20260525_hello_zolin2021context_DAN_timeseries_overlaid_with_locomotion

Compiled by: Rich Pang
Date: 2026-05-25

Dataset

From Zolin et al, Nat Neuro 2021.
Context-dependent representations of movement in Drosophila dopaminergic reinforcement pathways
https://www.nature.com/articles/s41593-021-00929-y

Calcium signals from DANs (dopamine neurons) impinging on the fruit fly mushroom body compartments γ2, γ3, γ4, γ5, via synaptically localized calcium indicator sytGCaMP6s, while flies walk on a trackball under various wind and olfactory conditions.

The complete publicly available dataset is about 3.5 Gb. The recording files themselves total about 1 Gb. The other 2.5 Gb in the repo are various post-processed files used in the paper and not necessary for this demo.

Each trial is a 5-minute recording of DAN activity in (typically) both hemispheres (8 compartments total, γ2-5 on left and right), simultaneous with locomotion on the ball, as well as other signals such as odor.

Each recording has been processed into a file of about 1-2 Mb, corresponding to a 5 minute recording sampled at 10 Hz (the imaging rate for the GCamp indicator).

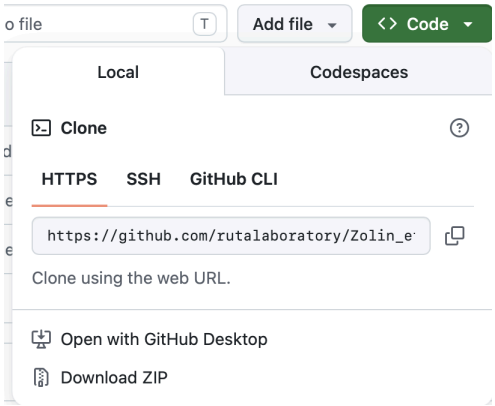
The total dataset contains several dozen flies spread across different conditions, with up to several trials per fly.

The main data files are stored as CSV files with column names like G3_avg, G4_avg, Motion, ForVel, etc. (“avg” means averaged over both hemispheres). Some files have individual left and right signals as well, of the format G2_L, G2_R, G3_L, etc. Columns beginning with G2_ are GCamp signals scaled relative to a red channel indicator that does not indicate neural activity but is useful for e.g. correcting motion artifacts.

Accessing data

Navigate to https://github.com/rutalaboratory/Zolin_etal_2021 .

Click on the green “Code” button and select Download Zip:



Save and unzip the file, which will produce a directory called Zolin_etal_2021.
The data we will use is in the subdirectory “data_”.
Move the directory “data_” into your main project directory.
The project structure should then look like

```
my_project
  - data_
    - RESULTS_
    - ASENSORY_AZ
      - 20151230.Fly1
        - 20151230.Fly1.1
        - 20151230.Fly1.2
        - 20151230.Fly1.3
      - 20160118.Fly1
        - 20160118.Fly1.2
        - 20160118.Fly1.3
    - CL_180_HighFlow_ACV
      - 20170111.Fly2
        - 20170111.Fly2.2
        - 20170111.Fly2.3
      ...
    ...
```

Software used below

Python version 3.12.12, with the following packages

- Numpy version 2.3.5
- Matplotlib version 3.10.6
- Pandas version 2.3.3

Code

Make a new file, `view_dans_and_locomotion.py`, within your `my_project` directory.

Add the following code and run in your terminal via: `python view_dans_and_locomotion.py`

```
import matplotlib.pyplot as plt
import numpy as np
```

```
import os
import pandas as pd

# loop over three experiment types
EXPTS = [
    'ASENSORY_AZ',  # asensory experiments
    'CL_360_LOWFLOW_ACV',  # experiments with odor + low-flow wind
    'CL_180_HighFlow_ACV',  # experiments with odor + high-flow wind
]

COLORS = {
    'G2': 'b', 'G3': 'r', 'G4': 'g', 'G5': 'm'  # G = "gamma"
}

DT = 0.1  # 1/sampling frequency (s) (10 Hz sampling)

N_TRL_PLOT = 2  # how many trials to plot per experiment

np.random.seed(0)  # choose same random trials to plot every time


# loop over experiments
for EXPT in EXPTS:

    print('Loading data from Experiment', EXPT, '...')

    DATA_DIR = os.path.join('data_', EXPT)

    if EXPT == 'ASENSORY_AZ':
        base = 'BigMAT.csv'
        mvng = 'Moving_Bouts.csv'
        mvng_cols = ['T1', 'T2']
        odor_file = None
    else:
        base = 'clean.csv'
        mvng = 'moving.csv'
        mvng_cols = ['Start', 'Stop']
        odor_file = 'odor_times.csv'

    trls = []
    dfs = {}  # dataframes
    odor_times = {}

    # load all flies/trials for this experiment
    nfly = 0
    for fly in os.listdir(DATA_DIR):
        if fly.startswith('.'):  # ignore .DS_Store etc
            continue
        fly_path = os.path.join(DATA_DIR, fly)
        nfly += 1
```

```
        for trl in os.listdir(fly_path):
            if trl.startswith('.'):  # ignore .DS_Store etc
                continue
            trl_path = os.path.join(fly_path, trl)

            # load data
            df = pd.read_csv(os.path.join(trl_path, base))
            dfs[trl] = df

            trls.append(trl)

            # load odor times
            if odor_file is not None:
                df_odor = pd.read_csv(os.path.join(trl_path, odor_file))
                odor_times_ = []
                for t_on, t_off in zip(df_odor['Odor_On'],
df_odor['Odor_Off']):
                    odor_times_.append((t_on, t_off))
                odor_times[trl] = odor_times_
            else:
                odor_times[trl] = []

    print(nfly, 'flies,', len(trls), 'trials')

    # make example plots

    # choose two random trials to show
    itrls = np.random.permutation(len(trls))[:2]

    # plot gamma3-motion overlay
    fig_g3_motion, axs_g3_motion = plt.subplots(
        1, 2, figsize=(15, 2.5), tight_layout=True, sharex=True)

    # loop over trials
    for itrl, ax in zip(itrls, axs_g3_motion):
        trl = trls[itrl]  # get trial name
        df = dfs[trl]  # get dataframe for this trial
        odor_times_ = odor_times[trl]  # get odor times for this trial

        # time vector
        t = df['Time']

        ax.plot(t, df['Motion'], c='k', lw=.5)

        ax_twin = ax.twinx()  # for overlay using different axis
        ax_twin.plot(t, df['G3_avg'], c=COLORS['G3'], lw=1)

        # odor times
        for t_on, t_off in odor_times_:
```

```
            ax.axvspan(t_on, t_off, color='r', alpha=.3)

        # limits & labels
        ax_twin.set_ylim(df['G3_avg'].min(), 3)
        ax.set_ylim(0, .2)

        ax.set_xlabel('Time (s)')
        ax.set_ylabel('Motion (a.u.)')
        ax.set_title(f'{EXPT}: {trl}')

        ax_twin.set_ylabel('G3 (a.u.)', color='red')


    # plot traces for different variables in individual panels
    fig, axs = plt.subplots(
        8, 2, figsize=(15, 10), tight_layout=True, sharex=True)

    # loop over trials
    for itrl, ax_col in zip(itrls, axs.T):
        trl = trls[itrl]  # trial name
        df = dfs[trl]  # dataframe for this trial
        odor_times_ = odor_times[trl]  # odor times for this trial

        # time vector
        t = df['Time']

        # plot DAN signals
        for gx, ax in zip(['G2', 'G3', 'G4', 'G5'], ax_col[:4]):
            ax.plot(t, df[f'{gx}_avg'], c=COLORS[gx], ls='-', lw=1)
            ax.set_ylabel(f'{gx} level\n(a.u.)', color=COLORS[gx])

            # uncomment the next 4 lines if you want to plot L and R DAN
signals
            # if f'{gx}_L' in df.columns:
            #     ax.plot(t, df[f'{gx}_L'], c=COLORS[gx], ls='-.', lw=1)
            # if f'{gx}_R' in df.columns:
            #     ax.plot(t, df[f'{gx}_R'], c=COLORS[gx], ls='--', lw=1)


        # plot movement signals
        ax_col[4].plot(t, df['Motion'], c='k')
        ax_col[4].set_ylabel('Motion')

        ax_col[5].plot(t, df['ForVel'], c='k')  # forward velocity
        ax_col[5].plot(t, df['LatVel'], c='gray')  # lateral velocity
        ax_col[5].set_ylabel('Fwd Vel (black),\nLat Vel (gray)\n(a.u.)')

        ax_col[6].plot(t, df['AngVel'], c='k')  # angular (turning) velocity
        ax_col[6].set_ylabel('AngVel (a.u.)')
```

```
        # heading (angle between fly axis and upwind)
        if 'AbsHeading' in df.columns:
            ax_col[7].plot(t, df['AbsHeading'], c='k')
        ax_col[7].set_ylabel('|Heading|\n(radians)')

        # show odor times
        for ax in ax_col:
            for t_on, t_off in odor_times_:
                ax.axvspan(t_on, t_off, color='r', alpha=.3)

        ax_col[-1].set_xlabel('Time (s)')

        # set title
        ax_col[0].set_title(f'{EXPT}: {trl}')

    plt.show()
```

Outputs

The above code will print:

```
Loading data from Experiment ASENSORY_AZ ...
75 flies, 178 trials
Loading data from Experiment CL_360_LOWFLOW_ACV ...
26 flies, 48 trials
Loading data from Experiment CL_180_HighFlow_ACV ...
22 flies, 52 trials
```

and produce the following plots. For each of the three experiment types, two random trials are shown.

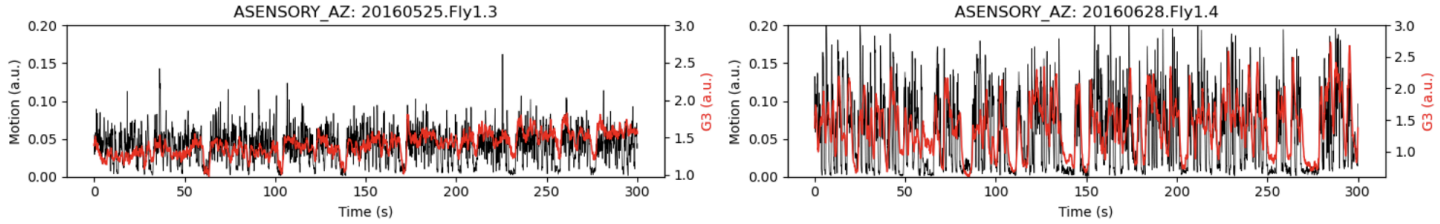
The first figure shows the total fly walking motion overlaid with the gamma3 DAN activity for both trials. Motion and gamma3 are highly correlated.

The second figure shows several individual traces of different DAN and motion variables. Transparent red sections indicate times when the odor (apple cider vinegar, ACV) was on.

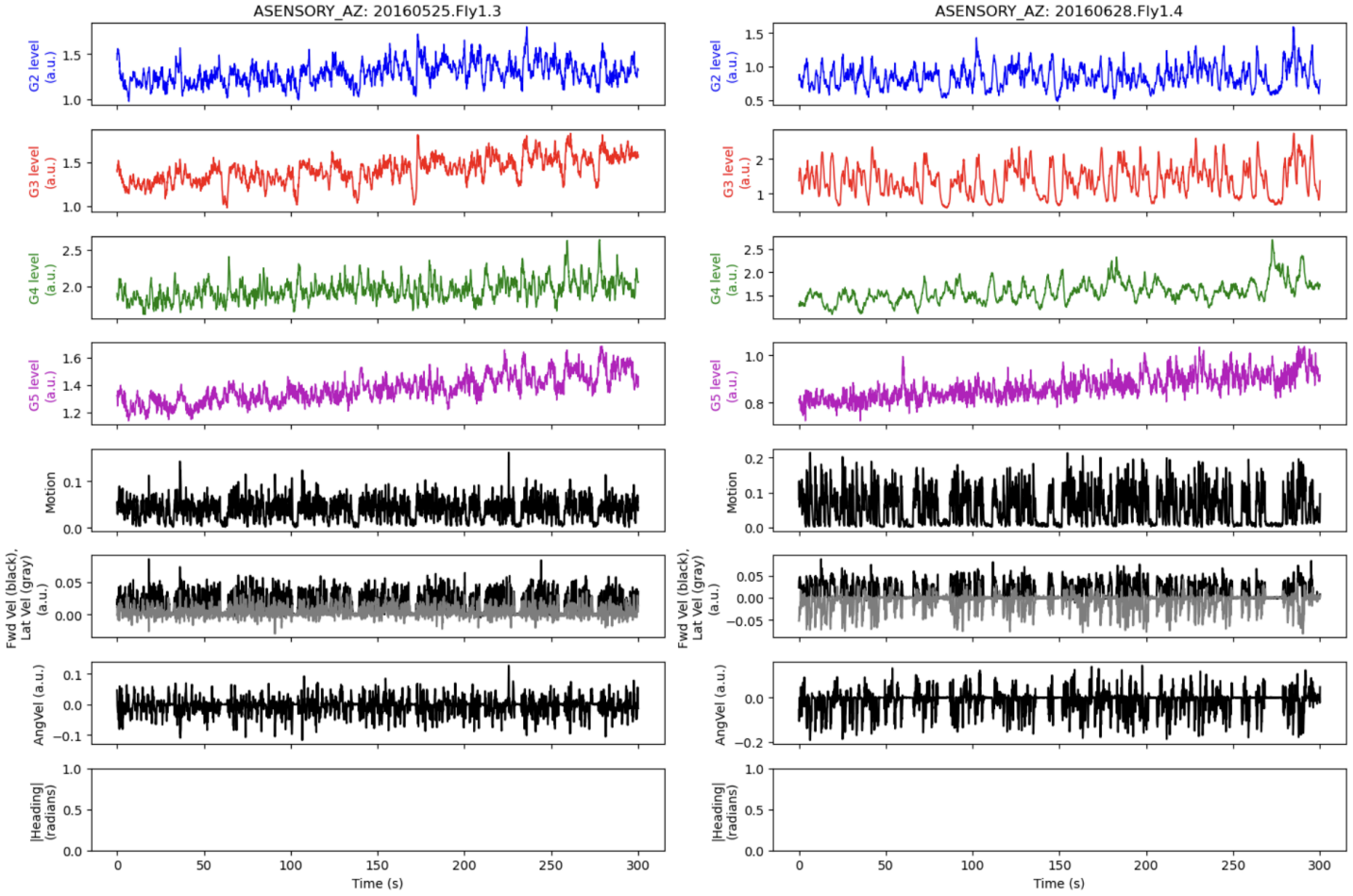
All of the DAN traces shown here are averages between the two hemispheres. E.g. G3 corresponds to the average of the left and right gamma3 signal.

ASENSORY experiments:

First fig

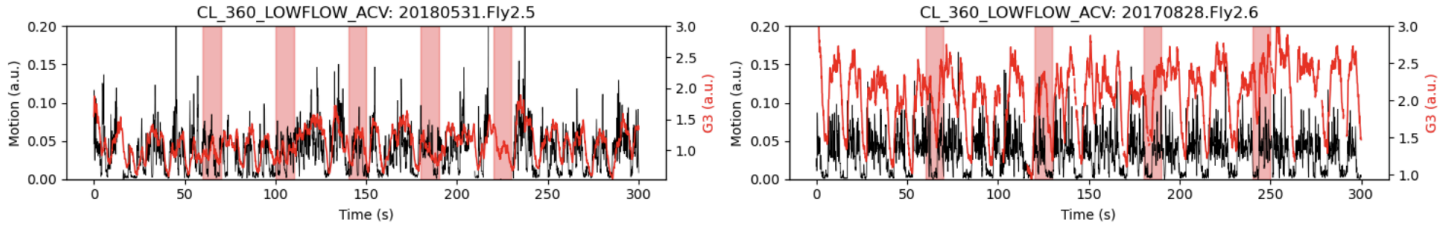


Second fig

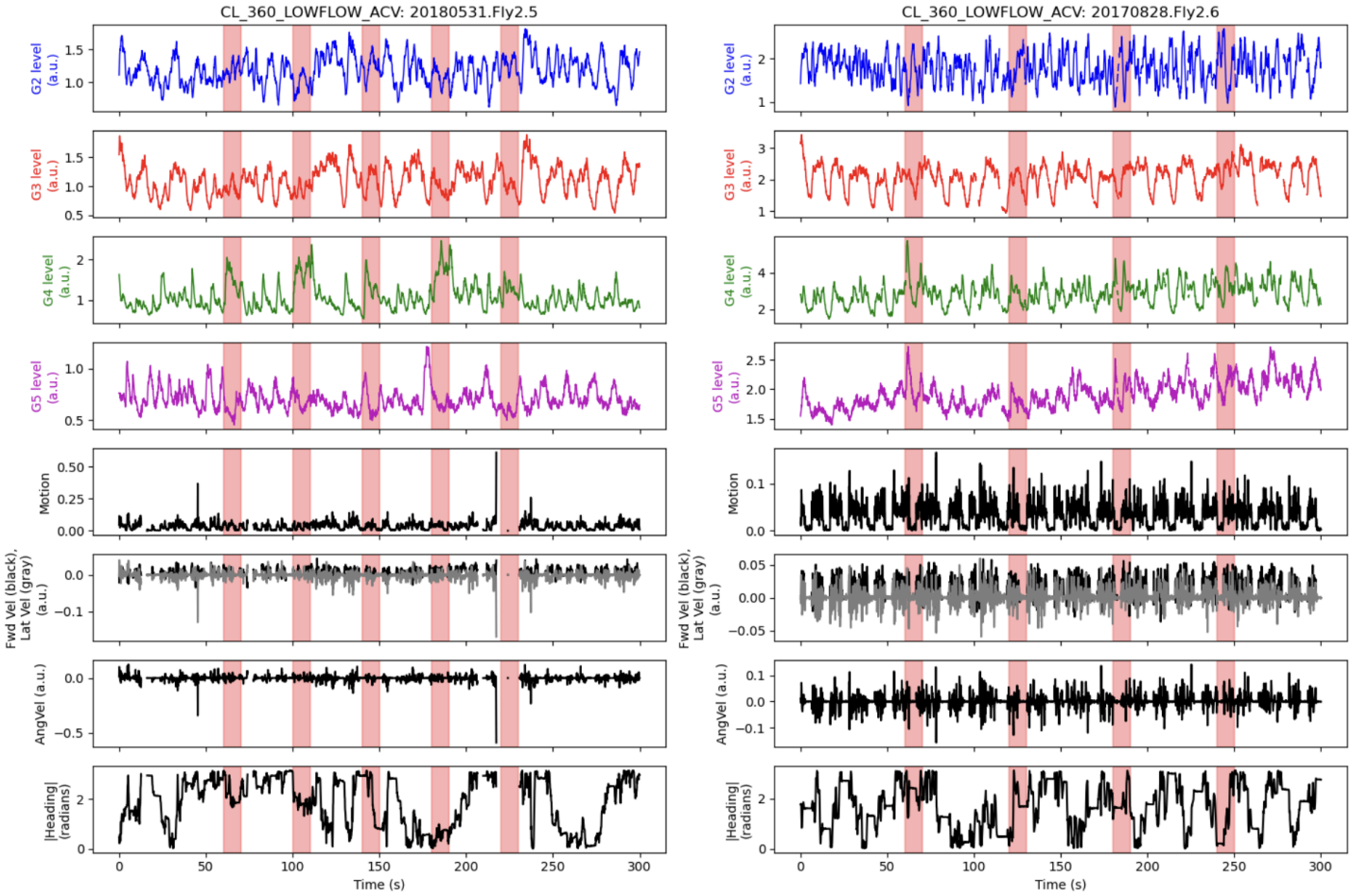


Closed-loop Low-flow experiments:

First fig

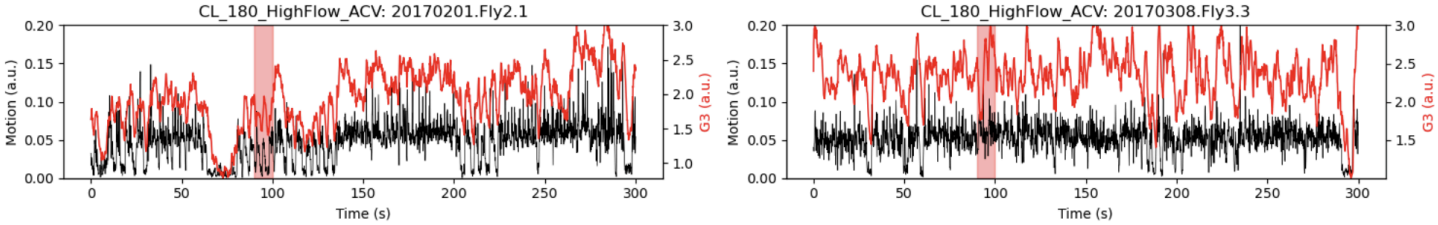


Second fig



Closed loop high-flow experiments

First fig



Second fig