

```
>> cd P:\Project_Sezen-EMS_VR\data
>> input_path = [study_folder raw_data_folder];
output_path = [study_folder mobi_data_folder];
>> input_path

input_path =

    'P:\Project_Sezen-EMS_VR\data\0_raw_data\'

>> output_path

output_path =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'

>> input_path = [study_folder mobi_data_folder];
>> input_path

input_path =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'

eeglab: options file is C:\Users\Lukas\eeg_options.m
EEGLAB: adding "ADJUST" v1.1.1 (see >> help eegplugin_adjust)
EEGLAB: adding "CleanLine" v1.03 (see >> help eegplugin_cleanline)
EEGLAB: adding "Fieldtrip-lite" to the path; subfolders (if any) might be missing from the path
EEGLAB: adding "HEDTools " v1.0.2 (see >> help eegplugin_hedtools)
EEGLAB: adding "Mutual_Info_Clustering" v1.00 (see >> help eegplugin_miclust)
EEGLAB: adding "PrepPipeline" v0.5 (see >> help eegplugin_prepPipeline)
EEGLAB: adding "SASICA" v1.3.4 (see >> help eegplugin_SASICA)
Initializing SIFT...
Start SIFTing!
EEGLAB: adding "SIFT" v1.41 (see >> help eegplugin_sift)
EEGLAB: adding "amica" v1.5 (see >> help eegplugin_amica)
EEGLAB: adding "bemobil_pipeline" v0.1 (see >> help eegplugin_bemobil_pipeline)
EEGLAB: adding "clean_rawdata" v0.31 (see >> help eegplugin_clean_rawdata)
EEGLAB: adding "corrmap" v2.02 (see >> help eegplugin_corrmap)
EEGLAB: adding "dipfit" v2.3 (see >> help eegplugin_dipfit)
EEGLAB: adding "firfilt" v1.6.1 (see >> help eegplugin_firfilt)
EEGLAB: adding "firfilt" v1.6.2 (see >> help eegplugin_firfilt)
EEGLAB: adding "mobilab" v? (see >> help eegplugin_mobilab)
EEGLAB: adding "postAmicaUtility" v1.00 (see >> help eegplugin_postAmicaUtility)
EEGLAB: adding "std_dipoleDensity" to the path; subfolders (if any) might be missing from the path
EEGLAB: adding "std_dipoleDensity" v0.36 (see >> help eegplugin_std_dipoleDensity)
EEGLAB: adding "xdfimport1.13b" v1.12 (see >> help eegplugin_xdfimport)
Warning:
A newer version of EEGLAB (14.1.2) is available here
This version fixes issues with Matlab 2018a. See Release notes for more informations.
You may disable this message in the Option menu but will miss critical updates.

Subject #1
Error using dataSourceMoBI (line 30)
The folder is empty.
```

```
>> filenames
```

```
filenames =
```

```
1x3 cell array
```

```
    'Training'    'Vibro'    'Visual'
```

```
>> input_path = [study_folder mobi_data_folder];  
output_path = [study_folder raw_EEGLAB_data_folder];
```

```
if ~exist('ALLEEG','var'); eeglab; end  
if ~exist('mobilab','var'); runmobilab; end
```

```
pop_editoptions( 'option_storedisk', 0, 'option_savetwofiles', 1, ↵  
'option_saveversion6', 0, 'option_single', 0, 'option_memmapdata', 0, ↵  
'option_eegobject', 0, 'option_computeica', 1, 'option_scaleicarms', 1, ↵  
'option_rememberfolder', 1, 'option_dontusetoolboxes', 0, 'option_checkversion', 1, ↵  
'option_chat', 1);
```

```
Using option file in directory C:\Users\Lukas
```

```
>> subject =1
```

```
subject =
```

```
1
```

```
>> disp(['Subject #' num2str(subject)]);  
    output_filepath = [output_path num2str(subject)];  
    mkdir(output_filepath); % create folder
```

```
Subject #1
```

```
>> output_filepath
```

```
output_filepath =
```

```
    'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1'
```

```
>> input_path
```

```
input_path =
```

```
    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'
```

```
>> length(filenames)
```

```
ans =
```

```
3
```

```
>> for file = filenames; disp(file); end
```

```
    'Training'
```

```
    'Vibro'
```

```
    'Visual'
```

```
>> input_path = [study_folder raw_data_folder];
output_path = [study_folder mobi_data_folder];

if ~exist('mobilab','var'); eeglab; runmobilab; end
>> input_path

input_path =

    'P:\Project_Sezen-EMS_VR\data\0_raw_data\'

>> subjects

subjects =

     1

>> subject = 1

subject =

     1

>> length(filenamees)

ans =

     7

>> disp(['Subject #' num2str(subject)]);
Subject #1
>> xdf_file = 1

xdf_file =

     1

>> disp(['Importing file: ' filenamees{xdf_file} '...']);
Importing file: block_Training...
>> input_filepath = [input_path num2str(subject) '\\' filenamees{xdf_file} '.xdf'];
    output_filepath = [output_path num2str(subject) '\\' filenamees{xdf_file}
'_MoBI'];
>> input_filepath

input_filepath =

    'P:\Project_Sezen-EMS_VR\data\0_raw_data\1\block_Training.xdf'

>> output_filepath

output_filepath =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\1\block_Training_MoBI'

>> mobilab.allStreams = dataSourceXDF(input_filepath,output_filepath);
```

```

disp('...done.');
```

Warning: Stream BrainVision RDA Markers has no time stamps. It cannot be imported.
> In dataSourceXDF (line 58)

```

1-> Stream BrainVision RDA:
    uuid:      44c23053-4984-4835-aa9b-4b5b71ec51aa
    host:      bpn-c012
    type:      EEG
    session id: default
    created at: 5647.6483552529999
    samples:   574950
    channels:   64
2-> Stream BrainVision RDA Markers:
    uuid:      f69748bc-fef8-4f22-9a6d-2ec612945980
    host:      bpn-c012
    type:      Markers
    session id: default
    created at: 5647.6501936539998
    samples:   0
    channels:   0
Warning: Stream BrainVision RDA Markers has no time stamps. It cannot be imported.
> In dataSourceXDF (line 78)
3-> Stream Unity_Markers_PredError:
    uuid:      fdd413dd-315a-4723-8e41-678eae004eb0
    host:      BPN-C043
    type:      Markers
    session id: default
    created at: 9619.1572963210001
    samples:   87
    channels:   1
Adding object: brainvision_rda_bpn-c012
Adding object: unity_markers_prederror_BPN-C043
Logs were saved in: P:\Project_Sezen-✓
EMS_VR\data\1_mobi_data\1\block_Training_MoBI\logfile.txt
...done.
Subject #1
Importing file: block_Training...
Warning: MoBILAB needs an empty folder to start a new session. All the existent files✓
in this folder will be compressed.
> In dataSource.checkThisFolder (line 592)
    In dataSourceXDF (line 41)
Zipping: P:\Project_Sezen-EMS_VR\data\1_mobi_data\1\block_Training_MoBI\lost+found.✓
zip...
Operation terminated by user during createArchive>addArchiveEntry (line 114)

In createArchive (line 41)
    addArchiveEntry(archiveFileName, createArchiveEntryFcn, entries(i), ...

In zip (line 60)
    archive = createArchive(zipFilename, files, rootDir, ...

In dataSource.checkThisFolder (line 596)
    zip(zipfile,files,folder);

In dataSourceXDF (line 41)

```

```
obj.checkThisFolder(mobiDataDirectory);

>>
>> xdf_file = 2

xdf_file =

    2

>> disp(['Importing file: ' filenames{xdf_file} '...']);

    input_filepath = [input_path num2str(subject) '\\' filenames{xdf_file} '.xdf'];
    output_filepath = [output_path num2str(subject) '\\' filenames{xdf_file} \
'_MoBI'];

    mobilab.allStreams = dataSourceXDF(input_filepath,output_filepath);
    disp('...done.');
```

Importing file: block_Vibro...

Warning: Stream BrainVision RDA Markers has no time stamps. It cannot be imported.
> In dataSourceXDF (line 58)

1-> Stream BrainVision RDA:

uuid:	44c23053-4984-4835-aa9b-4b5b71ec51aa
host:	bpn-c012
type:	EEG
session id:	default
created at:	5647.6483552529999
samples:	1501250
channels:	64

2-> Stream BrainVision RDA Markers:

uuid:	f69748bc-fef8-4f22-9a6d-2ec612945980
host:	bpn-c012
type:	Markers
session id:	default
created at:	5647.6501936539998
samples:	0
channels:	0

Warning: Stream BrainVision RDA Markers has no time stamps. It cannot be imported.
> In dataSourceXDF (line 78)

3-> Stream Unity_Markers_PredError:

uuid:	720cf29e-b4ca-411d-8326-af3ffc867cd7
host:	BPN-C043
type:	Markers
session id:	default
created at:	13225.93807758
samples:	535
channels:	1

4-> Stream Rigid_Leap_Hand:

uuid:	2c0cb3f6-8472-4c0a-907a-dd5e31098411
host:	BPN-C043
type:	rigidBody
session id:	default
created at:	13229.909623236999
samples:	51754
channels:	7

Adding object: brainvision_rda_bpn-c012

```

Adding object: rigid_leap_hand_BPN-C043
Adding object: unity_markers_prederror_BPN-C043
Logs were saved in: P:\Project_Sezen-✓
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI\logfile.txt
...done.
>> mobilab.allStreams.item{4}
Index exceeds matrix dimensions.

>> mobilab.allStreams(4).item
Index exceeds matrix dimensions.

>> mobilab.allStreams{3}.item
Cell contents reference from a non-cell array object.

>> mobilab.allStreams(3).item
Index exceeds matrix dimensions.

>> mobilab.allStreams.item{3}

ans =

Warning: Variable 'auxChannel' not found.
> In coreStreamObject/retrieveProperty (line 799)
   In coreStreamObject/get.auxChannel (line 403)
   In coreStreamObject/disp (line 835)
Saving auxChannel in: P:\Project_Sezen-✓
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI\unity_markers_prederror_BPN-C043_720cf29e-✓
b4ca-411d-8326-af3ffc867cd7_adb1ff0a-520f-47f9-92ef-0195dc2ba244.hdr

Class: markerStream
Properties:
    name:          unity_markers_prederror_BPN-C043
    uuid:          720cf29e-b4ca-411d-8326-af3ffc867cd7
    samplingRate:  9.428032e-01 Hz
    timeStamp:     <1x535 double>
    numberOfChannels: 1
    data:          <535x1 int16>
    artifactMask:  <535x1 sparse>
    event.latencyInFrame: <1x535 double>
    event.label:   <535x1 cell>
    label:        <1x1 cell>
    unit:         unknown
    owner.name:    Lukas
    owner.organization: SCCN
    owner.email:   Lukas@sccn.ucsd.edu
    sessionUUID:   adb1ff0a-520f-47f9-92ef-0195dc2ba244
    auxChannel.label: <0x1 cell>
    auxChannel.data: <535x0 int16>
    history:       mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-✓
EMS_VR\data\0_raw_data\1\block_Vibro.xdf','P:\Project_Sezen-✓
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
>> mobilab.allStreams.item{4}
Index exceeds matrix dimensions.

>> mobilab.allStreams.item{3}

```

ans =

Class: markerStream

Properties:

```

    name:                unity_markers_prederror_BPN-C043
    uuid:                720cf29e-b4ca-411d-8326-af3ffc867cd7
    samplingRate:        9.428032e-01 Hz
    timeStamp:           <1x535 double>
    numberOfChannels:    1
    data:                <535x1 int16>
    artifactMask:        <535x1 sparse>
    event.latencyInFrame: <1x535 double>
    event.label:         <535x1 cell>
    label:               <1x1 cell>
    unit:                unknown
    owner.name:          Lukas
    owner.organization:   SCCN
    owner.email:         Lukas@sccn.ucsd.edu
    sessionUUID:         adb1ff0a-520f-47f9-92ef-0195dc2ba244
    auxChannel.label:    <0x1 cell>
    auxChannel.data:     <535x0 int16>
    history:             mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-
EMS_VR\data\0_raw_data\1\block_Vibro.xdf','P:\Project_Sezen-
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
>> mobilab.allStreams.item{2}

```

ans =

Class: mocapRigidBody

Properties:

```

    name:                rigid_leap_hand_BPN-C043
    uuid:                2c0cb3f6-8472-4c0a-907a-dd5e31098411
    samplingRate:        90 Hz
    timeStamp:           <1x51754 double>
    numberOfChannels:    7
    data:                <51754x7 double>
    artifactMask:        <51754x7 sparse>
    event.latencyInFrame: <1x0 double>
    event.label:         <0x1 cell>
    label:               <7x1 cell>
    unit:                meters
    owner.name:          Lukas
    owner.organization:   SCCN
    owner.email:         Lukas@sccn.ucsd.edu
    sessionUUID:         adb1ff0a-520f-47f9-92ef-0195dc2ba244
    auxChannel.label:    <0x1 cell>
    auxChannel.data:     <51754x0 double>
    history:             mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-
EMS_VR\data\0_raw_data\1\block_Vibro.xdf','P:\Project_Sezen-
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
>> mobilab.allStreams.item{1}

```

ans =

Class: eeg

Properties:

```

    name:          brainvision_rda_bpn-c012
    uuid:          44c23053-4984-4835-aa9b-4b5b71ec51aa
    samplingRate:  2500 Hz
    timeStamp:     <1x1501250 double>
    numberOfChannels: 64
    data:          <1501250x64 double>
    artifactMask:  <1501250x64 sparse>
    event.latencyInFrame: <1x0 double>
    event.label:    <0x1 cell>
    label:         <64x1 cell>
    unit:          microvolts
    owner.name:    Lukas
    owner.organization: SCCN
    owner.email:   Lukas@sccn.ucsd.edu
    sessionUUID:   adb1ff0a-520f-47f9-92ef-0195dc2ba244
    auxChannel.label: <0x1 cell>
    auxChannel.data: <1501250x0 double>
    history:       mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-
EMS_VR\data\0_raw_data\1\block_Vibro.xdf', 'P:\Project_Sezen-
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
    channelSpace:  <64x3 double>
    surfaces:      ''
    atlas.colorTable: []
    atlas.label:   {[[]]}
    leadFieldFile: ''
>> mobilab.allStreams.item{3}

```

ans =

Class: markerStream

Properties:

```

    name:          unity_markers_prederror_BPN-C043
    uuid:          720cf29e-b4ca-411d-8326-af3ffc867cd7
    samplingRate:  9.428032e-01 Hz
    timeStamp:     <1x535 double>
    numberOfChannels: 1
    data:          <535x1 int16>
    artifactMask:  <535x1 sparse>
    event.latencyInFrame: <1x535 double>
    event.label:    <535x1 cell>
    label:         <1x1 cell>
    unit:          unknown
    owner.name:    Lukas
    owner.organization: SCCN
    owner.email:   Lukas@sccn.ucsd.edu
    sessionUUID:   adb1ff0a-520f-47f9-92ef-0195dc2ba244
    auxChannel.label: <0x1 cell>
    auxChannel.data: <535x0 int16>
    history:       mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-

```



```
EMS_VR\data\0_raw_data\1\block_Vibro.xdf','P:\Project_Sezen-
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
>> mobilab.allStreams.item{2}
```

```
ans =
```

```
Class: mocapRigidBody
```

```
Properties:
```

```
name: rigid_leap_hand_BPN-C043
uuid: 2c0cb3f6-8472-4c0a-907a-dd5e31098411
samplingRate: 90 Hz
timeStamp: <1x51754 double>
numberOfChannels: 7
data: <51754x7 double>
artifactMask: <51754x7 sparse>
event.latencyInFrame: <1x0 double>
event.label: <0x1 cell>
label: <7x1 cell>
unit: meters
owner.name: Lukas
owner.organization: SCCN
owner.email: Lukas@sccn.ucsd.edu
sessionUUID: adb1ff0a-520f-47f9-92ef-0195dc2ba244
auxChannel.label: <0x1 cell>
auxChannel.data: <51754x0 double>
history: mobilab.allStreams=dataSourceXDF('P:\Project_Sezen-
EMS_VR\data\0_raw_data\1\block_Vibro.xdf','P:\Project_Sezen-
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI');
>> disp(['Subject #' num2str(subject)]);
```

```
%%%%%%%%%
```

```
% processing Hand
```

```
% TODO: determine mobi stream index
```

```
input_filepath = [input_path num2str(subject) '\' filenames{3} '_MoBI']; % TODO do
for both files
```

```
mobilab.allStreams = dataSourceMoBI(input_filepath);
```

```
Subject #1
```

```
Error using dataSourceMoBI (line 30)
```

```
The folder is empty.
```

```
>> input_path = [study_folder mobi_data_folder];
```

```
>> disp(['Subject #' num2str(subject)]);
```

```
%%%%%%%%%
```

```
% processing Hand
```

```
% TODO: determine mobi stream index
```

```
input_filepath = [input_path num2str(subject) '\' filenames{3} '_MoBI']; % TODO do
for both files
```

```
mobilab.allStreams = dataSourceMoBI(input_filepath);
```

```
Subject #1
```

```
Error using dataSourceMoBI (line 30)
```

```
The folder is empty.
```

```
>> file = filenames(2)

file =

    cell

    'block_Vibro'

>> input_filepath = [input_path num2str(subject) '\\' file '_MoBI']; % TODO do for both files
mobilab.allStreams = dataSourceMoBI(input_filepath);
Error using dir
Function is not defined for 'cell' inputs.

Error in dataSourceMoBI (line 25)
    folder = dir(mobiDataDirectory);

>> input_path

input_path =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'

>> subject1
Undefined function or variable 'subject1'.

Did you mean:
>> subject = 1

subject =

    1

>> file

file =

    cell

    'block_Vibro'

>> input_filepath = [input_path num2str(subject) '\\' file '_MoBI']; % TODO do for both files
>> input_filepath

input_filepath =

    1x5 cell array

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'    '1'    '\\'    'block_Vibro'    '_MoBI'

>> input_filepath = [input_path num2str(subject) '\\' filenames{1} '_MoBI']; % TODO do for both files
```

```
>> input_filepath

input_filepath =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\1\block_Training_MoBI'

>> file = 1

file =

    1

>> input_filepath = [input_path num2str(subject) '\\' filenames{file} '_MoBI']; % TODO
do for both files
>> input_filepath

input_filepath =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\1\block_Training_MoBI'

>> file = 2

file =

    2

>> input_filepath = [input_path num2str(subject) '\\' filenames{file} '_MoBI']; % TODO
do for both files
    mobilab.allStreams = dataSourceMoBI(input_filepath);
Adding object: brainvision_rda_bpn-c012
Adding object: rigid_leap_hand_BPN-C043
Adding object: unity_markers_prederror_BPN-C043
>> % quaternion values sometimes flip their sign for mathematical reasons
    % which is bad for filtering. but the values stay the same if they are
    % flipped back, so this is done to allow filtering.
    unflip = mobilab.allStreams.item{2}.unflipSigns();

    % lowpass filtering with the specified cutoff frequency
    lowpass = unflip.lowpass(6);

    % quaternion orientation values are transformed to euler angle to be
    % interpretable for humans
    euler = lowpass.quaternionsToEuler();

    % 3 time derivatives are calculated (velocity, acceleration, and jerk)
    euler.timeDerivative(3);
'fallocate' is not recognized as an internal or external command,
operable program or batch file.
Adding object: unflip_rigid_leap_hand_BPN-C043
Running:
    mobilab.allStreams.item{2}.unflipSigns('channels',[1 2 3 4 5 6 7]);
Third argument must be the length of the filter (integer type). Using the default:
128.
'fallocate' is not recognized as an internal or external command,
operable program or batch file.
```

```

Adding object: filt_unflip_rigid_leap_hand_BPN-C043
'fallocate' is not recognized as an internal or external command,
operable program or batch file.
Adding object: quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Running:
    mobilab.allStreams.item{5}.quaternionsToEuler('newChannels',[1 2 3 4 5 6]);
Saving label in: P:\Project_Sezen-✓
EMS_VR\data\1_mobi_data\1_block_Vibro_MoBI\quat2eul_filt_unflip_rigid_leap_hand_BPN-✓
C043_3d9be14d-fa66-40ba-b29b-8a8422b0fa0c_adb1ff0a-520f-47f9-92ef-0195dc2ba244.hdr
Adding object: vel_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Adding object: acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Adding object: jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
>> input_path = [study_folder mobi_data_folder];
output_path = [study_folder raw_EEGLAB_data_folder];

if ~exist('ALLEEG','var'); eeglab; end
if ~exist('mobilab','var'); runmobilab; end
>> input_path

input_path =

    'P:\Project_Sezen-EMS_VR\data\1_mobi_data\'

>> output_path

output_path =

    'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\'

>> disp(['Subject #' num2str(subject)]);
    output_filepath = [output_path num2str(subject)];
    mkdir(output_filepath); % create folder
Subject #1
>> file = 2

file =

    2

>> % TODO: get filenames in folder to save the files with postfix _MOBI
    input_filepath = [input_path num2str(subject) '\\' filenames{file} '_MoBI'];
    mobilab.allStreams = dataSourceMoBI(input_filepath);

    disp(['Exporting Subject #' num2str(subject) ': ' filenames{file}✓
'_MoBI...']);
Adding object: acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Adding object: brainvision_rda_bpn-c012
Adding object: filt_unflip_rigid_leap_hand_BPN-C043
Adding object: jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Adding object: quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Adding object: rigid_leap_hand_BPN-C043
Adding object: unflip_rigid_leap_hand_BPN-C043
Adding object: unity_markers_prederror_BPN-C043
Adding object: vel_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
Exporting Subject #1: block_Vibro_MoBI...

```

```
>> mobilab.gui
```

```
Warning: A value of class "javax.swing.ToolTipManager" was indexed with no subscripts
specified. Currently the result of this operation is the indexed value itself, but in
a future release, it will be an error.
```

```
> In mobilabApplication/gui (line 392)
```

```
ans =
```

```
Figure (mobilabApplicationGUI) with properties:
```

```
Number: 2
Name: 'MoBILAB'
Color: [0.9300 0.9600 1]
Position: [1000 936 418 400]
Units: 'pixels'
```

```
Show all properties
```

```
Class: mocapRigidBody
```

```
Properties:
```

```
name: jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043
uuid: 321f05de-a2e1-4f61-9eca-8dfea277d00c
samplingRate: 90 Hz
timeStamp: <1x51754 double>
numberOfChannels: 6
data: <51754x6 double>
artifactMask: <51754x6 sparse>
event.latencyInFrame: <1x0 double>
event.label: <0x1 cell>
label: <6x1 cell>
unit: meters
owner.name: Lukas
owner.organization: SCCN
owner.email: Lukas@sccn.ucsd.edu
sessionUUID: adblff0a-520f-47f9-92ef-0195dc2ba244
auxChannel.label: <0x1 cell>
auxChannel.data: <51754x0 double>
history: mobilab.allStreams.item{6}.timeDerivative( [3] , [6] , [1 2 3
4 5 6]);
```

```
>> % TODO: add correct streams to export, see mobilab export loop
```

```
EEG = mobilab.allStreams.export2eeglab( [1 2 7 8 9] , [3]);
```

```
% the exported data set has the suffix _MoBI, since it contains both
% brain and body imaging data
```

```
EEG = pop_save( EEG, 'filename', [filenames{file} '_MoBI'], 'filepath',
output_filepath);
disp('...done');
```

```
Inserting 437 events.
```

```
eeg_checkset note: upper time limit (xmax) adjusted so (xmax-xmin)*srate+1 = number of
frames
```

```
Saving dataset...
```

```
pop_loadset(): loading file P:\Project_Sezen-
```

```
EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI\brainvision_rda_bpn-c012.set ...
```

```
Reading float file 'P:\Project_Sezen-
```

EMS_VR\data\1_mobi_data\1\block_Vibro_MoBI\brainvision_rda_bpn-c012.fdt'...

Saving dataset...

...done

>> eeglab **redraw**

>> EEG

EEG =

struct with fields:

```

    setname: 'brainvision_rda_bpn-c012'
    filename: 'block_Vibro_MoBI.set'
    filepath: 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1'
    subject: ''
    group: ''
    condition: ''
    session: []
    comments: ''
    nbchan: 89
    trials: 1
        pnts: 1455330
        srate: 2500
        xmin: 0
        xmax: 582.1316
        times: [1×1455330 double]
        data: [89×1455330×1 mmo]
    icaact: []
    icawinv: []
    icasphere: []
    icaweights: []
    icachansind: []
    chanlocs: [89×1 struct]
    urchanlocs: []
    chaninfo: [1×1 struct]
        ref: 'common'
        event: [1×437 struct]
        urevent: [1×437 struct]
    eventdescription: {'' '' '' '' ''}
    epoch: []
    epochdescription: {}
        reject: [1×1 struct]
        stats: [1×1 struct]
    specdata: []
    specicaact: []
    splinefile: ''
    icasplinefile: ''
    dipfit: []
    history: '←EEG.etc.eeglabvers = '14.1.0'; % this tracks which version of
EEGLAB is being used, you may ignore it←EEG = eeg_checkset( EEG );←pop_eegplot( EEG,←
1, 1, 1);'
        saved: 'justloaded'
        etc: [1×1 struct]
    datfile: 'block_Vibro_MoBI.fdt'

```

>> EEG

EEG =

struct with fields:

```

    setname: 'brainvision_rda_bpn-c012'
    filename: 'block_Vibro_MoBI.set'
    filepath: 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1'
    subject: ''
    group: ''
    condition: ''
    session: []
    comments: ''
    nbchan: 89
    trials: 1
    pnts: 1455330
    srate: 2500
    xmin: 0
    xmax: 582.1316
    times: [1x1455330 double]
    data: [89x1455330x1 mmo]
    icaact: []
    icawinv: []
    icasphere: []
    icaweights: []
    icachansind: []
    chanlocs: [89x1 struct]
    urchanlocs: []
    chaninfo: [1x1 struct]
    ref: 'common'
    event: [1x437 struct]
    urevent: [1x437 struct]
    eventdescription: {'' '' '' '' ''}
    epoch: []
    epochdescription: {}
    reject: [1x1 struct]
    stats: [1x1 struct]
    specdata: []
    specicaact: []
    splinefile: ''
    icasplinefile: ''
    dipfit: []
    history: '←EEG.etc.eeglabvers = '14.1.0'; % this tracks which version of
EEGLAB is being used, you may ignore it←EEG = eeg_checkset( EEG );←pop_eegplot( EEG,
1, 1, 1);'
    saved: 'justloaded'
    etc: [1x1 struct]
    datfile: 'block_Vibro_MoBI.fdt'

```

```

>> EEG.data
    1.0e+08 *

```

Columns 1 through 24

Operation terminated by user during mmo/display (line 25)

```
>> EEG.srate
```

```
ans =
```

```
2500
```

```
>> EEG.event
```

```
ans =
```

```
1×437 struct array with fields:
```

```
type  
latency  
duration  
urevent  
hedTag
```

```
>> EEG.event(1)
```

```
ans =
```

```
struct with fields:
```

```
type: 'box:spawned;condition:vibro;trial_nr:1;normal_or_conflict:0;cube:✓  
CubeMiddle (UnityEngine.GameObject);isi_time:1.893688'  
latency: 15647  
duration: 0  
urevent: 1  
hedTag: 'box:spawned;condition:vibro;trial_nr:1;normal_or_conflict:0;cube:✓  
CubeMiddle (UnityEngine.GameObject);isi_time:1.893688'
```

```
>> 15000/2500
```

```
ans =
```

```
6
```

```
>> EEG.event(400).type
```

```
ans =
```

```
'box:touched;condition:vibro;vibroFeedback:on;reaction_time:2.639322;trial_nr:1;✓  
normal_or_conflict:0;cube::;isi_time:1.110559;vibro_duration:-2.4'
```

```
>> EEG
```

```
EEG =
```

```
struct with fields:
```

```
setname: 'brainvision_rda_bpn-c012'  
filename: 'block_Vibro_MoBI.set'  
filepath: 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1'
```



```

        subject: ''
        group: ''
    condition: ''
        session: []
    comments: ''
        nbchan: 89
        trials: 1
            pnts: 1455330
            srate: 2500
            xmin: 0
            xmax: 582.1316
            times: [1×1455330 double]
            data: [89×1455330×1 mmo]
        icaact: []
        icawinv: []
        icasphere: []
        icaweights: []
    icachansind: []
        chanlocs: [89×1 struct]
    urchanlocs: []
        chaninfo: [1×1 struct]
            ref: 'common'
            event: [1×437 struct]
            urevent: [1×437 struct]
eventdescription: {'' '' '' '' ''}
        epoch: []
epochdescription: {}
        reject: [1×1 struct]
        stats: [1×1 struct]
        specdata: []
        specicaact: []
        splinefile: ''
    icasplinefile: ''
        dipfit: []
        history: '←EEG.etc.eeglabvers = '14.1.0'; % this tracks which version of
EEGLAB is being used, you may ignore it←EEG = eeg_checkset( EEG );←pop_eegplot( EEG,
1, 1, 1);'
        saved: 'justloaded'
        etc: [1×1 struct]
        datfile: 'block_Vibro_MoBI.fdt'

```

```
>> EEG.etc
```

```
ans =
```

```
struct with fields:
```

```

    desc: [1×1 struct]
    mobi: [1×1 struct]
eeglabvers: '14.1.0'

```

```
>> EEG.event
```

```
ans =
```

1×437 struct array with fields:

```
type
latency
duration
urevent
hedTag
```

>> EEG

EEG =

struct with fields:

```
    setname: 'brainvision_rda_bpn-c012'
    filename: 'block_Vibro_MoBI.set'
    filepath: 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1'
    subject: ''
    group: ''
    condition: ''
    session: []
    comments: ''
    nbchan: 89
    trials: 1
    pnts: 1455330
    srates: 2500
    xmin: 0
    xmax: 582.1316
    times: [1×1455330 double]
    data: [89×1455330×1 mmo]
    icaact: []
    icawinv: []
    icasphere: []
    icaweights: []
    icachansind: []
    chanlocs: [89×1 struct]
    urchanlocs: []
    chaninfo: [1×1 struct]
        ref: 'common'
        event: [1×437 struct]
        urevent: [1×437 struct]
    eventdescription: {'' '' '' '' ''}
    epoch: []
    epochdescription: {}
    reject: [1×1 struct]
    stats: [1×1 struct]
    specdata: []
    specicaact: []
    splinefile: ''
    icasplinefile: ''
    dipfit: []
    history: '←EEG.etc.eeglabvers = '14.1.0'; % this tracks which version of
EEGLAB is being used, you may ignore it←EEG = eeg_checkset( EEG );←pop_eegplot( EEG,
1, 1, 1);'
    saved: 'justloaded'
```

```
        etc: [1x1 struct]  
    datfile: 'block_Vibro_MoBI.fdt'
```

```
>> EEG.chaninfo
```

```
ans =
```

```
    struct with fields:
```

```
        nosedir: '+X'  
        plotrad: []  
        shrink: []  
        nodatchans: []  
        icachansind: []
```

```
>> EEG.chanlocs
```

```
ans =
```

```
    89x1 struct array with fields:
```

```
    labels  
    type  
    X  
    Y  
    Z  
    radius  
    theta  
    ref  
    sph_theta  
    sph_phi  
    sph_radius  
    urchan
```

```
>> EEG.chanlocs.labels
```

```
ans =
```

```
    'brainvision_rda_bpn-c012_Fp1'
```

```
ans =
```

```
    'brainvision_rda_bpn-c012_Fp2'
```

```
ans =
```

```
    'brainvision_rda_bpn-c012_F7'
```

```
ans =
```

```
    'brainvision_rda_bpn-c012_F3'
```

ans =

'brainvision_rda_bpn-c012_Fz'

ans =

'brainvision_rda_bpn-c012_F4'

ans =

'brainvision_rda_bpn-c012_F8'

ans =

'brainvision_rda_bpn-c012_FC5'

ans =

'brainvision_rda_bpn-c012_FC1'

ans =

'brainvision_rda_bpn-c012_FC2'

ans =

'brainvision_rda_bpn-c012_FC6'

ans =

'brainvision_rda_bpn-c012_C7'

ans =

'brainvision_rda_bpn-c012_C3'

ans =

'brainvision_rda_bpn-c012_Cz'

ans =

'brainvision_rda_bpn-c012_C4'

ans =

'brainvision_rda_bpn-c012_C8'

ans =

'brainvision_rda_bpn-c012_TP9'

ans =

'brainvision_rda_bpn-c012_CP5'

ans =

'brainvision_rda_bpn-c012_CP1'

ans =

'brainvision_rda_bpn-c012_CP2'

ans =

'brainvision_rda_bpn-c012_CP6'

ans =

'brainvision_rda_bpn-c012_TP10'

ans =

'brainvision_rda_bpn-c012_P7'

ans =

'brainvision_rda_bpn-c012_P3'

ans =

'brainvision_rda_bpn-c012_Pz'

ans =

'brainvision_rda_bpn-c012_P4'

ans =

'brainvision_rda_bpn-c012_P8'

ans =

'brainvision_rda_bpn-c012_PO9'

ans =

'brainvision_rda_bpn-c012_O1'

ans =

'brainvision_rda_bpn-c012_Oz'

ans =

'brainvision_rda_bpn-c012_O2'

ans =

'brainvision_rda_bpn-c012_PO10'

ans =

'brainvision_rda_bpn-c012_AF7'

ans =

'brainvision_rda_bpn-c012_AF3'

ans =

'brainvision_rda_bpn-c012_AF4'

ans =

'brainvision_rda_bpn-c012_AF8'

ans =

'brainvision_rda_bpn-c012_F5'

ans =

'brainvision_rda_bpn-c012_F1'

ans =

'brainvision_rda_bpn-c012_F2'

ans =

'brainvision_rda_bpn-c012_F6'

ans =

'brainvision_rda_bpn-c012_FT9'

ans =

'brainvision_rda_bpn-c012_FT7'

ans =

'brainvision_rda_bpn-c012_VC3'

ans =

'brainvision_rda_bpn-c012_FC4'

ans =

'brainvision_rda_bpn-c012_FT8'

ans =

'brainvision_rda_bpn-c012_FT10'

ans =

'brainvision_rda_bpn-c012_C5'

ans =

'brainvision_rda_bpn-c012_C1'

```
ans =
```

```
'brainvision_rda_bpn-c012_C2'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_C6'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_TP7'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_CP3'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_CPz'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_CP4'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_TP8'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_P5'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_P1'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_P2'
```

```
ans =
```

```
'brainvision_rda_bpn-c012_P6'
```


ans =

'brainvision_rda_bpn-c012_PO7'

ans =

'brainvision_rda_bpn-c012_PO3'

ans =

'brainvision_rda_bpn-c012_POz'

ans =

'brainvision_rda_bpn-c012_PO4'

ans =

'brainvision_rda_bpn-c012_PO8'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_X'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Y'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Z'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_quat_X'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_quat_Y'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_quat_Z'

ans =

'rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_quat_W'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_X'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Y'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Z'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Yaw'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Pitch'

ans =

'acc_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Roll'

ans =

'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_X'

ans =

'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Y'

ans =

'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Z'

ans =

'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Yaw'

ans =

```
'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Pitch'
```

ans =

```
'jerk_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Euler_Roll'
```

ans =

```
'vel quat2eul filt unflip rigid leap hand BPN-C043 Rigid Leap Hand X'
```

ans =

```
'vel_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Y'
```

ans =

```
'vel_quat2eul_filt_unflip_rigid_leap_hand_BPN-C043_Rigid_Leap_Hand_Z'
```

ans =

```
'vel quat2eul filt unflip rigid leap hand BPN-C043 Rigid Leap Hand Euler Yaw'
```

ans =

```
'vel quat2eul filt unflip rigid leap hand BPN-C043 Rigid Leap Hand Euler Pitch'
```

ans =

```
'vel quat2eul filt unflip rigid leap hand BPN-C043 Rigid Leap Hand Euler Roll'
```

Removing 25 channel(s)...

```
Warning: memory mapped object writing might not be up to date in cache on network ✓
driveCreating a new ALLEEG dataset 2
```

Done .

```
>> eeqh
```

```
[ALLEEG EEG CURRENTSET ALLCOM] = eeqlab;
```

```
[EEG ALLEEG CURRENTSET] = eeg_retrieve(ALLEEG,1);
```

```
EEG = eeg_checkset( EEG );
```

```
pop eegplot( EEG, 1, 1, 1);
```

```
EEG = eeg_checkset( EEG );
```

```
EEG = pop_select( EEG,'channel',{'brainvision_rda_bpn-c012_Fp1' 'brainvision_rda_bpn-c012_Fp2' 'brainvision_rda_bpn-c012_F7' 'brainvision_rda_bpn-c012_F3'})
```

```
'brainvision_rda_bpn-c012_Fz' 'brainvision_rda_bpn-c012_F4' 'brainvision_rda_bpn-c012_F8' 'brainvision_rda_bpn-c012_FC5' 'brainvision_rda_bpn-c012_FC1'
```

```
'brainvision rda bpn-c012 FC2' 'brainvision rda bpn-c012 FC6' 'brainvision rda bpn-c012 FC8'
```

```

c012_C7' 'brainvision_rda_bpn-c012_C3' 'brainvision_rda_bpn-c012_Cz'↵
'brainvision_rda_bpn-c012_C4' 'brainvision_rda_bpn-c012_C8' 'brainvision_rda_bpn-↵
c012_TP9' 'brainvision_rda_bpn-c012_CP5' 'brainvision_rda_bpn-c012_CP1'↵
'brainvision_rda_bpn-c012_CP2' 'brainvision_rda_bpn-c012_CP6' 'brainvision_rda_bpn-↵
c012_TP10' 'brainvision_rda_bpn-c012_P7' 'brainvision_rda_bpn-c012_P3'↵
'brainvision_rda_bpn-c012_Pz' 'brainvision_rda_bpn-c012_P4' 'brainvision_rda_bpn-↵
c012_P8' 'brainvision_rda_bpn-c012_PO9' 'brainvision_rda_bpn-c012_O1'↵
'brainvision_rda_bpn-c012_Oz' 'brainvision_rda_bpn-c012_O2' 'brainvision_rda_bpn-↵
c012_PO10' 'brainvision_rda_bpn-c012_AF7' 'brainvision_rda_bpn-c012_AF3'↵
'brainvision_rda_bpn-c012_AF4' 'brainvision_rda_bpn-c012_AF8' 'brainvision_rda_bpn-↵
c012_F5' 'brainvision_rda_bpn-c012_F1' 'brainvision_rda_bpn-c012_F2'↵
'brainvision_rda_bpn-c012_F6' 'brainvision_rda_bpn-c012_FT9' 'brainvision_rda_bpn-↵
c012_FT7' 'brainvision_rda_bpn-c012_VC3' 'brainvision_rda_bpn-c012_FC4'↵
'brainvision_rda_bpn-c012_FT8' 'brainvision_rda_bpn-c012_FT10' 'brainvision_rda_bpn-↵
c012_C5' 'brainvision_rda_bpn-c012_C1' 'brainvision_rda_bpn-c012_C2'↵
'brainvision_rda_bpn-c012_C6' 'brainvision_rda_bpn-c012_TP7' 'brainvision_rda_bpn-↵
c012_CP3' 'brainvision_rda_bpn-c012_CPz' 'brainvision_rda_bpn-c012_CP4'↵
'brainvision_rda_bpn-c012_TP8' 'brainvision_rda_bpn-c012_P5' 'brainvision_rda_bpn-↵
c012_P1' 'brainvision_rda_bpn-c012_P2' 'brainvision_rda_bpn-c012_P6'↵
'brainvision_rda_bpn-c012_PO7' 'brainvision_rda_bpn-c012_PO3' 'brainvision_rda_bpn-↵
c012_POz' 'brainvision_rda_bpn-c012_PO4' 'brainvision_rda_bpn-c012_PO8'}});
[ALLEEG, EEG, CURRENTSET] = eeg_store( ALLEEG, EEG, 0 );
>> file = 2

```

```
file =
```

```
2
```

```

>> input_path = [study_folder raw_EEGLAB_data_folder];
output_path = input_path;
if ~exist('ALLEEG','var'); eeglab; end
pop_editoptions( 'option_storedisk', 0, 'option_savetwofiles', 1,↵
'option_saveversion6', 0, 'option_single', 0, 'option_memmapdata', 0,↵
'option_eegobject', 0, 'option_computeica', 1, 'option_scaleicarms', 1,↵
'option_rememberfolder', 1, 'option_donotusetoolboxes', 0, 'option_checkversion', 1,↵
'option_chat', 1);
Using option file in directory C:\Users\Lukas
>> subject

```

```
subject =
```

```
1
```

```

>> disp(['Subject #' num2str(subject)]);

input_filepath = [input_path num2str(subject)];
output_filepath = [output_path num2str(subject) '\'];

% clean EEGLAB before each iteration
STUDY = []; CURRENTSTUDY = 0; ALLEEG = []; EEG=[]; CURRENTSET=[];

% load the _MoBI set which has EEG and mocap data combined
EEG = pop_loadset('filename', [filenames{file} '.set'], 'filepath',↵
input_filepath);
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 0, 'study', 0);

```

```

EEG = eeg_checkset( EEG );

% split it and keep just the EEG channels, save it with _EEG suffix

% check correct channels to kick out.
% TODO add standard channel names of 64 electrode layout
EEG = pop_select( EEG, 'channel', {'brainvision_rda_bpn-c012_Fp1'↵
'brainvision_rda_bpn-c012_Fp2' 'brainvision_rda_bpn-c012_F7' 'brainvision_rda_bpn-↵
c012_F3' 'brainvision_rda_bpn-c012_Fz' 'brainvision_rda_bpn-c012_F4'↵
'brainvision_rda_bpn-c012_F8' 'brainvision_rda_bpn-c012_FC5' 'brainvision_rda_bpn-↵
c012_FC1' 'brainvision_rda_bpn-c012_FC2' 'brainvision_rda_bpn-c012_FC6'↵
'brainvision_rda_bpn-c012_C7' 'brainvision_rda_bpn-c012_C3' 'brainvision_rda_bpn-↵
c012_Cz' 'brainvision_rda_bpn-c012_C4' 'brainvision_rda_bpn-c012_C8'↵
'brainvision_rda_bpn-c012_TP9' 'brainvision_rda_bpn-c012_CP5' 'brainvision_rda_bpn-↵
c012_CP1' 'brainvision_rda_bpn-c012_CP2' 'brainvision_rda_bpn-c012_CP6'↵
'brainvision_rda_bpn-c012_TP10' 'brainvision_rda_bpn-c012_P7' 'brainvision_rda_bpn-↵
c012_P3' 'brainvision_rda_bpn-c012_Pz' 'brainvision_rda_bpn-c012_P4'↵
'brainvision_rda_bpn-c012_P8' 'brainvision_rda_bpn-c012_PO9' 'brainvision_rda_bpn-↵
c012_O1' 'brainvision_rda_bpn-c012_Oz' 'brainvision_rda_bpn-c012_O2'↵
'brainvision_rda_bpn-c012_PO10' 'brainvision_rda_bpn-c012_AF7' 'brainvision_rda_bpn-↵
c012_AF3' 'brainvision_rda_bpn-c012_AF4' 'brainvision_rda_bpn-c012_AF8'↵
'brainvision_rda_bpn-c012_F5' 'brainvision_rda_bpn-c012_F1' 'brainvision_rda_bpn-↵
c012_F2' 'brainvision_rda_bpn-c012_F6' 'brainvision_rda_bpn-c012_FT9'↵
'brainvision_rda_bpn-c012_FT7' 'brainvision_rda_bpn-c012_VC3' 'brainvision_rda_bpn-↵
c012_FC4' 'brainvision_rda_bpn-c012_FT8' 'brainvision_rda_bpn-c012_FT10'↵
'brainvision_rda_bpn-c012_C5' 'brainvision_rda_bpn-c012_C1' 'brainvision_rda_bpn-↵
c012_C2' 'brainvision_rda_bpn-c012_C6' 'brainvision_rda_bpn-c012_TP7'↵
'brainvision_rda_bpn-c012_CP3' 'brainvision_rda_bpn-c012_CPz' 'brainvision_rda_bpn-↵
c012_CP4' 'brainvision_rda_bpn-c012_TP8' 'brainvision_rda_bpn-c012_P5'↵
'brainvision_rda_bpn-c012_P1' 'brainvision_rda_bpn-c012_P2' 'brainvision_rda_bpn-↵
c012_P6' 'brainvision_rda_bpn-c012_PO7' 'brainvision_rda_bpn-c012_PO3'↵
'brainvision_rda_bpn-c012_POz' 'brainvision_rda_bpn-c012_PO4' 'brainvision_rda_bpn-↵
c012_PO8'}});

[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1, 'setname', [filenames{file}↵
'_EEG'], 'savenew', [output_filepath filenames{file} '_EEG.set'], 'gui', 'off');

% go back to the first data set
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 2, 'retrieve', 1, 'study', 0);
EEG = eeg_checkset( EEG );

% split it and kick all EEG data channels out, save it with _mocap suffix
% TODO add standard channel names of 64 electrode layout
EEG = pop_select( EEG, 'nochannel', {'brainvision_rda_bpn-c012_Fp1'↵
'brainvision_rda_bpn-c012_Fp2' 'brainvision_rda_bpn-c012_F7' 'brainvision_rda_bpn-↵
c012_F3' 'brainvision_rda_bpn-c012_Fz' 'brainvision_rda_bpn-c012_F4'↵
'brainvision_rda_bpn-c012_F8' 'brainvision_rda_bpn-c012_FC5' 'brainvision_rda_bpn-↵
c012_FC1' 'brainvision_rda_bpn-c012_FC2' 'brainvision_rda_bpn-c012_FC6'↵
'brainvision_rda_bpn-c012_C7' 'brainvision_rda_bpn-c012_C3' 'brainvision_rda_bpn-↵
c012_Cz' 'brainvision_rda_bpn-c012_C4' 'brainvision_rda_bpn-c012_C8'↵
'brainvision_rda_bpn-c012_TP9' 'brainvision_rda_bpn-c012_CP5' 'brainvision_rda_bpn-↵
c012_CP1' 'brainvision_rda_bpn-c012_CP2' 'brainvision_rda_bpn-c012_CP6'↵
'brainvision_rda_bpn-c012_TP10' 'brainvision_rda_bpn-c012_P7' 'brainvision_rda_bpn-↵
c012_P3' 'brainvision_rda_bpn-c012_Pz' 'brainvision_rda_bpn-c012_P4'↵
'brainvision_rda_bpn-c012_P8' 'brainvision_rda_bpn-c012_PO9' 'brainvision_rda_bpn-↵

```

```

c012_O1' 'brainvision_rda_bpn-c012_Oz' 'brainvision_rda_bpn-c012_O2'↵
'brainvision_rda_bpn-c012_PO10' 'brainvision_rda_bpn-c012_AF7' 'brainvision_rda_bpn-↵
c012_AF3' 'brainvision_rda_bpn-c012_AF4' 'brainvision_rda_bpn-c012_AF8'↵
'brainvision_rda_bpn-c012_F5' 'brainvision_rda_bpn-c012_F1' 'brainvision_rda_bpn-↵
c012_F2' 'brainvision_rda_bpn-c012_F6' 'brainvision_rda_bpn-c012_FT9'↵
'brainvision_rda_bpn-c012_FT7' 'brainvision_rda_bpn-c012_VC3' 'brainvision_rda_bpn-↵
c012_FC4' 'brainvision_rda_bpn-c012_FT8' 'brainvision_rda_bpn-c012_FT10'↵
'brainvision_rda_bpn-c012_C5' 'brainvision_rda_bpn-c012_C1' 'brainvision_rda_bpn-↵
c012_C2' 'brainvision_rda_bpn-c012_C6' 'brainvision_rda_bpn-c012_TP7'↵
'brainvision_rda_bpn-c012_CP3' 'brainvision_rda_bpn-c012_CPz' 'brainvision_rda_bpn-↵
c012_CP4' 'brainvision_rda_bpn-c012_TP8' 'brainvision_rda_bpn-c012_P5'↵
'brainvision_rda_bpn-c012_P1' 'brainvision_rda_bpn-c012_P2' 'brainvision_rda_bpn-↵
c012_P6' 'brainvision_rda_bpn-c012_PO7' 'brainvision_rda_bpn-c012_PO3'↵
'brainvision_rda_bpn-c012_POz' 'brainvision_rda_bpn-c012_PO4' 'brainvision_rda_bpn-↵
c012_PO8'}});

[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1,'setname', [filenames↵
{file} '_mocap'],'savenew',[output_filepath filenames{file} '_mocap.↵
set'],'gui','off');
Subject #1
pop_loadset(): loading file P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1\block_Vibro.↵
set ...
Error using load
Unable to read file 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1\block_Vibro.set'. No↵
such file or directory.

Error in pop_loadset (line 108)
    TMPVAR = load('-mat', filename);

>> disp(['Subject #' num2str(subject)]);

input_filepath = [input_path num2str(subject)];
output_filepath = [output_path num2str(subject) '\'];

% clean EEGLAB before each iteration
STUDY = []; CURRENTSTUDY = 0; ALLEEG = []; EEG=[]; CURRENTSET=[];

% load the _MoBI set which has EEG and mocap data combined
EEG = pop_loadset('filename', [filenames{file} '_MoBI.set'], 'filepath',↵
input_filepath);
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 0,'study',0);
EEG = eeg_checkset( EEG );

% split it and keep just the EEG channels, save it with _EEG suffix

% check correct channels to kick out.
% TODO add standard channel names of 64 electrode layout
EEG = pop_select( EEG,'channel',{'brainvision_rda_bpn-c012_Fp1'↵
'brainvision_rda_bpn-c012_Fp2' 'brainvision_rda_bpn-c012_F7' 'brainvision_rda_bpn-↵
c012_F3' 'brainvision_rda_bpn-c012_Fz' 'brainvision_rda_bpn-c012_F4'↵
'brainvision_rda_bpn-c012_F8' 'brainvision_rda_bpn-c012_FC5' 'brainvision_rda_bpn-↵
c012_FC1' 'brainvision_rda_bpn-c012_FC2' 'brainvision_rda_bpn-c012_FC6'↵
'brainvision_rda_bpn-c012_C7' 'brainvision_rda_bpn-c012_C3' 'brainvision_rda_bpn-↵
c012_Cz' 'brainvision_rda_bpn-c012_C4' 'brainvision_rda_bpn-c012_C8'↵
'brainvision_rda_bpn-c012_TP9' 'brainvision_rda_bpn-c012_CP5' 'brainvision_rda_bpn-↵

```

```

c012_CP1' 'brainvision_rda_bpn-c012_CP2' 'brainvision_rda_bpn-c012_CP6'↵
'brainvision_rda_bpn-c012_TP10' 'brainvision_rda_bpn-c012_P7' 'brainvision_rda_bpn-↵
c012_P3' 'brainvision_rda_bpn-c012_Pz' 'brainvision_rda_bpn-c012_P4'↵
'brainvision_rda_bpn-c012_P8' 'brainvision_rda_bpn-c012_PO9' 'brainvision_rda_bpn-↵
c012_O1' 'brainvision_rda_bpn-c012_Oz' 'brainvision_rda_bpn-c012_O2'↵
'brainvision_rda_bpn-c012_PO10' 'brainvision_rda_bpn-c012_AF7' 'brainvision_rda_bpn-↵
c012_AF3' 'brainvision_rda_bpn-c012_AF4' 'brainvision_rda_bpn-c012_AF8'↵
'brainvision_rda_bpn-c012_F5' 'brainvision_rda_bpn-c012_F1' 'brainvision_rda_bpn-↵
c012_F2' 'brainvision_rda_bpn-c012_F6' 'brainvision_rda_bpn-c012_FT9'↵
'brainvision_rda_bpn-c012_FT7' 'brainvision_rda_bpn-c012_VC3' 'brainvision_rda_bpn-↵
c012_FC4' 'brainvision_rda_bpn-c012_FT8' 'brainvision_rda_bpn-c012_FT10'↵
'brainvision_rda_bpn-c012_C5' 'brainvision_rda_bpn-c012_C1' 'brainvision_rda_bpn-↵
c012_C2' 'brainvision_rda_bpn-c012_C6' 'brainvision_rda_bpn-c012_TP7'↵
'brainvision_rda_bpn-c012_CP3' 'brainvision_rda_bpn-c012_CPz' 'brainvision_rda_bpn-↵
c012_CP4' 'brainvision_rda_bpn-c012_TP8' 'brainvision_rda_bpn-c012_P5'↵
'brainvision_rda_bpn-c012_P1' 'brainvision_rda_bpn-c012_P2' 'brainvision_rda_bpn-↵
c012_P6' 'brainvision_rda_bpn-c012_PO7' 'brainvision_rda_bpn-c012_PO3'↵
'brainvision_rda_bpn-c012_POz' 'brainvision_rda_bpn-c012_PO4' 'brainvision_rda_bpn-↵
c012_PO8'}});

[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1,'setname',[filenames{file}↵
'_EEG'], 'savenew',[output_filepath filenames{file} '_EEG.set'],'gui','off');

% go back to the first data set
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 2,'retrieve',1,'study',0);
EEG = eeg_checkset( EEG );

% split it and kick all EEG data channels out, save it with _mocap suffix
% TODO add standard channel names of 64 electrode layout
EEG = pop_select( EEG,'nochannel',{'brainvision_rda_bpn-c012_Fp1'↵
'brainvision_rda_bpn-c012_Fp2' 'brainvision_rda_bpn-c012_F7' 'brainvision_rda_bpn-↵
c012_F3' 'brainvision_rda_bpn-c012_Fz' 'brainvision_rda_bpn-c012_F4'↵
'brainvision_rda_bpn-c012_F8' 'brainvision_rda_bpn-c012_FC5' 'brainvision_rda_bpn-↵
c012_FC1' 'brainvision_rda_bpn-c012_FC2' 'brainvision_rda_bpn-c012_FC6'↵
'brainvision_rda_bpn-c012_C7' 'brainvision_rda_bpn-c012_C3' 'brainvision_rda_bpn-↵
c012_Cz' 'brainvision_rda_bpn-c012_C4' 'brainvision_rda_bpn-c012_C8'↵
'brainvision_rda_bpn-c012_TP9' 'brainvision_rda_bpn-c012_CP5' 'brainvision_rda_bpn-↵
c012_CP1' 'brainvision_rda_bpn-c012_CP2' 'brainvision_rda_bpn-c012_CP6'↵
'brainvision_rda_bpn-c012_TP10' 'brainvision_rda_bpn-c012_P7' 'brainvision_rda_bpn-↵
c012_P3' 'brainvision_rda_bpn-c012_Pz' 'brainvision_rda_bpn-c012_P4'↵
'brainvision_rda_bpn-c012_P8' 'brainvision_rda_bpn-c012_PO9' 'brainvision_rda_bpn-↵
c012_O1' 'brainvision_rda_bpn-c012_Oz' 'brainvision_rda_bpn-c012_O2'↵
'brainvision_rda_bpn-c012_PO10' 'brainvision_rda_bpn-c012_AF7' 'brainvision_rda_bpn-↵
c012_AF3' 'brainvision_rda_bpn-c012_AF4' 'brainvision_rda_bpn-c012_AF8'↵
'brainvision_rda_bpn-c012_F5' 'brainvision_rda_bpn-c012_F1' 'brainvision_rda_bpn-↵
c012_F2' 'brainvision_rda_bpn-c012_F6' 'brainvision_rda_bpn-c012_FT9'↵
'brainvision_rda_bpn-c012_FT7' 'brainvision_rda_bpn-c012_VC3' 'brainvision_rda_bpn-↵
c012_FC4' 'brainvision_rda_bpn-c012_FT8' 'brainvision_rda_bpn-c012_FT10'↵
'brainvision_rda_bpn-c012_C5' 'brainvision_rda_bpn-c012_C1' 'brainvision_rda_bpn-↵
c012_C2' 'brainvision_rda_bpn-c012_C6' 'brainvision_rda_bpn-c012_TP7'↵
'brainvision_rda_bpn-c012_CP3' 'brainvision_rda_bpn-c012_CPz' 'brainvision_rda_bpn-↵
c012_CP4' 'brainvision_rda_bpn-c012_TP8' 'brainvision_rda_bpn-c012_P5'↵
'brainvision_rda_bpn-c012_P1' 'brainvision_rda_bpn-c012_P2' 'brainvision_rda_bpn-↵
c012_P6' 'brainvision_rda_bpn-c012_PO7' 'brainvision_rda_bpn-c012_PO3'↵
'brainvision_rda_bpn-c012_POz' 'brainvision_rda_bpn-c012_PO4' 'brainvision_rda_bpn-↵

```

```
c012_PO8'}});
```

```
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1, 'setname', [filenames
{file} '_mocap'], 'savenew', [output_filepath filenames{file} '_mocap.
set'], 'gui', 'off');
```

```
Subject #1
```

```
pop_loadset(): loading file P:\Project_Sezen-
```

```
EMS_VR\data\2_raw_EEGLAB\1\block_Vibro_MoBI.set ...
```

```
Reading float file 'P:\Project_Sezen-EMS_VR\data\2_raw_EEGLAB\1\block_Vibro_MoBI.
fdt'...
```

```
Creating a new ALLEEG dataset 1
```

```
Removing 25 channel(s)...
```

```
Saving dataset...
```

```
Creating a new ALLEEG dataset 2
```

```
Removing 64 channel(s)...
```

```
Saving dataset...
```

```
Creating a new ALLEEG dataset 3
```

```
>> help pop_select
```

```
pop_select() - given an input EEG dataset structure, output a new EEG data structure
retaining and/or excluding specified time/latency, data point,
channel,
and/or epoch range(s).
```

```
Usage:
```

```
>> OUTEEG = pop_select(INEEG, 'key1', value1, 'key2', value2 ...);
```

```
Graphic interface:
```

```
"Time range" - [edit box] RETAIN only the indicated epoch latency or continuous
data
```

```
time range: [low high] in ms, inclusive. For continuous data,
```

```
several
```

```
time ranges may be specified, separated by semicolons.
```

```
Example: "5 10; 12 EEG.xmax" will retain the indicated
stretches of continuous data, and remove data portions outside
the indicated ranges, e.g. from 0 s to 5 s and from 10 s to 12 s.
Command line equivalent: 'time' (or 'notime' - see below)
```

```
"Time range" - [checkbox] EXCLUDE the indicated latency range(s) from the data.
For epoched data, it is not possible to remove a range of latencies
from the middle of the epoch, so either the low and/or the high
```

```
values
```

```
in the specified latency range (see above) must be at an epoch
```

```
boundary
```

```
(EEG.xmin, EEGxmax). Command line equivalent: [if checked]
```

```
'notime'
```

```
"Point range" - [edit box] RETAIN the indicated data point range(s).
```

```
Same options as for the "Time range" features (above).
```

```
Command line equivalent: 'point' (or 'nopoint' - see below).
```

```
"Point range" - [checkbox] EXCLUDE the indicated point range(s).
```

```
Command line equivalent: [if checked] 'nopoint'
```

```
"Epoch range" - [edit box] RETAIN the indicated data epoch indices in the dataset.
```

```
This checkbox is only visible for epoched datasets.
```

```
Command line equivalent: 'trial' (or 'notrial' - see below)
```

```
"Epoch range" - [checkbox] EXCLUDE the specified data epochs.
```

```
Command line equivalent: [if checked] 'notrial'
```

```
"Channel range" - [edit box] RETAIN the indicated vector of data channels
```

```
Command line equivalent: 'channel' (or 'nochannel' - see below)
```


"Channel range" - [checkbox] EXCLUDE the indicated channels.

Command line equivalent: [if checked] 'nochannel'

"..." - [button] select channels by name.

"Scroll dataset" - [button] call the eegplot() function to scroll the channel activities in a new window for visual inspection.

Commandline equivalent: eegplot() - see its help for details.

Inputs:

INEEG - input EEG dataset structure

Optional inputs

'time' - [min max] in seconds. Epoch latency or continuous data time range to retain in the new dataset, (Note: not ms, as in the GUI text entry specified, above). For continuous data (only), several time ranges can be specified, separated by semicolons. Example: "5 10; 12 EEG.xmax" will retain the indicated times ranges, removing data outside the indicated ranges e.g. here from 0 to 5 s and from 10 s to 12 s. (See also, 'notime')

'notime' - [min max] in seconds. Epoch latency or continuous dataset time range to exclude from the new dataset. For continuous data, may be [min1 max1; min2 max2; ...] to exclude several time ranges. For epoched data, the latency range must include an epoch boundary, as latency ranges in the middle of epochs cannot be removed from epoched data.

'point' - [min max] epoch or continuous data point range to retain in the new dataset. For continuous datasets, this may be [min1 max1; min2 max2; ...] to retain several point ranges. (Notes: If both 'point'/'noint' and 'time' | 'notime' are specified, the 'point' limit values take precedence. The 'point' argument was originally a point vector, now deprecated).

'noint' - [min max] epoch or continuous data point range to exclude in the new dataset. For epoched data, the point range must include either the first (0) or the last point (EEG.pnts), as a central point range cannot be removed.

'trial' - array of trial indices to retain in the new dataset

'notrial' - array of trial indices to exclude from the new dataset

'sorttrial' - ['on'/'off'] sort trial indices before extracting them (default: 'on').

'channel' - vector of channel indices to retain in the new dataset. Can also be a cell array of channel names.

'nochannel' - vector of channel indices to exclude from the new dataset. Can also be a cell array of channel names.

Outputs:

OUTEEG - new EEG dataset structure

Note: This function performs a conjunction (AND) of all its optional inputs. Using negative counterparts of all options, any logical combination is possible.

Author: Arnaud Delorme, CNL/Salk Institute, 2001; SCCN/INC/UCSD, 2002-

see also: eeglab()

>> help bemobil_interp

bemobil_interp() - Interpolates missing channels with spherical interpolation and average references the data.

Usage:

>> [ALLEEG, EEG, CURRENTSET] = bemobil_interp_avref(EEG , ALLEEG, CURRENTSET, channels_to_interpolate)

>> [ALLEEG, EEG, CURRENTSET] = bemobil_interp_avref(EEG , ALLEEG, CURRENTSET, channels_to_interpolate, out_filename, out_filepath)

Inputs:

ALLEEG - complete EEGLAB data set structure
 EEG - current EEGLAB EEG structure
 CURRENTSET - index of current EEGLAB EEG structure within ALLEEG
 channels_to_interpolate - vector of channel numbers that should be interpolated; if [], attempts to interpolate all missing (already deleted) channels from urchanlocs
 out_filename - output filename (OPTIONAL ARGUMENT)
 out_filepath - output filepath (OPTIONAL ARGUMENT - File will only be saved on disk if both a name and a path are provided)

Outputs:

ALLEEG - complete EEGLAB data set structure
 EEG - current EEGLAB EEG structure
 Currentset - index of current EEGLAB EEG structure within ALLEEG

.set data file of current EEGLAB EEG structure stored on disk (OPTIONALLY)

See also:

eeglab, bemobil_interp_reref, bemobil_interp_avref_copy_spatial_filter, bemobil_copy_spatial_filter, pop_interp, pop_reref, pop_interp,

Authors: Lukas Gehrke, Friederike Hohlefeld, Marius Klug, 2017

>> interpolated_filename

interpolated_filename =

'interpolated.set'

>>