Contents

[Goal - Surprise Onset Detection 2](#_Toc103021653)

[Overview 2](#_Toc103021654)

[Tasks 3](#_Toc103021655)

[Task 1 - Time Synchronization 3](#_Toc103021656)

[Requirements 3](#_Toc103021657)

[Process 3](#_Toc103021658)

[Deliverables 4](#_Toc103021659)

[Task 2 - EEG Labelling 4](#_Toc103021660)

[Requirements 4](#_Toc103021661)

[Deliverables 4](#_Toc103021662)

[Task 3 - Data Collection 4](#_Toc103021663)

[Requirements 4](#_Toc103021664)

[Deliverables 5](#_Toc103021665)

[Task 4 - Model Creation for Surprise Onset 5](#_Toc103021666)

[Requirements 5](#_Toc103021667)

[Deliverables 5](#_Toc103021668)

[Task 5 - Surprise Detection over Unseen Data 5](#_Toc103021669)

[Requirements 5](#_Toc103021670)

[Deliverables 5](#_Toc103021671)

# Goal - Surprise Onset Detection

In this project, our goal is to find EEG correlates of "surprise onset" incidences, while watching video clip(s).

## Overview

The diagrams below depict the process of modelling the "surprise onset" and detecting it over un-seen data.

Model Creation for “Surprise Onset” with Labelled data



EEG Data Extraction and Manual/Programmatic Frame Based Labelling

Video Frames and EEG Synchronization



CNN Model

Model Usage for “Surprise Onset” Detection over Unseen Data



Video Frames and EEG Synchronization

EEG Data Extraction



CNN Model

Predict the “frame number in the video” for surprise onset

## Tasks

This project is divided into following tasks:

1. Time Synchronization
2. EEG Labelling
3. Data Collection
4. Model Creation for Surprise Onset
5. Surprise Detection for Unseen Data

Each of these tasks is explained below in detail.

## Task 1 - Time Synchronization

### Requirements

As a part of this project, we need to design a method to synchronize the video and EEG. Here, synchronization means that ***with the display of each frame, we need to monitor the corresponding EEG potential across each channel***. How to achieve that? The EEG recording devices will give us a mapping of “EEG Potential of each channel” Vs “timestamp”. Whereas, the video recording gives us a mapping of “frame number” Vs “timestamp”. Since we have a **common timestamp** across the two mappings, we can easily find a mapping of “frame number” Vs “EEG Potential for each channel”. Below is the understanding of mapping in diagram format.

timestamp

EEG Potential of each channel

If we know **Vs**

timestamp

Frame Number

And we know **Vs**

Frame Number

EEG Potential of each channel

We can find **Vs**

### Process

1. Record the EEG Potential of each channel with corresponding timestamp, this will be considered a process-1. The data will be recorded in the following tabular format.

Table 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TimeStamp | TP9 | AF7 | AF8 | TP10 |

1. Record the appearance of frame against its timestamp, this will be considered as process-2. The data will be recorded in the following tabular format

Table 2

|  |  |
| --- | --- |
| TimeStamp | Frame Number |

1. Using the above two tables, map the EEG Potential of each channel against the frame number. This data will appear in the following format.

Table 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Frame Number | TP9 | AF7 | AF8 | TP10 |

### Deliverables

1. A python script for streaming the EEG Device data using Lab Streaming Layer (LSL) support
2. A python script to record the EEG data in the Table 1 format
3. A python script to stream the video and record the data in Table 2 format
4. A python script to denoise the EEG data in Table 1 with 50 Hz notch filter and make the data zero mean and unit variance
5. A python script to create data in Table 3 format using Table 1 and Table 2

## Task 2 - EEG Labelling

### Requirements

As a part of this project, we also need methods for EEG data labelling specific to "surprise onset". We already know that we get a mapping of “frame number” Vs “EEG Potential for each channel”, as an outcome of Time Synchronization section. Each of these mappings is labelled as ‘0’, if it is not a “surprise frame” and it is labelled as “1” if it is “surprise frame”. “surprise frame” is identified by **manual observation** of frame numbers. For example: in the commercial <https://youtu.be/X-OB_lUlF2g> the zombie appears and the user is surprised at the frame numer 353. In this case frame 353 is labelled as “1”. Rest all the frames are labelled as “0”. The outcome of data labelling module will look like in the following table (assume that frame 6, is a surprise frame).

Table 4

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Frame Number | TP9 | AF7 | AF8 | TP10 | Label |
| 1 | 19 | 23 | 12 | 14 | 0 |
| 2 | -25 | -15 | -21 | -18 | 0 |
| 3 | -34 | -11 | -28 | -22 | 0 |
| 4 | 13 | 33 | 19 | 5 | 0 |
| 5 | 36 | 43 | 22 | 22 | 0 |
| 6 | 10 | 10 | 35 | 31 | 1 |
| 7 | -11 | -22 | -2 | -12 | 0 |
| 8 | -5 | -9 | -6 | --4 | 0 |
| 9 | -1 | -26 | -16 | -7 | 0 |
| 10 | -23 | -12 | -18 | -34 | 0 |

**Note:** While collecting the surprise data across multiple subjects for the same video, we have to remember that the surprise frame (in this case, the frame showing appearance of zombie) will always be at the same frame number.

### Deliverables

1. A python script which will take the “surprise frame number(s)” as input and create Table 4 as an output (data with labels). This script should work across the data collected for multiple users for different videos.

## Task 3 - Data Collection

### Requirements

We need to collect the data in the above tabular format with multiple subjects (Nearly 15) and multiple videos (Nearly 3) with one or more surprise scenarios in each one of them.

### Deliverables

1. We need to create a Comma Separated Values (CSV) file with the combined data (in format of Table 4) across all subjects and across all videos with appropriately labelled data.

## Task 4 - Model Creation for Surprise Onset

### Requirements

Feed this "labelled" data to machine learning algorithms to build a **deep learning** model to detect "surprise onset". Apply the features of Data Augmentation and Data Balancing as per requirement to increase the accuracy.

### Deliverables

1. A python script which will create a deep learning model for detecting surprise onset. The input to this script will be the CSV file with the labelled data as mentioned in Task 3 - Data Collection. The outcome of this script will be a confusion matrix for validation data and a training and validation graph.

## Task 5 - Surprise Detection over Unseen Data

### Requirements

We will use the model created in “Task 4 - Model Creation for Surprise Onset” over unseen video clips, in order to detect surprise onset incidences. The exact frame(s) over which surprise is detected is highlighted in different color.

### Deliverables

1. A python script which will take the unseen video as an input and identify the surprise frames as an output.
   1. This script will be capable of doing all the three steps in “Task 1 - Time Synchronization” and will be able to generate a table like Table 3 (after denoising the data). This will be for a single subject and single video. Note that this data is NOT tabelled.
   2. This script will be capable of loading the model which is created in “Task 4 - Model Creation for Surprise Onset”.
   3. This script will be capable of predicting the label of each of the video frame as 1 or 0.
   4. This script will be capable of creating a video with python libraries, to mark each surprise-frame (with predicted label 1) by RED border and each normal frame (with predicted label 0) by BLUE border.