

Online Stability Estimation based on Inertial Sensor Data for Human and Humanoid Fall Prevention

Laura Steffan, Lukas Kaul and Tamim Asfour

Abstract—Distinguishing between dynamically stable and unstable body poses during the execution of whole-body motions is of equal importance for humanoid robots and humans assisted by robotic exoskeletons. In this work, we present a study for developing a real-time system for detecting dynamic instability based on a small number of body-mounted inertial measurement units (IMUs). To this end, we systematically evaluate different online capable classifiers, operating on the data of 1 to 6 body mounted sensors, trained on a dataset of 50 disturbed motions with nearly 30,000 motion frames recorded at 100 Hz. In contrast to the majority of related studies, our system does not make use of thresholding certain sensor values but instead uses machine learning techniques to detect characteristics and patterns of features of unstable movements. We show that the right combination of classification method and sensor placement on the human body leads to very good detection results with only 3 sensors.

I. INTRODUCTION

For bipedal humanoid robots as well as for humans (and for humans wearing exoskeletons), controlling and maintaining a stable upright configuration during walking is essential. Falls can cause severe damage or injury and therefore need to be prevented. The research presented in this paper is concerned with estimating the instantaneous dynamic stability of humans during walking based on the data acquired from body-mounted inertial measurement units (IMUs). Detecting unstable situations is necessary as input for reactive measures to regain stability, e.g. a recovery step in the case of a robot or enhanced torque augmentation in the case of a human/exoskeleton system.

To achieve this, we use methods known from robotics (the Zero Moment Point, ZMP [1]), and apply them to humans, where significantly less sensor data than on a typical robot is available even if body-mounted sensors are deployed. This lack of data, such as joint angles and velocities, makes the computation of the ZMP more complex. To overcome this challenge we use the ZMP for labeling human motion capture data, where sufficient information for its calculation is available. Based on the labeled motion data we train automatic classifiers that learn to distinguish dynamically stable from unstable body poses in an end-to-end fashion on the data provided by body mounted inertial sensors only. Those sensors can be deployed as wearable devices outside a motion capture environment.

The main contribution of this paper is twofold: We present a method to automatically label human locomotion

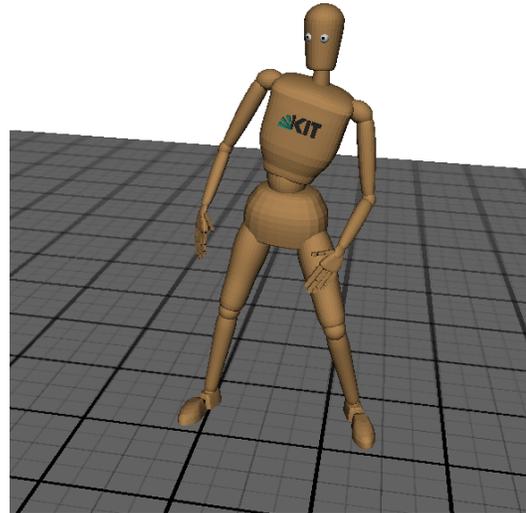


Fig. 1: Normalized Representation of a human subject regaining balance after a push from the right side at shoulder height.

data acquired with an optical marker-based motion capture system (VICON) as dynamically stable/unstable, based on the relation of the ZMP and the support polygon. The degree of instability we aim at detecting is such that it can still be recovered from (i.e. we do not just want to detect inevitable falls). Based on the labeled data, a thorough analysis of sensor positions and machine learning algorithms (binary classifiers) for detecting unstable configurations is performed. Special attention is thereby paid to identifying the minimal subset of sensors that lead to satisfactory results since real-world applications benefit from lower numbers of required sensors. We consider the placement of up to six body-mounted sensors as feasible and hence restrict our study to this number of sensors. All combinations of up to six sensors from all 34 possible sensor positions on the body in combination with six classification methods are tested and evaluated in terms of their instability detection performance based on their respective F1-score, taking into account precision and recall. Through this analysis we are able to identify the best suited sensor positions on the human body and data classification methods that allow reliable and efficient stability estimates in real-time.

Such methods are primarily relevant for humans wearing exoskeletons, where an active exoskeleton can potentially intervene and prevent a fall that would have otherwise arisen from the detected unstable state. The required inertial

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under the project INOPRO (16SV7665).

The authors are with the Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology, Karlsruhe, Germany, {lukas.kaul, asfour}@kit.edu

sensors could either be mounted on the exoskeleton directly or, in case sensor position not covered by the exoskeleton are favorable, be integrated in the wearer's clothes. Since maintaining balance and preventing falls is of equal importance for bipedal humanoid robots, the same methods may find application in state estimation and balance control in humanoid robotics.

II. RELATED WORK

Research in the context of stability estimation and fall detection is mainly driven by the requirements of human health and assistive systems on the one hand, and humanoid robotic research concerned with the generation of human-like bipedal motions on the other hand. Therefore, related research work may not only be found in the context of human movement observation and analysis, but also in the work done on humanoid walking and balancing. For instability detection, a key parameter is the time between triggering the alarm and the time the fall becomes inevitable. The earlier the instability is detected, the more time remains for interventions like executing motions that prevent a fall or selecting damage minimization strategies, like moving the arms to a safe position or releasing an airbag system.

A. Stability Estimation in Humans

For human fall detection, sensors and devices applied in the past include cameras ([2], [3], [4]), vibration and sound sensors ([5], [6]), smartphones ([7], [8]) as well as accelerometers and gyroscopes ([9], [10], [11], [12]). For real world supervision, body-mounted sensors like accelerometers and gyroscopes, which may be integrated in smart devices like phones and watches, are preferable, as they overcome shortcomings of video and sound systems which are restricted to certain environments where they may be installed and trained. In the past, numerous systems deploying accelerometers and gyroscopes were presented. A common way is to build a threshold based model filtering high accelerations, which are then matched with further patterns common in falls before triggering alarm ([13], [14]). For those approaches, the reaction time after high impacts is relatively high. Therefore, recent approaches tend to apply more sophisticated machine learning methods to reveal patterns related to disturbances occurring earlier than the high accelerations directly preceding the floor contact in falls. Promising approaches in this category include the use of Support Vector Machines (SVM) [9], decision trees [10] and k-Nearest Neighbor methods [11].

An outstanding result from the literature in this area of research was achieved with the use of a Hidden Markov Model (HMMs) that detects falls with an accuracy of 100%, 200 ms - 400 ms before ground impact (which might still be too late to prevent the fall) [12]. This and other approaches are not only tested for fall data, but are compared to a range of other motions, denoted Activities of Daily Life (ADL) in literature. Such activities usually include walking and postural transitions, e.g. from sitting to standing and vice versa. Most of the studies use only one single sensor

and choose its position according to practical reasons rather than attempting a data-driven analysis of different sensor positions. However, some studies (e.g. [13], [15]) evaluate different possible positions and state that sensors positioned on the torso yield more reliable detection systems.

B. Stability Estimation in Humanoid Robotics

In the field of humanoid robot locomotion, one of the most common method for achieving dynamic balancing is controlling the Zero Moment Point (ZMP), a concept that was initially introduced by Vukobratovic [1]. The ZMP is often applied to reduced dynamic models such as the rimless wheel or the (linear) inverted pendulum to derive control systems for humanoid robots ([16], [17]). Since the effects of complex, multi-joint motions on robots inevitably defer from the predictions of simplified dynamic models, the supervision and stability control remain an important part of humanoid robot motion execution.

Rather than labeling momentary states as stable or unstable, other approaches observe the results of entire movements and divide them in those that remain stable and those that inevitably lead to falls. Thereby, the set of safe (stable) motions are called *viable* and constitute the 'viability kernel', i.e. motions that can be executed without potential damage ([18], [19]). In [19], the authors label the existent data using actual state and result, in the classes 'balanced' (motion currently stable and remaining stable), 'fallen' (motion that already led to a fall) and 'falling' (the sequence that leads to a fall, but prior to impact). They use machine learning techniques and distinguish between those classes and manage to trigger an alarm at least 700 ms before the robot actually falls. However, they have a false positive rate of 10%, which could cause many false alarms in practice.

In the work described in [20], push-recovery methods for a small humanoid robot are presented that leverage a single torso-mounted IMU (attitude sensor and gyroscope) as well as the joint encoders to generate a stability estimate, which is then mapped to a recovery action. To enhance this mapping, reinforcement learning with a simulated robot is used for optimization. In the context of soccer-playing humanoid robots, the authors in [21] argue that good soccer players, particularly goalkeepers, should be able to intentionally fall down. They describe a classification method for unstable situations and manage to execute a damage-preventing fall primitive if necessary. Also in the context of small-scale humanoid soccer robots, [22] presents a simulation-based study on an effective instability detector with attitude sensors. Deviations from expected attitude signals are aggregated into an instability indicator that triggers one of two reflexive behaviors (slowing down or stopping).

III. METHODOLOGY

This study is based on human locomotion data taken from a large database of human whole-body motions, the KIT Whole-Body Human Motion Database (see [23] and [24]). The motions were captured using an accurate multi-marker optical motion capture system (VICON) and processed with

SupportPolygon and ZMP in every 10th frame, push recovery (push from left side)

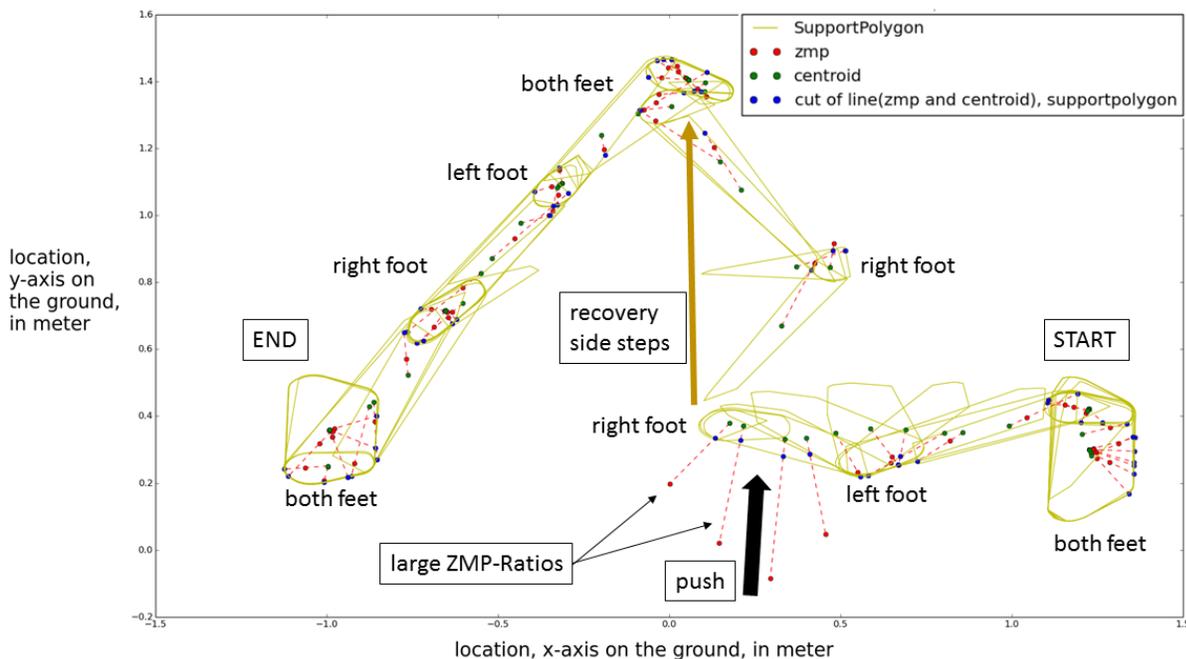


Fig. 2: Support Polygon with its centroid and ZMP of a person walking from right to left over time. At about 0.25 m on the x-axis, the subject experiences a push from the left, leading to a significant deviation of the ZMP (red dots and dashed lines). After that, the subject takes two recovery steps to the right in order to maintain balance. From there, three more steps lead the subject to the target position. The blue dots (“cut of line”) are the intersections of the line from the centroid to the ZMP with the current support polygon (*Boarder* in Equation 1). The support polygon, its centroid and the ZMP are plotted at every 10th motion frame (i.e. 100 ms apart).

the Master Motor Map toolchain [25] to obtain a normalized motion representation across different subjects (see Figure 1). The Master Motor Map (MMM) provides a unifying framework for representing whole-body human motion data in conjunction with tools and methods for motion analysis and reproduction. In its core, the MMM contains a reference model of the human body providing a well-specified kinematic configuration and dynamics, and defines the data formats for capturing, representing and mapping of human motion to a target robot system. This reference model is used to generalize motion recordings from the subject-specific properties and, thus, facilitates the implementation of motion mapping mechanisms to different embodiments. Motions are represented as frames at a rate of 100 Hz, containing the marker positions, the Cartesian pose of the root body and all joint angles of the underlying reference model of the human body (the MMM model), from which all velocities and accelerations can be numerically derived. The data in the KIT Whole-Body Human Motion Database is collected using a reference marker set consisting of 56 markers which are derived from specific anatomical landmarks of the human body. More information about the marker set is available online.¹ The database currently contains more than 9,400 human motion recordings of 154 subjects and a total duration of 28.8 hours of manually labeled human motion data.

¹https://motion-database.humanoids.kit.edu/marker_set

In the 50 recordings used in this study, the subject is disturbed heavily enough to be considered dynamically unstable, requiring active recovery actions to avoid falling. In these 50 motions, the subject is either walking or standing, and at some point in time unknown to the subject is experiencing a disturbing force, causing the motion to be momentarily unstable. The disturbances are externally applied pushes to the torso from either the front, back, left or right side. None of the disturbed motions actually led to a fall, which ensures that any classifier trained on this data does not learn to detect when a person has fallen but is much more sensitive, detecting unstable situations from which the subject can recover. This is a significant difference to many of the related works presented in Section II.

A. Data Preparation

The goal of this work is to distinguish stable from unstable situations based on body-mounted inertial sensors with a frame-wise, automatic classifier. The classifier needs to be trained on labeled data, which requires two major steps of data preparation: (1) adding IMU-data to the motion recordings and (2) labeling them.

The IMU-data consists of three-axis linear acceleration, three-axis rotational rate (both in the sensor frame) and absolute orientation, which can be provided by state-of-the-art wearable sensors such as Mbiendlab’s MetaMotionR [27]. Using the MMM framework ([25], [24]) and the Simox

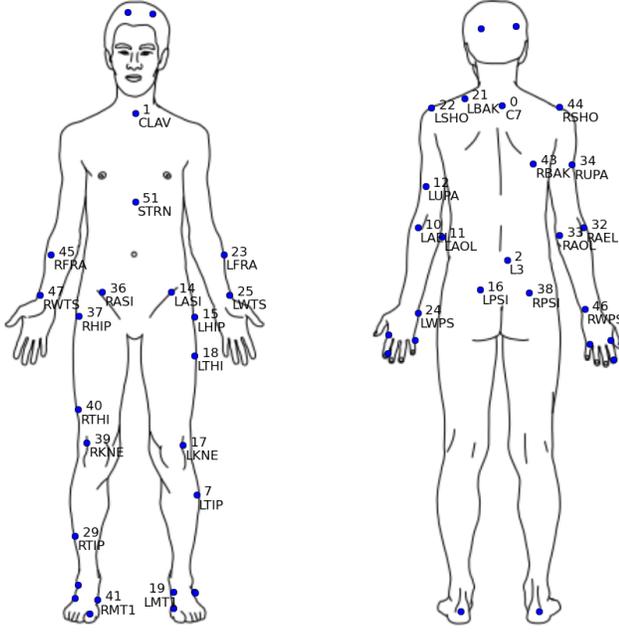


Fig. 3: All optical markers from the standard MMM marker set depicted as blue dots on the human body. 34 of these markers served as locations for emulated IMUs. These IMU-positions are indicated by blue dots having a numerical index and are referenced in Table I. The figure of the human body in this figure is under creative common license and can be found at [26].

toolbox ([28]), we emulate the IMU data by computing those measurements for 34 positions on the human body and adding them to the motion data. The sensor positions coincide with a subset of the locations of the optical reflective markers from the standard MMM marker set, which consists of 56 markers at predefined anatomical locations, since their positions and motions are exactly known without the need for further interpolation. This selection was based on considerations of practicability in later applications (e.g. attaching wearable IMU sensors to the head or toes is impractical, hence those sensor locations were not considered). Figure 3 shows the positions of all optical markers as blue dots on the human body, where the 34 emulated IMU sensor positions are indicated by the markers with numerical indices. Note that higher indices than 34 occur since they denote MMM marker positions which were not all used as emulated sensors.

For training a frame-wise binary classifier, an appropriate label (stable/unstable) must be associated with each of the nearly 30,000 frames of the 50 motions containing unstable situations. While this could be done manually, it would be extremely tedious. We therefore developed an automatic procedure to derive the correct label based on the position of the ZMP in relation to the support polygon. The support polygon can be directly computed from the kinematic motion data, and the ZMP can be computed from the mass distribution of the human model and its motion, using the method described

in [29]:

$$zmp_p_x = \frac{\sum_{i=1}^N \left\{ \vec{r}_i \times m_i (\vec{a}_i - \vec{g}) + [d(\vec{I}_i \vec{\omega}_i)/dt] \right\}_Y}{M(\ddot{Z}_{CoM} + g)}$$

$$zmp_p_y = \frac{\sum_{i=1}^N \left\{ \vec{r}_i \times m_i (\vec{a}_i - \vec{g}) + [d(\vec{I}_i \vec{\omega}_i)/dt] \right\}_X}{M(\ddot{Z}_{CoM} + g)}$$

Hereby we use the physical formulations for linear and angular momenta, as well as the gravitational force, to compute the values of the ZMP on the ground in x- and y- axis separately. The used variables denote the position \vec{r}_i , mass m_i and acceleration \vec{a}_i of every body part $i = 1, \dots, n$, the inertial momenta I_i and angular velocity $\vec{\omega}_i$ of these as well as the overall mass M and the height of the center of mass of the whole body Z_{CoM} . \vec{g} describes the downward acting gravitational acceleration. Using this formulation, the ZMP computed from recorded motion data does not always remain inside the support polygon. In fact, it frequently leaves the support polygon even for undisturbed dynamic motions. However, the *distance* between the centroid of the support polygon and the ZMP can still be considered a measure of dynamic stability - the greater this distance, the less stable (or more disturbed) the motion.

To account for varying sizes of the support polygon that arise from the kinematic differences of single and double support phases, we define the *ZMP-Ratio* that puts into relation the distance between the centroid of the support polygon and the ZMP to the distance between the centroid of the support polygon and its border, all on the line passing through centroid and ZMP.

$$\text{ZMP-Ratio} = \frac{\text{Distance}(\text{Centroid}, \text{ZMP})}{\text{Distance}(\text{Centroid}, \text{Border})} \quad (1)$$

This value can be computed for every frame and serves as a measure for dynamic stability. It can be interpreted as follows:

- ZMP-Ratios close to 0 indicate a ZMP that is close to the centroid (stable)
- ZMP-Ratios greater than 1 indicate a ZMP that lies outside of the support polygon (potentially unstable)

Since values of the ZMP-Ratio greater than 1 do not automatically indicate instability, we resorted to empirically defining a threshold that reliably indicates an instability in the sense that the subject would have fallen if no dedicated recovery action was performed. This value was found by examining videos of the 50 disturbed motion trials and computing the respective ZMP-Ratio values during the disturbances. We found a threshold of 2.5 as a reliable indicator of a significant instability that requires recovery actions. That is, push disturbances that lead to clearly visible countermeasures by the pushed subject in the form of either whole-body motions, stepping or a change in walking direction induce a ZMP-Ratio of 2.5 or more, whereas this value remains smaller during undisturbed motion. Based on this

threshold of the ZMP-Ratio, all frames of the investigated motions are assigned a binary label (stable/unstable), so that a classifier that directly operates on the IMU-data (not on the ZMP) can learn to distinguish those two classes.

B. Classification

Two criteria define a good classifier in the context of this study: (1) It should reliably (measured by the F1-score) distinguish between stable (ZMP-Ratio < 2.5) and unstable (ZMP-Ratio > 2.5) poses from IMU-data and (2) require a small number of body mounted sensors to ensure feasibility in real-world applications. To achieve the latter and reduce the number of used sensors from the initially available 34 to a smaller number, a feature selection process is necessary. Since each sensor provides 9 features (3-axis acceleration, 3-axis rotation rate, 3D orientation) it is not feasible to remove features individually and they have to be treated in a bulk of 9, either using all of them or removing the entire sensor. Feature selection in our case thus becomes synonymous to sensor selection. We further need to select from a set of available methods of automatic classifiers, where we consider Bayesian regression, Support Vector Machines, Nearest Neighbors classification as well as Perceptron methods, both single layer and multilayer (i.e. neural networks). To assess the effect that bagging of classifiers has on the presented learning task we also evaluated a bagged version of the Nearest Neighbors method, which we chose for its promising results even in the non-bagged version.

We rely on the implementations of these methods available from the SciKit-learn toolbox for Python [30]. All of these methods are trained and evaluated for every possible sensor combination of up to 6 sensors, which we consider the upper limit of practicability in later applications. The very significant computation time required for this complete search for the best combination of sensor set and classification method (on the order of days) must be invested only once and leads us to a trained, ready-to-use classification tool that can be evaluated in real-time for each motion frame to detect unstable situations that require balance recovery control.

IV. RESULTS

To get a baseline for the classification results obtained from the different classifiers, training and evaluation was first conducted with all 34 sensors (i.e. 306 features) for each classifier. Table I shows the F1-score for these experiments in the last row. The Bayes, Bagged 10 Nearest Neighbors and Support Vector Classifier (SVC) perform equally well with an F1-score of 66%. The 10 Nearest Neighbors without bagging, the multi-Layer perceptron and the single-layer perceptron follow with an F1-score of 64%, 63% and 61%, respectively. When considering the mere percentage of correctly classified frames, it is important to take into account our comparatively high frame rate of 100 Hz. We consider much shorter time intervals than most related approaches, which implies that our system may appear less accurate at first sight, even when actually reporting better results. We performed a complete search for the sensor setups from 1 to

6 sensors that perform best in terms of the F1-score for each classifying technique. The results, consisting of the sensor identifiers (numbers) and the F1-score for each classifier, is presented in Table I, where the listed sensor set with n sensors is the one that led to the highest score with this classifier out of all possible sets of n sensors. The sensor identifiers relate to sensor positions as indicated in Figure 3. To be able to use as much data as possible for training and still get meaningful evaluation results, four-fold cross-validation was used for all training and evaluation. This leads to an F1-score difference of approximately $\pm 2\%$ among the four evaluation sets. Table I shows the average of the score over all evaluation sets.

As can be seen, the results vary significantly when applying different methods to different numbers of demanded sensors. When adding higher numbers of sensors, higher order approaches like the Nearest Neighbors method and the multilayer perceptron outperform linear methods such as the Bayesian classifier.

A. Observations and Discussion of the Results

Several interesting and noteworthy observations can be derived from the presented results.

1) *Sensor Reoccurrence*: Most notably, optimal sensor sets for a given classifier with n sensors are generally not subsets of the optimal sensor set with $n + m$ sensors. For example, the best single sensor for the bagged 10-NN classifier is the sensor at position 32, which does not reoccur in any of the other optimal sensor combinations for this classifier for up to 6 sensors. Moreover, the optimal two-sensor setup consists of the sensors positions 0 and 36, both of which do not reoccur in the three- or four-sensor setup (both reappear in the five-sensor setup).

2) *Number of Sensors*: More sensors, i.e. more dimensions in the feature space, do not automatically lead to better results, emphasizing the relevance of appropriate feature selection to avoid the 'curse of dimensionality' when employing high dimensional machine learning tools on limited training data. A good example for this observation in the presented study is the 10-NN classifier that reaches an F1-score of 64% when using all 34 sensors, a result that is surpassed by even using the single best sensor alone (68%). Most classifying methods show a relatively steep improvement when adding a second and third sensor (e.g. the SVC, improving from 55% with one sensor to 73% with three sensors) and then continue to improve at a much lower rate, if improving at all. The 10-NN classifier for example reaches its highest score (78%) with only 5 sensors, and the single layer perceptron shows its best performance (76%) for both 4 and 5 sensors.

3) *Sensor Location*: The question for the single best sensor location on the body to detect unstable states can only be answered in the context of a specific classification algorithm. All six classifiers presented here achieve their best single-sensor F1-scores with a different sensor (depicted in Figure 4). We conclude that the proper choice of sensors is highly dependent on the classification method, and that being

# Sensors	Bayes	10 Nearest Neighbors	10 Nearest Neighbors Bagged	Perceptron	ML-Perceptron	SVC
1	40 67%	17 68%	32 70%	33 63%	38 70%	34 55%
2	14,40 67%	34,40 75%	0,36 76%	32,51 70%	24,36 77%	33,43 67%
3	39,40,43 70%	33,38,40 76%	10,16,44 79%	17,32,43 73%	1,40,47 80%	21,41,47 73%
4	1,2,41,46 70%	0,15,38,39 75%	29,34,37,44 80%	22,25,33,43 76%	0,39,40,46 80%	10,14,43,47 74%
5	14,32,40,41,51 70%	16,34,39,40,44 78%	0,43,36,37,39 81%	15,24,36,44,45 76%	23,38,39,46,51 81%	14,21,29,39,47 76%
6	15,33,39,40,41,51 70%	29,34,37,39,40,44 77%	29,33,34,36,38,43 81%	0,1,24,33,36,41 73%	1,2,23,41,46,47 82%	10,15,21,39,40,46 76%
34 (all)	66%	64%	66%	61%	63%	66%

TABLE I: Listing of the 36 classifiers as combinations of classification method (row) and number of used IMU sensors (column) that led to the best F1-scores among all possible systems with the same number of sensors. Each cell contains the sensors used (see Figure 3 for reference) together with the achieved F1-score. The result for each classification method operating on all 34 sensors is given in the last row. Note that the numbers represent marker positions of the MMM reference marker set (that translate to emulated IMU positions) and thus can be higher than the number of 34 sensors (see also Section III-A).

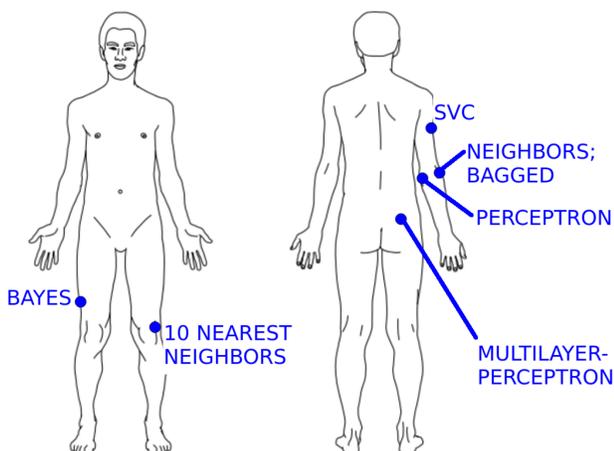


Fig. 4: Location of the sensor that leads to the best detection result in a single-sensor setup for the six investigated classification methods.

the “best” sensor is not an attribute of the sensor but of the combination of sensor and method.

As exemplary visualized in Figure 4, the optimal sensor sets show a bias towards the right side of the body. We hypothesize that the reason for this lies in an asymmetric preference for the choice of the stepping foot when pushed from the back or the front while standing.

B. Best System

Of all the 42 classification systems summarized in Table I, consisting of the combination of sensors and a machine learning method, using six sensors (1, 2, 23, 41, 46 and 47) along with a multilayer perceptron (one hidden layer, trained via back-propagation) leads to the best results (F1-score of 82%). For actual use, the same system but with only three sensors (1, 40 and 47) might be even better suited due to the lower number of sensors at the small cost of an F1-score dropping 2 percentage points. A single classification using the six-sensor model takes 1.45 ms on average on an *Intel Core i7*, 4th generation with 16 GB RAM. Using

the trained system on real-time data with a frame-rate of 100 Hz will therefore be possible without the need for further optimizations.

V. CONCLUSION

We presented a method to automatically label frames of whole-body motion recordings as either stable or unstable based on a quantitative, ZMP-based criterion and applied it to a set of 50 different recordings of disturbed motions with nearly 30,000 motion frames. We computationally added 9-dimensional IMU data to the motion recordings for 34 sensor positions on the body. With this dataset, we systematically searched for the best frame-wise detection system for unstable states that operates on the IMU-data, comprising a binary classifier and a set of up to six sensors. Six different classifying techniques were evaluated in terms of their F1-score, each on every possible sensor combination. The best result was achieved with a multilayer Perceptron (neural net) with one hidden layer and six body mounted sensors (i.e. 54 features). The F1-score for this setup is 82%. Interestingly, results of similar quality can be achieved with the same method but only three sensors (F1-score of 80%).

A. Discussion and Future Work

The methods presented here were solely evaluated on motion data acquired with the optical VICON system, while the IMU sensor data, that is the foundations of the classifier, were computationally emulated. It therefore remains to be investigated how well the results can be reproduced from data gathered directly from body-mounted IMUs, where noise level and other sensor-specific parameters may differ from the data used for the presented study. Furthermore, the labeling-method relies on the ZMP-Ratio introduced in Section III and a numerical threshold that is based on heuristics. While a smaller threshold would lead to more false positives and a larger to more false negatives, an in depth analysis of the implication of this threshold remains to be done. Another point to investigate is the potential for improvement when using a version of the classifier

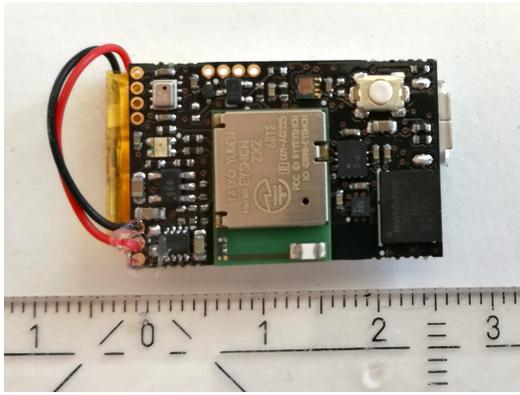


Fig. 5: Self-contained, wireless IMU module [27] for studies with body-mounted sensors. Centimeter scale for size comparison.

that uses evaluation of three or more consecutive frames to verify results. Our evaluation showed that many of the false positives only occur in a single motion frame and could be suppressed by evaluating a window of several consecutive frames, still maintaining a high frame rate.

In our future work, we will evaluate the frame-wise stability estimation method with up to six body mounted wireless inertial orientation sensors like the one shown in Figure 5.

REFERENCES

- [1] M. Vukobratovic and D. Juricic, "Contribution to the synthesis of biped gait," *IEEE Transactions on Biomedical Engineering*, no. 1, pp. 1–6, 1969.
- [2] T. Lee and A. Mihailidis, "An intelligent emergency response system: preliminary development and testing of automated fall detection," *Journal of telemedicine and telecare*, vol. 11, no. 4, pp. 194–198, 2005.
- [3] G. Wu, "Distinguishing fall activities from normal activities by velocity characteristics," *Journal of biomechanics*, vol. 33, no. 11, pp. 1497–1500, 2000.
- [4] E. E. Stone and M. Skubic, "Fall detection in homes of older adults using the microsoft kinect," *IEEE journal of biomedical and health informatics*, vol. 19, no. 1, pp. 290–301, 2015.
- [5] M. Alwan, P. J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, and R. Felder, "A smart and passive floor-vibration based fall detector for elderly," in *Information Communication Technologies, 2nd International Conference on*, vol. 1. IEEE, 2006, pp. 1003–1007.
- [6] Y. Zigel, D. Litvak, and I. Gannot, "A method for automatic fall detection of elderly people using floor vibrations and sound—proof of concept on human mimicking doll falls," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 12, pp. 2858–2867, 2009.
- [7] M. V. Albert, K. Kording, M. Herrmann, and A. Jayaraman, "Fall classification by machine learning using mobile phones," *PLoS one*, vol. 7, no. 5, p. e36556, 2012.
- [8] S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio, "A smartphone-based fall detection system," *Pervasive and Mobile Computing*, vol. 8, no. 6, pp. 883–899, 2012.
- [9] T. Zhang, J. Wang, L. Xu, and P. Liu, "Fall detection by wearable sensor and one-class svm algorithm," in *Intelligent computing in signal processing and pattern recognition*. Springer, 2006, pp. 858–863.
- [10] O. Ojetoala, E. I. Gaura, and J. Brusey, "Fall detection with wearable sensors—safe (smart fall detection)," in *Intelligent Environments (IE), 7th International Conference on*. IEEE, 2011, pp. 318–321.
- [11] S. Z. Erdogan, T. T. Bilgin, and J. Cho, "Fall detection by using k-nearest neighbor algorithm on wsn data," in *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*. IEEE, 2010, pp. 2054–2058.
- [12] L. Tong, Q. Song, Y. Ge, and M. Liu, "HMM-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.
- [13] K. Doughty, R. Lewis, and A. McIntosh, "The design of a practical and reliable fall detector for community and institutional telecare," *Journal of Telemedicine and Telecare*, vol. 6, no. 1-suppl, pp. 150–154, 2000.
- [14] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE transactions on information technology in biomedicine*, vol. 10, no. 1, pp. 156–167, 2006.
- [15] M. Kangas, A. Konttila, P. Lindgren, I. Winblad, and T. Jämsä, "Comparison of low-complexity fall detection algorithms for body attached accelerometers," *Gait & posture*, vol. 28, no. 2, pp. 285–291, 2008.

- [16] M. Wisse, G. Keliksdal, J. Van Frankenhuyzen, and B. Moyer, "Passive-based walking robot," *IEEE Robotics & Automation Magazine*, vol. 14, no. 2, pp. 52–62, 2007.
- [17] A. L. Hof, "The 'extrapolated center of mass' concept suggests a simple control of balance in walking," *Human movement science*, vol. 27, no. 1, pp. 112–125, 2008.
- [18] J. Tay, I.-M. Chen, and M. Veloso, "Fall Prediction for New Sequences of Motions," in *Experimental Robotics*. Springer, 2016, pp. 849–864.
- [19] S. Kalyan Krishnan and A. Goswami, "Predicting falls of a humanoid robot through machine learning," in *Innovative Applications of Artificial Intelligence (IAAI), IEEE Conference on*, 2010.
- [20] S.-J. Yi, B.-T. Zhang, D. Hong, and D. D. Lee, "Learning full body push recovery control for small humanoid robots," in *Robotics and Automation (ICRA), IEEE International Conference on*. IEEE, 2011, pp. 2047–2052.
- [21] J. Moya, J. Ruiz-del Solar, M. Orchard, and I. Parra-Tsunekawa, "Fall detection and damage reduction in biped humanoid robots," *International Journal of Humanoid Robotics*, vol. 12, no. 01, 2015.
- [22] R. Renner and S. Behnke, "Instability detection and fall avoidance for a humanoid using attitude sensors and reflexes," in *Intelligent Robots and Systems, IEEE/RSJ International Conference on*. IEEE, 2006, pp. 2967–2973.
- [23] C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "The KIT whole-body human motion database," in *Advanced Robotics (ICAR), International Conference on*. IEEE, 2015, pp. 329–336.
- [24] C. Mandery, O. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "Unifying representations and large-scale whole-body motion databases for studying human motion," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 796–809, August 2016.
- [25] Ö. Terlemez, S. Ulbrich, C. Mandery, M. Do, N. Vahrenkamp, and T. Asfour, "Master Motor Map (MMM)—Framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots," in *Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on*. IEEE, 2014, pp. 894–901.
- [26] C. D. N. Chan. (27 February 2013) Silhouette humain asexue anterieur posterieur.svg. [Online]. Available: https://commons.wikimedia.org/wiki/File:Silhouette_humain_asexue_anterieur_posterieur.svg#globalusage
- [27] Mblentlab, "MetaMotionR Product Description," <https://mblentlab.com/product/metamotionr/>, accessed: 2017-07-06.
- [28] N. Vahrenkamp, M. Kröhnert, S. Ulbrich, T. Asfour, G. Metta, R. Dillmann, and G. Sandini, "Simox: A robotics toolbox for simulation, motion and grasp planning," in *Intelligent Autonomous Systems 12*. Springer, 2013, pp. 585–594.
- [29] M. B. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *The International Journal of Robotics Research*, vol. 24, no. 12, pp. 1013–1032, 2005.
- [30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

APPENDIX

List of the motion recordings used for this study, all available from the KIT Whole-Body Human Motion Database²:

```

push_recovery_right{01,02,03,04,05,06,07,08,09,10,11}.xml,
push_recovery_stand_back{01,02,03,04,05,06,07,08,09}.xml,
push_recovery_back{02,04,05,06,07,08,10}.xml,
push_recovery_left{01,02,03,05,06,07,08,09,10}.xml,
push_recovery_front{04,05,06,07,09}.xml,
push_from_behind{08,11,12}.xml,
push_from_the_left_side{10,11,12}.xml,
push_from_the_front09.xml,
push_front_hard03.xml,
push_recovery_stand_right11.xml,

```

²<https://motion-database.humanoids.kit.edu/>