5-2023

Poultry Pose Estimation with DeepLabCut

Chiyou Vang

Poultry Pose Estimation with DeepLabCut

An Undergraduate Honors College Thesis

in the

Department of Computer Science and Computer Engineering

College of Engineering

University of Arkansas

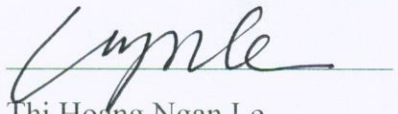
Fayetteville, AR

by

Chiyou Vang

This Thesis is approved. **Poultry Pose Estimation with DeepLabCut**

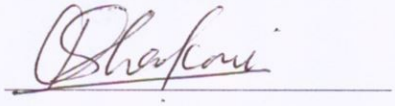
Thesis Advisor:



Thi Hoang Ngan Le

Thesis Committee:

Susan Gauch



Ukash Nakarmi

Table of Contents

Abstract

This honors thesis dives into the realm of deep learning-based pose estimation research and investigates the potential of DeepLabCut (Lauer, et al., 2021) in accurately and efficiently estimating the pose of poultry. With accurate pose estimation being a crucial aspect in understanding the behavior and movement of animals, this thesis aims to contribute to the development of more effective methods for pose estimation, especially for poultry.

To comprehensively evaluate the performance of DeepLabCut, two different types of chickens were tested in this thesis: a model toy chicken and actual live chickens. Videos were recorded for both types, and key points were manually labeled for selected frames. This labeling process served as the foundation for the creation of a DeepLabCut model, which was trained on the dataset and then evaluated for its performance on a separate validation dataset.

The result of this thesis showcases the good capabilities of DeepLabCut in accurately and efficiently estimating the pose of poultry when provided with sufficient data. The trained models were able to accurately predict the pose of the chickens in the videos when the models had sufficient training data on the poses. However, due to insufficient data, at certain poses with insufficient training data, the model that was created for the live chickens was overall not great performance-wise.

To enhance the model's performance, a selective approach was employed to increase the size of the training dataset by focusing on troublesome body parts and poses that had insufficient training data. In the end, the model's overall performance demonstrated improvement, although the increase was modest.

1. Introduction

Pose estimation is useful in animal welfare because it provides important monitoring of the health and welfare of the animals. In poultry this is particularly important, due to the high mortality of poultry where chickens in a flock can have deaths every week. This thesis explores the use of DeepLabCut (Lauer, et al., 2021), an open-source package for deep learning-based pose estimation, to estimate the pose of poultry accurately and efficiently. The aim is to contribute to the development of better methods for pose estimation, especially for poultry. The methodology involved data collection, labeling, model training, and evaluation. The study used both a model toy chicken and actual live chickens. The results show that DeepLabCut can accurately and efficiently estimate the pose of poultry when provided with sufficient data. However, the model's performance was not great overall for certain poses due to insufficient training data for live chickens.

1.1 DeepLabCut

DeepLabCut (Lauer et al., 2021) is a deep convolutional neural network (CNN) that is designed for pose estimation and tracking of body parts in images and videos. The architecture of DeepLabCut combines two parts: pretrained ResNets (He, Zhang, Ren, & Sun, 2015) and deconvolutional layers.

ResNets are a type of CNN that uses residual connections to allow for deeper networks without suffering from the vanishing gradient problem. In DeepLabCut, ResNets are used as the feature extractor for the input images, and their weights are typically fixed during training. The deconvolutional layers in DeepLabCut are used to up-sample the feature maps produced by the ResNets and to generate spatial probability densities for each body part. These probability densities represent the likelihood that a given body part is located at a particular pixel in the image.

To produce a final tracking result, DeepLabCut uses a readout layer for each body part of interest. Each readout layer takes as input the feature maps and the probability densities for that body part and generates a score map that represents the probability that the body part is located at each pixel in the image. During training, the network is trained to minimize the difference between the predicted score maps and the ground-truth score maps, which are generated from the labeled data. The weights of both the deep neural network and the readout layers are adjusted during training.

Once the network is trained, it can be used to track the positions of the body parts in new, unlabeled data by applying the network to each frame of the video or image data. The output of the network is a set of predicted score maps, which can be used to estimate the positions of the body parts.

2. Methodology

The methodology for this thesis involved several steps, including data collection, data labeling, model training, and evaluation. After evaluation, the overall goal was to improve the model, and the following subsections describe each step in detail.

2.1 Data Collection

Two different types of chickens were utilized in this thesis, namely a model toy chicken and actual live chickens. The model toy chicken was recorded at a resolution of 1920 x 1080 pixels and a frame rate of 30 frames per second on an iPhone 13. The toy chicken was placed on a flat rotating disk with folded green fabric under its feet in front of a green screen. While someone else

was rotating the chicken, there was another one recording it. The toy chicken was recorded in the University of Arkansas's Computer Vision and Image Understanding Lab.

The videos of live chickens were recorded on an at a resolution of 1280 x 720 pixels and a frame rate of 15 frames per second using a RealSense Depth Camera D435i. The live chickens were recorded at the University of Arkansas's Poultry Science Feed Mill. The environment was built using a wooden frame filled with shavings for the chicken to walk on. The background was a green piece of plywood that was painted for preparation beforehand. The chicken was persuaded to walk to the left with a feeder waiting on it and encouragement from a person.

In total, nine videos of live chickens were captured for training with each about a minute, and a single video lasting over a minute was recorded for the toy chicken.

Figure 1 and 2: Left is of the toy chicken. Right is of the live chicken in the environment.

2.2 Data Labeling

The recorded videos were human labeled using the DeepLabCut (Lauer, et al., 2021) and an online open-source software called ImgLab. There were 9 key points labeled:

1. Beak
2. Comb
3. Back of Head
4. Chest
5. Back
6. Start of Tail
7. End of Tail
8. Left Foot*
9. Right Foot*

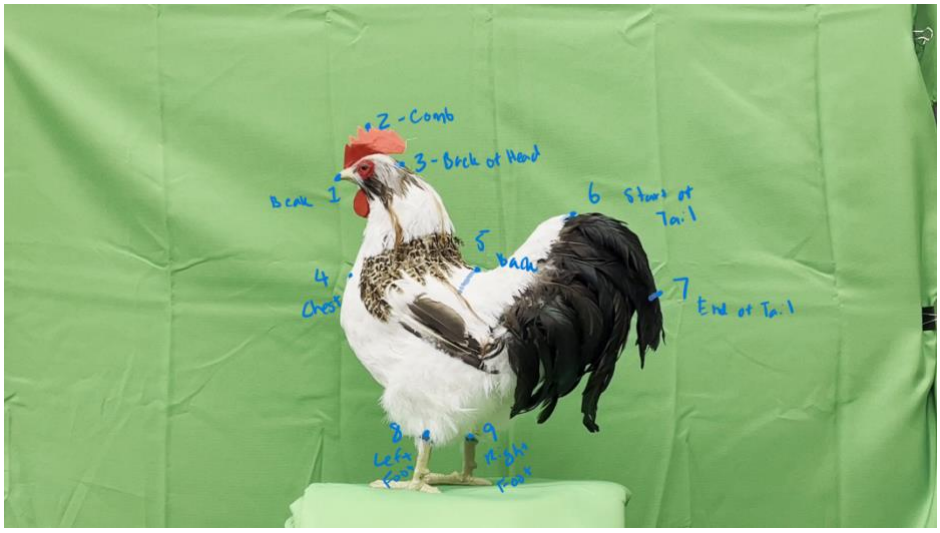


Figure 3: Landmarking Scheme

*A thing to note would be is that the feet were labeled from the top of the leg.

The labeling process involved manually placing a point on each key point of the chicken in each frame of the video. The data was stored as a coordinate on a Euclidean plane as (x, y) per body part on a single frame. A total of 373 labeled frames were created for the toy chicken while 300 labeled frames were created for the live chickens. The labeled frames using DeepLabCut were simultaneously stored in a csv and h5 file, and the ImgLab frames were stored into an XML file and converted afterwards into a csv file using a python script.

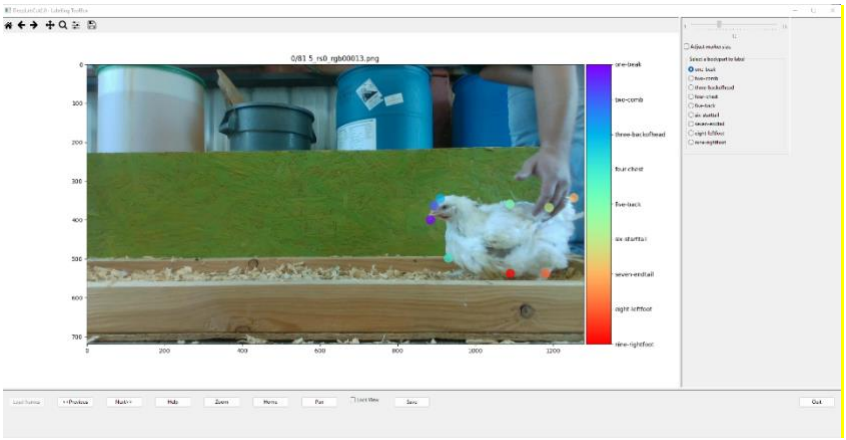


Figure 4: The DeepLabCut labeling software environment. (Nath, et al., 2019)

2.3 Model Training

The labeled frames were used to train a DeepLabCut model using the default settings of the software. Each model was trained using a ResNet-50 (He, Zhang, Ren, & Sun, 2015) backbone and a batch size of 1. The training process was performed using a graphics processing unit (GPU), specifically an RTX 3060 with 12Gb of VRAM. Each live chicken model was trained for 100,000 iterations while the toy chicken model was trained for 50,000 iterations, and the loss function used was the entropy loss function used in the DeepLabCut software. The trained models were saved for evaluation.

2.4 Evaluation

Each trained model was evaluated on a separate validation dataset consisting of three videos with 99 frames extracted total, so 33 frames for each video. The evaluation process involved running the trained model on the three videos and comparing the predicted key points with the manually labeled key points using both of their csv files in a python script. Then, the performance of each trained model was calculated using several metrics, including the root mean squared error (RMSE) and the average Euclidean distance. The evaluation of the trained models was conducted

not only on the overall performance, but also on a per-body part and per-frame basis. This allowed for a deeper analysis of the models' accuracy and identified specific body parts or frames that may have been more troublesome for the models to predict.

2.5 Improving the Previous Models

To improve the performance of the DeepLabCut model specifically for the live chickens, a selective approach was employed focusing on the insufficient training data for specific body parts and poses. These troublesome body parts and poses were determined by the per-body part and per-frame evaluation metrics.

To implement this approach, frames were selected from the validation dataset, containing poses that were incorrectly predicted by the model. The additional frames were then added to the training dataset. This resulted in a larger and more flexible training dataset, which aimed to improve the model's performance on troublesome poses. The updated model was then evaluated on the same validation dataset, and then its performance was compared to the previous model. This was done in total twice. In all there were a total of three models for the live chickens, consisting of 100 frames, 200 frames, and 300 frames.

3. Experimentation

This thesis involved two experiments, one using a toy chicken and the other using live chickens to create multiple models trained from DeepLabCut. A total number of four models were created.

Model	Number of Images Trained	Validation Set Size Used
0*	354	99
1	100	99

2	200	99
3	300	99

*Table 1: Number of images trained per model. * = Toy Chicken Frames*

3.1 Toy Chicken

For the toy chicken, a single video lasting over a minute was recorded, and a total of 373 labeled frames were created. The chicken was labeled with blue stickers covering each body part. Then, before inputting the images into DeepLabCut they were modified to have the blue stickers blurred out. After that, 95% (354) of these labeled frames were used to train a DeepLabCut model. The model was trained for 50,000 iterations and evaluated on the validation dataset consisting of three videos with 99 frames extracted total. At 50,000 iterations the cross-entropy loss finished at .00392 and the learning rate at .02. The toy chicken model did not achieve good results when it was evaluated on the 99-frame validation dataset, but this is to be expected when the chickens and environments were completely different. On the other hand, when it looked at the training dataset that DeepLabCut created using 5% of the 373 frames it was much more respectable.

Figure 5: Loss curve of Toy Chicken Model

3.2 Live Chickens

Nine videos of live chickens were used for the training data, each about a minute long, and a total of 300 labeled frames were created. The model was trained for 100,000 iterations and evaluated on a validation dataset consisting of three videos with 99 frames extracted total. At 100,000 iterations the cross-entropy loss ended at for each model from 1-3 was at .0025, .00326, and .0039 respectively while the learning rate finished at .02 for all. The first model achieved a decent amount of accuracy, but there were still some difficult body parts and poses that were challenging for the model to predict which caused the RMSE to be high because those specific frames had a high RMSE. Specifically, the challenging body parts that were focused on were the feet. In addition, those poses that were troublesome were where the chicken's right-side and back-side were on the frame. To improve the model's accuracy, a selective approach was employed, where additional frames containing the chicken's feet, right-side poses and back-side poses were brought into the training data. This resulted in two larger models that attempted to improve the overall performance of the model.

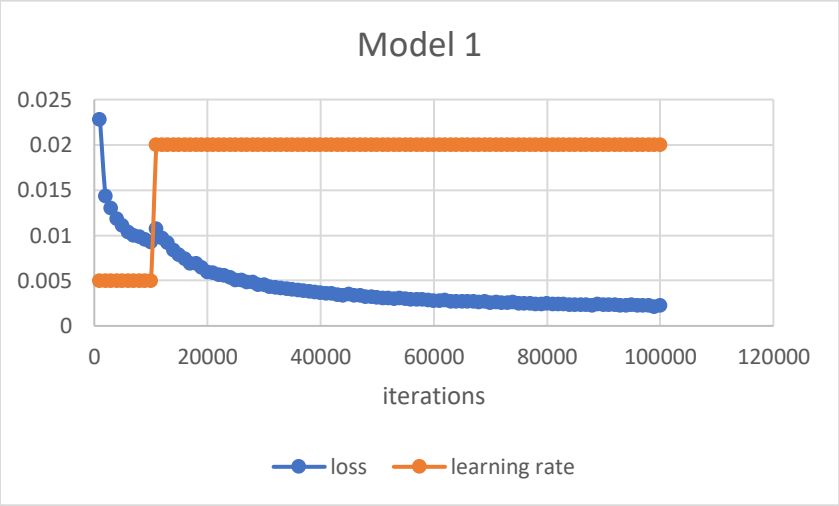


Figure 6: Loss curve of Model 1

Figure 7: Loss curve of Model 2

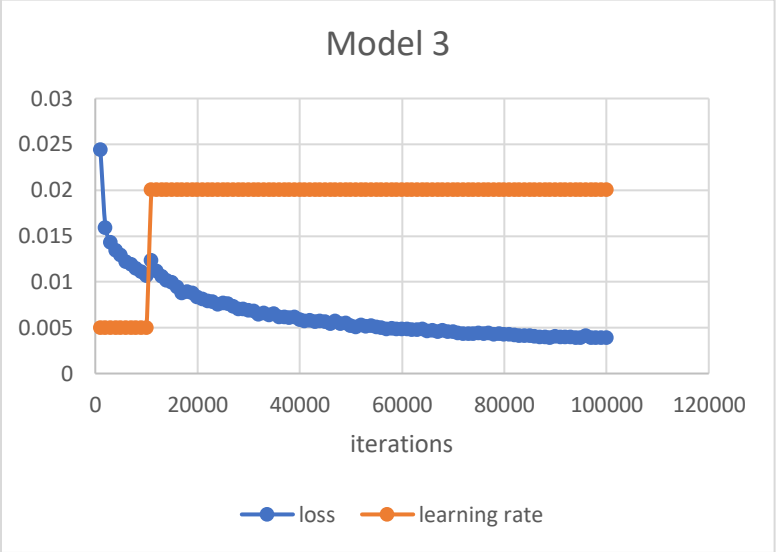


Figure 8: Loss curve of Model 3

4. Results

In total four models were trained, one toy chicken model and three live chicken models. The toy chicken model was labeled as model 0 as a baseline, while the other three models were labeled as Model 1-3.

4.1 Overall Results

Model Number	Training Dataset Size (Images)	RMSE (pixels)	Average Euclidean Distance (pixels)
0	354	423.3	346.85
1	100	31.81666667	16.76
2	200	38.43333333	18.23333333
3	300	29.99	16.71333333

Table 2: Overall Results for Each Model

The results presented in *Table 2* demonstrate that the initial model, model 0, developed for the toy chicken, performed poorly, which was anticipated given the difference between the toy chicken and live chickens. Notably, despite having a larger amount of training data, Model 2 had lower overall performance than Model 1. However, some of the body parts showed improved performance in Model 2 compared to Model 1 as seen in *Figure 9* and *Table 3* below, likely due to the selective approach employed, which emphasized increasing data for problematic body parts in this case the feet. Overall, it was not effective in improving the model's overall performance. However, in Model 3, the focus was shifted to increasing data for problematic poses, which resulted in a significant improvement in performance compared to Model 2, and a modest improvement in performance compared to Model 1.

4.2 Analysis: Each Body Part

#	Name	Model 1 RMSE	Model 2 RMSE	Model 3 RMSE
1	Beak	24.80535229	25.02269906	3.775533166
2	Comb	28.24472192	10.92814892	11.90033009
3	Back of Head	44.38343519	44.70036707	38.51959628

4	Chest	24.82678563	28.96018912	22.38868519
5	Back	21.16401652	47.26282822	24.9608201
6	Start of Tail	26.54049394	38.92452928	36.55669676
7	End of Tail	14.79500495	27.43706583	28.83080819
8	Left Foot	30.92961711	29.63552216	33.98016577
9	Right Foot	40.61869596	41.34056215	40.73725891

Table 3: The RMSE of every body part for each model.

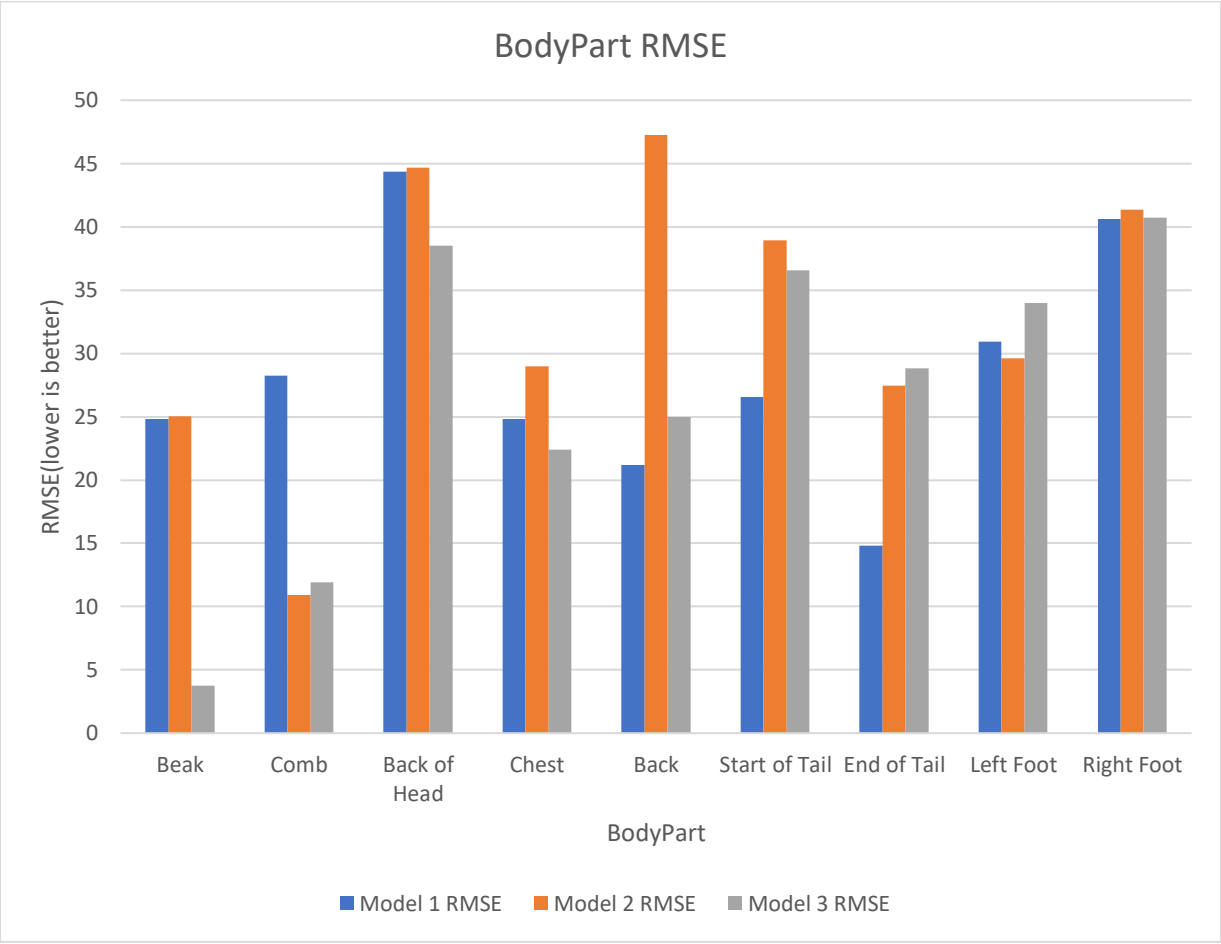


Figure 9: The RMSE of every body part of each model in a bar graph.

Overall, the results in *Figure 9* and *Table 3* provide valuable insights into the performance of the models on different body parts. The beak showed the most improvement from Model 3 to

other models, whereas the back of the head and the feet were the most challenging parts for all models. This is consistent with the fact that occlusions from the chicken's position, particularly when lying down, can make it difficult to accurately track these parts.

4.3 Analysis: Looking at a Particularly Troublesome Pose

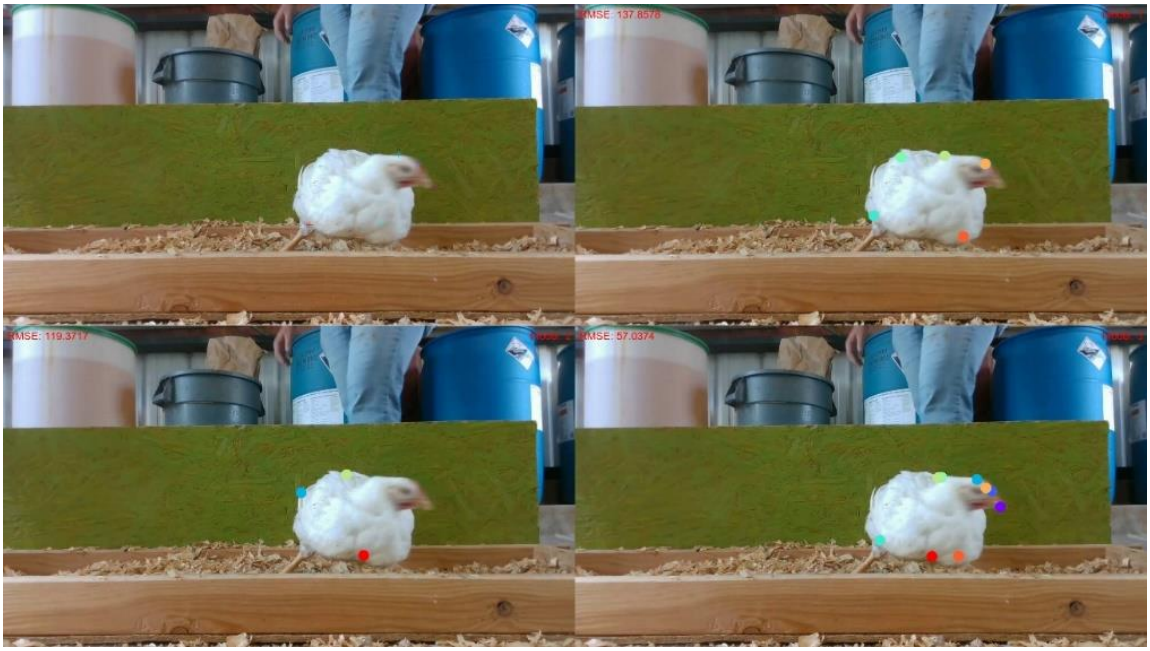


Figure 10: Image comparing the three model's performance on a troublesome frame/pose. The points are labeled with a probability cut off of .6 and DeepLabCut's built-in software.

When identifying troublesome poses using the per frame basis evaluator, a particularly troublesome pose caught the attention which is displayed in *Figure 10.* Here, we can see the progression of the models' performance on this pose, with the RMSE improving with each iteration of the model. The major decrease in the RMSE can be seen when observing the purple point representing the beak. When looking at Model 1 and Model 2, both models aren't sure where the beak is hence it not showing on the displaying algorithm where the probability cutoff is .6. But in Model 3, the beak is properly displayed in the correction position. However, even in Model 3, the

RMSE is still quite high at 57.0, and the orange point representing the left foot is clearly off from the ground truth values. The problem with this pose is that there was insufficient data where the chicken's right side was shown in the frame. This was since most of the videos recorded of the live chicken had the chicken walking in one direction which was a flaw in the methods approached in this thesis.

5. Conclusion

In this thesis, DeepLabCut models were trained on both a toy chicken and live chickens to assess their ability to accurately track various body parts. The results showed that the initial model developed for the toy chicken performed poorly on the validation dataset, which was expected due to the differences between the toy and live chickens. Using the 100-frame model, Model 1, as a base, a selective approach to increasing the training data was taken to increase performance. Model 2, which employed a selective approach to increase data for problematic body parts, showed some improvement in specific body parts compared to Model 1 but did not improve the overall model performance. However, Model 3, which focused on increasing data for problematic poses, resulted in a significant improvement in performance compared to Model 2 and a modest improvement compared to Model 1.

The analysis of individual frames/poses helps show how the selective approach increased the performance on those specific troublesome poses from model to model. This can also be seen from the increase in the performance of the beak's evaluation in Model 3 where the main error was when the chicken was turning in the other direction. In conclusion, this shows that DeepLabCut would be effective when sufficient data is given.

6. Further Work

Despite the improvements seen in Model 3, there is still room for improvement in accurately tracking the more challenging body parts. Future work could focus on collecting more data on different angles and poses of poultry. For example, above views, left views, right views should be used to gain the full range of motion of a chicken's movements. This should help improve the ability to track the body parts accurately and effectively. In addition, adding a larger variety of chickens would help generalize the models for different variants of chickens. With those changes, a huge improvement could be made for these models.

7. References

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Li, F.-F. (2009). ImageNet: a Large-Scale Hierarchical Image Database. *IEEE Conference on Computer Vision and Pattern Recognition*, 248-255. doi:10.1109/CVPR.2009.5206848

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *CoRR, abs/1512.03385*. Retrieved from http://arxiv.org/abs/1512.03385

Lauer, J., Zhou, M., Ye, S., Menegas, W., Nath, T., Rahman, M. M., . . . Mathis, A. (2021). Multi-animal pose estimation and tracking with DeepLabCut. *bioRxiv*. doi:10.1101/2021.04.30.442096

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 1281-1289. doi:10.1038/s41593-018-0209-y

Nath, T., Mathis, A., Chen, A., Patel, A., Bethge, M., & Mathis, M. W. (2019). Using DeepLabCut for 3D markerless pose estimation across species and behaviors. *Nature Protocols, 14*(7), 2152-2176. doi:10.1038/s41596-019-0176-0