

*Proceedings*

# Novel Experimental Protocol to Capture Movement Data and Predict Shot Execution in Cricket Batting <sup>†</sup>

Pubudu Dias <sup>\*</sup>, Séan R. Mitchell and Andy R. Harland

Sports Technology Institute, Wolfson School of Mechanical, Electrical and Manufacturing Engineering, Loughborough University, Loughborough LE11 3QF, UK; s.r.mitchell@lboro.ac.uk (S.R.M.); a.r.harland@lboro.ac.uk (A.R.H.)

<sup>\*</sup> Correspondence: w.p.p.dias@lboro.ac.uk; Tel.: +44-753-910-6568

<sup>†</sup> Presented at the 13th Conference of the International Sports Engineering Association, Online, 22–26 June 2020.

Published: 15 June 2020

**Abstract:** Shot execution in cricket batting is reliant on intricate movement patterns of crucial body segments. When there is a substantial amount of batting movement data available, supervised machine learning can be used to classify when a batting shot execution takes place in a cricket batting cycle. An automated approach to identify and assess cricket batting could be useful for the applications including performance evaluation, talent identification and injury prevention. Current evaluation of movements and shot execution are generally undertaken in an artificial environment with camera-based, motion tracking systems to collect batting movement data, which require careful preparation, data collection and post-processing, and risk changing the natural gameplay of a batsman. By training a model based on data obtained from a close representation of a cricket batting innings, supervised machine learning was found to be capable of reliably predicting cricket batting shot execution.

**Keywords:** cricket; cricket batting; inertial measuring units; IMU; supervised machine learning; accuracy; F-score

---

## 1. Introduction

Effective stroke play in cricket requires the batsman to execute a combination of a complex body segment movements with high spatial and temporal control. Analysis of cricket batting shot execution is useful for numerous applications including performance evaluation, talent identification, injury prevention, and surveillance. Selection of the correct data capturing systems and methodologies are vital to the success of any sports activity recognition application. The data capturing system should not inconvenience or compromise the performance of the player, and the sensing elements should not disrupt their natural playing techniques. Camera-based, 3D position tracking methods are widely used in biomechanics applications, although the complexity of their use and lengthy post-processing generally limit their use to research studies. Body-worn inertial measuring units (IMUs), typically consisting of accelerometers, gyroscopes, and magnetometers, have been introduced to sports applications to capture movement data of athletes [1]. The benefits of using IMUs for batting movement data capture includes; the low cost of equipment; the experiments are not limited to laboratory-based setup; the dimensions of IMUs do not restrict body movements of the batsman; and direct measurement of linear acceleration or angular velocity, which eliminates the noise associated with differentiating data collected in a camera-based system [2]. While having the above-mentioned benefits, the recent innovations in Micro-electromechanical system (MEMS) technology also reduced the power consumption of the sensors [3], which makes those more suitable to capture batting movement data over extended periods of time.

Experimental designs used in previous research into cricket shot execution have not replicated a real-world batting session [4]. The limitations of the experimental design and the equipment used to capture movement data are likely to have affected batting style and the data captured. This paper reports a new experimental protocol, which captures both batting movements and routine activities in a simulated cricket batting session using body-worn IMUs that offer minimal obstruction to the wearer. The paper also discusses the potential of the redesigned batting data capturing method to forecast shot execution of the batsmen by applying supervised machine learning models on the captured batting movement data set.

## 2. Materials and Methods

Ten right-handed cricket batsmen were recruited for the experiment (age =  $21.5 \pm 1.5$ ). All batsmen were experienced cricket players, playing at least once a week. Nine players were of club standard, and eight were playing in professional county second-tier teams. The experiment was performed indoors at the National Cricket Centre and batsmen were wearing their preferred protective equipment which variously included; helmet, arm guard, chest protector, groin protector, gloves, thigh protector, and pads. A right-arm fast bowler was used to deliver the balls, and a professional cricket coach was used to assist the batting session. Each experiment session lasted for three overs (one over = six deliveries).

### 2.1. Equipment

Fifteen Shimmer3 IMUs manufactured by Shimmer Sensing (Dublin, Ireland) were attached at key body segments of a batsman using a minimally invasive Velcro strap. Each IMU contained three tri-axial sensors; an accelerometer, a magnetometer, and a gyroscope, which were sampled at a rate of 512 Hz and recorded onto an embedded memory card. Two synchronised high definition GoPro version 4 cameras were used to record the movements of batsman and bowler at 240 fps. The video recording system was synchronised with the IMUs during data annotation.

### 2.2. Experiment Protocol

The purpose of the experimental protocol to achieve a close representation of a real-world cricket batting scenario successfully. Therefore, it was designed not only to capture the batting shot execution but also to capture the routine movements of the batsman in between deliveries that form part of an innings. The routine movements can be divided into two broad categories as listed in Table 1. The experiment was designed to capture data for three consecutive overs. The batsman was given the freedom to select shots and, based on the shot executed, the coach decided the runs each batsman should complete by running between the wickets. The coach also instructed the batsman to perform the activities listed in Table 1 in between the deliveries as well as overs. The bowler was encouraged to bowl unexpected deliveries to the batsman once every over. That was either a “yorker”, which was pitched on the batting crease or a “bouncer”, which was pitched and raised to the chest level of the batsman or higher. The objective was to record responses of a batsman for an unexpected delivery.

### 2.3. Post-Processing of Data and Supervised Machine Learning

Captured raw IMU data collected from 15 sensors were synchronised with the video recording using ELAN multimedia annotation software [5]. Once synchronised, the raw data were labelled into two classes using the same software platform:

- Batting shot execution (Positive Class):  
*Start:* Release of the ball from the hand of the bowler;  
*Termination:* The end of the shot execution, which can be either bat-ball impact, ball-body impact or the ball passing the plane representing the batsman
- Non-Batting execution activities (Negative Class): All the other activities.

The labelled data set (180 batting cycles of 10 players) was divided into; a training and validation data set (first 13 batting cycles for each player) and test data set (last five batting cycles for each player).

**Table 1.** Routine activities of a batsman during a cricket batting inning.

Category A <sup>1</sup>	Category B
Drinking water during breaks	Taking guard
Talking to the batter at the other end	Adjusting gloves or PPE
Tying the laces of the shoe	Fix uneven parts of the pitch
	Look at the field

<sup>1</sup> Time consumption per activity is higher for the activities listed in category A.

The methodology used in this study is based on that used by Putra et al. and Khan et al. [6,7]. Two data segmentation techniques: Fixed Overlapping Sliding Window (FOSW) method and Fixed Non-overlapping Sliding Window (FNSW) method were used to construct the dataset for machine learning model training and validation [6]. The 180 batting cycles were captured for 10 batsmen, and batting shot execution time per batting cycle accounted for a mean of  $685.28 \pm 80.63$  ms with a minimum being 477 ms. Therefore, the raw training data set was segmented using two sliding windows (SW); 0.125 s and 0.25 s. In the FOSW method, 75% was used as the overlapping percentage of the sliding windows. A feature vector was formed to represent the data collected from 15 IMUs (three sensors with nine channels per IMU) for each SW. Mean, maximum, minimum, signal energy, signal entropy, and the total rate of change were calculated for each sensor axis, with root mean square for each sensor to form the feature vector [7]. The feature vector was limited to 20 highly relevant features using Neighbourhood Component Analysis (NCA) [8] and then used for classification.

Three different classification algorithms; Support Vector Machine (SVM), *k*-nearest neighbours (*k*-NN), and Discriminant Analysis algorithms, available in the statistics and machine learning toolbox of MATLAB R2018a, were used for classification of batting shots. All models were trained and validated using *k*-fold cross-validation (*k* = 5) by splitting the dataset into *k* holdouts and using each fold as a validation set by conducting holdout validation *k* times. Finally, in the testing stage, every model was used on the unseen dataset containing 50 batting cycles, that was separated in the beginning, to assess the performance of the model. The test data set was also segmented similarly to the corresponding training dataset, which used to train and validate the model.

#### 2.4. Forecasting Batting Shot Execution

A reliable classifier should be capable of minimising two types of errors; false positives (FP) and false negatives (FN). The assessment method that is used to measure the performance of a supervised machine learning model should reflect the model’s ability to minimise those errors. Most of the widely used model performance assessment criteria in activity monitoring are defined based on a confusion matrix. The confusion matrix for the experiment is shown in Figure 1, in which cricket batting shot execution is considered as the positive class. When accuracy is used as a performance measure, (Equation (1)) it considers both True Positives (TP) and True Negatives (TN) equally important while minimising FNs and FPs. F-score is another performance measure widely used for the applications with activity class imbalance, in which one activity class consumes significantly more time to the other [9], and F-score captures the success of minimising FN, FP, and capturing TPs (Equation (2)). Both Accuracy and F-score were computed for each classifier; (i) at the training and validation stage and (ii) at the testing stage.

$$\text{Accuracy} = (TP + TN)/(TP + FP + TN + FN), \tag{1}$$

$$\text{F-Score} = 2TP/(2TP + FP + FN), \tag{2}$$

		Predicted	
		Shot Execution	Non-Shot Execution
Actual	Shot Execution	TP (True Positive)	FN (False Positive)
	Non-Shot Execution	FP (False Positive)	TN (True Negative)

Figure 1. Confusion matrix for risk classification.

### 3. Results

Twenty-eight classifiers were constructed for the experiment as a result of the multiple segmentation techniques and algorithms utilised in the training and validation stage, and Table 2 lists the computed accuracy and F-score values for every model. All models recorded Accuracy above 95.15% with a maximum of 99.36%. While Medium Gaussian SVM recorded the highest (highlighted in grey), Quadratic Discriminant Analysis recorded the lowest Accuracy values for all segmentation scenarios. According to Figure 2a, all classifiers except Fine *k*-NN recorded an increase in ΔAcc (Equation (3)) for FOSW while FNSW segmentation recorded negative ΔAcc.

$$\Delta Acc = (Accuracy)_{Test} - (Accuracy)_{Train}, \tag{3}$$

$$\Delta FS = (F-Score)_{Test} - (F-Score)_{Train}, \tag{4}$$

Table 2. Accuracy and F-score performance of the classifiers when applied on the unseen test data set.

Algorithm		Accuracy				F-Score			
		SW: 0.125 s		SW: 0.25 s		SW: 0.125 s		SW: 0.25 s	
		FNSW	FOSW	FNSW	FOSW	FNSW	FOSW	FNSW	FOSW
<i>Support Vector Machines</i>	Quadratic	98.97	99.07	99.25	99.20	95.94	96.34	97.45	97.29
	Cubic	99.02	99.00	99.22	99.18	96.13	96.06	97.33	97.22
	Med: Gau:	99.27	99.26	99.36	99.21	97.13	97.12	97.83	97.34
<i>k-Nearest Neighbours</i>	Fine	99.02	99.08	99.07	98.97	96.19	96.40	96.86	96.52
	Cosine	99.24	99.23	99.29	99.10	97.04	96.99	97.56	96.95
	Weighted	99.17	99.16	99.27	99.07	96.77	96.71	94.54	97.50
<i>Disc: Analysis</i>	Quadratic	95.16	96.67	96.23	95.72	83.82	87.85	88.42	87.23

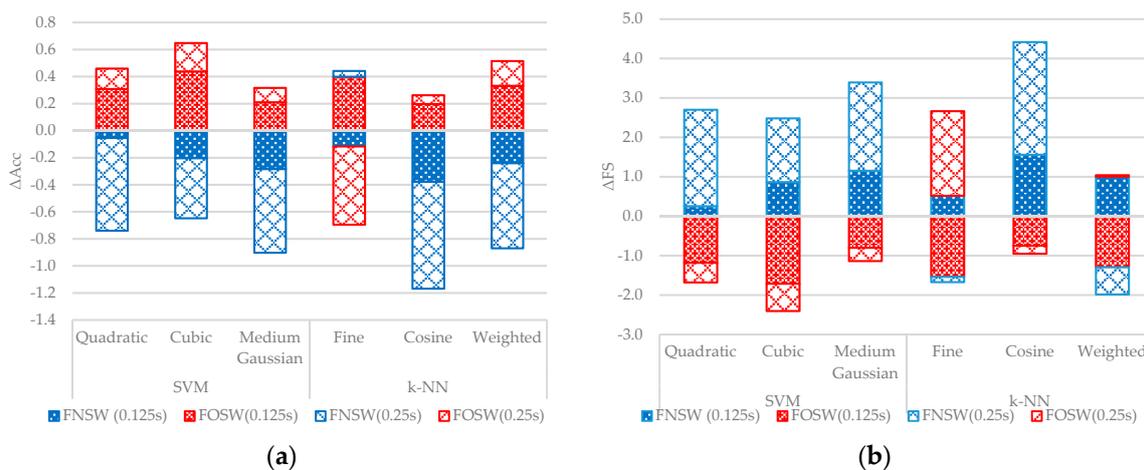


Figure 2. (a) ΔAcc and (b) ΔFS calculated for each classifier for different segmentation scenarios.

When misclassification rates were analysed, the FN percentage was found to be higher than the FP percentage for each model (Figure 3). The combination of classifiers and segmentation techniques also resulted in different FN and FP percentages. When F-score is calculated, the highest of 97.8% was recorded by a Medium Gaussian SVM classifier and Quadratic Discriminant Analysis classifier

recorded the lowest F-score values (<90% across all segmentation scenarios). However, in contrast to accuracy, all SVM models and Cosine  $k$ -NN recorded an increase in  $\Delta FS$  (Equation (4)) at the testing stage for FOSW and while recording a negative  $\Delta FS$  for FNSW segmentation.

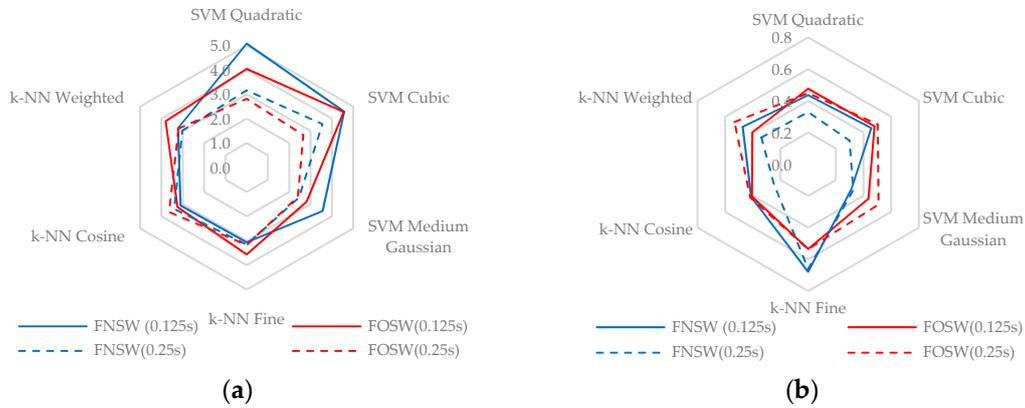


Figure 3. Percentage misclassifications: (a) FN and (b) FP for each classifier during the testing phase.

#### 4. Discussion

The proposed experimental protocol was effectively followed by the batsmen and bowlers who participated in the experiment. Each batting session lasted for approximately 12–15 min. Fifteen IMUs, each sampled at 512 Hz generated a high volume of movement data representing 15 body segments of the batsman. Supervised machine learning was used for the task of predicting a batting shot execution from the collected data set, that cannot be identified using the naked eye. A SW technique was used to construct the training data set and both SWs selected for the segmentation were smaller than the average duration of batting shot execution. The purpose of applying NCA was to extract 20 highly relevant features for every SW to construct the training data set by removing irrelevant and redundant features. A number of validation techniques are available for use in supervised machine learning. Holdout validation is used when a dataset is significantly large, which uses an arbitrary dividing point to separate the dataset into training and validation segments. However,  $k$ -fold cross-validation ( $k = 5$ ) was selected as the validation technique for this study to prevent overfitting problem due to a relatively small number of test subjects used in the experiment [10].

The data set was treated as a binary machine learning problem, and batting shot execution was considered as the positive class. Once trained, each machine learning model was used on the unseen data set and performance assessed based on predicted activity classes. The performance of models was assessed using two widely used measures; Accuracy and F-score. Each measure was calculated twice for every model for the training stage and testing stage. The purpose of computing accuracy and F-score in two stages was to measure the change in performance  $\Delta Acc$  and  $\Delta FS$ , which provides a fair assessment on model performance deterioration when it is exposed to an unseen data set.

Medium Gaussian SVM classifier showed the best performance with the highest values for Accuracy and F-score (99.36% and 97.83%), which suggest that the batting shot execution can be forecasted with high prediction accuracy and reliability. Figure 3 shows that the percentage of FNs are higher than FPs for all classifiers and different classifiers recorded different FP and FN percentages based on the segmentation method used for training and validation. Due to the imbalanced nature of the batting data set, significantly higher data points were available for the negative class, which resulted in the reduction of FP percentage. However, for the prediction of cricket batting shot execution, the priorities were detected shot execution (TP), missed shots (FN), and false alarms (FP), and all three outcomes were considered for F-score calculation except TNs [6]. Therefore, F-score was deemed to provide a fair assessment of the models used for batting shot execution. An increased length of SW allows for more complex activity detection at a cost of real-time performance [11]. This is evident in the Medium Gaussian classifiers trained with a data set

segmented using FNSW with a larger sliding window, which produced the highest F-score while also increasing the performance on the test data set.

The suggested experimental protocol to collect batting movement data proven to be successful in forecasting batting shot execution in cricket by using IMUs and supervised machine learning. However, as shown in the results, the success of forecasting batting shot execution depends on multiple factors such as the type of classifier used and the type of data segmentation technique used to construct the training data set. Therefore, special attention should be given when selecting those parameters for the experiment.

## 5. Conclusions

Based on a dataset of cricket batter movements captured throughout a simulated inning, it has been shown to be possible to forecast the batting shot execution using supervised machine learning with high prediction accuracy (Accuracy = 99.36% and F-score = 97.83%). IMUs were shown to be capable of capturing movement data in cricket batting, with minimal preparation time and post-processing while causing minimal disruptions to the movement patterns of the athlete. Based on the data collected, supervised machine learning has been shown to be effective in forecasting batting shot execution in an unlabelled data set. The performance of a machine learning model was analysed using multiple methods and F-score shown to be the most appropriate based on its ability to detect missed shots and false alarms. The prediction performance was affected by many factors such as the type of the classifier, the segmentation technique, size of the sliding window, extracted features, and the volume of data available for training and validation. Therefore, it can be concluded that the cricket batting experimental protocol described here has been shown to be effective in forecasting when a batting shot execution takes place in a cricket batting cycle and is recommended for use in future studies in cricket batting such as performance evaluation, talent identification, injury surveillance, and prevention where real-world representation of cricket batting movement data are important.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Chardonens, J.; Favre, J.; Cuendet, F.; Gremion, G.; Aminian, K. A system to measure the kinematics during the entire ski jump sequence using inertial sensors. *J. Biomech.* **2013**, *46*, 56–62, doi:10.1016/j.jbiomech.2012.10.005.
2. Kavanagh, J.J.; Menz, H.B. Accelerometry: A technique for quantifying movement patterns during walking. *Gait Posture* **2008**, *28*, 1–15, doi:10.1016/j.gaitpost.2007.10.010.
3. Yang, C.C.; Hsu, Y.L. A review of accelerometry-based wearable motion detectors for physical activity monitoring. *Sensors* **2010**, *10*, 7772–7788, doi:10.3390/s100807772.
4. Peplow, C.; King, M.; Harland, A. The effects of different delivery methods on the movement kinematics of elite cricket batsmen in repeated front foot drives. *Procedia Eng.* **2014**, *72*, 220–225, doi:10.1016/j.proeng.2014.06.039.
5. Lausberg, H.; Sloetjes, H. Coding gestural behavior with the NEUROGES-ELAN system. *Behav. Res. Methods* **2009**, *41*, 841–849, doi:10.3758/BRM.41.3.841.
6. Putra, I.P.E.S.; Brusey, J.; Gaura, E.; Vesilo, R. An event-triggered machine learning approach for accelerometer-based fall detection. *Sensors* **2018**, *18*, 20, doi:10.3390/s18010020.
7. Khan, A.; Nicholson, J.; Plötz, T. Activity Recognition for Quality Assessment of Batting Shots in Cricket using a Hierarchical Representation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **2017**, *1*, 62, doi:10.1145/3130927.
8. Yang, W.; Wang, K.; Zuo, W. Neighborhood component feature selection for high-dimensional data. *J. Comput.* **2012**, *7*, 162–168, doi:10.4304/jcp.7.1.161-168.
9. Kubat, M.; Matwin, S. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. *ICML* **1997**, *97*, 179–186.

10. Borovicka, T.; Jirina, M., Jr.; Kordik, P.; Jirina, M. Selecting Representative Data Sets. In *Advances in Data Mining Knowledge Discovery and Applications*; Karahoca, A., Ed.; IntechOpen Limited: London, UK, 2012; pp. 43–70, doi:10.5772/50787.
11. Lee, J.; Kim, J. Energy-Efficient Real-Time Human Activity Recognition on Smart Mobile Devices. *Mob. Inf. Syst.* **2016**, *2016*, 2316757, doi:10.1155/2016/2316757.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).