Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 7, No. 2, May, Pp. 179-186

A wearable device for enhancing basketball shooting correctness with MPU6050 sensors and support vector machine classification

Baginda Achmad Fadillah[1], Aji Gautama Putrada*[2], Maman Abdurohman[3]
School of Computing, Telkom University, Indonesia[1,2,3]

Article Info

Keywords:
Accelerometer, Gyroscope, Basketball, Shooting, Support Vector Machine, Internet of Things, Wearable Device, MPU6050 Sensor

Article history:
Received: April 12, 2022
Accepted: May 14, 2022
Published: May 31, 2022

Cite:
B. A. Fadillah, A. G. Putrada, and M. Abdurohman, "A Wearable Device for Enhancing Basketball Shooting Correctness with MPU6050 Sensors and Support Vector Machine Classification", *KINETIK*, vol. 7, no. 2, May. 2022.
https://doi.org/10.22219/kinetik.v7i2.1435

*Corresponding author.
Aji Gautama Putrada
E-mail address:
ajigps@telkomuniversity.ac.id

Abstract

One of the impacts of Covid-19 is the delay of basketball sports competitions, which influences the athlete's fitness and the athlete's ability to play, especially for shooting techniques. Existing research in wearable devices for basketball shooting correctness classification exists. However, there is still an opportunity to increase the classification performance. This research proposes designing and building a smartwatch prototype to classify the basketball shooting technique as correct or incorrect with enhanced sensors and classification methods. The system is based on an Internet of things architecture and uses an MPU6050 sensor to take gyroscope data in the form of X, Y, and Z movements and accelerometer data to accelerate hand movements. Then the data is sent to the Internet using NodeMCU microcontrollers. Feature extraction generates 18 new features from 3 axes on each sensor data before classification. Then, the correct or incorrect classification of the shooting technique uses the Support-Vector-Machine (SVM) method. The research compares two SVM kernels, linear and 3rd-degree polynomial kernels. The results of using the max, average, and variance features in the SVM classification with the polynomial kernel produce the highest accuracy of 94.4% compared to the linear kernel. The contribution of this paper is an IoT-based basketball shooting correctness classification system with superior accuracy compared to existing research.

1. Introduction

One of the impacts of Covid-19 is the delay of basketball sports competitions. It influences the athlete's fitness and ability to play, especially for shooting techniques [1]. One of the technologies of the industrial revolution 4.0, namely the Internet of Things (IoT), is expected to be a solution. There have been several previous studies for research on basketball using wearable devices. In a study by Kuhlman *et al.* [2], wearable devices analyze and classify shooting forms in basketball by collecting shooting data through the accelerometer sensor and producing nine main features plus one feature as the target of the classification. The classification method used is the support vector machine (SVM) and processed on the MATLAB platform to produce an accuracy of 86.3%. In another study by Putra *et al.* [3], the use of accelerometer and gyroscope sensors with the help of features extracted from the Hjorth parameter in the decision tree classification method can produce an accuracy of 91%. The use of inertial measurement unit (IMU) sensors such as MPU6050 on gesture detection are ample. An example research is Ameliasari *et al.* [4] that used IMU sensors for smart lighting control based on hand gesture.

Furthermore, Antunes *et al.* used IoT technology to analyze and make recommendations based on the device used to monitor the performance of shooting techniques [5]. Shankar *et al.* [6] developed a wearable device that can monitor the performance and analyze the shooting form of basketball athletes remotely when the athlete is doing free-throw shooting. Ali *et al.* [7] said that the application of IoT in basketball is by placing wearable devices on basketball athletes during matches and transmitting all information obtained from sensors embedded into a database to be stored and processed so that later the information can be forwarded.

Apart from basketball, there is much research on wearables in various other sports. Woldu *et al.* [8] developed a boxing glove with accelerometer and gyroscope sensors to determine the type of fist the user releases (jab, cross, left & right uppercut, and left & right hook). Iancu, B. *et al.* [9] used a wearable device with accelerometer, gyroscope, and magnetometer sensors to analyze goalkeepers' ability during practice. Finally, Krüger *et al.* [10] used a smartwatch to recognize and classify different golf club swings (putters, irons, and drivers) using data from the accelerometer sensor.

This study proposes to design and build a smartwatch prototype that aims to classify the correct or incorrect technique of shooting basketball with the help of gyroscope and accelerometer sensors that exist in an MPU6050 module. The gyroscope sensor records angular velocity data when the smartwatch rotates, while the accelerometer sensor measures and records the acceleration of movement of body parts [4]. In this case, the MPU6050 sensor resides on both players' wrists.

The recording results of these sensors will then be stored on the Internet using the help of the NodeMCU microcontroller so that the data can then be processed. Data processing starts from feature extraction, which produces max, average, and variance features. Then, classification using the SVM method determines whether the shooting technique is correct or incorrect.

2. Research Method

The methodology used in writing this research is as seen in Figure 1. The first process carried out by the author is a Literature Study. The second and third processes are the prototype design and the classification model layout. The fourth, fifth, and sixth processes are prototype design implementation, data collection, and classification model implementations. The seventh and eighth processes are the results analysis, evaluation, and conclusions.

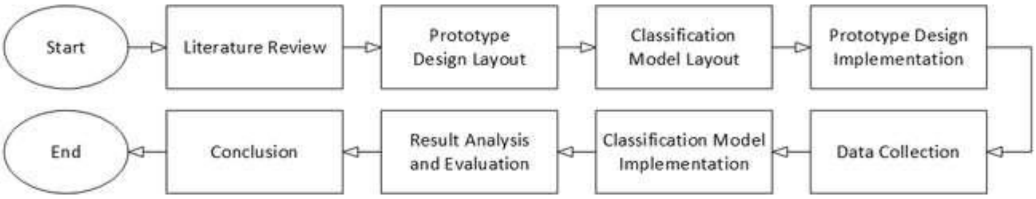


Figure 1. Research Methodology

2.1 IoT System and Data Collection

Some hardware builds a motion detection system for throwing basketballs. NodeMCU is an electronic board based on the ESP8266 chip, which has a similar function to a microcontroller and has an Internet connection [11]. The selection of the NodeMCU is because this microcontroller has advantages compared to Arduino microcontrollers, namely the ability to connect to the Internet via a Wi-Fi module [12].

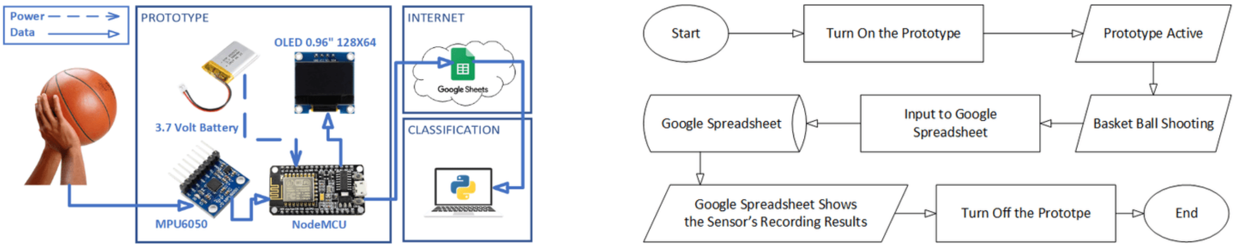
Furthermore, MPU6050 is a micro-electro-mechanical system (MEMS) that can record a movement or change in position and acceleration. This sensor combines two sensors: an accelerometer and a gyroscope. This sensor has 3-axis, namely Roll (x), Pitch (y), and Yaw (z) [13]. The utilization of the MPU6050 in this study is because the two sensors can record the wrist movement of a basketball player. The system also utilizes batteries and a liquid crystal display (LCD). In this study, the battery used is a primary battery, namely lithium. This type of battery is selected because it is cheap, light, and often used for portable electronic devices [14].

Moreover, in this research, the screen used is a 0.96" 128x64 OLED screen. The selection of 0.96 Inch OLED is suitable for smartwatch prototypes because of its small size [15]. Other supporting hardware is a breadboard, jumper cables, and a MicroUSB cable. Mini Breadboard 400 Pins is small, so it is comfortable to wear on the wrist when shooting. Jumper cables in use are Male-to-Male type with 10pcs and Male-to-Female 4pcs. Then a 30cm MicroUSB Cable programs the NodeMCU.

Some software is involved in system building. Arduino Integrated Development Environment (IDE) is an independent platform for Arduino hardware. This research uses Arduino IDE because it is open-source, can run on various operating systems, and program various types of boards outside the Arduino board, including NodeMCU [16]. Google Sheets is one of the tools provided by Google for free, can be accessed anywhere and anytime via mobile devices, tablets, or computers, and can be used both inside and outside the network [17]. In this research, Google Spreadsheet is used to store sensor recording data. This research uses Google Sheets because, compared to other platforms such as ThingSpeak and Blynk, this platform is lightweight and free of charge in getting sensor data in real-time.

The implementation of machine learning uses Python. Python is a popular modular programming language that is open-source, free, powerful, and lightweight [18]. The research adopts this programming language because documentations are ample, while libraries can process data generated from sensors. The research adopts Google Colab because of its integration with Google Drive, while general libraries are directly available and manual installation is unrequired. The libraries used in this research are Pandas (reading coma separated values (CSV) files), NumPy (computing), Os and Glob (changing directories), Seaborn and Matplotlib (graphics), and Skicit-Learn (accuracy results).

Figure 2 shows the data collecting process. The first thing to do is turn on the prototype with a 3.7 Volt battery so that the NodeMCU can activate the 0.96" 128x64 OLED to display the results of the MPU6050 sensor recording. Next, the user will carry out the shooting process. Then the latest sensor recording results will be forwarded to Google Sheets using an Internet connection. Afterward, Google Sheets display the results of the MPU6050 sensor recording in the user's browser. Finally, users can save the Google Spreadsheet results in CSV format and remove the 3.7 Volt battery to turn off the prototype. The saved CSV file becomes useful in the classification process.



(a) (b)

Figure 2. The System Block Diagram (a) and Data Collection Flowchart (b)

Data collection scenarios consist of shooting experiments with professional players and novice players. Each type of player conducted 30 shootings, where each shoot produced one set of shooting data. This number of data is a novelty, whereas in [3], the collected data were only ten samples. Two shooters volunteered for the research. Each shooter represented a type of player. For the correct shooting label, the volunteer is an athlete from the Indonesian Basketball Association, Bandung City Chapter, who has been the National Sports Week basketball athlete from Bandung City and has won various basketball competitions at the provincial and national levels. Meanwhile, the volunteer for the incorrect shooting label is a beginner amateur player.

2.2 Classification Model

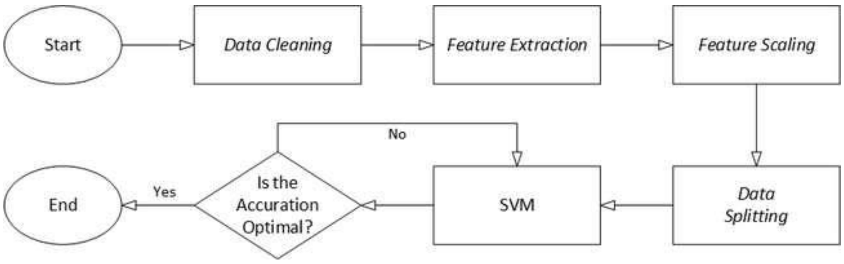


Figure 3. Classification Flowchart

In Figure 3, the classification process starts with data cleaning, removing noise and missing/null values in the dataset. Next, feature extraction extracts six features obtained from three axes on the two sensors. After feature extraction, the feature scaling process equalizes the values on a scale of 0 – 1 [19]. After the values are on one scale, the data splitting process divides the data into training and testing data for classification needs. After the data division, the next step is to use the data in the SVM classification process. If the accuracy of the SVM is not optimal, the parameters are changed, increasing the SVM model accuracy.

In the feature extraction process, three new values emerge from three axes on two sensors, namely max, average, and variance, resulting in 18 new features. The mentioned features are statistical features that can be derived from sequential data [20]. The following Equation 1 mathematical equation is to find the variance value.

$$\sigma^2 = \frac{\sum_{i=1}^{n} (v_i{}')^2}{n-1} \tag{1}$$

Where σ^2 is the variance or squared deviation, n is the dataset length, and $v_i{}'$ is the deviation between a data item and the mean of the dataset.

The feature scaling process is necessary because the data from the accelerometer sensor has different units and value ranges from the data from the gyroscope sensor. It results from the accelerometer sensor outputs a value in units of m/s^2 $(meter\ per\ second^2)$, while the gyroscope sensor outputs a value in units of rps (rotation per second).

Based on this, MinMax feature scaling or normalization of values in the range 0-1 uses the following mathematical Equation 2 [21].

$$x_{new} = \frac{x - \min(X)}{\max(X) - \min(X)} \tag{2}$$

Where x_{new} is the transformed data item x of feature X, $\max(X)$ is the maximum value in X, and $\min(X)$ is the minimum value in X.

The normalized data shape has 19 columns in this process: 18 columns are features, while one column is the label. Then, the data is split into training and testing data randomly. The split is to apply the data to the SVM classification training.

SVM is a supervised learning technique in machine learning [22]. SVM works to minimize structural risk by finding the best hyperplane to separate data from existing classes. The best hyperplane has the largest margin with the smallest error. The margin is the distance between the first-class hyperplane and the second-class hyperplane. The hyperplane in the existing class consists of the data points closest to the hyperplane, commonly called support vectors. The mathematical Equation 3 of the hyperplane is as follows.

$$y(x) = w^T x + b \tag{3}$$

Where y is the class label, w is the vector of weight parameters, x is the data in the dataset, and b is the bias with a scalar value.

The hyperplane becomes a form that divides the data into two classes, namely positive and negative classes. A mathematical Equation 4 explains the process for dividing these data as follows (assuming that i is the number of class labels).

$$y_i(w^T x + b) \geq 1, i = 1,2,\dots,n \tag{4}$$

A mathematical equation defines the distance between two hyperplanes (margins) as follows Equation 5.

$$\frac{|w^T x_i + b|}{||w||} = \frac{1}{||w||} \tag{5}$$

The result of the total distance between the two hyperplanes is $\frac{2}{||w||}$. To get the maximum margin, the value of $||w||$ must be small. The formula for finding the smallest value of $||w||$ is as follows Equation 6.

$$min \frac{1}{2} ||w||^2 \tag{6}$$

2.3 Test Metrics and Scenario

The shape of an SVM hyperplane depends on its kernel, where two types of SVM kernels are linear and polynomial. The type of kernel influences the SVM model performance. The most suitable kernel depends on how each label in the dataset separates. This research conducsts a comparison test to decide which kernel is more suitable in the case of this research. Confusion matrices, accuracy, precision, recall, and f1-score are metrics to compare the two kernels. A confusion matrix shows the prediction characteristics of a classification model [23].

3. Results and Discussion

3.1 Results

The research successfully built a smartwatch prototype, which mounts on the wrist, as seen in Figure 4. As prototype functionalities run by requirements, the illustration also depicts the data collection process.

The results specifications of the data collecting process are as seen in Table 1. Each shooting produces one CSV file. There are 25 to 35 lines of MPU6050 sensor data for professional players in one CSV file and 20 to 25 lines of MPU6050 sensor data for beginner players. The time taking one data shooting professional player requires longer data than beginner player, which is 3 to 5 seconds compared to 2 to 3 seconds.

(a) (b) (c)

Figure 4. Smartwatch Prototype (a) and (b); Collecting Data Process (c)

Table 1. Collecting Data Results

	Professional Player	Beginner Player
Shooting Attempts	30 times shooting	30 times shooting
Rows per Dataset	25-35 rows	20-25 rows
Features per Dataset	6	6
Shooting Estimation Time	3-5 seconds	2-3 seconds

The feature extraction process extracts three statistic variables from six main features of each CSV file. From the process, each CSV file transforms into a data item consisting of 18 new features, meaning there are 60 data items with 30 items for each label. The 18 new features are as follows:

- Maximum values of AccX, AccY, AccZ, GyrX, GyrY, and GyrZ
- Average values of AccX, AccY, AccZ, GyrX, GyrY, and GyrZ
- Variance values of AccX, AccY, AccZ, GyrX, GyrY, and GyrZ

The random forest feature importance function from Python's Scikit-Learn library calculates the importance of each feature [24]. Figure 5 shows the score of each feature. The average value of the accelerometer Z-axis (AccZ_ave) is the most important feature, while the maximum value of the gyroscope Z-axis (GyrZ_max) is the least important.

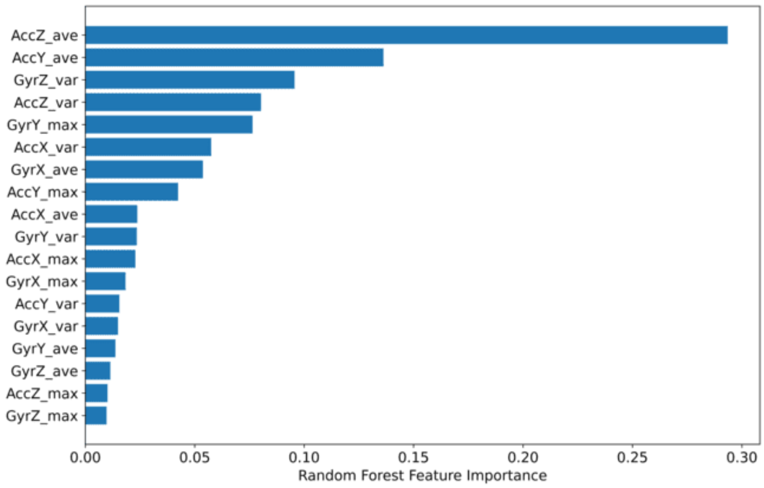


Figure 5. Feature Scores with the use of Random Forest Feature Importance

Figure 6 displays the transformation results in the form of kernel density estimation (KDE) plots. Before scaling, each feature has a varying range of values. The X-axis from the before scaling plot ranges from 0 to 80, with each feature having a different range. After min-max scaling, the range of each feature becomes uniform, *i.e.*, 0 to 1.

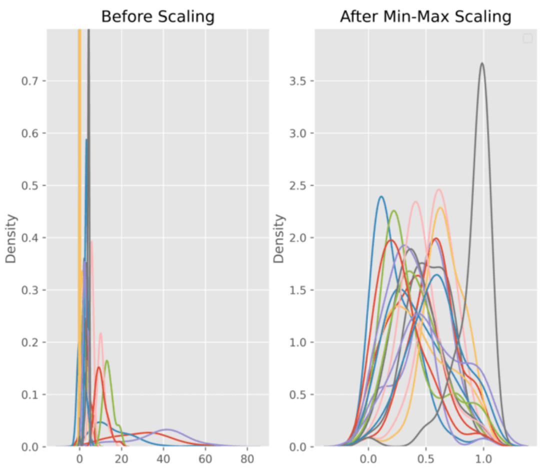


Figure 6. Original vs Scaled Dataset

Table 2 shows the confusion matrix comparison results—the greater the value of TP and TN, the better the model's performance. Conversely, the greater the value of FP and FN, the worse the performance. According to the comparison, the SVM model with the polynomial kernel has a higher TP value than the linear kernel, ten versus 7. The results indicate that the polynomial kernel SVM model performs better than the linear kernel.

Table 2. Linear Kernel Confusion Matrix (a) and Polynomial Kernel Confusion Matrix (b)

(a)
Linear Kernel

	Shooting	Actual	
		Correct	Incorrect
Predicted	Correct	TP = 7	FP = 0
	Incorrect	FN = 4	TN = 7

(b)
Polynomial Kernel

	Shooting	Actual	
		Correct	Incorrect
Predicted	Correct	TP = 10	FN = 0
	Incorrect	FN = 1	TN = 7

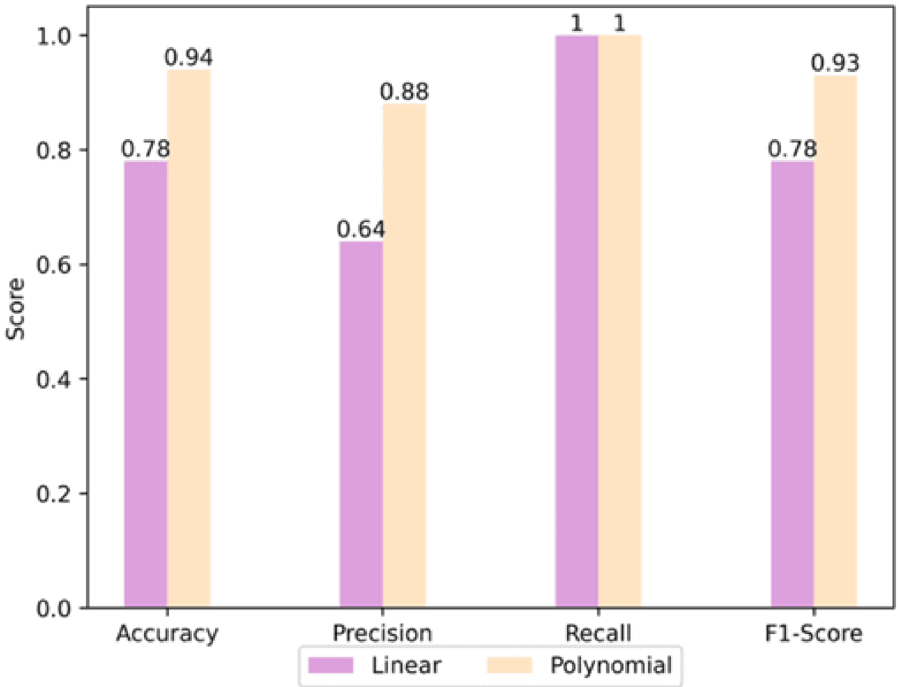


Figure 7. Comparative Analysis Between SVM Kernels

Figure 7 shows the results of the performance of the classification model in predicting 18 shooting experimental data items. The accuracy, precision, and f1-score of the SVM model with a polynomial kernel are higher than the model with a linear kernel. The difference in precision is the highest, 0.88 versus 0.64, where the difference in f1-score is the lowest at 0.93 versus 0.78. The two models have the same recall value, whereas the recall value of the two models is the highest parameter value compared to others.

Linear kernels work best on data with linearly separable labels. If one line is not enough to separate the data, then the data is called linearly inseparable [25]. Thus, other kernels, such as the 3rd-degree polynomial, are more suitable for classifying the data. Figure 8 shows how a scatterplot matrix displays labels' separation. Pairing plots with Python's Seaborn Library generate these graphs. Principal component analysis (PCA) derived from the Python Skicit-Learn Library reduced the number of features from 18 to two to make graphics more compact.

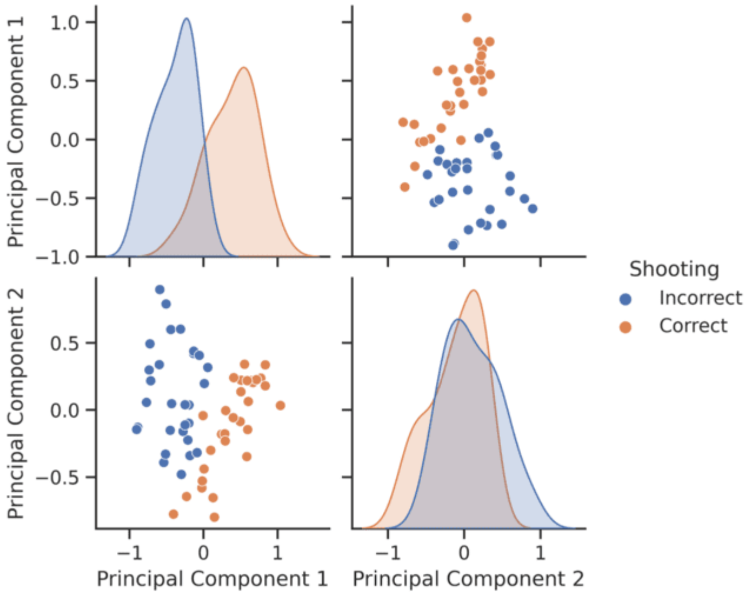


Figure 8. Scatterplot Matrix

3.2 Disussion

This research uses SVM with a 3rd-degree polynomial kernel to classify shooting correctness in basketball games. Compared to [2], the results obtained in this study are more accurate. Both studies use SVM classification, but the previous studie does not use a gyroscope sensor and uses fewer datasets. Whereas, in our research, we proved that the the gyroscope z-axis variance is the third most important feature in the training process. The contribution of this research is to prove that the addition of a gyroscope sensor and an increase in the number of datasets can significantly boost accuracy.

Moreover, this research is also more accurate than research [3]. The earlier research uses Hjorth parameters and decision tree classification. Then the previous research conducted classification on data items instead of a whole shooting process (one shooting dataset), so it could not correctly classify a basketball player's shooting. The contribution of this research is increasing the accuracy of throwing classification using the SVM method and providing the ability to classify a player's basketball shooting. Table 3 emphasizes the contribution of this study compared to previous studies.

Table 3. Results Comparison with Previous Works on Basketball Shooting

Reference Paper	Sensor(s)		Feature Extraction		Classification Method		Accuracy (%)
	Accelerometer	Gyroscope	Hjorth Parameter	Statistical	Decision Tree	SVM	
[2]	√	-	-	√	-	√	86,3
[3]	√	√	√	-	√	-	91
Proposed Method	√	√	-	√	-	√	94

Furthermore, the direction of future works is to commercialize prototypes for better identification of basketball throws and identification of other techniques in basketball, such as dribbling and passing. So, in that way, improving basketball skills during the pandemics while social distancing can still occur.

4. Conclusion

This study succeeded in building an IoT system using accelerometer and gyroscope sensors to record the acceleration and rotation of movement and a classification model that can classify the correctness of basketball shooting techniques. The test results show that using the max, average, and variances features in the SVM classification method with a 3rd-degree polynomial kernel produces better accuracy than the linear kernel, 94%. The contribution of this research is a system for detecting the correctness of throwing basketballs that is more accurate than existing studies.

Acknowledgement

Authors would like to thank Indonesian Baksetball Association, Bandung City Chapter because their player have volunteered in the data collection process of the research. Without them, important findings could not be reached.

References

[1] P. D. Y. Patil and M. Wasnik, "Important Skills in Basketball and Different Methods," *Aayushi Int Interdiscip Res Journa*, pp. 140–2, 2020.

[2] N. Kuhlman and C.-H. Min, "Analysis and classification of basketball shooting form using wearable sensor systems," in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, 2021, pp. 1478–1482. https://doi.org/10.1109/CCWC51732.2021.9375925

[3] A. P. Putra, Analisis Pergerakan Tangan Saat Free Throw Pada Olahraga Bola Basket Menggunakan Sensor IMU dengan Metode Decision Tree. Universitas Telkom, S1 Informatika, 2021. Accessed: Apr. 11, 2022.

[4] M. Ameliasari, A. G. Putrada, and R. R. Pahlevi, "An Evaluation of SVM in Hand Gesture Detection Using IMU-Based Smartwatches for Smart Lighting Control," *JURNAL INFOTEL*, vol. 13, no. 2, Art. no. 2, May 2021. https://doi.org/10.20895/infotel.v13i2.656

[5] A. S. dos S. de A. Antunes, "Use of IoT technologies to improve shooting performance in basketball," PhD Thesis, 2018.

[6] S. Shankar, R. P. Suresh, V. Talasila, and V. Sridhar, "Performance measurement and analysis of shooting form of basketball players using a wearable IoT system," in *2018 IEEE 8th International Advance Computing Conference (IACC)*, 2018, pp. 26–32. https://doi.org/10.1109/CIMCA.2018.8739721

[7] M. Ali, "IOT Based Architecture for Basketball Supervision," *Lahore Garrison University Research Journal of Computer Science and Information Technology*, vol. 3, no. 4, pp. 30–38, 2019. https://doi.org/10.54692/lgurjcsit.2019.030490

[8] K. Sosopoulos and M. T. Woldu, "IoT smart athletics: Boxing glove sensors implementing machine learning for an integrated training solution." 2021.

[9] B. Iancu, "Goalkeeper analytics using a wearable embedded system," *Academy of Economic Studies. Economy Informatics*, vol. 18, no. 1, pp. 5–12, 2018.

[10] A. Krüger, "Recognizing and classifying a golf swing using accelerometer in a Smartwatch." 2017.

[11] I. Mahesa, A. G. Putrada, and M. Abdurohman, "Egg quality detection system using fuzzy logic method," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 207–216, 2019. https://doi.org/10.22219/kinetik.v4i3.839

[12] F. A. Rachman, A. G. Putrada, and M. Abdurohman, "Distributed Campus Bike Sharing System Based-on Internet of Things (IoT)," in *2018 6th International Conference on Information and Communication Technology (ICoICT)*, 2018, pp. 333–336. https://doi.org/10.1109/ICoICT.2018.8528778

[13] F. Z. A. Atoir, A. G. Putrada, and R. R. Pahlevi, "An evaluation of complementary filter method in increasing the performance of motion tracking gloves for virtual reality games," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 2021. https://doi.org/10.22219/kinetik.v6i2.1234

[14] M. H. Alfarisy, A. G. Putrada, and M. Abdurohman, "Energy Harvesting Pada Ban Mobil Menggunakan Piezoelektrik Transducer Untuk Wsn Suhu Ban," eProceedings of Engineering, vol. 8, no. 5, 2021.

[15] G. D. Wicaksana, M. Abdurohman, and A. G. Putrada, "Peningkatan quality of experience pada permainan online multiplayer berbasis Arduino dengan menggunakan MQTT server," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 1, pp. 36–43, 2020. https://doi.org/10.14710/jtsiskom.8.1.2020.36-43

[16] S. Amarta, A. G. Putrada, and N. A. Suwastika, "Asesmen Kebisingan Di Open Library Telkom University Menggunakan Sistem Monitoring Suara Berbasis Iot," eProceedings of Engineering, vol. 6, no. 1, 2019.

[17] Q. Aini, U. Rahardja, I. Handayani, M. Hardini, and A. Ali, "Utilization of google spreadsheets as activity information media at the official site alphabet incubator," in Proc. Int. Conf. Ind. Eng. Oper. Manag, 2019, no. 7, pp. 1330–1341.

[18] N. H. Anh, "Development of a Software Environment and User Interface for the Analysis of Accelerometer Data," 2020.

[19] S. Karuniawati, A. G. Putrada, and A. Rakhmatsyah, "Optimization of Grow Lights Control in IoT-Based Aeroponic Systems with Sensor Fusion and Random Forest Classification," in *2021 International Symposium on Electronics and Smart Devices (ISESD)*, 2021, pp. 1–6. https://doi.org/10.1109/ISESD53023.2021.9501863

[20] R. R. Pamungkas, A. G. Putrada, and M. Abdurohman, "Performance improvement of non invasive blood glucose measuring system with near infra red using artificial neural networks," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 315–324, 2019. https://doi.org/10.22219/kinetik.v4i4.844

[21] A. Pandey and A. Jain, "Comparative analysis of KNN algorithm using various normalization techniques," *International Journal of Computer Network and Information Security*, vol. 9, no. 11, p. 36, 2017. https://doi.org/10.5815/ijcnis.2017.11.04

[22] A. F. Aryuni, A. G. Putrada, and M. Abdurohman, "Klasifikasi Penumpang Naik Dan Turun Dengan Sensor Load Cell Menggunakan Ekstraksi Fitur Dan Metode Support Vector Machine," eProceedings of Engineering, vol. 8, no. 2, 2021.

[23] P. Nando, A. G. Putrada, and M. Abdurohman, "Increasing The Precision Of Noise Source Detection System using KNN Method," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 157–168, 2019. https://doi.org/10.22219/kinetik.v4i2.757

[24] E. S. Saputra, A. G. Putrada, and M. Abdurohman, "Selection of Vape Sensing Features in IoT-Based Gas Monitoring with Feature Importance Techniques," in *2019 Fourth International Conference on Informatics and Computing (ICIC)*, 2019, pp. 1–5. https://doi.org/10.1109/ICIC47613.2019.8985807

[25] A. Kesumawati, "Perbandingan Metode Support Vector Machine (SVM) Linear, Radial Basis Function (RBF), dan Polinomial Kernel dalam Klasifikasi Bidang Studi Lanjut Pilihan Alumni UII," 2018.