Pose Estimation in NWB

ndx-pose

NDX

Neurodata
Extensions
Catalog

NEURODATA
WITHOUT BORDERS

Pose Estimation in NWB

ndx-pose

NWB Currently supports writing from two popular pose estimation frameworks

- SLEAP
- DeepLabCut (DLC)

We do not have time to cover exactly *how* to use or understand these packages

- See the main keynote talk by Mackenzie Mathis instead

So we will simply explain the input-output (I/O) workflow

Note: Storing natural behavior in NWB

- The NWB data type for videos called the *ImageSeries*

- It has two modes
  - Internal
    - each frame of the video is extracted and written as a Dataset
    - same as any other array-valued series you might find in an NWB file
  - External
    - data type instead contains a *path* to the video file(s) outside the NWB file

- Reasoning for this is a bit nuanced…

- For history of decision, see…
  - NWB:#1647, DANDI:#769 and DANDI-helpdesk:#30

Note: Storing natural behavior in NWB

- Best Practices
  - Internal mode
    - videos of physiological recordings (such as optical imaging)
    - stimulus presentations
  - External mode
    - videos of natural behavior

- DANDI
  - strongly recommends these practices for storage
  - will automatically remap all paths and organize folder storage at time of upload

Step 1: Install SLEAP or DLC

- SLEAP - Talmo Lab - instructions - GitHub repo
- DLC - Mathis Lab - instructions - GitHub repo

- This step can take a while, but just be patient and persistent

- If anything goes differently from instructions, when in doubt ask
  - Google
  - ChatGPT
  - or raise an issue on their repos/helpdesks

Step 2: Launch GUI

- These can be run headless through their API but I recommend using the **G**raphical **U**ser **I**nterface (GUI) to help walk you through it

- SLEAP can import NWB ImageSeries and export NWB PoseEstimation

- Both are able to load in original videos (`.mpg`, `.mp4`, `.avi`, etc…)

- Due to the previous discussion of storing ImageSeries of natural behavior it is recommended to always load in the raw video to both platforms

Steps 3-7

- Step 3: Load videos
- Step 4: Extract frames for training
- Step 5: Apply points and labels on training frames
- Step 6: Train the model
- Step 7: Run the prediction

Saving results

- The output of the prediction over the entire video can then be saved in either the source `.slp` or `.h5` (DLC) formats

- These can in turn be converted to NWB format via two utility packages
  - Dlc2nwb (`.h5` → `.nwb`)
  - sleap-io.nwb sub-module (`.slp` → `.nwb`)

- BONUS: SLEAP can export directly to NWB through the GUI

Understanding PoseEstimation

- In either case (or for any future integrations with keypoint tracking tools) these are modelled using a common structure in NWB called the `PoseEstimation`

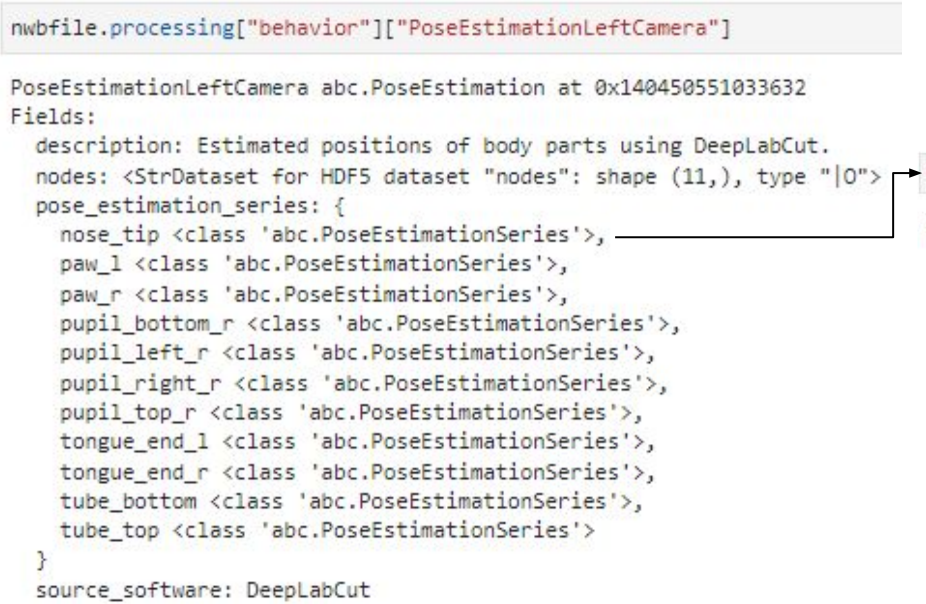
- A single `PoseEstimation` is a container of multiple `PoseEstimationSeries`, one for each node (usually a body part) being tracked

- The `data` field of each `PoseEstimationSeries` represents the `x`/`y`/`z` positions being tracked over time

- There is also the `confidence` (‘likelihood’) associated with each position over time

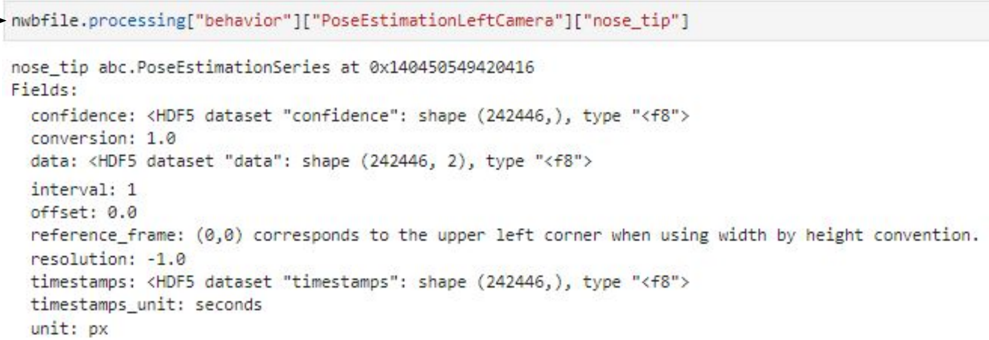
Understanding PoseEstimation

- See it used in practice in the IBL Brain Wide Map (dandiset #409)



```
nwbfile.processing["behavior"]["PoseEstimationLeftCamera"]
```

```
PoseEstimationLeftCamera abc.PoseEstimation at 0x140450551033632
Fields:
  description: Estimated positions of body parts using DeepLabCut.
  nodes: <StrDataset for HDF5 dataset "nodes": shape (11,), type "|O">
  pose_estimation_series: {
    nose_tip <class 'abc.PoseEstimationSeries'>,
    paw_l <class 'abc.PoseEstimationSeries'>,
    paw_r <class 'abc.PoseEstimationSeries'>,
    pupil_bottom_r <class 'abc.PoseEstimationSeries'>,
    pupil_left_r <class 'abc.PoseEstimationSeries'>,
    pupil_right_r <class 'abc.PoseEstimationSeries'>,
    pupil_top_r <class 'abc.PoseEstimationSeries'>,
    tongue_end_l <class 'abc.PoseEstimationSeries'>,
    tongue_end_r <class 'abc.PoseEstimationSeries'>,
    tube_bottom <class 'abc.PoseEstimationSeries'>,
    tube_top <class 'abc.PoseEstimationSeries'>
  }
  source_software: DeepLabCut
```



```
nwbfile.processing["behavior"]["PoseEstimationLeftCamera"]["nose_tip"]
```

```
nose_tip abc.PoseEstimationSeries at 0x140450549420416
Fields:
  confidence: <HDF5 dataset "confidence": shape (242446,), type "<f8">
  conversion: 1.0
  data: <HDF5 dataset "data": shape (242446, 2), type "<f8">
  interval: 1
  offset: 0.0
  reference_frame: (0,0) corresponds to the upper left corner when using width by height convention.
  resolution: -1.0
  timestamps: <HDF5 dataset "timestamps": shape (242446,), type "<f8">
  timestamps_unit: seconds
  unit: px
```