The manual of *LabGym* (v1.9)

1. Installation of *LabGym* (https://github.com/umyelab/LabGym)[1]

a. Download Python3 [2] (version >= 3.9.7) from its official website (https://www.python.org/downloads/).

Note: recommend not to download the latest version of Python3 as some Python libraries used in *LabGym* might be updated yet to be compatible with the latest version of Python3.

b. Open the Terminal (Mac) or Command Prompt (Windows)

c. To install *LabGym*, in the Terminal / Command Prompt, type:

```
python3 -m pip install LabGym
```

2. Initiate the graphical user interface (GUI) of *LabGym* for each use

a. Open the Terminal (Mac) or Command Prompt (Windows)

b. To activate Python3 interactive shell, in the Terminal / Command Prompt, type:

```
python3
```

c. After the Python3 interactive shell is activated, in the Terminal / Command Prompt, type:

```
from LabGym import gui
```

d. Then in the Terminal / Command Prompt, type:

```
gui.gui()
```

Now the GUI of *LabGym* is initiated and is ready to use.

3. Explanations and tips on each option in the GUI of *LabGym*

The GUI of *LabGym* consists of 9 functional units: '***Generate Object Images***', '***Train Detectors***', '***Test Detectors***', '***Generate Behavior Examples***', '***Train Categorizers***', '***Test Categorizers***', '***Process Data***', '***Analyze Behaviors***' and '***Mine Analysis Results***'.

3.1. '*Generate Object Images*'

Use this functional unit to extract frames (images) from videos. The extract frames can be used for annotating the outline of animals / objects in them. The annotated images can then be used to train a Detector in '***Train Detectors***' functional unit.

(Button) 'Select the video(s) to generate animal / object images'

Select one or more videos to extract frames. Common video formats (mp4, mov, avi, m4v, mkv, mpg, mpeg) are supported except for wmv format.

- (Pop-up option) '(Optional) resize the frames?'

Users can specify whether to resize the frames of the videos. The resizing will keep the original "width / height" ratio.

(Button) 'Select a folder to store the generated image examples'

The extracted frames will be stored in this folder.

(Button) 'Specify when generating image examples should begin (unit: second)'

The beginning time to extract frames.

(Button) 'Specify how long generating examples should last (unit: second)'

The duration for generating image examples.

(Button) 'Specify how many frames to skip when generating two consecutive images'

To increase the generalizability of a Detector, users should make the training images as diverse as possible. Therefore, repeated images, for example, frames extracted from a period when the animals are in the same location (they look the same), can be avoided by setting an interval between two consecutive extractions.

(Button) 'Start to generate image examples'

After images are generated, users may use online annotation tools such as Roboflow (https://roboflow.com) or CVAT (https://www.cvat.ai) or VGG Image Annotator (https://www.robots.ox.ac.uk/~vgg/software/via/) to annotated the outlines (NOT bounding box) of animals / objects to detect in images and use them to train a Detector in '***Train Detectors***' functional unit. When annotated images, make sure to select "Instance Segmentation" for the annotation type. When export the annotation file, make sure to select "COCO instance segmentation" format, which will be a '*.json' file.

3.2. '*Train Detectors*'

Use this functional unit to train Detectron2-based (https://github.com/facebookresearch/detectron2)[3] Detectors. The trained Detectors will be listed in the detection methods in '***Test Detectors***', '***Generate Behavior Examples***', and '***Analyze Behaviors***' functional units. A well-trained Detector can detect and segment animals / objects of interest in various background and is useful for videos with changing illumination / background or differentiate different individuals even when they entangle with others.

(Button) 'Select the folder containing all the training images'

The folder that stores all the training images.

(Button) 'Select the *.json annotation file'

The .json file for the annotation that was done on all the training images. When export this file after annotation, make sure to select “COCO instance segmentation” format.

(Button) ‘Specify the inferencing framesize for the Detector to train’

This is the input size of a Detector, which determines the speed-accuracy trade-off of Detector performance. Larger size means higher accuracy but slower speed. This size should be divisible by 32 to achieve best training efficiency. Users can start with small size such as 256 and increase it if the accuracy is not ideal, especially when there is only one animal in the videos, or the animal occupies most area in a frame. Size that is smaller than 192 or larger than 1024 is not recommended for general scenarios.

(Button) ‘Specify the iteration number for the Detector training’

This is the number of training loops. More iterations typically yield better accuracy but too many may cause overfitting. A number between 50 ~ 1000 is good for most scenarios. Instead of increasing the iterations, users may rather increase the diversity and amount of training images.

(Button) ‘Start to train the Detector’

Users need to name the Detector to train. English letters, numbers, underscore “_”, or hyphen “-” are acceptable but do not use special characters such as “@” or “^”.

3.3. ‘*Test Detectors*’

Users can input a video in this functional unit to test a trained Detector. The outline of detected animals / objects will be annotated in the copy of testing video.

(Button) ‘Select a Detector to test’

As indicated by the title.

(Button) ‘Select a testing video’

The testing will start at the beginning of the selected video.

(Button) ‘Specify the testing duration (unit: second)’

As indicated by the title.

(Button) ‘Select a folder to store the annotated video’

As indicated by the title.

(Button) ‘Test the selected Detector’

As indicated by the title.

(Button) ‘Delete a Detector’

The deletion CANNOT be restored.

3.4. ‘***Generate Behavior Examples***’

Use this to generate stand-alone, visualizable behavior examples from videos. A behavior example pair contains an “animation” and its paired “pattern image”. These behavior example pairs can be sorted into different folders according to their behavior types (categories) and input to the ‘***Train Categorizers***’ functional unit to train Categorizers for identifying user-defined behaviors. They can also be sorted and input to the ‘***Test Categorizers***’ functional unit for testing the accuracy of a trained Categorizer.

(Button) ‘Select the video(s) to generate behavior examples’

Select one or more videos for generating behavior examples. Common video formats (mp4, mov, avi, m4v, mkv, mpg, mpeg) are supported except for wmv format.

- (Pop-up option) '(Optional) resize the frames?'

The resizing will keep the original "width / height" ratio and can be used to increase the speed for generating behavior examples.

(Button) 'Select a folder to store the generated behavior examples'

Will create a subfolder for each video in the selected folder. The name of each subfolder is the video file name. In the folder for each video, a subfolder will be created for an animal in this video, which is named after the animal identity (ID) and stores the behavior examples for this animal.

(Button) 'Specify when generating behavior examples should begin (unit: second)'

- (Pop-up option) 'Illumination shifts?'

Specify whether there are sudden bright-to-dark or dark-to-bright illumination transitions. If choose 'Yes', there will be 3 options to specify the beginning time: '*Automatic (for light on and off)*', '*Decode from filenames: _bt_*', and '*Enter a time point*'. If choose 'No', only the latter two options can be chosen.

'*Automatic (for light on and off)*' is for videos involving lighting on and off (e.g., optogenetics) and the first time point of illumination change will be automatically detected and used as the beginning time to generate behavior examples.

'*Decode from filenames: _bt_*' can be used if multiple videos are selected and the beginning time for each video is different. Users need to add a code tag '_bt_' ('b' stands for 'beginning time' and 't' should be number) into the file name of each video to let *LabGym* decode the beginning time. For example, suppose the video filename is 'A.avi' and the user wants the

beginning time to be at the 12.35 second, the user can rename the video file to ‘A_b12.35.avi’ (the ‘_’ is unnecessary if it is at the end of file name).

'*Enter a time point*' can be used if only one video is selected, or multiple videos share the same beginning time.

(Button) ‘Specify how long generating examples should last (unit: second)’

Must be an integer number. All the videos in a batch will share the same duration.

(Button) ‘Specify the number of animals in a video’

There are two options: ‘*Decode from filenames: _nn_*' and *'Enter the number of animals*'.

*'Decode from filenames: _nn_*' can be used if multiple videos are selected and the number of animals in each video is different. A code tag ‘_nn_’ (the first ‘n’ stands for ‘number of animals’ and the second ‘n’ should be an integer) needs to be added into the file name to let *LabGym* decode the animal number. For example, suppose the video file name is ‘A.avi’ and the number of animals in this video is 8, rename the video file to ‘A_n8.avi’ (the last ‘_’ is unnecessary if it is at the end of file name).

'*Enter the number of animals*' can be used if only one video is selected, or multiple videos share the same animal number.

(Button) ‘Specify the method to detect animals or objects’

There are two options: ‘*Subtract background*’ and ‘*Use trained Detectors*’

‘*Subtract background*’ is the first choice for videos in which the background is static, the illumination is stable overtime, and the total behavior events are more important than animal IDs, because this method is fast and accurate in such videos but cannot distinguish entangled animals and the IDs might switch after they re-separate. When using this method, users need to specify the scenarios of their experiments: '*Animal brighter than background*', '*Animal darker*

than background', or '*Hard to tell*'. A pop-up option '(Optional) load existing background?' will need to be specified. *LabGym* will output the extracted backgrounds (as images) for each video processed. Therefore, this pop-up option can be used to save the step of background extraction if the background for this video has already been extracted and output. Users also need to specify whether the illumination in videos is unstable. If the illumination is very stable overtime, choose 'No' to increase the processing speed. Finally, users need to specify a time window for background extraction. An appropriate time window for background extraction should be a period (typically 10~60 seconds) during which the animals move around. This time window should be as short as possible for increasing processing speed. There are 3 options to specify this time window: (1) '*Use the entire duration of a video*': not recommended if processing speed is critical. (2) '*Decode from filenames: _xst_ and _xet_*': can be used if multiple videos are selected and the time window for background extraction is different for each video. Two code tags '_xst_ and _xet_' ('xs' stands for 'extraction start time', 'xe' stands for 'extraction end time', and 't' should be an integer) need to be added into the file name of each video to let *LabGym* decode the time window. For example, suppose the video file name is 'A.avi' and the time window for background extraction is from the 25th second to the 47th second, rename the video file to 'A_xs25_xe47.avi' (the '_' is unnecessary if it is at the end of file name). (3) '*Enter two time points*': can used if only one video is selected, or multiple videos share the same time window for background extraction.

'*Use trained Detectors*' is useful in any kind of videos or experimental settings. It is also useful for differentiate individual animals when they entangle. The only caveat of this method is slow.

(Button) 'Specify the number of frames for an animation / pattern image'

The animations and their paired pattern images in the behavior examples spans a user-defined duration (the number of frames, an integer), which should approximate the duration of a

behavior episode. This duration needs to be the same across all the behavior examples that are used to train one Categorizer. If the duration of different behavior episode is different, use the longest one.

(Button) 'Specify how many frames to skip when generating two consecutive behavior examples'

LabGym generates a pair of behavior example which spans a user-defined duration at a frame. If two consecutively generated examples are too close in time, say, one generated at the 10^{th} frame and the other at 12^{th} frame, suppose their duration is 10 frames, they will have 8 overlapping frames. These two behavior examples are too similar and will hurt the training efficiency when training a Categorizer. Generating too many such similar examples will make the example sorting labor intensive. Therefore, users can choose to skip certain frames between the two consecutively generated examples. A practical recommendation to achieve a balance between getting the perfect examples that just span the behavior episode and reducing the labor is to set this number as the half of the duration for a behavior episode.

(Button) 'Start to generate behavior examples'

- (Pop-up option) 'Including background?'

  Choose 'No' if the background information is behavior irrelevant.

- (Pop-up option) 'Including body parts?'

  Choose 'Yes' if the motion pattern of individual body parts such as limbs or noses are critical for behavioral identification. If choose 'Yes', a 'STD' need to be entered, which should be an integer between 0 and 255. The STD value decides the threshold to show the how many 'motion pixels' of the body parts in the pattern images. Lager STD, less motion pixels will be shown.

3.5. '***Train Categorizers***'

This functional unit has two modules: one is for preparing the training examples; the other is for using the prepared training examples to train a Categorizer. The Categorizers can be customized by users, for example, whether to include both Animation Analyzer and Pattern Recognizer and how complex they are. 'Preparing training examples' means making the training examples suitable to be directly input into Categorizers for training. Before this step, users need to first select and sort the behavior examples generated by '***Generate Behavior Examples***' functional unit into different folders named after the behavior names. The '*Sort Behavior Examples*' module in '***Process Data***' functional unit can be used to help sorting. The trained Categorizers will be automatically added into the Categorizer list for the usage in '***Test Categorizers***' and '***Analyze Behaviors***' functional units.

<u>(Button) 'Select the folder that stores the sorted behavior examples'</u>

This folder should contain all the sorted behavior examples. Each subfolder in this folder should contain behavior examples of a behavior type. The names of the subfolders will be read by *LabGym* as the behavior names.

<u>(Button) 'Select a new folder to store all the prepared behavior examples'</u>

This folder will store all the prepared behavior examples and can be directly used for training. Preparing behavior examples is the process of copying all the examples into this folder and renaming them to put behavior name labels to their file names.

- (Pop-up option) 'Resize the frames?'

During the training, the behavior examples will be resized to the input frame size of the Categorizer. Users may choose to downsize the frame / image size at this step to increase the training speed. The targeted frame / image size should not be smaller than the targeted input size of the Categorizers to train.

- (Pop-up option) 'Background-free animations?'

Users need to specify whether the animations used for training include the background.

(Button) ‘Start to prepare the training examples’

As indicated by the title.

(Button) ‘Specify the type / complexity of the Categorizer to train’

- (Pop-up option) 'Categorizer types'

There are two types of Categorizers. One is the Categorizer with both Animation Analyzer and Pattern Recognizer; the other is the Categorizer with only Pattern Recognizer. The latter is much faster but might be a little less accurate.

- (Pop-up option) 'Complexity level'

There are 7 complexity levels (1~7, from simpler to more complex) of either Animation Analyzer or Pattern Recognizer. The higher complexity level, the deeper of neural networks (more layers and more complex structures), and the slower of the training / analysis speed. Users may always start with the simplest ones and increase the complexity gradually until the accuracy is satisfying. Users may make the complexity levels of Animation Analyzer and Pattern Recognizer the same or higher for the latter because the latter analyzes pattern images that have 3 color channels, and the these colors are important information that indicates temporal sequence of the behaviors.

(Button) ‘Specify the input shape for Animation Analyzer / Pattern Recognizer’

The input frame / image size should be an even integer and greater than 8. The greater frame / image size, the wider of neural networks (more parameters in each layer), and the slower of the training / analysis speed. Users may always start with the smaller input frame /

image sizes and increase them gradually until the accuracy is satisfying. And always go deeper (increase complexity level) first, rather than go wider (increase input frame / image size) first.

- (Pop-up option) 'Grayscale Animation Analyzer?'

  Choose ‘Yes’ if the color of animals is behavior irrelevant.

(Button) ‘Specify the number of frames for an animation / pattern image’

This should be the same number as the duration of behavior examples.

(Button) ‘Select the folder that stores all the prepared training examples’

The folder that stores all the prepared training examples.

- (Pop-up option) 'Background-free animations?'

  Specify whether the animations in the behavior examples include background.

- (Pop-up option) 'Body parts in pattern images?'

  Specify whether the pattern images in the behavior examples include body parts. If choose ‘Yes’, users need to enter the STD. The value of STD should match that of the generated patten images. This information can be found in the file names of the generated pattern images. If users choose ‘including body parts’ when generating pattern images and set the STD to 50, the file name of generated pattern images will be ‘xxx_std50.jpg’.

(Button) ‘Specify the methods for data augmentation’

The data augmentation is a way to artificially increase the amount and diversity of the training examples. Briefly, '*rotation*' will rotate the animal blobs; '*flipping*' will flip the animal blobs; '*brightening*' and '*dimming*' will increase and decrease the brightness of the animal blobs; '*shearing*' will distort the animal blobs; '*rescaling*' will change the width / height ratio of the animal blobs; '*deletion*' will delete one or two frames in the animations (mimic the scenario in which animals are not detected in one or two frames during analysis).

- (Pop-up option) 'Use default augmentation?'

  Default augmentation will use '*rotation*', '*flipping*', '*brightening*' and '*dimming*'.

- (Pop-up option) 'Augment validation data?'

  If the total number of behavior example pairs used for training a Categorizer is less than 1,000 before augmentation, users may choose to augment validation data as well.

(Button) 'Select a folder to export training reports'

As indicated by the title.

(Button) 'Start to train the Categorizer'

Users need to give a name to the Categorizer to train. English letters, numbers, underscore "_", or hyphen "-" are acceptable but do not use special characters such as "@" or "^".

3.6. '***Test Categorizers***'

Use this to test the accuracy of a trained Categorizer. Users may also delete a trained Categorizer in this functional unit. Before testing, users first need to use '***Generate Behavior Examples***' functional unit to generate some behavior examples and sort them to build a ground truth dataset.

(Button) 'Select a Categorizer to test'

As indicated by the title.

(Button) 'Select the folder that stores the sorted ground truth behavior examples'

The names of its subfolders should be the behavior names. Each subfolder stores the behavior examples of this behavior type.

(Button) ‘Select a folder to export testing reports’

As indicated by the title.

(Button) ‘Test the selected Categorizer’

As indicated by the title.

(Button) ‘Delete a Categorizer’

The deletion CANNOT be restored.

3.7. ‘***Process Data***’

Two modules in this functional unit: ‘*Preprocess Videos*’ and ‘*Sort Behavior Examples*’. The former is for preprocessing the videos to make them more suitable for analysis; the latter is for using shortcut keys to sort behavior examples in an easier way.

‘*Preprocess Videos*’

(Button) ‘Select the video(s) for preprocessing’

Select one or more videos. Common video formats (mp4, mov, avi, m4v, mkv, mpg, mpeg) are supported except for wmv format.

(Button) ‘Select a folder to store the processed videos’

Will create a subfolder for each video.

(Button) ‘Specify when the preprocessing should begin (unit: second)’

Different videos will use the same beginning time.

(Button) ‘Specify whether to trim a video into shorter video clips’

If choose ‘Yes’, a duration needs to be entered, which is for every trimmed video clip.

(Button) ‘Specify whether to crop the video frames’

Cropping frames to exclude irrelevant areas in the frames can increase the analysis efficiency. Users need to specify the 4 corners of the cropping window. The first frame of the first video will be shown and put in a coordinate for users to see and determined the 4 points: most left, most right, most top, and most bottom. Close this frame to enter the 4 points. This cropping window will be applied for all videos selected.

(Button) ‘Specify whether to enhance the contrast in videos’

Enhancing video contrast will increase the detection accuracy especially when the detection method is background subtraction based. Enter the contrast value and the first frame of the first video will be shown to be applied with this value. Close this frame to specify whether to apply the entered contrast value or re-enter the value.

(Button) ‘Start to preprocess the videos’

As indicated by the title.

‘*Sort Behavior Examples*’

(Button) ‘Select the folder that stores unsorted behavior examples’

Each behavior example is a pair of one animation and one pattern image, as generated in ‘***Generate Behavior Examples***’ functional unit.

(Button) ‘Select the folder to store the sorted behavior examples’

A subfolder will be created for each behavior type under the behavior name.

(Button) 'Enter the behavior names and corresponding shortcut keys'

When press a shortcut key, the behavior example pair will be automatically moved to the corresponding folder of this behavior type.

(Button) 'Sort behavior examples'

User will see each example pair in the screen one by one and can use shortcut keys to sort them into folders of the behavior types.

3.8. '*Analyze Behaviors*'

Identify and quantify user-defined behaviors in videos.

(Button) 'Select a Categorizer for behavior classification'

Choose a Categorizer for behavior classification in analysis, or just let *LabGym* track the animals and calculate their motion parameters and body kinematics. If the former, users need to specify a 'Uncertainty level' for the Categorizer. This number in percentage determines the threshold for the Categorizer to output an 'NA' for behavioral classification. For example, when the probabilities of behavior A, B, and C are 60%, 10%, and 30%, respectively. If the uncertain level is set to be 31, the behavior classification will be 'NA', as the uncertainty level exceeds the difference between probability of the highest-likely behavior (A, 60%) and the second highest-likely behavior (C, 30%), which is 30%. Setting uncertainty level can reduce the possible false positives in behavior classification since ambiguous classification will be output as an 'NA'. If choose to not classify behaviors, users need to specify a time window for calculating the motion parameters and body kinematics.

(Button) ‘Select the video(s) for behavior analysis’

Select one or more videos for a behavior analysis batch. One analysis batch will yield one raster plot showing the behavior events of all the animals in all selected videos. Common video formats (mp4, mov, avi, m4v, mkv, mpg, mpeg) are supported except for wmv format.

- (Pop-up option) ‘(Optional) resize the frames?’

Downsizing the frame is **highly recommended**, which will exponentially increase the analysis speed. The resizing will keep the original “width / height” ratio. The analysis accuracy will not decline if the animal size after downsizing is still larger than the input size of the Categorizer used for analysis. For example, suppose the original size of a video frame is 1000 x 500, the size of an animal is approximately 1/4 to that of a frame, and input size of the Categorizer is 48 x 48. After downsizing the video frames to 500 x 250, the animal size is roughly 125 x 63 and still is larger than the input size of the Categorizer (48 x 48). In this scenario, the analysis accuracy will not decline since the animal blob will be downsized to 48 x 48 anyway when input to the Categorizer for analysis.

(Button) ‘Select a folder to store the analysis results’

Will create a subfolder for each video in the selected folder. Each subfolder is named after the file name of the video and stores the detailed analysis results for this video.

(Button) ‘Specify when the analysis should begin (unit: second)’

- (Pop-up option) ‘Illumination shifts?’

Specify whether there are sudden bright-to-dark or dark-to-bright illumination transitions. If choose ‘Yes’, there will be 3 options to specify the beginning time: '*Automatic (for light on and off)*', '*Decode from filenames: _bt_*', and '*Enter a time point*'. If choose ‘No’, only the latter two options can be chosen.

'*Automatic (for light on and off)*' is for videos involving lighting on and off (e.g., optogenetics) and the first time point of illumination change will be automatically detected and used as the beginning time to generate behavior examples.

'*Decode from filenames: _bt_*' can be used if multiple videos are selected and the beginning time for each video is different. Users need to add a code tag '_bt_' ('b' stands for 'beginning time' and 't' should be number) into the file name of each video to let *LabGym* decode the beginning time. For example, suppose the video filename is 'A.avi' and the user wants the beginning time to be at the 12.35 second, the user can rename the video file to 'A_b12.35.avi' (the '_' is unnecessary if it is at the end of file name).

'*Enter a time point*' can be used if only one video is selected, or multiple videos share the same beginning time.

(Button) 'Specify the analysis duration (unit: second)'

Must be an integer number. All the videos in a batch will share the same duration.

(Button) 'Specify the number of animals in a video'

There are two options: '*Decode from filenames: _nn_*' and *'Enter the number of animals*'.

*'Decode from filenames: _nn_*' can be used if multiple videos are selected and the number of animals in each video is different. A code tag '_nn_' (the first 'n' stands for 'number of animals' and the second 'n' should be an integer) needs to be added into the file name to let *LabGym* decode the animal number. For example, suppose the video file name is 'A.avi' and the number of animals in this video is 8, rename the video file to 'A_n8.avi' (the last '_' is unnecessary if it is at the end of file name).

'*Enter the number of animals*' can be used if only one video is selected, or multiple videos share the same animal number.

- (Pop-up option) 'Relink IDs?'

Sometimes animals might be lost track for several frames and re-tracked after that. If an animal is lost track for over 2 seconds, its ID and the matrix linked with the ID for storing information of this animal will be deregistered temporally. If choose not to '*Relink the IDs*', a re-tracked animal will be registered to a new ID-matrix. In this scenario, a deregistered ID-matrix will never be re-initiated. If users choose to 'relink the IDs', a re-tracked animal will be linked to a deregistered ID-matrix unless there is no available deregistered ID-matrix and the re-tracked animal will then be registered to a new ID-matrix. An animal that is lost track for over 50% of the entire duration of analysis will be excluded from the analysis results permanently.

(Button) 'Specify the method to detect animals or objects'

There are two options: '*Subtract background*' and '*Use trained Detectors*'

'*Subtract background*' is the first choice for videos in which the background is static, the illumination is stable overtime, and the total behavior events are more important than animal IDs, because this method is fast and accurate in such videos but cannot distinguish entangled animals and the IDs might switch after they re-separate. When using this method, users need to specify the scenarios of their experiments: '*Animal brighter than background*', '*Animal darker than background*', or '*Hard to tell*'. A pop-up option '(Optional) load existing background?' will need to be specified. *LabGym* will output the extracted backgrounds (as images) for each video processed. Therefore, this pop-up option can be used to save the step of background extraction if the background for this video has already been extracted and output. Users also need to specify whether the illumination in videos is unstable. If the illumination is very stable overtime, choose 'No' to increase the processing speed. Finally, users need to specify a time window for background extraction. An appropriate time window for background extraction should be a period (typically 10~60 seconds) during which the animals move around. This time window should be as short as possible for increasing processing speed. There are 3 options to specify this time window: (1) '*Use the entire duration of a video*': not recommended if processing speed

is critical. (2) '*Decode from filenames: _xst_ and _xet_*': can be used if multiple videos are selected and the time window for background extraction is different for each video. Two code tags '_xst_ and _xet_' ('xs' stands for 'extraction start time', 'xe' stands for 'extraction end time', and 't' should be an integer) need to be added into the file name of each video to let *LabGym* decode the time window. For example, suppose the video file name is 'A.avi' and the time window for background extraction is from the 25th second to the 47th second, rename the video file to 'A_xs25_xe47.avi' (the '_' is unnecessary if it is at the end of file name). (3) '*Enter two time points*': can used if only one video is selected, or multiple videos share the same time window for background extraction.

'*Use trained Detectors*' is useful in any kind of videos or experimental settings. It is also useful for differentiate individual animals when they entangle. The only caveat of this method is slow.

(Button) 'Select the behaviors for annotations and plots'

The behavior categories are determined by the selected Categorizer. Users may select which behaviors to show in the annotated videos and the raster plot for behavior events.

- (Pop-up option) 'Specify colors for behaviors?'

Specify a color to represent a behavior category in the annotated videos and the raster plot for behavior events. In the annotated videos, the value of % confidence of behavior categorization will be shown; in the raster plot, the color intensity indicates the value of % confidence, from 0% of the color intensity (clear) indicating 0% of the confidence, to 100% of the color intensity indicating 100% of the confidence. If users choose not to specify the colors, *LabGym* will use the default colors to represent the behaviors.

- (Pop-up option) 'Legend in video?'

Specify whether to show the legend of behavior names in the annotated videos.

(Button) 'Select the quantitative measurements for each behavior'

There are 13 quantitative measurements (parameters) for each behavior for users to choose:

- ◊ The *count* is the summary of the behavioral frequencies, which is the occurrence number of a behavior within the entire duration of analysis. Consecutive single occurrences (at a single frame) of the same behavior are considered as one count.
- ◊ The *latency* is the summary of how soon a behavior starts, which is the time starting from the beginning of the analysis to the time point that the behavior occurs for the first time.
- ◊ The *duration* is the summary of how persistent a behavior is, which is the total time of a behavior within the entire duration of analysis.
- ◊ The *speed* is the summary of how fast the animal moves when performing a behavior, which is the total distance traveled (can be back and forth) (d) (between the two centers of mass of the animal) during the time window (t_w) for categorizing the behavior divided by t_w.
- ◊ The *velocity* is the summary of how efficient the animal's movement is when performing a behavior, which is the maximum shortest distance between the start and the end positions (dt) (between the two centers of mass of the animal) divided by the time (t) that such displacement takes place.
- ◊ The *acceleration* / *velocity reduction* is the summary of how fast the animal's velocity changes while performing a behavior, which is the difference between maximum velocity (v_{max}) and minimum velocity (v_{min}) divided by the time (t_v) that such velocity change takes place.
- ◊ The *distance* is the total distance traveled of the animal by performing a behavior within the entire duration of analysis.
- ◊ The *intensity (area)* / *intensity (length)* is the summary of how intense a behavior is, which is the accumulated proportional changes of the animal body area (a) / length (l) between frames divided by the time window for categorizing the behaviors (t_w) when performing a behavior.
- ◊ The *magnitude (area)* / *magnitude (length)* is the summary of the motion magnitude, which is the maximum proportional change in animal body area (a) or length (l) when performing a behavior.

◊ The *vigor (area)* / *vigor (length)* is the summary of how vigorous a behavior is, which is the magnitude (area) / magnitude (length) divided by the time (t_a or t_l) that such a change takes place.

Details on how they are calculated are in *LabGym* paper [1].

- (Pop-up option) 'Normalize the distances?'

If choose 'No', all the distances will be output in pixels. The unit of all the distance related measurements will be 'pixel' or pixel related. If choose 'Yes', all the distances (calculated in pixels) will be normalized to (divided by) the size of a single animal (also calculated in pixels). In this scenario, all distance related measurements will be normalized measurements (e.g., normalized speed) and will not have a unit. In this way, users do not need to worry about the ratio of pixel / actual size (length) across different videos with different recoding methods, if the animals used in these videos are of similar size. The ratio of pixel / actual size (length) is not easy to obtain and is subject to change easily (e.g., when the zoom-in level changes). With the option of normalizing distances to the size of a single animal, users can compare the analysis results across different recordings or experimental sessions without worrying about the changes in the ratio of pixel / actual size (length).

(Button) 'Start to analyze the behaviors'

As indicated by the title.

3.9. '***Mine Analysis Results***'

This functional unit automatically performs parametric / non-parametric statistical analysis among groups that users selected, according to the data distribution, to compare the mean / median of different groups and display the significant findings.

(Button) 'Select the folder that stores the data files'

This folder should contain the subfolders of each control / experimental group. Users can directly move the *LabGym* analysis output folders (each folder contains one raster plot) into this folder.

- (Pop-up option) 'Paired data?'

Specify whether the data is paired. Different statistical analysis method will be applied accordingly.

(Button) ‘Select the control group’

All groups will be listed, and users can select one group as control for post-hoc comparison. If no control group is selected, post-hoc comparison will be performed between each pair of two groups.

(Button) ‘Select the folder to store the data mining results’

As indicated by the title.

(Button) ‘Start to mine data’

A p value needs to be specified for determine the significance threshold for statistical analysis. The Shapiro test will be first performed to assess the normality of data distribution. For normally distributed data, if unpaired, unpaired t-test for 2 groups, ANOVA for more than 2 groups, with either Tukey (comparing each pair) or Dunnett's (comparing all groups against the control group) posthoc comparison; if paired, paired t-test for 2 groups, ANOVA for more than 2 groups, with either Tukey (comparing each pair) or Dunnett's (comparing all groups against the control group) posthoc comparison. For data that is not normally distributed, if unpaired, Mann Whitney U test for 2 groups, Kruskal Wallis for more than 2 groups, with Dunn's posthoc comparison for both comparison of all groups and against control; if paired, Wilcoxon test for 2

groups, Friedman for more than 2 groups with Dunn's posthoc. The selections of the tests are consistent with those in GraphPad Prism 9.

REFERENCES

1. Hu, Y., Ferrario, C.R., Maitland, A.D., Ionides, R.B., Ghimire, A., Watson, B., Iwasaki, K., White, H., Xi, Y., Zhou, J., and Ye, B. (2023). LabGym: Quantification of user-defined animal behaviors using learning-based holistic assessment. Cell Rep Methods *3*, 100415. 10.1016/j.crmeth.2023.100415.
2. Van Rossum, G., and Drake, F.L. (2000). Python reference manual (iUniverse Indiana).
3. Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2.