Generating IMU Data for Horse Activity Recognition from Horse Videos

Xin Wan
University of Twente
PO Box 217, 7500 AE Enschede
The Netherlands
x.wan@student.utwente.nl

ABSTRACT
Inertial Measurement Units (IMU) which including accelerometers, gyroscopes and sometimes contains magnetometers, these data are widely used to recognize animal activities in the farm, when it comes to the wildlife, it is costly and dangerous to install the IMU sensors on the wild animals, Moreover, it is also a challenge to monitor and track these wild animals, therefore this makes it difficult to obtain a larger number of labelled IMU data sets to train a desirable machine learning model for Animal activity recognition (AAR). This paper will conduct research on generating IMU data from horse videos of which contains 3 activities of horses (standing, walking and trotting) and then evaluate whether these generated IMU data can be used to classify these three horse activities with a classification model that trained with the real IMU data for horse activity classification. This research will consist of four major parts, the first part is to generate the two-dimensional coordinates of each key point of the horse from its video through the animal pose estimation tool. The second part is to use the coordinates of these key points and the real IMU data obtained from the horse to train a regression model with a neural network and then use the regression model to predict the simulated IMU data. The third part is using the real IMU data to train a classification model for classifying these three horse activities. Finally, using the simulated IMU data as the augment data to predict these three horse activities with the trained classification model to evaluate its performance. Since accelerometers are used more often than gyroscopes and magnetometers for Animal activity recognition (AAR), therefore, in this research, I only consider generating the accelerations from horse videos. In the future, it will be improved for generating a greater variety of synthesis IMU data (also including the Gyroscope data) from videos of more animal species and more animal activities.

Keywords
Machine learning, pose estimation, regression model, IMU data, classification model

35th Twente Student Conference on IT, July. 2nd, 2021, Enschede, The Netherlands.

1. INTRODUCTION
Recently, with the fast development of deep learning techniques, which achieved good performance in many fields, for example, speech recognition, visual object recognition and genomics [7]. Animal activity recognition (AAR) based on wearable IMU sensors using deep learning has been widely used for tracking and monitoring the behavior of animals in the natural environment [1,2,9,10]. To train a highly precise machine learning model a large amount of labelled datasets should be obtained. Nevertheless, there is a challenge for obtaining such a large amount of labelled data, since in general there are two ways to recapture the sensor data: one is transmitted through the wireless network and another one is retrieved the data later [10]. However, due to the limited power supply, transmission efficiency and other electrical issues the former approach is costly [9]. For the second approach, it is even dangerous and costly to recapture these sensor data from the wild animals later, especially when the testing animals are some ferocious carnivores such as tigers, leopards. Therefore, based on these two situations, it is sometimes impossible to retrieve these data from wild animals, thus making it difficult to obtain a large amount of labelled IMU sensor data. This IMU data generator can deal with this challenge since it can generate synthesis IMU data according to the type of horse activity that used to train the regression model. It means that these simulated data will be automatically labelled with the regression model according to the type of the horse activity Therefore, when I want to train a new classification model for horse activity classification that classify more than three activities, this synthesis labelled IMU data can be used as the data augmentation together with the real IMU data and further be mixed with the real IMU data to train this model. It solves the lack of labelled IMU data.

2. PROBLEM STATEMENT
There is some research done for generating IMU data from videos, but most of them are generating IMU data from videos of humans [3,6,8], so researches on generating IMU data from videos based on animals are still lacking. Therefore, this research will conduct research on generating simulated IMU data from horse videos and then using these generated IMU data to train a classification model for horse activities classification and to evaluate whether the simulated IMU data can be used for horse activities recognition like real IMU data and in what extent?

2.1 Research Question
There are two research questions based on the problem statement that need to be answered during the research:

1. Whether the simulated IMU data can be used as augmenting data to improve the performance of the classification model for horse activity classification?
2. How to evaluate the classification model for horse activity classification?

3. RELATED WORK

Some researches related to generating the simulated IMU data from videos, but most of these researches are generating IMU data from videos of humans [3,6,8]. These three types of research present three different ways to generate simulated IMU data from videos of humans. For the first research, the researchers use both the collected real IMU data and 2D poses generated from the synchronized videos when collecting the IMU data to train a regression model with a deep neural network and then use this trained model to generate simulated IMU data from new videos [6]. For the second research, the researchers create a pipeline tool called IMUTube to extract IMU data directly from videos without training a regression model. "The procedure is based on a computer vision pipeline that first extracts 2D pose information, which is then lifted to 3D. Through tracking individual joints of the extracted 3D poses, we are then able to generate sensor data, such as tri-axial acceleration values, at many locations on the body" [4]. In these two types of research, they used the technique called pose estimation to extract the 2D pose information from the videos, there are many existing open-source tools are used to implement this function, the most famous one for human pose estimation is Openpose [4], while another one mainly for animals pose estimation is DeepLabCut [5].

For the last research, the researchers create a generative cross–model to generate a synthesis accelerometer data from videos. The process contains two parts, one is a video encoder which uses the technique to compress video into feature vectors, another one is conditional GAN (cGAN) which uses the feature vector from the video as additional information to be conditioned on [8].

4. METHODS

In this section steps of doing this research will be shown in the following sections. All the data processes and model training processes are running using the python code.

4.1 Data Collection

The real IMU data and its synchronized videos are collected from a horse riding school (Manege De Horstlinde) near the campus, there are in total 4 different horses which involves 3 activities: standing, walking and trotting with different colors and heights are used to collect data. For the IMU data part, it contains 3-axis (x, y, z) acceleration. These data are collected with a 30 Hz sample rate, which means that the data is collected 30 rows in one second. For the video part, the video is taken with three activities of horses, for example, standing, walking and trotting and the sample rate of the video is also 30 Hz which can fit the sample rate of the IMU data. Since all videos were taken using the same free camera app, therefore, sizes of all frames of these videos are the same, with 960 widths and 544 height, respectively. In order to make the position of the horse in all videos roughly the same, I put a bag on the ground at a fixed position as a reference object, and the position of the person who took the video and the position of the mobile phone are also fixed, so the mobile phone used for taking videos does not move. In this case, the total shooting time of the horse walking from one end of the video to the other end is about 8 seconds. For trotting, the total shooting time of the horse trotting from one end of the video to the other end is about 5 seconds. Since for standing, the horse is not moving, I make the length of the video a bit longer which is 10 seconds. Therefore, in total for each horse, there are three videos: standing videos, walking videos, and trotting videos, with the length of the video around 10s, 8s, 5s respectively. The reason for taking the video like this is that I would train three regression models with these 3 activities respectively and to make the x-y coordinates obtained from the DeepLabCut not be affected too much by the position of the horse in the video, since the x-y coordinates obtained are based on the size of the frame of the video and the position of the horse in the video, in this case, the x -y coordinates are related to the width and height of the frame of the video and if the horse is in a different position or the man taking the video in a different position or angel, then the x-y coordinates will always be different, thus making it very hard to predict the simulated IMU data from the regression model.

4.2 Acquiring 2D poses of horses

The researchers have shown that for obtaining the information from videos, they all using pose estimation to extract the 2D pose information of all key points from these videos [4,6], therefore, in this research, I also used this method to extract information from videos.

To obtain the 2D poses of horses, I used the open-source tool mainly for pose estimations from animals based on transfer learning with a deep neural network with a good performance called DeepLabCut, it works with a variety of animals, for example, cats, dogs, cheetah, macaque and horses and it also contains pretrained models for these animals that you do not need to train the model yourself [5].

The whole process can be done using its Google Colab code to analyze videos with pre-trained models [5]. The pre-trained model of a horse using 22 key points to label the horse of which these 22 key points are shown from table 1 as below:

Table 1. 22 key points of a horse

Name of Keypoint			
Nose	Eye	Nearknee	Nearfrontfetlock
Nearfrontfoot	Offknee	Offfrontfetlock	Offfrontfoot
Shoulder	Midshoulder	Elbow	Girth
Wither	Nearhindhock	Nearhindfetlock	Nearhindfoot
Hip	Stifle	Offhindhock	Offhindfetlock
Offhindfoot		Ischium	

After running following the instruction of its code, you will get two files, one is a CSV file with all x-y coordinates of all 22 key points and one is the simulated horse video with 22 key points added as shown from figure 1. Since the smartphone was put and fixed vertically on the left front leg of the horse, which is the nearest to the key point Nearfrontfetlock, so for all these 4 horses, I used the x-y coordinates of this key point as the input of the regression model.

Figure 1: The simulated horse video with 22 key points added.

4.3 Smartphone placement

The smartphone is used as an IMU data collector in my research since nowadays, almost all smartphones have a built-in IMU sensor, and it is the sensor that people can obtain the easiest, since almost everyone has a smartphone, furthermore, it is easier to put and fix a smartphone on the horse.

The smartphone is put and fixed vertically on the left front leg of the horse using a bandage and the location and direction of the smartphone can be seen in the figure 2 below. All four horses are using this way to put and fix smartphones.

Figure 2: The red circle indicates the position and orientation of the smartphone.

4.4 Data Preprocessing

To synchronize the IMU data and the video, I used the timestamp of the sensor data and then convert it to the real-time it represents using python code, and for displaying time on videos, I used a free camera app that can display current time when taking the video, this app can accurate the time to millisecond, then I convert both the start time and end time of the video into a timestamp, and find the closest starting and ending IMU data based on these two timestamps. To train a regression model, the length of the input and output must be the same, that is the reason why I sync IMU data and video like this. When I need to train a classification model, I also need to label which activities belong to which IMU data, therefore, I will also use this way to label the activity.

After I sync the IMU data and the video of each horse with all 3 activities, I combined the x-y coordinates of the key point Nearfrontfetlock with the real IMU data from three horses into one file and use this data of this file to train the regression model and the remaining one horse data to test the regression model. This is the same way as I used to train a classification model, since there are 3 activities standing, walking and trotting, so I trained 3 regression models specifically to these 3 activities respectively. It means that I have three regression models, one for predicting horse standing, one for predicting horse walking and one for predicting horse trotting. The details of these three activities can be seen from the table 2. All data were normalized using the z-score normalization, after the z-score normalization, all the processed data will fit the standard normal distribution, that is, the mean value is 0 and the standard deviation value is 1.

Table 2. Descriptions about the 3 horse activities

Activity	Description
standing	The horse is standing still and does not move.
walking	The horse is moving from one place to another place in a normal speed which is slower than the trotting.
trotting	The horse is moving from one place to another place in a quicker speed than walking.

4.5 Train the regression model

The regression model that I used is a neural network called multilayer perceptron (MLP) which using a number of hidden layers contains neurons, since the dataset that collected is relatively small, then I only used one neuron [9], this regression model with a neural network was implemented using the Python modules for machine learning and data mining called sklearn [13]

The parameter that chosen for the neural network is the default setting of a neural network of the sklearn module, with 1 hidden layer which contains 100 neurons and the activation function for the hidden layer is relu, the solver weight optimization is lbfgs which can converge faster and perform better for small datasets. The L2 penalty (regularization term) is the default value 0.0001 and the maximum number of iterations is 1000.

4.6 Train the Classifier model

The research has shown that the KNN (k-Nearest Neighbor algorithm) classifier obtains the highest performance in animal activities recognition [12]. In this research, I used the KNN classifier to classify the 3 activities (standing, walking and trotting) of horses. The attributes that I used for the KNN classifier is that the number of neighbors that should be used to classify activities is 5. It means that for a given sample that will be used to predict activities, this KNN algorithm will find the 5 number of training samples closest to it in the training set based on the distance metric, the distance metric that I used is the default one called minkowski, then make predictions based on the activities of the majority of these 5 neighbors. This classification model with KNN algorithm was implemented using the Python modules for machine learning and data mining called sklearn [13].

5. RESULTS AND DISCUSSION

In this section, the results of the classification model will be analyzed, since the two research questions of this research are one is "Whether the simulated IMU data can be used as augment data to improve the performance of the classification model for horse activity classification?", the other one is "how to evaluate the classification model for horse activity classification?", therefore, I do not discussed the results of the regression model in this research, only discuss the results of the classification model.

5.1 Results of the classification model

To evaluate results of the classification model, there are four most importance performance metrics, which are precision, recall, F1-score and accuracy. These will be discussed in the following subsections.

5.1.1 Precision

The precision describes that among all the results where the model prediction is positive, the proportion of the model prediction being correct.

The formula to calculate the precision can be seen:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

where TP represents the true positives, FP represents the false positives.

5.1.2 Recall

The recall describes that among all the results where the model prediction is negative, the proportion of the model prediction being correct.

The formula to calculate the recall can be seen:

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

where TP represents the true positives, FN represents the false negatives.

5.1.3 F1-score

The F1-score combines the results of the precision and recall's output.

The formula to calculate the F1-score can be seen:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

Where precision represents the value from section 5.1.1 and recall represents the value from section 5.1.2.

5.1.4 Accuracy

The accuracy describes the proportion of all the correct results in the classification model to the total observation value.

The formula to calculate the accuracy can be seen:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Where TP represents the true positives, TN represents the true negatives, FP represents the false positives, FN represents the false negatives.

5.1.5 Result of the classification model trained with real IMU data

In order to see whether the simulated IMU data can improve the performance of the classification model for the horse activities recognition. In this section, firstly, I only trained the classification model with real IMU data, these training real IMU data was combined from all 3 horses, which contains in total 2020 samples, 900 for standing, 770 for walking and 350 for trotting, then I used the one horse IMU data which comprises in total 660 samples, 300 for standing, 240 for walking and 120 for trotting to test the trained classification model. To see the performance of using the training data to test the trained classification model and the performance of using the test data to test the trained classification model. I analyzed 2 figures: figure 3, figure 4 and 2 tables: table 3, table 4. The classification model trained both with real and simulated IMU data will be discussed in the section 5.1.6.

For calculating these four performance metrics, I used the function from the "sklearn.metrics" which is the API from the sklearn [13]. The detailed performance metrics of the classification model trained only with real IMU data can be found in table 3, 4. Since the classification can classify more than 2 activities, I also calculate the macro average values of the precision, recall and F1-score of these three activities, respectively. It can be seen from the table 3 that the overall accuracy of these total 2020 sample is 0.88 which is an acceptable value, since from the other research the acceptable value for accuracy on animal activities recognition is around 90% [14]. It can also be seen from the table 3 that the overall performance for both the standing is quite good, but the overall performance for both the walking and trotting activity is relatively bad, especially for the trotting activity, this is because when collecting the IMU data some part of the trotting data is in the same scale of the walking data, thus it leads that when the real activity is trotting then the classifier highly potential predict it as walking.

For the performance of the classification model which trained with only real IMU data (testing using the 1 horse testing set), the result can be seen from the table 4, the overall performance of standing is as good as it of table 3, but the overall performance of both walking and trotting is even worse, this is reasonable, since the number of samples of the test data is smaller than the training data, thus makes the classifier even higher potential to predict it wrong. Furthermore, this also leads that the overall accuracy is 0.80 which is lower than the value of table 3. The detailed confusion matrix that describe the exact number of samples that predict true or false of both situations can be seen from the figure 3 and figure 4.

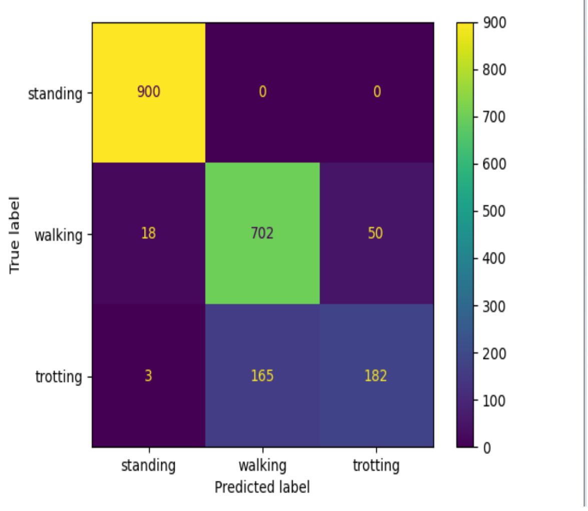

Figure 3: The confusion matrix of the classification model which trained with only real IMU data (testing using the 3 horses training set).

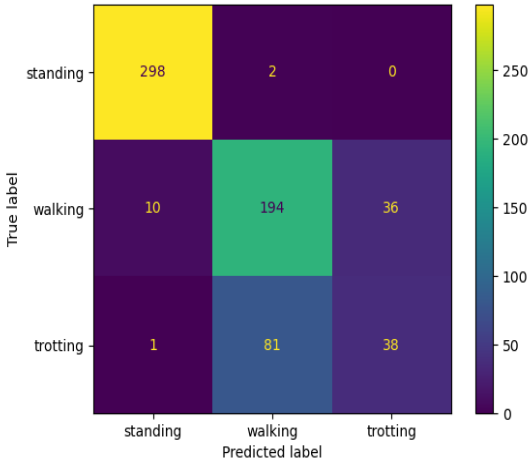

Figure 4: The confusion matrix of the classification model which trained with only real IMU data (testing using the 1 horse testing set).

Table 3. Main metrics of the classification model which only trained with real IMU data (testing using the 4 horses training set)

activity	precision	recall	F1-score	support
standing	0.98	1.00	0.99	900
walking	0.81	0.91	0.86	770
trotting	0.78	0.52	0.63	350
accuracy		0.88		2020
Macro average	0.86	0.81	0.82	2020

Table 4. Main metrics of the classification model which only trained with real IMU data (testing using the 1 horse testing set)

activity	precision	recall	F1-score	support
standing	0.96	0.99	0.98	300
walking	0.70	0.81	0.75	240
trotting	0.51	0.32	0.39	120
accuracy		0.80		660
Macro average	0.73	0.71	0.71	660

5.1.6 Result of the classification model trained with both real and simulated IMU data

In this section, I used the same test data (one horse dataset) for the classification model to generate the simulated IMU data from the regression model which comprises in total 660 number of samples, 300 for standing, 240 for walking and 120 for trotting. And then I combined these 660 simulated IMU data with the original 2020 real IMU data to train the classification model again. Therefore, the total number of samples that fed into the new classification is 2680. I introduced the same tables and figures as the section 5.2.5. which results the figures 5, figure 6 and tables 5, tables 6.

It can be seen clearly from the table 5 that the overall performance of all three activities are improved compared with the values from table 3 (training with only the real IMU data). The macro average values of precision, recall and F1 score are all increased, increased from 0.86 to 0.87, 0.81 to 0.84 and 0.82 to 0.85, respectively. Since the overall performance is increased, the overall accuracy is also increased from 0.88 to 0.90. The overall performance of the trotting is improved noticeable from 0.78 to 0.81 in precision, 0.52 to 0.60 in recall and 0.63 to 0.69, respectively.

It can also be seen from the table 6 that there is also an overall improvement compared with the value from table 4, but this improvement is very small, the accuracy increased from 0.80 to 0.81 and the macro averages values of precision, recall and F1 score are all increased 0.01 with 0.73 to 0.74, 0.71 to 0.72 and 0.71 to 0.72, respectively. From figure 6, it shows clearly that there are 43 trotting activities are True Positives which means that these 43 activities are predicted trotting and also the actual values are trotting. The number of true positives of the trotting is increased 5 from the value of figure 4.

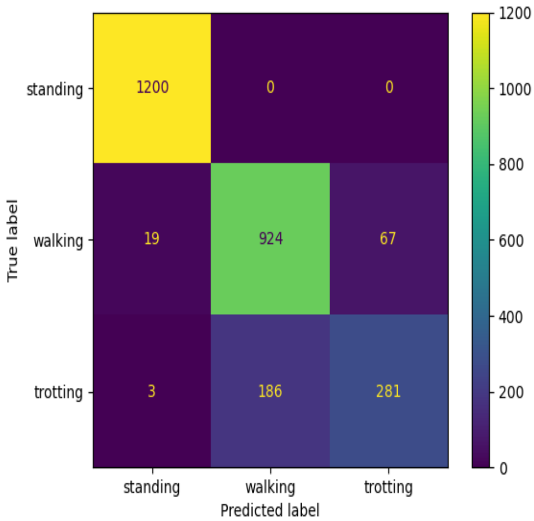

Figure 5: The confusion matrix of the classification model which trained with both real and simulated IMU data (testing using the 3 horses training set).

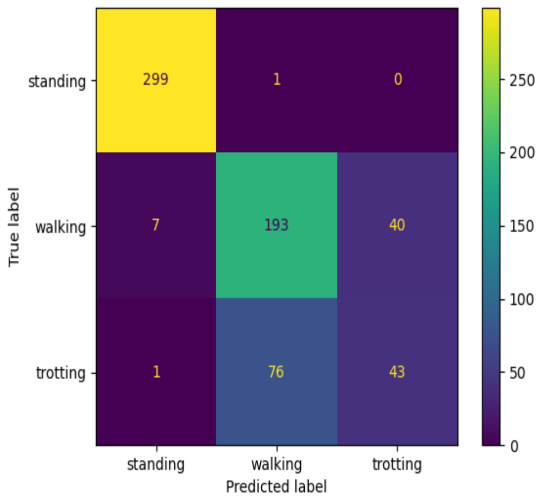

Figure 6: The confusion matrix of the classification model which trained with both real and simulated IMU data (testing using the 1 horse testing set).

Table 5. Main metrics of the classification model which trained with both real and simulated data (testing using the 3 horses training set)

activity	Precision	recall	F1-score	support
standing	0.98	1.00	0.99	1200
walking	0.83	0.91	0.87	1010
trotting	0.81	0.60	0.69	470
accuracy		0.90		2680
Macro average	0.87	0.84	0.85	2680

Table 6. Main metrics of the classification model which trained with both real and simulated data (testing using the 1 horse testing set)

activity	precision	Recall	F1-score	support
standing	0.97	1.00	0.99	300
walking	0.71	0.80	0.76	240
trotting	0.52	0.36	0.42	120
accuracy		0.81		660
Macro average	0.74	0.72	0.72	660

6. CONCLUSION

In conclusion, this research describes a way of using machine learning to generate simulated IMU data from horse videos. The whole process contains firstly, using the open source tool DeepLabCut [5] with deep learning to generate the 2D pose coordinates of the 22 key points of a horse, and then train a regression model with neural network to predict the simulated IMU data from the 2D pose coordinates of the key point of a horse, and then combining the simulated IMU data with the real IMU data can improve the performance of the classification model for horse activity classification.

7. FUTURE WORK

Above all, this IMU data generator can generate simulated IMU data and these data can improve the performance for horse activities recognition, but there are still many places can be improved due to the restriction of collecting both the real IMU data and its synchronized videos of animals. Firstly, this generator can only work with horses, if we can collect the same types of dataset as I collected (both IMU data and its synchronized videos) from other animal species then this generator can also work for these animal species, secondly, this generator can only work with 3 activities, standing still, walking and trotting, if the dataset contains other activities, for example, grazing, sitting can be collected, then these activities can also be added into the generator. Thirdly, the pretrained model called horse_sideview that used to generate the 2D pose estimation of horses from the DeepLabCut [5] is currently only working for horses moving from left to right [11]. It means that in this research, only horse videos that taken the same way as shown in figure 7 can be used to generate the IMU data.

Figure 7: The example of a horse video that works for this IMU data generator

8. REFERENCES

[1] Jacob. W. K. 2020. Hiding in the Deep: Online Animal Activity Recognition using Motion Sensors and Machine Learning. University of Twente. DOI= https://doi.org/10.3990/1.9789036550550

[2] Yingqi. P., Naoshi. K., Tateshi. F., Tetsuhito. S., Wulandari, Hidetsugu.Y. and Erina. I. 2019. Classification of multiple cattle behavior patterns using a recurrent neural network with long short-term memory and inertial measurement units, Volume 157, Pages 247-253, ISSN 0168-1699, Computers and Electronics in

Agriculture. DOI = https://doi.org/10.1016/j.compag.2018.12.023

[3] Hyeokhyen. K., Catherine. T., Harish. H., Yan. G., Gregory D. A., Nicholas. D. L., and Thomas. P. 2020. IMUTube: Automatic Extraction of Virtual on-body Accelerometry from Video for Human Activity Recognition. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4, 3, Article 87 (September 2020), 29 pages. DOI = https://doi.org/10.1145/3411841

[4] Zhe. C., Gines. H., Tomas. S., Shih - En. W. and Yaser. S. 1 Jan. 2021. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172-186. DOI = 10.1109/TPAMI.2019.2929257

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

[6] Vitor. F. R., Peter. H., Onorina K. and Paul L. 2019. Let there be IMU data: generating training data for wearable, motion sensor-based activity recognition from monocular RGB videos. In Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct). Association for Computing Machinery, New York, NY, USA, 699–708. DOI = https://doi.org/10.1145/3341162.3345590

[7] LeCun. Y., Bengio. Y. and Hinton. G. 2015. Deep learning. Nature 521, 436–444 (2015). DOI = https://doi.org/10.1038/nature14539

[8] Shibo. Z. and Nabil. A. 2020. Deep generative cross-modal on-body accelerometer data synthesis from videos. In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (UbiComp-ISWC '20). Association for Computing Machinery, New York, NY, USA, 223–227. DOI = https://doi.org/10.1145/3410530.3414329

[9] Jacob. W. K., Helena. C. B., Duc. V. L., Nirvana. M. and Paul. J. M. H. 2017. Generic online animal activity recognition on collar tags. In Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers (UbiComp '17). Association for Computing Machinery, New York, NY, USA, 597–606. DOI = https://doi.org/10.1145/3123024.3124407

[10] González. L. A., Bishop-hurley. G. J., Handcock. R. N., and Crossman. C. 2015. Behavioral classification of data from collars containing motion sensors in grazing cattle. Computers and Electronics in Agriculture 110 (2015), 9102. DOI = http://dx.doi.org/10.1016/j.compag.2014.10.018

[11] Mathis. A., Biasi. T., Schneider. S., Yuksekgonul. M., Rogers. B., Bethge. M., & Mathis. M. W. 2021. Pretraining boosts out-of-domain robustness for pose estimation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 1859-1868).

[12] Hamelers. L. H. 2018. Investigating generic online animal activity recognition across different animal species (Bachelor's thesis, University of Twente).

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011

[14] Gao. L. L., Campbell. H., Bidder. O., and Hunter. J. 2012. A Web-based semantic tagging and activity recognition system for species' accelerometry data. Ecological Informatics. DOI = https://doi.org/10.1016/j.ecoinf.2012.09.003