

## Short description

Script to predict new data by BOVIDS. Given a .csv file containing the information which nights should be predicted, BOVIDS proceeds in the following steps:

1. Extract single frames from videos per enclosure
2. Detect the individuals on those images per enclosure (object detection)
3. Predict the behavior on those cut-out images per individual (action classification)
4. Move files from a temporary storage (local HDD) to a final storage (maybe a server).
5. Post-Process data and create a human-readable output per night.
6. Report the object detection density.

The necessary concepts to create the required csv file can be found in readme.md, paragraph “data prediction and evaluation”.

Consists of predict\_csv.py and configuration.py of folder “prediction”.

## Requirements

- packages: almost all packages of “bovids” environment
- folder “global/” and “global/yolo-v4-tf.keras-master/”
- prediction csv file (see readme.md)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Date	Species	Zoo	Enclosure	Video	Individual	Start	End	PP-total	PP-binary	TruncTop	TruncBot	TruncLef	TruncRig
2	09.01.2020	oryx	FantasyZoo	3	3;3;4	17	7	Standard	Standard-binary	0	2000	0	2000	
3	10.01.2020	wildebeest	NiceZoo	1	1;4;3;4	17	7	Standard	Standard-binary	0	2000	0	2000	

## Step 1 – open spyder

- Terminal / shell:
  - conda activate bovids
  - spyder

## Step 2 – adjust parameters

### configuration.py:

- INPUT\_CSV\_FILE: Path to csv file containing the necessary information (see above)  
[string]
- TMP\_STORAGE\_IMAGES = Temporary storage (folder) for the images extracted from the videos. The HDD needs enough space to store those images for all conducted nights at once. [string]

- `TMP_STORAGE_CUTOUT` = Temporary storage (folder) for those images (single frame / multiple frame) that are cut-out by the object detector per individual. [string]
- `FINAL_STORAGE_CUTOUT` = Final storage (folder) for content of `TMP_STORAGE_CUTOUT` [string]
- `FINAL_STORAGE_PREDICTION_FILES` = Final storage for .csv files containing the action classification per night and individual. Contains furthermore the post-processed data as well as a human readable overview per night. [string]
- `LOGGING_FILES` = The output of the console will be saved in `LOGGING_FILES` (folder). [string]

#### *Conducted Steps:*

Set a variable to **True**, if a step should be **skipped**. Set it to False if a step is conducted.

[boolean]

- `SKIP_IMAGE_CREATION` = In this step, video streams are merged, and black regions are drawn (as given through `global_configuration.py`). Furthermore, four images per time-interval will be saved in `TMP_STORAGE_IMAGES`. Creates a subfolder per enclosure, containing subfolders for each night. Those images might be used to check whether the black regions are decently drawn and which values of truncation might be useful.
- `SKIP_INDIVIDUAL_DETECTION` = The object detection phase on all images from `TMP_STORAGE_IMAGES`. Creates a folder per individual in `TMP_STORAGE_CUTOUT` containing all nights. Each night contains multiple frame, single frame images as well as position files (bounding box coordinates).
- `SKIP_BEHAVIOR` = For each night the action classifier is used to predict the images of `TMP_STORAGE_CUTOUT`. Produces the .csv files in `FINAL_STORAGE_PREDICTION_FILES/species/zoo/individual/total/raw_csv/`.
  - `SKIP_BEHAVIOR_TOTAL_SF` = total classification task and single frame.
  - `SKIP_BEHAVIOR_TOTAL_MF` = total classification task and multiple frame.
  - `SKIP_BEHAVIOR_BINARY_SF` = binary classification task and single frame.
  - `SKIP_BEHAVIOR_BINARY_MF` = binary classification task and multiple frame.

- SKIP\_MOVING\_FILES = Copies data from TMP\_STORAGE\_CUTOUT into FINAL\_STORAGE\_CUTOUT.
  - SKIP\_REMOVING\_TEMPORARY\_FILES = Deletes all files in TMP\_STORAGE\_CUTOUT and TMP\_STORAGE\_IMAGES
  - SKIP\_PP = Post-processing and data visualisation. The folder /raw\_csv/ from FINAL\_STORAGE\_PREDICTION\_FILES and the folder /position\_files/ from FINAL\_STORAGE\_CUTOUT are required for all nights. Produces an .xlsx-sheet as well as a graphic („timeline“).
    - SKIP\_PP\_TOTAL
    - SKIP\_PP\_BINARY
  - SKIP\_OD\_DENSITY = Produces an overview file indicating in how many images the object detector could draw a bounding box. Required to ensure accurate evaluation of the system. Requires /position\_files/ from FINAL\_STORAGE\_CUTOUT.
- 
- BASE\_PATH\_DATA = Path to video storage, DATA\_STORAGE of readme.md [string]
  - BEHAVIOR\_NETWORK\_BASEPATH = Path (folder) in which the action classifiers are stored. BOVIDS decides which of those networks needs to be used by global\_configuration.py [string]
  - OD\_NETWORK\_BASEPATH = Path (folder) in which the object detectors are stored. BOVIDS decides which of those networks needs to be used by global\_configuration.py [string]
  - CUT\_OFF = Step 1 (extraction of images) will stop after CUT\_OFF many time-intervals. If BOVIDS is only used to generate some of those images to check videostream merging or to decide truncation effects, this might be set to 1. 7200 corresponds to 14 hour videos. [integer]

### **Predict\_csv.py:**

- AI\_LIBRARY\_LOCAL: path to the folder prediction of this repository (e.g. '/home/user/bovids/prediction/') [string]

- AI\_LIBRARY\_GLOBAL: path to the folder global of this repository (e.g. '/home/mnt/fancy\_nas/bovids/global/') [string]
- YOLO\_LIBRARY: path to the yolov4 library of this repository (e.g. '/home/mnt/fancy\_nas/bovids/global/yolo-v4-tf.keras-master/') [string]
- GPU\_TO\_USE = Decide which GPU is used. Set to '0' if only one GPU is available. [string]

### Step 3 – run the script:

- Save configuration.py.
- Either run predict\_csv.py in spyder's console, or if this causes problems, use the terminal (bovids environment) and type
  - *python ...../bovids/predict/predict\_csv.py*