

motion-data-book

Mark Roman Miller

9/27/2022

Table of contents

Overview	3
1 MYREC File Format	4
1.1 File structure	4
1.2 File Contents	4
1.2.1 count.txt	4
1.2.2 master.txt	5
1.2.3 audioothers.mp3	5
1.2.4 events.txt	5
1.2.5 stream.txt	5
2 BIDS (Brain Imaging Data Structure)	7
3 Converting MYREC to BIDS	8
3.1 Extracting Myrec	8
3.2 Forming to BIDS	8
4 Examples of Social Analysis	9
4.1 libraries	9
4.2 total motion over time	9
4.3 average distance between people	9
4.4 Heatmap	9
4.5 Gaze distribution	9
References	10

Overview

[What is this book for]

Part of it is motion in general...

- Thinking about space, scientifically
 - geospatial vs. human-spatial
 - personal space findings
- Thinking about space, mathematically
- Tidy spatial data with `dddr`
 - why a separate library
- Key spatial operations

Part of it is a technical guide...

The process is (in order):

- Understanding the `.myrec` file format
- Understanding the BIDS format
- Converting from `.myrec` to BIDS
- Example analysis

1 MYREC File Format

The .myrec file format is a proprietary, undocumented, and unsupported VR recording format from ENGAGE. In it contains everything that the ENGAGE client needs to re-render a recording. Fortunately for those who study human behavior, this recording contains quite a view streams of data in high temporal fidelity. The recording includes position and rotation data for each participant, as well as IFX interactions, avatar choices, audio, and button events.

1.1 File structure

The .myrec file is a ZIP file archive that contains several compressed files within it. This file structure is consistent across all .myrec files that I've encountered from June to December 2021. Once unzipped recursively, the file structure is:

- `count.txt`
- `master` (ZIP)
 - `master.txt`
- `stream0` (ZIP)
 - `audioothers.mp3`
 - `events.txt`
 - `stream.txt`
- `stream1` (ZIP)
- ...

Each file is described below.

1.2 File Contents

1.2.1 `count.txt`

This file is a one-line file in what appears to be key-value pairs. Pairs are separated by a semicolon, and keys and values are separated by a pipe.

- **streamCount**: the number of **streamN** archives in the top-level archive, where N is a non-negative integer.
- **sceneID**: the environment in which the recording takes place.

1.2.2 master.txt

This file gives information about the recording as a whole. For example, there is some information about the timing of the different **streamN** files. Information about the recorder's and participants' avatars is also stored here. While its file suffix is **.txt**, it is in fact a JSON file. There are too many entries to be described here.

1.2.3 audioothers.mp3

This file is an mp3 file recording the spoken audio. Presumably, based on its name, it only captures audio spoken by people who were not recording, but this has not been verified directly. This file only exists in streams in which someone was speaking. In order to create an audio track for the entire file, one needs to create a space for silence for the duration of each stream without this file.

1.2.4 events.txt

This file mainly deals with changes that aren't necessarily tied to an avatar. It is also a JSON file despite its suffix. Much of the activity in this file is related to IFX motion and usage. Other than hints given by the names in the keys, it is not understood what these values refer to specifically.

1.2.5 stream.txt

This file is what appears to be a custom format storing the values of several variables for several users over a number of frames (140, in 2021). Breaking down each level from largest granularity to smallest:

Users are demarcated by one or multiple leading > characters. Each successive > increments the user's ID by one. For example, if users 1, 2, 5, and 6 are in the recording, then the > characters will be distributed as follows: > (1's data) > (2's data) >>> (5's data) > (6's data). Note that at the point in the file with user data, the user ID is equal to the number of >s that precede it in the entire file.

Variables are demarcated with a name and one | character at the beginning, and one ; character at the end. Variable names follow the format <name><type>x. Here, <type> can be **int** for integer, **flo** for float, or **v3** for a 3D vector type.

The spatial variables we used had a `<name>` following the convention `<tracked_point>(Positions|Rotations)`. The tracked points included `AvaRoot`, `Head`, `LeftHand`, `RightHand`, `LeftFoot`, `RightFoot`, and `Hip`. For all the data collected in the summer and fall 2021 studies, the values for the feet and the hips were unused and not meaningful. `AvaRoot` stands for Avatar Root, and it defined the coordinate transformation from the coordinate space for `Head`, `LeftHand`, and `RightHand` into the global coordinate system.

The only other variable we used was `LipSyncAverageflox`, which indicated the amount of an avatar’s lip-flapping and was presumably based upon volume. There were several other values that are included in the recording but we have not used, such as `IFXScaleflox`, `(Left|Right)TriggerPressedintx`, `(Left|Right)HandPointingintx`, `(Left|Right)HandLaserPointerintx`, `(Left|Right)HandWhiteboardingintx`, `TabletOutintx`, `AvatarEmotionStateintx`, `RaisingHandintx`, `Clappingintx`, `OutfitOverrideStateintx`, `UseSitTriggerOverridesintx`, `IsAwayintx`, `IsInSitTriggerintx`, `(Left|Right)HandWhiteboardEmitterPos`, `IfxPositionsv3x`, and `IfxRotationsv3x`.

Samples are demarcated with with infix `|`. Any values that are equal to the previous value are not included, so several variables often are written `|0|||||...`

3D Vectors are demarcated in a special way relative to other data types. The three elements of the vector are separated with infix `<` symbols, and changes propagate along dimension individually. For example, the value `|<2.345<|` means the y-value changed to 2.345, but the x- and z- values remained the same. Position and rotation vectors were written the same way; they only difference in the data itself is the variable name discussed above. Positions are interpreted as `X<Y<Z`, where axis conventions are used according to Unity (Y up, Z out, left-handed). Rotations are interpreted as `pitch<yaw<roll`, where angle conventions are also used according to Unity.

2 BIDS (Brain Imaging Data Structure)

link: <https://bids-specification.readthedocs.io/en/stable/>

also: https://docs.google.com/document/d/1iaaLKgWjK5pcISD1MVxHKexB3PZWfE2aAC5HF__pCZW0/edit

3 Converting MYREC to BIDS

I've written some RMarkdown on this, this is helpful

3.1 Extracting Myrec

(see the Rmd file I sent over a long while ago)

3.2 Forming to BIDS

4 Examples of Social Analysis

4.1 libraries

synsyn (<https://github.com/markromanmiller/synsyn>) dddr

4.2 total motion over time

4.3 average distance between people

4.4 Heatmap

4.5 Gaze distribution

References