The manual of *LabGym*

1. Installation of *LabGym* (https://github.com/umyelab/LabGym) (Hu et al. 2023)

a. Download Python3 [1] (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 8 functional units: '***Generate Object Images***', '***Train Detectors***', '***Test Detectors***', '***Generate Behavior Examples***', '***Train Categorizers***', '***Test Categorizers***', '***Analyze Behaviors***' and '***Mine Analysis Results***'.

3.1. '*Generate Object Images*'

Coming soon!

3.2. '*Train Detectors*'

Coming soon!

3.3. '*Test Detectors*'

Coming soon!

3.4. '*Generate Behavior Examples*'

This functional unit is used 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 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?'

Users can specify whether to resize the frames of the videos. The resizing will keep the original “width / height” ratio. Downsizing the frames will significantly increase the processing speed and **is highly recommended**. 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, the size of an animal is approximately 1/4 to that of a frame. Suppose the original size of a video frame is 1000 x 500, and the input size of the Categorizer is 48 x 48. If users downsize the video frames to 500 x 250, the animal size is approximately 125 x 63 after downsizing, which 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 generated behavior examples’

Will create a subfolder for each video in the selected folder. The name of each subfolder is the file name of the video. In the folder for each video, a subfolder will be created for each animal in this video, which is named after the animal identity (ID) and stores the generated 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 typically for videos involving lighting on and off (e.g., optogenetics) and the time point of sudden illumination changes for the first time 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 for generating behavior examples and the beginning time for each video is different. If choose this option, users need to add a code tag '_bt_' ('b' stands for 'beginning time' and 't' should be an integer or floating number) into the file name of each video to let *LabGym* decode the beginning time. For example, if the file name of a video is 'ABC-0102.avi' and the user wants the beginning time to be at the 12.35 second, the user can rename the video file to 'ABC-0102_b12.35_.avi' or 'ABC-0102_b12.35.avi' (the last '_' is unnecessary if it is at the end of the file name).

'*Enter a time point*' can be used if only one video is selected, or multiple videos are selected and the beginning time for each video is the same. In this scenario, users can manually enter a time point (should be an integer).

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

The number entered must be an integer. All the videos selected will use this duration to generate behavior examples.

(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 for generating behavior examples and the number of animals in each video is different. If choose this option, users need to add a code tag '_nn_' (the first 'n' stands for 'number of animals' and the second 'n' should be an integer) into the file name of each video to let *LabGym* decode the animal number. For example, if the file name of a video is 'ABC-0102.avi' and the number of animals in this video is 8, the user can rename the video file to 'ABC-0102_n8_.avi' or 'ABC-0102_n8.avi' (the last '_' is unnecessary if it is at the end of the file name).

'*Enter the number of animals*' can be used if only one video is selected, or if multiple videos are selected and the number of animals in each video is the same. In this scenario, users can manually enter a number (should be an integer).

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

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

'*Subtract background*' is useful when the backgrounds of the videos are static and the illumination in the video is stable overtime. And if this is the case, this method is much faster and more accurate in detecting the animals. However, this method is not useful for social behaviors as it cannot distinguish individual animals when they have body contact. After selecting this method, users need to then specify the scenarios of their experiments: '*Animal brighter than background*', '*Animal darker than background*', or '*Hard to tell*'. Then a pop-up option '(Optional) load existing background?' will need to be specified. The animal detection and tracking in *LabGym* is based on background subtraction. *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 (to save time) if the background for this video has already been extracted and output. Note that if choose this option, the loaded backgrounds will be used for background subtraction for all the selected videos. After that, users will be asked whether the illumination in videos is unstable. If the illumination in the videos is very stable overtime, users can 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. If select a time window during which the animals always stay at one place, the animals will be considered as the background and the detection and tracking will fail. The time window should be as short as possible since longer time window takes longer time and more memory to process. There are 3 options to specify this time window: (1) '*Use the entire duration of a video*'.

This is generally not recommended if the processing speed is critical. Do not use this option in any following scenario unless the computer memory is larger than 64 GB: the video frame size is over 2000 x 2000 unless users downsize them (check 'Select the video(s) to generate behavior examples' to see how downsize the video frame); the video duration is over 5 minutes; the video fps is over 60. (2) '*Decode from filenames: _xst_ and _xet_*'. This can be used if multiple videos are selected for generating behavior examples and the time window for background extraction is different for each video. If choose this option, users need to add two code tags '_xst_ and _xet_' ('xs' stands for 'extraction start time', 'xe' stands for 'extraction end time', and 't' should be an integer) into the file name of each video to let *LabGym* decode the time window. For example, if the file name of a video is 'ABC-0102.avi' and the time window for background extraction is from the 25th second to the 47th second, the user can rename the video file to 'ABC-0102_xs25_xe47_.avi' or 'ABC-0102_xs25_xe47.avi' (the last '_' is unnecessary if it is at the end of the file name). (3) '*Enter two time points*'. This be used if only one video is selected, or if multiple videos are selected and the time window for background extraction for each video is the same. In this scenario, users can manually enter two time points (should be integers).

'*Use trained detectors*' is useful in any kind of videos or experimental settings. It is also useful for social behaviors. The only caveat of this method is slow. It's coming soon.

(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, should be an integer), which should approximate the duration of a behavior episode. The animation duration needs to be the same across all the animations that are used to train one Categorizer. If the duration of different behaviors is different, use the longest one as the animation duration.

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

By default, *LabGym* generates a pair of behavior example at each frame during the specified duration. For example, suppose the animation duration is 10 frames and the first animation is generated at the 10th frame, which spans from the 1st to the 10th frame. The second animation is generated at the 11th frame, which spans from the 2nd to the 11th frame. These two consecutively generated animations have 9 overlapping frames (from the 2nd to the 10th). They are too similar and using the similar examples to train Categorizers will significantly reduce the training efficiency. Moreover, selecting the appropriate examples from so many redundant ones is labor intensive. Therefore, users can choose to skip some frames when generating two consecutive behavior examples. In the above case, if users choose to skip 10 frames, the second animation will be generated at the 20th frame, which spans from the 11th to the 20th frame. These two consecutive animations have no overlapping frame. The workload of selecting appropriate examples for building training dataset is also reduced. However, the more frames to skip, the lower chance to get an animation that perfectly spans a complete behavior episode. In the above case, if a complete episode of behavior X spans from the 5th to the 15th frame, skipping 10 frames will result in two animations with only 5 frames of behavior X in each. To get an animation that perfectly spans the complete episode of behavior X, users need to choose to skip 5 frames so that the second animation spans from the 5th to the 15th frame. Therefore, deciding how many frames to skip is a judgment call to the users.

A practical recommendation to achieve a balance between getting the perfect animations and reducing the labor is to set this number as the half of the duration of a behavior episode (1/2 of the animation duration).

(Button) ‘Start to generate behavior examples’

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

Specify whether to include background in the generated animations. Users can choose ‘No’ if the background information is behavior irrelevant. Sometimes the background information might be critical for distinguishing the behaviors. For example, if users want to distinguish sitting on a wheel from sitting on the ground, including the background (the wheel or the ground) will help the behavior classification.

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

Specify whether to include body parts (such as the nose or limbs when they fall within the silhouette of the body) in the generated pattern images. Users can choose ‘Yes’ if the motion pattern of these body parts are critical to distinguish different behaviors. If choose ‘Yes’, users need to enter a ‘STD’, 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. The ‘motion pixels’ are the pixels whose values have significant changed during a behavior episode (the duration of an animation). If STD is set to 0, all the pixels whose values have any changes will be shown; if STD is large, only those pixels whose values have large changes will be shown. Users may try different values of STD to see which value achieves the desired pattern images.

- (Pop-up option) 'Start to generate examples?'

Just to confirm whether to start to generate the behavior examples.

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 by the behavior names. The trained Categorizers will be automatically added into the Categorizer list for the usage in either '***Test Categorizers***' or '***Analyze Behaviors***' functional unit.

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

This folder should contain all the sorted behavior examples. Users need to select and sort behavior examples into different subfolders under this folder according to their behavior categories. The names of subfolders should be the behavior names. For example, subfolder A should store all example pairs of behavior A.

(Button) 'Select a new folder to store all the prepared behavior examples'

This folder will store all the prepared behavior examples. 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 animations will be resized to the input frame size of Animation Analyzer and the pattern images will be resized to the input image size of Pattern Recognizer. Users may choose to downsize the frame / image size of the behavior examples before training to increase the processing speed in training. The targeted frame / image size should not be smaller than the targeted input size of the Categorizers to train. For example, if users want to train a Categorizer with Animation Analyzer of 32 x 32 input frame size and Pattern Recognizer of 48 x 48 input image size, the targeted size of resizing at this step should not be smaller than 48 x 48. If users choose to 'Resize the frames to 48' at this step, all the animations will be resized to 48 x 48 at this step and further downsized to 32 x 32 during training and all the pattern images will be resized to 48 x 48 and no further downsizing will be performed during training.

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

Users need to specify whether the animations used for training include the background or not.

(Button) ‘Start to prepare the training examples’

Press to start to prepare the training examples.

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

- (Pop-up option) 'Categorizer types'

There are two types of Categorizers for users to choose. One is the Categorizer with both Animation Analyzer and Pattern Recognizer; the other is the Categorizer with only Pattern Recognizer. The one with only Pattern Recognizer is much faster in training and behavior analysis but a little less accurate. Users may choose the one with both Animation Analyzer and Pattern Recognizer unless in the scenario when processing speed is more critical.

- (Pop-up option) 'Complexity level'

There are 7 complexity levels (1~7) of either Animation Analyzer or Pattern Recognizer for users to choose. 1 is the simplest and 7 is the most complex. 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. If the color of animals is behavior irrelevant (or the animals are just black and white), users may make the complexity level of Pattern Recognizer a little higher than that of Animation Analyzer since the colors in the pattern images are useful information, which indicate the temporal sequence of the behaviors.

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

Specify the input frame size of Animation Analyzer and the input image size of 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 (or the animals are just black and white) is behavior irrelevant.

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

Specify the number of frames input into the Categorizer, which should be equal to the duration of a behavior episode (the duration of the animations in the behavior examples).

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

This should be 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. For example, 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 manipulate the training examples and amplify them to benefit the training. If users want to know more details about these augmentation methods, they may read the *LabGym* paper for the detailed descriptions. Briefly, '*random rotation*' will randomly rotate the animals in both the animations and the pattern images; '*flipping*' will flip the animals in both the animations and the pattern images; '*random brightening*' will increase the brightness of the animal blobs in the animations to a random extent; '*random dimming*' will decrease the brightness the animal blobs in the animations to a random extent; '*random shearing*' will distort the animal blobs in both the animations and pattern images to a random degree; '*random rescaling*' will change the width / height ratio of the animal blobs in both the animations and pattern images to a random ratio; '*random deletion*' will randomly delete one or two frames in the animations.

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

If users do not know how to choose these augmentation methods, they may choose to use the default combination of the augmentation methods.

- (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.

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

This folder will store the training reports.

(Button) 'Start to train the Categorizer'

Press to start to train the Categorizer. Users need to give a name to the Categorizer to train.

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

This functional unit is used 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'

Select a Categorizer to test its accuracy.

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

Users first need to use '***Generate Behavior Examples***' functional unit to generate some behavior examples and sort them according to their behavior types into the subfolders of this folder. The names of its subfolders should be the behavior names.

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

This folder will store the testing reports.

(Button) 'Test the selected Categorizer'

Press to start to test the selected Categorizer.

(Button) 'Delete a Categorizer'

Select a Categorizer and delete it. Note that the deletion cannot be restored.

3.7. '*Analyze Behaviors*'

This functional unit is used to analyze behaviors in videos.

(Button) 'Select a Categorizer for behavior classification'

Users may choose a Categorizer for behavior classification in analysis. They can also choose not to do behavior classification in analysis. In the latter scenario, *LabGym* will only track the animals and calculate their motion parameters and body kinematics and users need to provide a time window for calculating them. If users would like to choose a Categorizer that is stored in a user-created folder, they can choose the option of '*Choose a new directory of the Categorizer*'.

- (Pop-up option) 'Uncertainty level'

This number in percentage determines the threshold for the Categorizer to output an 'NA' for behavioral classification. For example, at a certain frame, the Categorizer outputs the probabilities of behavior A, B, and C as 50%, 20%, and 30%, respectively. If the uncertain level is set to be 21, the behavior classification at this frame will be 'NA', as the difference between probability of the highest-likely behavior (A, 50%) and the second highest-likely behavior (C, 30%), which is 20%, is less than the uncertainty level 21%. This uncertainty level can be used to reduce the possible false positives in behavior classification since any ambiguous classification will be output as an 'NA'.

<u>(Button) 'Select the video(s) for behavior analysis'</u>

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?'

Specify whether to resize the frames of the videos. The resizing will keep the original "width / height" ratio. Downsizing the frames will significantly increase the processing speed and **is highly recommended**. 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, the size of an animal is approximately 1/4 to that of a frame. Suppose the original size of a video frame is

1000 x 500, and the input size of the Categorizer is 48 x 48. If users downsize the video frames to 500 x 250, the animal size is approximately 125 x 63 after downsizing, which 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 typically for videos involving light on and off (e.g., optogenetics) and the time point of sudden illumination changes for the first time will be automatically detected and used as the beginning time for behavior analysis.

'*Decode from filenames: _bt_*' can be used if multiple videos are selected for behavior analysis and the beginning time for each video is different. If choose this option, users need to add a code tag '_bt_' ('b' stands for 'beginning time' and 't' should be an integer or floating number) into the file name of each video to let *LabGym* decode the beginning time. For example, if the file name of a video is 'ABC-0102.avi' and the user wants the beginning time to be at the 12.35 second, the user can rename the video file to 'ABC-0102_b12.35_.avi' or 'ABC-0102_b12.35.avi' (the last '_' is unnecessary if it is at the end of the file name).

'*Enter a time point*' can be used if only one video is selected, or multiple videos are selected and the beginning time for each video is the same. In this scenario, users can manually enter a time point (should be an integer).

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

The number entered must be an integer. All the videos selected will use this duration for behavior analysis.

(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 for behavior analysis and the number of animals in each video is different. If choose this option, users need to add a code tag ‘_nn_’ (the first ‘n’ stands for ‘number of animals’ and the second ‘n’ should be an integer) into the file name of each video to let *LabGym* decode the animal number. For example, if the file name of a video is ‘ABC-0102.avi’ and the number of animals in this video is 8, the user can rename the video file to ‘ABC-0102_n8_.avi’ or ‘ABC-0102_n8.avi’ (the last ‘_’ is unnecessary if it is at the end of the file name).

'*Enter the number of animals*' can be used if only one video is selected, or if multiple videos are selected and the number of animals in each video is the same. In this scenario, users can manually enter a number (should be an integer).

- (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 1 second, its ID and the matrix linked with the ID for storing all information of this animal will be deregistered temporally. If users 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. In this scenario, if there is no deregistered ID-matrix left for re-initiation, the re-tracked animal will then be registered to a new ID-matrix. An animal that is lost track for over 25% 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 useful when the backgrounds of the videos are static and the illumination in the video is stable overtime. And if this is the case, this method is much faster and more accurate in detecting the animals. However, this method is not useful for social behaviors as it cannot distinguish individual animals when they have body contact. After selecting this method, users need to then specify the scenarios of their experiments: '*Animal brighter than background*', '*Animal darker than background*', or '*Hard to tell*'. Then a pop-up option '(Optional) load existing background?' will need to be specified. The animal detection and tracking in *LabGym* is based on background subtraction. *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 (to save time) if the background for this video has already been extracted and output. Note that if choose this option, the loaded backgrounds will be used for background subtraction for all the selected videos. After that, users will be asked whether the illumination in videos is unstable. If the illumination in the videos is very stable overtime, users can 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. If select a time window during which the animals always stay at one place, the animals will be considered as the background and the detection and tracking will fail. The time window should be as short as possible since longer time window takes longer time and more memory to

process. There are 3 options to specify this time window: (1) '*Use the entire duration of a video*'. This is generally not recommended if the processing speed is critical. Do not use this option in any following scenario unless the computer memory is larger than 64 GB: the video frame size is over 2000 x 2000 unless users downsize them (check 'Select the video(s) to generate behavior examples' to see how downsize the video frame); the video duration is over 5 minutes; the video fps is over 60. (2) '*Decode from filenames: _xst_ and _xet_*'. This can be used if multiple videos are selected for generating behavior examples and the time window for background extraction is different for each video. If choose this option, users need to add two code tags '_xst_ and _xet_' ('xs' stands for 'extraction start time', 'xe' stands for 'extraction end time', and 't' should be an integer) into the file name of each video to let *LabGym* decode the time window. For example, if the file name of a video is 'ABC-0102.avi' and the time window for background extraction is from the 25th second to the 47th second, the user can rename the video file to 'ABC-0102_xs25_xe47_.avi' or 'ABC-0102_xs25_xe47.avi' (the last '_' is unnecessary if it is at the end of the file name). (3) '*Enter two time points*'. This be used if only one video is selected, or if multiple videos are selected and the time window for background extraction for each video is the same. In this scenario, users can manually enter two time points (should be integers).

'*Use trained detectors*' is useful in any kind of videos or experimental settings. It is also useful for social behaviors. The only caveat of this method is slow. It's coming soon.

(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 14 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 *angle* is the movement direction (against to the animal body axis) of the animal during a behavior episode, which is the mean of all the included angle (θ) between animal body axis and the movement direction during the time window (t_w) for categorizing the behavior.
- ◊ 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.

Users may read the *LabGym* paper for detailed descriptions on how these measurements are calculated.

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

Specify whether to normalize the distances in calculating these quantitative measurements. If users choose 'No', all the distances will be output in pixels. The unit of all the distance related measurements will be 'pixel' or pixel related. For example, speed will be in the unit of pixels per second. If users 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), if the animals used in the experiments are of similar size. In fact, 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'

Press to start analyzing the behaviors.

3.8. '*Mine Analysis Results*'

This functional unit is used to help users to mine the analysis results. For example, tell users which behavior parameters show statistically significant differences among groups. It's coming soon.

ACKNOWLEDGEMENTS

This research was supported by National Institutes of Health (NIH) to B.Y. (R01NS104299 and R01EB028159). The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

1. Van Rossum, G., and Drake, F.L. (2000). Python reference manual (iUniverse Indiana).

2. Yujia Hu, Carrie R. Ferrario, Alexander D. Maitland, Rita B. Ionides, Anjesh Ghimire, Brendon Watson, Kenichi Iwasaki, Hope White, Yitao Xi, Jie Zhou, Bing Ye. LabGym: Quantification of user-defined animal behaviors using learning-based holistic assessment. Cell Reports Methods. Volume 3, Issue 3, 2023, 100415, ISSN 2667-2375, https://doi.org/10.1016/j.crmeth.2023.100415.