SoftwareX 17 (2022) 100933

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

DRT: A new toolbox for the Standard EEG Data Structure in large-scale EEG applications

Li Dong[a,b,c,*,1], Yufan Zhang[a,1], Lingling Zhao[a], Ting Zheng[a], Weidong Wang[d], Jianfu Li[a,b,c], Diankun Gong[a,b,c], Tiejun Liu[a,b,c], Dezhong Yao[a,b,c,e,*]

[a] *The Clinical Hospital of Chengdu Brain Science Institute, MOE Key Lab for Neuroinformation, School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, Sichuan, China*
[b] *Sichuan Institute for Brain Science and Brain-Inspired Intelligence, Chengdu, Sichuan, China*
[c] *Research Unit of NeuroInformation, Chinese Academy of Medical Sciences, Chengdu, Sichuan, China*
[d] *The Affiliated Cancer Hospital, School of Medicine, University of Electronic Science and Technology of China, Chengdu, Sichuan, China*
[e] *School of Electrical Engineering, Zhengzhou University, Zhengzhou, Henan, China*

ARTICLE INFO

Article history:
Received 6 September 2021
Received in revised form 16 November 2021
Accepted 1 December 2021

Keywords:
EEG
Datafile Restructuring Toolbox
Standard EEG Data Structure (SEDS)
Standardization

ABSTRACT

The current evolution of "open neuroscience" has led to an increased amount of research on large-scale electroencephalography (EEG) applications, resulting in large quantities of accumulated EEG data. The batch sharing and processing of these massive EEG data play an important role in EEG studies within or across laboratories and result in an increasing requirement for a standard data file structure for existing EEG data. In this work, a new and more flexible data structure, named the Standard EEG Data Structure (SEDS), was proposed to meet the needs of both small-scale EEG data batch processing in single-site studies and large-scale EEG data sharing and analysis in single-/multisite studies (especially on cloud platforms). Furthermore, two versions (MATLAB and Docker versions) of the EEG Datafile Restructuring Toolbox (DRT) were developed to restructure EEG data files according to the SEDS. The DRT GUI (MATLAB version) dramatically reduces the time required for novice researchers, while the DRT (Docker version) is more efficient for experienced researchers. All materials including SEDS documents, tools, example datasets, etc., are available on the WeBrain website (https://webrain.uestc.edu.cn/) and Wiki (https://github.com/WeCloudHub/DRT). We hope that these two user-friendly toolboxes can make the relatively novel SEDS easier to collaboratively study, especially for applications in large-scale EEG studies.

Code metadata

Current code version	v1.0
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-21-00166
Code Ocean compute capsule	
Legal Code License	BSD 3-Clause
Code versioning system used	Git
Software code languages, tools, and services used	MATLAB, EEGLAB
Compilation requirements, operating environments & dependencies	Windows, Linux, or MacOS
If available Link to developer documentation/manual	https://github.com/WeCloudHub/DRT
Support email for questions	201922140323@std.uestc.edu.cn

1. Motivation and significance

Since the human scalp electroencephalogram (EEG) was first reported by Berger in 1929 [1], the scalp EEG has been a common technique for noninvasively detecting brain activity, with a high

[*] Correspondence to: Xiyuan Ave, Chengdu, Sichuan, 611731, China.
E-mail addresses: Lidong@uestc.edu.cn (Li Dong), dyao@uestc.edu.cn (Dezhong Yao).

[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.softx.2021.100933

temporal resolution and low cost. The scientific report statistics shows that there is an increasing amount of interest in EEG due to its irreplaceable value in brain science. The recommendations for open science and the rise of cloud neuroscience [2,3] have further led to more efforts with large-scale EEG applications in the EEG community. In addition, there has been the further accumulation of a number of public and local large EEG datasets, such as the TUH-EEG Corpus dataset with more than 30000 clinical EEGs (https://www.isip.piconepress.com/projects/tuh_eeg/index.shtml) [4] and the "EEG Motor Movement/Imagery Dataset" (https://archive.physionet.org/pn4/eegmmidb/) with 1500 EEGs [5,6]. Sharing and processing these massive EEG datasets play important roles in large-scale brain imaging collaborative studies across laboratories and further increase the requirement for a standard data file structure of existing large-scale EEG data. In addition, in the neuroscience field, there is a growing concern about data replication and reproducibility [7], i.e., whether the original data and analysis results can be well replicated by others [8]. However, many potential issues including various data file structures, confusing file organization and a lack of interpretation of the raw, intermediate and final data may largely decrease the efficiency of data sharing and analysis, as well as data reproducibility. Therefore, a tool of standard EEG data file structure for large-scale EEG sharing and processing (especially using cloud high-performance computing (HPC) facilities) is required, which is critical for establishing broader collaborative large-scale EEG studies across laboratories.

Due to the versatile advantages of EEG, including light weight, high compatibility with environments and systems, low cost, wearability, being wireless, etc., the field of EEG applications is broad. Thus, there are far more EEG equipment manufacturers than manufacturers of other noninvasive imaging equipment, and these EEG equipment manufacturers are building different hardware systems with different software and data formats. For example, the EEG data recorded by the Neuroscan EEG system may generate a set of "*.dat, *.dap, and *.rs3" files using Curry 7, while the data recorded by the Brain Product EEG system may generate a set of "*.vhdr, *.vmrk, *.dat" files using the Brain Vision Analyzer. Furthermore, because commercial or free EEG tools always have own proprietary data formats for processing data, the file formats of the intermediate and final EEG data generated by different tools are still different (e.g., EEGLAB [9] can save data in the "*.SET" format, and Curry 7 can save data in the "*.CNT" format). Such diversity of EEG data files perhaps is an impediment to the reuse of data, as well as building large-scale EEG databases for sharing across laboratories. To address the abovementioned issue of EEG data heterogeneity, some efforts have been made in the neuroscience community in recent years. Teeters et al. [10] proposed a common neurophysiology data format based on the HDF5 (http://www.hdfgroup.org/HDF5); this format uses "HDF5 groups" for the directories and "HDF5 datasets" corresponding to the files to store arbitrary array-type data. However, this data standardization is mainly used for the data of cellular electrophysiology and optical imaging experiments and not scalp EEGs, and a graphical utility named HDFView must be used to browse files while using HDF5. Bigdely-Shamlo et al. developed a "containerized" approach, named the EEG Study Schema (ESS) [11], to organize EEG data and metadata using a standardized file structure and metadata encapsulation schema. The limitation of this approach is the unmet need for easy manual or semiautomated usage, while the users have to manually program MATLAB scripts to import the metadata from semistructured formats, which increases the requirements on the programming skills for novice users. Recently, as an extension to the Brain Imaging Data Structure (BIDS), BIDS-EEG [12] has been proposed for readily organizing and sharing raw EEG data within and between EEG laboratories. The basic definition of BIDS-EEG assumes that each subject has a directory of raw data containing subdirectories for each modality and session. A number of EEG tools and public datasets are supported or organized using this standard. However, BIDS-EEG does have some limitations including the need to support more data formats (4 kinds of formats are supported currently), compatibility for both small-scale EEG batch processing in single-site and large-scale EEG sharing and processing across multisites (especially on cloud platforms), as well as relatively low flexibility for "pure" EEGs (e.g., for separated EEG data without other modalities).

In this work, a new and more flexible data structure, named the Standard EEG Data Structure (SEDS), was proposed to meet both the needs of small-scale EEG data batch processing in single-site studies and large-scale EEG data sharing and analysis in single-/multisite studies (especially on cloud platforms). The structure may increase the reproducibility and extensibility of EEG data. Furthermore, two versions of the EEG Datafile Restructuring Toolbox (DRT) were developed to restructure EEG data files according to the SEDS. In addition, because the quality assessment (QA) of the raw EEG data is an important issue for the reproducibility of EEG results, a QA module was integrated into the DRT as an optional function when converting data to the SEDS.

2. Software description

The basic definition of the SEDS assumes that each subject (person-time) has a folder or a zip file containing only one sample of raw/processed EEG data accompanied by metadata files (i.e., descriptive files containing basic information on data acquisition, events and electrodes). The specification of the SEDS in detail can be seen in the supplementary materials, and comparisons between SEDS and current data file structures are showed in Table 1. To conveniently convert the raw EEG data into the SEDS, the EEG Datafile Restructuring Toolbox (DRT) is developed to provide a feasible solution for large-scale EEG sharing and processing. The DRT is designed with the following guidelines: (1) it is easy for nonprogrammers to install and use it, and (2) it can be extended by developers and combined with other EEG tools.

2.1. Software architecture

The architecture of the DRT is shown in Fig. 1. First, raw EEG data with different data formats are imported using the DRT, and configuration information including the sampling rate, channel locations, events, etc., are automatically detected from the raw data. If the raw data such as ".txt" and ".mat" files do not contain configuration information, essential information on the sampling rate and channel locations can be imported, if needed. Next, there are three workflows that can be used. (1) "SEDS" only: all raw data files are restructured to the SEDS. (2) "SEDS->QA": all raw data files are restructured to the SEDS and assessed using the QA module. (3) "QA" only: all raw data files are assessed using the QA module. Finally, if applicable, the outputs of the SEDS/QA can be filtered according to the key words using the "Capture Filter". The current supported data formats are shown in Table 2. Materials relative to the SEDS are available on the WeBrain website (https://webrain.uestc.edu.cn/) and Wiki (https://github.com/WeCloudHub/DRT). These materials include the specification document of the SEDS, the DRT user manual, example datasets, and other information relative to the release, development and contributions of the DRT. In addition, the usage of the SEDS on the WeBrain cloud platform is user-friendly, and the workflow contains (1) using the DRT to convert raw EEG data files into the SEDS zip files, (2) uploading these zip files to WeBrain, and (3) managing and analyzing these data on WeBrain.

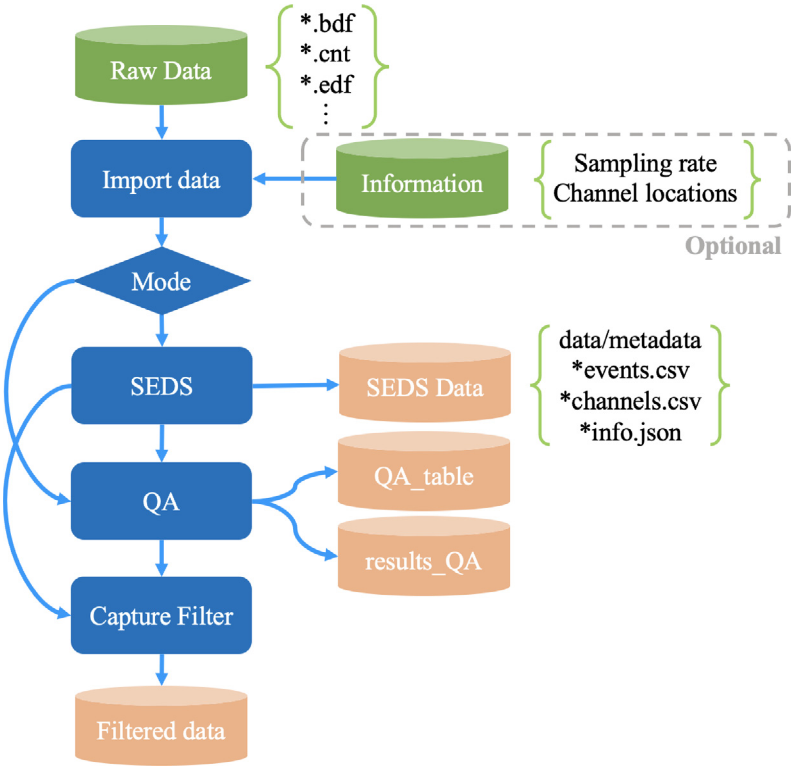


Fig. 1. Overview of the DRT architecture. First, raw data are imported using the DRT, and information including sampling rates, channel locations, events, etc., are automatically detected. If the raw data such as ".txt" and ".mat" files do not contain configuration information, essential information on the sampling rate and channel locations can be imported, if needed. Next, three kinds of workflows can be used. (1) "SEDS" only: all raw data files are standardized into the SEDS. (2) "SEDS->QA": all raw data files are standardized into the SEDS and assessed using the QA module. (3) "QA" only: all raw data files are assessed using the QA module. Finally, if applicable, outputs can be filtered according to the key words using the "Capture Filter".

Table 1
Comparisons between SEDS and current data file structures.

	SEDS	BIDS-EEG	ESS	HDF5
File structure	✓	✓	✓	✓
No. of supported EEG formats	9	4	–	–
Descriptive file	json	json	xml	–
Extensibility	✓	✓	✓	✓
Modality	EEG	EEG sMRI	EEG Behavioral data	Cellular electrophysiology Optical imaging
Cloud support	✓	✓	–	–
Tool version	MATLAB Docker	MATLAB	MATLAB	Java/Fortran/C++
Batch processing	✓	✓	✓	✓
Capture filter	✓	–	–	–
Meta information	Automatic Manual	Manual	Manual	–

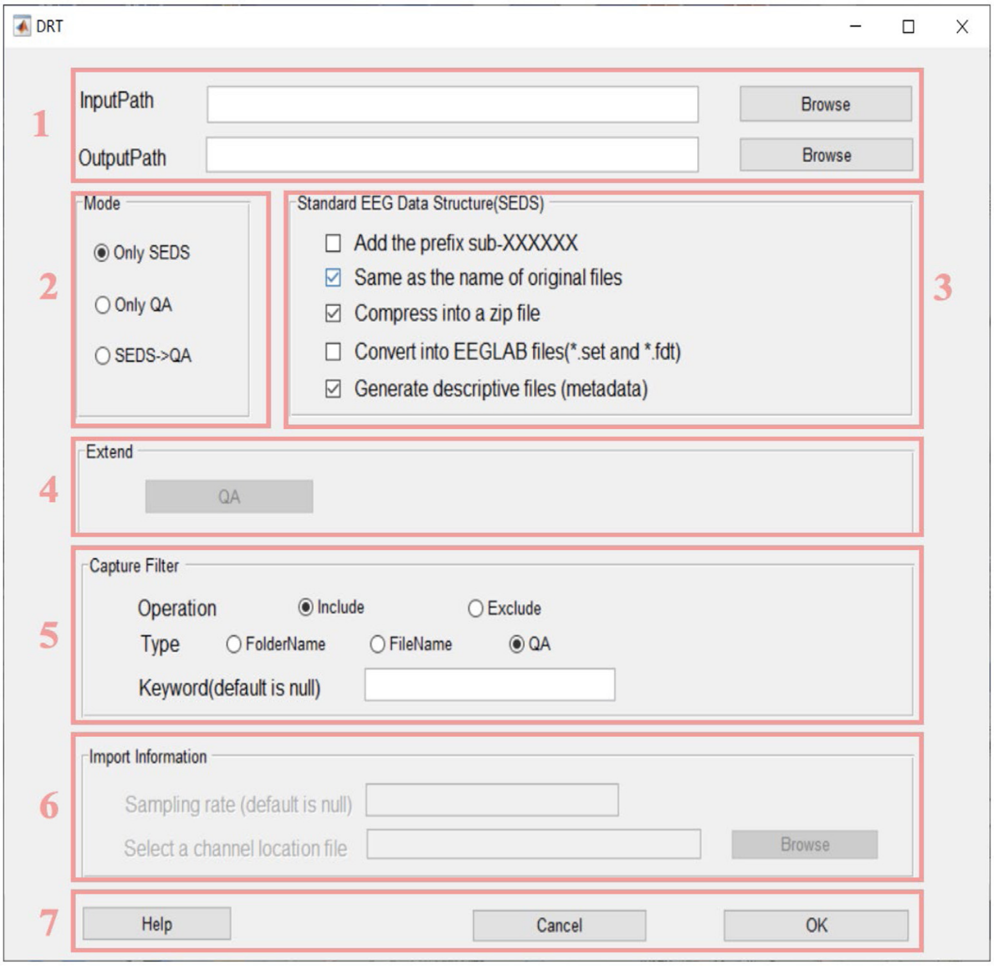
Table 2
Current EEG data file formats supported by the DRT.

Manufacturer/Tool/Data	File formats
ASCII/Float	*.txt
Biosemi	*.bdf
BrainProducts/BrainVision	*.vhdr/*.vmrk/*.dat | *.vhdr/*.vmrk/*.eeg
Curry 6/7	*.dat/*.dap/*.rs3
EEGLAB	*.set/*.fdt
European Data Format	*.edf | *.edf+
General Data Format	*.gdf
MATLAB	*.mat | *.dat
NeuroScan	*.cnt | *.eeg | *.avg

2.2. Software usage

The DRT was developed using MATLAB 2015b (MathWorks, Inc., Natick, MA) [13] and the suggested MATLAB version is R2014b or above. The latest MATLAB version (v1.0) of the DRT is available on http://webrain.uestc.edu.cn/. The installation steps of the DRT are as follows: (1) download and unzip the DRT toolbox; (2) click the "Set Path" button on the home interface of MATLAB and click the "Add folders" button to add the path of the DRT in MATLAB; and (3) enter the "DRT" in the MATLAB command line window, and the popup interface of the DRT will be displayed (Fig. 2). The usage of the DRT consists of the following steps:

(1) Click "Browse" to select the input path of the folder containing the EEG data files (e.g., "/root/Desktop/RawData"). Click "Browse" to select the output path of the directory. The output directory should be created first (Fig. 2(1)).
(2) Select one mode (Fig. 2(2), "Only SEDS", "Only QA" or "SEDS->QA").
(3) Set the parameters (Fig. 2(3–6)) corresponding to the running mode. The detailed descriptions of the options can be found in Section 3 of the DRT user manual.



InputPath

OutputPath

Mode

● Only SEDS

○ Only QA

○ SEDS->QA

Standard EEG Data Structure(SEDS)

☐ Add the prefix sub-XXXXXX

☑ Same as the name of original files

☑ Compress into a zip file

☐ Convert into EEGLAB files(*.set and *.fdt)

☑ Generate descriptive files (metadata)

Extend

Capture Filter

Operation ● Include ○ Exclude

Type ○ FolderName ○ FileName ● QA

Keyword(default is null)

Import Information

Sampling rate (default is null)

Select a channel location file

Fig. 2. Popup interface of the DRT. The MATLAB version is equipped with a graphical user interface that is easy to use for beginners. The interface panels consist of the following: (1) input/output path selection, (2) mode selection, (3) parameter settings of SEDS, (4) extend QA module, (5) capture filter, (6) import essential information of sampling rate and channel locations, and (7) "Help", "Cancel" and "OK" buttons for tool instruction, quit DRT and run DRT, respectively.

(4) Click the "OK" button (Fig. 2(7)) to run the program. All the outputs will be located in the output directory on the local computer when the process is completed.

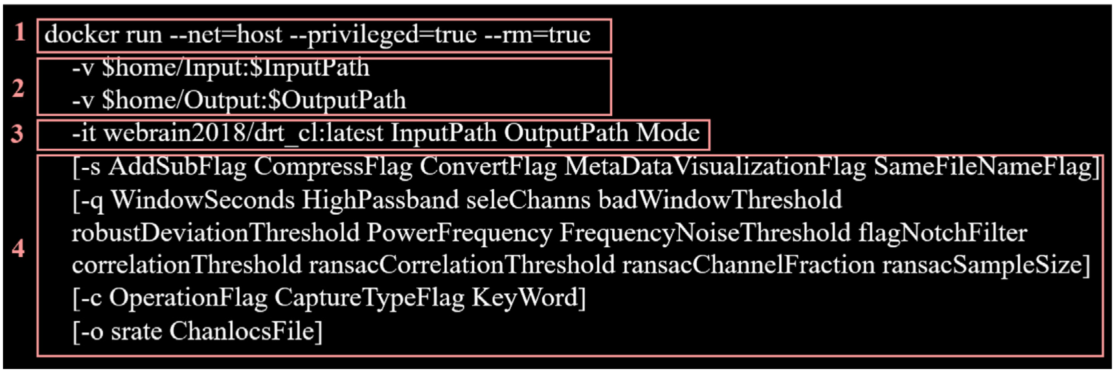
Docker [14] is a lightweight container technology that allows users to package any applications, configurations and operating environment into a standard image that can run on any environment. Regarding the resource isolation and allocation of the operating system, Docker technology is more portable and efficient than traditional virtual machine technology. The Docker version of the DRT is stored in Docker Hub (https://hub.docker.com/), which is a cloud-based registry service of Docker. The Docker Hub provides workflow automation throughout the development pipeline based on Docker images. The Docker-based version of the DRT is designed for experienced researchers. After installing the Docker engine, the latest version of the DRT can be run on the Windows PowerShell or terminal on a Linux system by running the following command:

```
$ docker pull docker.io/webrain2018/drt_cl
```

The following Docker command (Fig. 3) can start the container that is the instantiation of a DRT image and run/bin/bash interactively at your terminal (due to the -i and -t flags). The Docker engine attaches an external read–write filesystem to the container for storage (due to the -v flag), allowing native data interaction with the container in the local filesystem. The image name (webrain2018/drt_cl: latest) is followed by configuration parameters, among which the first three parameters are required, representing the input file path, output file path and DRT function mode. Fig. 3(4) show the optional configuration parameters of the container. The -s, -q, -c and -o flags represent the SEDS, QA, capture filter and import information functions of the DRT, respectively, and the flags are followed by the configuration parameters of the corresponding function.

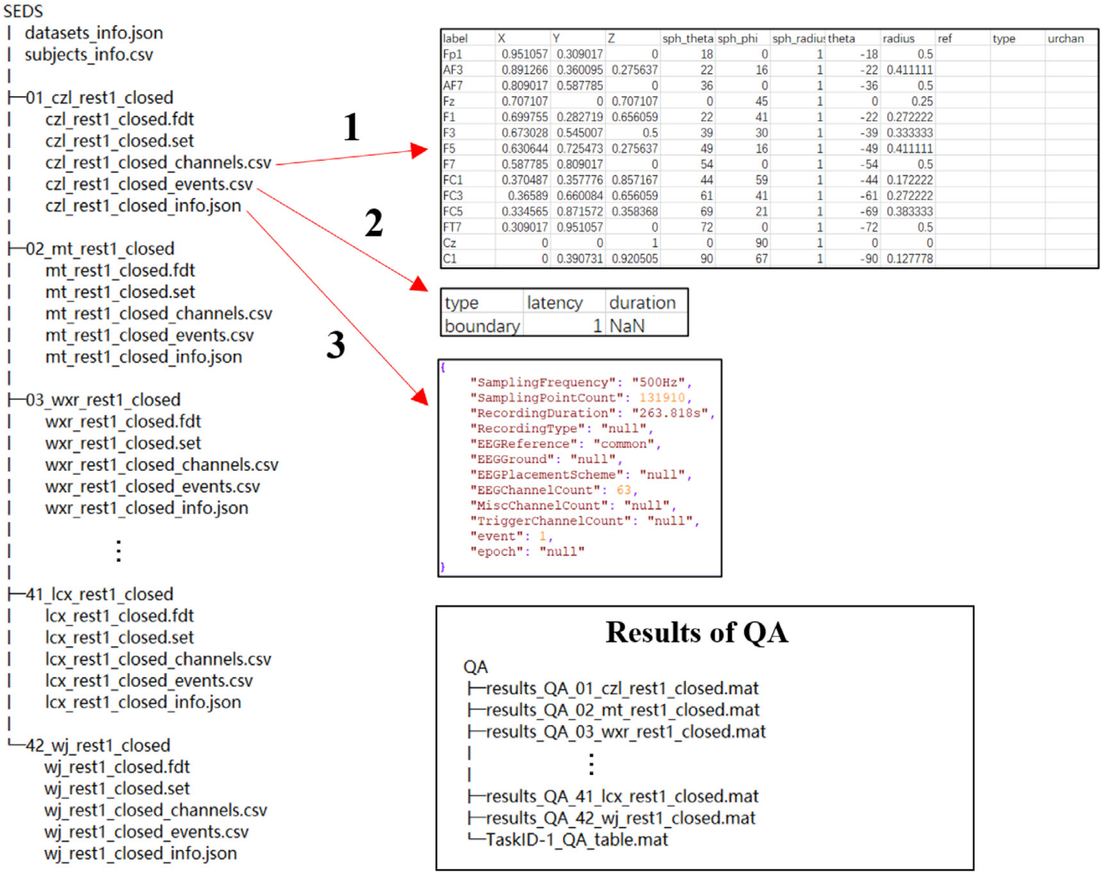
3. Illustrative examples

Two previous datasets were used to demonstrate the functionalities of the SEDS tools. For dataset 1, a resting-state (eyes closed) EEG dataset was collected from 42 healthy subjects through 63 electrodes located at the standard 10–20 scalp in a previous study [15]. Written informed consent was obtained from all subjects before the experiment, and the study was approved by the local Ethics Committee of the University of Electronic Science and Technology of China. However, the raw data are in BrainVision format (*.vhdr, *.vmrk, *.dat) (see the section 1 of supplementary material), and the EEG-BIDS advocated internationally does not currently support this format. Here, we standardized the raw data into the SEDS via the DRT, which supports more EEG data formats. The results are shown in Fig. 4. In the directory of the output path, the subject information list file ("subjects_info.csv"), the data information text file ("datasets_info.json") and the folders corresponding to each data example were generated automatically. Each data case was associated with "*events.csv", "*channels.csv" and "*info.json" files to describe the detailed information of the



```
docker run --net=host --privileged=true --rm=true
   -v $home/Input:$InputPath
   -v $home/Output:$OutputPath
   -it webrain2018/drt_cl:latest InputPath OutputPath Mode
   [-s AddSubFlag CompressFlag ConvertFlag MetaDataVisualizationFlag SameFileNameFlag]
   [-q WindowSeconds HighPassband seleChanns badWindowThreshold
   robustDeviationThreshold PowerFrequency FrequencyNoiseThreshold flagNotchFilter
   correlationThreshold ransacCorrelationThreshold ransacChannelFraction ransacSampleSize]
   [-c OperationFlag CaptureTypeFlag KeyWord]
   [-o srate ChanlocsFile]
```

Fig. 3. Command of the Docker version of the DRT. (1) Basic command line for running Docker images. (2) "$ home" is the path of the localhost, and "$ InputPath" and "$ OutputPath" need to be consistent with "InputPath" and "OutputPath", respectively. (3) "webrain2018/drt_cl" is the Docker image of the DRT, and the following three parameters cannot be defaulted. (4) Optional; refer to Section 3 of supplementary material for the details.



```
SEDS
|  datasets_info.json
|  subjects_info.csv
|
├─01_czl_rest1_closed
|      czl_rest1_closed.fdt
|      czl_rest1_closed.set
|      czl_rest1_closed_channels.csv
|      czl_rest1_closed_events.csv
|      czl_rest1_closed_info.json
|
├─02_mt_rest1_closed
|      mt_rest1_closed.fdt
|      mt_rest1_closed.set
|      mt_rest1_closed_channels.csv
|      mt_rest1_closed_events.csv
|      mt_rest1_closed_info.json
|
├─03_wxr_rest1_closed
|      wxr_rest1_closed.fdt
|      wxr_rest1_closed.set
|      wxr_rest1_closed_channels.csv
|      wxr_rest1_closed_events.csv
|      wxr_rest1_closed_info.json
|
|              ⋮
|
├─41_lcx_rest1_closed
|      lcx_rest1_closed.fdt
|      lcx_rest1_closed.set
|      lcx_rest1_closed_channels.csv
|      lcx_rest1_closed_events.csv
|      lcx_rest1_closed_info.json
|
└─42_wj_rest1_closed
       wj_rest1_closed.fdt
       wj_rest1_closed.set
       wj_rest1_closed_channels.csv
       wj_rest1_closed_events.csv
       wj_rest1_closed_info.json
```

label	X	Y	Z	sph_theta	sph_phi	sph_radius	theta	radius	ref	type	urchan
Fp1	0.951057	0.309017	0	18	0	1	-18	0.5			
AF3	0.891266	0.360095	0.275637	22	16	1	-22	0.411111			
AF7	0.809017	0.587785	0	36	0	1	-36	0.5			
Fz	0.707107	0	0.707107	0	45	1	0	0.25			
F1	0.699755	0.282719	0.656059	22	41	1	-22	0.272222			
F3	0.673028	0.545007	0.5	39	30	1	-39	0.333333			
F5	0.630644	0.725473	0.275637	49	16	1	-49	0.411111			
F7	0.587785	0.809017	0	54	0	1	-54	0.5			
FC1	0.370487	0.357776	0.857167	44	59	1	-44	0.172222			
FC3	0.36589	0.660084	0.656059	61	41	1	-61	0.272222			
FC5	0.334565	0.871572	0.358368	69	21	1	-69	0.383333			
FT7	0.309017	0.951057	0	72	0	1	-72	0.5			
Cz	0	0	1	0	90	1	0	0			
C1	0	0.390731	0.920505	90	67	1	-90	0.127778			

type	latency	duration
boundary	1	NaN

```
{
    "SamplingFrequency": "500Hz",
    "SamplingPointCount": 131910,
    "RecordingDuration": "263.818s",
    "RecordingType": "null",
    "EEGReference": "common",
    "EEGGround": "null",
    "EEGPlacementScheme": "null",
    "EEGChannelCount": 63,
    "MiscChannelCount": "null",
    "TriggerChannelCount": "null",
    "event": 1,
    "epoch": "null"
}
```

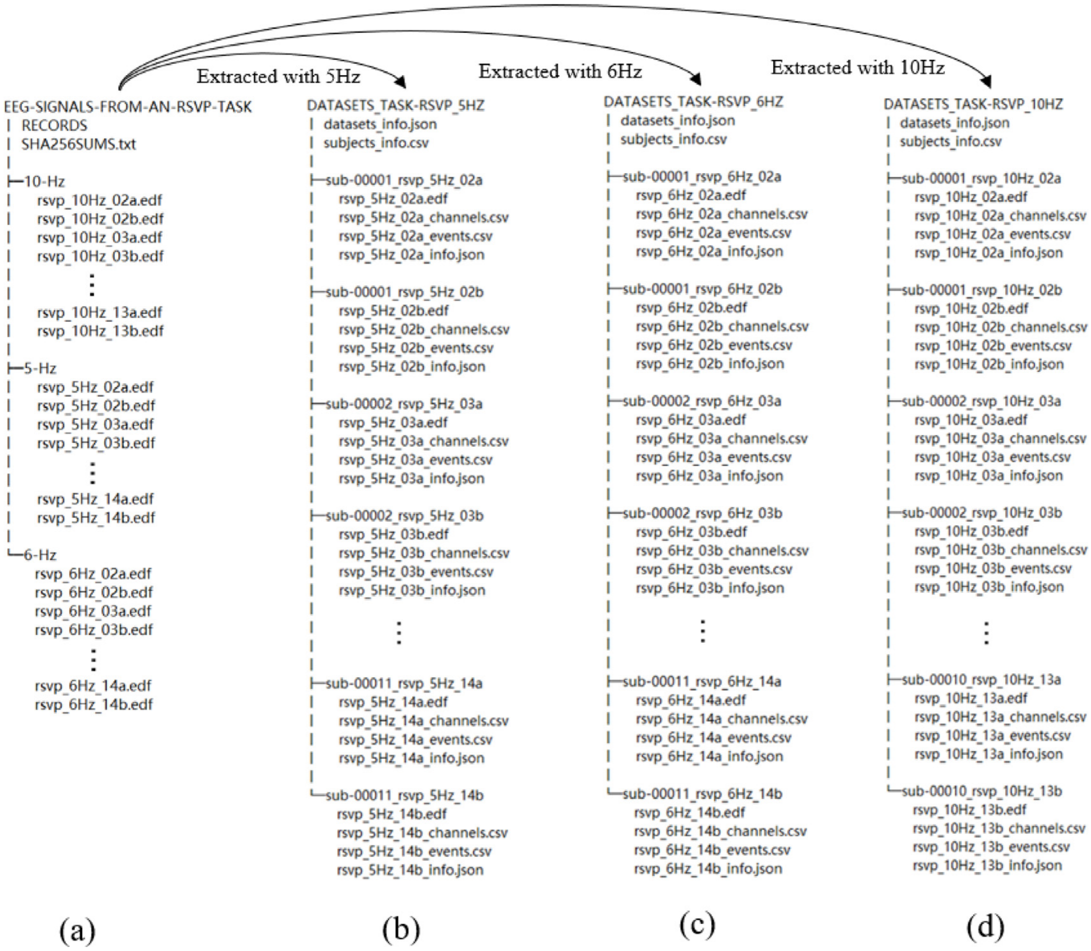
Results of QA

```
QA
├─results_QA_01_czl_rest1_closed.mat
├─results_QA_02_mt_rest1_closed.mat
├─results_QA_03_wxr_rest1_closed.mat
|
|             ⋮
├─results_QA_41_lcx_rest1_closed.mat
├─results_QA_42_wj_rest1_closed.mat
└─TaskID-1_QA_table.mat
```

Fig. 4. Results of dataset 1. The left side is the standardized file tree structure of the raw data of 42 healthy subjects (see the section 1 of supplementary material for the original file tree structure of the raw data). (1) "*channels.csv" records the spatial coordinates of all electrodes. (2) "*events.csv" records the type, latency and duration of all stimuli. (3) "*info.json" describes the data information from the metadata of raw data. The "results_QA*.mat" file of each subject contains the parameters of each step and the QA results. "TaskID-1_QA_table.mat" lists all QA indices of all subjects. All the details and contents are provided in the section 2 of supplementary material.

raw data (automatically extracted from raw data). This approach not only facilitates data sharing and batch processing in a multisite or cloud platform [16] but also improves the readability and repeatability of EEG raw data [17,18], especially for large-scale brain imaging collaborative studies across laboratories. It is worth noting that the SEDS was more flexible and specific for the organization of "pure" EEGs (e.g., for separated EEG data without other modalities), which increased the batch data processing and large-scale data management efficiencies. In addition, the results of the quality assessment (QA) could be calculated via the DRT (shown in the bottom right corner of Fig. 4), generating a "results_QA*.mat" file containing the parameters of each step and the QA results for each subject and a "QA_table.mat" file listing all the main QA indices of all subjects. The section 2 of supplementary material provides the details of the example content of the QA result, and the QA method refers to the QA manual available from "https://webrain.uestc.edu.cn/".

Dataset 2 was collected from public databases (https://www.physionet.org/content/mssvepdb/1.0.0/) that contained EEGs from 11 healthy participants based upon the rapid presentation of



```
EEG-SIGNALS-FROM-AN-RSVP-TASK
|  RECORDS
|  SHA256SUMS.txt
|
├─10-Hz
|     rsvp_10Hz_02a.edf
|     rsvp_10Hz_02b.edf
|     rsvp_10Hz_03a.edf
|     rsvp_10Hz_03b.edf
|          ⋮
|     rsvp_10Hz_13a.edf
|     rsvp_10Hz_13b.edf
|
├─5-Hz
|     rsvp_5Hz_02a.edf
|     rsvp_5Hz_02b.edf
|     rsvp_5Hz_03a.edf
|     rsvp_5Hz_03b.edf
|          ⋮
|     rsvp_5Hz_14a.edf
|     rsvp_5Hz_14b.edf
|
└─6-Hz
      rsvp_6Hz_02a.edf
      rsvp_6Hz_02b.edf
      rsvp_6Hz_03a.edf
      rsvp_6Hz_03b.edf
           ⋮
      rsvp_6Hz_14a.edf
      rsvp_6Hz_14b.edf
```

```
DATASETS_TASK-RSVP_5HZ
|  datasets_info.json
|  subjects_info.csv
|
├─sub-00001_rsvp_5Hz_02a
|     rsvp_5Hz_02a.edf
|     rsvp_5Hz_02a_channels.csv
|     rsvp_5Hz_02a_events.csv
|     rsvp_5Hz_02a_info.json
|
├─sub-00001_rsvp_5Hz_02b
|     rsvp_5Hz_02b.edf
|     rsvp_5Hz_02b_channels.csv
|     rsvp_5Hz_02b_events.csv
|     rsvp_5Hz_02b_info.json
|
├─sub-00002_rsvp_5Hz_03a
|     rsvp_5Hz_03a.edf
|     rsvp_5Hz_03a_channels.csv
|     rsvp_5Hz_03a_events.csv
|     rsvp_5Hz_03a_info.json
|
├─sub-00002_rsvp_5Hz_03b
|     rsvp_5Hz_03b.edf
|     rsvp_5Hz_03b_channels.csv
|     rsvp_5Hz_03b_events.csv
|     rsvp_5Hz_03b_info.json
|
|          ⋮
|
├─sub-00011_rsvp_5Hz_14a
|     rsvp_5Hz_14a.edf
|     rsvp_5Hz_14a_channels.csv
|     rsvp_5Hz_14a_events.csv
|     rsvp_5Hz_14a_info.json
|
└─sub-00011_rsvp_5Hz_14b
      rsvp_5Hz_14b.edf
      rsvp_5Hz_14b_channels.csv
      rsvp_5Hz_14b_events.csv
      rsvp_5Hz_14b_info.json
```

```
DATASETS_TASK-RSVP_6HZ
|  datasets_info.json
|  subjects_info.csv
|
├─sub-00001_rsvp_6Hz_02a
|     rsvp_6Hz_02a.edf
|     rsvp_6Hz_02a_channels.csv
|     rsvp_6Hz_02a_events.csv
|     rsvp_6Hz_02a_info.json
|
├─sub-00001_rsvp_6Hz_02b
|     rsvp_6Hz_02b.edf
|     rsvp_6Hz_02b_channels.csv
|     rsvp_6Hz_02b_events.csv
|     rsvp_6Hz_02b_info.json
|
├─sub-00002_rsvp_6Hz_03a
|     rsvp_6Hz_03a.edf
|     rsvp_6Hz_03a_channels.csv
|     rsvp_6Hz_03a_events.csv
|     rsvp_6Hz_03a_info.json
|
├─sub-00002_rsvp_6Hz_03b
|     rsvp_6Hz_03b.edf
|     rsvp_6Hz_03b_channels.csv
|     rsvp_6Hz_03b_events.csv
|     rsvp_6Hz_03b_info.json
|
|          ⋮
|
├─sub-00011_rsvp_6Hz_14a
|     rsvp_6Hz_14a.edf
|     rsvp_6Hz_14a_channels.csv
|     rsvp_6Hz_14a_events.csv
|     rsvp_6Hz_14a_info.json
|
└─sub-00011_rsvp_6Hz_14b
      rsvp_6Hz_14b.edf
      rsvp_6Hz_14b_channels.csv
      rsvp_6Hz_14b_events.csv
      rsvp_6Hz_14b_info.json
```

```
DATASETS_TASK-RSVP_10HZ
|  datasets_info.json
|  subjects_info.csv
|
├─sub-00001_rsvp_10Hz_02a
|     rsvp_10Hz_02a.edf
|     rsvp_10Hz_02a_channels.csv
|     rsvp_10Hz_02a_events.csv
|     rsvp_10Hz_02a_info.json
|
├─sub-00001_rsvp_10Hz_02b
|     rsvp_10Hz_02b.edf
|     rsvp_10Hz_02b_channels.csv
|     rsvp_10Hz_02b_events.csv
|     rsvp_10Hz_02b_info.json
|
├─sub-00002_rsvp_10Hz_03a
|     rsvp_10Hz_03a.edf
|     rsvp_10Hz_03a_channels.csv
|     rsvp_10Hz_03a_events.csv
|     rsvp_10Hz_03a_info.json
|
├─sub-00002_rsvp_10Hz_03b
|     rsvp_10Hz_03b.edf
|     rsvp_10Hz_03b_channels.csv
|     rsvp_10Hz_03b_events.csv
|     rsvp_10Hz_03b_info.json
|
|          ⋮
|
├─sub-00010_rsvp_10Hz_13a
|     rsvp_10Hz_13a.edf
|     rsvp_10Hz_13a_channels.csv
|     rsvp_10Hz_13a_events.csv
|     rsvp_10Hz_13a_info.json
|
└─sub-00010_rsvp_10Hz_13b
      rsvp_10Hz_13b.edf
      rsvp_10Hz_13b_channels.csv
      rsvp_10Hz_13b_events.csv
      rsvp_10Hz_13b_info.json
```

Fig. 5. Results of dataset 2. (a) File tree structure of the raw data. (b) SEDS of the raw data at 5 Hz experimental conditions. By using "5 Hz" as the keyword for the capture filter, the eligible data files are filtered and standardized into the SEDS. (c) The SEDS of the raw data at 6 Hz experimental conditions. (d) The SEDS of the raw data at 10 Hz experimental conditions.

images through the Rapid Serial Visual Presentation (RSVP) protocol at speeds of 5, 6, and 10 Hz [19]. The experiment was also approved by the local Ethics Committee. For dataset 2, the raw data files were grouped and stored according to the experimental tasks, and each folder contained one type of experimental data (Fig. 5(a)). Fig. 5(b–d) shows the file tree structure of the raw data after DRT collation. Since different public datasets may be organized differently, the batch sharing and processing of a public dataset (a part of or whole data) play important roles in large-scale brain imaging collaborative studies across laboratories, indicating an increasing requirement for a standard data file structure. The SEDS structure perhaps meets this requirement by providing rich machine-readable metadata [16], and it satisfies both small-scale EEG batch processing at a single site and large-scale EEG processing across/within multisites.

4. Conclusion

Based on MATLAB and Docker, two versions of the DRT toolbox were developed to restructure EEG data files according to the proposed Standard EEG Data Structure (SEDS). This development may meet the needs of both small-scale EEG data batch processing in single-site studies and large-scale EEG data sharing and analysis in single-/multisite studies (especially on cloud platforms). The DRT GUI dramatically reduces the time required for novice researchers, while the Docker DRT is more efficient for experienced researchers. We hope that these two user-friendly toolboxes make the relatively novel SEDS easier to study in collaborative studies, especially for applications in large-scale EEG studies.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding information

The present study was financially supported by the National Natural Science Foundation of China (No. 81861128001), the '111' project, China (B12027), the Sichuan Science and Technology Program, China (2018JZ0073) and the CAMS Innovation Fund for Medical Sciences, China (2019-I2M-5-039).

Data availability

The dataset 1 used in this study is available on reasonable request to the corresponding author. The dataset 2 involved in the paper is all publicly available from website: https://www.physionet.org/content/mssvepdb/1.0.0/.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.softx.2021.100933.

References

[1] Berger Hans. Über das Elektrenkephalogramm des Menschen. Archiv für Psychiatrie und Nervenkrankheiten 1929;87(1):527–70.

[2] Koch C, Jones A. Big science, team science, and open science for neuroscience. Neuron 2016;92(3):612–6.

[3] To the cloud! a grassroots proposal to accelerate brain science discovery. Neuron 2016;92(3):622–7.

[4] Obeid I, Picone J. The temple university hospital EEG data corpus. Front Neurosci 2016;10:196.

[5] Schalk G, et al. BCI2000: A general-purpose brain-computer interface (BCI) system. IEEE Trans Biomed Eng 2004;51(6):1034–43.

[6] Goldberger AL, et al. PhysioBank, PhysioToolkit, And PhysioNet: components of a new research resource for complex physiologic signals. Circulation 2000;101(23):E215–20.

[7] Ioannidis JP, Khoury MJ. Improving validation practices in omics research. Science 2011;334(6060):1230–2.

[8] Peng RD. Reproducible research in computational science. Science 2011;334(6060):1226–7.

[9] Delorme A, Makeig S. EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. J Neurosci Meth 2004;134(1):9–21.

[10] Teeters JL, et al. Neurodata without borders: Creating a common data format for neurophysiology. Neuron 2015;88(4):629–34.

[11] Bigdely-Shamlo N, Makeig S, Robbins KA. Preparing laboratory and real-world EEG data for large-scale analysis: A containerized approach. Front Neuroinform 2016;10(7).

[12] Pernet CR, et al. EEG-BIDS, An extension to the brain imaging data structure for electroencephalography. Sci Data 2019;6(1):103.

[13] Gilat A. MATLAB. An introduction with applications. 5th ed.. 2014.

[14] Merkel D. Docker: lightweight linux containers for consistent development and deployment. Linux J 2014;2014(239):2.

[15] Li FL, et al. Relationships between the resting-state network and the P3: Evidence from a scalp EEG study. Sci Rep 2015;5.

[16] Maumet C, et al. Sharing brain mapping statistical results with the neuroimaging data model. Sci Data 2016;3:160102-160102.

[17] Nichols TE, et al. Best practices in data analysis and sharing in neuroimaging using MRI. Nat Neurosci 2017;20(3):299–303.

[18] Pernet CR, et al. Best practices in data analysis and sharing in neuroimaging using MEEG. 2018, OSF Preprints.

[19] Oikonomou VP, et al. Comparative evaluation of state-of-the-art algorithms for SSVEP-based BCIs. 2016, arXiv preprint, arXiv:1602.00904.