Monterey Bay Aquarium Research Institute

Machine Learning for Motion Capture of Deep Sea Animals

Claire Lin, University of California, Irvine

Mentors: Kakani Katija, Joost Daniels, Paul Roberts

Summer 2022

Keywords: Machine Learning, Computer Vision, DeepLabCut

ABSTRACT

There is a lot to learn when it comes to understanding underwater movement of marine animals. The starting point for understanding marine animal movement is motion capture, or in this case, landmark tracking. In the past, digital labels have been placed one by one on points of interest by a human. This process is tedious and time consuming. Machine learning offers a method for collecting motion data with much less human effort. Using DeepLabCut, an open source machine learning software, the practicality of using machine learning for landmark tracking on marine animals is explored. Tests were done with DeepLabCut using varying parameters and datasets to determine the best way to use DeepLabCut with Tomopteris video data.

INTRODUCTION

TOMOPTERIS MOVEMENT

The Tomopteris worm is a midwater polychaete which uses metachronal movement to swim through the water column. In addition to its body undulation, it uses its appendages called parapodia to create thrust in the direction of movement. The coordination of body wave and parapodia thrusts maximizes the stroke length and therefore increases total thrust parallel to movement (Daniels et al. 2021). The points on the Tomopteris we are interested in are the tips of the parapodia, base of the parapodia, head, tailbase, and antennae as a result of their unique swimming style. A total of 52 points were tracked using a Matlab application, DLTdv5 (Hedrick 2008). Eight pairs of more central parapodia are tracked (shown below). That same data is used here where we tested another mode of motion capture.

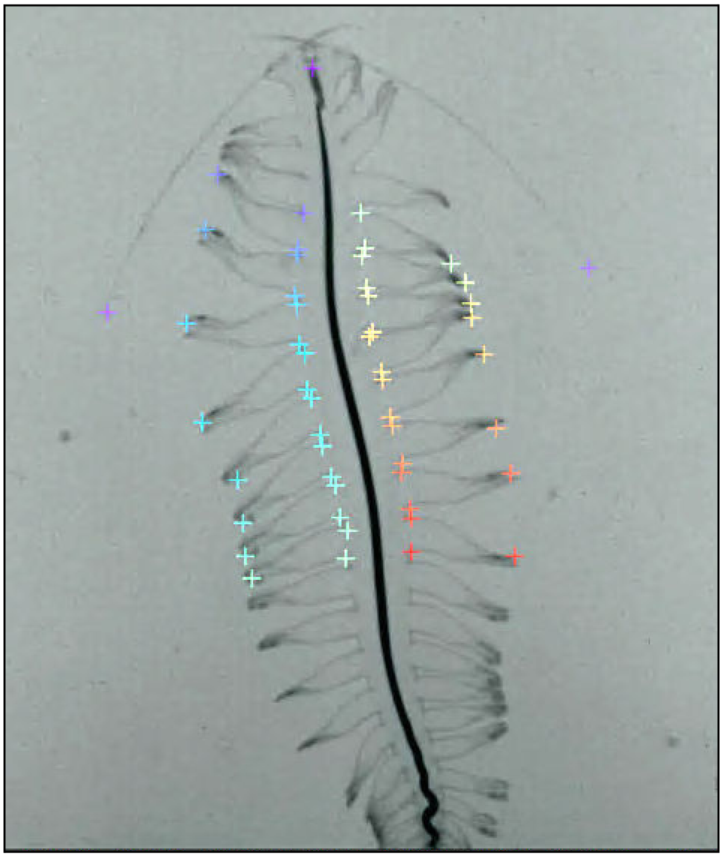

Figure 1. Correct label placement on Tomopteris

HISTORICALLY USED METHODS FOR MOTION CAPTURE

There are other approaches for collecting landmark location data such as manually digitizing position data using something like Loggerpro for blacktip shark biomechanics (Porter et al. 2020). Loggerpro was used to track 4 points of interest on shark video data. This involved frame by frame labeling by a human. Another motion capture example was mentioned before, the Matlab DLTdv5 application which uses pixel appearance and previous point location data to predict landmark position. Since this method relies on information specific to the video like pixel value and location, it cannot generalize for any Tomopteris video. Lastly, a more direct approach is creating physical landmarks by attaching reflectors for a computer to track while video data is being collected. This last approach has been done on animals such as bearded dragons (Frohnwieser et al. 2016), however it is very impractical for marine animal applications.

DEEPLABCUT

DeepLabCut is described to be used to create labels on novel video data. DeepLabCut is a python-based machine learning software that specializes in featureless landmark tracking on animals (Nath et al. 2018). With the implementation of a deep neural network, limited human labeling is required to gain the ability to have the computer track landmark points, without the use of reflectors and without the dependence on pixel value or previous location. The model is trained using labeled data to recognize and track features on an animal. With this labeled data, the model is trained over a specified number of iterations through a pre-trained network. The type and amount of data used for training was varied over different trials using the ResNet50 network to determine the best approach for using DeepLabCut on novel Tomopteris data.

MATERIALS AND METHODS

DATASET

Training data consisted of labeled frames from the Bioinspiration Lab, with size of training datasets ranging from 30 labeled frames to around 8,800 labeled frames with up to 52 landmarks in each. Two methods for using DeepLabCut were explored: “novel video tests", and “training video tests”. The former included labeled frames from different videos than the one pose estimation was applied to. The latter test consisted of some labeled frames being used as training data, and the rest of the frames being used as test frames. For training video tests, the DeepLabCut ‘kmeans’ algorithm was used to select frames that contained varying animal position and location. Other parameters that were considered were number of training iterations which were decreased as training loss values plateaued, and number of videos from which training data was taken from. Meanwhile, training sessions were timed to keep track of how the changing parameters affected the model training performance. Spreadsheet and detailed log for trials ran can be found at these links: Daily DLC Updates DLC Trials

PYTHON AND OTHER SOFTWARE

Python scripts were written to configure label data to input into DeepLabCut. Libraries such as csv, OpenCV, PiL, Math, Numpy, and MatplotLib were used to format training data .csv files as well as to create graphs of the resulting inference-labeled points. ImageJ was also used for increasing brightness and contrast further. Scripts written can be found at this link: DLC_Graphing&Configuration.ipynb

RESULTS

DEEPLABCUT TESTS WITHOUT PREVIOUSLY LABELED DATA

The preliminary trials using DeepLabCut with Tomopteris data involved following the tutorials provided by DeepLabCut creators. The basic workflow of these tutorials used 20 labeled frames from one video as training data, and running inference on the remaining frames of the same video. In these tests, the DeepLabCut GUI was used to label between 4 and 9 points of interest. These tests on the training videos worked fairly well with points on the correct spots throughout the video with some noise. The next step in the tutorial is running inference on a novel video. Landmark tracking on the novel video did not work nearly as well as with the training video, with many of the points placed in completely different parts of the frame. Novel videos chosen were of the same Tomopteris as in the training video. By this point, the ‘correct’ label locations were only given on those 20 frames used for training. So no metric was used to measure performance of pose estimation for the entire video.

DEEPLABCUT TESTS WITH PREVIOUSLY LABELED DATA

Using the ‘convertcsv2h5’ function in DeepLabCut, previously labeled data was used to train a model to track up to 52 points of interest on a single Tomopteris. For these training datasets, all frames of all training videos were used as labeled training data and DeepLabCut was used to run inference on a novel video. The range of video sets used for training included many videos with different individuals from the novel video as well as only including videos of the same individual as in the novel video. Using only one individual for training and inference also means there was much less training data than when multiple videos of different individuals were used. Despite smaller amounts of training data, the landmark inference results were actually much better when using the same individual versus different ones. With many fully labeled videos of different individuals, the inference of a novel video had placed points all along the perimeter of the frame on the tank rather than on the Tomopteris at all. With a smaller number of labeled videos with all of the same individual, the points were actually placed on the animal. They were either not in the correct spot, or ‘sliding’ down the length of the Tomopteris throughout the video. Since at this point, previously labeled data was being used, results could be compared with human-level accuracy for the entirety of the inference videos (figure 2). DeepLabCut also calculates a pixel error using training data and test data - both of these come from the training dataset. This value essentially tests model performance on a portion of the training frames - in this case a 95-5% split was used for training and test data.

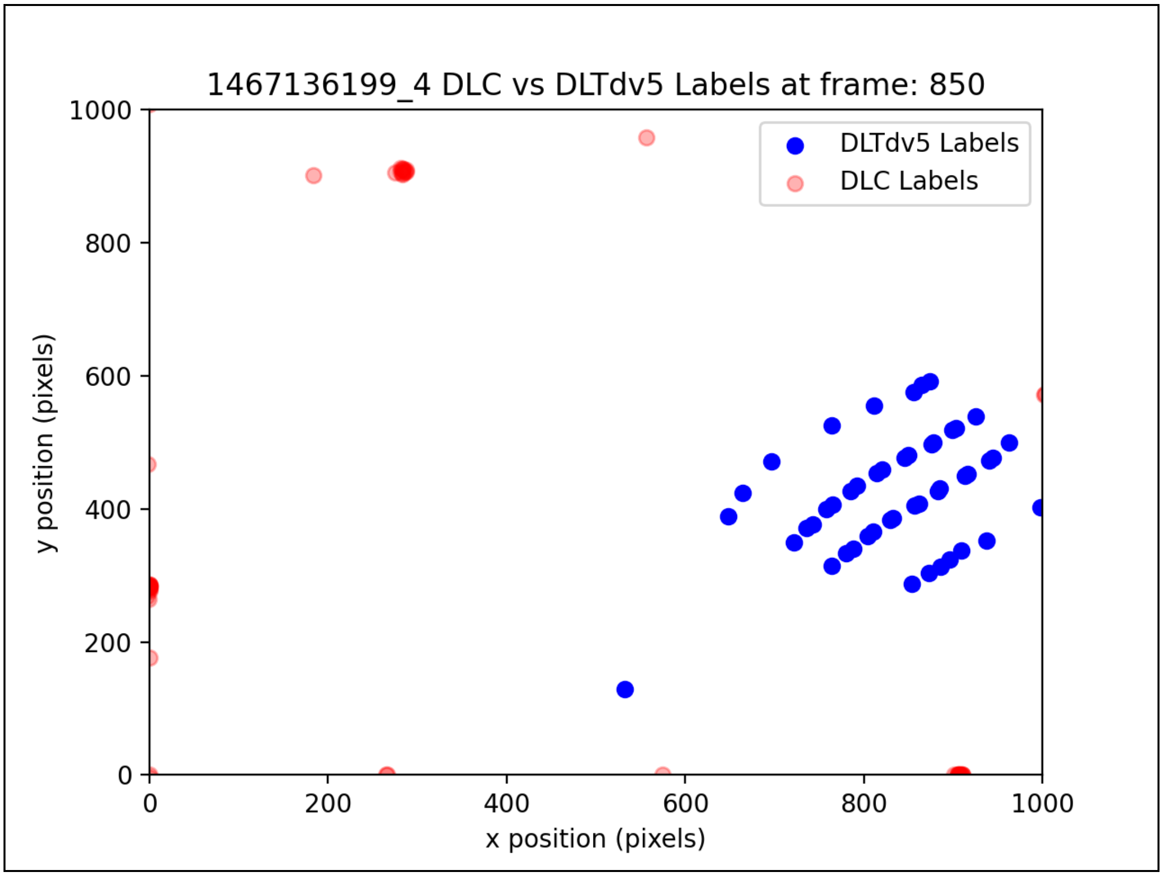

Figure 2. Label locations produced by DeepLabCut (red) and DLTdv5 (blue) in analysis of a novel video given 8,791 labeled frames for training from 9 different videos

To further evaluate model performance the root mean square error (RMSE) was calculated using the distance between pixels for each point tracked. In the equation below, x and y represent the x and y pixel difference between DeepLabCut labels and human-accuracy labels, and n represents the number of inference frames. Values were left out of calculations if DLTdv5 labels weren't available for that point or frame.

$$\sqrt{\frac{(\Sigma\sqrt{x^2+y^2})^2}{n}} \tag{1}$$

IMAGE PROCESSING TESTS

Additional tests involving adjusting brightness and contrast were done to determine whether DeepLabCut could differentiate the animal from the background at all, due to the transparency of the animal. ImageJ and PiL in Python were used to adjust image contrast for training data. The results did not noticeably improve. In a study using multi-animal DeepLabCut, the shoulders, ears, and tailbase of pigs were tracked. These points were not distinct from the rest of the animals colorwise, however performance was quite good with pixel errors ranging from 5.78 to 10.01 (Farahnakian et al. 2021). These results along with our contrast test suggest contrast of features with respect to the rest of the animal or background do not necessarily affect model performance and feature recognition. Part of the ‘creating training dataset’ step includes an image augmentation option, which alters the appearance of training frames for a more thoroughly trained model. In the end, it was decided that using image processing as a step before using DeepLabCut on Tomopteris data was not effective with pixel errors around 50 pixels and RMSE values of up to above 40,000.

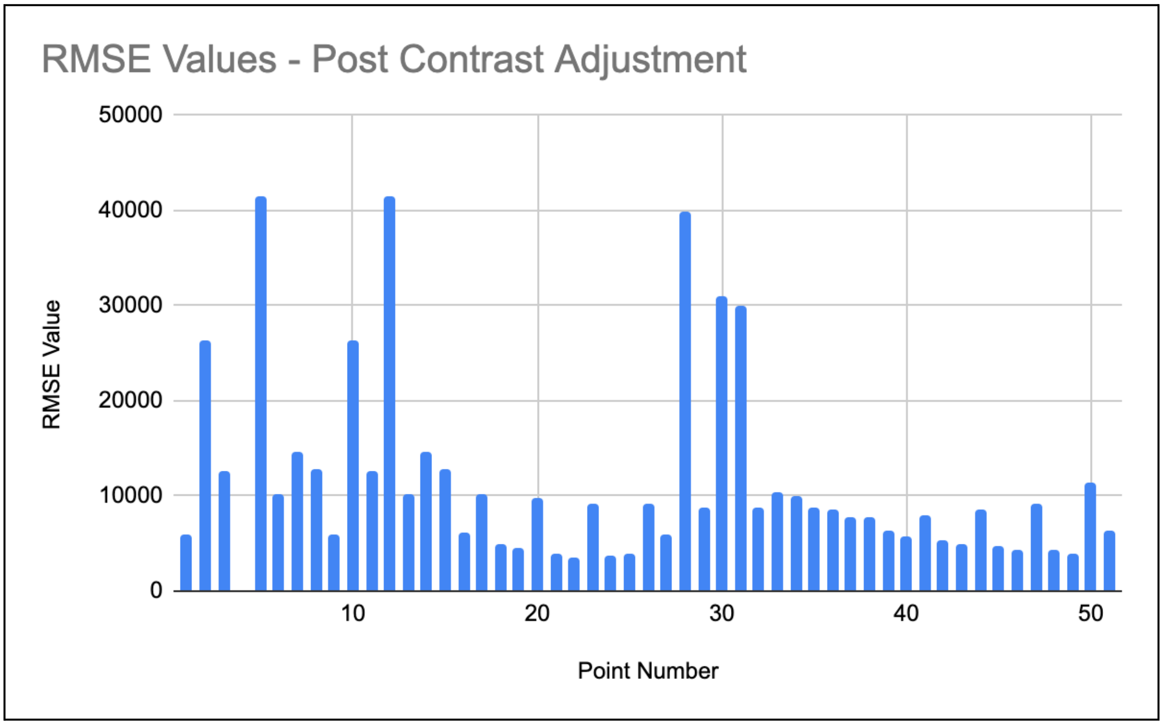

Figure 3. RMSE values for analysis of a novel video given 8,791 labeled frames for training from 9 different videos with increased contrast and brightness tracking 52 points

Train error(px)	Test error(px)
52.59	53.13

Figure 4. Evaluation results from contrast test

ANALYZING TRAINING VIDEOS

Another experiment done included only one individual in the training and testing data, so that there was no difference in size or appearance of the individual. When using this approach to analyze novel videos, results were still very inaccurate. However, when using this approach to analyze the same video like in the tutorial, pose estimation results were much closer to the human labels. This method of using DeepLabCut as a tool to fill in labels for a single video has been used in other studies on jumping spiders (Brandt et al. 2021), and centipedes (Diaz et al. 2022) - both animals which have greater than four similar looking appendages. With improved results using this method, the number of points were gradually increased again. Increasing the number of points to track from 4 to 12 decreased model performance. Different videos were used for each of these tests. To determine whether the number of points or the specific video was affecting model performance, the same video as used for 12 points was used in the same exact test with the only difference being that 6 points were used - the head, tailbase, antennae, and one pair of parapodia. The RMSE values between the 6 point and 12 point test are compared below in Figure 13.

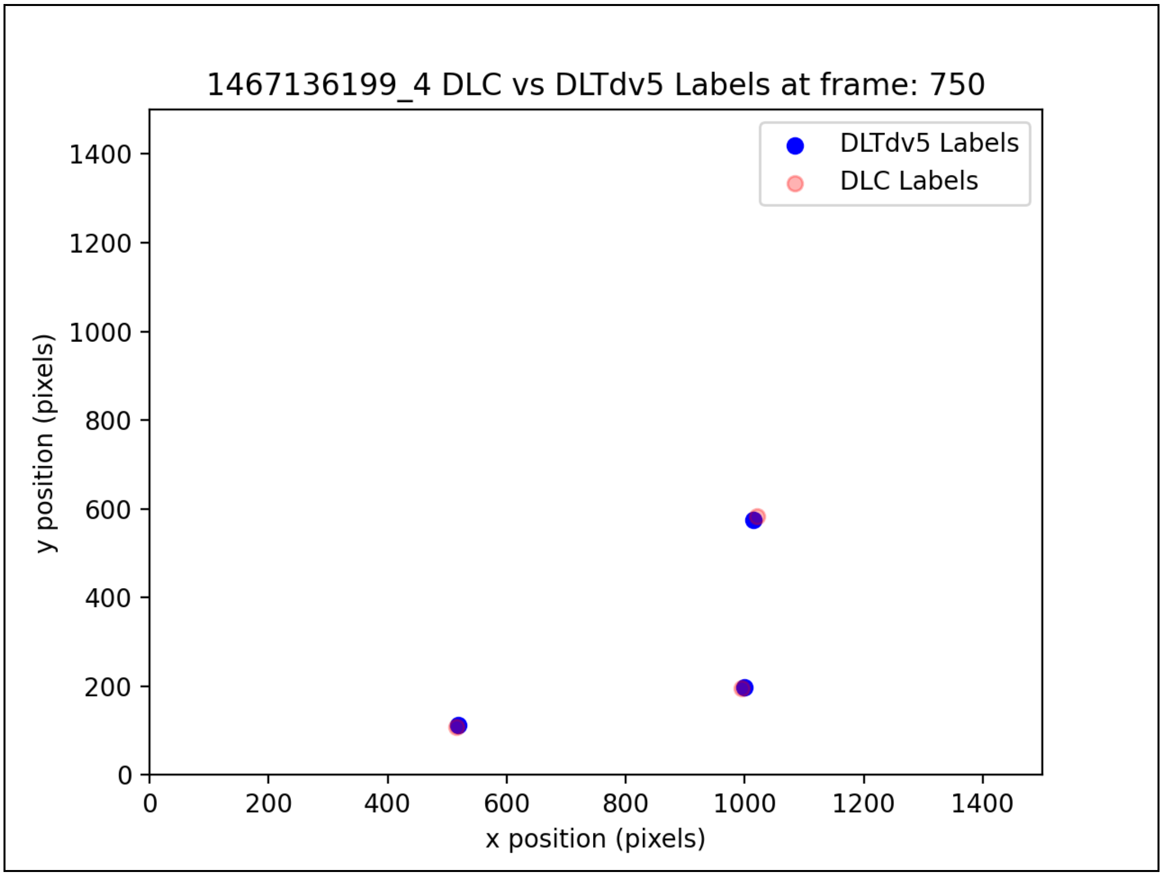

Figure 5. Label locations produced by DeepLabCut (red) and DLTdv5 (blue) in analysis of a training video given 40 labeled frames for training

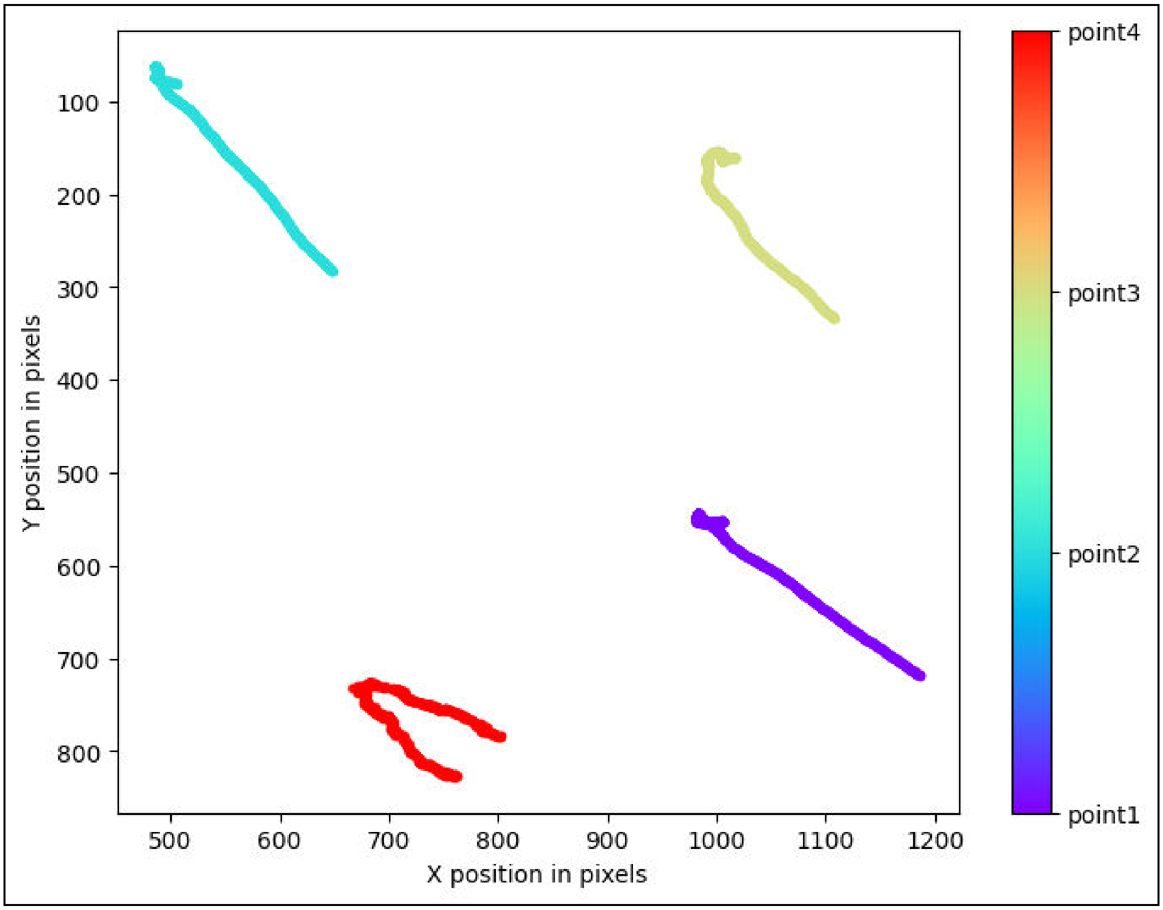

Figure 6. Trajectory plot of position estimation produced by Deeplabcut for full video

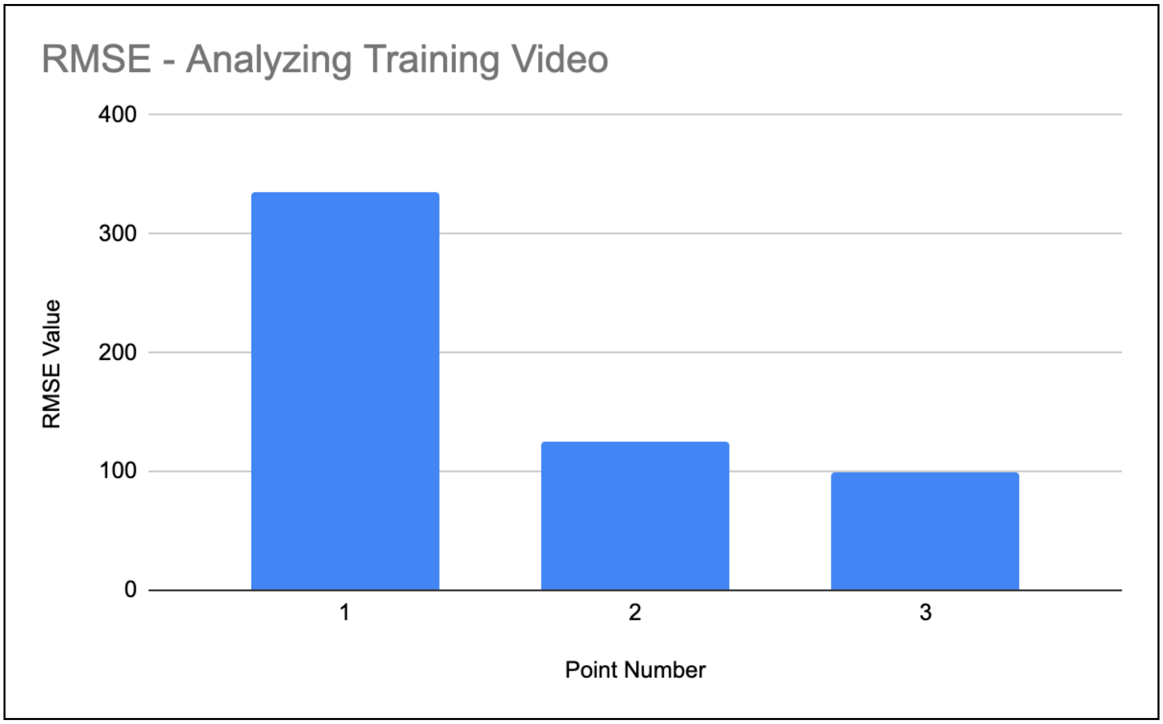

Figure 7. RMSE values for analysis of a training video given 40 labeled frames for training

Train error(px)	Test error(px)
2.43	3.28

Figure 8. Evaluation results from initial analysis of a training video

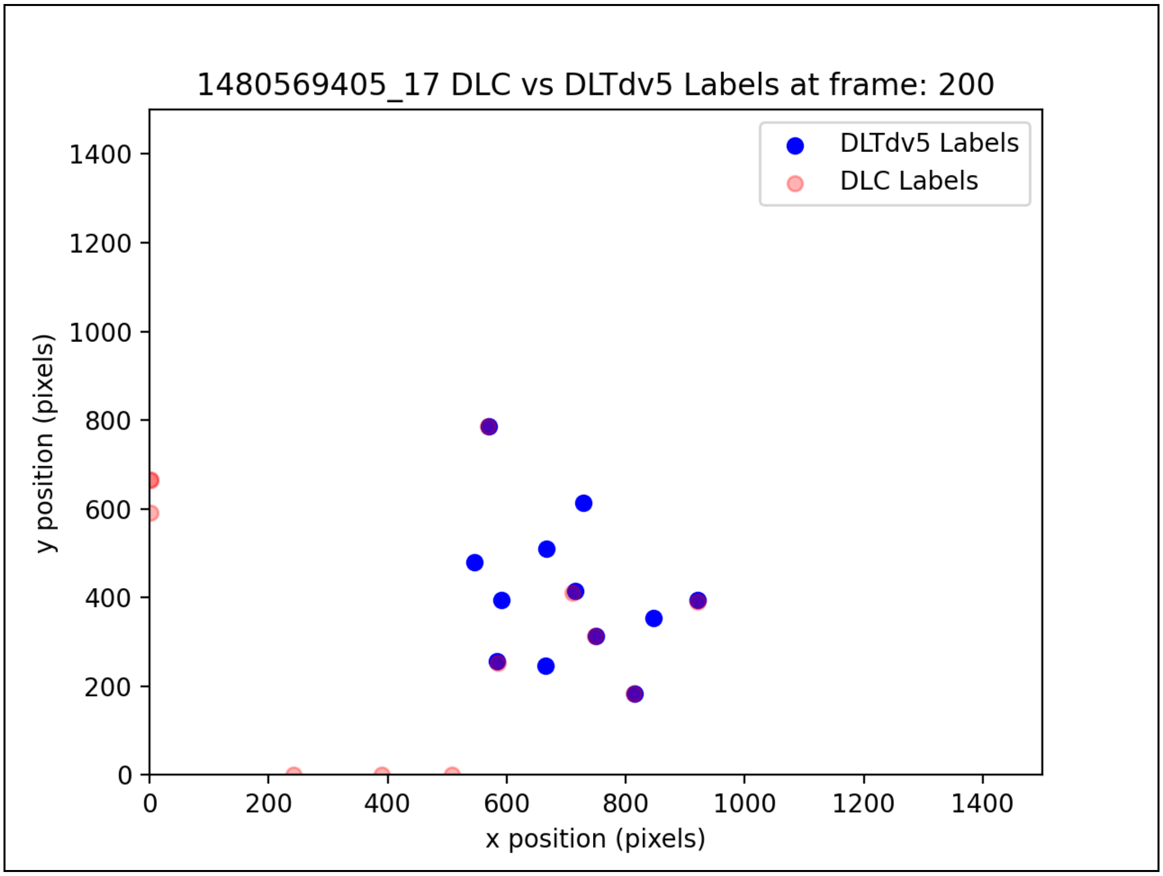

Figure 9. Label locations produced by DeepLabCut (red) and DLTdv5 (blue) in analysis of a training video given 33 labeled frames

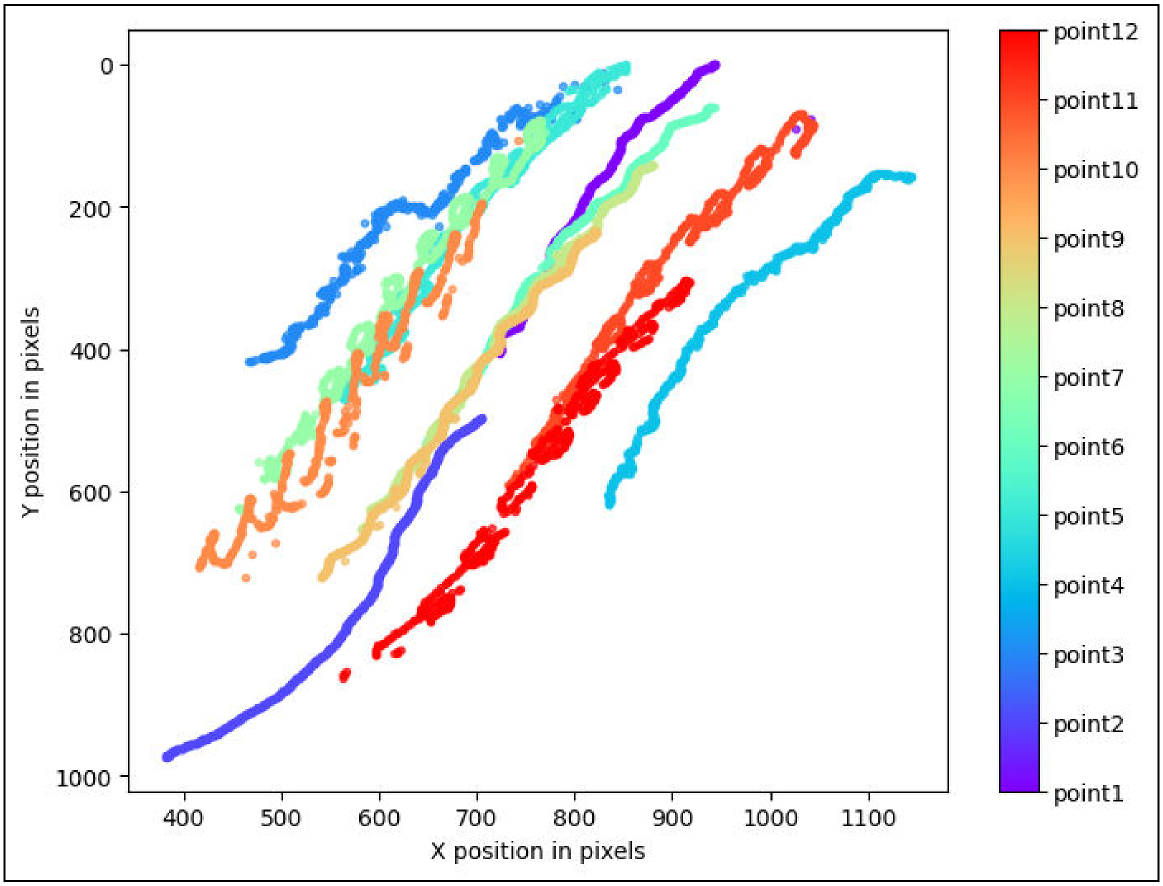

Figure 10. Trajectory plot of position estimation produced by DeepLabCut for full video

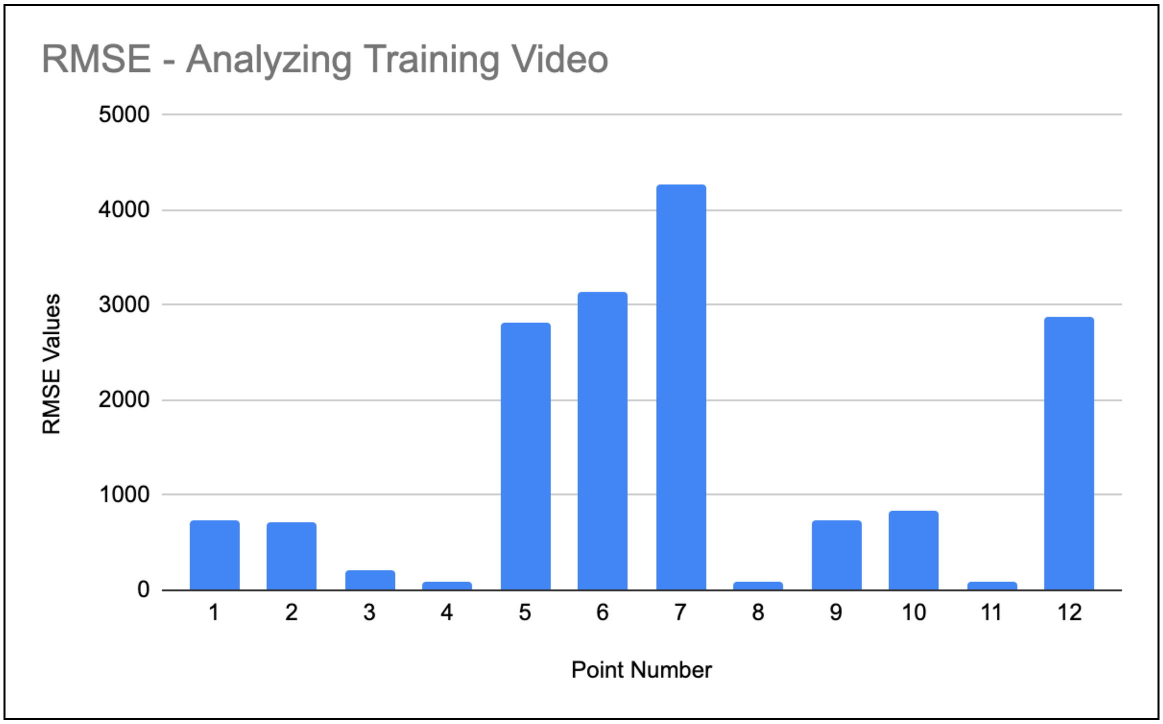

Figure 11. RMSE values for analysis of a novel video given 33 labeled frames for training

Train error(px)	Test error(px)
3.03	12.23

Figure 12. Evaluation results from analysis of training video

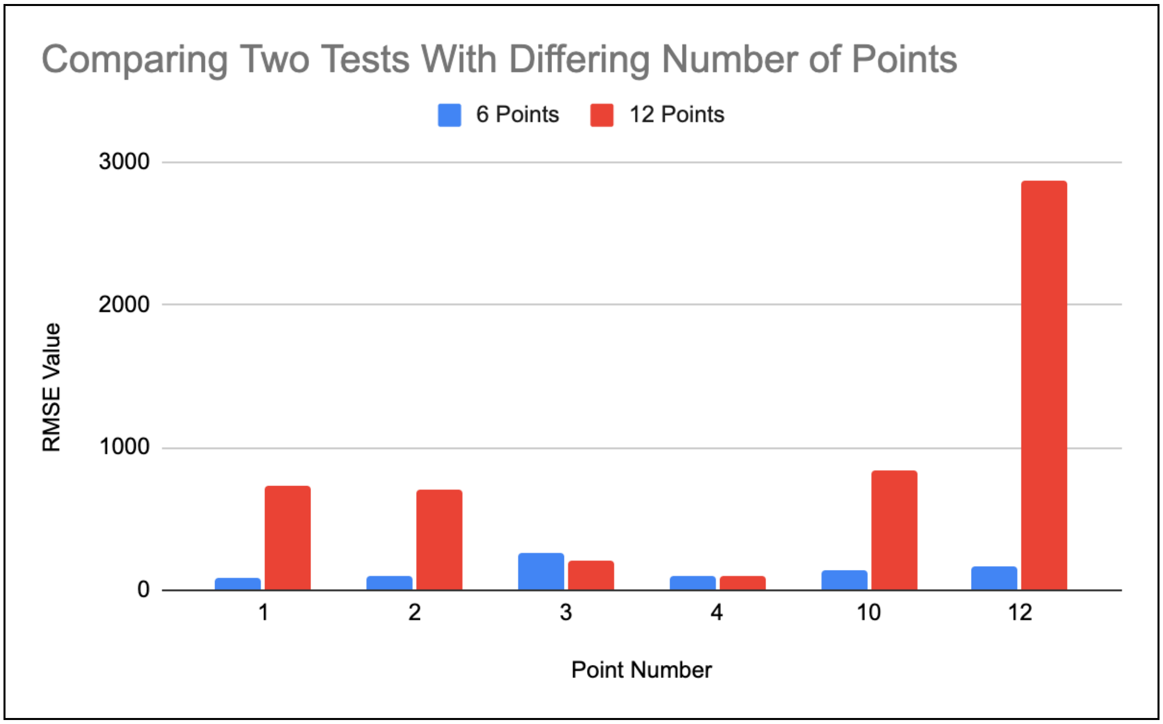

Figure 13. RMSE value comparisons between 6 and 12 point tests on training videos using 33 training frames.

The RMSE values for the 6 point test range from about 89 to 173 whereas for the 12 point test, they range from about 99 to 2870. The latter has much greater error values showing that DeepLabCut performs much better when fewer points are being tracked. This is also shown in the pixel error values that are shown below.

Test	Train error(px)	Test error(px)
12 Point	3.03	12.23
6 Point	1.86	7.46

Figure 14. Evaluation results from analysis of training video for 12 and 6 point tests given 33 frames of training data from the same video

DISCUSSION

When used for more unique animals such as a Tomopteris, DeepLabCut is a useful tool for finding landmark locations on a video when given a small percentage of training data from the same video. It is not as effective for inferring landmark locations on completely novel videos from the training dataset. In these cases, it tended to either put points on another object in frame, or the points would switch between all of the landmarks throughout the video. Though a large amount of training data (8,800 labeled frames) was provided in these novel video tests, perhaps there wasn't enough variety (only up to 9 different videos) in terms of background, perspective, and individual appearance for the model to be able to generalize for any Tomopteris worm. Guillermo Hidalgo Gadea's DeepLabCut tutorial used 14 videos to generalize for another clock. A clock is a fairly simple object compared to a Tomopteris, which can have up to 39 pairs of identical looking appendages. So if it took 14 separate cases to be able to correctly identify the three hands of a clock in a novel video, it could be that lacking a variety in training data was the real issue in these experiments.

Though in this case, DeepLabCut was not able to analyze novel Tomopteris videos, it did expedite the labeling process for a single video and reduced the number of human hours required to generate label data. The model had some trouble finding the points with higher accuracy as the number of points to track was increased. This method will be useful going forward for research of more newly discovered animal movement, where not much video data yet exists. The novel video approach may work for other deep sea animals with fewer appendages so the 'sliding' issue will not be as prominent.

CONCLUSIONS/RECOMMENDATIONS

With limited video data, DeepLabCut could not match human level accuracy for landmark tracking of novel Tomopteris data. We did find that DeepLabCut still worked to infer point locations on familiar data. This process is similar to that of the Matlab DLTdv5 method for motion capture in that some information for a video is given and the program labels the remaining frames of the video. The difference between the two is that DeepLabCut predictions are not based on previous point location. It is instead based on what the neural network is trained to identify. This should be more useful with more erratic movements, or where interpolation would fail. One could try expanding the training dataset on a simpler animal to attempt the novel video approach for future work.

ACKNOWLEDGEMENTS

I would like to thank my mentors, Drs. Kakani Katija, Paul Roberts, and Joost Daniels for guiding me through the research process and supporting me greatly this summer as well as for providing me with the resources and data required to complete this project. I would also like to thank Drs. Eric Orenstein and Denis Klimov for their additional support and advice on project work and the engineering industry. Another thanks to Dr. George Matsumoto, Megan Bassett, and Lyndsey Claassen for making this opportunity possible and putting in so much effort to make it a full, safe experience despite COVID-19 complications. Lastly, the MBARI Summer Internship Program is generously supported through a gift from the Dean and Helen Witter Family Fund and the Rentschler Family Fund in memory of former MBARI board member Frank Roberts (1920-2019) and by the David and Lucile Packard Foundation. Additional funding is provided by the Maxwell/Hanrahan Foundation.

References:

Daniels, J, Aoki, N., Havassy, J., **Katija, K.**, Osborn, K.J., (2021). Metachronal swimming with flexible legs: A kinematics analysis of the midwater Polychaete Tomopteris. *Integrative and Comparative Biology*, **61**: 1658-1673. https://doi.org/10.1093/icb/icab059

Kelimar Diaz, Eva Erickson, Baxi Chong, Daniel Soto, Daniel I. Goldman. Active and passive mechanics for rough terrain traversal in centipedes bioRxiv 2022.06.17.496557; doi: https://doi.org/10.1101/2022.06.17.496557
Baxi Chong et al 2022. A general locomotion control framework for multi-legged locomotors Bioinspir. Biomim. 17 046015

Tanmay Nath, Alexander Mathis, An Chi Chen, Amir Patel, Matthias Bethge, Mackenzie Weygandt Mathis. Using DeepLabCut for 3D markerless pose estimation across species and behaviors bioRxiv 476531; doi: https://doi.org/10.1101/476531

Brandt, E.E., Sasiharan, Y., Elias, D.O. *et al.* Jump takeoff in a small jumping spider. *J Comp Physiol A* 207, 153–164 (2021). https://doi.org/10.1007/s00359-021-01473-7

Wei Zhan, Yafeng Zou, Zhangzhang He, Zhiliang Zhang, "Key Points Tracking and Grooming Behavior Recognition of *Bactrocera minax* (Diptera: Trypetidae) via DeepLabCut", *Mathematical Problems in Engineering*, vol. 2021, Article ID 1392362, 15 pages, 2021. https://doi.org/10.1155/2021/1392362

F. Farahnakian, J. Heikkonen and S. Björkman, "Multi-pig Pose Estimation Using DeepLabCut," *2021 11th International Conference on Intelligent Control and Information Processing (ICICIP)*, 2021, pp. 143-148, doi: 10.1109/ICICIP53388.2021.9642168.

Raman, Srinivasan, Rytis Maskeliūnas, and Robertas Damaševičius. 2022. "Markerless Dog Pose Recognition in the Wild Using ResNet Deep Learning Model" *Computers* 11, no. 1: 2. https://doi.org/10.3390/computers11010002

Ryan D. Hunt, Ryan C. Ashbaugh, Mark Reimers, Lalita Udpa, Gabriela Saldana De Jimenez, Michael Moore, Assaf A. Gilad, Galit Pelled. Swimming direction of the glass catfish is responsive to magnetic stimulation bioRxiv 2020.08.13.250035; doi: https://doi.org/10.1101/2020.08.13.250035

R. Labuguen, D. K. Bardeloza, S. B. Negrete, J. Matsumoto, K. Inoue and T. Shibata, "Primate Markerless Pose Estimation and Movement Analysis Using DeepLabCut," *2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, 2019, pp. 297-300, doi: 10.1109/ICIEV.2019.8858533.

R. Labuguen, D. K. Bardeloza, S. B. Negrete, J. Matsumoto, K. Inoue and T. Shibata, "Primate Markerless Pose Estimation and Movement Analysis Using DeepLabCut," *2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, 2019, pp. 297-300, doi: 10.1109/ICIEV.2019.8858533.

Pereira, T.D., Tabris, N., Matsliah, A. *et al.* SLEAP: A deep learning system for multi-animal pose tracking. *Nat Methods* 19, 486–495 (2022). https://doi.org/10.1038/s41592-022-01426-1

Porter, Marianne E., Braden T. Ruddy, and Stephen M. Kajiura. 2020. "Volitional Swimming Kinematics of Blacktip Sharks, *Carcharhinus limbatus*, in the Wild"

Drones 4, no. 4: 78. https://doi.org/10.3390/drones4040078

Frohnwieser, A., Willmott, A.P., Murray, J.C., Pike, T.W., Wilkinson, A. (2016). Using Marker-Based Motion Capture to Develop a Head Bobbing Robotic Lizard. In: Tuci, E., Giagkos, A., Wilson, M., Hallam, J. (eds) From Animals to Animats 14. SAB 2016. Lecture Notes in Computer Science(), vol 9825. Springer, Cham. https://doi.org/10.1007/978-3-319-43488-9_2

Hedrick TL. 2008. Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. Bioinspirat Biomimet 3:034001.

Gadea, Guillermo Hidalgo. "Training Your First Deeplabcut Model – a Step by Step Example." *Guillermo Hidalgo Gadea*, 16 Mar. 2021, https://guillermohidalgogadea.com/openlabnotebook/training-your-first-dlc-mode-/.