Internship Report

Name Ms. Chomchaba Chanakiat

Supervisor Prof. Dr. Poramate Manoonpong

Mr. Binggwong Leung

Period June-July, 2021

Introduction

Dung beetles can perform several versatile behaviors, one of which includes dung ball rolling. Dung ball rolling is a complex and unique gait because it provides locomotion and object transportation (dung ball) in the meantime. In the previous work (Leung et al., 2020), four underlying rules for leg coordination during ball rolling were proposed. They can be applied to build a simulation of a dung beetle-like robot. These rules may not be enough to make the model realistic. Therefore, the objective of this work is to further investigate the insight of leg coordination during ball rolling. But this time, DeepLabCut(DLC) was used as a tracking program. DeepLabCut is a software package for animal pose estimation. It is an efficient method for 3D markerless pose estimation based on transfer learning with deep neural networks.

Methods

DeepLabCut Process

DeepLabCut relies on supervised learning, there must be the use of labeled datasets to train algorithms that classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately, which occurs as part of the validation process. Therefore, the method is divided into four main processes as shown in Figure 1. The first process is creating input or labeled data. The second process is to train the neural network using the input data obtained from the previous. The third process is evaluating network performance, and finally, run inference on new videos to get results.

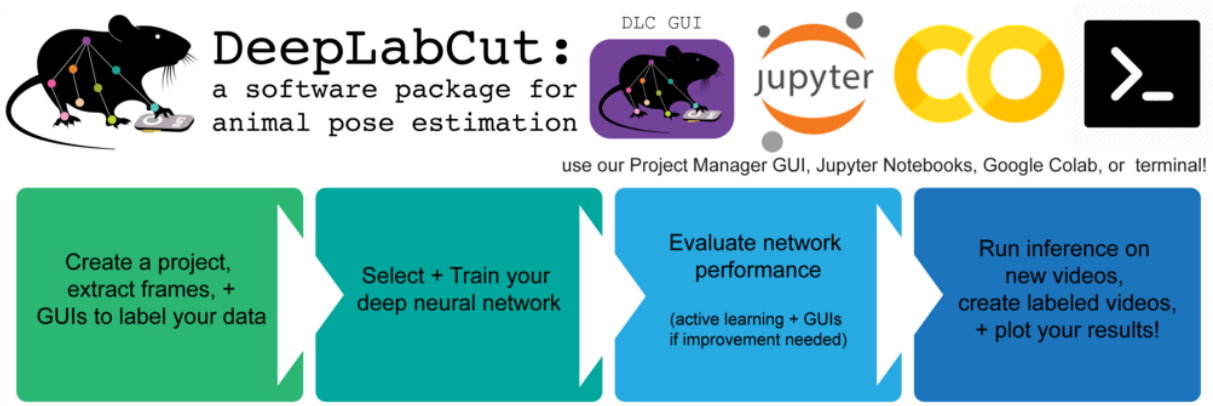


Figure 1: An overview of the pipeline and workflow for project management (DeepLabCut).

DeepLabCut Project Management

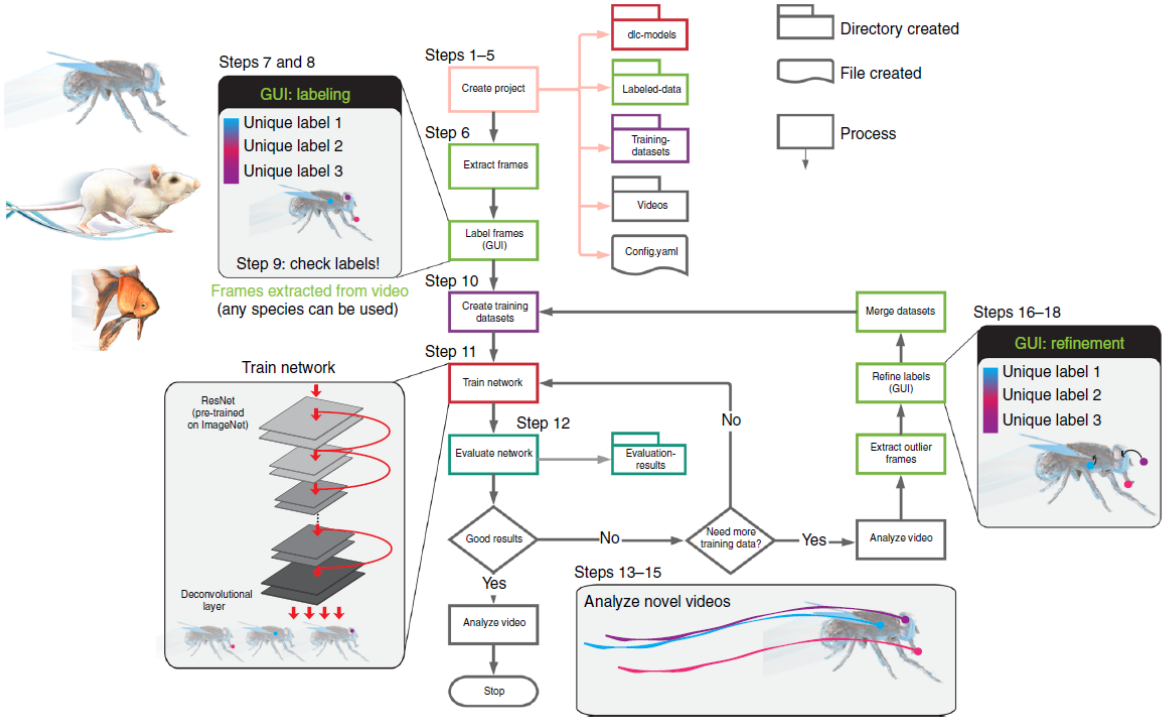


Figure 2: Pipeline and workflow for project management on DeepLabCut (Nath et al., 2019).

The diagram shows the workflow, as well as the directory and file structures. The workflow is color-coded to represent the locations of the output of each step or section. The main steps are opening a Python session, importing DeepLabCut, creating a project, selecting frames, labeling frames, and training a network. Once trained, this network can be used to apply labels to new

videos, or the network can be refined if needed. The process runs with interactive graphical user interfaces (GUIs) at several key steps (Nath et al., 2019). The special details for some steps are:

- *Creating a project*

After importing DeepLabCut, the first thing to do is creating a project. The project directory consists of subdirectories: dlc-models, labeled-data, training-datasets, and videos. All the outputs generated during the course of a project will be stored in one of these subdirectories. The project directory also contains the main configuration file called config.yaml. The config.yaml file contains many important parameters of the project, which can be adjusted there.

- *Labeling frames*

This stage is to label body parts of interest in the frames extracted from the previous step (extracting frames). Body parts can be added and edited in the config file.

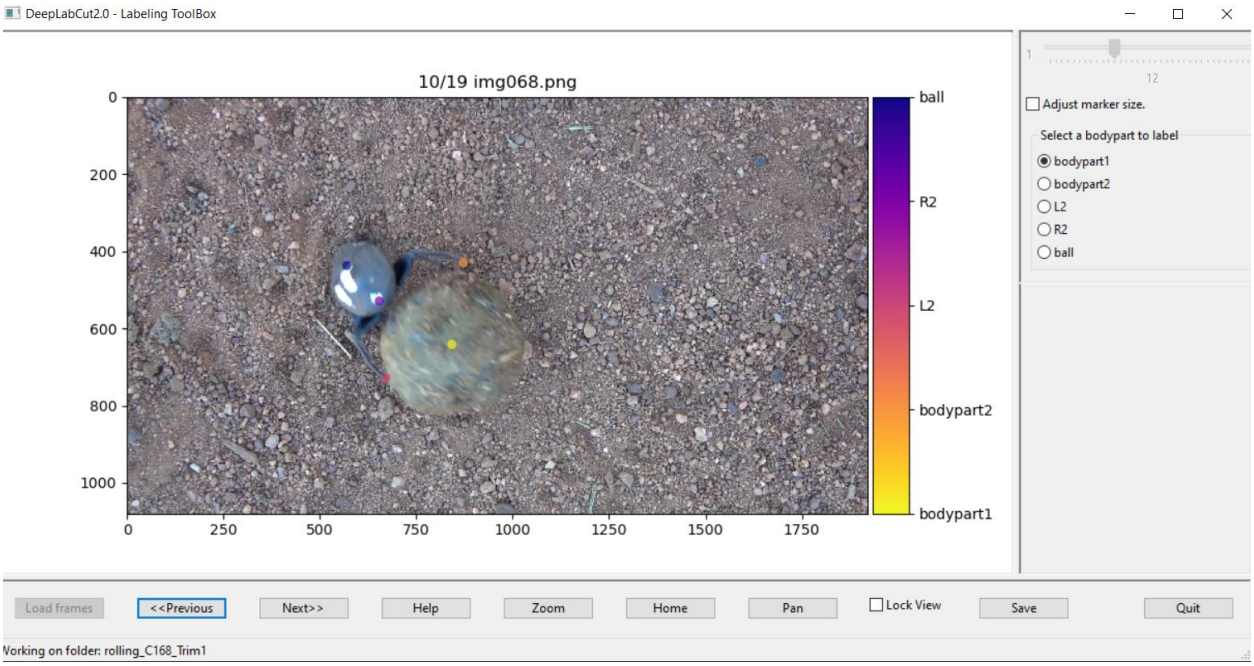


Figure 3: Labeling toolbox (GUI) used to label frames

The remaining steps: creating training dataset, training network, evaluating network, analyzing videos, and creating plots and labeled videos, can be performed in Google Colaboratory via notebook template provided by DeepLabCut. The code template is ready to use, only required uploading labeled data into Google Drive.

Results

Dung beetle videos performing three different tasks were analyzed: walking, rolling a light ball, and rolling a heavy ball, respectively.

Table 1: The table shows the video information and the number of extracted frames used as labeled data.

Video Type (Dung beetle's behavior)	Number of videos	Frame rate (frames/second)	Frame number (all videos)	% Frames used in labeled data	
				Before refining labels	After refining labels
Walking	1	50	1368	3.58	7.09
Rolling a light ball	4	30	318	25.16	-
Rolling a heavy ball	7	30	1337	5.24	-

Dung beetle walking video

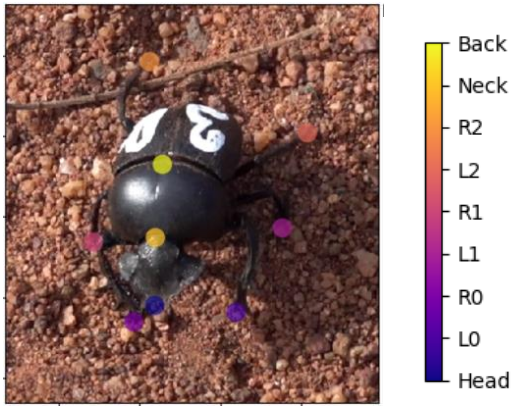


Figure 4: The example of labeled data for the walking video. The colored points represent the features to be measured. The features include head, L0(front left leg), R0(front right leg), L1(middle left leg), R1(middle right leg), L2(hind left leg), R2(hind right leg), neck, and back as shown in the figure.

In this type of video, walking, there is only one analyzed video. The video is 27 seconds long. The labeled data was created from 49 frames, which is 3.58% of the total frames in the video. Figure 4 shows the dung beetle's body parts used to train a network.

Table 2: Evaluation results of EfficientNet-B3 model training for the walking video.

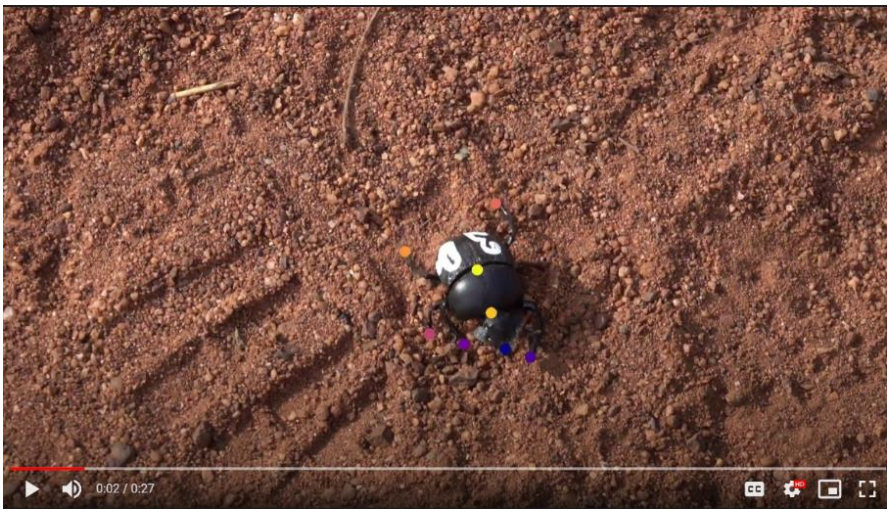
Trial	Training iterations	%Training dataset	Shuffle number	Train error(px)	Test error(px)	p-cutoff used	Train error with p-cutoff	Test error with p-cutoff
1	4000	95	1	85.91	112.8	0.6	9.57	8.59
2	9000	95	1	7.72	46.21	0.2	7.72	27.95
3	10000	95	1	7.55	58.59	0.2	7.54	18.48
4*	10000	95	1	8219.38	11705.46	0.2	900.14	971.44
5*	9000	95	1	968.03	949.86	0.2	-	-
6*	14000	95	1	857.6	831.04	0.2	902.51	888.57

The yellow bar indicates the trial in which the snapshot will be applied.

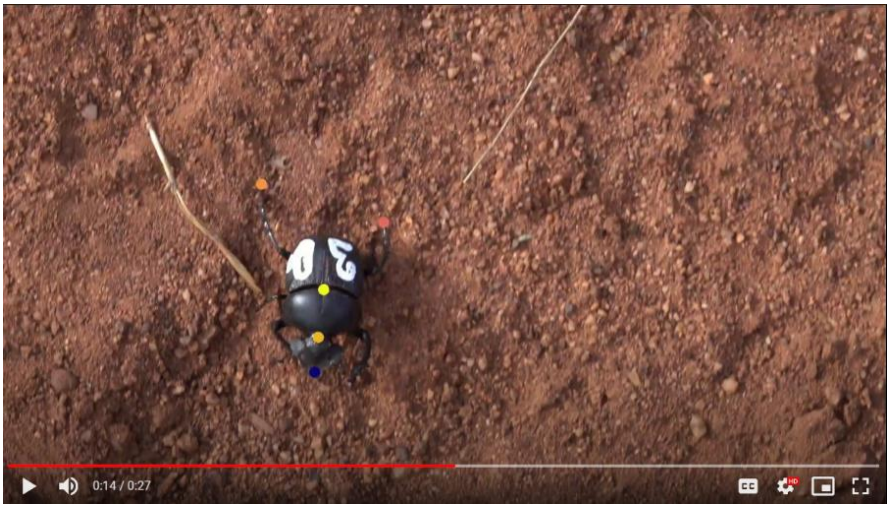
* indicates the results after extracting the outlier frames and refining labels.

There was a total of six trials of the training network with the EfficientNet-B3 model. Subsequent training continued the snapshot from the previous. With the increase in test errors between Trials 2 and 3, it seems that the network has reached a point where a more suitable variable cannot be found, possibly as a result of an inadequate training dataset. Therefore, I decide to extract the outlier frames and refine labels. Trial 4-6 used new labeled data but continue the neural network snapshot from Trial 3. However, the evaluation results were not better as they could be. All errors are significantly high, even proceeded more than 30,000 training iterations. So, the training was stopped at Trial 6 and the network from Trial 2 was used to analyze the video in the next step. Figures 5-8 are the results derived from analyzing the same walking video.

a.



b.



c.

Figure 5: Screenshot examples of the labeled video (walking video).

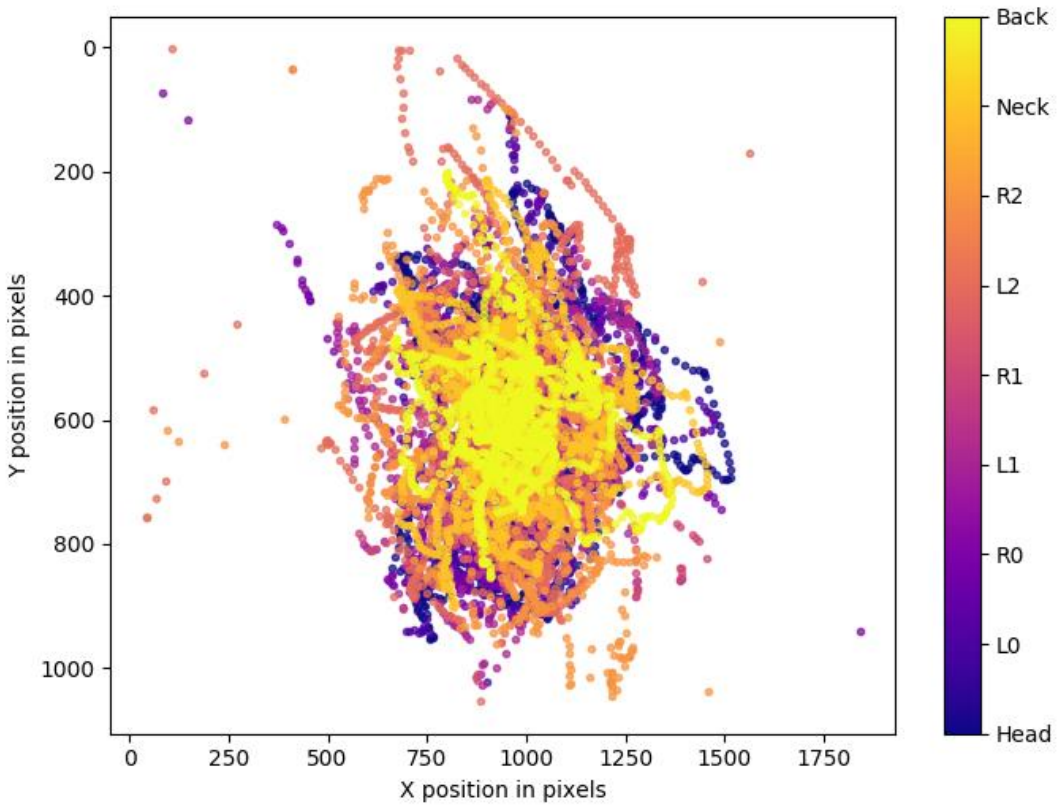


Figure 6: Plot of trajectory of each feature in the labeled video (walking video).

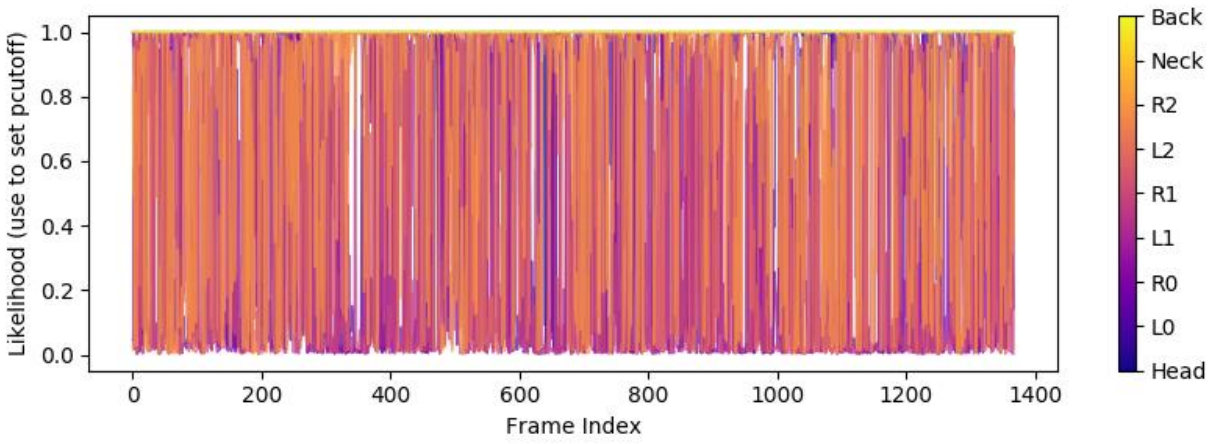


Figure 7: Likelihood of each feature versus frame index in the labeled video (walking video).

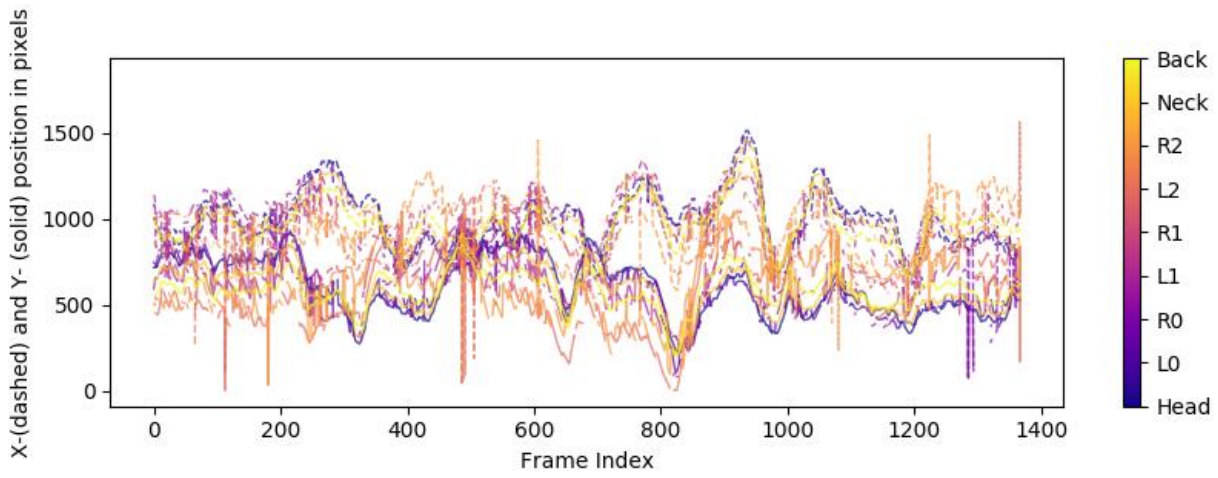


Figure 8: Positions of each feature versus frame index in the labeled video (walking video).

From Figure 8, some errors can be seen from the graph lines that are not continuous. Some points are abnormally high and low. The cause of the error may come from two factors. Firstly, in the video, the background changes all the time, not a uniform ground. Secondly, the video, with a length of 27 seconds, is too long. The labeled data is not enough to cover all variety of frames. Therefore, it is hard to adjust the weight for a fit model.

Dung beetle video rolling light ball

The second type of dung beetle video is the light ball rolling video. In the original video, there are some periods the dung beetle does not exist in the frame. To keep the video short and easy to train, all mentioned periods were cut off. As a result, there are four short videos used to train and analyze. Each video shows the ball rolling of dung beetle through the camera with the length of 3, 2, 2, 2 seconds. The percentage of labeled data increased from the previous training, with approximately 25.16% (80 out of 318 frames, 20 frames from each video). The following figure shows all considered features to analyze.

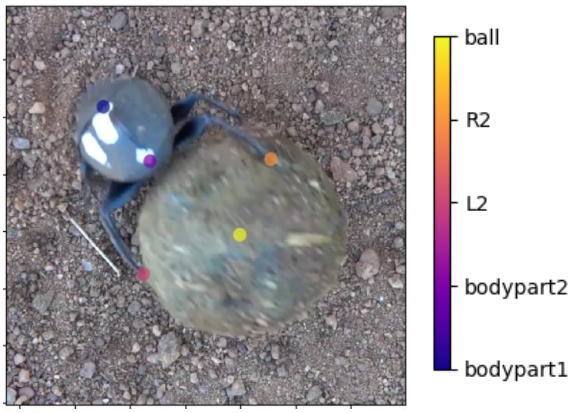


Figure 9: The example of labeled data for the light ball rolling video.
The colored points represent the features to be measured. The features include ball(dung ball), R2(hind right leg), L2(hind left leg), bodypart1, and bodypart2 as shown in the figure.

Table 3: Evaluation results of EfficientNet-B3 model training for the light ball rolling video.

Trial	Training iterations	%Training dataset	Shuffle number	Train error(px)	Test error(px)	p-cutoff used	Train error with p-cutoff	Test error with p-cutoff
1	10000	95	1	1120.05	1094.82	0.4	-	-

For network training, initially, the EfficientNet-B3 model was used for training. The evaluation results indicate very high errors even after ten thousand training iterations, leading to a change to the model ResNet-50.

Table 4: Evaluation results of ResNet-50 model training for a light ball rolling video.

Trial	Training iterations	%Training dataset	Shuffle number	Train error(px)	Test error(px)	p-cutoff used	Train error with p-cutoff	Test error with p-cutoff
1	52000	95	1	2.18	48.6	0.4	2.1	4.79
2	4000	95	1	1.87	21.96	0.4	1.79	4.47
3	10000	95	1	1.73	6.07	0.4	1.66	4.23
4	6000	95	1	1.67	7.42	0.4	1.59	4.18

The yellow bar indicates the trial in which the snapshot will be applied.

There was a total of four trials of the training network with the ResNet-50 model. The error value decreased gradually as the number of training increased. Because of the low values of the errors, this model seems to fit well with input labeled data. Additionally, ResNet-50 took less training time per iteration than EfficientNet-B3. Figures 10-13 are the results derived from analyzing one of the original videos using the snapshot from the last trial.

a.

b.

c.

Figure 10: Screenshot examples of the labeled video (the light ball rolling video).

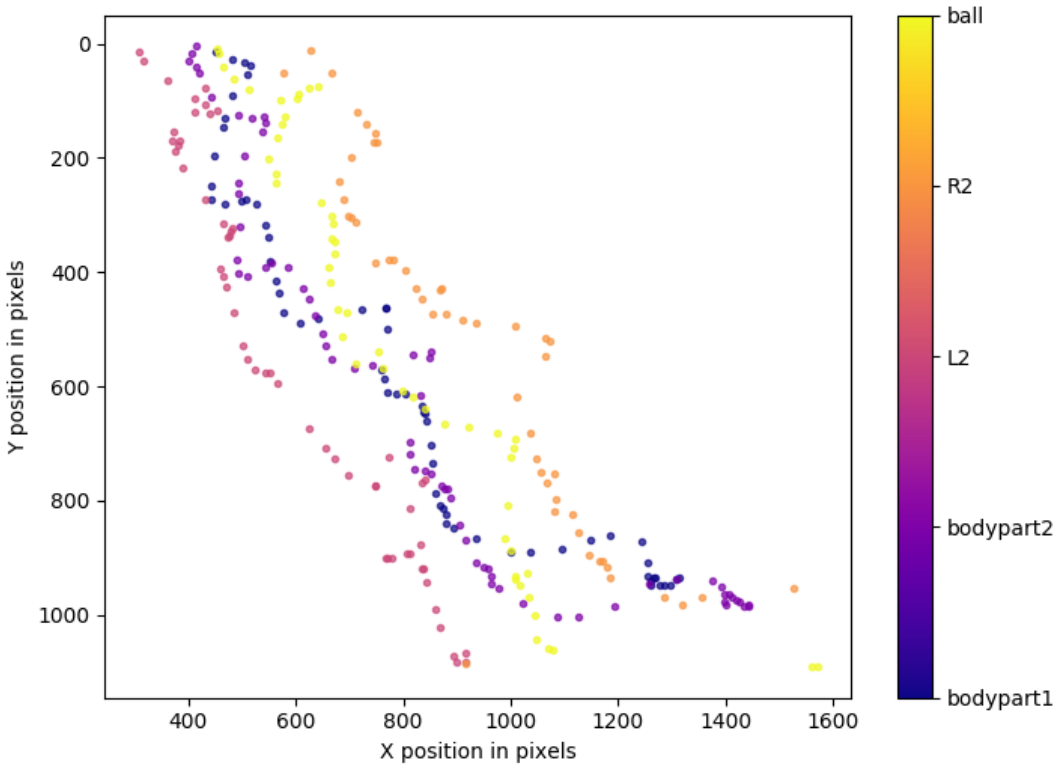


Figure 11: Plot of trajectory of each feature in the labeled video (the light ball rolling video).

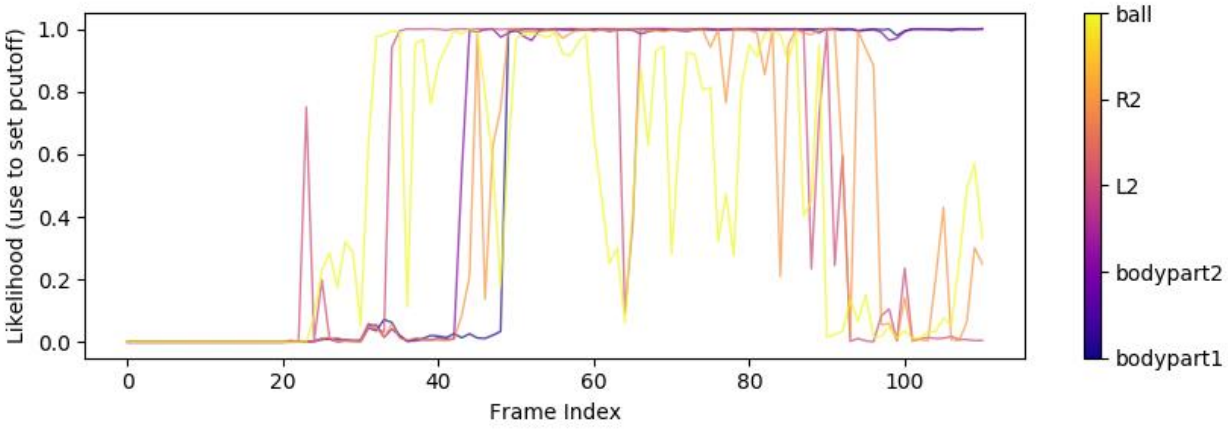


Figure 12: Likelihood of each feature versus frame index in the labeled video (the light ball rolling video).

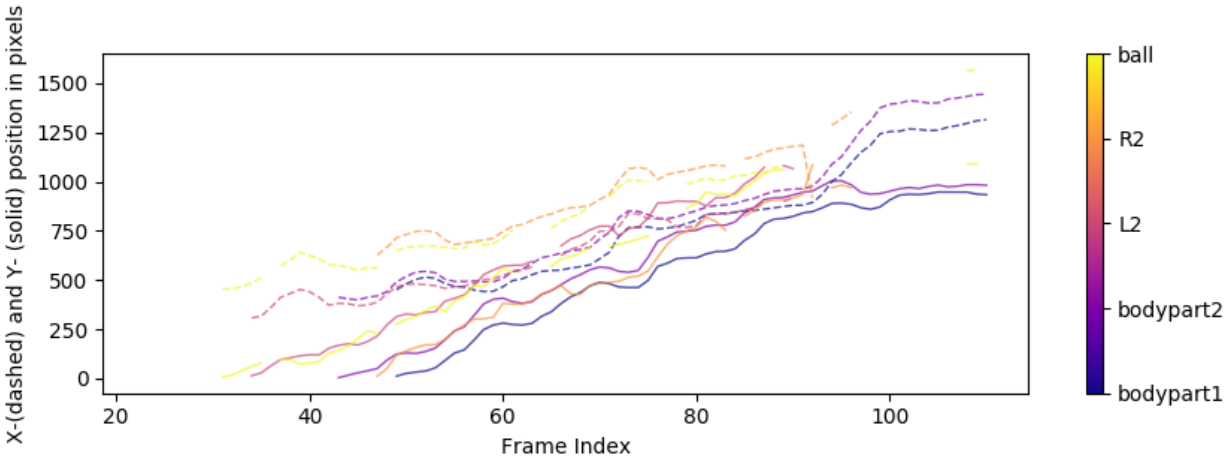


Figure 13: Positions of each feature versus frame index in the labeled video (the light ball rolling video).

Dung beetle video rolling heavy ball

Figure 14: The example of labeled data for the heavy ball rolling video.

The colored points represent the features to be measured. The features include ball(dung ball), B0(head), B1(middle body), B2(bottom), L0(front left leg), R0(front right leg), L1(middle left leg), R1(middle right leg), L2(hind left leg), R2(hind right leg), JL0(front left joint), JR0(front right joint), JL1(middle left joint), JR1(middle right joint), JL2(hind left joint), JR2(hind right joint) as shown in the figure.

This video is similar to the previous video in that it has the same background and camera angle, and the only difference is that the dung beetle moves more slowly due to the heavier ball. As before, the video parts that did not show the dung beetle's movement were cut off, resulting in a total of 7 sub videos. Ten frames were extracted from each video to create labeled data, approximately 5.24% of all videos (70 out of 1337 frames). Figure 14 shows all considered features to analyze.

Table 5: Evaluation results of EfficientNet-B3 model training for the heavy ball rolling video.

Trial	Training iterations	%Training dataset	Shuffle number	Train error(px)	Test error(px)	p-cutoff used	Train error with p-cutoff	Test error with p-cutoff
1	6000	95	1	84.01	96.71	0.6	13.64	8.76
2	16000	95	1	1026.16	890.08	0.6	-	-

Initially, the EfficientNet-B3 model was used for training. As the same, after many training iterations, the evaluation results still indicate very high errors. Therefore, in the new network training, the model was changed to ResNet-50.

Table 6: Evaluation results of ResNet-50 model training for the heavy ball rolling video.

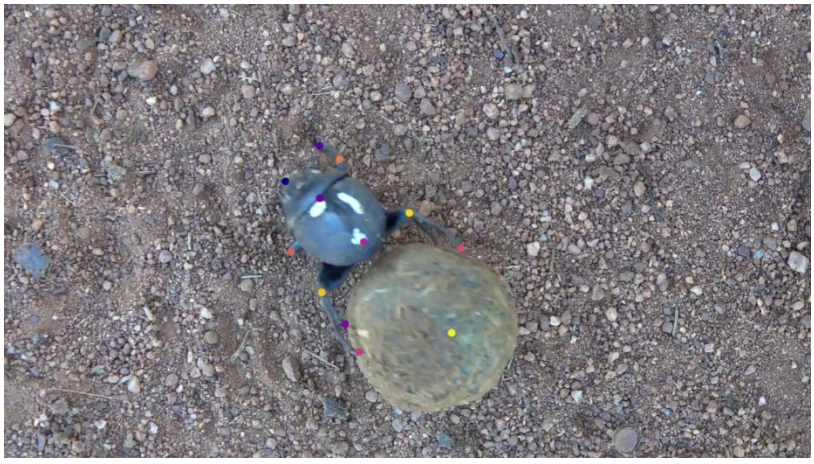
Trial	Training iterations	%Training dataset	Shuffle number	Train error(px)	Test error(px)	p-cutoff used	Train error with p-cutoff	Test error with p-cutoff
1	90,000	95	1	2.81	8.62	0.6	2.2	8.92
2	100,000	95	1	1.44	8.7	0.6	1.44	8.83
3	100,000	95	1	1.17	8.46	0.6	1.17	8.46

The yellow bar indicates the trial in which the snapshot will be applied.

There was a total of three trials of the training network with the ResNet-50 model. In this training, there were almost three hundred thousand iterations, resulting in small values of errors. Figures 15-18 are the results derived from analyzing one of the original videos using the snapshot from the last trial.

a.

b.



c.

Figure 15: Screenshot examples of the labeled video (the heavy ball rolling video).

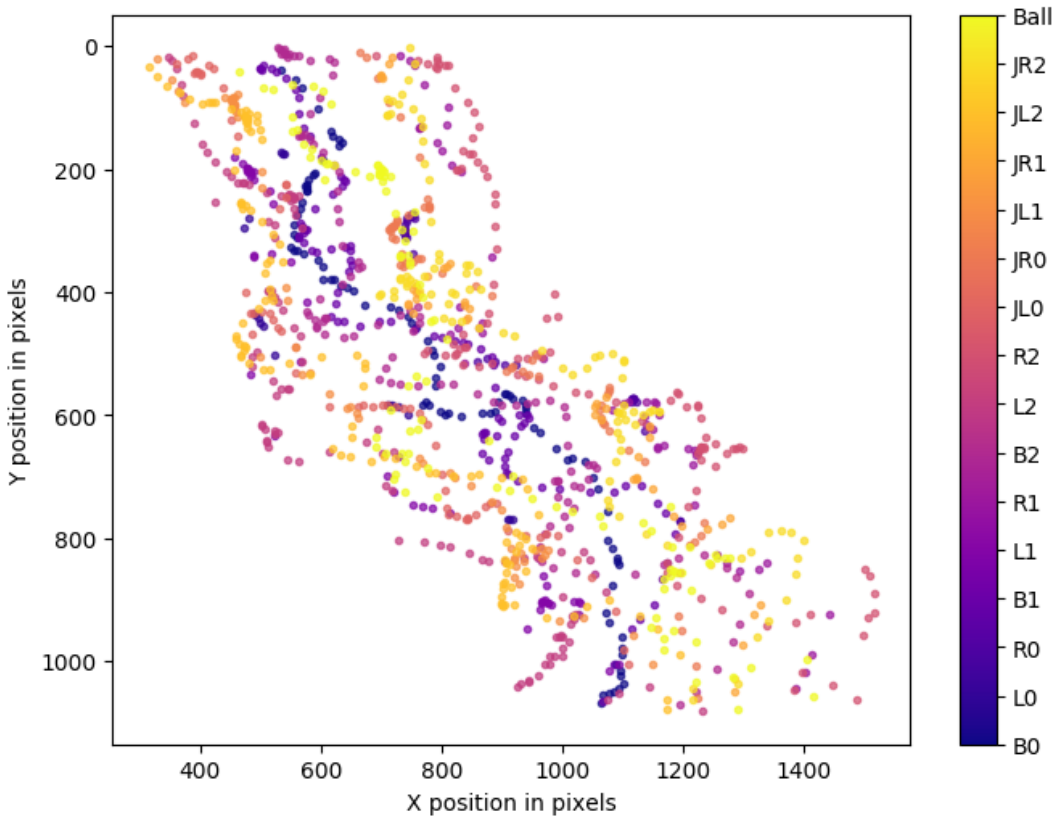


Figure 16: Plot of trajectory of each feature in the labeled video (the heavy ball rolling video).

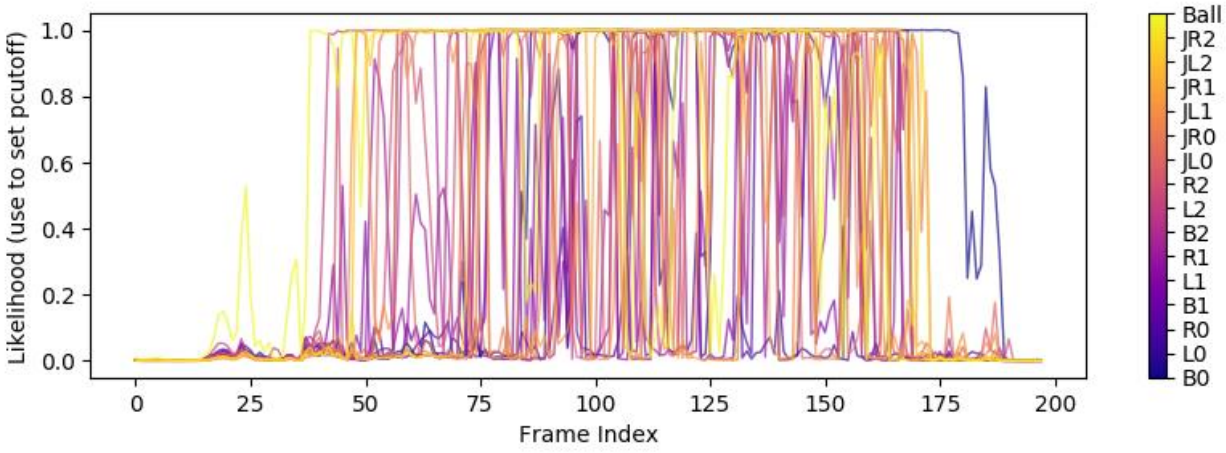


Figure 18: Likelihood of each feature versus frame index in the labeled video (the heavy ball rolling video).

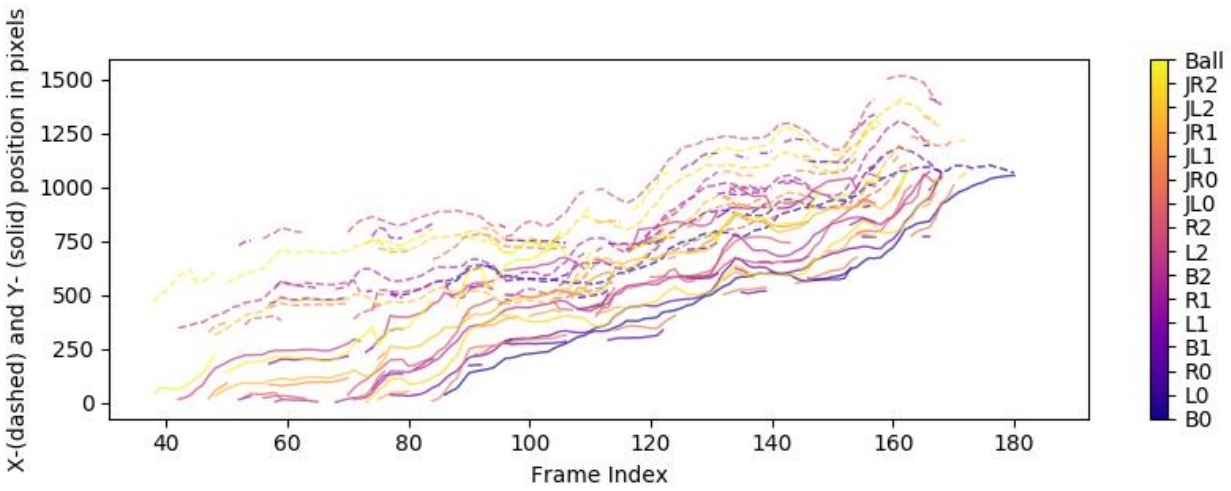


Figure 17: Positions of each feature versus frame index in the labeled video (the heavy ball rolling video).

Discussion

From the evaluation results, the errors of the walking video do not seem to be able to be reduced, while the errors of both light and heavy ball rolling videos can still be reduced further. Additionally, the errors of walking video were significantly higher than that of the others. Therefore, the training of both ball rolling videos is noticeably more accurate than that of the walking video. Here are some reasons why walking video is complicated to analyze:

1. No single background: The walking video has a ground that changes with time as the camera pans to follow the movements of the dung beetle.
2. Free motion: In the walk video, the dung beetle moved in all directions. This makes it complicated to distinguish between left and right legs.
3. Shadow: There was a shadow of a dung beetle all the time, and distinguishing a blackleg in the shadows is complicated.

And all three of which are not present in the ball rolling videos.

Moreover, from the difference between errors using different models in training of both ball rolling videos, it may conclude that the ResNet model is more suitable than the EfficientNet model for this dung beetle video analysis.

From Figure 16, it is found that the trajectory is swaying. By assuming that tilting affects walking patterns, one interesting for further study is the relation between the tilt of the insect body and the stretch of the forelegs. Maybe try plotting a graph of these two variables.

References

DeepLabCut. (n.d.). *DeepLabCut/DeepLabCut: Official implementation OF DEEPLABCUT: MARKERLESS pose estimation of USER-DEFINED features with deep learning for all ANIMALS INCL. HUMANS*. GitHub. https://github.com/DeepLabCut/DeepLabCut.

Nath, T., Mathis, A., Chen, A.C. *et al.* Using DeepLabCut for 3D markerless pose estimation across species and behaviors. *Nat Protoc* **14,** 2152–2176 (2019). https://doi.org/10.1038/s41596-019-0176-0

Leung, B., Bijma, N., Baird, E. *et al.* Rules for the Leg Coordination of Dung Beetle Ball Rolling Behaviour. *Sci Rep* **10,** 9278 (2020). https://doi.org/10.1038/s41598-020-66248-7