EEG Recording and Analysis

For each group (withPCS and withoutPCS), 5 minutes of resting state with eyes open and 5 minutes of resting state with eyes closed were recorded using high-density EEG. Since the eyes-closed condition represents a simple, standardized procedure (Babiloni et al., 2016), it is the most commonly used (Babiloni et al., 2022) and will therefore be analyzed in this study to ensure comparability. EEG signals were recorded using a 128-channel EEG cap (128Ch Standard Brain Cap for actiCHamp Plus, Easycap GmbH, Wörthsee, Germany) with electrodes positioned in an equidistant layout, connected to an actiCHamp Plus Amplifier (Brain Products GmbH, Gilching, Germany).

The sampling rate was 1000 Hz with an amplitude resolution of 0.1 µV. Electrolyte gel was applied to improve conductivity between skin and electrodes, ensuring impedances remained below 20 kΩ. Eye movements and changes in the resting potential of the retina (EOG activity) were monitored using two EOG electrodes placed below each eye, with impedances also maintained below 20 kΩ. In addition, a ground electrode was positioned on the forehead, and a reference electrode was positioned on the tip of the nose. Impedances for both the reference and ground electrode were kept below 5 kΩ.

Preprocessing

Data preprocessing/analysis was performed using the FieldTrip toolbox (Fieldtrip-20240504; Oostenveld, Fries, Maris & Schoffelen, 2011) and the EEGLab toolbox (v2024.0; Delorme & Makeig, 2004) in Matlab (v24.1.0.2578822 (R2024a) Mathworks Inc., 2024, MathWorks® https://de.mathworks.com) on Windows.

The participants' EEG data were organized in BIDS (Brain Imaging Data Structure) format (Gorgolewski et al., 2016; Pernet et al., 2019). BIDS is a community standard that ensures homogeneity in the organization and description of raw neurocognitive/brain-derived/neuroscientific data, enabling efficient data sharing, minimizing errors, and supporting completely automated analysis workflows (Gorgolewski et al., 2016; Pernet et al., 2019; Truong, Robbins, Delmore & Makeig, 2023). The rsEEG data, organized according to this standard, were identified and imported into MATLAB using the FieldTrip Toolbox. A trial defining function was built to select the data from the eyes-open condition for subsequent processing. This resulted in approximately 300 s per participant.

Filtering and Resampling

Commented [Janka Hau1]: The analysis of this study was focused on the condition eyes open only.
All code described in the following sections can be found in this public GitHub repository: https://github.com/LGodbersen/Masters-thesis.

Commented [JH2]: Here or methods?

Commented [CN3]: Methods

Commented [Janka Hau4]: Here or earlier? Where should I mention how many Versuchsleiter!?

Commented [Janka Hau5]: Or just: EOG activity was recorded using two dedicated EOG electrodes, placed below each eye.

Commented [Janka Hau6]: Matlab is widely used by the EEG community and enabed us to use well-established Matlab-based EEG toolboxes that provide robust functions for computing functional, connectivity measures (Avila et al., 2023) - they followed a pragmatic approach towards preprocessing and adopted a simple, established, and automatic workflow in EEGLAB proposed by Pernet et al. And originally developed for ERP data. Adapted this pipeline to resting-state data and detail the seven preprocessing steps below.

Commented [Janka Hau7]: EEGLAB (Delorme and Makeig, 2004) is the most commonly
used platform for EEG data analysis (Hanke and Halchenko, 2011; Martínez-Cancino et al., 2020) and all steps proposed can
also be reproduced from the user interface

Commented [Janka Hau8]: How to cite that correctly?

Commented [Janka Hau9]: Relevant?

Commented [Janka Hau10]: What describes EEG and MRI data best together?

Commented [Janka Hau11]: Where to find?

Commented [Janka Hau12]: Correct like that?

A finite impulse response (FIR) windowed-sinc (firws) filter, designed with a hamming windowed sinc function and implemented in the FieldTrip toolbox, was used for both high-pass and low-pass filtering of the continuous data. For high-pass filtering, a cut-off frequency of 0.1 Hz was applied to eliminate very low frequencies (drift) (Keil et al., 2013). This cut-off was based on the findings of Delorme (2023) and Winkler, Debener, Müller and Tangermann (2015), where filtering at 0.1 Hz or higher significantly improved data quality compared to no filtering. Filters above 0.1 were not used due to….

Commented [Janka Hau13]: .The FIR filter was chosen for its stability and linear phase characteristics.

is that true?

Commented [Janka Hau14]: Do I need literature?

Commented [Janka Hau15]: I think Lara changed that. Before more. Why not 0.1? Need to check voice messages

Commented [JH16]: Maybe just delete?

Prior to applying low-pass filtering, the data was downsampled from 1000 Hz to 250 Hz, to reduce computational load while preserving sufficient temporal resolution for subsequent analysis. A cut-off frequency of 45 Hz was then used to eliminate high-frequency noise and mitigate potential 50 Hz line noise (Delorme, 2023). Finally, the data underwent re-referencing using the Common Average Reference (CAR) technique to remove the influence of the reference and improve signal quality (Ludwig et al., 2009). As the name implies, an average of the recordings from all electrode sites was computed and used as the reference (Ludwig et al., 2009; Offner, 1950). The function reref() was used (to perform this step). The data was converted into the EEGLAB data structure for further processing.

Commented [Janka Hau17]: Avila et al., 2023

Commented [Janka Hau18]: True?

Due to empty dataset from one participant, the participant was excluded, leaving the dataset with 69 participants (something like that. But where should I write that?)

Commented [Janka Hau19]: One participant with empty dataset (KA14HH)

Artifact removal

After the initial filtering and resampling, the preprocessing pipeline continued with detecting and removing artifacts. First, large artifacts, including the removal of flat-line channels, noisy channels, and short-time bursts of noise, were removed from the data using the EEGLAB pop_clean_rawdata() function with specific parameters. Channels with flat lines for more than 5 seconds were removed (FlatlineCriterion = 5), based on the default recommendation (for this parameter) by Pernet et al. (2021). This ensured the exclusion of "dead" or disconnected channels, thereby improving data quality. Channels were further excluded if their signal could not be predicted from a randomly selected subset of the remaining channels for at least 85% of the recording time (ChannelCriterion = 0.85), to remove those that were highly dissimilar from the rest of the channels (Gil Ávila et al., 2023; Pernet et al., 2021). The euclidean distance metric was used to calculate the similarity between channels. Data segments with abnormally high amplitude bursts, exceeding 100 SD compared to neighboring segments, were eliminated (BurstCriterion = 100), as such extreme bursts are considered unlikely to

Commented [Janka Hau20]: Data was cleaned

Commented [RH21]: Delete?

Commented [Janka Hau22]: But they suggested 0.8 which is the default reflect brain signals (Chang et al., 2018). The default BurstCriterion is set to 20, but it may be adjusted if the default setting results in rejecting too many data segments. Some scientist recommend setting the threshold to 100 (EEGLAB, "Automated Pipeline Tutorial", 2024), which aligns with the optimal cut-off range of 10 to 100 suggested by Chang et al. (2018). Therefore, a mild threshold of 100 was chosen here, as it still effectively removes large-amplitude artifacts while retaining valuable data (Chang et al., 2018). Time windows where more than 40% of the channels were marked as noisy were removed (WindowCriterion = 0.4), to ensure the quality of the remaining data. A more lenient threshold of 0.4 was chosen over the default of 0.25 to retain more data (even if it is potentially noisier). How many “bad” (excessively noisy) channels were detected or removed in this process?

Again the data is re-referenced to the average reference (CAR), this time using the EEGLAB function pop_reref() (Gil Ávila et al., 2023).

Secondly, Independent Component Analysis (ICA) (Bell and Sejnowski, 1995; Hyvärinen, 2013, Jung et al., 2000; Lee et al., 1999; Palmer et al., 2008) was performed on the data, to detect and reject further artifacts, such as eye or muscle movements (Makeig et al., 1995). ICA was performed with the “runica” algorithm and function pop_runica() with the extended InfoMax method. The runica algorithm was employed with the extended InfoMax method. using the pop_runica function implemented in EEGLAB. To avoid rank deficiency, the number of components was set to one less than the total number of channels (Kim, Luo, Chu, Cannard, Hoffman & Miyakoshi, 2023). This approach decomposes the EEG signal into independent components, potentially separating artifacts from neural activity. Due to the non-deterministic nature of the ICA algorithm, its results vary across repetitions. That is, every repetition of the ICA algorithm leads to small differences in the reconstructed time series after removing artifactual components (Gil Ávila et al., 2023). The resulting ICA weights, which represent the transformation matrix for this decomposition, were saved in a separate file. Automatic component rejection was implemented using ICALabel (Pion-Tonachini, Kreutz-Delgado & Makeig, 2019), as automatic artifact rejection is preferred over the manual one to ensure standardization (Miljevic et al., 2022). Artifactual components are automatically classified by the ICLabel classifier (Pion-Tonachini et al., 2019). Thresholds were set at probabilities of 0.8 (80%) for muscle-related components (Pernet et al., 2021) and 0.5 (50%) for eye-related components. Components exceeding these thresholds were flagged and automatically removed using the EEGLAB function pop_subcomp(). By default, only components whose probability of being “muscle” is higher than 80% were subtracted from the

Commented [Janka Hau23]: Seems weird but that‘s how it is written in tutorial

Commented [Janka Hau24]: Criterion for removing time windows that were not repaired completely

Commented [Janka Hau25]: Or bad

Commented [RH26]: Was?

Commented [RH27]: double

Commented [Janka Hau28]: stolen

data (Pernet et al., 2021). The two EOG channels (31 and 32) were removed from the dataset. The cleaned dataset was then checked for consistency using eeg_checkset().

After this steps, an average of 110.3 good channels remained in Cluster 1 (*SD* = 11.4, Range = 75 - 125) and 109 good channels in Cluster 2 (*SD* = 11, Range = 78 - 124) in the two-cluster solution. This corresponds to approximately 85-86% good channels in both groups.

Commented [JH29]: 86 in cluster1 and 85 in cluster2

In the four-cluster solution an average of 110.8 good channels remained in Cluster 1 (*SD* = 11.84, range = 75–125), 109 good channels in Cluster 2 (*SD* = 11.03, range = 78–124), 108.6 good channels in Cluster 3 (*SD* = 12.19, range = 77–122), and 113 good channels in Cluster 4 (*SD* = 6.64, range = 105–123) after ICA. This corresponds to approximately 85-88% good channels.

Commented [JH30]: **Cluster 1: 86.56%**
Cluster 2: 85.16%
Cluster 3: 84.84%
Cluster 4: 88.28%

Thirdly and finally, an additional artifacts removal step was implemented to address any remaining problematic channels. This process involved a statistical approach to identify outlier channels based on their signal characteristics. The standard deviation and mean were calculated for each channel across all time points. Then, overall mean values for these standard deviations and means were computed across all channels. Thresholds were established at 2.5 standard deviations above and below the overall mean, creating an acceptable range for channel activity. Channels with standard deviations falling outside this range were identified as outliers. These outlier channels were then removed from the dataset using the EEGLAB function pop_select(), further refining the EEG data quality. This step ensures that channels with unusually high or low variability, which might represent persistent artifacts or malfunctioning electrodes, are excluded from subsequent analyses. As a result, the dataset retained an average 108 good channels in Cluster 1 (*SD* = 11.2, Range = 73 - 121) and 106.6 good channels in Cluster 2 (*SD* = 10.9, Range = 77 - 121) in the two-cluster solution. This corresponds to approximately 83-84% good channels in both groups.

Commented [JH31]: 84 in cluster1 and 83 in cluster2

Commented [JH32R31]: Or should I just mention on number here

In the four-cluster solution an average of 108.4 good channels remained in Cluster 1 (*SD* = 11.46, range = 73–121), 106.6 good channels in Cluster 2 (*SD* = 10.92, range = 77–121), 106.3 good channels in Cluster 3 (*SD* = 12.17, range = 74–120), and 111.1 good channels in Cluster 4 (*SD* = 6.20, range = 104–121). This corresponds to approximately 83-87% good channels.

Commented [JH33]: **Cluster 1: 84.69%**
Cluster 2: 83.28%
Cluster 3: 83.05%
Cluster 4: 86.80%

Interpolate bad channels

Channels removed in the previous step were interpolated using the EEGLAB function pop_interp() with the default spherical splines method (Perrin, Pernier, Bertrand & Echallier, 1989), ensuring a consistent number of channels across participants (Gil Ávila et al., 2023).

Interpolated channels were inserted into the original channel order. On average 16-17% of the channels in each group were interpolated.

Epoch length and number

Lastly, the continuous data were segmented into epochs using the eeg_regepochs() function implemented in the EEGLAB toolbox, which defaults to 2-second epochs (Gil Ávila et al., 2023). However, longer epochs improve frequency resolution, making them preferable for slower frequencies like delta. To achieve higher resolution while maintaining an adequate trial count, EEG data for each participant were segmented into 5-second nonoverlapping epochs. This function then outputs the new epoch EEG as a dataset on EEGLAB (Bonello, Garg, Garg & Audu, 2018).

The preprocessing resulted in an average of 37.7 good epochs (SD = 14.7, Range = 4 – 60) in Cluster 1, 37.9 good epochs (SD = 16.6, Range = 3 – 59) in Cluster 2 in the two-cluster solution. A two-sided *t*-test did not indicate a significant differences in epoch number between groups, $t = -0,0585$, $p = .95$. In the four-cluster solution, the dataset retained an average of 37.6 good epochs (*SD* = 16.2, range = 4–58) in Cluster 1, 37.9 good epochs (*SD* = 16.6, range = 3–59) in Cluster 2, 35.9 good epochs (*SD* = 14.5, range = 9–60) in Cluster 3, and 42.0 good epochs (*SD* = 12.5, range = 24–60) in Cluster 4.

Power Analysis

Preprocessed EEG data were converted from EEGLAB format to FieldTrip format using the eeglab2fieldtrip function. A common method for characterizing rsEEG is to decompose oscillatory signal into spectral power across distinct frequency bands (Babiloni et al., 2016; Perez et al., 2024). Spectral power reflects the distribution of neural activity at specific frequencies and is associated with various cognitive processes (Babiloni et al., 2016; Perez et al., 2024; Ward, 2003). Spectral parameterization was performed using SpecParam (formerly FOOOF, Fitting Oscillations & One Over F; Donoghue et al., 2020), which is implemented in the Brainstorm Toolbox (Tadel et al., 2011) and available in FieldTrip. This approach separates the periodic and aperiodic components of the power spectrum.

Since changes in absolute power may not solely reflect true neural activity but could also result from shifts in the aperiodic exponent, broadband offset, or frequency center (Donoghue et al., 2020), this analysis focused on relative power while also examining the aperiodic exponent and offset to account for these cofounding factors.

Spectral analysis of relative power across the 128 scalp electrodes was conducted using FieldTrips's multitaper spectral estimation with Hanning taper, analyzing frequencies between 0.3 and 30 Hz with a frequency resolution of 0.2 Hz. The fooof output was set to a fixed

Commented [JH34]: Only one number here?

Commented [JH35]: Two-cluster solution

Commented [Janka Hau36]: If I want to do the whole thing in steps

Commented [RH37]: Might not need to explain why

Commented [RH38]: Farina et al., 2020 also chose that

Commented [Janka Hau39]: Not in zotero yet

Commented [JH40]: Delete?

Commented [JH41]: Do I need to mention this?

aperiodic mode. To obtain relative delta and beta power, the aperiodic components were substracted from the original power spectra. Delta power was defined as 0.6-4 Hz, and beta power as 14-30 Hz. The summed power across all frequencies within each band was used to compute the relative power per channel.

Once the relative power per channel was computed, the data were transferred to R Studio (…) for fruther analysis using R Statistical Software (….). To identify and remove extreme values, an initial outlier detection was performed. For each participant, channels exceeding ±3 SD from the mean relative power were excluded. This process was applied seperately for delta power, beta power, aperiodic exponent, and aperiodic offset. Starting with … channels per cluster, this step retained ..% of channels in Cluster 1 and …% of channels in Cluster 2 for delta power, …% for beta power, …% for the aperiodic exponent, and ….% (Cluster 1) and …% (Cluster 2) for the aperiodic offset. Further outlier removal was considered. André (2022) advocates for a hypothesis-blind approach, where outliers are removed across clusters rather than within them. However, Karch (2023) questions this method, suggesting that extreme values should either be corrected or removed and that statistical methods less sensitive to outliers, such as a sign-rank test, may be more appropriate. For delta and beta power, an outlier removal approach across clusters was used, retaining X% of channels in cluster 1 and X% in cluster 2 for delta power, and X% in both clusters for beta power. A comparison between within-cluster and across-cluster outlier removal showed that the choice of method did not influence the significance of the results.

EEG power values are inherently non-negative; however, when using the FOOOF/SpecParam method, negative power values can sometimes occur. To ensure meaningful relative power estimates, these negative values were set to zero. Specifically, X% of channels in Cluster 1 and X% in Cluster 2 had negative values for delta power, while for beta power, the proportion was considerably smaller, at X% and X%, respectively. Removing these values may reduce the overall variance in the data.