Protocol 25: Lameness Detection

16.08.2019

1. Introduction

The past few weeks have been dedicated to pose refinement, as the errors from pose estimation seemed to be a big issue. The errors occur mostly when there are occlusions, including self-occlusion. Even after a few rounds of model training (train/test errors: 3.38/4.95 pixels), the predicted pose still contained several errors. Hence, the keypoints were manually refined to ensure reliable data for lameness detection.
Instead of doing the multi-class classification, binary classification is tried first to see if the network (HRNN) [1] can learn to identify lameness. A cow is considerd as clinically lame if the locomotion score is higher than 2 [2].

2. Experiment 1: Binary classification

The refined data of cow's pose were used to train the hierarchical RNN [2]. Two experiments with different number of sequences were executed: one with 120 and the other with 150 frames.

- **Dataset:** The original dataset contains 501 samples, each of which contains the coordinates of 25 skeletal joints from more than 100 video frames and a locomotion score as the label. The locomotion scores fall into two classes.

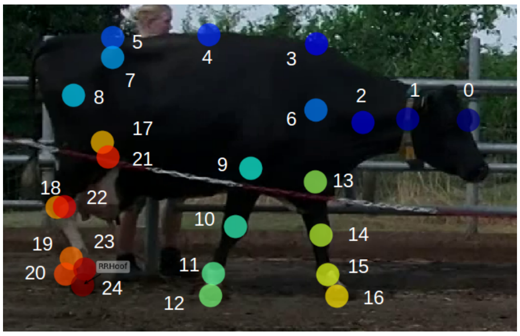

Figure 1: The 25 skeletal joints of a cow, and are divided into five parts.

- **Data split:** The dataset was first split into training and test set based on the cow's ID, with a ratio of 0.85/0.15. The training set was further split into five parts, each part was separately used for validation while the remaining was used for training (5-fold cross validation). Since the cows did not appear the same number of times, the size of each fold is slightly different.
- **Strategies:**
  - The value of learning rate was fixed in the beginning (0.001) and reduced by a rate of 0.1 after 10 epochs. The number of epochs is 100.
  - Sequence length: The length was fixed as 120 and 150 frames for each video.
  - Weight noise was added during the training process.
  - Batch size: 16.
- **Notes:**
  - The validation accuracy oscillates drastically when the learning rate is 0.001. One reason may be the highly unbalanced class.
  - The network seems to stop learning after reaching an accuracy of 80%.

– There are more false negatives in validation set, but more false positives in test set.

Result: The overall accuracy is 78% and 85% by taking 120 and 150 frames, respectively. The accuracy and loss curves

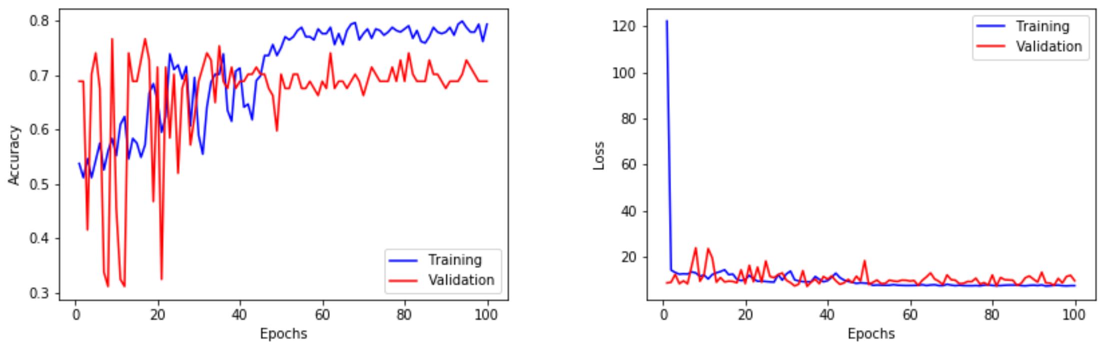

Figure 2: The curves of accuracy (left) and loss (right) in 100 epochs of training. The learning rate was reduced at 50th epoch.

2. Experiment 2: Regression

The difference of the network between classification and regression is the last layer. For regression, the last layer is the fully connected layer with only one neuron at the output.

- **Dataset:** The same dataset as in classification.
- **Data split:** The dataset split into training, validation, and test set based on the cow's ID, with a ratio of around 0.70/0.15/0.15.
- **Strategies:**
  – The value of learning rate was fixed in the beginning (0.001) and reduced by a rate of 0.1 after 20 epochs. The number of epochs is 100.
  – Sequence length: The length was fixed as 150 frames for each video.
  – Batch size: 16.
- **Notes:**
  – The training loss keeps decreasing with some jitters. The validation loss jumps up and down, not showing much improvement with the training.
  – The predicted scores converge to around 2 with small variations.
  – Increasing the number of neurons do not improve the result, so the problem could be that the keypoints do not provide enough information.

Result:

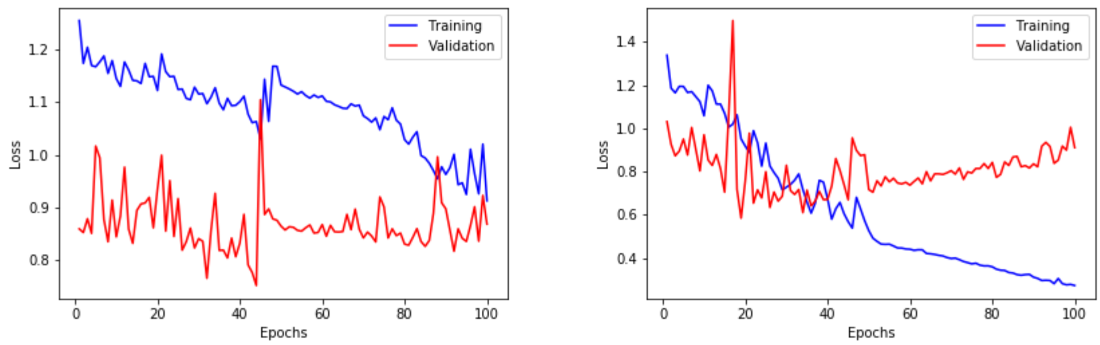

Figure 3: The loss curves in 100 epochs of training. The learning rate was reduced after 50 epochs. The figure on the right takes the output from several time steps into account.

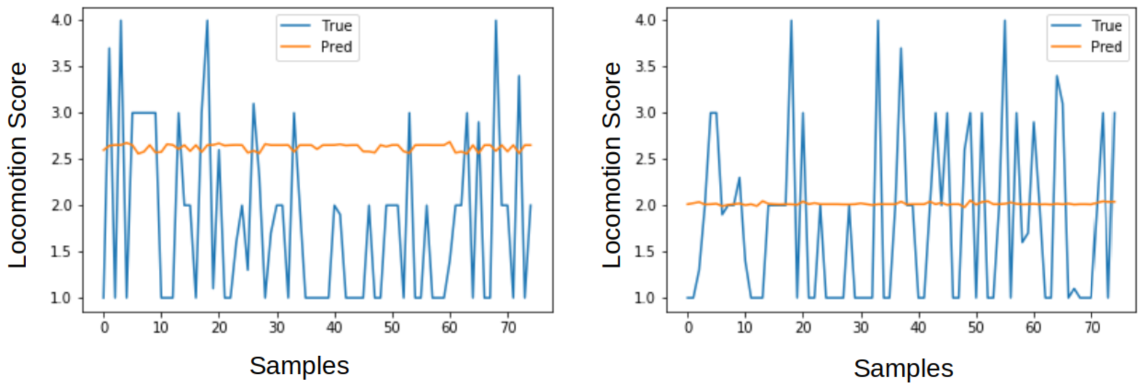

Figure 4: The true and predicted labels in test set. The figure on the left takes the output from several time steps into account.

3. Discussion

The lameness detection with binary classification was able to reach an accuracy of 80%, but the network could not learn much further by adding additional layers, neurons. The reasons may be the keypoints do not tell enough information about the lameness features, and that the coordinates of keypoints are still not accurate enough. Another issue is the unbalanced class distribution because of the small amount of data: both the validation and test sets contained only ten cows. This could explain why the validation accuracy jumped a lot in the beginning of training. In order to improve the result, the experiment should be repeated and the misclassified samples should be analyzed. The boundary will also be changed to see if the results will be different. The network show little training in regression. The predicted scores only stay around 2. Hence, the network should be further investigated.

Reference

[1] Y. Du, W. Wang, and L. Wang, Hierarchical recurrent neural network for skeleton based action recognition, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 11101118, 2015.

[2] D. Sprecher, D. Hostetler, and J. Kaneene, A lameness scoring system that uses posture and gait to predict dairy cattle reproductive performance, Theri- ogenology, vol. 47, no. 6, pp. 11791187, 1997.