

Video-LFP Synchronization Tool

This MATLAB script creates synchronized videos that display clinical video recordings alongside real-time LFP (Local Field Potential) neural data visualizations. The tool is designed for ERP (Exposure and Response Prevention) analysis in DBS-OCD patients, allowing researchers to observe behavioral and neural correlates simultaneously.

Table of Contents

1. Overview and Purpose
2. Requirements
3. Input Files
4. Script Workflow
5. Output Video Layout
6. Configuration Parameters
7. Usage Instructions

1. Overview and Purpose

The script generates a composite video that synchronizes clinical video recordings with neural signal visualizations, enabling frame-by-frame analysis of behavioral-neural correlations during ERP sessions.

Key Features

- Automatic neural-video alignment using beep marker (Event Code 10)
- Real-time display of raw LFP signals from both hemispheres
- Theta band (4-8 Hz) filtered signal visualization
- Live spectrograms showing frequency content (0-30 Hz)
- Clock time and elapsed time overlays
- Audio track preservation from original video

2. Requirements

Software

- **MATLAB** (R2020a or later recommended)
- **Signal Processing Toolbox** (for bandpass filtering)
- **FFmpeg** (for audio merging)

FFmpeg Installation

The script requires FFmpeg for adding the audio track to the final video. The default path is configured for macOS Homebrew installation:

```
ffmpeg_path = '/opt/homebrew/bin/ffmpeg';
```

Installation options:

- **macOS:** brew install ffmpeg
- **Windows:** Download from ffmpeg.org and update path
- **Linux:** sudo apt install ffmpeg

3. Input Files

3.1 Neural Data (.mat file)

The script expects a preprocessed .mat file containing the following structure:

| Field | Description |
|-------------------------------------|----------------------------------------------------------|
| data.neural.combined_data_table | Table with LFP channels, Timestamp, and Datetime columns |
| data.neural.fs | Sampling frequency (typically 250 Hz) |
| data.behavior.events | Event table with Timestamp and Event_Code columns |
| data.behavior.first_event_unix_time | Unix timestamp for task start reference |

Supported LFP Channel Names

The script automatically detects one of the following channel naming conventions:

- TD_Aic_ONE_THREE_LEFT / TD_Aic_ONE_THREE_RIGHT
- TD_Aic_ZERO_TWO_LEFT / TD_Aic_ZERO_TWO_RIGHT
- TD_Other_ZERO_TWO_LEFT / TD_Other_ZERO_TWO_RIGHT
- TD_Other_ONE_THREE_LEFT / TD_Other_ONE_THREE_RIGHT

3.2 Video File (.mp4)

Clinical video recording in MP4 format. The video must contain an audio track with a synchronization beep that corresponds to Event Code 10 in the neural data.

4. Script Workflow

Processing Steps

1. **Load Neural Data:** User selects .mat file via file dialog
2. **Extract Subject ID:** Parses filename to create output folder name
3. **Synchronization:** Finds Event Code 10 (beep) timestamp and aligns neural data
4. **Channel Detection:** Automatically identifies LFP channel naming convention
5. **Preprocessing:** Demeans LFP signals and replaces NaN values with zeros
6. **Theta Filtering:** Applies 4-8 Hz bandpass filter to extract theta oscillations
7. **Spectrogram Computation:** Calculates time-frequency representation using short-time FFT
8. **Video Selection:** User selects corresponding MP4 video file
9. **Frame Animation:** Iterates through video frames, updating all plots in sync
10. **Audio Merge:** Uses FFmpeg to combine silent output with original audio

Signal Processing Details

Preprocessing

```
% Demean: remove DC offset  
LFP_demeaned = LFP - mean(LFP, 'omitnan');  
% Replace NaN with 0  
LFP_demeaned(isnan(LFP_demeaned)) = 0;
```

Theta Band Extraction

```
theta = bandpass(LFP_demeaned, [4 8], fs);
```

Spectrogram Parameters

| Parameter | Value |
|-----------------|----------------------------------------------|
| Window | 0.5 seconds × sampling rate (Hamming window) |
| Overlap | 75% of window size |
| NFFT | max(256, next power of 2 after window) |
| Frequency range | 0-30 Hz (displayed) |
| Color scale | 0-30 dB (fixed) |

5. Output Video Layout

The output video has a resolution of 1920×1080 pixels with the following panel arrangement:

Panel Layout (5×2 Grid)

| Row | Left Column | Right Column |
|-----|--------------------------------------------|-----------------------------|
| 1-2 | Clinical Video (spans both columns) | |
| 3 | Raw LFP - Left Hemisphere | Raw LFP - Right Hemisphere |
| 4 | Theta Wave - Left (4-8 Hz) | Theta Wave - Right (4-8 Hz) |
| 5 | Spectrogram - Left | Spectrogram - Right |

Time Display Overlays

- Clock Time:** Shows actual datetime (hh:MM:SS PM format)
- Elapsed Time:** Shows seconds since video start

Plot Characteristics

- Time Window:** ±5 seconds centered on current frame
- Amplitude Range:** ±30 µV for LFP and theta plots
- Current Time Marker:** Black vertical line at t=0
- Raw LFP Color:** Red
- Theta Band Color:** Blue

6. Configuration Parameters

| Parameter | Default | Description |
|---------------------|--------------------------|---------------------------------------|
| ffmpeg_path | /opt/homebrew/bin/ffmpeg | Path to FFmpeg executable |
| time_window | 5 seconds | Display window (±seconds) |
| max_time | 10 seconds | Testing limit (remove for full video) |
| outputVideo.Quality | 95 | Output video quality (0-100) |
| Figure size | 1920 × 1080 | Output resolution (pixels) |

Note: For full video processing, comment out or remove the `max_time` condition in the animation loop.

7. Usage Instructions

Step-by-Step Guide

1. **Configure FFmpeg path:** Modify the ffmpeg_path variable if needed
2. **Run the script:** Execute video_and_LFPs_ERP.m in MATLAB
3. **Select neural data:** Choose the preprocessed .mat file when prompted
4. **Select video:** Choose the corresponding .mp4 video file
5. **Wait for processing:** Progress updates will appear in Command Window
6. **Find output:** Final video saved in subfolder named after subject

Output Files

| File | Description |
|-------------------------------|---------------------------------------------|
| {subject}_sync_output.mp4 | Intermediate file (no audio, deleted) |
| {subject}_sync_with_audio.mp4 | Final output with synchronized audio |

Troubleshooting

- **"Could not find LFP channels":** Check that .mat file contains expected variable names
- **FFmpeg error:** Verify FFmpeg is installed and path is correct
- **Video/neural mismatch:** Ensure Event Code 10 exists and corresponds to audio beep
- **Memory issues:** Process shorter segments using max_time parameter

8. Console Output Example

```
Using ONE_THREE channels
Time: 0.03s | Progress: 0.0% | Frame: 1
Time: 0.07s | Progress: 0.0% | Frame: 2
...
 Video generation complete!
⌚ Adding audio with FFmpeg...
 Final video created with audio: /path/to/output.mp4
⌚ Total elapsed time: 245.3 seconds (4.1 minutes)
```