Graphical Abstract

A framework for energy-efficient equine activity recognition with leg accelerometers.

Anniek Eerdeken, Margot Deruyck, Jaron Fontaine, Luc Martens, Eli De Poorter, David Plets, Wout Joseph

Highlights

A framework for energy-efficient equine activity recognition with leg accelerometers.

Anniek Eerdekens, Margot Deruyck, Jaron Fontaine, Luc Martens, Eli De Poorter, David Plets, Wout Joseph

- Novel end-to-end framework for energy-efficiently monitoring of equine behavior.
- A CNN, hybrid CNN, and RF classifier were first used in equestrian applications.
- The hybrid CNN detects 99.59% of the behaviours with 10 Hz one leg accelerometer data.
- Optimized hybrid CNN permits over 500 days of classification with a 600 mAh battery.

A framework for energy-efficient equine activity recognition with leg accelerometers.

Anniek Eerdekens^{a,c}, Margot Deruyck^a, Jaron Fontaine^b, Luc Martens^a, Eli De Poorter^b, David Plets^a, Wout Joseph^a

^a WAVES, Department of Information Technology, Ghent University - imec, Technologiemark-Zwijnaarde 126, B-9052 Ghent, Belgium

^b IDLab, Department of Information Technology, Ghent University - imec, Technologiemark-Zwijnaarde 126, B-9052 Ghent, Belgium

^c Corresponding author: anniek.eerdekens@ugent.be

Abstract

Automated behavioral detection and classification through sensors can enhance the horses' health and welfare. Since monitoring needs to be carried out continuously, an energy-efficient method is needed. The number of logging axes, sampling rate, and selected features of accelerometer data not only have a significant impact on classification accuracy in activity recognition but also on the sensors' energy needs. Three models are designed for detecting horses' activities namely, a Random Forest classifier (RF), a Convolutional Neural Network (CNN) and a hybrid CNN, i.e. a CNN fused with statistical features that retain knowledge about the global time series form. The models are validated using an experimental dataset obtained from six different horses each performing seven different activities. The results indicate that using one leg accelerometer data is sufficient for high classification accuracies ($> 98.6\%$) for the three models. The hybrid CNN substantially improves over the RF and CNN at a sampling rate of 5 Hz with an increase in accuracy of 1.88% and 2.79%, respectively. The hybrid CNN is capable of excellent performance, detecting nearly 99.59% of the behaviours at 10 Hz. The experiments show that the CNN

and hybrid CNN use as much as 17.2 and 13.5 times less energy respectively, than the RF-based method. The experimental results showed that, although the recognition rate of the proposed optimized hybrid CNN model is similar to that of the original hybrid CNN model, the former requires only 6% of the Multiply-Accumulate (MAC) operations. For automatic detection of the horses' behavior those results suggest using one leg accelerometer data sampled at 10 Hz classified by an optimized hybrid CNN.

Keywords:

equines, accelerometers, signal processing, behaviour classification, internet-of-animals, energy-efficient method, machine learning

1. Introduction

Tracking equine behaviour can serve four purposes. First, monitoring of a sequence of pain-related activities such as rolling, pawing and flank watching can act as a health indicator and allow early identification of time-sensitive health problems such as colics (van Loon and Van Dierendonck, 2015). Second, the study of different gaits during training such as standing, walking, trotting and cantering provide insights into training and improve the performance of sporthorses, third, the study of a pregnant mare's movements suggests the time of birth (Giannetto et al., 2015), and fourth, the fitness level of horses is also an important determinant of injuries and therefore, quantification of training sessions may have important added value in the assessment of performance ability and potential future injuries in sport horses (Munsters et al., 2014). Animals can be directly observed but this is time consuming and labour intensive (Pen-

ning, 1983; Müller and Schrader, 2003). Sensors that automatically monitor animal behavior are a valuable solution for these purposes (Rushen et al., 2012). Also, machine learning provides new ways to detect possible health problems early by picking up signs hidden in the activities of horses not visible to the human eye.

Since equine activity must be monitored over a long period of time, energy consumption of the wearable device is a critical parameter. The battery life can be extended by for example operating at lower sampling rates, logging fewer accelerometer axes or reducing the number of features (Lee and Annavaram, 2012). A recent study analyzing the lying, standing, and feeding behavior of cows concluded that the classification performances were not highly influenced by the use of fewer logging axes (Benaissa et al., 2019b). However, different studies have proven that a decrease in sampling rate negatively influences the performance of traditional machine learning tools such as Support Vector Machines, Random Forest and Naive Bayes classifiers (Walton et al., 2018; Hounslow et al., 2019). This same result was found in a study that investigated the usage of accelerometers in combination with Convolutional Neural Networks (CNNs) to detect seven behaviours from six different horses (Eerdekens et al., 2020). The study revealed that CNNs perform excellently on classification problems based upon accelerometer data to learn and discriminate among different equine activities but the accuracy of the model decreased on average with about 4.75% when the sampling rate was decreased from 200 Hz to 25 Hz.

A well-informed decision on optimum sensor measurement settings such as sampling rate, logging axes and selecting a relevant subset of features will result in significant improvements in

the algorithm’s energy consumption. Although literature studies assess these aspects of accelerometer data for human activity classification, there are no studies in equine tracking that simultaneously assess the impact of these variables on behavior recognition accuracy and at the same time consider the energy usage of the machine learning algorithms.

To generate and compare training models for predicting different activity types based on accelerometer data, three supervised machine learning (ML) classifiers – a Convolutional Neural Network (CNN), an hybrid CNN and a random forest (RF) – are used in this study. Hybrid CNNs, i.e. a CNN for local feature extraction combined with simple statistical features (mean, standard deviation, etc.) that preserve information about the global form of time series, outperform baseline approaches and establish state-of-the-art results on major activity classification tasks, while reducing computational cost (Ignatov, 2018). A CNN has the advantage of automatic feature extraction by using low computational complexity while for RFs manual feature extraction is necessary (Srivastava et al., 2014). However, Random Forest analysis is a leading ML algorithm that has been successfully applied to behaviour classification based upon accelerometer data in a variety of species but not yet for the detection of behaviors of horses (Casale et al., 2011; Nathan et al., 2012; Graf et al., 2015; Wang et al., 2015).

1.1. Our contributions

This research introduces a machine learning-based approach to identify equine behavior using accelerometer data. This work utilizes three classifiers, namely: Convolutional Neural Network, hybrid CNN, and Random Forest to classify experimental ac-

Figure 1: Proposed approach.

celerometer data from six horses performing seven activities. A comprehensive optimization is performed to reduce the computational cost and energy usage of the algorithms. Therefore, the effectiveness of the classifiers is examined using fewer logging axes, lower sampling rates and determining the most relevant features. In this work also energy experiments are performed on an embedded system to analyze the energy consumption of the different models for classifying equine behavior. Then, further network parameter optimization is performed. A final energy prediction model obtained from the experimental energy measurements enables us to make a prediction using regression models without having to execute and measure the energy of a the optimized CNN or hybrid CNN. Finally, we envision the energy consumption of a monitoring system that sends out information on the horses' activities every 10 minutes.

To summarize, the novelty and major contributions of this paper are the following: for the first time (i) a performance assessment and experimental evaluation of energy consumption of three machine learning algorithms detecting seven distinct behaviors from six different horses is carried out simultaneously

with determination of the (ii) optimal sampling rate (5 Hz, 10 Hz, 12.5 Hz, 25 Hz and 50 Hz), (iii) necessary number of logging axes (ranging from one to six) and the (iv) ideal model parameter settings.

This study envisions a practical high-performance behavior detection system for equine activities and will provide useful information, i.e., the required input vector information and energy use, to develop wearables for horses to detect seven equine behaviors via the machine learning-based classification engine.

The remainder of this paper is organized as follows. Section 2 describes the accelerometer data used in this study, collection and preprocessing of the accelerometer data, an explanation of the used classifiers, a description of the energy consumption measurements and an overview of the model optimization process. Section 3 demonstrates the performance of the classifiers for axes reduction and undersampled data. For the latter, also the experimental results of the energy measurements on an embedded device are presented. This section concludes with a list of optimal parameter settings. Finally, discussion and conclusions of the findings in this study are presented in Section 4 and 5.

2. Proposed methodology

This research aims to provide an high performance approach for energy-efficient activity detection using accelerometer data and machine learning. This section describes the methodology for collecting accelerometer data from subjects, using machine learning classifiers with or without features, pre- and post-processing data, measuring the energy consumption of the algorithms and optimizing the algorithms. Figure 1 displays the

complete working for the proposed approach.

The overall approach proposed in this study is based on behavior detection with three different classifiers, namely: a Random Forest classifier, a Convolutional Neural Network and a hybrid CNN, i.e. a CNN fused with features that retain knowledge about the global time series form. This proposal includes horse data acquisition as the first step (Block 1 in Figure 1). For the data collection two Axivity AX3 off-the-shelf accelerometers are utilized (Axivity, 2015). Once recorded, the data is annotated and preprocessed (Block 2 in Figure 1). The CNN extracts features automatically from the raw data, while manual feature extraction is necessary for the RF classifier and thus different time and frequency domain features are extracted from preprocessed data (Block 3 in Figure 1). The next step is to reduce input signal dimensions (Block 4 in Figure 1). This helps to speed up classification, reduces computational complexity and also decreases the energy consumption of the algorithms. Therefore, three steps are taken, namely, selecting the most important features to use as input for the RF classifier, using fewer logging axes and reducing the sampling rate. This also helps to distinguish the most and least significant features and logging axes as well as the optimum sampling rate of the accelerometer signal when identifying activities. Before feeding the input signal to the three models, the weights of the classes are balanced and the data is split in a training and test set (Block 5 in Figure 1). In addition, the energy consumption of the three proposed algorithms is measured on the embedded platforms using different sampling rates. This enables to identify the best performing algorithm with the lowest energy consumption for the problem at hand (Block 6 in Figure 1). The last step is the further model

optimization to decrease the calculation cost by tuning the number of parameters of the CNN and hybrid CNN and number of trees of the RF (Block 7 in Figure 1).

2.1. Configuration: Horses and Activities

The present study makes use of data gathered during our previous published study (Eerdeken et al., 2020). Datasets were acquired with two Axivity AX3 accelerometers (size of 23 x 32.5 x 7.6 mm and weight of 11 g) fitted to the lateral side of the tendon boot of both the left and right front leg of six horses performing seven behaviors. An attachment convention for device orientation assists in consistent and comparable datasets being gathered. When the horse stands still, the positive x, y and z-direction point towards the ground, forward along the horse, and inwards the horse’s leg, respectively, as depicted in Figure 3. During the collection of the datasets, the horses were either ridden or longed for about 15 min each and moved with the four most common natural gaits (stand, walk, trot and canter). An exemplar exercise regimen is depicted in the Figure 2. Two horses performed also other movements like rolling, pawing and flank watching. Exercising is carried out at a local training arena with a size of 25 m x 38 m and a track surface of sand mixed with GEOPAT polyflakes. A minority of the data is gathered on a clinker brick underground. Observations on the activities of the horses were made with video recordings at the same time as data from the sensors is collected.

2.2. Description of accelerometer data

The accelerometers log data with configurable sampling rates varying from 12.5 Hz to 3200 Hz and are powered by a lithium-polymer battery of 150 mAh, which can be recharged via USB

Figure 2: Exemplar exercise regimen of horse 5 during the data collection.

Figure 3: Position and orientation (X, Y, and Z axes) of the right accelerometer

connection. This allows measurements at 12.5 Hz for 30 days and at 100 Hz for 14 days. Acceleration is measured on x-, y-, z-axes with a maximum sensitivity of $\pm 16g$ [$g = 9.81m/s^2$]. The AX3 has an integrated, real-time clock and calendar that provides the time base for the recorded acceleration data. The AX3 sensor configuration and setup for monitoring is done using the open source software AX3 OMGUI Configuration and Analysis Tool. Each AX3 was set to record at a sampling rate of 50 Hz with a range of $\pm 8g$ for all the datasets. The data is collected on a 512 MB built-in memory and transferred after recording in a Continuous Wave Accelerometer format to a computer via USB connection.

2.3. Preprocessing of accelerometer data

To fuse the accelerometer data of the two sensors, they are aligned according to their timestamps. Activities are manually annotated with the specialized video annotation software ELAN which is used by animal scientists for the video analysis and codification of images, by selecting the length of the segment where the behaviour is performed and typing the annotation (Brugman et al., 2004; Max Planck Institute for Psycholinguistics, The Language Archive, Nijmegen, The Netherlands, 2019; Liebal et al., 2013). Table 1 lists the considered activities in this study with their descriptive definitions. Then, the sensor data is separated into windows of 2 s since (Eerdeken et al., 2020) revealed that a 99% accuracy for detecting seven behaviors could be reached using this time window and this list with chunks of data is then shuffled preventing the model from overfitting. Afterwards, the data chunks are filtered so that the 2 s value vector stands for one activity at a time. Finally, the data is normalized between 0 and 1 per batch of measurements along every axis with the normalization function of the Scikit-learn Python library (Pedregosa et al., 2011). The experimental dataset is split into two disjoint sets: training and test data. Ten-fold validation with automatic split testing (66% train, 34% test) is used which means that the apportion of the data into training and test sets is done at random.

2.4. Dimension reduction of input signal

The final accelerometer dataset size has the following dimensions: $H \times W$ with H the height of the accelerometer signal (e.g. for a 2 s time interval measured at $f = 50$ Hz, $H = 2 \cdot f = 100$) and W the number of logging axes.

Observed activities	Description
Stand	The horse stands at least three legs without moving to another position.
Walk	Walk is a four-beat gait that follows this pattern of the legs: left hind leg, left front leg, right hind leg, right front leg, leaving three feet on the ground.
Trot	Trot is a two-beat diagonal gait where diagonal pairs of legs move forward simultaneously with a pause between each beat.
Canter	Canter is a three-way gait that starts with the hind leg, which continues to the front in a rolling motion.
Roll	The horse starts by lying down, rotating the body over its back, alternating from side to side, remaining parallel to the surface.
Paw	The horse scrapes a forelimb over the ground.
Flank	The horse is staring at its side or flank.
watching	

Table 1: Description of the observed activities (Sutton et al., 2013)

2.4.1. Feature extraction and selection

In this study, we will extract a set of 109 features from each 2 s segment of labelled sensor data. We used features that already demonstrated to be important for classifying animal activities using accelerometer data identified by (Benaissa et al., 2019a) and (Le Roux et al., 2017). Those features include thirteen time domain features namely, average (Average x, Average y, Average z), maximum (Max_x, Max_y, Max_z), minimum (Min_x, Min_y, Min_z), signal magnitude average (SMA), standard deviation (Std_x, Std_y, Std_z) and six frequency domain features namely, spectral energy (Spectral_energy_x, Spectral_energy_y, Spectral_energy_z) and spectral entropy (Spectral_entropy_x, Spectral_entropy_y, Spectral_entropy_z). Our analysis introduces 90 more features, the coordinates of the first five peaks with a minimum peak height of 10% of the signal’s highest value of the FFT, PSD and autocorrelation functions - in addition to the features used in the previous two studies. Those functions are used to study the temporal characteristics of the movements. The peak frequencies of the FFT and PSD signals are induced by horse movements’ repetitive nature. The peaks of the autocorrelation function indicate how self-similar the signal is over different delay times. The details of all features are listed in Table 2.

Feature selection is an essential step in processing any high-dimensional classification dataset. This step’s main objective is to reduce the dimensionality of data by eliminating unnecessary or non-discriminatory features and retaining only the relevant ones. It helps to effectively perform the classification stage, since less data needs to be processed. It not only results in computational efficiency, but also reduces algorithm energy consumption.

Feature	Description
Peaks	x and y of 5 first peaks of fft, psd and autocorrelation functions (90)
Average	Average value of X, Y and Z-axis (3)
Maximum	Maximum value of X, Y and Z-axis (3)
Minimum	Minimum value of X, Y and Z-axis (3)
Signal magnitude average	$SMA = \frac{\sum_{i=1}^N \sqrt{a_{xi}^2 + a_{yi}^2 + a_{zi}^2}}{N}$ (1)
Standard deviation	Standard deviation of X, Y and Z-axis (3)
Energy	Energy of X, Y and Z-axis (3)
Entropy	Entropy of X, Y and Z-axis (3)

Table 2: All 109 features extracted from each 2 s segment of labeled accelerometer data (N samples). The number between the brackets indicates the number of features.

The selection technique used in this work is feature importance. The important features are identified based on the gain in all the nodes of the trees of the RF classifier.

2.4.2. Logging axes

To take the effect of the W logging axes on the classification accuracy into account, existing axes from accelerometer data measured at 50 Hz were selected ranging from one to six logging axes (left and right tri-axial leg accelerometer data). In order to study the effect of a different number of accelerometer axes on the model performance, for classification with the RF classifier, the features presented in Section 2.4.1 were again calculated using one or two axes instead of three axes of the left and right accelerometer.

2.4.3. Resampling

For the effect of resampling (i.e. H changes) on the classification accuracy, existing one leg accelerometer data measured at 50 Hz was resampled to lower sampling rates of 5 Hz, 10 Hz, 12.5 Hz, 25 Hz. For classification with the RF classifier, the features presented in Section 2.4.1 were computed for each sampling rate. In order to adequately reproduce the signal it is periodi-

Behavior	Mean duration (s)
Walk	1.23
Trot	0.77
Canter	0.58
Roll	2.20
Paw	0.54
Flank watching	4.87

Table 3: Mean duration of behaviour for the participating horses.

cally resampled at a rate that is two times the behavior with the highest frequency that was occurring in the dataset. Therefore, the mean duration of each behaviour is taken in consideration. From the videofiles combined with the accelerometer data, the mean duration of each behavior as shown in the Table 3 was calculated by taking 10 samples of each horse according to their description in Table 1. The behavior with the shortest duration and thus the highest expected frequency is pawing. One cycle of pawing had a duration of 0.54 s and thus a frequency of 1.85 Hz. This means that we cannot sample under 3.7 Hz.

2.5. ML algorithms for behavior classification

Three detection algorithms are investigated in this work namely, a CNN, a hybrid CNN and an RF classifier. The CNN and hybrid CNN are a “blackbox” i.e., the result is dependent on how many samples were recognized without understanding the internal learned features for classifying each activity. The CNN and Hybrid CNN automatically learn features so the raw data can be used as input. But for the RF it is necessary to add a subset of manually calculated features. The CNN and Hybrid CNN can automatically learn features such that raw data can be used as input. But the RF requires using a subset of manually calculated features.

2.5.1. CNN

In this study, a Convolutional Neural Network (CNN) is used, which is a deep learning algorithm that takes in an input vector and outputs a vector of probability distributions over the number of predictable classes (i.e. stand, walk, trot, etc.). The convolution layer (CONV) uses filters, also known as kernels, that execute a convolution operation as it is scanning the input data with respect to its dimensions. Figure 4 shows the used multilayer convolutional network with two convolutional layers, which are followed by two fully connected layers. The output of the last fully-connected layer is fed to a 7-way softmax layer, containing the Softmax activation function to introduce non-linear properties to the neural network (Ronao and Cho, 2016). This layer produces a distribution over the seven class labels: stand, walk, trot, canter, roll, paw, flank watching. The first convolutional layer filters the $H \times W \times 1$ input accelerometer data with 64 kernels (M_1) of size 3×1 ($R_1 \times S_1$) and stride 1 (U_1). The stride indicates the number of steps by which the filter moves over the data. The L2 regularization technique is used in this layer with a weight decay coefficient of 0.01 (Mallouh et al., 2019). The first convolutional layer is followed by a zero-padding and max-pooling operation. The second convolutional layer takes as input the (pooled) output of the first convolutional layer and filters it with 16 kernels (M_2) of size 5×1 ($R_2 \times S_2$) and stride 1 (U_2). Both layers contain an activation layer using rectified linear units (ReLUs) and a dropout value of 0.55 is used (Srivastava et al., 2014). The dropout regularization technique works by randomly and temporarily deleting a node in the network during the training phase to inhibit the neurons from adjusting exceedingly well to the training data and thus

prevent the network from overfitting. The Adam optimizer is used for training the neural network through back propagation (Shirwaikar et al., 2019). Training is carried out for 400 epochs, with an early stop criterion for halting training if there is no increase in accuracy over the last 60 epochs (Ronao and Cho, 2016). The input vector of the CNN contains raw accelerometer data.

Figure 4: CNN architecture used in the analysis of horse behavior with $H \times W$ the size of the input feature map, the number of input feature maps N_1 of 1, filter size $R \times S$ of 3×1 - 5×1 , filters M of 64-16, stride U of 1 and pooling size of 2×1

2.5.2. Hybrid CNN

For the hybrid CNN the same convolutional network as depicted in Figure 4 is used but the max-pooling layer output is then flattened and fused with additional time domain features: mean, maximum, minimum, signal magnitude average and standard deviation.

2.5.3. Random forest

Random Forest is an easy-to-use, flexible machine learning algorithm. Random forests are a type of ensemble learning method, created by combining multiple decision trees trained on the training set. To classify a new instance from an input feature vector, the instance descends for each tree in the forest.

Each forest predicts a class which is called a vote. The random forest selects the class with the most votes. High accuracy results are known to be achieved, even without hyperparameter tuning (Yazici et al., 2018). Within RF the number of trees was set at 100. The input vector of the RF consists of a subset of selected features calculated from raw accelerometer data.

2.5.4. Class weight adjustment

Table 4 lists the number of instances for all horses, the total class proportions and the modified class weights. Class weight adjustment is done because representatives of some of the classes appear less frequently (i.e., roll, paw and flank-watching). Due to this, learning algorithms will be biased towards the majority group and risk to not perform well for the classification of the minority classes (Krawczyk, 2016). The cost function penalizes losses of certain classes more severely, which can help the model adapt better to the characteristics of a minority class. The adjusted class weights, $w_{classes}$, are calculated in equation (1) (Li et al., 2018):

$$w_{class,i} = \frac{n_{instances}}{n_{classes} * n_{instances,i}} \quad (1)$$

with $n_{instances}$ the total number of instances, $n_{classes}$ the number of classes and $n_{instances,i}$ the number of instances for each class. The walk behaviour is most present with a class proportion of 49.6% and therefore, this gets the smallest weight of 0.29. Paw is with only 20 instances the behaviour which is the least present in the dataset and therefore gets the highest class weight of 15.99.

2.6. Energy consumption measurement on embedded platform

Health-related horse behavior must be continuously tracked, which is inefficient in terms of energy consumption. Also, the

	Classes						
	Stand	Walk	Trot	Canter	Roll	Paw	Flank watching
Number of instances	197	1111	658	187	30	20	35
Class proportions(%)	8.8	49.6	29.4	8.4	1.3	0.9	1.6
Adjusted weights	1.62	0.29	0.49	1.71	10.66	15.99	9.13

Table 4: Movement class instances and proportions of the studied movements of all horses and adjusted class weights.

need to regularly recharge monitoring devices can cause user dissatisfaction. When considering deploying algorithms on embedded devices, the impact of sampling frequency and logging axes on the energy budget is important, and more precisely what trade-offs may be expected in terms of acceptable accuracy versus battery life. Also, increasing the sampling rate inevitably increases the energy required. For example, for the AX3 accelerometer, the difference between 12.5 Hz and 100 Hz will be around 30 days battery life versus 14 days (assuming a 150 mAh battery). Therefore, if the accuracy at or below 12.5 Hz is comparable to the performance of the model at 100 Hz, there is an obvious gain in using the lower sampling rate. This leads us to investigate the behavior recognition method’s energy consumption on embedded platforms. The effect on the energy consumption is measured when using a 5, 10, 12.5, 25, and 50 Hz sampling rate for the three machine learning models on a Jetson Nano embedded system using the 10-Watt power mode with a 1.4 GHz 64-bit quad-core CPU, 128-core NVIDIA GPU, and 4 GB RAM. The data intensive or computationally intensive part i.e. *execution of the algorithms* is computed on the GPU whereas the simple and complex instructions i.e. *feature calculation* are computed on the CPU in order to achieve speedup in execution time of the computer program executed on the embedded system (Qasaimeh et al., 2019). Hence, the average energy

consumption is the mean of the sum of the total CPU and GPU energy usage to extract features of and classify 761 input samples of 50 executions. Regarding the CNN, only the GPU energy consumption is considered as no features need to be calculated. Using the Tegrastats utility, available on NVIDIA Jetson platforms, the energy consumption of behavior recognition methods on embedded platforms was measured.

2.7. Further model optimization

After model validation and experimental energy measurements, models are further optimized to reduce their computational cost and therefore energy needs.

It is possible to estimate the energy consumption of any given CNN by a linear relationship, solely using the Multiply-Accumulate (MAC) counts (Rodrigues et al., 2018). The energy usage per sample (mJ) can be approximated as (2):

$$E_{CNN} = a \cdot MACs \quad (2)$$

with a the slope of the linear relationship. The number of MACs in a convolution layer are calculated in equation 3:

$$MACs = \frac{H - R + U}{U} \cdot \frac{W - S + T}{T} \cdot M \cdot N \cdot R \cdot S \quad (3)$$

with $R \times S$ the size of the filter, M the number of filters, $H \times W$ the size of the input feature map, N the number of input feature maps, U the vertical stride and T the horizontal stride. Hence, decreasing the number of MACs decreases the CNN’s energy consumption. Since H and W are optimized by adjusting the logging axes (Section 2.4.2) and resampling (Section 2.4.3), only R , S , U , T , M and N are tuned. Based on the energy consumption versus accuracy of different parameter settings, a Pareto front is generated. A Pareto front provides optimal solutions with the best choice for high accuracy versus low energy use.

3. Experiments and results

3.1. Performance analysis

Correct selection of the evaluation criteria is crucial for evaluating the merits of a model. In this work, the overall model accuracy and confusion matrices are considered for the model performance assessment (Ruuska et al., 2018). The overall model accuracy is the number of true positive instances of all behavioural classes divided by the total number of instances in the test set. Here, a true positive corresponds to the number of instances where the behaviour was correctly classified by the algorithm using video-observations as reference. The objective is to obtain a target accuracy of 99% or higher. Finally, confusion matrices are used to evaluate the designed model where the diagonal elements represent whether the predicted label of the test data is equal to the true label, while off-diagonal elements are those that are mislabelled by the classifier.

3.2. Feature importance

The main purpose of feature selection is to find a feature subset which is small in size but high in predictive accuracy. The features as described in Section 2.4.1 are calculated for one leg. Results for the left leg accelerometer data are presented as the performance of the models is higher than for the right leg (see Section 3.3.1). Random forest ranking of importance was conducted to systematically determine the usefulness and identify the most important features for discriminating different activities. This reduction in input vector size helps to decrease the energy consumption of the algorithm since fewer features need to be calculated and classified. The relative feature importance is calculated by cumulating node impurity decrease by splitting

Figure 5: Left axis: the feature importance of the fourteen most important features based on the random forest algorithm. The importance is ranked based on which of the features the most informative for the algorithm to make a decision. Right axis: the overall model accuracy of the random forest algorithm by adding extra features one by one.

at a certain feature across all the trees (Kong and Yu, 2018; Takahashi and Takahashi, 2019; Lee, 2019). The first fourteen most important features were selected to be included in the RF classification algorithm. Adding extra features did not lead to a further significant increase in recognition accuracy. With this subset of fourteen features a mean accuracy of about $98.92\% \pm 0.12$ was reached. Figure 5 shows that the signal magnitude average (SMA) has the most important role (feature importance of 0.054) in predicting the behaviours, followed by the standard deviation of the acceleration in the y-direction (Std_y), the spectral energy in the x-direction (Spectral_energy_x), the height of the first peak of the Fast Fourier Transformation of the acceleration in the x-direction (peaks_y_fft_1_ax), etc. Figure 3 shows the orientation of the x,y and z axes. Furthermore, Figure 5 shows the accuracy of ten-fold validation of the random forest model while adding extra features according to descending feature importance. An accuracy of $90.29\% \pm 0.04$ can already be reached when the only selected feature is the SMA. Adding 'Std_y' results in the largest increase in overall accuracy to $94.70\% \pm 0.08$. An acceptable accuracy ($> 95\%$) can already

Model	Acc. left leg (%)	Acc. Right leg (%)
CNN	99.30	98.29
Hybrid CNN	99.29	98.02
RF	98.92	96.98

Table 5: Performance of the three models for 50 Hz accelerometer data from the left and right leg.

be reached with a subset of the three first features 'SMA', 'Std.y' and 'Spectral_energy_x'. Figure 6 shows the confusion matrices when the features SMA, SMA + Std.y and SMA + Std.y + Spectral_energy_x are taken into account for classification. The behaviours 'roll' and 'paw' get misclassified more often (100% and 75%) when only the SMA is selected as feature but adding 'Spectral_energy_x' solves this misclassification problem.

If clustering of data points from the same behavioral event occurs, then higher classification accuracies can be reached since distinct decision boundaries can be defined by the classifier. This strong clustering of the behavioural events is highlighted in Figure 7. There is a clear clustering pattern of the gaits along the SMA axis: stand ($1.52 \text{ g} \pm 0.09$), walk ($2.49 \text{ g} \pm 0.27$), trot ($4.31 \text{ g} \pm 0.55$) and canter ($5.85 \text{ g} \pm 0.67$) but roll ($2.19 \text{ g} \pm 0.43$) and paw ($1.41 \text{ g} \pm 0.03$) show overlap with the SMA values of the gaits, while there is a clear distinction between the clustering areas if the Std.y values are taken into consideration.

3.3. Analysis of results

3.3.1. Logging axes

Table 5 lists the classifiers' performance when selecting left leg versus right leg accelerometer data.

All three models perform less well (99.17% versus 97.76%)

(a) SMA, accuracy = 90.54%

(b) SMA + Std.y, accuracy = 94.88%

(c) SMA + Std.y + Spectral.energy.x, accuracy = 96.72%

Figure 6: Normalized confusion matrices of the validation data for the Random Forest model validated on (a) the signal magnitude average (SMA), (b) SMA + standard deviation of the acceleration in y-direction (Std.y) and (c) SMA + Std.y + spectral energy in the x-direction (Spectral.energy.x) of 50 Hz accelerometer data from one leg

Figure 7: 2D scatterplot of mutually exclusive behavioural events extracted from the leg accelerometer using SMA and Std_y metric values.

when using right accelerometer data. RF misclassifies pawing and flank watching. In this dataset pawing was performed mostly with the right leg and is thus classified as walking because the movement of the leg is similar. As shown in Figure 8(a), the pawing movement is also visible in the non-pawing leg accelerometer data, as an increased signal fluctuation can be seen compared to when the horse stands still (Figure 8(b)). The CNN misclassifies mainly flank watching as stand when selecting the right leg details, because it is a head movement when the horse stands. Hybrid CNN misclassifies mainly flank watching but also paw. Therefore, the left leg is used as a baseline and the right leg information is added.

Figure 9 shows the mean performance of the three models for a time interval of 2 s accelerometer data with increasing number of axes taken into account.

When only one or two axes are taken into account, the hybrid CNN is performing better than CNN (97.57% vs 96.98%). If information from one axis is used, the CNN and hybrid CNN are best performing (94.88% and 95.53%) for information from the y-direction (forward direction along the horse) but the RF is best performing (97.36%) with information from the x-direction (direction towards the ground). For the CNN and hybrid CNN, the accuracy increases sharply (on average with 4% and 3.2%,

(a) Paw

(b) Stand

Figure 8: Typical accelerometer patterns of paw and stand in a 2 s window. The red, green and blue lines represent X,Y,Z signals from the left accelerometer.

Figure 9: Classification accuracy as a function of the accelerometer axes for 50 Hz accelerometer data for three machine learning models: a Random-Forest classifier, a CNN and a hybrid CNN. The labels annotated in red contain information of only one leg. The striped line indicates an accuracy of 99%. (XL and XR, YL and YR and ZL and ZR are x, y and z-axis accelerometer data of the left leg and right leg, respectively.)

respectively) when the number of axes is increased from one to two. The CNN reaches the highest mean accuracy of 99.74%

Figure 10: Classification accuracy as a function of the sampling rate for one leg accelerometer data for three machine learning models: a RandomForest classifier, a CNN and a hybrid CNN.

when four axes (XLYLZLYR) are taken into account. The RF classifier is more fluctuating due to a misclassification of a variety of classes. For all the axes, the RF classifier is least performing reaching an average accuracy of 97.68% versus 98.56% for the hybrid CNN and 98.34% for the CNN. Good results ($> 98.9\%$) are reached for the three models when information from one leg (XLYLZL) is taken into account. We advise to use one leg accelerometer data. Therefore, further model validation and energy measurement experiments are performed with one leg accelerometer data.

3.3.2. Sampling rate

Figure 10 shows the mean performance of the three models for a time interval of 2 s one leg accelerometer data with increasing sampling rate.

For the CNN and RF models, the accuracy decreases sharply (on average with 3.2% and 1.6%, respectively) when the sampling rate is decreased from 10 Hz to 5 Hz, while the average accuracy of the hybrid CNN only decreases slightly (on average with 0.46%). For a sampling rate of 5 Hz, the hybrid CNN performs the best, reaching an average accuracy of 99.13% while the

RF and CNN models only reach an average accuracy of 97.25% and 96.34%, respectively. Figure 11 shows the confusion matrices of the three models at this sampling rate. It is clear that this reduction in accuracy is mainly due to the misclassification of the classes 'trot' and 'canter'. The hybrid CNN reaches the highest mean accuracy of 99.59% at a sampling rate of 10 Hz but also the CNN and RF reach a peak at this sampling rate. When the sampling rate increases from 10 Hz to 12.5 Hz an average decrease in accuracy of 0.32% can be noticed for all the classifiers mainly due to more misclassification of the classes 'trot' and 'canter'. Increasing the sampling rate from 12.5 Hz to 50 Hz leads to an increase in accuracy of 0.73% for the RF classifier while the CNN and hybrid CNN performances decrease slightly with 0.17% and 0.13%, respectively. For all the sampling rates the RF classifier is least performing reaching an average accuracy of 98.3% versus 99.4% for the hybrid CNN and 98.8% for the CNN. We advise to use a sampling rate of 10 Hz to detect seven distinct behaviors of different horses.

Effect of downsampling on FFT, PSD and autocorrelation peaks.

Figure 12 shows an example of the FFT, PSD and autocorrelation function of canter at 50 Hz and 5 Hz, accompanied by the original accelerometer data . During the feature selection process it became clear that four peaks were most important, i.e. the y-value of the first fft peak of the acceleration in the x-direction, the y-value of the the first psd peak of the acceleration in the z-direction, the y-value of the fourth peak of the acceleration in the y-direction and the y-value of the second psd peak of the acceleration in the x-direction. During downsampling, information of the two latter peaks will be lost.

(a) CNN, accuracy = 96.71%

(b) Hybrid CNN, accuracy = 99.34%

(c) RF, accuracy = 97.11%

Figure 11: Confusion matrices of the validation data for the CNN, hybrid CNN and RF model validated on 5 Hz accelerometer data from one leg.

Figure 12: From top to bottom: one leg accelerometer data, fft, psd and autocorrelation function with peaks of canter at (a) 50 Hz and (b) 5 Hz.

3.3.3. Energy consumption

Figure 13 shows the average measured energy consumption per sample as function of the sample rate to analyze tri-axial leg accelerometer data with a length of 2 s.

The experimental findings show that the amount of energy consumed by the behavior recognition system using RF is more than 17.2 and 13.5 times higher than those of the CNN and hybrid CNN algorithm, respectively. This difference in energy consumption can be attributed to the calculation of computationally-intensive features such as the FFT. The energy consumption for the RF classifier remains more or less constant for each sampling rate with an average of 1.25 mJ/sample. However, the energy consumption of the CNN classifier E_{CNN} (mJ) is linearly depen-

Figure 13: Results of measurements of the average energy consumed per sample by the activity recognition methods on the embedded platform.

dent on the sampling rate as is presented in Equation 4:

$$E_{CNN} = 0.0036f - 0.0016 \quad (r^2 = 0.99) \quad (4)$$

with f the sampling rate (Hz) and r^2 is the square of the correlation coefficient. The CNN energy consumption is the lowest among all sampling rates with an average of 0.072 mJ/sample. The CNN decreases energy consumption on average by around 21.47% compared to the hybrid CNN-approach. This higher energy consumption of the hybrid CNN can be attributed to the extra calculation step of the time domain features. Very good results of the recognition accuracy ($> 98.8\%$) are reached for the three models when 10 Hz one leg accelerometer data is considered.

We advise to use one leg accelerometer data at a sampling rate of 10 Hz classified with the CNN or hybrid CNN. Therefore, further model optimization is performed with one leg accelerometer data at a sampling rate of 10 Hz for the CNN and hybrid CNN.

3.3.4. Further model optimization

To optimize the CNN and hybrid CNN, using the same network architecture, the network parameters shown in Equation 3

are tuned since a decrease in MACs leads to a decrease in the consumed network energy (Equation 2).

Since a linear relationship exists, the energy consumption of a new CNN can be predicted without having to execute and measure the energy. Based on Equation 2 and the experimental Equation 4, the energy consumption of the CNN E_{CNN} (mJ) based on the MACs can be predicted as:

$$E_{CNN} = 2.4 \cdot 10^{-7} \cdot MACs \quad (5)$$

Because only basic statistical time domain features are additionally introduced for the hybrid CNN, the discrepancy in energy consumption between the CNN and the hybrid CNN can be attributed to this relationship. At a sampling rate of 10 Hz this difference is $6.12 \mu\text{J}$ per sample classification.

Figure 14 shows the energy usage per sample for the CNN and hybrid CNN, the most important parameter settings to decrease the number of MACs and the model performance to analyse 2 s one leg accelerometer data at a sampling rate of 10 Hz. For each model, 23 parameter settings were tested with $M1$: 8-16-32-64, $M2$: 4-8-16, $R1$: 3-4-5-6, $R2$: 5-6-7, $S1$ and $S2$: 1-2-3 and $U1$ and $U2$: 1-2. The red line in the top left of Figure 14 represents the Pareto front for both models providing a set of solutions from which to find the best choice according to preferences (high accuracy vs low energy usage). The blue line represents the Pareto front for the hybrid CNN model. Table 6 lists the Pareto-optimal solutions.

If both models are considered, the Pareto-optimal solutions contain only the CNN models, since lower energy consumption is expected as no extra time domain features need to be calculated. The goal is to achieve 99% or higher target accuracy while

Figure 14: Energy usage per sample, parameters settings (with M_1 and M_2 the number of filters, R_1 and R_2 the filter heights and S_1 and S_2 the filter widths and the accuracy of the CNN and hybrid CNN to analyze a 2 s sample of 10 Hz one leg accelerometer data. The red line represents the Pareto front for both models and the blue line represents the Pareto front for the hybrid CNN model.

minimizing energy consumption. From the Pareto-optimal solutions, the model that complies with this target is the CNN with 25632 MACs reaching an accuracy of 99.61%. For this model an energy use of $6.15 \mu\text{J}$ per sample classification is expected which is 6 times lower than the original CNN with 102528 MACs which consumed $37.62 \mu\text{J}$ per sample classification. The Pareto-optimal solution for the hybrid CNN model reaches an accuracy of 99.08% with 6240 MACs consuming $7.61 \mu\text{J}$ using only 6% of the MACs of the original hybrid CNN. Since the difference in energy consumption per classification for the optimized hybrid CNN versus the optimized CNN is small ($1.46 \mu\text{J}$) and the hybrid CNN showed to be more stable at lower sampling rates as

Model	Energy usage per sample (μ J)	Accuracy (%)	MACs	$M1$	$M2$	$R1$	$R2$	$S1$	$S2$	$U1$	$U2$	$T1$	$T2$
CNN	1.00	96.19	4176	8	4	3	5	1	1	1	1	1	1
	1.69	98.95	7056	8	8	3	5	1	1	1	1	1	1
	6.15	99.61	25632	16	16	3	5	1	1	1	1	1	1
Hybrid CNN	7.61	99.08	6240	8	8	4	6	1	2	1	2	1	1
	7.96	99.47	7680	8	8	5	6	1	3	1	1	1	1
	7.89	99.34	7392	8	8	4	6	3	1	1	1	1	1
	7.12	98.69	4176	8	4	3	5	1	1	1	1	1	1
	7.81	99.21	7056	8	8	3	5	1	1	1	1	1	1
	30.73	99.59	102528	64	16	3	5	1	1	1	1	1	1

Table 6: Pareto-optimal solutions

discussed in Section 3.3.2, we advise to use the optimized hybrid CNN model with $M1 = M2 = 8$, $R1 = 4$, $R2 = 6$, $S1 = 1$, $S2 = 2$, $U1 = 1$, $U2 = 2$ and $T1 = T2 = 1$ to detect horses' behavior based on leg accelerometer data.

4. Discussion

Throughout this work, leg accelerometer measurements were performed to automatically identify behaviors of horses based on machine learning algorithms. We investigated the most important features (time and frequency domain), the influence of fewer accelerometer axes logged and lower sample rates on the performance and the energy consumption of the developed automatic classification system through experimental measurements. Finally, the network parameters were fine-tuned based on the information of a theoretical energy consumption model for the CNN and hybrid CNN deducted from the energy usage experiments. To summarize the results and advise, Table 7 shows the optimal experimental setup, average accuracy and energy usage per sample to analyze the data obtained by the sensor of the proposed method, i.e. the hybrid CNN.

First, we selected a subset of fourteen features calculated from three axes and the highest sampling rate (50 Hz). As ex-

Parameter	Value
Algorithm	Hybrid CNN
The size of the input vector	20 (2 s at 10 Hz)
The number of input channels	3
The number of feature maps	8-8
Filter size	$4 \times 1 - 6 \times 2$
Vertical Stride	1-2
Horizontal Stride	1-1
Pooling size	1×2
Activation function	ReLu and Softmax
Weight decay	0.01 (L2 regularization)
The probability of dropout	0.55
Maximum epochs	400
Optimization (back propagation)	Adam optimizer
Accuracy (%)	99.08
Energy usage per sample (μJ)	7.61

Table 7: Experimental setup, average accuracy and energy usage per sample of the proposed method.

pected, the SMA feature presented good classification ($90.29\% \pm 0.04$) performances mainly for the gaits (stand, walk, trot and canter) since one study found that one could determine gaits by means of individual acceleration value ranges (Burla et al., 2014). However, when only SMA was used for the classification, two behaviours were often misclassified: roll and paw.

Second, the hybrid CNN performed better than the other algorithms (CNN and RF). The hybrid CNN reached on average for all sampling rates an accuracy of 99.4% versus 98.3% for the RF and 98.8% for the CNN. The same results are found for the average over all the axes where the hybrid CNN reaches an accuracy of 98.56% versus 97.68% for the RF and 98.56% for the CNN. Similar results were found in a study concerning human activity recognition where an hybrid CNN outperforms the baselines by more than 10% (Ignatov, 2018).

Third, the number of accelerometer-logged axes was investigated. With two accelerometers operating concurrently (combining two legs), classification efficiency was marginally higher (+0.2 pp) compared to one accelerometer. This means that reducing the number of axes logged by the accelerometers not only minimizes energy consumption and data load, but also reduces the number of accelerometers needed. Moreover, the results of other axis configurations greater than three axes were usually comparable with the results of three axis configuration. A cow feeding, lying, and standing behavior study also showed that one or two axis combination resulted in higher accuracies than three axes (Benaissa et al., 2019b).

Fourth, the sampling rate was investigated. When comparing different sampling rates for the leg data, recognition accuracies for sampling rates from 5 to 10 Hz generally increased by

1.75 %. Such a considerable reduction of 50% in sampling rate could save the energy of the sensors and minimize the monitoring system’s storage load. In other words, an increase in classification performance up to 3.2% may come at the cost of reducing battery life by up to 50% (Khan et al., 2016). However, when 5 Hz was used, the hybrid CNN algorithm accuracy still exceeded 99%. The increase in the ability of machine learning algorithms to identify standing, walking, and lying behavior patterns was also noted when monitoring sheep behaviors at a sampling rate of 16 Hz versus 32 Hz (Walton et al., 2018).

Fifth, the energy consumption was measured and compared for the three models. The energy consumption to classify one sample of 2 s long of the CNN was lower than for the hybrid CNN and RF for all the sampling rates (0.072 mJ/sample versus 0.092 mJ/sample and 1.25 mJ/sample). The energy consumption of our proposed model is much lower (15111 times) than reported by other studies analyzing six human activity classes, i.e., walking, jogging, ascending and descending stairs, sitting, standing, and lying down (Jeong and Kim, 2019). In this study the CNN reaching accuracies of 94.66% consumes 115 mJ/sample of energy to analyze one sample at a sampling rate of 20 Hz of 2.25 s long. This is due to the higher complexity of this neural network since a linear relationship is found between the number of MACs and the CNN’s energy consumption. In literature, this linear relationship has a slightly different factor ($E_{CNN} = 1.32 \cdot 10^{-6} \cdot MACs$ versus $E_{CNN} = 2.4 \cdot 10^{-7} \cdot MACs$) (Rodrigues et al., 2018).

Moreover, the results show that model optimization can reduce the energy consumption of the CNN-based and hybrid CNN-based methods to classify a sample of 10 Hz with a length

of 2 s by approximately 84% and 83% while still maintaining high accuracies (>99%). Similar results were found in human activity detection literature, where only 10% of network parameters were required to reach the previous recognition rates (Jeong and Kim, 2019).

4.1. Envisioned energy consumption of embedded device

Energy efficiency is the key requirement to maximize a wearable’s lifetime. Most embedded devices are battery powered. Therefore, their battery life depends on the energy usage of the used hardware components and firmware. To develop an autonomous embedded system, it is necessary to model the energy consumption for different tasks. Each task consumes a quantity of energy for a period of time. The three main units of the embedded device are the sensor unit, the embedded processor and the radio module. For the envisioned device the use of a 600 mAh battery is proposed. We calculate the energy consumption for one day sensing, processing and transmitting the activity data every 10 minutes.

4.1.1. Sensor unit

To detect the horse’s leg acceleration, the *LIS3DH* ultra-low-power high-performance three-axis linear accelerometer is considered (STMicroelectronics, 2016). This accelerometer consumes 4 μA of current at a sampling rate of 10 Hz. For a day of sensing the accelerometer consumes 96 μAh of energy.

4.1.2. Embedded processor

The embedded processor is responsible for collecting the output signal from the sensor, processing and classifying data after acquisition, and transmitting it to the transceiver. The energy

consumption of this unit was measured on a Jetson Nano embedded system. The proposed hybrid CNN model consumes $7.61 \mu J$ of energy to classify a 2 s sample accelerometer data which means an energy consumption of $0.0014 \mu Ah$ for each classification with a 1.5 V battery. Therefore, 43200 classifications for a day consume $60.48 \mu Ah$ of energy.

4.1.3. Communicating Unit

Once data is processed, the information can be communicated to a remote access point. To optimize the device's consumed energy and have long communication range, Low Power Wide Area Network technology (LoRaWAN) using the *RN2483* transceiver is considered (Microchip, 2020). The LoRaWAN has different configuration parameters: the bandwidth (BW), the coding rate (CR) and the spreading factor (SF) (Bouguera et al., 2018). The combination of these parameters provides different energy values and transmission ranges. The BW is configured to 125 kHz value due to the LoRaWAN specifications. The coding rate (CR) is set to 4/5 which that every four useful bits are encoded by five transmission bits. The higher the spreading factor (SF), the higher the communication range but the more time is taken to send a packet and thus more energy is consumed for each transmission (Podevijn et al., 2018). For that, Semtech presents a *LoRa modem calculator tool* that allows quick evaluation of the on-air time transmission calculation, this tool is free of charge and could be downloaded from (Semtech, 2015). The SF is configured to 7 (best case, horse is near access point) and 12 (worst case, horse is far from access point). For a SF of 7 and 12 an energy consumption of $0.26 \mu Ah$ and $6.46 \mu Ah$ is calculated for each transmission, respectively. One transmission

Process	Energy consumption (μAh)
Gathering	96
Classifying	60.48
Transmitting (SF = 7/12)	37.44/930
Energy consumption/day	192/1086
Available energy	600000
Battery life (days)	3110/552

Table 8: Best (SF = 7) and worst (SF = 12) case energy consumption and battery life of embedded device for gathering, classifying and transmitting the data of one day of the proposed method using a 600 mAh 1.5 V battery.

every 10 minutes (144 transmissions in a day) result in an energy consumption of 37.44 μAh and 930 μAh for an SF of 7 and 12, respectively.

Table 8 shows the daily energy consumption of gathering, classifying and transmitting data for the two cases (SF = 7 and 12) together with the battery life. The battery life is the available battery energy divided by the daily energy consumption. In the best case, a battery life of more than 3000 days is expected which is 6 times higher than the worst case with a battery life of more than 500 days.

5. Conclusion

In this paper, we proposed an energy-efficient approach for a horse activity recognition which is needed for horse training and health monitoring. We compared the recognition rate of three machine learning algorithms using an experimental dataset with seven behaviors from six different types of horses and explored the energy consumption of activity recognition methods using deep learning on an embedded platform. The results obtained

demonstrate that the proposed hybrid CNN-based model outperforms baseline approaches and establishes state-of-the-art results at lower sampling rates, fewer logging axes, and fewer network parameters. It has the advantage of using fewer logging axes up to three and small sampling rates up to 5 Hz for 99% accuracy. The experiments conducted on an embedded platform to measure energy consumption showed that the proposed method uses 13.5 times less energy than the RF-based method when comparing only the energy consumed for activity recognition. Despite using only 6% of the MACs, the optimized hybrid CNN achieved similar results compared to the original. The proposed method consumes only $7.61 \mu J$ of energy to analyze one sample of 2 s measured at a sampling rate of 10 Hz reaching an accuracy of 99.08%. Finally, the energy consumption of an envisioned embedded device was calculated. Gathering, classifying and transmitting data consumes only in the worst case $1086 \mu Ah$ of energy for daytime sensor data analysis resulting in over 500 days of battery life for device with a 600 mAh 1.5 V battery capacity.

In future, this work can be extended in multiple ways. One addition is to measure various degrees of training intensity, e.g. low, moderate, and high. This can be achieved by detecting and incorporating other activities such as how many jumps a jumping horse has taken, how much collected work (piaffe, pirouette, passage, etc.) a dressage horse has done, etc. Future work may also include a detailed assessment of the specific link between activities and health issues by capturing more behaviors related to horses experiencing a colic episode or foaling like: kicking the abdomen, stretching, and trying to lie down. Furthermore, measurements of long-term activity could be performed to conclude

if a horse is agitated or depressed. In most cases, our suggested approach demonstrates superior potential as shown by the above experimental results. The findings of this paper are expected to help with the development of wearable horse activity recognition applications.

6. Acknowledgements

This work was executed within the imec.icon project Hoof-MATE, a research project bringing together academic researchers and industry partners. The Hoof-MATE project was co-financed by imec and received project support from Flanders Innovation Entrepreneurship (project nr. HBC.2018.0536).

References

- Axivity, 2015. Axivity ax3 accelerometer. https://axivity.com/files/resources/AX3_Data_Sheet.pdf. Accessed: 2020-03-16.
- Benaissa, S., Tuytens, F.A., Plets, D., Cattrysse, H., Martens, L., Vandaele, L., Joseph, W., Sonck, B., 2019a. Classification of ingestive-related cow behaviours using rumiwatch halter and neck-mounted accelerometers. *Applied animal behaviour science* 211, 9–16.
- Benaissa, S., Tuytens, F.A., Plets, D., De Pessemier, T., Troghe, J., Tanghe, E., Martens, L., Vandaele, L., Van Nuffel, A., Joseph, W., et al., 2019b. On the use of on-cow accelerometers for the classification of behaviours in dairy barns. *Research in veterinary science* 125, 425–433.

- Bouguera, T., Diouris, J.F., Chaillout, J.J., Jaouadi, R., Andrieux, G., 2018. Energy consumption model for sensor nodes based on lora and lorawan. *Sensors* 18, 2104.
- Brugman, H., Russel, A., Nijmegen, X., 2004. Annotating multimedia/multi-modal resources with elan., in: *LREC*.
- Burla, J.B., Ostertag, A., Westerath, H.S., Hillmann, E., 2014. Gait determination and activity measurement in horses using an accelerometer. *Computers and electronics in agriculture* 102, 127–133.
- Casale, P., Pujol, O., Radeva, P., 2011. Human activity recognition from accelerometer data using a wearable device, in: *Iberian Conference on Pattern Recognition and Image Analysis*, Springer. pp. 289–296.
- Eerdeken, A., Deruyck, M., Fontaine, J., Martens, L., De Poorter, E., Joseph, W., 2020. Automatic equine activity detection by convolutional neural networks using accelerometer data. *Computers and Electronics in Agriculture* 168, 105139.
- Giannetto, C., Bazzano, M., Marafioti, S., Bertolucci, C., Piccione, G., 2015. Monitoring of total locomotor activity in mares during the prepartum and postpartum period. *Journal of Veterinary Behavior: Clinical Applications and Research* 10, 427–432.
- Graf, P.M., Wilson, R.P., Qasem, L., Hackländer, K., Rosell, F., 2015. The use of acceleration to code for animal behaviours; a case study in free-ranging eurasian beavers *castor fiber*. *PloS one* 10.

- Hounslow, J., Brewster, L., Lear, K., Guttridge, T., Daly, R., Whitney, N., Gleiss, A., 2019. Assessing the effects of sampling frequency on behavioural classification of accelerometer data. *Journal of experimental marine biology and ecology* 512, 22–30.
- Ignatov, A., 2018. Real-time human activity recognition from accelerometer data using convolutional neural networks. *Applied Soft Computing* 62, 915–922.
- Jeong, C.Y., Kim, M., 2019. An energy-efficient method for human activity recognition with segment-level change detection and deep learning. *Sensors* 19, 3688.
- Khan, A., Hammerla, N., Mellor, S., Plötz, T., 2016. Optimising sampling rates for accelerometer-based human activity recognition. *Pattern Recognition Letters* 73, 33–40.
- Kong, Y., Yu, T., 2018. A deep neural network model using random forest to extract feature representation for gene expression data classification. *Scientific reports* 8, 1–9.
- Krawczyk, B., 2016. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence* 5, 221–232.
- Le Roux, S.P., Marias, J., Wolhuter, R., Niesler, T., 2017. Animal-borne behaviour classification for sheep (*dohne merino*) and rhinoceros (*ceratotherium simum* and *diceros bicornis*). *Animal Biotelemetry* 5, 25.
- Lee, M.H., 2019. Machine learning for understanding the relationship between the charge transport mobility and electronic

- energy levels for n-type organic field-effect transistors. *Advanced Electronic Materials* 5, 1900573.
- Lee, S., Annavaram, M., 2012. Wireless body area networks: Where does energy go?, in: 2012 IEEE International Symposium on Workload Characterization (IISWC), IEEE. pp. 25–35.
- Li, S., Song, W., Qin, H., Hao, A., 2018. Deep variance network: An iterative, improved cnn framework for unbalanced training datasets. *Pattern Recognition* 81, 294–308.
- Liebal, K., Waller, B.M., Slocombe, K.E., Burrows, A.M., 2013. *Primate communication: a multimodal approach*. Cambridge University Press.
- van Loon, J.P., Van Dierendonck, M.C., 2015. Monitoring acute equine visceral pain with the equine utrecht university scale for composite pain assessment (equus-compass) and the equine utrecht university scale for facial assessment of pain (equus-fap): a scale-construction study. *The Veterinary Journal* 206, 356–364.
- Mallouh, A.A., Qawaqneh, Z., Barkana, B.D., 2019. Utilizing cnns and transfer learning of pre-trained models for age range classification from unconstrained face images. *Image and Vision Computing* .
- Max Planck Institute for Psycholinguistics, The Language Archive, Nijmegen, The Netherlands, 2019. Elan. URL: <https://archive.mpi.nl/tla/elan>.
- Microchip, 2020. Rn2483. <http://ww1.microchip.com/downloads/en/DeviceDoc/>

RN2483-Data-Sheet-DS50002346E.pdf. Accessed: 2020-06-20.

Müller, R., Schrader, L., 2003. A new method to measure behavioural activity levels in dairy cows. *Applied Animal Behaviour Science* 83, 247–258.

Munsters, C.C., van Iwaarden, A., van Weeren, R., van Oldruitenborgh-Oosterbaan, M.M.S., 2014. Exercise testing in warmblood sport horses under field conditions. *The Veterinary Journal* 202, 11–19.

Nathan, R., Spiegel, O., Fortmann-Roe, S., Harel, R., Wikelski, M., Getz, W.M., 2012. Using tri-axial acceleration data to identify behavioral modes of free-ranging animals: general concepts and tools illustrated for griffon vultures. *Journal of Experimental Biology* 215, 986–996.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research* 12, 2825–2830.

Penning, P., 1983. A technique to record automatically some aspects of grazing and ruminating behaviour in sheep. *Grass and Forage Science* 38, 89–96.

Podevijn, N., Plets, D., Trogh, J., Martens, L., Suanet, P., Hendrikse, K., Joseph, W., 2018. Tdoa-based outdoor positioning with tracking algorithm in a public lora network. *Wireless Communications and Mobile Computing* 2018.

- Qasaimeh, M., Denolf, K., Lo, J., Vissers, K., Zambreno, J., Jones, P.H., 2019. Comparing energy efficiency of cpu, gpu and fpga implementations for vision kernels, in: 2019 IEEE International Conference on Embedded Software and Systems (ICESS), IEEE. pp. 1–8.
- Rodrigues, C.F., Riley, G., Luján, M., 2018. Synergy: An energy measurement and prediction framework for convolutional neural networks on jetson tx1, in: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA), The Steering Committee of The World Congress in Computer Science, Computer pp. 375–382.
- Ronao, C.A., Cho, S.B., 2016. Human activity recognition with smartphone sensors using deep learning neural networks. Expert systems with applications 59, 235–244.
- Rushen, J., Chapinal, N., De Passille, A., et al., 2012. Automated monitoring of behavioural-based animal welfare indicators. Animal Welfare-The UFAW Journal 21, 339.
- Ruuska, S., Hämäläinen, W., Kajava, S., Mughal, M., Matalainen, P., Mononen, J., 2018. Evaluation of the confusion matrix method in the validation of an automated system for measuring feeding behaviour of cattle. Behavioural processes 148, 56–62.
- Semtech, 2015. Sx1272 lora calculator. <https://sx1272-lora-calculator.software.informer.com/download/>. Accessed: 2020-06-20.
- Shirwaikar, R.D., Acharya, D., Makkithaya, K., Surulivelrajan, M., Srivastava, S., et al., 2019. Optimizing neural networks for

- medical data sets: A case study on neonatal apnea prediction. *Artificial intelligence in medicine* 98, 59–76.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1929–1958.
- STMicroelectronics, 2016. Lis3dh. <https://www.st.com/resource/en/datasheet/lis3dh.pdf>. Accessed: 2020-06-20.
- Sutton, G.A., Dahan, R., Turner, D., Paltiel, O., 2013. A behaviour-based pain scale for horses with acute colic: scale construction. *The Veterinary Journal* 196, 394–401.
- Takahashi, K., Takahashi, L., 2019. Creating machine learning-driven material recipes based on crystal structure. *The journal of physical chemistry letters* 10, 283–288.
- Walton, E., Casey, C., Mitsch, J., Vázquez-Diosdado, J.A., Yan, J., Dottorini, T., Ellis, K.A., Winterlich, A., Kaler, J., 2018. Evaluation of sampling frequency, window size and sensor position for classification of sheep behaviour. *Royal Society open science* 5, 171442.
- Wang, Y., Nickel, B., Rutishauser, M., Bryce, C.M., Williams, T.M., Elkaim, G., Wilmers, C.C., 2015. Movement, resting, and attack behaviors of wild pumas are revealed by tri-axial accelerometer measurements. *Movement ecology* 3, 2.
- Yazici, M., Basurra, S., Gaber, M., 2018. Edge machine learning: Enabling smart internet of things applications. *Big Data and Cognitive Computing* 2, 26.