Learning to judge like a human: convolutional networks for classification of ski jumping errors

Conference Paper · September 2017

DOI: 10.1145/3123021.3123038

3 authors:

Heike Brock
Siemens
39 PUBLICATIONS **313** CITATIONS
SEE PROFILE

Yuji Ohgi
Keio University
132 PUBLICATIONS **637** CITATIONS
SEE PROFILE

James Lee
Charles Darwin University
60 PUBLICATIONS **711** CITATIONS
SEE PROFILE

Learning to Judge Like a Human: Convolutional Networks for Classification of Ski Jumping Errors

Heike Brock
Honda Research Institute JP
Honcho 8-1, Wako, Saitama,
Japan
h.brock@jp.honda-ri.com

Yuji Ohgi
Keio University
5322 Endo, Fujisawa,
Kanagawa, Japan
ohgi@sfc.keio.ac.jp

James Lee
Charles Darwin University
Ellengowan Drive, Casuarina,
NT, Australia
jim.lee@cdu.edu.au

ABSTRACT

Advanced machine learning technologies are seldom applied to wearable motion sensor data obtained from sport movements. In this work, we therefore investigated neural networks for motion performance evaluation utilizing a set of inertial sensor-based ski jump measurements. A multi-dimensional convolutional network model that related the motion data under aspects of time, placement and sensor type was implemented. Additionally, its applicability as a measure for automatic motion style judging was evaluated. Results indicate that one multi-dimensional convolutional layer is sufficient to recognize relevant performance error representations. Furthermore, comparisons against a Support Vector Machine and a Hidden Markov Model show that the new model outperforms feature-based methods under noisy and biased data environments. Architectures such as the proposed evaluation system can hence become essential for automatic performance analysis and style judging systems in future.

Author Keywords

specialized activity recognition; convolutional neural networks; motion data; motion analysis

ACM Classification Keywords

F.1.1. Theory of Computation: Models of Computation– Self-modifying machines; H.1.2. Information Systems: User/Machine Systems– Human information processing; I.2.1. Computing Methodologies: Artificial Intelligence– Applications and Expert Systems

INTRODUCTION

Neural network architectures became a common tool for the creation of autonomous machine intelligence within the last decades [4]. Given their success with text, speech and image content, neural network models are gradually being introduced to the human motion domain. The most common application scenario is human activity recognition (HAR), typically used on motion capture data collected with maker-based camera systems or wearable sensor devices. Especially ubiquitous recognition systems using wearable sensor data appear meaningful since the devices enable various application scenarios. However, current studies commonly utilize data sets that contain acted movements of a certain number of motion patterns (e.g. walking, sitting, throwing) [20, 26] captured under laboratory conditions. Real world application tasks such as motion performance analysis in sports are likely not able to display similarly distinctive and equally distributed motion categories for information retrieval. Therefore, we applied a wearable motion capture framework of nine inertial measurement units (IMUs) to a HAR problem of a sport scenario in this research. Specifically, we developed a network model for the automatic classification of nine motion style errors in flight and landing of a ski jump performance.

This application is a topical issue serving augmented motion feedback systems in multiple sport activities: for the acquisition and maintenance of any form of correct motor skills, it is essential to know about the performed motion structure [24]. However, an optimal motion can often not be defined numerically due to varying technical skills and anthropometrics of different athletes [19]. Reliable motion error information provides a suitable performance measure here. Moreover, it offers the chance to considerably enhance the credibility of judged sports known to suffer from biased evaluation [28]: to prevent controversies and allegations of subjectivity or even fraud, it is reasonable to include objective measures whose output cannot be manipulated in the final performance scores. Consequently, the creation of autonomous machine intelligence that retrieves and classifies style information from an easily accessible stream of motion data appears an important future issue in affected sports.

Ski jumping has a clearly defined motion structure that facilitates data segmentation. Differences between good and sub-optimal performances on the other hand can be very fine-scaled and might hence require particularly strong feature representations to provide any useful error information [17, 18, 23]. Furthermore, one has to expect that sensor data captured under field conditions suffers from noise, bias or missing data that impair the data quality. Common hand-crafted features might not be sufficient in such case. Previous analysis of inertial ski jumping data confirmed this assumption, showing that specific motion characteristics of a ski jump performance could not be reliably recognized by shallow, feature-based network architectures [6]. It therefore appears reasonable to employ deep learning models that are not dependent on hand-crafted feature representations, but intrinsically learn features from the underlying data [14]. In

ISWC '17, September 11-15, 2017, Maui, HI, USA.
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5188-1/17/09...$15.00.
DOI: 10.1145/3123021.3123038

the present study, we therefore evaluated both shallow and deep models to define a network model that reliably represents skeletal and temporal correlations of different ski jump style errors. Consequentially, this model should be applicable as a measure of performance quality, even under a collection of uncontrolled and imperfect inertial field motion data.

HUMAN ACTIVITY RECOGNITION

Motion streams obtained by inertial sensors are multi-variate time-series data and typically highly dimensional as compared to image, speech or text data. Therefore, common wearable sensor-based HAR systems usually follow a fixed pipeline of work steps that aim to control and restrict the multi-dimensional output [7]. The two most influential steps in this procedure are the data segmentation into activity segments and the extraction of meaningful feature representations. Both methods rely on manual, hand-crafted algorithms and data transformations, and it is often cumbersome to determine those data properties that work best for a given task. For example, potential feature extractors can range from simple statistical or spectral descriptions [27, 2] to kinematic-induced features such as body pose and body joint position [10]. Considering the power of deep network architectures to learn hidden features within a data set, it can be assumed that the ability of automated motion information systems is not yet fully explored.

A number of deep HAR networks have been presented that belong to any of the two strategies: probabilistic approaches with deep belief networks using Restricted Boltzmann Machines [21] or recurrent Long Short-Term Memory cells [11] and convolutional neural networks (CNNs) [12, 25]. A recent investigation by Hammerla et al. [9] showed that recurrent networks outperform convolutional networks on short and temporally ordered motion sequences, whereas CNNs worked better under long and repetitive actions. Although ski jumping does not constitute a cyclic motion, we chose CNNs for the following system implementation. The rationale for this decision was that motion errors of a ski jump performance are expected to occur in arbitrary variations that cannot be contextualized.

To date, most CNN-based HAR architectures do not fully explore the possibilities of a convolutional layer. Instead, they apply one-dimensional filter along the temporal axis only [12, 20, 25, 26]. Architectures that process sensor measurements as multi-variate data over two or three dimensions are rare. Examples are the network by Ronao and Cho [22], which orders the different sensor input data along the image color channel, and the network introduced by Jiang and Yin [13], which orders the data channels of one sensor along the vertical image axis. Bashivan et al. [3] introduced a CNN learning on images generated from EEG data. In this work, an even more locally connected network along two dimension plus the color channel should be designed and evaluated to learn all relevant motion dependencies by the network.

SKI JUMP DATA BASE

Data necessary for the implementation of the intended error classification system was collected during two regional summer ski jump training camps at a normal ski jump hill. Four junior athletes and one ski jumping judge volunteered to participate in the present study. The study was conducted in accordance with the Declaration of Helsinki for ethical research and the safety of the participating athletes was constantly ensured.

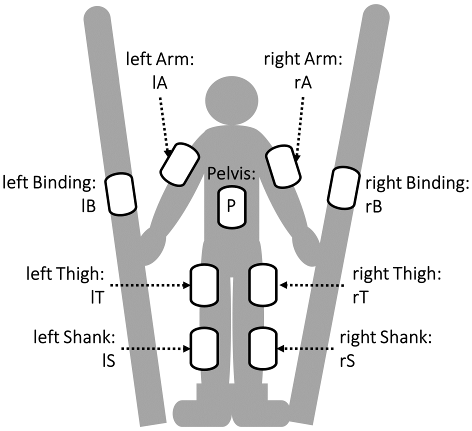

Figure 1. Schematic representation of the sensor placement used for the collection of the underlying ski jump motion data and their abbreviation used in this manuscript as seen from behind. Scaling does not represent the actual relations between sensor size and body proportions.

To obtain meaningful motion data, nine waterproof IMUs (Logical Product. SS-WS1215/ SS-WS1216) [15] were directly attached to the participating athletes using strong adhesive and kinesiology tape. Anthropometric regions chosen as measurement input data were the pelvis (P), right and left thigh (rT, lT), right and left shank (rS, lS), right and left ski anterior to the ski binding (rB, lB) and right and left upper arm (rA, lA) (Figure 1). After sensor set-up, the body-worn sensors were located within the ski jump suit, whose tight fit kept the sensors in place and prevented any unintended sensor artifact. Each sensor contained triads of gyroscopes, accelerometers and magnetometers of 16 bit quantization rate and was set to sample data at 500 Hz. The gyroscopes were specified with a full-scale range of ± 1500 dps with 0.67 mV/dps sensitivity. Accelerometer specification varied in dependence on the placement between either a minimum full-scale range of ± 5 G (body placement) or ± 16 G (ski placement) with 191.7 mV/G sensitivity. Magnetic field sensors had ± 1.2 Gauss full-scale range. Measurements of all sensors were synchronized before each jump from a computer at the top of the jump slope via Bluetooth data connection. All sensor settings were set to capture and internally store data for a time interval of 45 seconds.

The training camps had a duration of three and four days respectively with a morning and afternoon session of five to eight jumps per athlete. Sensor measurements were obtained for three to six of these jumps from three athletes each based on wind conditions, battery lifetime and personal preferences of every athlete. Unstable data connection at the ski jump hill and the jump's high landing impact further influenced the quality of the collected data. Especially the landing impact led to errors in the angular velocity readings of the ski-

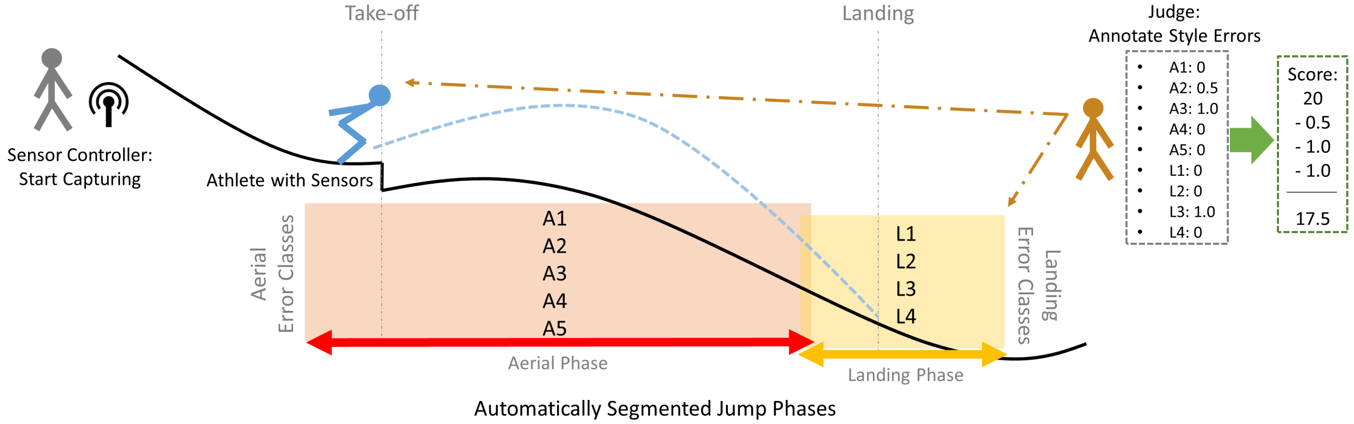

Figure 2. Schematic overview of the motion capture environment with its sensor data collection and jump error annotation procedure. The combined point deductions of every error class make up the final style score of the full jump.

mounted sensors. Therefore, from a total of 180 measured jumping motions, 85 data captures were chosen for the subsequent learning task. These were all captures that contained a complete data set of nine error-free IMU data files, as well as additional annotations on their implicit performance errors. This style information was annotated by the participating judge during data collection and complied with judging guidelines of the International Skiing Federation (FIS). Nine identifiable different classes of motion errors that were well represented within the data set were chosen from the official ski jumping rulebook [8]. These were five errors that typically occur during the aerial phase (in the following designated as $A1$ – $A5$) and four landing errors (in the following designated as $L1$ – $L4$) (Figure 2). Whereas flight errors were mostly concerned with aspects of symmetry and posture, landing errors were mostly concerned with the correct execution of the typical landing posture known as 'Telemark'. Precisely, the errors were defined as follows:

A1: Insufficient control over body or skis during the formation of the flight posture
A2: Instability of the flight posture
A3: Unsymmetrical positioning of the arms
A4: Unsymmetrical positioning of the legs
A5: Unsymmetrical positioning or unevenness of the skis
L1: No Telemark landing (landing with parallel feet)
L2: No smooth transition into the landing posture
L3: Slight Telemark landing (little bending of the knees only)
L4: Insufficient absorption of the landing impact by the Telemark or instable Telemark posture

The judging scores were given in real-time from the judging tower based on their gravity and therefore conform to judging scores as obtained in competitions. After data collection, all scores were digitized and used as error labels for network training.

PROBLEM DEFINITION

The judging of a ski jump is performed by deducting error scores from the maximal (perfect) style score of 20 points. Within one jump, multiple errors can occur that build up the final style score of a competition. Respectively, point deductions are given in steps of 0.5 points within a fixed range for every found motion error. All annotated error deductions depend on the severity of the movement error, whereas 0 points indicate no error. For example, a jump that was assigned 0.5 error points for class $A2$, 1.0 error points for class $A3$ and 1.0 error points for class $L3$ results in a final style score of 17.5 points (Figure 2). However, due to the small size of available training data, the present study was simplified to a binary classification problem defined by the pure presence or absence of specific motion style errors. This means that all data captures were given an error or non-error label with respect to all nine chosen jump error categories irrespectively of their actual numeric error value. Accordingly, nine individual classifier were trained that could then be combined for a final jump evaluation.

DATA PRE-PROCESSING

The raw sensor data was subject to internal sensor noise and distortions caused by magnetic interferences, sensor misplacement and varying starting positions. To account for this data bias, the following pre-processing steps were performed on all data streams per IMU and data capture: (a) removal of internal sensor noise, (b) sensor alignment to the bone direction of its mounted body segment using a standardized calibration measurement procedure, (c) neutralization of the sensor measurement according to its initial orientation determined using a factorized quaternion estimation algorithm and (d) segmentation of the motion streams into jump phases. All of these processing steps [5] were fully automated and could be applied to the sensor readings without manual data manipulation. For the segmentation, this meant that specific movement patterns of take-off and landing represented in the accelerometer and gyroscope measurements were utilized to automatically cut the sensor readings into an aerial phase segment (comprising the beginning of the take-off motion until the preparation of landing) and a landing phase segment (comprising the beginning of landing preparation until the maintenance of a stable Telemark posture) [6]. This segmentation was suitably applicable to the given style error definition. Finally, all sensor streams were down-sampled by a

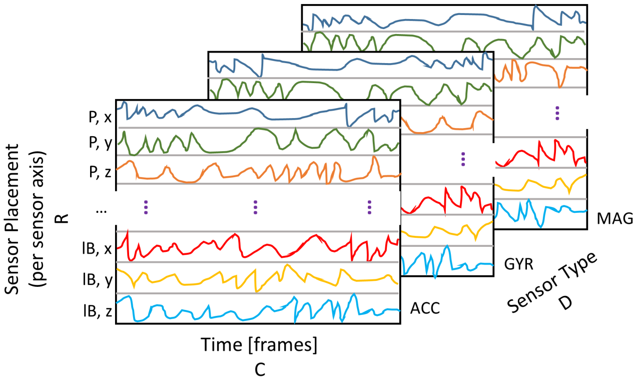

Figure 3. Sample structure of a motion image with dimension [R,C,D] built from the data captures of one ski jump using nine IMUs of three tri-axial sensor measurements (ACC, GYR, MAG) each.

factor of 2 along the temporal domain to reduce the maximal data length per phase segment. The resulting standardized and down-sampled phase-wise motion streams were then utilized as input for the subsequent learning task.

MULTI-DIMENSIONAL DEEP CNN ARCHITECTURE

In general, it should be assumed that errors in the execution of a ski jump (e.g. misplacement of the right arm) were reflected among multiple body parts (e.g. compensating movements of the upper body or left arm). Similarly, data channels obtained from the different sensor types within an IMU should be of similar information content. Therefore, it was reasonable to depict a motion performance not only along the temporal dimension, but also its body segment and sensor relationships. To learn all eventual correlations within the collected multi-dimensional time-series data, we first transformed every preprocessed data segment to a multi-channel motion image of size $[R, C, D]$ with $D = 3$ (Figure 3). This motion image was structured in the following way:

- The coordinate frames of all nine sensor placements were arranged row-wise along the image height R. Every sensor allocated three rows of the image for its respective x, y and z axis data.
- Temporal information was represented column-wise along the image width C. Every captured data sample (frame) constituted a new column.
- The three different sensor measurement types obtained by each IMU (acceleration ACC, angular velocity GYR and magnetic field data MAG) were stacked along the image color channel D.

Using the transformed motion images as training input to a convolutional network, all aspects of the respective phase measurements could then be inter-related by applying a filter of kernel $[k_r, k_c, k_d]$ to the training data with $k_r >= 2$, $k_c >= 2$ and $k_d = 3$. As in conventional deep CNNs, a stack of multiple convolutional layers furthermore provided a deep learning architecture for the motion sensor data streams.

Previous research suggests that a spectral transformation using a Discrete Fourier Transform (DFT) of the raw sensor data can achieve better classification results [13]. Additionally to the raw sensor data, we therefore also built a second motion image from the magnitude of the DFT-transformed sensor streams of every jump segment for later evaluation.

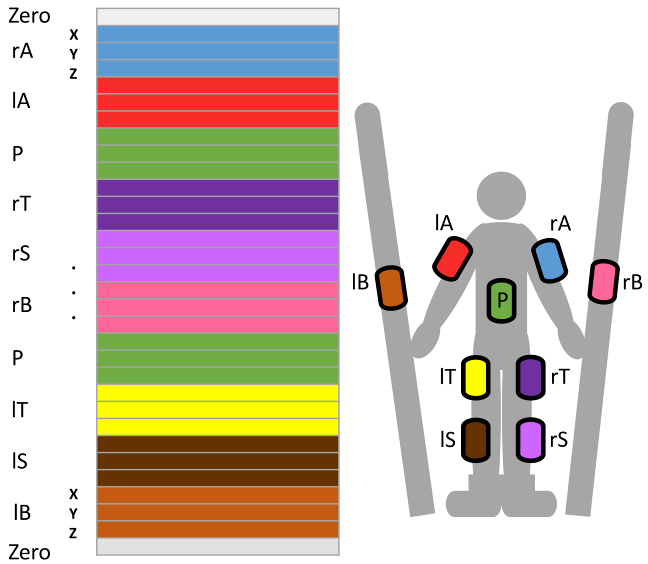

Figure 4. Arrangement of sensor measurements per sensor placement (as defined in Figure 1) and zero padding at the upper and lower boundary defining the final image height $R = 32$ of the proposed motion image.

The initial height of every motion image was defined as $R_i = 27$ according to the nine employed IMUs of tri-axial measurements each. To adjust R to a size better suited for the subsequent convolutional filter, R_i was padded to $R = 32$. This extension was implemented by adding zero padding at the boundaries and a second set of measurements from the P sensor functioning as root of the jumper-ski system (Figure 4). All data measurements were arranged in a pseudo-semantic order which ensured the proximity of every kinematic chain (arms, right and left leg including ski) to P.

To build motion phase images of uniform width, the segmented motion streams of each data capture were extended to a standardized length. For this, first the maximum phase length of all segments within the data set was determined. The respective image width was then designated as the next largest factor of 64, namely $C_A = 1024$ for the aerial and $C_L = 640$ for the landing phase. For the aerial phase, all segmented motion streams were next designated to fill the image columns $[C_A - c_i : C_A]$, with c_i constituting the length of an individual phase segment. Finally, missing sequence frames were filled with the original preceding sensor measurement samples. This padding strategy was applied because the take-off in ski jumping follows a long and fully static in-run phase. Consequently, measurements collected immediately before the initiation of the take-off were not to contain any measurements that bias the network training. For the landing phase, motion segments were designated to fill the image columns $[\frac{C_L - c_i}{2} : c_i + \frac{C_L - r_i}{2}]$ and were padded using an equal number of preceding and consecutive sensor measurements.

CNN Model Fitting

Two different multi-dimensional CNN architectures were implemented for validation of the proposed motion image: (I) a network with one convolutional, one fully connected and one softmax layer referred to as 1-mCNN and (II) a deeper CNN with three convolutional, one fully connected and one softmax layer referred to as D-mCNN. Both models were implemented in Python using Tensorflow library. The specific network parameters were defined in consideration of the given motion images and were held as similar as possible:

- The height and width of every filter kernel was kept constant as $f_r x f_c$ with $f_r = 4$ and $f_c = 8$ for every convolutional layer. The choice of f_r ensured that every convolution step comprised measurements of two different sensor units. The larger f_c constituted a contribution to the high-resolution of each measurement stream.
- Starting with $K = 10$, the number of convolutional filter K was increased by 5 in every layer.
- Dimension reduction of each network layer was achieved by a filter stride rate of $s_r = 2$ for the image height and $s_c = 4$ for the image width. As a result, a common pooling layer could be omitted.
- The first fully connected layer vectorized the output of the previous convolutional layer to a 200-dimensional feature vector. This vector was subsequently fed to the softmax layer to determine the binary error probability of every motion image.
- Every hidden network layer was followed by a ReLU activation.

Under the D-mCNN architecture, these principles reduced the input motion image to a last hidden convolutional layer of size 4×16×20 for the aerial phase, and a size of 4×10×20 for the landing phase (Figure 5). Under the one-layered 1-mCNN architecture, the input to the fully connected layer was of size 16×256×10 (aerial) respectively 16×160×10 (landing), omitting the two deeper convolutional layers of the previous architecture.

To prevent network overfitting as a cause of the small number of training data, we applied multiple regularization to the network model. A regularization loss of 0.01 was appended to the weight update step of every network layer. Additionally, every network layer was subject to a dropout with node keep probability of 50%. The learning rate of the network was chosen as exponentially decreasing learning rate in the range $[0.0001 \quad 0.003]$. During the network training, the learning rate and network parameter were not fine-tuned to facilitate comparability of results.

BASELINE NETWORKS

To put the proposed multi-dimensional CNN into context, the pre-processed sensor measurements were fed into two shallow and one deep baseline network models.

The first shallow architecture was a Support Vector Machine (SVM) with radial kernel. This method is very efficient with binary classification tasks as given by the current problem [1] and a popular method of choice for HAR scenarios. To obtain suitable data for the SVM, a set of statistical feature descriptions (Table 1) that achieved stable results in a previous study [6] was applied to all data streams of a phase segment. The resulting features were then combined into an input vector. The second shallow baseline utilized a Hidden Markov Model (HMM) for classification. This probabilistic model is a popular tool for the recognition of patterns in time-serial data [16]. The data streams of a phase segment could be used without further modifications as input to the learning model.

Deep one-dimensional CNNs were demonstrated to outperform conventional classifiers on sensor data from activity recognition data sets [20, 22]. However, they are not known to have been compared to a multi-dimensional network as proposed in this work. Therefore, a simplified, one-dimensional deep CNN following the network described by Yang et al. [25] was implemented that only correlated the temporal aspect of the motion data. For this, the data segments of all sensor types and axes were row-wise stacked without consideration of the color channel. This led to an image height of $R = 81$. Additionally, the image width was padded to C_A and C_L and a filter of kernel size $[1, k_c, 1]$ ap-

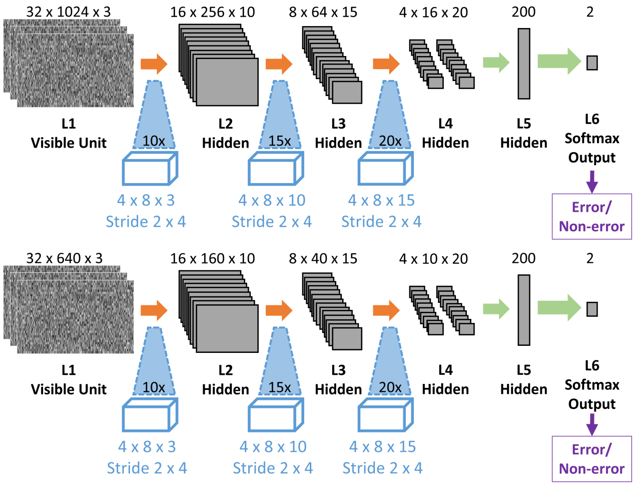

Figure 5. Network architectures for the deep multi-dimensional CNN. Layer L1 represents the motion image input, layer L2 to L5 the three convolutional and one fully connected hidden layer and L6 the softmax output layer. Top: aerial phase. Bottom: landing phase.

ID	Type	Description
F_{D1}	mean(a, g, m)	Signal mean
F_{D2}	std(a, g, m)	Signal variance
F_{D3}	skew(a, g, m)	Signal skewness
F_{D4}	curt(a, g, m)	Signal kurtosis
F_{D5}	acor(a, g, m)	10 samples of signal autocorrelation sequence
F_{D6}	zcr(a, g, m)	Zero crossing rate (ZCR)
F_{D7}	mcr(a, g, m)	Mean crossing rate (MCR)
F_{D8}	pow(fft(a, g, m))	Power of the spectrum obtained with the signal FFT

Table 1. Statistic features applied to the pre-processed sensor measurements of accelerometer (a = $[a_x \; a_y \; a_z]$), angular velocity (g = $[g_x \; g_y \; g_z]$) and magnetometer (m = $[m_x \; m_y \; m_z]$) for training of the baseline SVM.

plied under the same principles as for the multi-dimensional CNN. For the following evaluations, this temporal CNN shall be referred to as CNN-1D.

TRAINING DESIGN

The training and evaluation of a respective error assessment system is a challenging task: for safety reasons, athletes could not be requested to intentionally perform certain motion errors. Consequently, a well-balanced data base for the subsequent machine learning task could not be ensured. To account for the comparably small number of sample data and prevent bias in the database split, every network model evaluated the data collection in an 8-fold cross-validation cycle per motion error. The split within each cross-validation cycle was randomly determined in accordance with the annotated error labels. This procedure ensured that every cross-validation cycle contained a minimum number of error and non-error motion segments.

RESULTS

For evaluation of our proposed network model, we first compared the deep multi-dimensional CNN to the one-layered model variation. Simultaneously, we investigated differences in error classification performance between the motion images built from the raw sensor data and the motion images built from the Fourier-transformed sensor data. As a second step, both network models were compared to the three chosen baseline methods. In all cases, the error classification accuracy and cross-entropy loss averaged over all 8 cross-validation cycles as well as the respective standard deviation served as primary validation measures.

Model Design

Accuracy values of the proposed D-mCNN model show that the learned network is well capable to recognize motion errors within the testing data (Table 2). Throughout all nine learned errors categories, classification rates of the network that utilizes raw sensor data as input were between 71% and 84%. Accuracy rates obtained using the 1-mCNN model are even higher, reaching up to 92%. This suggests that already one multi-dimensional convolutional layer is sufficient to learn representative motion correlations.

Commonly, one would expect the deeper network with larger receptive fields to be more powerful than the 1-mCNN model. One reason for the given reverse accuracy distribution could be that the chosen learning rate was better suited for the given training data. Another reason could be that the 1-mCNN model is less prone to overfitting as a result of its larger number of filter weights. Indeed, error classification with the 1-mCNN model shows a smaller cross-entropy loss than the error classification with the D-mCNN model (Table 3). However, the proposed D-mCNN model shall not be disapproved completely at this point: as discussed earlier, the given data collection constitutes a very small and uncontrolled data set. Higher accuracy values might be achieved by fine tuning of the network parameters or by use of a larger field data set in future.

Different than in previous research, data furthermore does not indicate a clear advantage of the spectral DFT motion images over the raw sensor input. Here, it is likely that the spectral features of a specific (mostly instantaneous) motion error are not more discriminative than its raw signal representation.

	D-mCNN	1-mCNN	D-mCNN DFT	1-mCNN DFT
A1	0.87 ±0.13	0.93 ±0.08	0.84 ±0.09	0.94 ±0.07
A2	0.77 ±0.08	0.81 ±0.08	0.78 ±0.05	0.83 ±0.05
A3	0.85 ±0.10	0.87 ±0.04	0.85 ±0.05	0.90 ±0.07
A4	0.75 ±0.08	0.80 ±0.07	0.69 ±0.07	0.78 ±0.05
A5	0.80 ±0.07	0.87 ±0.08	0.82 ±0.09	0.90 ±0.09
L1	0.81 ±0.07	0.87 ±0.07	0.81 ±0.07	0.85 ±0.06
L2	0.71 ±0.12	0.80 ±0.06	0.80 ±0.12	0.83 ±0.07
L3	0.78 ±0.04	0.79 ±0.09	0.74 ±0.08	0.81 ±0.09
L4	0.78 ±0.05	0.81 ±0.09	0.80 ±0.10	0.80 ±0.07

Table 2. Accuracy of every error label averaged over all cross-validation cycles and the corresponding standard deviation for the D-mCNN and 1-mCNN network model using motion images based on raw and spectral sensor input data.

	D-mCNN	1-mCNN	D-mCNN DFT	1-mCNN DFT
A1	8.5 ±3.7	7.0 ±2.8	9.3 ±4.2	5.8 ±1.7
A2	13.7 ±3.1	11.3 ±3.5	10.8 ±1.2	10.0 ±3.2
A3	9.4 ±3.8	8.3 ±3.9	9.4 ±3.1	8.1 ±3.0
A4	14.4 ±4.0	11.3 ±4.1	14.9 ±4.7	13.8 ±4.6
A5	10.4 ±2.9	7.5 ±2.4	10.4 ±2.3	8.8 ±2.5
L1	11.6 ±3.9	10.9 ±3.9	13.4 ±4.5	12.4 ±4.3
L2	11.3 ±2.8	10.1 ±3.0	11.7 ±2.6	10.0 ±2.6
L3	14.1 ±5.8	14.8 ±4.4	15.3 ±3.5	13.0 ±5.6
L4	15.7 ±8.6	9.9 ±3.2	13.8 ±8.7	10.8 ±2.5

Table 3. Cross-entropy loss of every error label averaged over all cross-validation cycles and the corresponding standard deviation for the D-mCNN and 1-mCNN network model using motion images based on raw and spectral sensor input data.

Model Efficiency

Since the DFT-based motion images did not show any distinct advantages, we only used the raw sensor data for the following baseline comparisons. Listing the accuracy of all classification methods per motion error, especially one major difference between the shallow and convolutional architectures becomes visible. While the SVM and HMM perform similarly as D-mCNN, 1-mCNN and CNN-1D for the aerial style errors $A1$ to $A4$, $A5$ and all landing errors $L1$-$L4$ cannot be recognized correctly with the same reliability (Figure 6). A closer look into the underlying data annotations seems to explain this difference well: specific landing error categorization are of high semantic similarity and differ vaguely in their descriptions only. Similarly, analysis of the underlying data revealed bias and ambiguity in the error labels awarded by the volunteering judge. Therefore, one can assume that the training data given for each landing errors were not distinct enough to be learned by the shallow systems.

In contrast to the shallow networks that utilize hand-crafted features and sequential probabilities, data quality did not considerably influence any of the convolutional network architectures. Consequently, the present study emphasizes the benefit of intrinsically learned (deep) feature representations for HAR more explicitly than similar investigations. Given that most former investigations were performed on activity recognition data sets acquired under laboratory conditions, partic-

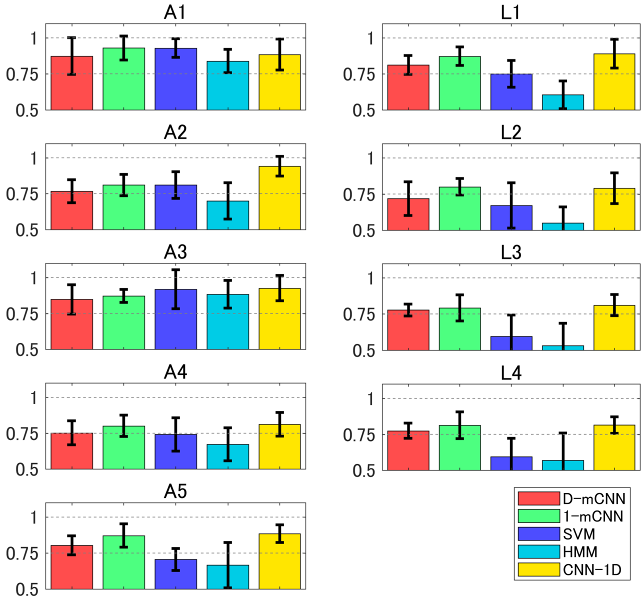

Figure 6. Accuracy values per error category for the proposed D-mCNN and 1-mCNN model as well as the three chosen baseline values and the corresponding standard deviation.

ularly noisy and biased field data seem to benefit from automatic feature learning of the CNN architecture.

Finally comparing accuracies of the convolutional architectures only, one can see that the CNN-1D baseline network does not perform worse than the multi-dimensional CNNs. In contrary, it reaches even better error classification results than the D-mCNN network. Inspections of the cross-entropy loss during network training suggest that the respective network training process is smoother than for the proposed multi-dimensional CNN models (Figure 7). Here, it might be possible that the given data collection is not extensive enough to represent all variations of different skeletal and sensor measurement type features. Since respective interrelations are not included in the one-dimensional model, a resulting network training might converge faster and hence be trained easier than the multi-dimensional network. However with the present data set, this assumption cannot be confirmed and verification is subject to further investigations with a different or extended data set.

CONCLUSION

In this work, we presented a human activity recognition system for wearable motion sensor data specialized in the error classification of ski jumping motions.

In particular, we developed a multi-dimensional convolutional network model based on the idea of a three-dimensional motion image connected along the temporal, skeletal and sensor domain. To date, such highly inter-related data representation is not known to be applied to similar wearable sensor data sets. Results show that the proposed model is well applicable for use in 'wild' motion data collected in actual sporting environments. Already one convolutional layer based on the proposed three-dimensional motion image appears sufficient for reliable classification of motion errors and could foster the future implementation of motion style judging systems. However, comparisons to previously developed one-dimensional convolutional network models do not show further enhancement of classification accuracy. This missing improvement might be subject to the structure and quality of the underlying data collection. Therefore, it appears reasonable to further investigate the application of multi-dimensional HAR network models with different or extended data sets.

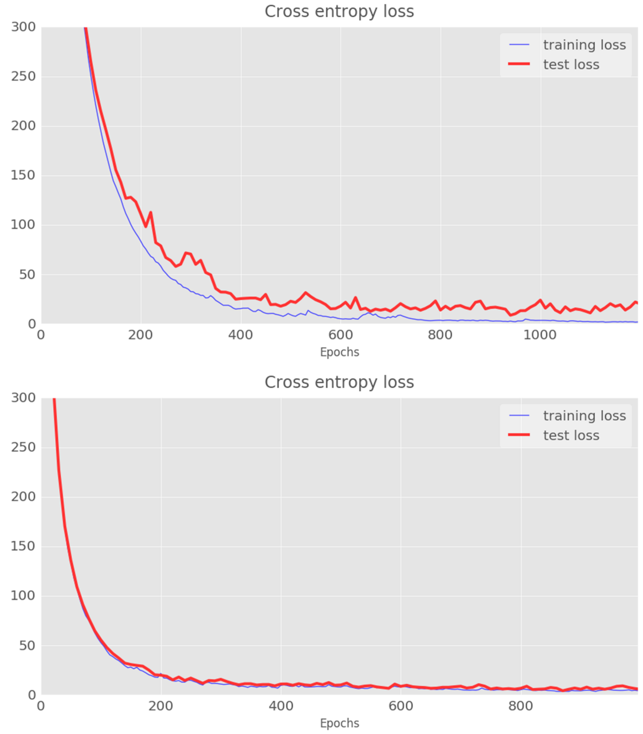

Figure 7. Sample evolution of cross-entropy over the network training process loss for the same error and cross-validation fold. Top: D-mCnn network model. Bottom: CNN-1D network model.

Comparisons to a SVM and HMM show that all convolutional models perform significantly better under noisy and biased sensor data. This indicates that the general error classification rate could especially be increased for motion categories under which performance errors were not discriminated by the conventional, shallow networks. Consequently, the utilization of convolutional networks may considerably improve general robustness and reliability of activity recognition systems. In particular motion analysis systems that are based on wearable sensor data captured under actual field conditions should benefit from the utilization of advanced learning techniques.

All in all, this work demonstrates the applicability and usability of convolutional neural networks on inertial motion sensor data captured under field conditions of a sport environment. With little pre-processing required, it was possible to achieve a superior level of error recognition that has not been possible to date. When compared to shallow network technologies, convolutional networks brought the proposed judging system closer to the status quo in human motion style assessment, and might eventually also contribute to even better and more objective judging in the future.

ACKNOWLEDGMENTS

The acquisition of the inertial sensor data was supported by JSPS KAKENHI,Grant-in-Aid for Scientific Research (B) under Grant No.:25282197.

REFERENCES

1. Abe, S. *Support vector machines for pattern classification*, vol. 2. Springer, 2005.
2. Bao, L., and Intille, S. Activity recognition from user-annotated acceleration data. *Pervasive computing* (2004), 1–17.
3. Bashivan, P., Rish, I., Yeasin, M., and Codella, N. Learning representations from eeg with deep recurrent-convolutional neural networks. *arXiv preprint arXiv:1511.06448* (2015).
4. Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 8 (2013), 1798–1828.
5. Brock, H., and Ohgi, Y. Development of an inertial motion capture system for kinematic analysis of ski jumping. *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*, [Online] (2016), 1–12.
6. Brock, H., and Ohgi, Y. Assessing motion style errors in ski jumping using inertial sensor devices. *IEEE Sensors Journal 17*, 12 (June 2017), 3794–3804.
7. Bulling, A., Blanke, U., and Schiele, B. A tutorial on human activity recognition using body-worn inertial sensors. *ACM CSUR 46*, 3 (2014), 33.
8. FIS. *The international ski competition rules (ICR). Book III. Ski jumping*. FIS, 2013.
9. Hammerla, N. Y., Halloran, S., and Ploetz, T. Deep, convolutional, and recurrent models for human activity recognition using wearables. *arXiv preprint arXiv:1604.08880* (2016).
10. Helten, T., Brock, H., Müller, M., and Seidel, H.-P. Classification of trampoline jumps using inertial sensors. *Sports Engineering 14*, 2-4 (2011), 155–164.
11. Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9*, 8 (1997), 1735–1780.
12. Holden, D., Saito, J., Komura, T., and Joyce, T. Learning motion manifolds with convolutional autoencoders. In *SIGGRAPH Asia 2015 Technical Briefs*, ACM (2015), 18.
13. Jiang, W., and Yin, Z. Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proc ACM IC MM*, ACM (2015), 1307–1310.
14. LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature 521*, 7553 (2015), 436–444.
15. Logical Product. Sports sensing 9-axial waterproof inertial sensor. `http://www.sports-sensing.com/products/motion/motionwp01.html`. Accessed: 2017-01-23.
16. MacDonald, I. L., and Zucchini, W. *Hidden Markov and other models for discrete-valued time series*, vol. 110. CRC Press, 1997.
17. Marqués Bruna, P., and Grimshaw, P. Mechanics of flight in ski jumping: Aerodynamic stability in pitch. *Sports Technology 2*, 1-2 (2009), 24–31.
18. Marqués Bruna, P., and Grimshaw, P. Mechanics of flight in ski jumping: aerodynamic stability in roll and yaw. *Sports Technology 2*, 3-4 (2009), 111–120.
19. Müller, W. Determinants of ski-jump performance and implications for health, safety and fairness. *Sports medicine 39*, 2 (2009), 85–106.
20. Ordóñez, F. J., and Roggen, D. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors 16*, 1 (2016), 115.
21. Plötz, T., Hammerla, N. Y., and Olivier, P. Feature learning for activity recognition in ubiquitous computing. In *PROC IJCAI*, vol. 22 (2011), 1729.
22. Ronao, C. A., and Cho, S.-B. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Systems with Applications 59* (2016), 235–244.
23. Schmölzer, B., and Müller, W. Individual flight styles in ski jumping: results obtained during olympic games competitions. *Journal of Biomechanics 38*, 5 (2005), 1055 – 1065.
24. Wulf, G., Shea, C., and Lewthwaite, R. Motor skill learning and performance: a review of influential factors. *Medical education 44*, 1 (2010), 75–84.
25. Yang, J., Nguyen, M. N., San, P. P., Li, X., and Krishnaswamy, S. Deep convolutional neural networks on multichannel time series for human activity recognition. In *PROC IJCAI* (2015), 3995–4001.
26. Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., and Zhang, J. Convolutional neural networks for human activity recognition using mobile sensors. In *MobiCASE*, IEEE (2014), 197–205.
27. Zhang, M., and Sawchuk, A. A. A feature selection-based framework for human activity recognition using wearable multimodal sensors. In *PROC IC BAN*, ICST (2011), 92–98.
28. Zitzewitz, E. Nationalism in winter sports judging and its lessons for organizational decision making. *Journal of Economics & Management Strategy 15*, 1 (2006), 67–99.