**README.md**

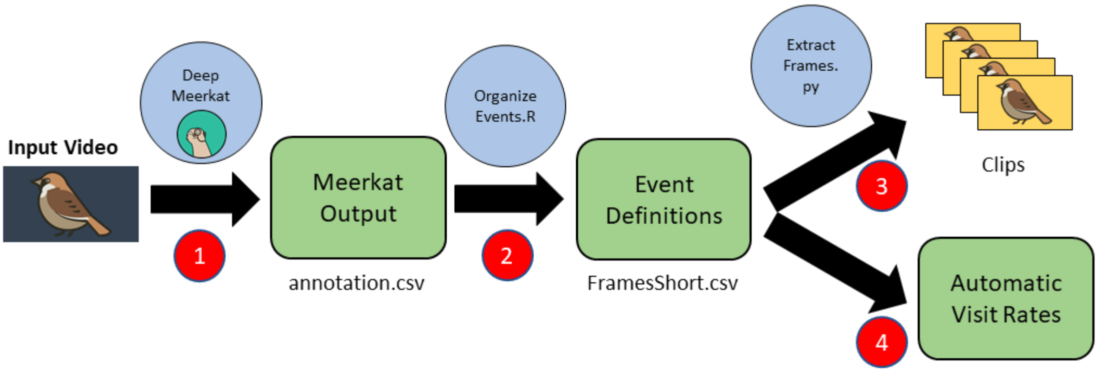
# Sparrow Vis: A pipeline to pre-process and analyse sparrow provisioning videos

Hi and welcome to SparrowVis!! This is the documentation and instructions to run and extract data from House Sparrow provisioning videos on Lundy. I will try to break it down and make it as easy as possible!

Here is the pipeline and the general workflow:

1. Open source software called Deep Meerkat (Weinstein, 2018) to identify movement frames
2. Process output and group frames into “events”
3. Extract short video clips of the events
4. Summarize processed videos into a biological meaningful “visit rate”

The pipeline uses a mix of R and python, and requires a little bit of running code on the command line/terminal, but don’t be intimidated!



## Installation Instructions

The pipeline is unfortunately written for unix systems (linux/mac). To use the pipline on Windows, you will need a few extra steps. If you are a mac/ linux user, just scroll down and get started!

### Windows Instructions

You will need to get the ubuntu shell. I followed this youtube video to set it up. It should be quick and easy! To test this, I used Ubuntu 20.04 LTS, so I recommend using that as well!

Once you have the ubuntu terminal set up, you can then run the following to access file explorer, which allows you to copy files to and from your windows system.

```
explorer.exe .
```

Finally, if you copy this current folder to your linux subsystem, you can follow the linux/mac instructions below and run the pipeline.

### Linux/ Mac Instructions

To install all software are packages required, you will first need to install Anaconda. Follow this link and install it on your system. (If you are a windows user and just set up the ubuntu terminal from above, you should choose the "Installing for Linux option!)

Once anaconda is installed, you have to navigate to the SparrowVis_Pipeline/ folder using `cd` , and run the following to install all packages required:

```
conda create --file SparrowVis.yml
```

This creates something called a "virtual environment", which has the packages and the correct version of those packages for the pipeline to run! The virtual environment that was created should be call "SparrowVis".

To activate the environment, you will need to run the following everytime you are running the pipeline.

```
conda activate SparrowVis
```

## Running the Pipeline

The following instructions should provide everything you need to run the pipeline. If you want to dive into what each script does, go look at the readme from the Code directory

### File Management

One important thing to note is that where and what you put under each folder **matters a lot** (mainly because I got lazy when writing the code). Generally, you want to put all the videos inside the **MeerkatInput**/ folder, and you will get the results in the **Output**/ folder.

### Running the Code

Once you have everything set up, you are ready to go! **Make sure you run everything below in the Code/ directory** (you can do that by using `cd` to change directory to /SparrowVis_Pipeline/Code/)

To run the pipeline for 1 video, use:

```
bash SamplePipeline.sh [video]
```

Where [video] is the path to the input video. For example, to run the pipeline for "VK0001", I would run

```
bash SamplePipeline.sh ../MeerkatInput/VK0001.mp4
```

If you have a bunch of videos, I also provided a script that runs every video in a loop. So you can run them overnight! To run that, just run:

```
bash SamplePipeline_Loop.sh
```

The script reads in all the files under **MeerkatInput**/ and runs the pipline through it. Afterwards, the video will be moved to **RawVideos**/ so you can keep track which videos were ran or not!

### Intepreting Output

If everything ran smoothly, you should get a bunch of short clips named EventX.mp4, with a csv called Short.csv under a folder with the video's name. Short.csv stores all the event names and timesteps, which is useful when doing manual annotations for the clips!

Once all videos are processed, you can run the following to get automatic visit rates:

```
Rscript OrganizeOutput.R
```

This will create a csv file called "AutoVisitRate.csv" which stores counts the visits detected for each video and the effective observation time (see Nakagawa et al., 2007)

## Common Errors

- **"No such file or directory"**: Either you are not running the code under the Code/ directory, or the videos you inputted are not under MeerkatInput/. Solve this by checking where all the files are and make sure you `cd` to the Code/ directory before running anything
- **"AttributeError: module 'tensorflow' has no attribute 'Session'"** or any **"module not found"** errors: You likely forgot to activate the virtual environment before running the code. Remember to run `conda activate SparrowVis` !
- Empty short.csv and no clips in output folders. This is likely not an error, try double check annotations.csv (output from deepmeerkat). Likely no motion was detected throughout the video!
- If you are struggling or have lots of error, feel free to contact me! My affiliations are below.

## References

- Nakagawa, S., Gillespie, D.O.S., Hatchwell, B.J., Burke, T., 2007. Predictable males and unpredictable females: sex difference in repeatability of parental care in a wild bird population. J. Evol. Biol. 20, 1674–1681.
- Weinstein, B.G., 2018. Scene-specific convolutional neural networks for video-based biodiversity detection. Methods Ecol. Evol. 9, 1435–1441.

## Author and Affiliations

Alex Chan Hoi Hang
the.alex.chan2@gmail.com/ hhc4317@ic.ac.uk
MRes Computational Methods in Ecology and Evolution
Department of Life Sciences
Imperial College Silwood Park
UK. SL5 7PY