M.J., Robineau, F., Heinen, K., S, W.R., Weiskopf, N., Vuilleumier, P., Van De Ville, D., Scharnowski, F. (2013). Connectivity-based neurofeedback: dynamic causal modeling for real-time fMRI. *NeuroImage* 81, 422-430.
- Koush, Y., Zvyagintsev, M., Dyck, M., Mathiak, K. A., & Mathiak, K. (2012). Signal quality and Bayesian signal processing in neurofeedback based on real-time fMRI. *NeuroImage*, 59, 478–489.
- Krylova, M., Skouras, S., Razi, A., Nicholson, A. A., Karner, A., Steyerl, D., Boukrina, O., Rees, G., Scharnowski, F., & Koush, Y. (2021). Progressive modulation of resting-state brain activity during neurofeedback of positive-social emotion regulation networks. *Science and Reports*, 11, 23363.
- Lancaster, J. L., Woldorff, M. G., Parsons, L. M., Liotti, M., Freitas, E. S., Rainey, L., Kochunov, P. V., Nickerson, D., Mikiten, S. A., & Fox, P. T. (2000). Automated Talairach Atlas labels for functional brain mapping. *Human Brain Mapping*, 10, 120–131.
- Lorenz, R., Monti, R.P., Hampshire, A., Koush, Y., Anagnostopoulos, C., Faisal, A.A., Sharp, D., Montana, G., Leech, R., Violante, I.R. (2016). Towards tailoring non-invasive brain stimulation using real-time fMRI and Bayesian optimization. *2016 6th International Workshop on Pattern Recognition in Neuroimaging (Prni)*, 49–52.
- Lu, W., Dong, K., Cui, D., Jiao, Q., & Qiu, J. (2019). Quality assurance of human functional magnetic resonance imaging: A literature review. *Quantitative Imaging in Medicine and Surgery*, 9, 1147–1162.
- MacInnes, J. J., Adcock, R. A., Stocco, A., Prat, C. S., Rao, R. P. N., & Dickerson, K. C. (2020). Pyneal: Open Source Real-Time fMRI Software. *Frontiers in Neuroscience*, 14, 900.
- Maziero, D., Rondinoni, C., Marins, T., Stenger, V.A., Ernst, T. (2020). Prospective motion correction of fMRI: Improving the quality of resting state data affected by large head motion. *NeuroImage* 212.
- Misaki, M., Barzigar, N., Zotev, V., Phillips, R., Cheng, S., & Bodurka, J. (2015). Real-time fMRI processing with physiological noise correction - Comparison with off-line analysis. *Journal of Neuroscience Methods*, 256, 117–121.
- Murphy, K., Bodurka, J., & Bandettini, P. A. (2007). How long to scan? The relationship between fMRI temporal signal to noise ratio and necessary scan duration. *NeuroImage*, 34, 565–574.
- Nakai, T., Bagarinao, E., Matsuo, K., Ohgami, Y., & Kato, C. (2006). Dynamic monitoring of brain activation under visual stimulation using fMRI—the advantage of real-time fMRI with sliding window GLM analysis. *Journal of Neuroscience Methods*, 157, 158–167.
- Parkes, L., Fulcher, B., Yucel, M., & Fornito, A. (2018). An evaluation of the efficacy, reliability, and sensitivity of motion correction strategies for resting-state functional MRI. *NeuroImage*, 171, 415–436.
- Patel, A. X., Kundu, P., Rubinov, M., Jones, P. S., Vertes, P. E., Ersche, K. D., Suckling, J., & Bullmore, E. T. (2014). A wavelet method for modeling and despiking motion artifacts from resting-state fMRI time series. *NeuroImage*, 95, 287–304.
- Penny, W. D., Stephan, K. E., Mechelli, A., & Friston, K. J. (2004). Comparing dynamic causal models. *NeuroImage*, 22, 1157–1172.

- Posse, S., Ackley, E., Mutihac, R., Rick, J., Shane, M., Murray-Krezan, C., Zaitsev, M., & Speck, O. (2012). Enhancement of temporal resolution and BOLD sensitivity in real-time fMRI using multi-slab echo-volumar imaging. *NeuroImage*, *61*, 115–130.
- Power, J. D., Barnes, K. A., Snyder, A. Z., Schlaggar, B. L., & Petersen, S. E. (2012). Spurious but systematic correlations in functional connectivity MRI networks arise from subject motion. *NeuroImage*, *59*, 2142–2154.
- Power, J. D., Mitra, A., Laumann, T. O., Snyder, A. Z., Schlaggar, B. L., & Petersen, S. E. (2014). Methods to detect, characterize, and remove motion artifact in resting state fMRI. *NeuroImage*, *84*, 320–341.
- Ros, T., Enriquez-Geppert, S., Zotev, V., Young, K.D., Wood, G., Whitfield-Gabrieli, S., Wan, F., Vuilleumier, P., Vialatte, F., Van De Ville, D., Todder, D., Surmeli, T., Sulzer, J.S., Strehl, U., Sterman, M.B., Steiner, N.J., Sorger, B., Soekadar, S.R., Sitaram, R., Sherlin, L.H., Schonenberg, M., Scharnowski, F., Schabus, M., Rubia, K., Rosa, A., Reiner, M., Pineda, J.A., Paret, C., Ossadtchi, A., Nicholson, A.A., Nan, W., Minguez, J., Micoulaud-Franchi, J.A., Mehler, D.M.A., Luhrs, M., Lubar, J., Lotte, F., Linden, D.E.J., Lewis-Peacock, J.A., Lebedev, M.A., Lanius, R.A., Kubler, A., Kranczioch, C., Koush, Y., Konicar, L., Kohl, S.H., Kober, S.E., Klados, M.A., Jeunet, C., Janssen, T.W.P., Huster, R.J., Hoedlmoser, K., Hirschberg, L.M., Heunis, S., Hendler, T., Hampson, M., Guggisberg, A.G., Guggenberger, R., Gruzelier, J.H., Gobel, R.W., Gninenko, N., Gharabaghi, A., Frewen, P., Fovet, T., Fernandez, T., Escolano, C., Ehls, A.C., Drechsler, R., Christopher deCharms, R., Debener, S., De Ridder, D., Davelaar, E.J., Congedo, M., Cavazza, M., Breteler, M.H.M., Brandeis, D., Bodurka, J., Birbaumer, N., Bazanova, O.M., Barth, B., Bamidis, P.D., Auer, T., Arns, M., Thibault, R.T. (2020). Consensus on the reporting and experimental design of clinical and cognitive-behavioural neurofeedback studies (CRED-nf checklist). *Brain*, *143*, 1674–1685.
- Sato, J.R., Basilio, R., Paiva, F.F., Garrido, G.J., Bramati, I.E., Bado, P., Tovar-Moll, F., Zahn, R., Moll, J. (2013). Real-time fMRI pattern decoding and neurofeedback using FRIEND: an FSL-integrated BCI toolbox. *PLoS One* *8*, e81658.
- Satterthwaite, T. D., Elliott, M. A., Gerraty, R. T., Ruparel, K., Loughead, J., Calkins, M. E., Eickhoff, S. B., Hakonarson, H., Gur, R. C., Gur, R. E., & Wolf, D. H. (2013). An improved framework for confound regression and filtering for control of motion artifact in the preprocessing of resting-state functional connectivity data. *NeuroImage*, *64*, 240–256.
- Scheinost, D., Papademetris, X., & Constable, R. T. (2014). The impact of image smoothness on intrinsic functional connectivity and head motion confounds. *NeuroImage*, *95*, 13–21.
- Shirer, W. R., Ryali, S., Rykhlevskaia, E., Menon, V., & Greicius, M. D. (2012). Decoding Subject-Driven Cognitive States with Whole-Brain Connectivity Patterns. *Cerebral Cortex*, *22*, 158–165.
- Simmons, A., Moore, E., & Williams, S. C. R. (1999). Quality Control for Functional Magnetic Resonance Imaging Using Automated Data Analysis and Shewhart Charting. *Magnetic Resonance in Medicine*, *41*, 1274–1278.
- Stöcker, T., Schneider, F., Klein, M., Habel, U., Kellermann, T., Zilles, K., & Shah, N. J. (2005). Automated quality assurance routines for fMRI data applied to a multicenter study. *Human Brain Mapping*, *25*, 237–246.
- Triantafyllou, C., Polimeni, J. R., & Wald, L. L. (2011). Physiological noise and signal-to-noise ratio in fMRI with multi-channel array coils. *NeuroImage*, *55*, 597–606.
- van der Zwaag, W., Marques, J. P., Kober, T., Glover, G., Gruetter, R., & Krueger, G. (2012). Temporal SNR characteristics in segmented 3D-EPI at 7T. *Magnetic Resonance in Medicine*, *67*, 344–352.
- Van Dijk, K. R., Sabuncu, M. R., & Buckner, R. L. (2012). The influence of head motion on intrinsic functional connectivity MRI. *NeuroImage*, *59*, 431–438.
- Weiskopf, N., Sitaram, R., Josephs, O., Veit, R., Scharnowski, F., Goebel, R., Birbaumer, N., Deichmann, R., & Mathiak, K. (2007). Real-time functional magnetic resonance imaging: Methods and applications. *Magnetic Resonance Imaging*, *25*, 989–1003.
- Welford, B. P. (1962). Note on a Method for Calculating Corrected Sums of Squares and Products. *Technometrics*, *4*, 419–420.
- Welvaert, M., Rosseel, Y. (2013). On the definition of signal-to-noise ratio and contrast-to-noise ratio for FMRI data. *PLoS One* *8*, e77089.
- Wu, D. H., Lewin, J. S., & Duerk, J. L. (1997). Inadequacy of motion correction algorithms in functional MRI: Role of susceptibility-induced artifacts. *Journal of Magnetic Resonance Imaging*, *7*, 365–370.
- Zilverstand, A., Sorger, B., Slaats-Willemse, D., Kan, C.C., Goebel, R., Buitelaar, J.K. (2017). fMRI Neurofeedback Training for Increasing Anterior Cingulate Cortex Activation in Adult Attention Deficit Hyperactivity Disorder. An Exploratory Randomized, Single-Blinded Study. *PLoS One* *12*, e0170795.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Nikita Davydov^{1,2,3} · Lucas Peek⁴ · Tibor Auer⁵ · Evgeny Prilepin¹ · Nicolas Gninenko^{6,7} · Dimitri Van De Ville^{6,7} · Artem Nikonov^{2,3} · Yury Koush⁸ 

✉ Yury Koush
yurykoush@gmail.com

¹ Aligned Research Group, Los Gatos, USA

² Samara National Research University, Samara, Russia

³ Image Processing Systems Institute, Russian Academy of Science, Samara, Russia

⁴ Department of Fundamental Neurosciences, University of Geneva, Geneva, Switzerland

⁵ School of Psychology, University of Surrey, Guildford, UK

⁶ Institute of Bioengineering, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

⁷ Department of Radiology and Medical Informatics, University of Geneva, Geneva, Switzerland

⁸ Department of Radiology and Medical Imaging, Yale University, New Haven, USA