Classifying Martial Arts Motion from a Single Wearable Sensor

Christopher Samiullah and **Sam Devlin** and **Urs Suess**
christopher.samiullah@gmail.com, sammdevlin@gmail.com, urssuess@gmail.com

Abstract

We describe the design and implementation of a wearable sensor and its data streams with applications in grappling martial arts. We classify eight martial arts positions using sensor data from one compound device possessing a gyroscope and accelerometer. The algorithm development process includes 90 minutes of labelled training data, and the evaluation of several machine learning algorithms. Combining Random Forest classifiers with a Hidden Markov Model method is shown to significantly improve the classification performance. All data-sets and application code have been made available publicly.

I. Introduction

Wearable inertial sensors such as accelerometers and gyroscopes are becoming widespread, as most modern smartphones come equipped with them. These sensors can be used to analyze human motion in a variety of contexts, from healthcare to entertainment. In this paper we consider applications for sports, specifically grappling martial arts.

There is a growing demand for performance evaluation tools which leverage inertial data to provide practitioners with rapid feedback on their technique and training habits. Such coaching devices used to be the tools of only elite athletes due to prohibitive costs, but have become more accessible due to lower sensor prices. This growing demand is evidenced by the emergence of multiple mainstream wearable companies in this space in recent years [21][22][23]. To provide useful feedback to users, it is often necessary to understand subtle movement patterns and sequences for a given activity. Distinguishing between subtly different movements and positions is challenging.

This paper lays out an approach for overcoming the above mentioned classification challenges in the realm of grappling martial arts, chiefly submission wrestling and Brazilian Jiu-Jitsu. We built a prototype wearable for practitioners, and ran extensive tests to optimize for key position detection. Crucially, the prototype was built to be commercially viable, which meant that we only used one motion sensor due to safety and cost constraints. This had the effect of keeping the projected manufacturing costs low, but made the classification challenge far harder. We have made all our recorded data and application code publicly available[24], along with video demonstrations of the prototype being used in a variety of live scenarios [25].

II. Related Work

Research into Human Activity Recognition can be clustered into two main areas: analysis of video recordings [3][4][19], and the use of inertial sensors [1][2][7]. Our work falls into the inertial sensor category.

Multiple sensors attached to different body positions are the most common solutions[12], with accelerometers being the most frequently used type of sensor deployed [1]

Sensor fusion, particularly of accelerometers and gyroscopes, is widely acknowledged as resulting in higher accuracy when conducting motion analysis on time series data from inertial sensors [6][8][20], while the usefulness of adding magnetometer data remains unclear [6]. We selected the fusion of accelerometer and gyroscope data.

Regarding algorithm composition, the use of both ensemble learning methods, such as Random Forest [1][14] and Hidden Markov Models [5][13] for classifying human motion from accelerometer data has been explored, but their use together is relatively obscure.

Other commonly used classification methods for human motion include Support Vector Machines, K-Nearest Neighbor, and Naive Bayes [6], and we considered these in our initial experiments.

In the following sections we describe how we created the prototype wearable for exploring these motion classification approaches, followed by results.

III. Prototype Creation

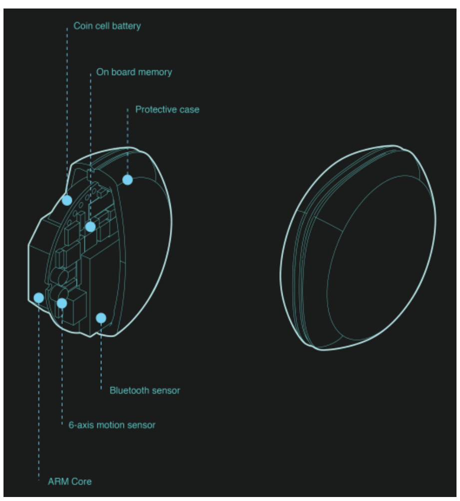
We set out to develop a wearable training shirt that could effectively classify eight key positions from Brazilian Jiu-Jitsu during training activity:

1: Your guard
2: Opponent guard
3: Your back control
4: Opponent back control
5: Your mount
6: Your side control
7: Opponent mount/side control
8: Other

These positions represent the fundamental ground control positions. We specified “Other” for training motion that did not match the above positions (e.g. intricate sub-positions, or “scrambles”) and also motion that did not represent training (e.g. standing around watching, sitting doing nothing).

Grappling martial arts consistently place an extremely high level of stress on the user (and hence the wearable). Furthermore, we were interested in testing a prototype with a potential mass-market price range. These constraints meant that we opted to embed just a single sensor in the wearable, thus allowing for greater safety, comfort and a potentially lower retail price. Primarily due to safety concerns (because limbs are constantly grabbed and rubbed), we opted to place this sensor slightly below the center of the torso. The torso area is a location that has been shown to be effective for classifying activities such as standing and squatting [1][12], and placing the sensor below the sternum increased comfort for users. We used a commercially available Bluetooth Low Energy sensor, the “MetaWear C” from mbientlab, which has a board size of 24mm diameter x 2.0mm, and an inbuilt Bosch BMI160 triaxial accelerometer and gyroscope [15].

Figure 1: Prototype Sensor and Case Components



Time series inertial data was streamed from the board’s accelerometer and gyroscope to an Android mobile application, using the mbientlab Android API [16]. The sensor was placed in a padded neoprene ‘pouch’, inside a commonly worn skin tight t-shirt called a rashguard. The pouch had dimensions of 7cm x 7cm, and was located approximately 5cm below the sternum. The pouch and sensor were marked to ensure that the sensor orientation was consistent throughout testing.

To begin recording motion, users would open the Android app we developed and activate the sensor’s accelerometer and gyroscope. This would then trigger the sensor readings to be streamed and saved to the phone. Once the user finished their training session, they would send their logged data for analysis on our server via HTTP.

Our data analysis server was written in Python, due to the maturity of both its machine learning libraries and the Scientific Python stack. The server ran the uploaded user data through our classification algorithm and returned a breakdown of the amount of time spent in the respective eight positions for the given training session. The training principle here is that allowing a practitioner to better understand how they are spending their time whilst training will enable them to optimize their key focus areas and address their weaknesses, a practice that has been noted in studies on high level athletes in other sports [26].

IV. Data Collection and Processing

Approximately ninety minutes of time-series sensor data was collected and manually labelled for training purposes. Training data was collected one classification position at a time, allowing for simple batch labelling. Additional test data with mixed positions and transitions was collected with a stop watch and video camera. Data was collected from three separate male practitioners of similar size (170-180cm, 60-70kg) and age (28-35).

The data was streamed at a frequency of 25Hz. There are studies where similar frequencies have been shown to be sufficient for activity recognition [8][9], though 50Hz is more common. In our case we opted to use this frequency to better preserve the coin-cell battery on the sensor.

A key part of the classification process is the search for effective data features. Previous similar research has shown the importance of selecting the correct size of sliding window during data sampling. We opted for 1.6 seconds, with a 50% overlap, based on previous studies [5][10][11]

Feature selection was a crucial part of our work, and the result of much gradual optimization. The accelerometer and gyroscope both report values along their three dimensions (x, y, z), with the accelerometer reporting acceleration in meters per second squared (m/sec^2) and the gyroscope showing the rate of rotations in radians per second (rad/sec) along each axis.

We focused on standard statistical features, and best practice motion analysis data features [20]:

- Mean
- Median
- Max/Min
- Sum
- Standard deviation
- Tilt
- Magnitude
- Root sum square
- Root mean square

All calculations were done on a rolling window basis. Standard statistical data features were applied to raw data from all three axes of both the accelerometer and the gyroscope. Magnitude, root sum square and root mean square were recorded by sensor, giving two data features each.

We also took polynomials of degree three for every feature, although it did impact the speed of the classifier (which is why we did not attempt to use polynomials of a higher degree). In total, we worked with 54 data features (162 when including in the use of polynomials).

Table 1 summarizes the ten features with the most significant impact on algorithm performance:

Table 1: Ten Most Significant Data Features

Data Feature	Impact on Classification
Accelerometer Z-Axis rolling mean	6.9%
Accelerometer Z-Axis rolling sum	6.5%
Accelerometer Z-Axis rolling maximum	6.3%
Accelerometer Z-Axis rolling median	5.4%
Accelerometer Z-Axis rolling minimum	5.2%
Accelerometer X-Axis rolling sum	3.9%
Accelerometer X-Axis rolling mean	3.8%
Accelerometer X-Axis tilt	3.6%
Accelerometer X-Axis rolling minimum	3.1%
Accelerometer X-Axis rolling max	3.1%

V. Algorithm Composition

In order to analyze and classify the pre-processed data, we applied different classification algorithms to compare performance. We used 10-fold cross-validation to balance our accuracy calculations. We evaluated the following commonly used classifiers from previous human motion recognition research [6]:
qLogistic regression, K Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, Naïve Bayes and Adaboost.

We evaluated the performance of these classifiers using the scikit-learn cross_val_score method, checking for accuracy. The results are summarised in Table 2 below:

Table 2: Classification Algorithm Accuracy Comparison

Classification Algorithm	Mean Accuracy from 10-fold Cross Validation	Accuracy 95% confidence interval
Logistic Regression	59%	+/- 13%
K Nearest Neighbors	44%	+/- 22%
Support Vector Machine	34%	+/- 3%
Decision Tree	66%	+/- 21%
Random Forest	72%	+/- 20%
Naïve Bayes	31%	+/- 12%
AdaBoost	59%	+/- 19%

The above results were gathered after significant parameter optimization, with the optimal number of nodes for the Random Forest classifier falling at approximately 5000. The Random Forest algorithm gave us the highest and most consistent levels of accuracy, which other research has also found [1][2][18]

Hidden Markov Model

Despite the overall high accuracy of the Random Forest algorithm for classifying different body positions, high false-positive rates were detected for specific grappling positions which, in terms of their position in 3D space, are very similar.

Figure 2: Comparison of Accelerometer Z-Axis Readings

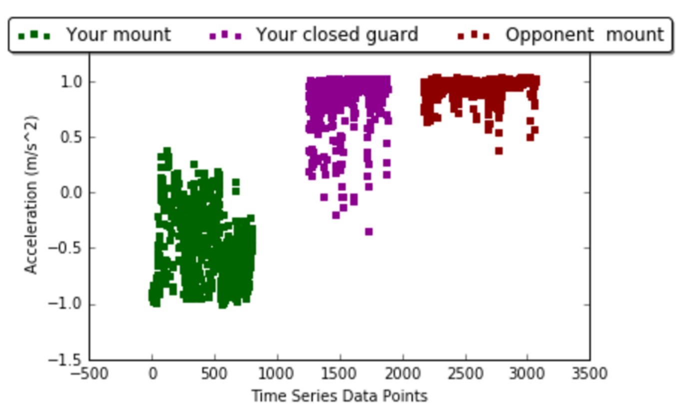
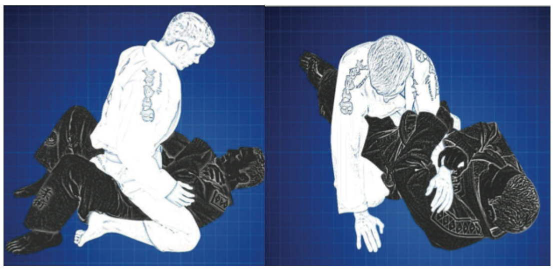


Figure 2 illustrates this challenge with raw data from the accelerometer z-axis: The "Your mount" and "Your closed guard" positions are clustered differently, whereas "Your closed guard" and "opponent mount" overlap a great deal. However, these overlapping positions in grappling martial arts are markedly different, and the inability to differentiate between them would render the wearable highly impractical. An example of this is shown in Figure 3, where if the user in white is wearing the sensor, in the picture on the left, they are in a dominant position (mount), and in the picture on the right they are in a neutral position (guard). However, white's position in space is very similar, as in both instances he/she is on their knees, upright.

Figure 3: Challenging Positions to Discern



We sought a way to address this challenge, and generally improve the overall accuracy of our method, by injecting some context awareness and domain-specific knowledge. Since our prototype was not required to give real-time predictions, we were able to retrospectively correct certain unlikely errors using a Hidden Markov Model approach. For this we used the *hmmlearn* library, which is set of Hidden Markov Model algorithms written in Python [17].

We setup a Multinomial Hidden Markov Model since we were working with discrete existing prediction data. We assumed that our predictions operated as a discrete Markov chain, where the random forest outputs are the observations, and the correct classifications are the hidden states. This led us to assign the following key attributes of the model:

- N components: The number of our classification states (eight)
- Start probabilities: an N x 1 matrix of the likelihood of a given state being the initial state
- Transition matrix values: N x N matrix
- Emission matrix values: N x N matrix

We made use of the Viterbi algorithm [27] to decode the sequence of observations to find the hidden states (in this case, the eight motion classifications). With the Viterbi algorithm, every observation value from the random forest classifier is re-evaluated based on the previous observation. Since many grappling positions occur in predictable sequences (with some sequence permutations being highly unlikely), we used our domain specific knowledge to assign probability weightings for a) transitions between positions and b) a given position "emitting" the next position. This additional step allowed us to take the wider context into account to a greater degree than the Random Forest classifier on its own.

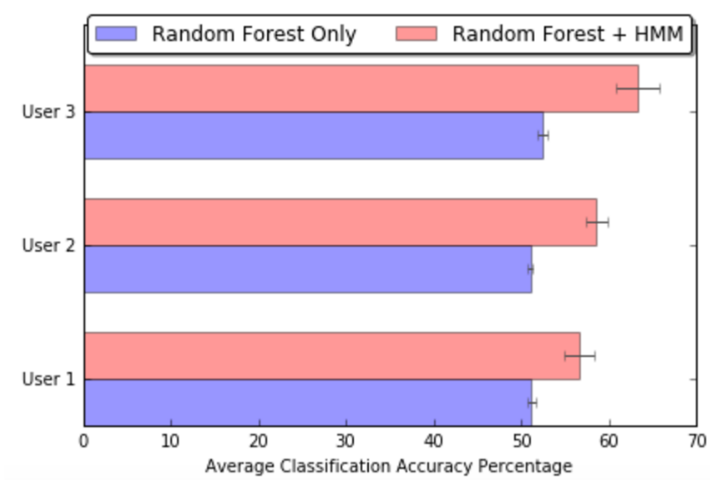
Future Post Processing

The key draw back of the HMM adjustment is that if the initial value given is incorrect, then the "corrections" accentuate errors, rather than reducing them. One technique we explored to reduce the likelihood of this (highly undesirable) possibility, was isolating particular motion sequences and then scanning for that particular sequence for all data classified a certain way. For example, instances of a user standing up only occur during specific positions. If a stand up occurs and the initial classification is not of a position where stand-ups typically occur, it is likely a misclassification has occurred.

VI. Results

The comparison shown below in Figure 4 was based on a specifically created test sequence, which three test users were recorded moving through. This test sequence involved all eight martial arts positions over the course of approximately two minutes. It tested classification accuracy in a much more realistic setting, including transitions between different positions. This represented a highly challenging classification sequence (resulting in lower classification accuracy generally).

Figure 4: Accuracy Comparison of Random Forest only and Random Forest Combined with HMM



Classifications were run fifty times for each data set and averaged. The results show an improvement in average accuracy of 8% with HMM post-classification adjustment.

The standard error from the mean for the results in Figure 4 was between 0.5% and 2%. A paired t-test was conducted on the average accuracy results for the random forest test only and the random forest combined with HMM. Taking the null hypothesis that the two methods have the same mean accuracy, the t-test p values summarised in Table 3 show that we can reject the null hypothesis with strong certainty.

Table 3: T-Test p Values and Cohen's d Value Summary

User	T-Test p Value	Cohen's d Value
User 1	0.00315	2.76
User 2	1.86×10^{-8}	5.07
User 3	0.000169	3.27

VII. Conclusion and Further Research

Motion classification using the approach outlined in this paper, combining a Random Forest method with a Hidden Markov optimisation, yields a significantly higher level of accuracy than using the Random Forest in isolation. Higher levels of accuracy allow for more nuanced motion analysis, which in turn enables practitioners to optimise their training with the aid of wearable technology.

Although the prototype built addresses a very specific activity, the techniques deployed are widely applicable. These findings were collected from a single sensor because grappling martial arts place so much stress on the body and wearable technology. Many other sports or activities would afford opportunities to use the same techniques with multiple sensors. Furthermore, the collection and labelling of data for martial arts is particularly time-consuming and challenging. Other activities would be able to easily acquire larger datasets, which would further increase accuracy and might open up new techniques requiring larger data quantities, such as recurrent neural networks.

Further study is required to understand the impact of very different body shapes on accuracy, as test participants were of a similar size, and were all male. Our technique of isolating specific motions and retrospectively updating motion predictions based on their recent detection also showed promise, and is a method that could also prove useful in the classification of other activities.

References

[1] Ravi, N., Nikhil, D., Mysore, P., Littman, M.L. 2005. Activity recognition from accelerometer data. In *Proceedings of Innovative Applications of Artificial Intelligence*, pp. 1541–1546

[2] Ellis, Katherine, et al. 2014. A random forest classifier for the prediction of energy expenditure and type of physical activity from wrist and hip accelerometers. *Physiological measurement* 35.11: 2191.

[3] T.B.Moeslund; A.Hilton; V. Kruger, 2006. A survey of advances in vision-based human motion capture and analysis, *Computer Vision Image Understanding* 104 (2-3) 90–126

[4] Bodor, R.; Jackson, B.; and Papanikolopoulos N. 2003. Vision-based human tracking and activity recognition. In *Proceedings of the 11th Mediterranean Conference on Control and Automation*, June 2003

[5] Mannini, Andrea and Sabatini Angelo. 2010. Machine Learning Methods for Classifying Human Physical Activity from On-Body Accelerometers. *Sensors* 10, 1154-1175.

[6] Shoaib, Muhammad; Scholten, Hans; Havinga P. 2013. Towards Physical Activity Recognition Using Smartphone Sensors *Institute of Electrical Engineers 10th International Conference on Ubiquitous Intelligence & Computing* and *IEEE 10th International Conference on Autonomic & Trusted Computing*

[7] Incel, Ozlem Durmaz, Mustafa Kose, and Cem Ersoy. 2013. A review and taxonomy of activity recognition on mobile phones. *BioNanoScience* 1-27.

[8] Wu, Wanmin; Dasgupta, Sanjoy; Ramirez, Ernesto E; Carlyn Peterson, and Gregory J. Norman. 2012. Classification accuracies of physical activities using smartphone motion sensors. *Journal of medical Internet research* 14, no. 5: e130.

[9] Anjum, Alvina, and Muhammad U. Ilyas. 2013. Activity recognition using smartphone sensors. *In Consumer Communications and Networking Conference* (CCNC) pp. 914-919.

[10] Preece, Stephen J., John Yannis Goulermas, Laurence PJ Kenney, and David Howard. 2009. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. *Biomedical Engineering*, IEEE Transactions on 56, no. 3 (2009): 871-879.

[11] L. Bao and S. S. Intille. 2004. Activity recognition from user-annotated acceleration data. *Personal Computing*, Lecture Notes in Computer Science, vol. 3001, pp. 1--17

[12] Daniel Olgu Olgu, Alex (Sandy) Pentland. 2006. Human Activity Recognition: Accuracy across Common Locations for Wearable Sensors. *IEEE 10th International Symposium on Wearable Computers*. Montreaux, Switzerland. October, 2006.

[13] Ghahramani, Z. 2001. An Introduction to Hidden Markov Models and Bayesian Networks. *International Journal of Pattern Recognition and Artificial Intelligence*. 15(1):9-42

[14] Shotton J, Fitzgibbon A, Cook M, Sharp T, Finocchio M, Moore R, Kipman A, Blake A. 2011. Real-time human pose recognition in parts from single depth images. *Proceedings of the Conference on Computer Vision and Pattern Recognition.* pp. 1297–1304.

[15] Metawear C specifications available at

https://mbientlab.com/metawearc/ [Accessed on 23rd August 2016].

[16] Mbientlab Android API code base available at: https://github.com/mbientlab/Metawear-AndroidAPI [Accessed: 24th August 2016]

[17] hmmlearn source code and documentation available at: https://github.com/hmmlearn/hmmlearn [Accessed: 24th August 2016]

[18] Casale, Pierluigi, Oriol Pujol, and Petia Radeva. 2011. Human activity recognition from accelerometer data using a wearable device. *Iberian Conference on Pattern Recognition and Image Analysis. Springer Berlin Heidelberg*.

[19] Clarkson, B., Pentland, A. 1999. Unsupervised Clustering of ambulatory audio and video. In: *International Conference on Acoustics, Speech, and Signal Processing* pp. 3037–3040

[20] Sachs, David. 2010. Sensor Fusion on Android Devices: A Revolution in Motion Processing. Google Tech Talks. Uploaded on 8 Aug 2010. https://www.youtube.com/watch?v=C7JQ7Rpwn2k [Accessed 24th August, 2016]

[21] Zepp

http://www.zepp.com/en-us/

[Accessed 24th August 2016]

[22] Athos

https://www.liveathos.com/

[Accessed 24th August 2016]

[23] Hexoskin

http://www.hexoskin.com/

[Accessed 24th August 2016]

[24] Code and data available at this github repository: [url: https://github.com/ChristopherGS/rolltec_motion]. Accessed 10th September, 2016.

[25] *RollTec:* [url: https://www.kickstarter.com/projects/1489178162/rolltec-grappling-the-first-wearable-made-for-grap]

[Accessed 26th August 2016]

[26] Natalie Durand-Bush and John H. Salmela. 2002.

The Development and Maintenance of Expert Athletic Performance: Perceptions of World and Olympic Champions

Journal Of Applied Sport Psychology. Vol. 14 , Iss. 3

[27] Viterbi AJ. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory.* 13 (2): 260–269.