

Ger J Exerc Sport Res 2023 · 53:187–195
<https://doi.org/10.1007/s12662-022-00866-3>
 Received: 24 February 2022
 Accepted: 4 November 2022
 Published online: 8 December 2022
 © The Author(s) 2022



Lucas Woltmann¹ · Claudio Hartmann¹ · Wolfgang Lehner¹ · Paul Rausch² · Katja Ferger²

¹ Dresden Database Research Group, Technische Universität Dresden, Dresden, Germany

² Institute of Sport Science, Justus Liebig University Giessen, Giessen, Germany

Sensor-based jump detection and classification with machine learning in trampoline gymnastics

Introduction

An exercise in trampoline sport consists of ten elements (jumps) and the order in which the elements are performed is at the discretion of the gymnast. Each jump consists of various twists and somersault rotations and their number and combination determine the difficulty of each jump. According to the regulations of the Fédération Internationale de Gymnastique, the degree of difficulty, execution, time of flight, and horizontal displacement scores define a final total value using of the following equation:

$$\begin{aligned} \text{Total value} = & DD + E(\text{max } 20\text{pts}) \\ & + ToF + HD(\text{max } 10\text{pts}) \\ & - \text{penalty deductions} \end{aligned}$$

DD = the degree of difficulty

E = execution

ToF = time of flight

HD = horizontal displacement

pts = points

For a long time, the total value in trampoline competitions consisted of two variables: the degree of difficulty and the overall skill execution. In order to make trampoline gymnastics more attractive and the evaluation of the gymnasts more

objective, the technical committee of the Fédération Internationale de Gymnastique established the time of flight as a new performance value in 2010 and the horizontal displacement value in 2017 (Feger, Helm, & Zentgraf, 2020). The increased efforts to objectify performance measurement in trampoline gymnastics through the introduction of the time of flight and horizontal displacement has led to the development of a new measurement system (called HDTS: horizontal displacement, time of flight, and synchronicity) and changes in the international scoring rules (Feger & Hackbarth, 2017; Feger, Hackbarth, Mylo, Müller, & Zentgraf, 2019; Feger et al., 2020). In addition to these objectively measurable parameters, two other parameters are currently being collected with the help of judges. The execution, and, thus, the quality of the movement (E), is evaluated by four judges, and the difficulty of elements (D) by one judge. The duty of the judge of difficulty is to check the elements and difficulty values entered on the competition cards. The difficulty of each element is calculated based on the number of twist and somersault rotations. Each $\frac{1}{4}$ somersault and each $\frac{1}{2}$ twist increases the difficulty score by 0.1. A fully completed somersault is given 0.5 of a difficulty point, a double somersault 1, a triple somersault 1.6, and a quadruple somersault 2.2 points. If a jump type has both somersault and twist rotations, the difficulty points are added up. Individual somersaults of 360–630° without twists,

with straight or piked execution, receive an additional 0.1 points. Multiple somersaults of 720° or more, with or without a twist, executed in a straight or piked position, receive an additional 0.1 points per somersault (FIG Executive Committee, 2016).

Therefore, this judge has to recognize the execution of ten jumps in rapid succession, which can be a very challenging task, for example, the distinction between “Full-in Full-out” (one twist in the first somersault, one twist in the second somersault—822) and “Half-in Rudy out Fliffis” (half-twist in the first somersault, one and a half-twists in the second somersault—813). Furthermore, athletes and coaches have to recognize the exact execution of and, even more important, deviations from these jumps during their training sessions. Currently, there is no reliable method to automatically recognize the various jump types.

However, current developments in sensor technology make it possible to measure complex whole-body movements as they occur in technical compositional sports in order to assist the evaluation of movement quality (Camomilla, Bergamini, Fantozzi, & Vannozi, 2018). Thus, in addition to technique and match analysis, one finds, among other things, a sensor-based recognition of flight elements in half-pipe snowboarding (Harding, Small, & James, 2007) as well as the use of IMUs for detection of training load and classification of gymnastic elements in trampoline gymnastics

The data and code that support the findings of this work are not publicly available due to privacy, legal, or ethical restrictions but are available from the corresponding author on reasonable request.

Table 1 Distribution and description of the elements based on the competition system, number of jumps recorded (Q), and difficulty of the element (D)

Element	Description	Quantity Q	Difficulty D
Somersault A	A back somersault performed in straight (a) position	138	0.6
Somersault B	A back somersault performed in piked (b) position	138	0.6
Somersault C	A back somersault performed in tucked (c) position	252	0.5
Double back B	A double back somersault performed in piked (b) position	100	1.1
Double back C	A double back somersault performed in tucked (c) position	105	1.0
Barani A	A front somersault, straight (a) with twist	124	0.6
Barani C	A front somersault, tucked (c) with twist	107	0.6
Fliffis B	A front half-out (double front somersault with ½ twist in the second somersault)—piked (b)	100	1.4
Fliffis C	A front half-out (double front somersault with ½ twist in the second somersault)—tucked (c)	100	1.3
Back full	A full-twisting, straight back somersault	114	1.3

(Campbell, Bradshaw, Ball, Hunter, & Spratford, 2021; Helten, Brock, Müller, & Seidel, 2011).

From a practical coaching perspective, the use of wearable inertial sensors or sensor-based systems should add value to everyday training and competition. The IMUs are particularly well-suited to monitor performance in a reactionless way, in real time, and without cumbersome set-up procedures, such as calibration (Baca, 2006; Chambers, Gabbett, Cole, & Beard, 2015; Hood, McBain, Portas, & Spears, 2012; Knight et al., 2007; Li et al., 2016; Mendes, Vieira, Pires, & Stevan, 2016). These sensors are capable of measuring physical quantities related to the movement of a body and their measurements can be used to estimate temporal, kinematic, and dynamic parameters. This means that the systematic, objective, and reliable monitoring and evaluation of performance can strengthen the link between research and practical coaching, particularly in high-performance sports (Camomilla et al., 2018). However, the use of sensor-based jump detection in training practice has been scarcely practicable up to now. Complex mathematical calculations for relatively simple movements and a large number of sensors prevent the systematic use of these methods in training and competition (Harding et al., 2007; Helten et al., 2011).

New opportunities have arisen to advance the automated classification of sensor-based data with the expansion of machine learning (ML) and data science into other areas of research. This also includes recent research at the intersection between biomechanics, mobile sensors, and ML. Current research offers

new possibilities in this area and indicates the potential for integrating ML systems for application in sports, particularly in the analysis of requirements for complex movements (Ancillao, Tedesco, Barton, & O'Flynn, 2018; Camomilla et al., 2018; Stetter, Krafft, Ringhof, Stein, & Sell, 2020). However, the use of ML methods for the detection of complex jump movements, such as those occurring in trampoline gymnastics, using mobile sensor data has been insufficiently explored up to now.

This paper presents the feasibility of the automatic classification of trampoline jumps using data from a data logger with an integrated IMU manufactured by 2D Datarecording (Datasheet 2d-Datarecording, 2021). We also discuss how to transform raw inertial data into meaningful characteristic features that underlie the detection of a jump and its assignment to a jump type. The duration of a jump is defined by jump limits from the acceleration data (ACC) which determine the start and end of the movement. The conditions for the execution of the respective jumps cannot be derived from the characteristic acceleration values alone; it is necessary to use the angular velocities of rotation of the gyroscope (gyro) about its three axes for this.

We tested eight different approaches to automatically classify jumps based on the sensor data. These are common ML techniques used in manifold applications, including sensor-based data (Huang & Perry, 2016; Meyer et al., 2019; Woltmann et al., 2022). The primary goal of the ML models is to automatically detect different types of jumps (somersaults,

twists, and combinations thereof) in the ongoing training of trampoline gymnastics. Another goal is to simultaneously detect and record the various conditions of execution (tucked, piked, straight) and, thereby, expand the analysis of individual techniques in the long term. Finally, the classification of elements and determination of the difficulty of an exercise in competition could be automated.

Methods

Participants

Four trampoline gymnasts (male, $n = 3$; female, $n = 1$) who are competing at the national level were recruited from local gymnastics clubs located in Bad Kreuznach and Frankfurt, Germany. Written informed consent was obtained from the participants in advance. This project was approved by the local Human Ethics Committee of the University of Giessen.

Procedure

Data were collected over the course of several training units at two national sports bases in Germany. Participants were instructed to wear a chest strap along with a logger (Fig. 1) and perform a warm-up of their choosing prior to data collection. The data logger with an integrated IMU (with 6° of freedom, triple-axis ACC up to 16g, triple-axis GYRO up to 2000°/s, sampling rate 1000 Hz; 2D Datarecording, Karlsruhe, Germany) was secured to the upper back (T2 vertebra) with a chest strap (Datasheet 2D-Datarecording, 2021). Three-dimensional kinetic data, which

are used to decode the jumps performed, were logged over the entire course of the training session. The contents of the training session were recorded in training logs. In addition, the IMU data collected were synchronized with existing video data (50 fps) in the WinARace 2021 software (2D Datarecording, Karlsruhe, Germany) to validate the information on the sequence of somersault and twist rotations or the technique. It was not possible to use the video camera (Digital Camera Exilim EX-F1, Casio, Tokyo, Japan) in all units. A total of 5927 jumps were recorded, of which 3932 were classified as straight jumps which are irrelevant for classification as they are used to gain height. Furthermore, we also excluded jumps that occurred fewer than three times in the entire dataset. This left us with 2076 jumps of 50 different jump types. Ten out of the 50 jump types represent all important and possible somersault and twist combinations, therefore, these were ultimately used. This amount of data is enough to train a variety of ML models, which is very important, especially for complex models that require a lot of training data in order to achieve stable results.

Acceleration data along the three axes of motion and angular velocities around the three axes of motion were collected for ten jumps (Table 1) with different modes of execution (tucked, piked, straight). The raw data were preprocessed to facilitate the reliable detection of any overlaps occurring between somersault and twist rotations. Jump limits for jump detection were determined from the acceleration data, based on the starts and ends of the jumps. The number of lateral and longitudinal body axis rotations involved in the motion was derived for the respective axes from the angular velocity data. Three different calculation variables for the angle subtended in the overall jump were determined for this purpose and used for further calculation. Consequently, a total of 45 datasets (features) are available from the ACC and the angular velocity data (GYRO) for the classification of an individual jump.

This number of values measured for each individual jump should allow a nuanced identification of the individual

Ger J Exerc Sport Res 2023 · 53:187–195 <https://doi.org/10.1007/s12662-022-00866-3>
© The Author(s) 2022

L. Woltmann · C. Hartmann · W. Lehner · P. Rausch · K. Ferger

Sensor-based jump detection and classification with machine learning in trampoline gymnastics

Abstract

The task of the judge of difficulty in trampoline gymnastics is to check the elements and difficulty values entered on the competition cards and the difficulty of each element according to a numeric system. To do this, the judge must count all somersaults and twists for each jump during a routine and thus record the difficulty of the routine. This assessment can be automated with the help of inertial measurement units (IMUs) and facilitate the judges' task during the competition. Currently, there is no known reliable method for the automated detection and recognition of the various elements to determine the difficulty of an exercise in trampoline gymnastics. Accordingly, a total of 2076 jumps and 50 different jump types were recorded over the course of several training sessions. In the first instance, 10 different jump types were used to train different machine learning (ML) models. Eight ML models were used for the automatic jump

classification. Supervised learning approaches include a naive classifier, deep feedforward neural network, convolutional neural network, k-nearest neighbors, Gaussian naive Bayes, support-vector classification, gradient boosting classifier, and stochastic gradient descent. When all classifiers were compared for accuracy, i.e., how many jumps were correctly detected by the ML model, the deep feedforward neural network and the convolutional neural network provided the best matches with 96.4 and 96.1%, respectively. The findings of this study will help to develop the automated classification of sensor-based data to support the judge and, simultaneously, for automated training logging.

Keywords

Trampoline · Training · Automated classification · Machine learning · Inertial measurement unit

jumps and their variants in terms of execution. Utilizing ML methods, both the preprocessed and the raw data are used to estimate the correct motion detection for a jump to facilitate an assessment as to whether all 45 values measured (features) are actually relevant for the classification of the jumps and if the selected subsections were chosen correctly when observing the jumps, or whether the raw data is ultimately more suitable.

Data engineering

In order to study and validate the reliability of the jump detection results, the measurement data that were preprocessed in the IMU using range filtering and online filtering were processed again using the 2D Datarecording Analyzer software.

The data already prefiltered in the IMU were filtered even further using infinite impulse response filtering with several adjustable filter frequencies to smooth the measurement data by removing noise. As the sensor co-ordinate system does not match the athlete's body coordinate sys-

tem when the sensor is attached to the athlete's back, the 3D rotation provided in the 2D Datarecording calculation algorithm can be used to correct the sensor coordinate system to the same orientation as the body coordinate system for additional data processing. This is done by applying a rotation ($+90^\circ$ | 0° | $+90^\circ$) to the IMU data for the x, y, and z axes.

Both the jump limits and the limit values for different intensities of movement are defined along the three axes of movement based on the acceleration data. The movement intensities are distinguished by different high limit values and the sensor first detects a jump relevant for the jump detection at an acceleration of $>70 \text{ m/s}^2$. The start and end points are determined by delimiting the individual jumps based on the previous calculation or preprocessing of the data. This allows the integration of the gyro signal or the calculation of the angles traversed from the angular velocity around the three axes

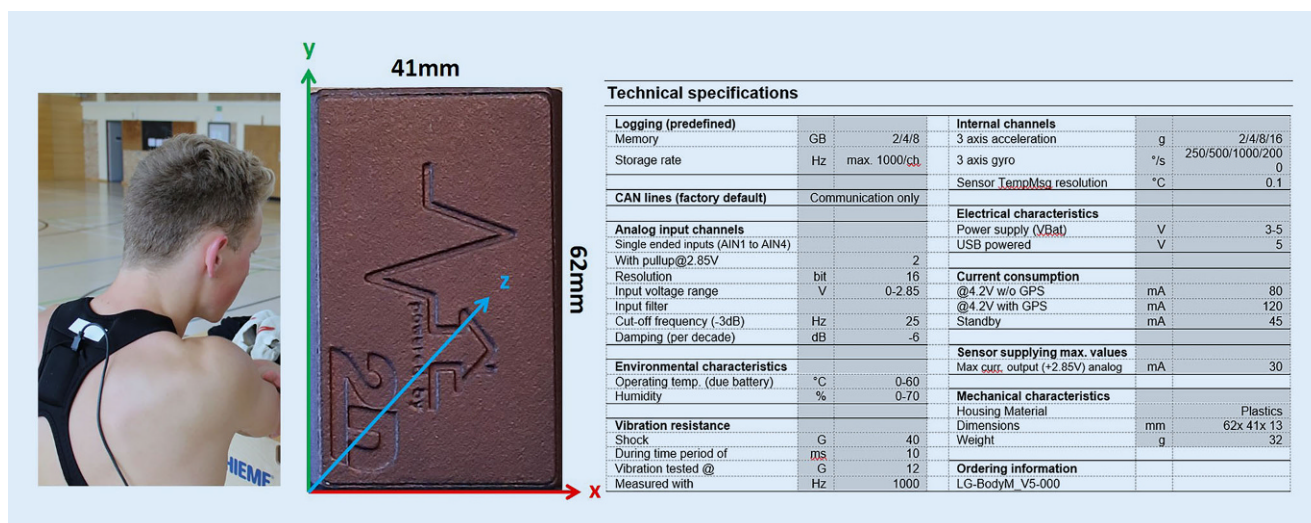


Fig. 1 ▲ Inertial measurement unit position for the upper back, configuration, and technical specifications

of movement within the limits of the respective jump ($Djump$).

$$Djump_{SigI...} = \int F(x) dx \quad (1)$$

$$Djump_{IAbs...} = \left| \int F(x) dx \right| \quad (2)$$

$$Djump_{AbsI...} = \int |F(x)| dx \quad (3)$$

When analyzing jumps with superimposed somersault and screw rotations, simple integration yields implausible results (Harding et al., 2007). This can be explained by the signs tied to the direction of rotation. With a simple forward somersault, i.e., a rotation around the frontal axis, the sensor records an angular velocity around the vertical (y) axis that remains constant within a certain range. This angular velocity is negative according to the body and the right-handed coordinate system and should reach a swept angle of approximately -360° integrated over the duration of the jump. This is because the integration result does not only contain the value of the swept angle but also directional information; the result is positive for backward rotations and negative for forward rotations. This directional sign then gives rise to implausible results when rotations are superimposed. The example of a Barani (Fig. 2) illustrates the problem.

In this case, the half-turn is performed after the first half of the somersault. This changes the direction of rotation of the somersault for both the body and the sensor. If the first half of the somersault

is still in the negative direction of rotation associated with the forward somersault, the second half of the somersault becomes positive in sign due to the half-twist. If the two sections are considered individually in an idealized execution, the integral would first result in a swept angle of -180° and then a swept angle of $+180^\circ$. The total swept angle would, thus, be 0° . In practice, the values reached are just above or just below 0° due to the different timing of the half screw in the jump. For this reason, an absolute of the respective gyro channel is formed for the duration of the jump over time prior to the integration of this channel. This removes all directional information and calculates the total swept angle without the forward and reverse rotations mathematically cancelling each other out. However, as the directional information is often helpful for a later classification of the types of jump, all three variants of the integration of the gyro channels are used (Fig. 3).

Following the integration of the gyro channels, the jumps are segmented based on the values calculated and the naming structures implemented. Based on these jump-specific values, the empirical limit values for each individual type of jump and its execution variant can be assigned to a specific category. The categories are jumps with rotations (JumpWithRot), jumps with twists (JumpWithTwist), and somersaults (JumpOnlySomersault).

We compared all models using different subsets of features to identify the most

important features and their influences on the models and determine the best segmentation. The first group of features is based on raw values, where each jump is divided into different percentage steps per dataset. Each percentage step incorporates the average and standard deviation for each feature within that particular percentage of a single jump. The models are trained on each feature set and used to identify the best length, expressed as a percentage, for a segment according to the models' best overall accuracy. The 5–20% segmentation is based on technical considerations and tested experimentally, whereas the 25% segmentation takes the jump phases of the elements into account (take-off phase, execution and opening phase, and landing). There are five raw value datasets with 5, 10, 20, and 25% steps, respectively. The second group of datasets also includes this segmentation and additional features derived and calculated from domain knowledge as presented in Eqs. 1, 2, and 3. Here, we used the same procedure to find the best model as for the datasets based on raw values.

A total of eight models on eight datasets (four from each group of feature sets) were evaluated, giving us 64 experiments. From this, we extracted the model with the highest degree of accuracy both for each individual group of datasets and for all the datasets as a whole. This was done by training all models using the same 80% from each

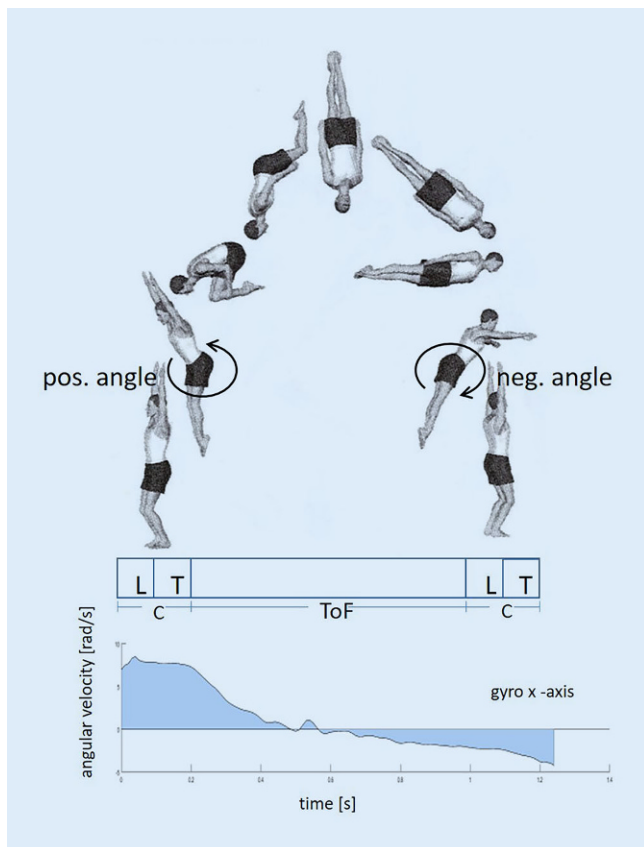


Fig. 2 ◀ Barani A (front somersault with twist-straight) with directional information from the sensor. TOF time of flight

dataset (training data) and then testing the model's accuracy on the remaining 20% (test data). All the models use open source implementations.

Another aspect of this work was to find the most important features for the ML models. The latter are those that are representative of a specific class of jump. These may differ from jump type to jump type. We used Shapley additive explanation (SHAP) values to assess the importance of each feature (Lundberg & Lee, 2017). These correlate the models' outputs to the input features by mutually disregarding features in subsets and comparing the models' outputs. The SHAP values can give insights into how influential a single feature is for a particular type of jump. This information is used to calculate the influence (expressed as a percentage) that each of the 45 features had on the model's output. We tested this for the best model to identify the most important phase and sensor measure for each type of jump.

Machine learning models

We evaluated a variety of eight ML models for the automatic jump classification. These models cover all the main approaches on classification models from the k-nearest neighbor classifier (KNN), over support-vector classification (SVC) and decision trees up to very complex neural networks (Table 2). The models can generally be divided into three complexity categories: naive, simple, and complex. These categories describe how expressive each model is and how much time and training data are required during model training. The naive category only contains the naive classifier (NC). The KNN, Gaussian naive Bayes (GNB), SVC, gradient boosting classifier (GBC), and stochastic gradient descent (SGD) are simple models. These are already capable of modelling complex contexts but still based on either simple mathematical principles or only a few calculations. The category of complex models contains the neural network-based deep feedforward neural network (DFN) and convolutional neural network (CNN). These two mod-

els are very strong in their ability to solve complex problems. However, their complexity requires more data and training time than models from the other two categories. All mentioned algorithms will be described in Table 2 with a short summary of their functionalities. Note that every jump is represented as a single vector, where each entry (feature) is either the mean, the standard deviation of a percentual step (raw data), or a derived feature (preprocessed data).

We will show that these models gain different levels of quality for the task of jump classification according to their complexity.

Additionally, SHAP values are used for the best model, i.e., the model with the best overall accuracy, for further analyses. Most models are opaque in their workings and, therefore, their decision-making is not traceable. However, one wants to be able to argue why a model has decided to classify a jump into a particular jump type. Therefore, the use of SHAP values can test different subsets of feature combinations and observe the differences in the model's output and quality. This results in a mapping for each jump containing the influences of every feature on the decision. With this mapping, we were able to recognize the most influential percentual phases and features for each decision according to the highest SHAP values. By analyzing every decision in that way, we can build a comprehensive understanding of the characteristics of a jump.

Results

All eight ML models were tested on the eight datasets, originating from the four different percentual splits and the use of raw or preprocessed data. This leads to 64 experiment values expressed in the final accuracies of each model on the test data specified. The detailed results can be found in Table 3. For brevity, we only present the data based on the best percentual split and the feature sets being used.

The best model is the DFN with just the raw data with 20% steps. It reaches an accuracy of 96.4% and, therefore, outperforms all other ML models by 0.3 to

Table 2 Overview of different models

Model		
Naive classifier	NC	Takes the probability distribution of jump types in the training data and randomly assigns labels to the test dataset according to this distribution. Therefore, the NC is independent of any features
K-nearest neighbors	KNN	Interprets each jump as a point in the high-dimensional vector space and classifies unknown jumps by assigning the most common jump type in the direct k-neighborhood of the unknown jump. The number of neighbors to consider (k) for this can be chosen to fit the ML task
Gaussian naive Bayes	GNB	Decides the affiliation of the jump to a jump type by the maximum likelihood over all features, which are modelled as independent Gaussian distributions
Support-vector classification	SVC	Takes the vector space from the KNN, but decides the jump types according to learned hyperplanes dividing the vector space into compartments
Gradient boosting classifier	GBC	Constructs a collection of decision trees that take the feature values for decision-making. Additionally, the GBC learns each new decision tree based on the solutions and errors of the previous tree (boosting)
Stochastic gradient descent	SGD	Takes the approximated gradient of the function the jump types create in the vector space and creates a separation of the vector space into compartments like SVC
Deep feedforward neural network	DFF	The simplest form of neural network. Fully connects several layers of neurons with each other and learns the weights along these connections. This is done in a completely self-contained way, making the DFF a black box approach
Convolutional neural network	CNN	Works similar to DFFs but uses 2D input. For each jump, the first axis is the percentual decomposition and the second axis contains the means, the standard deviations and the features derived for that particular percentual step of the jump

Table 3 The different models and the, respectively, best performing datasets

Model ^a	Percent step (%)	Mean (raw data)	Std (raw data)	Preprocessed data	Accuracy (%)	Training time (s)
NC	–	–	–	–	7.0	< 1
KNN	20	Yes	Yes	No	95.9	4
GNB	10	Yes	Yes	Yes	88.9	1
SVC	25	Yes	Yes	Yes	94.2	1
GBC	25	No	No	Yes	93.0	7
SGD	10	Yes	Yes	Yes	91.5	4
DFF	20	Yes	Yes	No	96.4	16
CNN	20	Yes	Yes	Yes	96.1	25

^aModels defined in [Table 2](#)

7.5% points. The CNN does not perform better, even though it should be able to model complex tasks better than the DFF. The higher complexity is also shown in the training times. Therefore, the DFF is preferable to the CNN.

From this conclusion, we want to analyze the influences of the features on the prediction. Accordingly, we calculated the SHAP values on the best model, i.e., the DFF from [Table 2](#). [Figure 4](#) details a single decision for a back tucked somersault (somersault C). The chart illustrates the influence of each feature on the decision of the model to say ‘somersault C’. Each bar shows how much the feature contributes to the model towards this decision. The grey values next to the feature names are the actual measurements. The features ‘20_mean_Gyro_y_R’ and ‘20_mean_Gyro_y_Fil’ have the highest

impact. These are the measurements for the rotation around the y-axis during the phase from 20 to 40% of the jump. Therefore, we argue that the rotation around the y-axis in the phase from 20 to 40% is the most expressive part of the jump type ‘somersault C’ not only for the model but also in general. This was tested for several other jumps and the SHAP values were able to identify several similar arguments, for example, the combined rotation around the y-axis in the first 20% and the z-axis in the first 40% is the most expressive part of a back full for the ML model. With this feature analysis and the averaged representative jump, there is a lot of potential for the application of the models in training and competitive scenarios. These applications are discussed in the following section.

Discussion

This article has presented and discussed eight different ML models for the classification of trampoline jumps. Based on raw inertial data from an inertial sensor, ACC along the three axes of motion and angular velocities (GYRO) around the three axes of motion were collected for selected jumps ([Table 1](#)). Utilizing the approach for automated jump recognition presented herein, the training process can be augmented regarding the achievement of a target value of skills and skill combinations (e.g., difficulty value of a freestyle or compulsory exercise in trampoline gymnastics) and related to the qualitative development of the performance of motor tasks with additional parameters, such as the application of force (horizontal displacement, time of flight, and synchronicity) and quality of execution. It shows that the systematic, objective, and reliable monitoring and evaluation of performance using IMUs can strengthen the link between research and practical coaching, particularly in high-performance sports (Camomilla et al., 2018). Furthermore, the accuracy of the DFF model of 96.4% shows that the use of sensors for automated jump detection is rewarding despite complex mathematical calculations (Harding et al., 2007; Helten et al., 2011).

With our experiments, we have demonstrated that the DFF works best

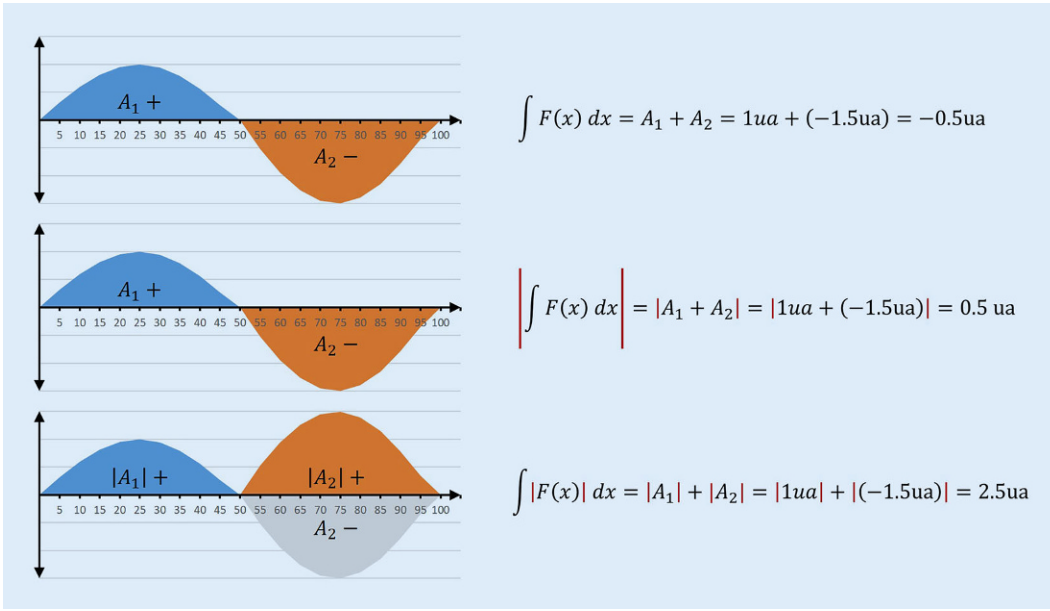


Fig. 3 ◀ Integration of the gyro signal of the angles traversed from the angular velocity around the three axes of movement. A_1 1 ua, A_2 1.5 ua, ua units of area

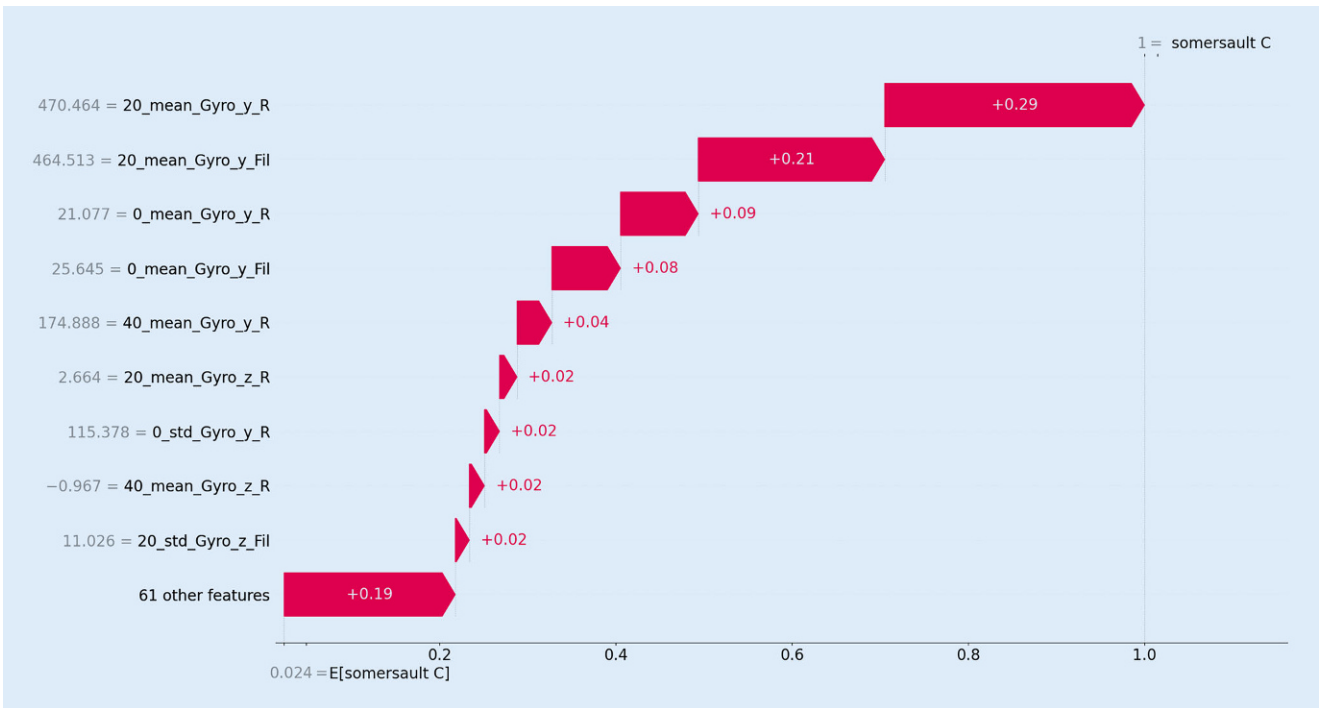


Fig. 4 ▲ The influences of each measurement on the decision for a back tucked somersault (somersault C) according to their Shapley additive explanation (SHAP) values

with the data based on raw values and that the CNN works best with the domain knowledge included. Additionally, this work also demonstrates that using the raw data and a DFF is slightly advantageous for the use case overall. However, there is no general recommendation for the use of a particular calculated feature set because each model requires its own adapted feature space. The NC performs

badly, because, as expected, it does not incorporate any knowledge about the data except the a priori distribution of the jump types. Furthermore, we cannot recommend a specific model per se. On the one hand, the DFF performs best, but, on the other hand, the SVC has the shortest training time while having an acceptable accuracy. However, all models except the DFF require prepro-

cessed data, which adds computational complexity to the data processing. Therefore, the application of a specific model depends on the particular use case.

Another interesting conclusion is the fact that the KNN performs well. Therefore, we can assume similarities of jumps of a certain jump type in their vector representation and raw measurements. This leads us to the assumption that these

jumps might follow the same visual patterns and an averaged representative jump can be derived for each jump type. This representative jump can be used to advise athletes in training and recognize the phases of a jump type.

This uncertainty in model and data selection and their influence is a common conclusion in other current work classifying jumps (Bitén, 2021; Echterhoff, Haladjian, & Brügge, 2018). One general notion of these works is the discussion that an ML solution should be tailored to the problem by experimental design. This is also independent of the type of sport as both publications show. Bitén also acknowledges the vast number of models that can be used and limits the studies to a certain subset. In this work, we use a similar argument to support our extensive combination of models and features in the experiments and the conclusions drawn in the results and discussion. Additionally, this problem makes explainable artificial intelligence (XAI) a strong candidate for further research, as shown by our use of SHAP values. These are already used to back up some of our claims. Other works support similar arguments by applying XAI to relevant sports-related analysis, like performance of basketball teams (Wang, Liu, & Liu, 2022). To the best of our knowledge, no work for jump classification includes explainable ML argumentation.

The documentation of training data in trampoline gymnastics has so far been limited, for example, to the collection of the training duration, athletics, number of exercises, number of individual jumps and jump combinations, whereas the technical quality of elements is not systematically recorded. The application of mobile sensors combined with predictive models for jump detection offers new possibilities in this area and indicates the potential for integrating ML systems for application in sports, particularly in the analysis of requirements for complex movements (Ancillao et al., 2018; Camomilla et al., 2018; Stetter et al., 2020).

The automated determination of the difficulty of jumps according to international scoring rules is a special challenge for competition. However, the approach

presented here also indicates prospects for supporting the difficulty judges and enables a direct derivation and formulation of individual target values for training. Finally, the competition data represent relevant factors influencing performance and can be made available in real time.

Conclusion

Machine learning methods can be used to detect jumps using sensor data. The application particularly of mobile sensors in combination with prediction models for jump detection has been insufficiently researched up to the present. The approach proposed herein basically shows considerable potential for expanding mobile applications in a sport with complex movement requirements. Future work is planned to apply these techniques to provide immediate feedback through which an athlete's performance is evaluated or the difficulty judge is supported.

Corresponding address



Lucas Woltmann
Dresden Database Research
Group, Technische Universität
Dresden
Helmholtzstr. 10,
01069 Dresden, Germany
lucas.woltmann@tu-
dresden.de

Acknowledgements. We would like to thank Dirk Debus and Florian Schneider from 2D Debus & Diebold Messsysteme GmbH for their continuous support.

Funding. Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest. L. Woltmann, C. Hartmann, W. Lehner, P. Rausch and K. Ferger declare that they have no competing interests.

For this article, data was collected from four trampoline gymnasts (male, $n = 3$; female, $n = 1$) who are competing at the national level from local gymnastics clubs located in Bad Kreuznach and Frankfurt, Germany. All studies mentioned were in accordance with the ethical standards indicated in each case. Written informed consent was obtained from the participants in advance. This project was approved by the local Human Ethics Committee of the University of Giessen.

Open Access. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ancillao, A., Tedesco, S., Barton, J., & O'Flynn, B. (2018). Indirect measurement of ground reaction forces and moments by means of wearable inertial sensors: a systematic review. *Sensors (Basel, Switzerland)*, 18(8), 2564. <https://doi.org/10.3390/s18082564>.
- Baca, A. (2006). Innovative diagnostic methods in elite sport. *International Journal of Performance Analysis in Sport*, 6(2), 148–156. <https://doi.org/10.1080/24748668.2006.11868380>.
- Bitén, F. (2021). *Trampoline jump classification using sparse data in machine learning*. Stockholm: KTH, School of Electrical Engineering and Computer Science (EECS).
- Camomilla, V., Bergamini, E., Fantozzi, S., & Vannozzi, G. (2018). Trends supporting the in-field use of wearable inertial sensors for sport performance evaluation: a systematic review. *Sensors (Basel, Switzerland)*, 18(3), 873. <https://doi.org/10.3390/s18030873>.
- Campbell, R. A., Bradshaw, E. J., Ball, N., Hunter, A., & Spratford, W. (2021). Inertial measurement units are 'all g': inter-trial reliability when assessing upper and lower body impact loading in artistic gymnastics. *International Journal of Sports Science & Coaching*, 16(2), 380–390. <https://doi.org/10.1177/1747954120970310>.
- Chambers, R., Gabbett, T. J., Cole, M. H., & Beard, A. (2015). The use of wearable microensors to quantify sport-specific movements. *Sports Medicine*, 45(7), 1065–1081. <https://doi.org/10.1007/s40279-015-0332-9>.
- Datasheet 2d-Daterecording (2021). Datenlogger bodymotion V5. <https://2d-daterecording.com/en/produkte/hardware/logger/analog-logger/bodymotion/bodymotion-v5/>. Accessed 26 Nov 2021.
- Echterhoff, J. M., Haladjian, J., & Brügge, B. (2018). Gait and jump classification in modern equestrian sports. In *Proceedings of the 2018 ACM international symposium on wearable computers* (pp. 88–91).
- Ferger, K., & Hackbarth, M. (2017). Potentialities for determining time of flight and positioning in competitive trampolining—an article on performance development in competitive trampolining. *Science of Gymnastics Journal*, 9(3), 303–310.
- Ferger, K., Hackbarth, M., Mylo, M. D., Müller, C., & Zentgraf, K. (2019). Measuring temporal and spatial accuracy in trampolining. *Sports Engineering*, 22(63), 3–4. <https://doi.org/10.1007/s12283-019-0310-9>.

- Ferger, K., Helm, F., & Zentgraf, K. (2020). Estimating horizontal displacement deduction in trampoline gymnastics by means of constant and variable errors of landing positions: a new gold standard? *Science of Gymnastics Journal*, 12(2), 203–216.
- FIG: Fédération Internationale de Gymnastique Executive Committee (2016). 2017–2020 Code of Points. http://www.fig-gymnastics.com/publicdir/rules/files/tra/TRA-CoP_2017-2020-e.pdf. Accessed 24 Okt 2022.
- Harding, J. W., Small, J. W., & James, D. A. (2007). Feature extraction of performance variables in elite half-pipe snowboarding using body mounted inertial sensors. In D. V. Nicolau, D. Abbott, K. Kalantar-Zadeh, T. Di Matteo & S. M. Bezrukov (Eds.), *SPIE Proceedings, BioMEMS and Nanotechnology III* (p. 679917). Bellingham: SPIE. <https://doi.org/10.1117/12.759259>.
- Helten, T., Brock, H., Müller, M., & Seidel, H.-P. (2011). Classification of trampoline jumps using inertial sensors. *Sports Engineering*, 14(2–4), 155–164. <https://doi.org/10.1007/s12283-011-0081-4>.
- Hood, S., McBain, T., Portas, M., & Spears, I. (2012). Measurement in sports biomechanics. *Measurement and Control*, 45(6), 182–186. <https://doi.org/10.1177/002029401204500604>.
- Huang, J., & Perry, M. (2016). A semi-empirical approach using gradient boosting and k-nearest neighbors regression for GEFCom2014 probabilistic solar power forecasting. *International Journal of Forecasting*, 32(3), 1081–1086. <https://doi.org/10.1016/j.ijforecast.2015.11.002>.
- Knight, J. F., Bristow, H. W., Anastopoulou, S., Baber, C., Schwirtz, A., & Arvanitis, T. N. (2007). Uses of accelerometer data collected from a wearable system. *Personal and Ubiquitous Computing*, 11(2), 117–132. <https://doi.org/10.1007/s00779-006-0070-y>.
- Li, R. T., Kling, S. R., Salata, M. J., Cupp, S. A., Sheehan, J., & Voos, J. E. (2016). Wearable performance devices in sports medicine. *Sports Health*, 8(1), 74–78. <https://doi.org/10.1177/1941738115616917>.
- Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 4765–4774). Red Hook: Curran Associates, Inc. <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>.
- Mendes, J. J. A., Vieira, M. E. M., Pires, M. B., & Stevan, S. L. (2016). Sensor fusion and smart sensor in sports and biomedical applications. *Sensors (Basel, Switzerland)*, 16(10), 1569. <https://doi.org/10.3390/s16101569>.
- Meyer, H. J., Grunert, H., Waizenegger, T., Woltmann, L., Hartmann, C., Lehner, W., Esmailoghli, M., Redyuk, S., Martinez, R., Abedjan, Z., Ziehn, A., Rabl, T., Markl, V., Schmitz, C., Serai, D. D., & Gava, T. E. (2019). Particulate matter matters—the data science challenge @ BTW 2019. *Datenbank-Spektrum*, 19(3), 165–182. <https://doi.org/10.1007/s13222-019-00322-x>.
- Stetter, B. J., Krafft, F. C., Ringhof, S., Stein, T., & Sell, S. (2020). A machine learning and wearable sensor based approach to estimate external knee flexion and adduction moments during various locomotion tasks. *Frontiers in Bioengineering and Biotechnology*, 8, 9. <https://doi.org/10.3389/fbioe.2020.00009>.
- Wang, Y., Liu, W., & Liu, X. (2022). Explainable AI techniques with application to NBA gameplay prediction. *Neurocomputing*, 483, 59–71.
- Woltmann, L., Volk, P., Dinzinger, M., Gräf, L., Strasser, S., Schildgen, J., Hartmann, C., & Lehner, W. (2022). Data science meets high-tech manufacturing—the BTW 2021 data science challenge. *Datenbank-Spektrum*, 22(1), 5–10.