

Fcp_2_5 ICA Checkpoint Databrowser Guide

A guide on how to identify components associated with artifacts

Fcp_2_5 Overview	1
General steps	1
Instructions	1
Step 1: Getting familiar with the data browser	1
Step 2: Identifying artifacts	4
Step 3: Saving the bad components	5

Note: This guide is written under the assumption that the user is familiar with the broad steps of the MEGneto pipeline and their functionality. The user should have run through fcp_2 up until this point and now requires clarification on proper selection of ICA components for fcp_2_5. If the user is not familiar with the pipeline and step fcp_2_5 they should first consult other documentation on the pipeline.

Fcp_2_5 Overview

The fcp_2_5_checkpoint.m step of the MEG pipeline is an interactive session that uses a data browser to guide the user through inspection of ICA components and identify components associated with artifacts (unwanted features in data such as heartbeats, blinks, etc.). After inspection, the pipeline backprojects ICA components to remove the signal corresponding with the bad ICA components. This guide is intended to give users an idea of how to use the data browser feature to identify the bad components.

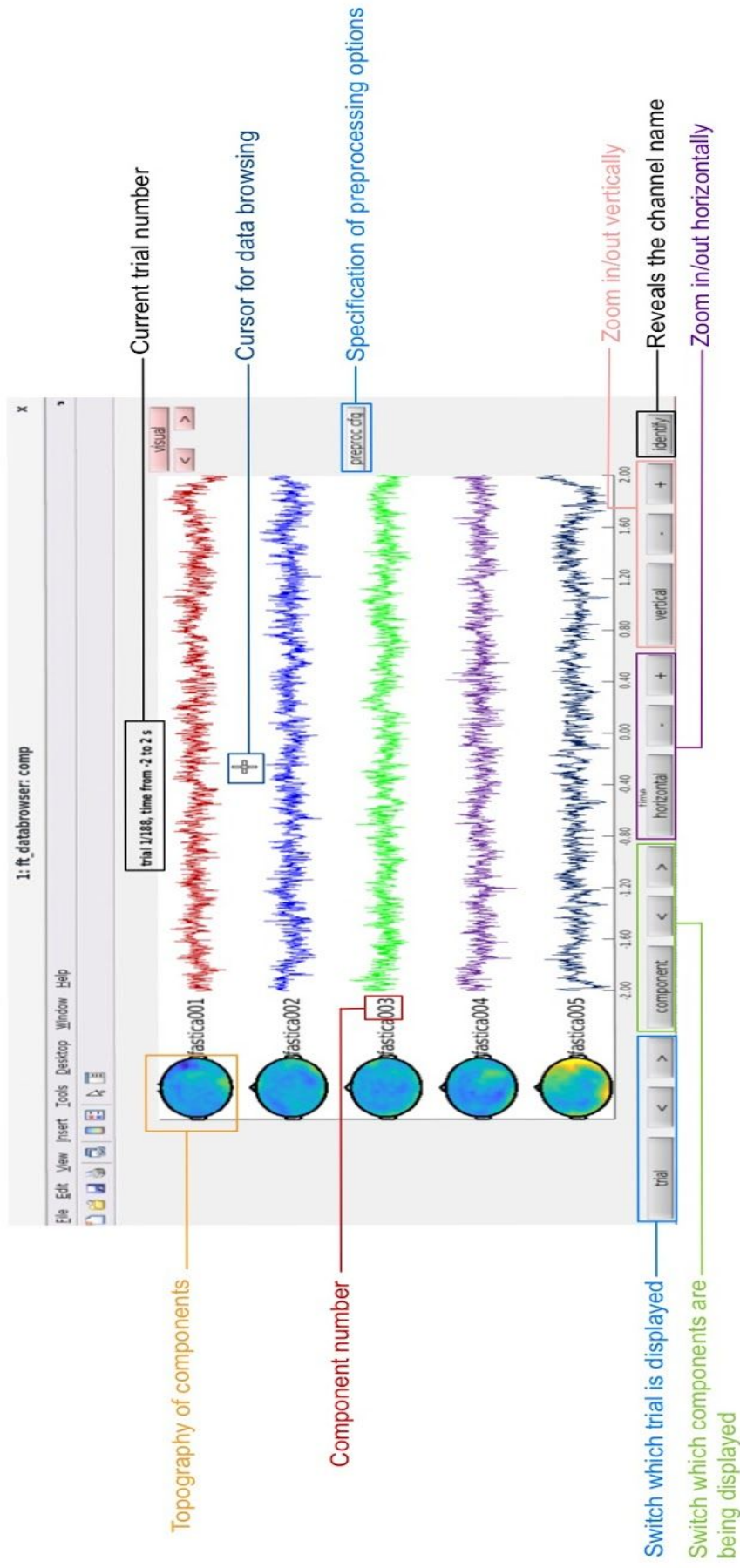
General steps

A broad overview of the steps in this guide is as follows: (1) getting familiar with the FieldTrip data browser (ft_databrowser), (2) identifying artifacts, (3) saving the bad components.

Instructions

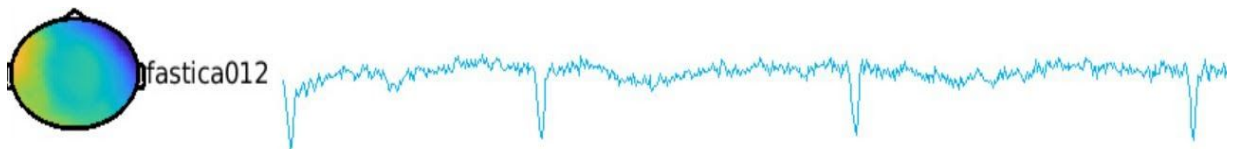
Step 1: Getting familiar with the data browser

- When the fcp_2_5_checkpoint step is run, a data browser will pop up for the first participant, require the user to browse through it to identify artifacts, and the user will then have to turn to Matlab's command window to enter either an array containing the bad components, 'skip' if no artifacts exist, or 'disp' to re-display the data browser. This will repeat for each participant.
- The data browser and its components are explained in the figure below. Additional information on FieldTrip's databrowser can be found [here](#).



Step 2: Identifying artifacts

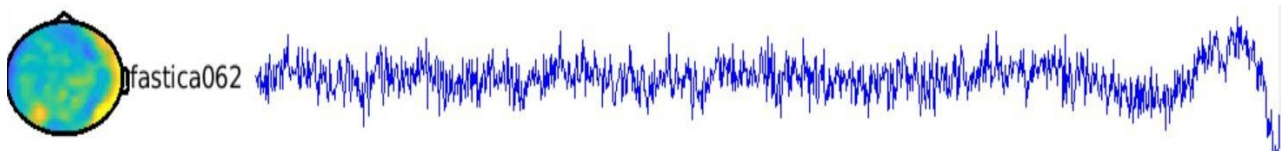
- To identify artifacts, users should use the data browser to scroll through the components from each trial. When identifying artifacts, it is important that users pay attention to markers given by both the component topography (displayed in the heads to the left of the component numbers) and the component signals (displayed to the right of the component numbers).
- It is important that users note the component number that contains an artifact, so that they can enter it in the pipeline for removal.
 - Typically, artifacts are found within the first 20 components. Users should not remove too many components. Typically, researchers aim for a maxim of 6 components to remove.
- The most common artifacts that users will run into and their respective topography/signal component markers are outlined below:
 - ❖ *Heartbeats*: Topography components will have a ring around the edges of the head boundaries. Component signals will display repeated ECG peaks.



- ❖ *Blinks*: Topography components will show coloration in the frontal area, near the eyes. Component signals will be uniform for the majority but have sudden dips.



- ❖ *Head/neck movement*: Topography components will be localized to the ears or the back of the neck. Component signal effects will depend on the type of head movement, but will often display a sudden spike..



- ❖ *Saccades*: Topography components will have the frontal area completely covered. Component signals will appear uniform followed by a sudden change.

- ❖ *Shunt/metal artifacts*: Topography components will be very localized and intense in coloration. Component signals will display a sudden spike, often towards the end of the signal.



- ❖ *Squid jumps*: Component signals will display very large and sudden spikes (straight, vertical lines).

Rule of thumb: If you are unsure whether a component should be removed, do not remove it.

Step 3: Saving the bad components

- After using the data browser, the user should return to the command window in Matlab where they should enter the bad components they have found (e.g. [2, 5, 7]), enter 'skip' if no bad components were identified, or enter 'disp' to re-display the data browser. If bad components were entered, they will be rejected from further data analysis.
- After running fcp_2_5, the bad components that the user has entered for each participant will be saved. The user should navigate to their designated output folder (specified by 'analysis name' at the beginning of the pipeline when filling in the paths for your project/analysis), and within that, find the 'analysis' folder, and then the 'group' folder where they can open 'fcp2_5_output.json'. Within this file, the field 'bad_comp' will contain an array of the bad components entered for each participant. The user should ensure that no participant has an excessive amount of bad components. If they do, they may need to be removed from analysis in subsequent steps of the pipeline.