IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.Doi Number

End-to-end horse gait classification in uncontrolled environments using inertial sensors

Mahaut Gérard[1,2,3,4], Sandrine Hanne-Poujade[4], Guillaume Dubois[4], Henry Chateau[3] and Neila Mezghani[1,2]

[1]Applied artificial intelligence Institut (I2A), TELUQ University, Canada
[2]LIO, CHUM research center, H2X0A9 Montréal, Canada
[3]Unité ACAP3-CIRALE, Ecole nationale vétérinaire d'Alfort, 94700 Maisons-Alfort, France
[4]LIM Group, 24300 Nontron, France

Corresponding author: Mahaut Gérard (e-mail: mahaut.gerard@vet-alfort.fr).

This work was supported in part by Agence Nationale de la Recherche et de la Technologie (ANRT) and by Région Nouvelle-Aquitaine.

ABSTRACT Locomotor injuries in horses are a major cause of underperformance and serious welfare issue. Veterinarians typically investigate horses' lameness through visual examination at separate gaits (walk, trot, gallop). To evaluate lameness objectively, Inertial Measurement Units (IMU) based systems have been developed. It is necessary to accurately identify the gait of each stride as vertical displacement symmetry is assessed at a defined gait, essentially trot. This study aimed to classify gaits into 6 classes and to assess the training sample size required to maximize the performance. Unlike previous methods, we used raw IMU data without manually preselecting specific signal segments. Seven sensors were strategically placed on the limbs, head, withers, and pelvis of horses. 1440 horses were used in our unsupervised model and the gait of 110 horses was labelled using IMU data for our supervised models. We divided the 6 gaits classification task into two subtasks: a four-gaits classification and a gallop-specific classification. In the first subtask, we compared the performance of a machine learning (XGBoost), a deep learning (LSTM) and a transfer learning (ENCOD-CNN) model, depending on the labelled training sample size. Our results show that the transfer learning approach outperformed the other models, achieving test accuracy of 91.9%. Our gallop classification task achieves 97.1% accuracy and the total pipeline reaches 91.2% accuracy. Beyond improving gait classification in a real clinical setting, this research demonstrates the potential of transfer learning for time-series datasets and provides a quantitative assessment of the required labeled sample size for effective implementation.

INDEX TERMS Deep learning, gait classification, horse, training sample size, transfer learning

I. INTRODUCTION

Locomotor injuries in horses have major impacts on their welfare, performance, also impacting the economy of their stakeholders [1], [2], [3]. Equine locomotion is commonly evaluated by visual observation of a veterinarian looking at the horse moving at different gaits (walk, trot, gallop), figure (straight line, circles), and on different types of ground (hard, soft). However, although widely used, the visual assessment of horses' lameness is not always consistently reliable [4]. To aid veterinarians in lameness assessment and objectify gait analysis, Inertial Measurement Unit (IMU) based systems have been developed with a common goal of being robust to a variety of horses [5], [6], [7]. These tools are particularly beneficial for detecting mild lameness, which is more challenging to identify than severe lameness [8]. They also are relevant as non-invasive tools for the longitudinal follow-up of horses and for injury prevention [9]. Though markerless video analysis tools are emerging as promising alternative, IMU sensors remain today a more reliable tool for accurately detecting symmetry indices [10], [11]. Timmerman *et al.* [12] suggest that there is a good inter-operator

reproducibility when repositioning these systems and repeating the same measurements at a succession of time intervals.

IMU-based systems for horses generate time-series data, often addressed in recent years with machine learning or deep learning methods. However, the number of horses used varies a lot from one study to the other: 6 horses were used in [13], [14], 65 in [15], 111 in [16], and 120 in [17]. To our knowledge, no rationale exists in the field to size the number of horses required to train machine learning or deep learning models with this kind of data.

Most of these systems compute symmetry indices of vertical displacements. However, these indices must be computed within a constant gait, necessitating precise identification of the gait for each stride. From symmetry indices, vertical displacement thresholds [18], [19], [20], [21] have been suggested to differentiate between sound and lame horses. Trot is the focus gait of these systems. However, other gaits, such as walk and gallop, have been shown to significantly influence locomotion variables [15], and some disorders are primarily detectable only at specific gaits [19]. As a result, multi-gait classification becomes essential to analyze and build indices for these additional gaits. Moreover, gait classification can serve as a foundational step toward achieving more advanced tasks, such as limb pose estimation [13].

In this study, we implemented deep learning and transfer learning approaches based on raw IMU signals, evaluating their performance as a function of the available labeled sample size.

II. RELATED WORKS

Applying neural network models on the same kind of IMU data, other works focused on ground type classification [16], gait recognition in humans [23], [24], [25], [26], [27] or behavior classification in animals [28]. Gait classification can be performed by rule-based methods such as the limb stance sequence [22]. However, it is not robust to horses with specific locomotion or types of lameness. Therefore, some works use machine learning models for gait classification.

Aggregated features can be built to assess horses' locomotion [14], [15], [16], [17], [29]. They typically are:

- temporal: stride duration or frequency, stance duration, suspension duration, duty factor (ratio of the stance duration over the stride duration)
- spatial: minimum, maximum, mean, median, standard deviation (std), variance, percentiles, kurtosis, skewness, Fast Fourier Transform (FFT) magnitude and phase
- spatio-temporal: diagonal dissociation, lateral dissociation

They were used as input of Support Vector Machine (SVM), Quadratic Discriminant Analysis (QDA), Decision Tree, or simple Neural Networks (NN).

More recently, raw IMUs data have been used as model's input [13], [17] along with Convolutional Neural Network (CNN) or Long Short Time Memory network (LSTM). Time-series datasets often lead to high-dimensionality data for which the LSTM model is known to be adapted [25], [27], [30]. Yet, Ismail Fawaz *et al.* [30] found that an additional data transformation phase was used in the most successful time-series classification algorithms. This phase maps the original time-series into a new feature domain. In addition, collecting large amounts of labeled data in real-word field applications is typically more challenging than gathering unlabeled data due to factors such as cost, time, and quality constraints. From these two observations, unsupervised feature extractor models based on autoencoder have been developed [31], [32]. Such autoencoder are trained to learn efficient feature representation from the base unlabeled dataset, before reconstructing the signal. Fatima *et al.* [33] showed that extracting features from IMU data for human gait recognition yield higher performance than other state-of-the-art methods.

Some authors highlighted the role of data augmentation in the robustness of the models [34], [35], [36]. Data augmentation is achieved by transformation of the existing data or by new samples generation [37], [38]. Transformation techniques, such as jittering, scaling, rotation, time warping, are less complex than generative methods [34] but are known to be efficient with time-series data [35], [36].

Others [13], [14], [29] focused on binary (walk, trot) gait classification, achieving up to 99% accuracy by training the model on data collected on horses moving on a treadmill [29]. Another work performed multi-class (walk, trot, right-gallop, left-gallop, pace, tolt, paso, trocha) [17] gait classification and achieved an accuracy of 97%. The models used are NN such as LSTM [13], [17], CNN [13], [14] or machine learning models such as SVM [14], [17], Discriminant Analysis [17], [22], [29]. However, all these studies rely on the manual selection of specific signal segments before gait detection, resulting in the signal being necessarily one of the defined gaits. This approach is not realistic in typical veterinary practice settings, where no operator is available to precisely mark the start and stop points of each region of interest. Additionally, horses in unfamiliar environments may experience fear or excitement, further complicating the accuracy of such manual segmentation.

The aim of this study is to propose a new approach to classify gaits (walk, trot, left-gallop, right-gallop, disunited gallop) from the entire signal by adding a class 'other' and with two subtasks: four-gaits classification (walk, trot, gallop, other) followed by gallop classification (left-gallop, right-gallop, disunited gallop) with aggregated features. The 'other' class contains horses' halt, kicks, shakes, gait transitions and other artifacts. It enables live use of gait classification in standard veterinary practice without manual action and therefore opens the way for extensive data collection. To leverage the constraint of possible few annotated data, an autoencoder approach is presented.

III. METHODOLOGY

A. DATA COLLECTION

A total of 1440 horses were examined either during standard veterinarian locomotor assessments at a clinic specialized in equine locomotion between 2021 and 2024 or during weekly checkups in elite horses stables between 2023 and 2024. Horses were presented in-hand at various gaits (walk, trot, gallop), figures (straight line, right and left circles, straight line after flexion), and on two types of ground-surfaces (hard and soft). The dataset gathers 8273 conditions (triplet {gait, figure, ground}). Data collection employed the EQUISYM system (Arioneo, LIM France, Nouvelle-Aquitaine, France) equipped with seven IMUs placed on the four cannons, head, withers, and pelvis (Fig. 1). A dedicated fixture system was used to ensure the correct positioning of the IMU at the attachment site, with trained operators handling its placement. Each IMU was sampled at 200 Hz, synchronized with the other six, and equipped with a three-dimensional accelerometer (measuring acceleration within a full-scale range of ±8 gravitational force equivalence) and a gyroscope (measuring angular velocity within a range of ±1000 degrees per second) in three dimensions. The entire raw time-series signals were used without any manual selection.

FIGURE 1. Horse equipped with the EQUISYM system.

B. DATA LABELING

Gait labeling was performed manually using the proximo-distal (vertical) axis of the gyroscopes on the four limbs for 110 horses across 1142 conditions. Samples of the time-series for walk, trot, gallop, and 'other' are shown in Fig. 2. Among the 110 horses, all had walk and trot conditions and 92 had at least one condition with gallop. Live videos were recorded during all the conditions to assist in the labeling process. The six classes consisted of five gait classes, namely walk, trot, right gallop, left gallop, and disunited gallop, as well as one additional class labeled 'other.' The later covers all non-constant gait signal parts. It resulted in 36500 seconds of IMU signals recorded at 200 Hz, categorized into 6 classes. The data were distributed into 18.6% walk, 48.1% trot, 4.6% left-hand gallop, 4.9% right-hand gallop, 0.6% disunited gallop and 23.2% other classes. The horse population from the labelled data was represented by gender (52 geldings, 32 females, 15 males, 11 unknown), age (ranging from 3 to 17 years old, and mean value of 9.5 ± 3.1), breed (52 Selle Français, 6 Origine Constatée, 6 Zangersheide, and 46 from 18 other breeds), and activity (59 in show jumping including 8 at international level,11 in dressage including 4 at international level, 11 in eventing including 3 at international level, 5 for leisure, and 24 from 5 other activities).

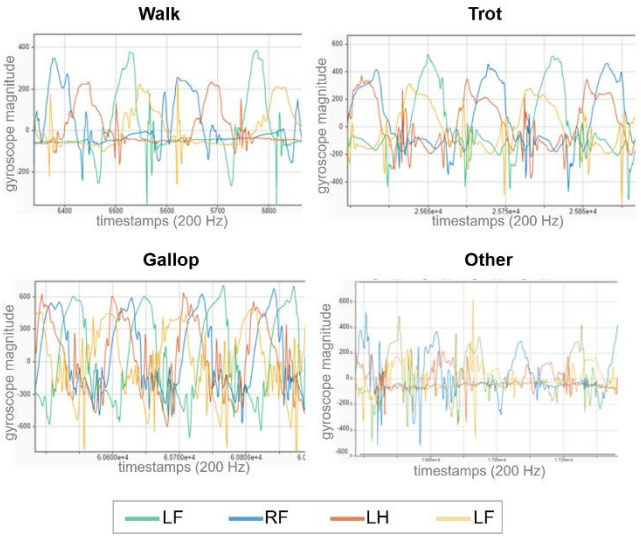

FIGURE 2. Walk, trot, gallop and 'other' samples of gyroscope dorso-ventral axis of each limb: left front limb (LF), right front limb (RF), left hindlimb (LH), right hindlimb (RH)

C. DATA PRE-PROCESSING

The time-series data from the sensors have 42 dimensions (7 sensors, 2 measures, 3 axis). The pre-processing involved the following steps:

1) Down-sampling: the time-series data were down-sampled to 100 Hz as recommended in [16]. This recommendation was based on a comparative study evaluating data quality across different sampling frequencies ranging from 10 to 200 Hz.

2) Standardization of each feature independently. The mean and standard deviation were taken from the training set. Standardization was preferred to minimum-maximum normalization as the data contain outliers due to the "other" class.

3) In the case of four-gaits classification (section E.1), data augmentation with time-warping (sigma=0.2 & knots=3) and scaling (sigma=0.01). These transformations were selected among basic methods for time-series augmentation [38]: jittering, axis rotation and random guided warping. Their parameters were set by fine-tuning.

4) Sequence length (*len_sequence*) was chosen to contain at least one stride of each gait. In the case of four-gaits classification, walk is the slowest gait and has an average stride duration at 100 Hz of 131 (±14) timestamps. In the case of gallop classification, only gallop is considered and has an average stride duration at 100 Hz of 59 (±4) timestamps. Sequences were created by overlapping sliding windows of

length *len_sequence*, every *k_train* timestamps. For four-gaits classification (section E.1), *len_sequence*=256 and *k_train*=10. For gallop classification (section E.2), *len_sequence*=64 and *k_train*=32. *k_train* values were selected with fine tuning.

5) To ensure continuous temporal signals, only sequences on one condition are retained.

D. FEATURE EXTRACTION AND SELECTION

Two different types of features were used as input of our models. First, raw features of size (*len_sequence, n_features*). Second, aggregated features of size (*n_features*n_metrics*). The aggregated features metrics, both temporal and spatial, are described in Table I. In the case of aggregated features, a final preprocessing step is added to perform the aggregation of raw signals using the defined metrics.

The data dimensionality was reduced to retain *n_features* and *n_metrics* with the following process. First, as correlation is often identified as a source of unreliable feature ranking [39], the feature set was filtered to keep only one feature in a set of correlated features [40]. A threshold of |0.7| was applied to identify strong correlations [41], with pairs of features exhibiting a correlation coefficient equal to or greater than this threshold considered correlated.

Features were then selected based on their contribution to classification model performance. Since we developed three models, namely a machine learning model (XGBoost), a deep learning model (LSTM), and a transfer learning model (ENCOD-CNN), we used feature selection method appropriate for each model. For aggregated features, the feature importance method of the XGBoost model was used.

For raw features, a forward feature selection process was applied using a standard CNN. The forward selection method was sequential [42] and described iteratively as follows: 1) the model was trained using one feature at a time; 2) the feature *y_best_1* associated with the best model performance was retained, and the model was then trained using a combination of this feature with each of the remaining features; 3) at step *i*, the feature set (*y_best_1* , ..., *y_best_i*) was retained, and the model was trained using a combination of this subset with each additional feature and 4) the process was stopped when the performance improvement when adding another feature fell within the 95% confidence interval of the 5-fold cross-validation.

The optimized CNN hyperparameters can vary with the feature set used, however we retained the optimized hyperparameter set found when training the models with all features for homogeneity reasons.

For consistency in the comparisons, the feature set retained for training of all models compared in each task was the concatenation of the optimized sets found for aggregated and raw features cases. It therefore combined accelerometer and gyroscope data from upper-body and limb sensors, the exact set is detailed in annex Table VIII for four-gaits classification task and in annex Table IX for gallop classification task.

TABLE I
AGGREGATED FEATURES DESCRIPTION

Feature	Description	Type
min	Minimum value of the window	Spatial
max	Maximum values of the window	Spatial
amplitude	*max-min*	Spatial
std	Standard deviation of the window	Spatial
mean	Mean value of the window	Spatial
median	Median value of the window	Spatial
FFT magnitude	Six first FFT coefficient magnitudes	Temporal
FFT phase	Six first FFT coefficient phases	Temporal
nb 0 crossings	Number of times the signal crosses 0 in the window	Temporal
mean stride frequency	Mean stride frequency in the window	Temporal
std stride frequency	Standard deviation of the stride frequency in the window	Temporal
shift between peaks	Shift between maximum peaks of two sensors (lateral limbs, diagonal limbs, front limbs, hindlimbs, withers & pelvis) in the window	Temporal
phase shift	Phase shift between signals (lateral limbs, diagonal limbs, withers & pelvis) of the window	Temporal

E. CLASSIFICATION MODELS

The six classes classification task was divided to account for the small proportion of gallop samples by splitting it into two subtasks (Fig. 3): a four-gaits (walk, trot, gallop, other)

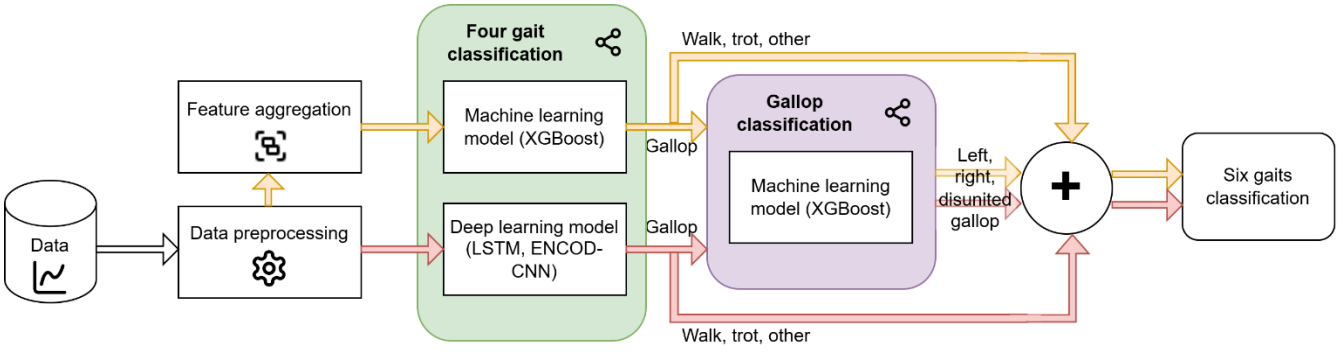

FIGURE 3. The six gaits task is divided into two consecutives subtasks: four-gaits classification and gallop classification. The machine learning model takes as input aggregated features (yellow path) whereas deep learning models takes raw features as input (red path).

classification task followed by a gallop classification task (right-hand, left-hand or disunited). At inference time, gallop classification is achieved for all samples classified as gallop. The classification models tested include a machine learning model (XGBoost) and two deep learning models (LSTM and ENCOD-CNN) with hyperparameters optimized through preliminary experiments. Hyperparameters not detailed in the model descriptions are either described for each experiment (optimized hyperparameter depending on the settings) or defaulted to library values. The seed was fixed, and random initialization was used for all neural networks.

1) FOUR-GAITS CLASSIFICATION

The models in this section classify the signal into 4 classes: walk, trot, gallop, and other. As a machine learning approach, XGBoost takes aggregated features as input. The LSTM network processes raw features as input and follows this architecture: 2 LSTM layers with batch normalization, tanh activation, and output size of 64 and 32 respectively. They are followed by a fully connected layer with batch normalization, ReLU activation, and an output size of 16. The output layer is a dense time-distributed layer with an output size of *n_classes*=4 and sigmoid activation. The return sequence parameter was set to True. It results in a network with 32692 trainable parameters.

The ENCOD-CNN network takes as input the raw features. It combines a feature extractor encoder which was trained as part of an autoencoder and a CNN for classification. The autoencoder gathers an encoder for feature extraction and a decoder for signal reconstruction. The encoder has 4 convolutional layers, ReLU activation, and maxpooling of size 2. Batch normalization was used in the first layer. The encoder maps the input data of size (*len_sequence*, *n_features*) to an extracted feature data of size (16, 4). It is followed by the decoder with 4 convolutional layers, ReLU activation, and upsampling of size 2. The kernel size was set to 3 and padding to 'same'. The decoder, with four convolutional layers, maps back the data to the output size of (*len_sequence*, *n_features*). The classifier CNN has the following architecture: 3 convolutional layers with batch normalization, ReLU activation, and output size of 64, 32, and 16 respectively. The first two layers have a maxpooling with size 2. It is followed by a flatten layer and a fully connected layer with sigmoid activation. Kernel size was set to 3, and padding to 'same'. The encoder layers were then used before the CNN network. Encoder layers were defined as non-trainable layers, whereas the classifier neural network was trained with the labelled dataset. It results in a network with 9044 trainable parameters.

2) GALLOP CLASSIFICATION

The goal of gallop classification is to differentiate between various types of gallop: right-hand, left-hand, and disunited gallop. The proposed classification model leverages an XGBoost algorithm, which uses aggregated features as input. The dataset used for gallop classification was imbalanced. To address this, class weights were set to 'balanced' in the XGBoost model, a common strategy for handling imbalanced datasets. This approach adjusts the weights inversely to the class frequencies in the input data, ensuring the model gives greater importance to the minority class. This approach ensures that the model adequately accounts for less frequent classes, resulting in improved overall performance and more accurate predictions across all classes.

F. DATA POST-PROCESSING

Data post-processing steps convert sequence predictions to timestamp predictions. It is built to have an equivalence between the accuracy seen on sequence classification, which is optimized by the classifier training, and the timestamp-wise classification accuracy. As displayed in Fig. 4, two types of data were obtained as output of our models: raw (LSTM) and aggregated (ENCOD-CNN and XGBoost) sequences. The post-processing steps were adapted for each of the output types. We define *k_test* the step between two consecutive sliding windows at inference time. With *k_test*<*len_sequence*, the sliding windows overlap. *k_test* was selected as a trade-off between reduced computation time and accuracy.

Raw outputs are of size (*len_sequence*, *n_classes*) and are the class prediction of the model for each timestamp in the sequence. As we use overlapping sliding windows at inference time, each timestamp potentially has several predictions. The final timestamp class is the result of majority voting, as it is commonly done in [43], [44]. We selected the step *k_test*=*len_sequence*//3=85 to have an average of 3 votes per timestamp, validated by fine-tuning.

Aggregated outputs are of size (*n_classes*) and are the class prediction of the model for the majority class of the sequence. At inference time, each input sequence results in an output class for the entire sequence. The predicted label is applied to all timestamps in the sequence. As we use overlapping sliding windows, each timestamp eventually has several predictions. The final timestamp class is the result of majority voting. We selected the step *k_test*=10 by fine-tuning.

Timestamps at the extremity of every sequence lack the context that middle timestamps have. A boundary of 10% of the sequence length was applied at the beginning and end of every sequence. The boundary represents the timestamps that are ignored in the majority voting. The 10% value was selected through fine-tuning.

G. MODEL TRAINING AND EVALUATION

All models were trained using the labelled dataset and 5-fold cross-validation on the training horses, while tested on fixed

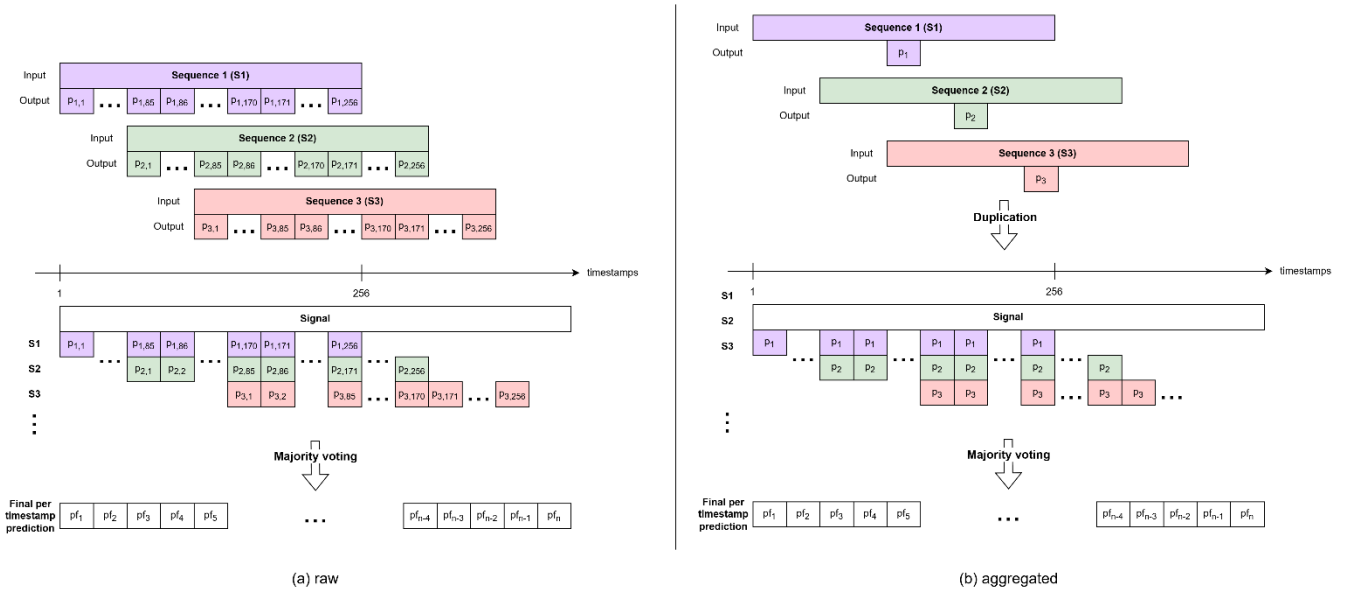

FIGURE 4. From sequence to timestamp classification. Post-processing steps for raw outputs (a) ($p_{i,j}$ being the predicted class in sequence i for timestamp j) and for aggregated outputs (p_i being the predicted class for sequence i).

set of horses chosen randomly before the first experiment. The number of training horses varied based on fixed sets built with a random sampling done once and for all before the first experiment. The sampling ensures that samples of the set with *m* horses are included in all sets with more horses.

Both tasks were therefore evaluated with the 5-fold cross-validation accuracy and standard deviation, the accuracy on the test sequences (before post-processing) and the accuracy on the test timestamps (after post-processing). Confusion matrices were used to evaluate class accuracy. The four-gaits classification task was also compared to a rule-based method based on limb stance sequence as described in *[22]*. Beginning with the left front limb, for instance, the walk foot-fall contact order is left front limb (LF), right hindlimb (RH) right front limb (RF), and left hindlimb (LH). Every defined gait (walk, trot, right gallop, left gallop, and disunited gallop) has a precise foot-fall contact order. Right, left, and disunited gallop are grouped in the same gallop class. All samples that did not fit a defined gait stance sequence were classified as 'other' in the rule-based method.

IV. RESULTS

The models were implemented in Python™ (version 3.11.5; Python Software Foundation) using scikit-learn (version 1.3.2) and keras (version 2.14.0) libraries. All models were trained using a fixed seed.

A. FOUR-GAITS CLASSIFICATION

A total of 40 horses were reserved for testing, while the number of training horses was incrementally increased in steps of 5, ranging from 10 to 70, to analyze the impact of adding more horses and to determine the minimum number required for a stable model. Upon completion of the preprocessing steps, the test set contains 66741 sequences, and the training set gathers between 30198 (10 horses for training) and 205102 (70 horses for training) sequences. The exact number of sequences in the training set for each number of training horses is described in Table VI (appendix).

All neural networks were trained using an Adam optimizer and a batch size of 1024. The autoencoder was trained on 10 epochs with a Mean Squared Error (MSE) loss and a learning rate of 1e-3. The LSTM and the ENCOD-CNN were trained on 100 epochs using a Mean Absolute Error (MAE) loss. The LSTM used a learning rate of 1e-4 whereas the ENCOD-CNN used a learning rate of 1e-3. The XGBoost model was trained with a learning rate of 1e-1 and a class weights set to 'balanced'. The performance of the classification models for each number of horses in the training set are gathered in Table II. In the following, we present the results of the testing set for timestamp-wise classification, as this is the final goal of our classification. Overall, the best classification accuracies were above 91.6% and achieved with the ENCOD-CNN trained with 55 or more horses, or with the LSTM trained with 65 horses. The XGBoost yielded the best accuracy (81.2%) for training with only 10 horses compared to the LSTM (63.5%) and the ENCOD-CNN (66.5%). The XGBoost model did not benefit from additional horses in the training set above 35 horses and reached an accuracy of 89.5%. The LSTM accuracy tended to improve with the number of horses in the training set. The best accuracy (91.6%) for this model was achieved with 65 horses. It was not stable for the other numbers of horses in the training set, whereas the ENCOD-CNN reached accuracies above 91% with 45 or more horses. With the ENCOD-CNN all accuracies were above 89.7% with 25 or more horses whereas the LSTM accuracy needed 55 or more horses to be above 89.5%

TABLE II

ACCURACY (ACC) FOR EACH NUMBER OF HORSES IN THE TRAINING SET AND FOR EACH CLASSIFICATION MODEL. ACCURACY IS DISPLAYED FOR 5-FOLD CROSS VALIDATION WITH STANDARD DEVIATION (STD), TEST SEQUENCES (TEST SEQ) AND TEST TIMESTAMPS (TEST T). BOLD VALUES REPRESENT FOR EACH MODEL THE BEST PERFORMING SETTING WITHIN CONFIDENCE INTERVALS OF TEST T ACCURACY.

Horse number	XGBoost			LSTM			ENCOD-CNN		
	5-cross-val Acc (std)	Acc test seq	Acc test t	5-cross-val Acc (std)	Acc test seq	Acc test t	5-cross-val Acc (std)	Acc test seq	Acc test t
10	91.7 (0.3)	80.7	81.2	83.2 (1.9)	62.4	63.5	87.5 (0.4)	65.7	66.5
15	91.4 (0.3)	87.5	87.5	87.4 (0.2)	80.2	80.9	88.5 (2.8)	84.9	85.7
20	90.9 (0.3)	88.3	88.3	88.4 (0.4)	85.5	86.6	88.9 (0.3)	88.5	88.9
25	90.8 (0.2)	88.6	88.6	89.0 (0.4)	87.8	88.6	89.2 (0.9)	90.2	90.2
30	90.7 (0.3)	89.1	89.1	89.3 (0.5)	89.7	90.5	89.9 (0.2)	88.7	89.7
35	90.7 (0.2)	89.3	**89.5**	89.6 (0.5)	89.1	89.8	89.8 (0.3)	90.6	90.8
40	91.0 (0.2)	89.1	**89.3**	90.0 (0.2)	87.9	88.6	90.1 (0.3)	88.7	89.8
45	91.1 (0.3)	89.3	**89.4**	89.7 (0.3)	88.1	88.5	90.3 (0.8)	91.3	91.1
50	90.8 (0.3)	89.2	**89.3**	89.8 (0.5)	88.3	88.9	90.2 (0.1)	90.9	91.3
55	90.8 (0.1)	89.4	**89.4**	90.0 (0.1)	89.0	89.5	90.1 (0.2)	92.1	**91.9**
60	90.8 (0.2)	89.2	**89.3**	90.2 (0.3)	89.3	89.6	90.5 (0.1)	91.9	**91.8**
65	90.8 (0.1)	89.3	**89.3**	90.4 (0.1)	91.0	**91.6**	90.6 (0.0)	91.6	**91.7**
70	90.9 (0.1)	89.4	**89.3**	90.4 (0.2)	90.3	90.8	90.7 (0.1)	91.7	**91.8**

TABLE III

CONFUSION MATRIX FOR THE ENCOD-CNN (A), THE LSTM (B), AND THE XGBOOST (C) TRAINED WITH 70 HORSES.

Predicted / *True*	Walk	Trot	Gallop	Other
Walk	209239 (96.7%)	112 (0.0%)	0 (0.0%)	7031 (3.3%)
Trot	3 (0.0%)	716700 (98.6%)	68 (0.0%)	10382 (1.4%)
Gallop	0 (0.0%)	360 (0.2%)	145347 (92.5%)	11371 (7.3%)
Other	39158 (12.4%)	36392 (11.5%)	10853 (3.4%)	229268 (72.6%)

(a) ENCOD-CNN

Predicted / *True*	Walk	Trot	Gallop	Other
Walk	191170 (90.4%)	14 (0.0%)	0 (0.0%)	20363 (9.6%)
Trot	14 (0.0%)	698034 (97.2%)	350 (0.0%)	20044 (2.8%)
Gallop	2 (0.0%)	185 (0.1%)	143193 (92.3%)	11707 (7.6%)
Other	36122 (12.1%)	22170 (7.5%)	16490 (5.5%)	223381 (74.9%)

(b) LSTM

Predicted / *True*	Walk	Trot	Gallop	Other
Walk	204027 (94.3%)	259 (0.1%)	0 (0.0%)	12096 (5.6%)
Trot	0 (0.0%)	684336 (94.1%)	4746 (0.7%)	38071 (5.2%)
Gallop	0 (0.0%)	6209 (4.0%)	145734 (92.8%)	5135 (3.2%)
Other	33179 (10.5%)	23782 (7.5%)	27890 (8.9%)	230820 (73.1%)

(c) XGBoost

TABLE IV

ACCURACY (ACC) DEPENDING ON THE NUMBER OF HORSES IN THE TRAINING SET FOR THE 5-CROSS VALIDATION WITH STANDARD DEVIATION (STD), THE TEST SEQUENCES (TEST SEQ) AND THE TEST TIMESTAMPS (TEST T).

Horse number	Acc (std) 5-cross-val	Acc test seq	Acc test t			
			Total	Left-gallop	Right-gallop	Disunited gallop
10	98.8 (1.0)	96.6	96.7	96.2	97.5	93.1
15	98.8 (0.6)	96.7	96.4	96.5	97.6	86.9
20	99.2 (0.3)	96.7	96.6	96.3	97.7	89.8
25	99.1 (0.2)	96.6	96.3	96.4	97.4	88.1
30	99.1 (0.3)	96.8	96.7	96.3	97.6	91.7
35	99.0 (0.3)	96.9	**96.8**	**96.7**	97.5	92.2
40	99.1 (0.3)	96.9	**96.9**	**96.7**	97.6	92.3
45	99.1 (0.3)	96.9	**97.0**	96.5	97.9	93.2
50	99.1 (0.07)	97.2	**97.1**	96.2	**98.2**	**94.0**
55	99.0 (0.2)	96.7	**96.8**	96.0	**98.0**	93.2

Confusion matrices for the three models trained with 70 horses are displayed in Table III. For all models, the accuracy of prediction of walk, trot and gallop classes were above 90%. However, the accuracy of prediction of the class 'other' was around 73%, the lowest achieved 72.6% with the ENCOD-CNN and the highest 74.9% with the LSTM. The prediction accuracy of trot was notably higher with the ENCOD-CNN (98.6%) or the LSTM (97.2%) than with the XGBoost (94.1%), whereas the prediction accuracy of walk was significantly higher with the ENCOD-CNN (96.7%) than with the LSTM (90.4%) and the XGBoost (94.3%). Gallop prediction accuracy, between 92.3% and 92.8%, was stable for all models. Focusing on out-of-diagonal values, the class 'other' gathered most of the confusion (last column and last row).

We compared our best model, the ENCOD-CNN, trained with 70 horses with a rule-based gait classification algorithm based on limb stance sequence. The comparison was done on our set of test horses. The ENCOD-CNN based pipeline achieved 91.7% accuracy while the rule-based method achieved 56.9% across all conditions. When only walk, trot and gallop classes are kept by removing the 'other' class, the ENCOD-CNN based method yielded 97.3% accuracy and the rule-based method 66.1% accuracy. In both cases, the machine learning method performed better than the rule-based method with a gain of +34.8% and +31.2%. We gathered in Fig. 5 the accuracy comparison for every condition type seen at least 3 times. For all condition types, the ENCOD-CNN based pipeline outperformed the rule-based method on average.

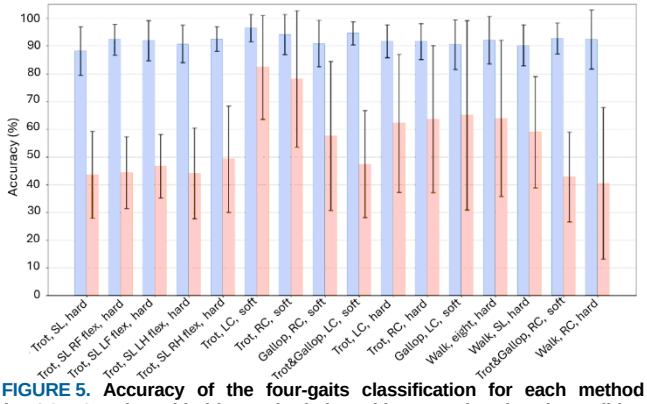

FIGURE 5. Accuracy of the four-gaits classification for each method (ENCOD-CNN based in blue and rule-based in orange) and each condition (seen at least 3 times) defined by a gait (WALK, TROT, GALLOP, TROT&GALLOP), a figure (SL for straight line, LC for left circle, RC for right circle, SL x flex for a straight line after the flexion of member x) and ground (SOFT, HARD). LF: left forelimb; RF: right forelimb; LH: left hindlimb; RH: right hindlimb.

B. GALLOP CLASSIFICATION

The test set represents the horses of the test set of the four-gaits classification that have at least one condition with gallop. The number of test horses is 36 and the number of train horses varied incrementally by a 5-step from 10 to 55. Upon completion of the preprocessing steps, the test set contains 4829 sequences, and the train set gathers between 1303 (10 horses for training) and 6283 (55 horses for training) sequences. The exact number of sequences contained in the training set for each number of horses used for training is described in Table VII (appendix).

The XGBoost model was trained with a learning rate of 1e-1 and class weights set to 'balanced'. The performances of the gallop classification are gathered in Table IV. For timestamp-wise classification, the best performance was achieved with 50 horses in the training set, with 97.1% accuracy. The accuracy was above 94% for all types of gallops with 96.2% for left-hand gallop, 98.2% for right-hand gallop and 94.0% for disunited gallop. In fact, we see in Table V that most of the confusions were made with the disunited gallop. The 5-fold cross validation accuracy was stable (range 98.8% to 99.3%) for all numbers of horses in the training set and the standard deviation below 1% in all cases. The test accuracy on timestamps was already 96.7% with 10 horses in the training set and achieved 96.8% with 55 horses. The prediction accuracy for disunited gallop stabilized over 91.7% with 30 and more horses in the training set. Prediction accuracies of the left and right gallop were stable for 10 and more horses in the training set: left-gallop prediction accuracy ranged between 96.0% and 96.7% and right-gallop prediction accuracy ranged between 97.4% and 98.2%.

The best model in terms of test accuracy on timestamps, and test accuracy on right and disunited gallop was achieved for 50 horses. We keep this model for the following.

The feature importance in XGBoost is presented in Fig. 6 for all features with importance greater than 1%. All these features are dorso-ventral gyroscope axis with time shift aggregation. The hindlimbs and forelimbs time shift are the two features with more than 30% accuracy (32.7% and 33.4% respectively).

TABLE V
CONFUSION MATRIX WITH NUMBER OF SAMPLES AND CLASS ACCURACY PERCENTAGE IN PARENTHESIS, FOR XGBOOST TRAINED WITH 50 HORSES IN THE TRAINING SET.

True \ *Predicted*	Left-gallop	Right-gallop	Disunited gallop
Left-gallop	63935 (96.2%)	609 (0.9%)	1917 (2.9%)
Right-gallop	143 (0.2%)	75401 (98.2%)	1252 (1.6%)
Disunited gallop	268 (2.7%)	330 (3.3%)	9357 (94.0%)

C. CLASSIFICATION PIPELINE

To achieve 6 classes classification, we combined the two subtasks described. All timestamps predicted as gallop in the four-gaits classification task become input of the gallop classification model. The best model of each subtask was kept, namely the ENCOD-CNN for the first and the XGBoost for the second, with the maximum number of horses in the training set. The pipeline achieved 91.2% accuracy, with prediction accuracy of 96.7% for walk, 97.2% for trot, 88.8% for left-hand gallop, 90.6% for right-hand gallop, 86.8% for disunited gallop, and 74.9% for the class 'other'.

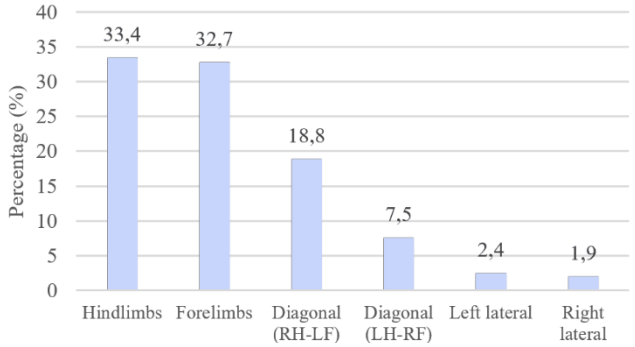

FIGURE 6. Feature importance for all features with importance > 1% in the XGBoost model, all from dorso-ventral gyroscope time shifts. LF: left forelimb; RF: right forelimb; LH: left hindlimb; RH: right hindlimb.

V. DISCUSSION

This study is the first to classify gaits into six distinct classes (walk, trot, left-hand gallop, right-hand gallop, disunited gallop, and other) without any manual preselection. It also evaluates the minimum number of horses with labelled gait needed to achieve robust classification. We classified gaits regardless of the ground, figure, or horse characteristics and in standard veterinary practice settings. We divided the 6 gaits classification task into two subtasks. First, four-gaits classification for which the ENCOD-CNN pipeline achieved the best accuracy for every number of horses in the training set. The performance is stable from 55 horses used at training time. Second, gallop classification with the XGBoost model achieving stable accuracy with 10 and more horses in the training set but more horses at training time improved the minority class (disunited gallop) prediction accuracy. Key features for classification were identified.

A. FOUR-GAITS CLASSIFICATION

The maximum difference between test accuracies on sequences and on timestamps is only 0.3%, confirming that the predicted accuracy on sequences remains consistent after the postprocessing steps. Therefore, the model optimization on sequences was efficient for the final goal of timestamp classification.

For all models, the standard deviation in the 5-fold cross-validation decreased with the number of horses. This highlights the individuals' locomotion diversity. Comparing the ENCOD-CNN and the rule-based method, standard deviations were higher for the rule-based method compared to the ENCOD-CNN pipeline. This emphasizes that the machine learning method is more robust to individuals' diversity than the rule-based method.

The ENCOD-CNN model is both our best performing model with up to 91.9% accuracy and the model that becomes robust with a smaller number of horses in the training set. The transfer learning approach, promising with a reduced number (12) of horses with gait labelled [45], proves to be the best model with more horses both in the training and in the testing sets.

We added the 'other' class to achieve automatic gait classification from signal gathered in an uncontrolled environment. None of the related works classified gaits with a possibility for a sample to be outside all defined gait classes. The multi-gait classification developed by [17] presents an accuracy of 97%. In comparison, our approach reaches a best accuracy of 91.9%. However, confusion matrices reveal that most misclassifications occur with the 'other' class. In fact, the challenge of this additional class is to gather very different samples (for instance, a halt has the opposite signal amplitude compared to kicks) Additionally, gait transitions can be mistakenly classified as specific gaits, especially given our goal of classifying gaits with a precision of 1/100th of a second. Another way to achieve the same goal would be to consider the 'other' class as a non-class by implementing an open-set classification model. It could be done by changing the activation of our output layer in our neural networks with an OpenMax activation [46]. Another approach could be to add a Variational Auto-Encoder (VAE) as an unknown-class detector to our closed-set classification pipeline [47], [48]. In fact, applying a threshold to the VAE reconstruction loss could be a good way to detect unknown class samples with the DTW distance adapted to time-series data [49], [50].

B. GALLOP CLASSIFICATION

The model achieves 97.1% accuracy with 50 horses in the training set and prediction accuracy of each class above 94%. The standard deviation was below 0.3% with 20 or more horses in the training set. This highlights that the training process is more robust with 20 or more individuals in the training set, resulting in 4 or more horses in every validation fold of the 5-fold cross-training.

To our knowledge, no previous work has classified the 3 gallop classes. We have identified a small set of features that are particularly useful for distinguishing between these gallop classes. Feature importance analysis highlights the effectiveness of the proximo-distal signal of the limbs gyroscope time shifts, as these features exhibit an importance greater than 1%. With symmetry, the time shifts between the forelimb and hindlimb are the two most important features. Forelimb (resp. hindlimb) time shift enables to differentiate between forelimb (resp. hindlimb) left and right gallop. Disunited gallop appears when forelimb and hindlimb gallops are different. Diagonal time shifts are the third (RH-LF with 18.8%) and fourth (LH-RF with 7.5%) most important features. As seen on Fig. 7, the diagonal RH-LF time shift is greater for left gallop (1st and last stance of each stride) than for right gallop (2nd and 3rd stance of each stride), and similarly for diagonal LH-RF. Finally, the lateral time shift has an importance of 2.4% for left gallop and 1.9% for right gallop. This feature seems particularly useful for distinguishing disunited gallop from left and right gallops.

C. CLASSIFICATION PIPELINE

For both aggregated and raw features, the selected sets were a combination of upper-body and limb sensors. The optimized set used for gait classification by [17] also gathered upper-body and limb sensors for raw features, but not for aggregated features. This can be attributed to the complexity of developing feature extraction algorithms, which risk being tailored to specific factors such as breed, surface type, or the age of the horses in the dataset. The features deemed important by the XGBoost model differed between our two subtasks. In the four gaits classification task, the model focused on spatial features (minimum, maximum, amplitude and standard deviation of the signal). In contrast, in the gallop classification task, the model concentrates on temporal features (time shift between limbs). Whereas completing ground classification, Parmentier *et al.* [16] found that using only limb sensors achieves the best performance across classifiers, which they linked to the fact that vibrations, differentiating grounds, are first absorbed by the limbs. All these observations emphasize that optimized feature sets vary depending on the task, even though the primary data is the same. With equal performances, this highlights the advantage of using raw features compared to aggregated features relying on specific algorithms.

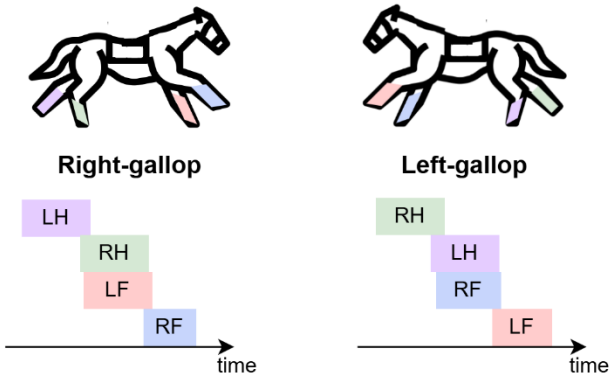

FIGURE 7. Limb stance order for right and left gallop inside one stride. LH: left hindlimb, RH: right hindlimb, LF: left forelimb, RF: right forelimb. Disunited gallop appears when the horse has different gallops on forelimbs and hindlimbs.

D. LIMITATIONS AND FUTURE WORK

Our postprocessing steps are based on majority voting, a common approach in time-series segmentation. The results demonstrate that it is a good way to maintain the test accuracy between sequence classification and timestamp classification. Therefore, the model optimization on the sequence is efficient for the final goal of time-series segmentation. Additionally, other postprocessing approaches could be implemented.

For aggregated outputs, we tested an alternative postprocessing pipeline that did not preserve classification accuracy across sequences and timestamps. In this approach, rather than predicting the majority class of the entire sequence, the model predicts the class for the middle timestamp of the sequence. The window provides context for the model to make this middle-timestamp prediction. At inference time, timestamps are predicted every *k_test* timestamp. To fill in the gaps between predicted timestamps, we applied linear interpolation to estimate the class for timestamps without direct predictions.

Other studies incorporate IMUs on the hoof, which we did not test in this study. The use of hoof-mounted IMUs is much more challenging in a clinical context due to the difficulty of attaching them quickly and securely to the hoof. In contrast, IMUs on the metacarpal region can be easily integrated into standard tack, such as boots. Concerning feature selection and extraction, the use of interpretable networks could help make an informed choice on the best feature set to use in a particular task or understand the feature extracted from the encoder.

Finally, we performed data augmentation with simple signal transformations. More recent methods, based on sample generation could be interesting to implement, especially when a small number of labelled samples are available. Some methods are developed primarily for image data [51], [52] and others are optimized for time-series data [53].

ACKNOWLEDGMENT

The authors sincerely thank the CIRALE team — including veterinarians, researchers, and technicians — for their crucial contribution to data collection.

REFERENCES

[1] S. Dyson, « Lameness and poor performance in the sport horse: Dressage, show jumping and horse trials », *J. Equine Vet. Sci.*, vol. 22, nº 4, Art. nº 4, 2002, doi: https://doi.org/10.1016/S0737-0806(02)70139-1.

[2] A. Egenvall *et al.*, « Days-lost to training and competition in relation to workload in 263 elite show-jumping horses in four European countries », *Prev. Vet. Med.*, vol. 112, nº 3, Art. nº 3, 2013, doi: https://doi.org/10.1016/j.prevetmed.2013.09.013.

[3] M. M. Sloet van Oldruitenborgh-Oosterbaan, W. GENZEL, et P. R. Van WEEREN, « A pilot study on factors influencing the career of Dutch sport horses », *Equine Vet. J.*, vol. 42, nº s38, Art. nº s38, 2010, doi: https://doi.org/10.1111/j.2042-3306.2010.00251.x.

[4] K. G. Keegan, « Reliability of equine visual lameness classification », *Vet. Rec.*, vol. 184, nº 2, Art. nº 2, 2019, doi: https://doi.org/10.1136/vr.k5366.

[5] J.-M. Denoix, « A Look at Lameness Through the Eyes of Functional Anatomy (and Biomechanics) », *AAEP Proc.*, vol. 67, 2021.

[6] P. R. van Weeren, T. Pfau, M. Rhodin, L. Roepstorff, F. Serra Bragança, et M. A. Weishaupt, « Do we have to redefine lameness in the era of quantitative gait analysis? », *Equine Vet. J.*, vol. 49, nº 5, Art. nº 5, 2017, doi: 10.1111/evj.12715.

[7] T. Pfau, A. Fiske-Jackson, et M. Rhodin, « Quantitative assessment of gait parameters in horses: Useful for aiding clinical decision making? », *Equine Vet. Educ.*, vol. 28, nº 4, p. 209-215, 2016, doi: 10.1111/eve.12372.

[8] S. D. Starke et M. Oosterlinck, « Reliability of equine visual lameness classification as a function of expertise, lameness severity and rater confidence », *Vet. Rec.*, vol. 184, nº 2, p. 63-63, 2019, doi: 10.1136/vr.105058.

[9] H. Darbandi, C. Munsters, J. Parmentier, et P. Havinga, « Detecting fatigue of sport horses with biomechanical gait features using inertial sensors », *PLOS ONE*, vol. 18, nº 4, p. e0284554, 2023, doi: 10.1371/journal.pone.0284554.

[10] T. Pfau *et al.*, « Comparing Inertial Measurement Units to Markerless Video Analysis for Movement Symmetry in Quarter Horses », *Sensors*, vol. 23, nº 20, Art. nº 20, 2023, doi: 10.3390/s23208414.

[11] F. J. Lawin *et al.*, « Is Markerless More or Less? Comparing a Smartphone Computer Vision Method for Equine Lameness Assessment to Multi-Camera Motion Capture », *Animals*, vol. 13, nº 3, Art. nº 3, 2023, doi: 10.3390/ani13030390.

[12] I. Timmerman *et al.*, « A Pilot Study on the Inter-Operator Reproducibility of a Wireless Sensors-Based System for Quantifying Gait Asymmetries in Horses », *Sensors*, vol. 22, nº 23, Art. nº 23, 2022, doi: 10.3390/s22239533.

[13] T. Yigit, F. Han, E. Rankins, J. Yi, K. H. McKeever, et K. Malinowski, « Wearable Inertial Sensor-Based Limb Lameness Detection and Pose Estimation for Horses », *IEEE Trans. Autom. Sci. Eng.*, vol. 19, nº 3, Art. nº 3, 2022, doi: 10.1109/TASE.2022.3157793.

[14] T. Yigit, F. Han, E. Rankins, J. Yi, K. McKeever, et K. Malinowski, « Wearable IMU-based Early Limb Lameness Detection for Horses using Multi-Layer Classifiers », in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, 2020, p. 955-960. doi: 10.1109/CASE48305.2020.9216873.

[15] M. Rhodin *et al.*, « Timing of Vertical Head, Withers and Pelvis Movements Relative to the Footfalls in Different Equine Gaits and Breeds », *Animals*, vol. 12, nº 21, Art. nº 21, 2022, doi: 10.3390/ani12213053.

[16] J. I. M. Parmentier, F. M. S. Bragança, E. Hernlund, et B. J. van der Zwaag, « Terrain Type Detection for Smart Equine Gait Analysis Systems Using Inertial Sensors and Machine Learning », in *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, 2023, p. 103-111. doi: 10.1109/DCOSS-IoT58021.2023.00029.

[17] F. M. Serra Bragança *et al.*, « Improving gait classification in horses by using inertial measurement unit (IMU) generated data and machine learning », *Sci. Rep.*, vol. 10, nº 1, Art. nº 1, 2020, doi: 10.1038/s41598-020-73215-9.

[18] C. Macaire *et al.*, « Asymmetry Thresholds Reflecting the Visual Assessment of Forelimb Lameness on Circles on a Hard Surface », *Animals*, vol. 13, nº 21, Art. nº 21, 2023, doi: 10.3390/ani13213319.

[19] C. Macaire *et al.*, « Investigation of Thresholds for Asymmetry Indices to Represent the Visual Assessment of Single Limb Lameness by Expert Veterinarians on Horses Trotting in a Straight Line », *Animals*, vol. 12, nº 24, Art. nº 24, 2022, doi: 10.3390/ani12243498.

[20] M. Rhodin, A. Egenvall, P. H. Andersen, et T. Pfau, « Head and Pelvic Movement Asymmetries at Trot in Riding Horses Perceived as Sound by Their Owner », *Equine Vet. J.*, vol. 47, nº S48, p. 10-11, 2015, doi: 10.1111/evj.12486_22.

[21] T. Pfau *et al.*, « Lungeing on hard and soft surfaces: Movement symmetry of trotting horses considered sound by their owners », *Equine Vet. J.*, vol. 48, nº 1, p. 83-89, 2016, doi: 10.1111/evj.12374.

[22] J. J. Robilliard, T. Pfau, et A. M. Wilson, « Gait characterisation and classification in horses », *J. Exp. Biol.*, vol. 210, nº 2, Art. nº 2, 2007, doi: 10.1242/jeb.02611.

[23] B. Hu, S. Li, Y. Chen, R. Kavi, et S. Coppola, « Applying deep neural networks and inertial measurement unit in recognizing irregular walking differences in the real world », *Appl. Ergon.*, vol. 96, p. 103414, 2021, doi: 10.1016/j.apergo.2021.103414.

[24] D. Kreuzer et M. Munz, « Deep Convolutional and LSTM Networks on Multi-Channel Time Series Data for Gait Phase Recognition », *Sensors*, vol. 21, nº 3, 2021, doi: 10.3390/s21030789.

[25] F. Sherratt, A. Plummer, et P. Iravani, « Understanding LSTM Network Behaviour of IMU-Based Locomotion Mode Recognition for Applications in Prostheses and Wearables », *Sensors*, vol. 21, nº 4, 2021, doi: 10.3390/s21041264.

[26] H. Prasanth *et al.*, « Wearable Sensor-Based Real-Time Gait Detection: A Systematic Review », *Sensors*, vol. 21, nº 8, Art. nº 8, 2021, doi: 10.3390/s21082727.

[27] K. Kluwak et T. Niżyński, « Gait Classification using LSTM Networks for Tagging System », in *2020 IEEE 15th International Conference of System of Systems Engineering (SoSE)*, 2020, p. 295-300. doi: 10.1109/SoSE50414.2020.9130487.

[28] H. Yu *et al.*, « An evaluation of machine learning classifiers for next-generation, continuous-ethogram smart trackers », *Mov. Ecol.*, vol. 9, nº 1, p. 15, 2021, doi: 10.1186/s40462-021-00245-x.

[29] C. Roepstorff *et al.*, « Reliable and clinically applicable gait event classification using upper body motion in walking and trotting horses », *J. Biomech.*,

vol. 114, p. 110146, 2021, doi: 10.1016/j.jbiomech.2020.110146.

[30] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, et P.-A. Muller, « Deep learning for time series classification: a review », *Data Min. Knowl. Discov.*, vol. 33, nº 4, p. 917-963, 2019, doi: 10.1007/s10618-019-00619-1.

[31] W. Yu, I. Y. Kim, et C. Mechefske, « Analysis of different RNN autoencoder variants for time series classification and machine prognostics », *Mech. Syst. Signal Process.*, vol. 149, p. 107322, 2021, doi: 10.1016/j.ymssp.2020.107322.

[32] Pankaj Malhotra, Vishnu TV, Lovekesh Vig, Puneet Agarwal, et Gautam Shrof, « TimeNet: pre-trained deep recurrent neural network for time series classification », *arxiv*, 2017.

[33] R. Fatima, M. H. Khan, M. A. Nisar, R. Doniec, M. S. Farid, et M. Grzegorzek, « A Systematic Evaluation of Feature Encoding Techniques for Gait Analysis Using Multimodal Sensory Data », *Sensors*, vol. 24, nº 1, 2024, doi: 10.3390/s24010075.

[34] B. K. Iwana et S. Uchida, « An empirical survey of data augmentation for time series classification with neural networks », *PLOS ONE*, vol. 16, nº 7, p. e0254841, 2021, doi: 10.1371/journal.pone.0254841.

[35] T. T. Um *et al.*, « Data Augmentation of Wearable Sensor Data for Parkinson's Disease Monitoring Using Convolutional Neural Networks », in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, in ICMI '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 216-220. doi: 10.1145/3136755.3136817.

[36] H. Uchitomi, X. Ming, C. Zhao, T. Ogata, et Y. Miyake, « Classification of mild Parkinson's disease: data augmentation of time-series gait data obtained via inertial measurement units », *Sci. Rep.*, vol. 13, nº 1, p. 12638, 2023, doi: 10.1038/s41598-023-39862-4.

[37] B. Hu, A. M. Simon, et L. Hargrove, « Deep Generative Models With Data Augmentation to Learn Robust Representations of Movement Intention for Powered Leg Prostheses », *IEEE Trans. Med. Robot. Bionics*, vol. 1, nº 4, p. 267-278, 2019, doi: 10.1109/TMRB.2019.2952148.

[38] G. Iglesias, E. Talavera, Á. González-Prieto, A. Mozo, et S. Gómez-Canaval, « Data Augmentation techniques in time series domain: a survey and taxonomy », *Neural Comput. Appl.*, vol. 35, nº 14, p. 10123-10145, 2023, doi: 10.1007/s00521-023-08459-3.

[39] L. Toloşi et T. Lengauer, « Classification with correlated features: unreliability of feature ranking and solutions », *Bioinformatics*, vol. 27, nº 14, p. 1986-1994, 2011, doi: 10.1093/bioinformatics/btr300.

[40] B. Venkatesh et J. Anuradha, « A Review of Feature Selection and Its Methods », *Cybern. Inf. Technol.*, vol. 19, nº 1, p. 3-26, 2019, doi: 10.2478/cait-2019-0001.

[41] P. Schober, C. Boer, et L. A. Schwarte, « Correlation Coefficients: Appropriate Use and Interpretation », *Anesth. Analg.*, vol. 126, nº 5, 2018, [En ligne]. Disponible sur: https://journals.lww.com/anesthesia-analgesia/fulltext/2018/05000/correlation_coefficients__appropriate_use_and.50.aspx

[42] G. Chandrashekar et F. Sahin, « A survey on feature selection methods », *40th-Year Commem. Issue*, vol. 40, nº 1, p. 16-28, 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[43] A. Jastrzebska, G. Nápoles, W. Homenda, et K. Vanhoof, « Fuzzy Cognitive Map-Driven Comprehensive Time-Series Classification », *IEEE Trans. Cybern.*, vol. 53, nº 2, p. 1348-1359, 2023, doi: 10.1109/TCYB.2021.3133597.

[44] M. Dallel, V. Havard, Y. Dupuis, et D. Baudry, « A Sliding Window Based Approach With Majority Voting for Online Human Action Recognition using Spatial Temporal Graph Convolutional Neural Networks », in *Proceedings of the 2022 7th International Conference on Machine Learning Technologies*, in ICMLT '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 155-163. doi: 10.1145/3529399.3529425.

[45] Mahaut Gérard, Guillaume Dubois, Sandrine Hanne-Poujade, Henry Chateau, et Neila Mezghani, « Automatic gait classification for equine lameness detection using transfer learning », *Multidiscip. Biomech. J.*, vol. Vol 1, 2024, doi: 10.46298/mbj.14521.

[46] Abhijit Bendale et Terrance E. Boult, « Towards Open Set Deep Networks », p. 1563-1572, 2016.

[47] M. T. H. Tonmoy, S. Mahmud, A. K. M. Mahbubur Rahman, M. Ashraful Amin, et A. A. Ali, « Hierarchical Self Attention Based Autoencoder for Open-Set Human Activity Recognition », in *Advances in Knowledge Discovery and Data Mining*, K. Karlapalem, H. Cheng, N. Ramakrishnan, R. K. Agrawal, P. K. Reddy, J. Srivastava, et T. Chakraborty, Éd., Cham: Springer International Publishing, 2021, p. 351-363.

[48] Xin Sun, Zhenning Yang, Chi Zhang, Keck-Voon Ling, et Guohao Peng, « Conditional gaussian distribution learning for open set recognition », présenté à IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, p. 13480-13489.

[49] A. Abanda, U. Mori, et J. A. Lozano, « A review on distance based time series classification », *Data Min. Knowl. Discov.*, vol. 33, nº 2, p. 378-412, 2019, doi: 10.1007/s10618-018-0596-4.

[50] T. Akar, T. Werner, V. K. Yalavarthi, et L. Schmidt-Thieme, « Open Set Recognition for Time Series Classification », in *Advances in Knowledge Discovery and Data Mining*, J. Gama, T. Li, Y. Yu,

E. Chen, Y. Zheng, et F. Teng, Éd., Cham: Springer International Publishing, 2022, p. 354-366.

[51] Irina Higgins *et al.*, « beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework », présenté à ICLR 2017, 2017.

[52] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, et Aaron Courville, « Improved Training of Wasserstein GANs », 2017, doi: 1704.00028.

[53] Z. Yang, Y. Li, et G. Zhou, « TS-GAN: Time-series GAN for Sensor-based Health Data Augmentation », *ACM Trans Comput Healthc.*, vol. 4, nº 2, 2023, doi: 10.1145/3583593.

APPENDIX

TABLE VI
NUMBER OF SEQUENCES FOR EACH NUMBER OF HORSES USED FOR TRAINING OF THE FOUR-GAITS CLASSIFICATION PIPELINE

Horse number	Sequence number
10	30198
15	41164
20	54742
25	71122
30	85140
35	105628
40	118528
45	130920
50	148266
55	161812
60	173638
65	188474
70	205102

TABLE VII
NUMBER OF SEQUENCES FOR EACH NUMBER OF HORSES USED FOR TRAINING OF THE FOUR GALLOP CLASSIFICATION PIPELINE

Horse number	Sequence number
10	1303
15	1882
20	2502
25	3262
30	3952
35	4198
40	4592
45	5032
50	5425
55	6283

TABLE VIII
FEATURE SET USED FOR FOUR-GAITS CLASSIFICATION, DESCRIBED BY THE SENSOR SET AND THE AGGREGATION METRICS WHEN APPLICABLE

Sensor set	Aggregation metrics
Accelerometer head dorso-ventral	Minimum Maximum Amplitude Standard deviation Time shift (when applicable)
Accelerometer withers dorso-ventral	
Accelerometer withers cranio-caudal	
Gyroscope withers cranio-caudal	
Accelerometer croup dorso-ventral	
Accelerometer croup medio-lateral	
Gyroscope croup cranio-caudal	
Gyroscope left forelimb dorso-ventral	
Gyroscope right forelimb dorso-ventral	
Accelerometer left hindlimb medio-lateral	
Accelerometer right hindlimb medio-lateral	

TABLE IX
FEATURE SET USED FOR GALLOP CLASSIFICATION, DESCRIBED BY THE SENSOR SET AND THE AGGREGATION METRICS WHEN APPLICABLE

Sensor set	Aggregation metrics
Accelerometer head dorso-ventral	Minimum Maximum Amplitude Standard deviation Time shift (when applicable)
Accelerometer withers dorso-ventral	
Gyroscope withers cranio-caudal	
Accelerometer croup dorso-ventral	
Gyroscope croup cranio-caudal	
Gyroscope left forelimb dorso-ventral	
Gyroscope right forelimb dorso-ventral	
Gyroscope left hindlimb dorso-ventral	
Gyroscope right hindlimb dorso-ventral	