Molecular & Cellular Biomechanics 2024, 21(2), 354.
https://doi.org/10.62617/mcb.v21i2.354

Article

Basketball player motion detection and motion mode analysis based on biomechanical sensors

Long Liu*, Yucui Pu

School of Health Care, Chongqing Preschool Education College, Chongqing 404047, China
*** Corresponding author:** Long Liu, liulong@cqyz.edu.cn

CITATION

Liu L, Pu Y. Basketball player motion detection and motion mode analysis based on biomechanical sensors. Molecular & Cellular Biomechanics. 2024; 21(2): 354. https://doi.org/10.62617/mcb.v21i2.354

ARTICLE INFO

Received: 11 September 2024
Accepted: 19 September 2024
Available online: 6 November 2024

Abstract: Basketball player motion detection and analysis are crucial for optimizing performance and preventing injuries. Traditional methods often rely on visual observation and video analysis, lacking precision and real-time feedback. In this study, a unique novel Intelligent Bayesian tuned-augmented Support Vector Machine (IB-ASVM) was proposed for predicting basketball players' motion modes and performance analysis using the biomechanical sensor data. Advancements in biomechanical sensors such as accelerometers, gyroscopes, and force sensors are deployed into ESP32 to build a player's wearable gadget. This gadget provides dynamic players with real-time sensing data. Data are transmitted to the cloud via Wi-Fi 7.0 for motion analysis and this model is stimulated using Arduino IDE. The Kalman Filter reduces noise and smoothens sensor data such as acceleration, and angular velocity. Then, the filtered data is employed in the Discrete Wavelet Transform (DWT) to capture time-frequency characteristics of motion signals, making it ideal for extracting relevant features. The featured data are utilized in the ASVM model to classify and detect the motion modes of the basketball players via IB optimization. The Tensor Flow software is used to implement the IB-ASVM model. The result demonstrates that IB-ASVM most accurately predicts the jump shot, layup, dribbling, running, pivoting, passing, free throw, and motion states of the basketball players. The IB-ASVM model accurately classifies basketball motion states using biomechanical sensor data, enhancing performance optimization and injury prevention through precise motion detection.

Keywords: basketball player; motion stage detection; Gadget; biomechanical sensors

1. Introduction

Player motion detection is the measurement and monitoring of players in sports or games to determine their performance or actions. It makes use of different ways of sensing with available tools such as sensors, cameras, and other intelligent algorithms in tracing the movements of different players in real time [1]. Through the application of computer vision, artificial intelligence, and motion capture technologies, player motion detection can accurately catalog actions, gestures, and positions. It is imperative for evaluating the players' performance, promoting the development of the training sessions, and augmenting the game strategies [2]. For example, in the athletic field, motion detection can give decision-makers accurate information about the rate of motion, acceleration, and other characteristics of an athlete's movement; this can be useful for coaching or adjusting a practicing procedure. To anyone involved in the game, it enables them to design better avatars that emulate real-life movements as well as improve on the players' kinetic feedback, which is converted to game-like movements [3]. Further, it supports the avoidance of injuries by tracking how stressed the players are and which movements are detrimental to them. In general, player motion detection is a very effective feature that allows one to address essential objectives of sports performance enhancement and interactive entertainment evolution [4].

A biomechanical sensor is an equipment used to assess mechanical characteristics and also the motion of biosystems most preferably human bodies. These sensors gather information on force, pressure, acceleration, and motion and they give insight into the mechanical characteristics of muscles, bones, and joints [5]. Biomechanical sensors are widely applied for sports science, rehabilitation, or ergonomics to increase the knowledge of human motion and improve the performance or therapy results. For example, in sports, sensors may be integrated into apparel or tools that the athletes use or general sensors built within their athlete's training gears and equipment to monitor athletes' movements and forces and provide information that can guide them on the right techniques, the efficiency of certain movements and possible areas that need adjustment [6]. In rehabilitation, they assist in overcoming the objective, as therapists can track the movements of the patients, as well as the impact of the therapeutic procedures. Also in ergonomics, biomechanical sensors measure human activities in the workplace, which reduces risk factors that result in musculoskeletal disorders and leads to designs of safer working environments and interfaces [7]. These sensors are normally connected to software that can help to analyze and create visualizations on the data that has been gathered and such data is useful for use in improving performance, avoiding injuries, and even aiding in the recovery process. Biomechanical sensors therefore help in closing the gap between movement and analysis by providing insight into human biomechanics [8].

Biomechanical sensors used in basketball player motion detection include the use of wearable technology that records players' movements and the forces applied during the game. These sensors obtain information on such elements as velocity, acceleration, and angles that describe player movements, thus offering valuable information on the players' skills and results [9]. Merging this information with a sophisticated analysis, a coach and an analyst comprehend the efficiency of the player and determine the tendencies for their improvement and prevention of injuries based on the development of special training programs. This makes it possible to measure the biomechanics in action and the process develops and implements the best strategies, hence improving the performances of players [10]. The objective of this research is to use advanced biomechanical sensor data to accurately identify basketball player movements and analyze their performance using an Intelligent Bayesian tuned-augmented Support Vector Machine (IB-ASVM).

Contributions

- Introduces the Intelligent Bayesian Tuned-Augmented Support Vector Machine (IB-ASVM) for precise classification of basketball motion modes using biomechanical sensor data.
- Real-time data processing employs wearable sensors and Wi-Fi 7.0 for dynamic data transmission, with Kalman Filtering and Discrete Wavelet Transform (DWT) optimizing, reducing noise and feature extraction.

- The IB-ASVM model enhances performance analysis and injury prevention by providing precise motion state classification through advanced data processing and real-time feedback.

This article's sections: In Section 2, an inventory of relevant works is provided. The suggested strategies are shown in Section 3. In Section 4, the performance analyses are shown. The outcomes of the suggested study are presented in Section 5.

2. Related work

Basketball posture analysis with inertial sensors was investigated by Li, [11] that used player data. The C4.5, support vector machine (SVM), random forest (RF), and K-nearest neighbor (KNN) algorithms' performances were contrasted. Although C4.5, RF, and SVM methods all obtained excellent classification accuracy, the KNN algorithm produced very high accuracy. The best classification accuracy was achieved by the RF algorithm, demonstrating its superiority in basketball posture detection.

A basketball movement classification framework utilizing body kinematic measurements and dynamic time warping (DTW) was described by Hu et al. [12]. The algorithm demonstrated good accuracy and efficiency in classifying four basketball moves. Twenty people participated in the investigation, including both seasoned gamers and beginners. The model attained specificity, recall, and precision. The viability of employing DTW for real-time sporting event classification was supported by their investigation.

A convolutional neural network (CNN) sensor motion capture system was suggested by Chen and Zhang [13] to track and prevent basketball-related sports injuries. Athletes' heart rates, energy expenditure, acceleration, speed, and stride frequency were all tracked in real time by the system. Training and optimization increase the system's precision and accuracy. In addition to helping with training and tactical modifications, their technology may provide coaches and players access to thorough, scientific data that would improve both the safety and effectiveness of basketball games.

To categorize the ability levels of amateur and professional basketball players, Guo et al. [14] employed information from a tri-axial inertial sensor regarding the wrist. For classification, a fully connected CNN was employed. The wrist angular velocity in the sagittal plane, upward acceleration, and forward acceleration were the most accurate components. Combining these three axes into a stack-up matrix produced the best accuracy.

Employing a model of machine learning founded on the Bayesian optimization of a light gradient boosting machine (LightGBM), Zhao et al. [15] described a method to quantitatively analyze four main activity indicators of shooting basketballs. For testing and training the model, 16 subjects' information was gathered. With the highest correlation scores for individual players, the LightGBM model was designed for regression forecasting. Their method can offer unbiased, fact-based advice for raising shooting efficiency.

A method for tracking biophysical, kinematic, and environmental variables during basketball practice was provided by Guembe et al. [16]. To give precise estimates of coverage and capacity, it made use of wireless body area networks and on-body nodes equipped with several sensors. During a game, the system was developed, put into practice, and tested in authentic training environments. An in-house stochastic 3D ray launching technique was used to generate the wireless channel assessment results, which provide pertinent data for each player and team feature.

A machine learning (ML) method was suggested by Taborri et al. [17] to calculate the threat of anterior cruciate ligament (ACL) injuries in players. Female athletes participated in the research, which assessed load absorption, leg mobility, and leg stability. The degree of danger was established utilizing the Landing Error Score System (LESS).

A squeeze convolutional gated attention (SCGA) deep learning (DL) method for recognizing basketball shooting postures using sensor fusion was presented by Fan et al. [18]. Their research provided an example of sporting technical movement recognition and demonstrated the model's capabilities in identifying different sensor-fusion basketball shooting positions.

Rugby players' inertial sensors can be used to apply machine learning to distinguish between healthy groups and the ones that have suffered an ACL injury it was explored in an investigation by Tedesco et al. [19]. Inertial data was analyzed to identify features, and male subjects were enlisted. Their investigation showed that post-ACL gait patterns can be identified in athletes completing sport-related tasks years after injury utilizing body-worn motion detectors and ML approaches.

Using deep learning techniques, Zhang [20] attempted to enhance human motion recognition (HMR) structures during sporting events. With a greater recognition rate over existing methods, a new HMR technique based on multidimensional feature fusion (MFF) and the kernel extreme learning machine (KELM) was proposed. The system took 30 and 15 seconds for training and testing, respectively, and has a higher feature identification rate for color dimension.

A unique early-warning detection method for athlete workload exposure was presented by Johnson et al. [21] as a means of preventing catastrophic non-contact knee injuries. Existing techniques were labor- and money-intensive, confined to laboratory equipment, and necessitated specialized knowledge. A novel approach was suggested to get ground kinetics in the environment by utilizing accelerometers from wearable sensors. The strongest mean across ground reaction forces (GRF) components and ground reaction moments (GRM) was observed in the deep learning workbench, which demonstrated correlations to ground truth.

A novel method for developing and evaluating wearable motion systems using biomechanical modeling and sensor data generation was presented by Derungs and Amft [22]. By affixing sensor models to body parts, the technology made it possible to combine motion sensor data with gait marker estimate algorithms. Two instances evaluating the methodology demonstrate how running speed influences the accuracy of gait marker estimation. With the help of their methodology, designers may swiftly evaluate potential solutions and develop customized devices for patients with mobility impairments.

Lisca et al. [23] concentrated on a useful technique for deriving analytics-relevant insights from goalie kinematics. To distinguish between different dive kinds and multi-class classifications, it assessed machine learning techniques using both raw and quaternion data. The findings demonstrated that, when taking into account both raw and quaternion data, XGBoost scores better than alternative strategies. Additionally, the model offered XGboost deployment that was both efficient and explained forecasts. Using online data collected from seven goalkeepers throughout 30-minute training sessions, the investigation assessed its approach.

An affordable artificial intelligence (AI) + Internet of Things (IoT) system framework for identifying football motion and assessing motion intensity was provided by Xie et al. [24]. To reduce communication latency and computational resource consumption, the model simultaneously completed classification and regression tasks using a multi-task learning approach. The multitasking model exhibited good computational accuracy and efficiency, according to experimental data.

3. Materials and methods

The system consists of several sub-systems at both hardware and software levels, as described below. For hardware, accelerometers, gyroscopes, as well as force sensors are used for tracking the players' movement. It connects these sensors to an ESP32 microcontroller for data transmission while a battery ensures a continuous power supply. Wi-Fi 7.0 is used in cloud data transfer. As for the organization of the software, the Kalman Filter is employed to work with noises, and the Discrete Wavelet Transform (DWT) is used for features of extraction. The type of model that is involved in classifying the motions is known as the IB-ASVM model, and TensorFlow is implemented for it. Arduino IDE is used in simulating and programming the proposed IB-ASVM model, while TensorFlow is used in machine learning operations. The general flow is shown in **Figure 1**.

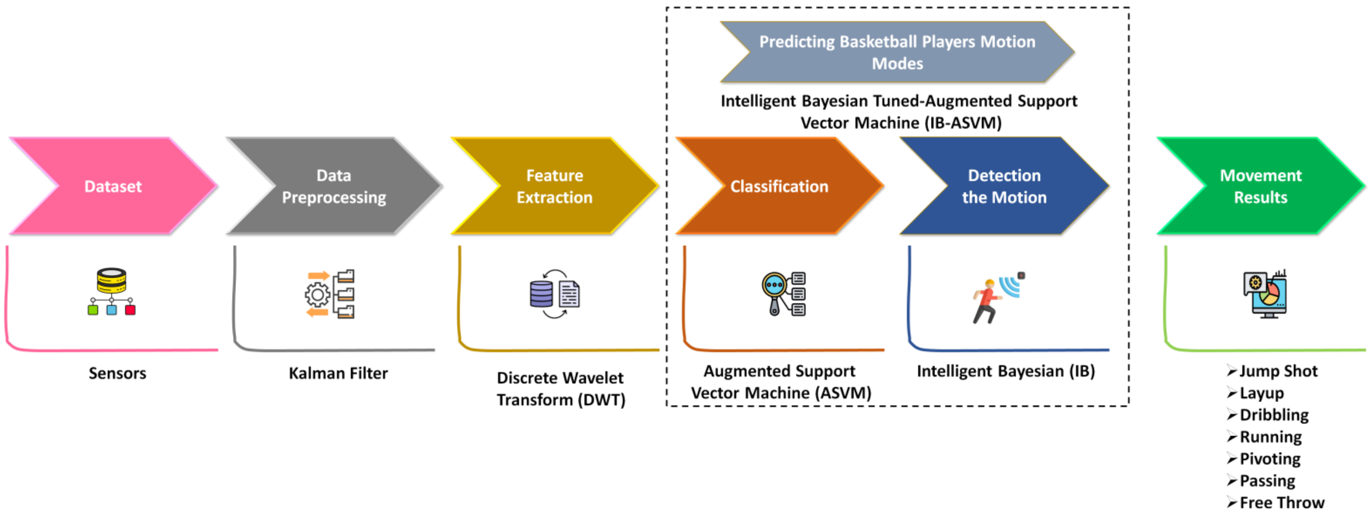

Figure 1. Overview of general flow diagram.

3.1. Data from sensors

In basketball player motion detection and analysis, three key sensors play a vital role including accelerator meters, gyroscopes, and force sensors.

- These include accelerometers, which capture an intensity of motion commonly referred to as velocity, and speed or the direction in which the player is moving, usually when he is sprinting or jumping.
- Gyroscopes record the speed at which an object rotates, such as when the player turns around or changes his or her orientation. All of the above sensors in combination assist in tracking complex movements such as dribbling or pivoting.
- To name a few examples, force sensors are used to determine pressure or impact made by the player: for example, when jumping or landing, or at the time of touching something during the game.

These sensors operate in parallel to provide a complete image and classify accurately with the help of our higher classification model namely the Intelligent Bayesian tuned-augmented Support Vector Machine (IB-ASVM) model to improve performance and avoid injuries.

A total of 2990 examples were collected, of which 1490 consisted of basketball motions performed with the upper limb and 1500 with the lower limb. The basketball player measured the number of distinct motions, as indicated in **Table 1**.

Table 1. The variety of motions made by the upper and lower limbs.

Upper and lower limbs	Movements	Quantity
Arm movements	Jump shot	250
	Layup	250
	Dribbling	250
	Pivoting	290
	Passing	200
	Free throw	250
Leg movements	Jump shot	250
	Layup	260
	Dribbling	300
	Running	240
	Pivoting	250
	Free Throw	200

3.2. The suggested system framework

3.2.1. Hardware

Accelerometers: An accelerometer is a kind of gadget that is used to measure velocity or acceleration. They measure the small changes in the magnitude and direction of the linear motion in different axes and are very important in determining the linear velocity. When talking about the basketball player’s motion detection system, accelerometers help record moves like running, jumping, or even dribbling by sensing the speed and direction of particular movements. It has been used when studying the dynamics of gameplay and more specifically the players to consider detailed movements of the players as well as enhance motion analysis.

Gyroscopes: Gyroscopes deal with angular velocity, which happens to be the rate of rotation in terms of an axis. They are very useful for sensing rotational movements and changes in the orientation of the object or frame of reference involved. In the system, a gyroscope records two types of movements, including turning, pivoting, or body orientation movements that occur when shooting basketballs. It appears to be especially important for an accurate representation of complex movements' execution and for increasing the level of motion data detail to provide a more refined assessment of players' performance.

Force sensors: Load cells determine the amount of force or pressure applied on a surface as it comes with impact, pressure, or strain. When choosing the basketball player's motion detection system, force sensors are used to quantify the impact forces during performing actions such as jumping or landing. This data assists in having details concerning the mechanical loads on players and this is important for performance evaluation and for reducing the chances of injury to the players.

ESP32: The ESP32 is a microcontroller that was specifically designed with in-built Wi-Fi and Bluetooth functionalities. It plays the role of a master data processing and transfer entity for different applications. In the motion detection system, the ESP32 microcontroller harmonizes all the sensors and also analyses the data collected before forwarding the data to the cloud through the Wi-Fi connection available for the module. It has a crucial function in performing management and relaying information as received from the sensors, hence having to deal with the actual data to analyze the situation in real-time.

Wi-Fi 7.0: Wi-Fi 7.0 is an enhanced standard in wireless communication with capabilities of enhanced data transfer rates and enhanced network accessibility than the previous standards. In the system, the sensor data of the wearable device could be transferred to the cloud more quickly and efficiently by Wi-Fi 7.0. This capability boosts the efficiency of motion detection and performance tracking by reducing the delay involved in transmitting data to improve the time and accuracy of the player's movements.

3.2.2. Software

Table 2. Details of hardware and software.

Category	Component	Details
Hardware	Sensors	Accelerometers, gyroscopes, and force sensors for tracking motion.
	Wearable device	ESP32 microcontroller for sensor integration and data transmission.
	Battery	Power source for continuous operation of the wearable device.
	Transmission	Wi-Fi 7.0 for cloud data transfer.
Software	Processing	Kalman Filter for noise reduction and Discrete Wavelet Transform (DWT) for feature extraction.
	Classification	IB-ASVM model for motion state classification. TensorFlow for model implementation.
	Development	Arduino IDE is used for simulating and programming the IB-ASVM model, while TensorFlow is implemented to handle machine learning tasks within the IB-ASVM model.

Arduino IDE: Arduino IDE, the integrated development environment, is software that allows writing, testing, and uploading code to microcontrollers such as

ESP32. It is easy to use and programs and tests embedded systems with the help of this application software. In the basketball player motion detection system, the actual programming of the ESP32 microcontroller happens in the Arduino IDE, where the simulation of the system takes place. With this kind of setup, the code that manages the collection and transmission of data from the sensors can be developed and tested to make sure that the system works as planned. Here, by using Arduino IDE, the individual can enhance various aspects of ESP32 in their motion detection system. **Table 2** shows the hardware and software details.

3.3. Data preprocessing

In the case of applying data preprocessing with the Kalman Filter, noise is removed and sensor data including acceleration and angular velocity is smoothed. This improves the reliability of the motion measurements significance that in the basketball player motion detection system, there will be more accuracy of the data to be used for analysis.

Kalman filter

The use of the Kalman filter in the basketball player motion detection system helps filter out the noise and increment the accuracy of measurements. PEC provides accurate estimations of motion parameters such as acceleration and angular velocity, which are very important in performance analysis and risk prevention of injuries.

A collection of mathematical formulas known as the Kalman filter, by reducing the average square error, may effectively and recursively anticipate the state of a process. The Kalman filter's ability to estimate a scenario from little data is one of its benefits. The equations for the time update and measuring update are the two groups that make up the Kalman filter equation. A time update, which derives its current state estimation from a past time estimate is also known as a forecast process. Simultaneously referred to as the "correct process," the measurements update is the measurement data that is currently utilized to enhance forecasts in the hopes of obtaining a more precise state estimate.

The Kalman filter's predictions and accurate equations are as follows:

Predict:

$$\widehat{W}_{s|s-1} = E_s \widehat{W}_{s-1|s-1} + A_s V_s \tag{1}$$

$$O_{s|s-1} = E_s O_{s-1|s-1} E_s^S + R_s \tag{2}$$

Correct:

$$\widehat{W}_{s|s} = \widehat{W}_{s|s-1} + L_s \left(z_s - G_s \widehat{W}_{s|s-1} \right) \tag{3}$$

$$L_s = O_{s|s-1} G_s^S \left(G_s O_{s|s-1} G_s^S + Q_s \right)^{-1} \tag{4}$$

$$O_{s|s} = (1 - L_s G_s) O_{s|s-1} \tag{5}$$

A state transition matrix is denoted by E, an estimated state by $\widehat{W}$, control variables by V and A, and measurement variables by G and z are all present. State variance matrix (O), matrix of process variance (R), Kalman filter gain (L), Process variance matrix (S), and matrix of measurement variance (Q) are all examples of variance matrices.

The results of the gyroscope and accelerometer sensors must be adjusted for the algorithm for the Kalman filter to function. Among the modifications are:

1) Predict the state

Equation (1) was used to estimate the initial state of the process; the parameters A and V are not employed, and the value E = 1. They both have a value of 0. Consequently, the equation that forecasts the situation is,

$$\widehat{W}_{s|s-1} = \widehat{W}_{s-1|s-1} \tag{6}$$

After that, Equation (6) is converted to a computer language and becomes $Ws\_1 = Ws\_1s\_1$;

2) Predict error

Equation (2) makes use of the equation in this procedure. E = 1's value had already been ascertained. Consequently, the equation that forecasts flaws is,

$$O_{s|s-1} = O_{s-1|s-1} + R_s \tag{7}$$

Next, Equation (7) is converted to the computer language as follows: $Os\_1 = Os\_1s\_1 + R$;

3) Calculate Kalman Gain

Equation (4) can be used to calculate the Kalman Gain for an adjustment if G = 1. Consequently, the Kalman Gain equation is,

$$L_s = \frac{O_{s|s-1}}{\left(O_{s|s-1} + R_s\right)} \tag{8}$$

After that, Equation (8) is converted to a computer language and becomes, $Ls = Os\_1/(Os\_1 + Q)$;

4) Modify the State Value by the test's findings.

Equation (3) can be employed to adjust the state value. The state value is updated using the outcomes of Equation (6), which represents the first state forecast, and Equation (8), which represents the value of Kalman Gain. Already established G = 1's value. As a result, the equation that updated state values is

$$\widehat{W}_{s|s} = \widehat{W}_{s|s-1} + L_s\left(z_s - G_s\widehat{W}_{s|s-1}\right) \tag{9}$$

Equation (9) is then converted to the computer language as follows, in which case z is the gyroscope/accelerometer sensor's result value: $Ws = Ws\_1 + Ls * (accelerometer\_w - Ws\_1)$;

5) Update Error value

Then it may employ Equation (5) to update the value of the error. Equation (7), representing the original error forecasting, and Equation (8), representing the value of Kalman Gain, are utilized to update the value of error. The $G = 1$ value was already established. Therefore, the formula that was applied to update the incorrect value is

$$O_{s|s} = (1 - L_s)O_{s|s-1} \tag{10}$$

Next, let's convert Equation (10) to the computer language as follows: $Os = (1 - Ls) * Os\_1$;

Initialization, which tries to ascertain the starting of certain variables' values since the actual parameter has not been received, requires the Kalman filtering method to be executed. The variable Ws_1s_1 is the value of the previous state, Os_1s_1 is the value of the prior error, R is an error of process and Q (measurement error) all have initialization values. To acquire good filter results, the results of these variables are determined by chance, and the observed findings are then examined. The values of the variables R and Q in this study are split into three categories: $R > Q, R = Q, and\ R < Q$.

3.4. Feature extraction

Feature extraction using DWT comprises the analysis of motion signals that were decomposed into time-frequency signals. Specifically, it considers features of the signals like patterns and variations in the signal, hence providing proper distinction of the motion states.

Discrete Wavelet Transform (DWT)

Basketball motion data is decomposed into time-frequency and allows the DWT to examine time-specific details of motion patterns. It improves upon the capability of identifying various types of motion that are involved in basketball, which include dribbling, and shooting among others.

A mathematical technique suitable for time scale evaluation in time series is DWT. DWT is a useful technique for dividing nonstationary signals into scale-different sub-bands. A method for processing nonstationary and nonlinear signals is called the DWT. Wavelets' output may mimic the original signal at every time-frequency scale. When analyzing nonstationary time series, such workload traces information from cloud data centers, the wavelets' output is utilized to determine the pattern of sequence changes by capturing the original signal's features. The original host load traces are broken down into a series with different frequencies by the DWT. Mallat's technique performs the decomposition, effectively breaking down nonstationary hosting load traces into several sequences of varying frequencies by the use of high-pass and low-pass filters. Sub-bands are thus present in the outputs of the high-pass and low-pass filters. The sub-bands defined by dB and dC in Equations (11) and (12) are referred to as approximation coefficients and detailed coefficients:

$$dB = \sum_{l=-\infty}^{+\infty} W[l]\, \varphi_k[2m - l] \tag{11}$$

$$dC = \sum_{l=-\infty}^{+\infty} W[l]\, \varphi_g[2m - l] \tag{12}$$

The original host load traces signal is represented by W; the filter is denoted by φ; the low-pass and high-pass filters are represented by L and H, respectively. **Figure 2** illustrates the three-level decomposition process using the Mallat algorithm. In the first layer, lowpass and high-pass filters break down into the initial hosting load traces W. The process of decomposition yields detailed and approximate sub-bands, d_1 and a_1. Two coefficients, a_2 and d_2, are obtained in the second layer by passing the acquired d_1 sub-band across two highpass and lowpass filters once more. Similarly, two low-pass and high-pass filters are applied to the resulting d_2 sub-band in the third layer to yield two coefficients, d_3 and a_3. It is possible to receive several components with various frequencies after the decomposition procedure.

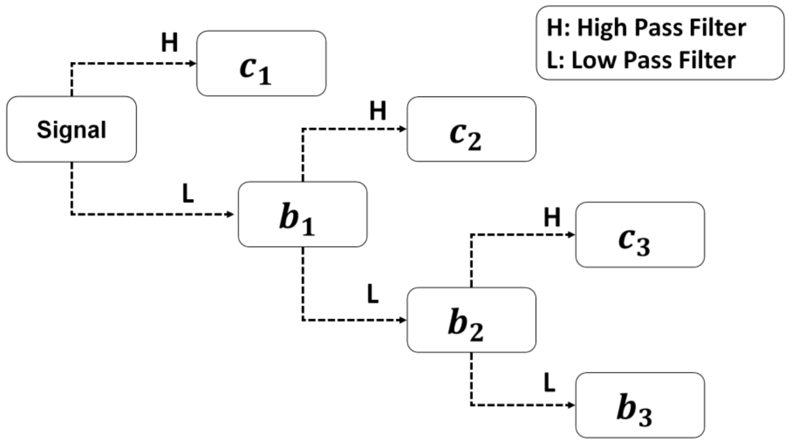

Figure 2. A DWT schematic diagram.

Following the extraction of data, the obtained data is subjected to the ASVM model with a view of categorizing the motion states. This process entails using the Augmented Support Vector Machine for the classification of different movements concerning the basketball movement and the features extracted from the sensor data.

3.5. A hybrid of Intelligent Bayesian Tuned-Augmented Support Vector Machine (IB-ASVM)

The Hybrid Intelligent Bayesian-tuned augmented Support Vector Machine (IB-ASVM) employs superior Bayesian optimization in improving the performance of an Augmented Support Vector Machine (ASVM) to identify basketball players' motion modes. As a result of the integration of intelligent Bayesian tuning, the IB-ASVM accurately fine-tunes hyperparameters that enhance the model's accuracy and stability. This dual approach is used with biomechanical sensor data and involves mode recognition of the different motion modes and the player analysis of performance with extreme accuracy. It provides a better approach to understanding the motion trajectories and performance indices in detail and helps to spit new knowledge for training and injury prevention in basketball. The algorithm for IB-ASVM is shown in Algorithm 1.

Algorithm 1 Pseudocode for IB-ASVM

```
import tensorflow as tf
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from bayes_opt import BayesianOptimization
import numpy as np
def create_and_train_svm(C, gamma):
svm_model = SVC(C = C, gamma = gamma, kernel = 'rbf')
svm_model.fit(X_train, y_train)
y_pred = svm_model.predict(X_val)
accuracy = np.mean(y_pred == y_val)
return accuracy
X, y = load_biomechanical_data() # Implement this function to load data
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size = 
0.2, random_state = 42)
def optimize_svm(C, gamma):
return create_and_train_svm(C, gamma)
param_bounds = {
'C': (0.1, 10),
'gamma': (0.001, 1)
}
optimizer = BayesianOptimization(
f = optimize_svm,
pbounds = param_bounds,
random_state = 42
)
optimizer.maximize(init_points = 10, n_iter = 30)
best_params = optimizer.max['params']
best_C = best_params['C']
best_gamma = best_params['gamma']
final_svm_model = SVC(C = best_C, gamma = best_gamma, kernel = 'rbf')
final_svm_model.fit(X_train, y_train)
final_accuracy = final_svm_model.score(X_val, y_val)
print(f'Final model accuracy: {final_accuracy}')
```

3.6. Classification through ASVM

The ASVM categorizes motion states using the information collected from the sensors. There are employed more complex optimization methods to improve the accuracy and to differentiate one or another kind of basketball movement, for instance, dribbling or shooting, based on the features of biomechanical data.

Augmented Support Vector Machine (ASVM)

By defining new features and optimizations, the ASVM improves the traditional concepts of SVMs. Specifically, in basketball motion detection, ASVM categorizes different types of movement of a player by using biomechanical sensor data and thus enhances the specificity and accuracy of motion pattern identification.

Augmented support vector machines can implicitly transfer non-linearly separable data points onto a new dimension while they're separable linearly due to kernels. Cost is involved while mapping the data points to higher dimensions. Larger vectors need more memory and take longer to calculate when they are more dimensional. There is no requirement for SVMs to explicitly store these large dimensional vectors. They are only used to store inner products after mapping the input data into a higher dimension. Various mappings are offered by various kernel functions. However, there isn't a perfect kernel to choose from. Every kernel has benefits and drawbacks. The accuracy of the SVM's data point classification is significantly impacted by the kernel selection. It adapts the current Gaussian kernel to suit our needs. The performance of this modified kernel is superior to that of the standard Gaussian kernel.

Each example from the input space Q is mapped into the feature space E using a nonlinear SVM with a mapping that is not linear $\emptyset$. An encapsulation of T into Eas a curve submanifold is defined by the mapping $\emptyset$.

In the highlighted space, $\emptyset(w)$ denote the mapped examples of T. The tiny vector dw is mapped to:

$$\emptyset(dw) = \nabla\emptyset, dw = \sum_j \frac{\partial}{\partial w^{(j)}} \emptyset(w) \partial w^{(j)} \tag{13}$$

where, $\nabla\emptyset = \frac{\partial}{\partial w^{(j)}} \emptyset(w)$

$\emptyset(dw)$squared length can be expressed as follows:

$$dt^2 = |\emptyset(dw)|^2 = \sum_{j,i} h_{ji}(w) dw^{(j)} dw^{(i)} \tag{14}$$

where:

$$h_{ji}(w) = \left(\frac{\partial}{\partial w^{(j)}} \emptyset(w)\right) \times \left(\frac{\partial}{\partial w^{(i)}} \emptyset(w)\right) \tag{15}$$

The dot represents the total over $\emptyset$ index α. Induced in T, the Riemannian metric tensor is the $m \times m$Positive-definite matrix $H(w) = \left(h_{ji}(w)\right)$.

$$h_{ji}(w) = \frac{\partial}{\partial w^{(j)}} \frac{\partial}{\partial w^{(j)}} L\left(w, w_j\right) \tag{16}$$

To boost the SVM's efficiency, we can extend the variance or the distances (ds) among classes. This causes us to enhance the Riemannian metric tensor at the boundary and decrease it around other samples when we consider Equation (14). It can change the kernel L so that $h_{ji}(w)$ is significant around the boundary in light of Equation (16).

Kernel modification according to Riemannian geometry framework: Let's say the kernel may be changed to:

$$\tilde{L}\left(w, w_j\right) = o(w) o\left(w_j\right) L\left(w, w_j\right) \tag{17}$$

Equation (17) is known as a kernel's conformal transformation with factor $o(w)$. It considers the Gaussian kernel, which is the kernel function utilized in SVM, i.e.

$$L\left(w, w_j\right) = \exp\left(-\frac{\left|\left|w - w_j\right|\right|^2}{\sigma^2}\right) \tag{18}$$

Kernel width is the value of parameter σ in this instance. The related Riemannian metric tensor is shown to be transformed into:

$$h_{ji}(w) = \frac{1}{\sigma^2}\delta_{ji} \tag{19}$$

Following kernel Riemannian metric modification, the tensor is transformed into:

$$\tilde{h}_{ji}(w) = o_j(w)o_i(w) + o^2(w)h_{ji}(w) \tag{20}$$

To guarantee that $o(w)$ has a large value surrounding the support vector (SV), the Gaussian kernel's conformal transformation.

$$o_j(w) = \frac{\partial o(w)}{\partial w_j} \tag{21}$$

$o_j(w) = 0$ for all maximal values of $o(w)$.

It is possible to create $o(w)$ in a data-dependent manner as follows to guarantee that it has significant amounts at the support vector positions:

$$o(w) = \sum\nolimits_{j\in SV} \alpha_j \exp\left(-\left|\left|w - w_j\right|\right|^2 /2\tau^2\right. \tag{22}$$

where the summation is applied across all support vectors and $\tau$is a free parameter. As can be seen, $o_j(w)$ and $o(w)$ grow when w approaches support vectors and decreases when w moves away from SVs. Additionally, $h_{ji}(w)$ increases when $w$approaches support vectors. The result is an increase in the spatial resolution at the boundary and a stronger SVM's capacity for classification.

Following the classification step, there is the Intelligent Bayesian (IB) optimization, which optimizes detection by tweaking the model using probability theory or reasoning. This process enhances the predictions of the motion state, hence enhancing the dependability as well as the accuracy of the basketball player's motion analysis.

3.7. Detection motion node through IB

Identification of motion nodes through intelligence (IB) entails the use of probability distribution models in identifying and categorizing specific movements. This approach helps make motion detection more accurate by using Bayesian inference to analyze the multivariate sensor data to increase the accuracy of the system.

Global hyperparameters with Bayesian optimization

Hyper-parameter optimization techniques fall into two categories: automatic and manual methods for searching. Because manual tuning of hyper-parameters relies heavily on trial and error, it is a difficult process to replicate. Higher dimensions are not compatible with grid search. By accepting local optimal instead of aiming for global optima, random search behaves similarly to the greedy strategy. Some evolutionary algorithms for optimization can be noisy and need more training cycles. As previously mentioned, Bayesian optimization that is based on the Bayes theorem can effectively discover the broad optima of the neural network's black box function, overcoming all of these limitations. A technique for finding the extrema in computationally expensive functions is called Bayesian optimization. The key elements in the optimization process are:

$E(w)$ as a Gaussian process framework. For every new assessment of $e(w)$, a Bayesian update mechanism is used to modify the process's Gaussian framework.

The Gaussian process framework provides the basis for the acquisition function $b(w)$that is maximized to identify the subsequent point w for assessment.

With the use of such a mechanism, it is possible to determine the point at which the function achieves its ideal value, minimizing loss and increasing the accuracy of the simulation. As mentioned, the optimization objective of this work is to determine the lowest possible loss value for a function that is unknown sampling point:

$$w_{opt} = \arg\min_{w \in C} e(w) \tag{23}$$

Where, C stands for w's search space

With additional Gaussian noise in the assessments, the fundamental probabilistic model for the goal function e is a Gaussian procedure prior. A multivariate distribution with Gaussian characteristics can be found for every finite sub-collection of unknown variables in a Gaussian process, which is a generalization of the Gaussian probability distribution. Since the Gaussian process operates under the assumption that outputs will resemble inputs, it relies on a statistical framework of the function.

$$O(N|F) \infty O(F|N) O(N) \tag{24}$$

The concept of Bayesian optimization can be seen in Equation (24). When considering the sample information F, the following probability $O(N|F)$ of an explanation N is equal to the product of the initial probability of $O(N)$ and the probability $O(F|N)$ of witnessing F given model N. In summary, the posterior of the function is obtained by integrating the present sample data with an earlier distribution determined by the Gaussian procedure of the function $e(w)$ to optimize the unknown function using Bayes optimization. The function $e(w)$ reduces through the criteria value in the following phase, which is determined using the posterior information. An acquisition function, or utility function, a, represents the criterion. To maximize the predicted utility, the subsequent point of sampling is defined using the function b. A small number of frequently utilized acquisition functions exist. The expected improvement (EI) function, which ignores values that raise the objective, is utilized in this work to assess the expected degree of enhancement in the target function. Operation when a point is explored in the region of the present optimum value, EI computes the expected degree of enhancement that it can achieve. If the lowest subsequent mean is located atw_{best}, and the lowest posterior mean value is represented by $\mu_R(w_{\text{best}})$, then the anticipated improvement is:

$$FJ(w, R) = F_R[\max(0, \mu_R(w_{best}) - e(w)]\, O(N|F) \infty O(F|N) O(N) \tag{25}$$

The present ideal value position may be the best option for the area if the method finds an ideal value point in different domain places and the enhancement of the value of the function is smaller than the predicted value once the procedure runs. To reduce the number of samplings, the sampling region is searched using both exploration (sample from areas with high unpredictability) and exploitation (testing from areas with high values). Ultimately, even in cases when the function has more than one local maximum, performance will indeed improve. The initial distribution of the coefficient e is a necessary component of the statistical calculation of the function's posterior distribution, and it is also dependent upon sample data in Bayesian optimization.

The following are the key phases in the optimization process:

1) In the present iteration, s for a specified amount of randomly selected points w_j inside the variable boundaries, evaluate $z_j = e(w_j)$.
2) To get an after distribution over functional $R(e|wj, zj\ for\ j = 1, \ldots \ldots, s)$, modify the Gaussian process framework for $e(w)$.
3) Locate the new point w where acquisition function $b(w)$ is maximized.

When the time and quantity of iterations limit is reached, the method terminates.

4. Result and discussion

Our strategy was carried out in Arduino IDE and TensorFlow on Windows 10 OS. The system is driven by an Intel Core i5 processor and features a high-performance IRIS graphics card, delivering strong capacity for carrying out complex ML tasks.

The performance of the proposed technique (IB-ASVM) was examined using a set of metrics including accuracy, precision, recall, and f1-score. The accuracy is compared with existing methods: ResC3D Network [25], EfficientNet-B0 [25], ShuffleNetV2 [25], and Deep Neural Network (DNN) [25]. The other metrics are compared with the movement factors of ASVM and our proposed method.

The improved performance of the IB-ASVM approach in these metrics highlights how well it can anticipate the motion modes of basketball players with increased accuracy and consistency. This major benefit is especially helpful for precise and in-depth motion analysis, which is necessary for maximizing player performance, customizing training regimens, and making wise strategic choices in basketball. **Figure 3** confirms that the IB-ASVM method is a highly sophisticated and successful strategy for motion mode recognition, enabling significant advancements over current methods and a reliable instrument for in-depth research and basketball performance improvement.

Precision: The percentage of actual positive motion modes among all the model's positive predictions, demonstrating the model's emphasis on pertinent cases. Comparatively, the ASVM technique and the IB-ASVM method were evaluated, with the IB-ASVM method demonstrating higher precision values across all motion modes. Layup had the lowest precision at 94.3% while dribbling and running achieved the highest precision at 98.7%. The suggested approach shows better precision scores for predicting basketball players' motion modes.

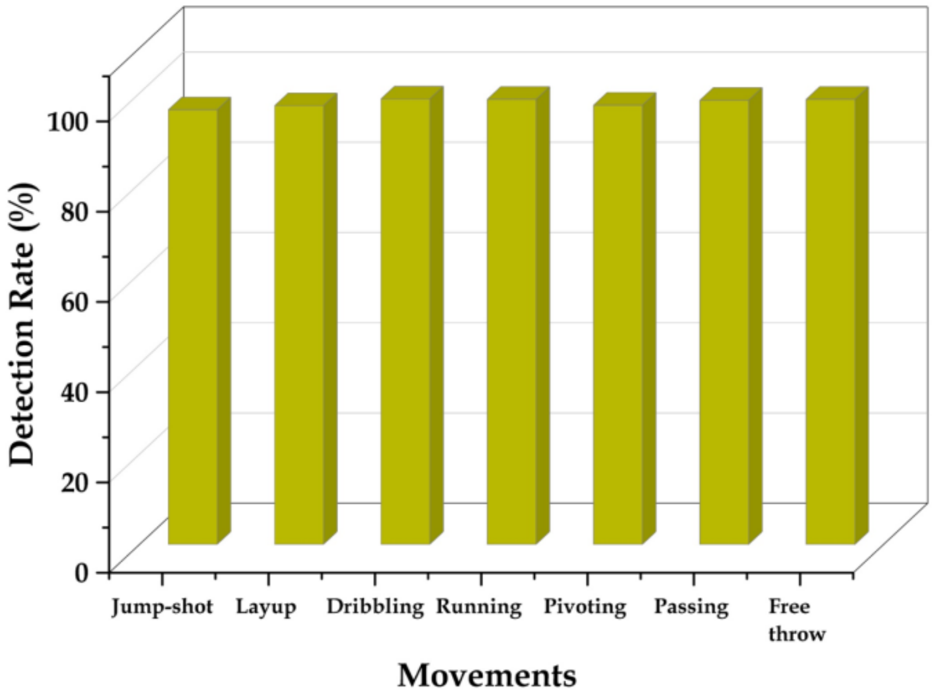

Figure 3. Result of detection rate.

Figure 4's results show that, when it comes to motion mode detection, the IB-ASVM method outperforms the ASVM technique in terms of precision. This increase in accuracy is a result of the model's capacity to correctly categorize the movements of basketball players, which facilitates more efficient player performance analysis and optimization.

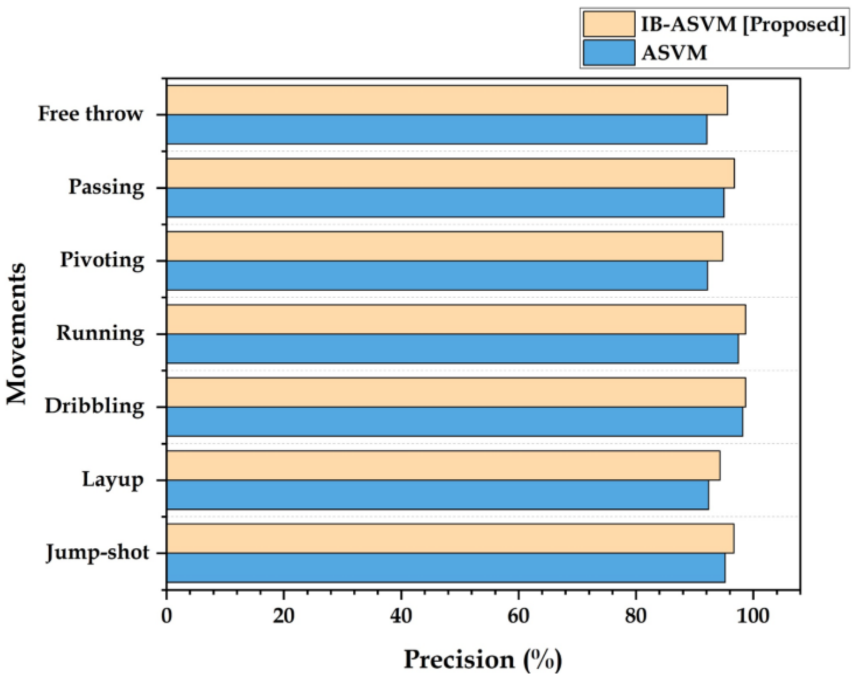

Figure 4. Comparison of precision.

Recall: The percentage of real positive motion modes that are correctly determined out among all actual positives, indicating how well the model can detect motion. Comparatively, the ASVM technique and the IB-ASVM method were evaluated, with the IB-ASVM method demonstrating higher recall values across all motion modes. Running had the lowest recall at 94.6% while passing achieved the highest recall at 98.6%. The suggested approach shows better recall scores for predicting basketball players' motion modes.

In comparison to the ASVM approach, **Figure 5** demonstrates that the IB-ASVM method offers better recall for identifying the motion modes of basketball players. To maximize player performance and training methods, more accurate and dependable motion detection is made possible by the model's increased memory power, which makes it more skilled at catching genuine occurrences of motion modes.

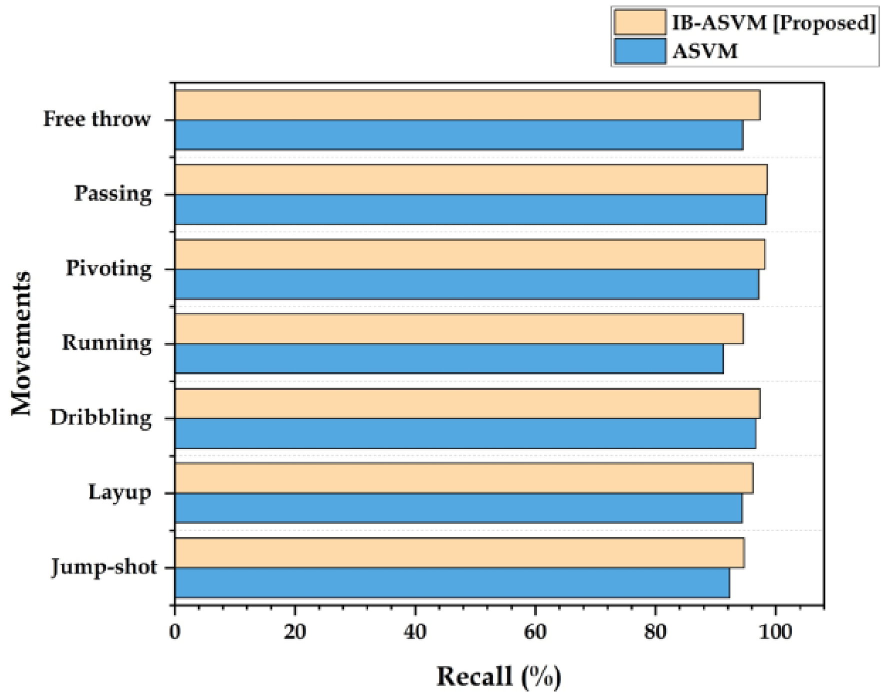

Figure 5. Comparison of recall.

F1-score: The model's overall efficacy in predicting motion modes under various settings is evaluated by calculating the harmonic average of recall and precision, combining both metrics. Comparatively, the ASVM technique and the IB-ASVM method were evaluated, with the IB-ASVM method demonstrating higher f1-score values across all motion modes. Jump-shot had the lowest f1-score at 95.7%, while free throw achieved the highest f1-score at 98.3%. The suggested approach shows better f1-scores for predicting basketball players' motion modes.

As shown by its higher F1 scores than the ASVM technique, **Figure 6** validates that the IB-ASVM method is superior at predicting the motion modes of basketball players. Basketball players' performance analysis and training interventions will be of higher quality because of the method's capacity to forecast motion modes in a balanced and precise manner, as demonstrated by the enhanced F1 scores.

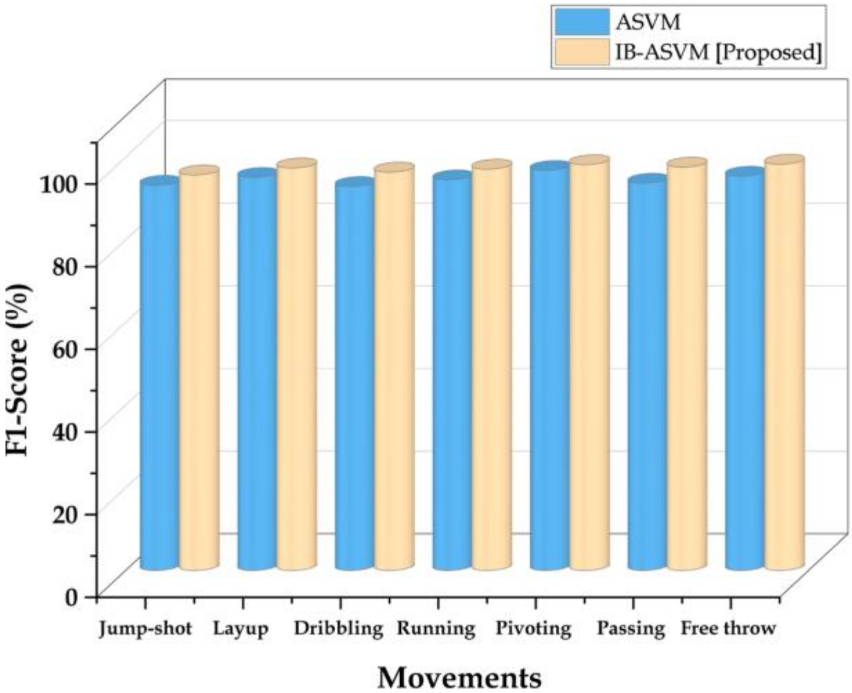

Figure 6. Comparison of f1-score.

Training time: The entire amount of time, calculated from the beginning of training to the moment of convergence, that the model needs to acquire information from the training data. It affects the viability of model deployment, particularly in situations involving real-time motion mode forecasting. The ASVM technique achieved 38.7 training time per minute, while the proposed IB-ASVM achieved 35.2 training time per minute. The suggested approach shows a better training time value for predicting basketball players' motion modes.

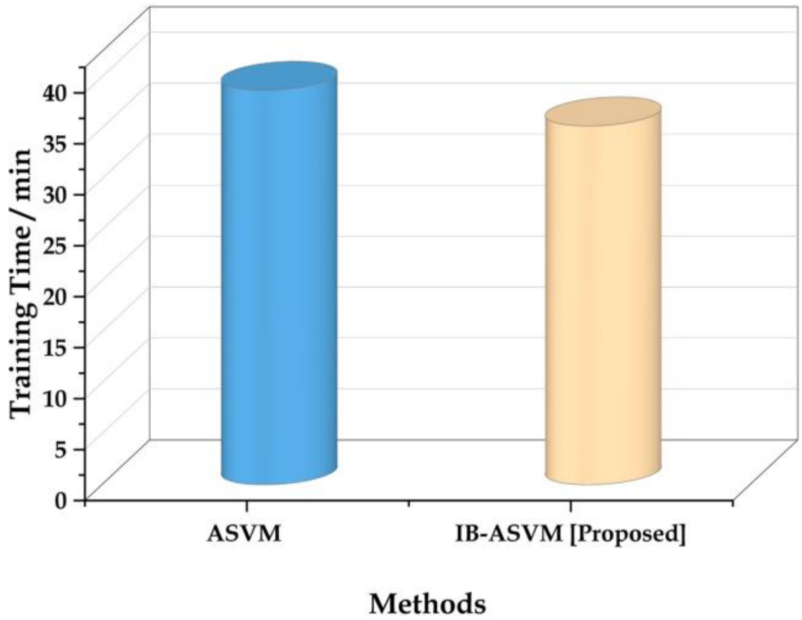

Figure 7. Comparison of training time.

The IB-ASVM approach outperforms the ASVM technique in terms of training time efficiency, as shown in **Figure 7**. As a result of this longer training period, the model performs better in predicting the motion modes of basketball players and is more appropriate for real-time applications and deployment. It also offers faster updates.

Accuracy: It is the proportion of all predicted accurate motion modes. It displays the overall performance of the model in all areas. The existing algorithms, such as ResC3D Network, EfficientNet-B0, ShuffleNetV2, and Deep Neural Network (DNN), achieve 86.5%, 88.3%, 88.9, and 97.82% accuracy, while the proposed IB-ASVM achieves 98.6%. The suggested approach shows a better accuracy score for predicting basketball players' motion modes.

Comparing the IB-ASVM technique to other existing algorithms, **Figure 8** shows that it obtains the highest accuracy in predicting the motion modes of basketball players. This increased accuracy highlights how well the IB-ASVM technique predicts motion modes with accuracy and dependability, which makes it a useful tool for analyzing and optimizing performance in basketball training.

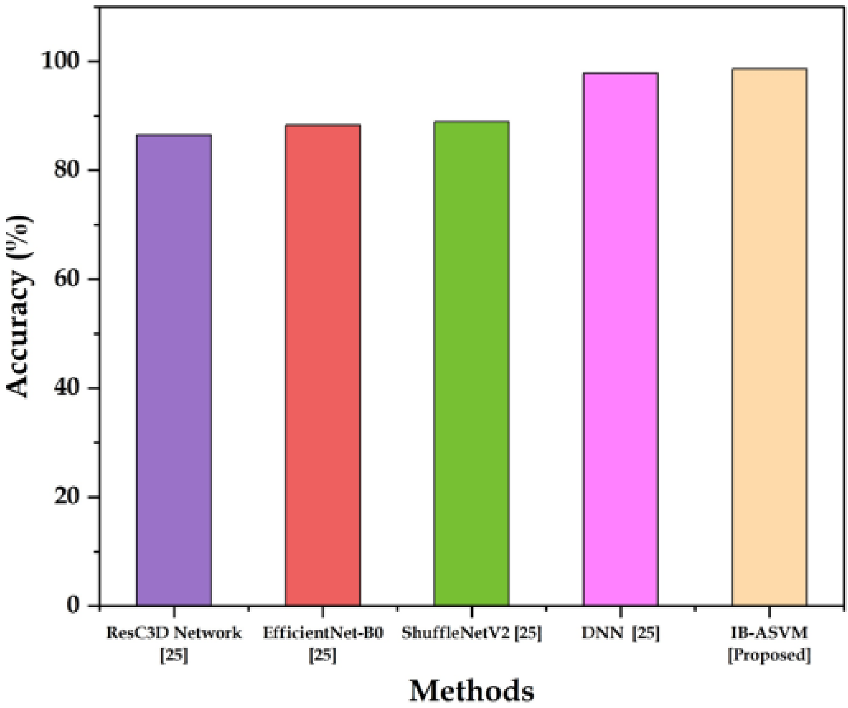

Figure 8. Comparison of accuracy.

5. Conclusion

To improve basketball player motion recognition and analysis, this study introduced a novel method called Intelligent Bayesian tuned-augmented Support Vector Machine (IB-ASVM). The model captures real-time data that is augmented by Kalman Filtering and Discrete Wavelet Transform (DWT) using biosensors and an ESP32-based smart wearable device. The features of basketball motion states including the shooting and dribbling motion are accurately classified by the developed IB-ASVM model applied on TensorFlow. To offer a very accurate analysis of movements, this approach enhances the development of players through training, as well as the reduction of casualties resulting from injuries, thereby proving to be a very useful improvement over simple observation of players in motion. The use of biomechanical sensors could adversely affect the performance of the players and their general acceptance due to the practicality and comfort when putting on the same. It may hinder some movements, or become uncomfortable to wear for the entire duration of training or a game. Additional studies could be performed to optimize the wearable sensors in size, location, and approach for reducing the impact on movement patterns, providing further comfort, and increasing acceptance among players. It remains to be noted that these constraints may be solved by new developments in lightweight materials and flexible designs for prolonged use during the games. By combining Bayesian optimization with sophisticated sensor data, the IB-ASVM model provides accurate and dependable motion detection. Basketball players' movements may be effectively tracked and evaluated thanks to its quick training and real-time performance analysis. Its thorough methodology improves decision-making in sports analytics and overall performance insights.

Author contributions: Writing—original draft preparation, LL and YP; writing—review and editing, LL and YP; visualization, LL and YP; supervision, LL and YP; project administration, LL and YP; funding acquisition, LL and YP. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Supported by the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJZD-K202302901); project name: Optimization of key technology of visual object detection based on complex moving image and its application in teaching and training. And the Chongqing Preschool Education College Scientific Research Platform in 2024. Project name: Digital Elderly Care Service Big Data Application Research Center (Grant Number: 2024KYPT-01).

Ethical approval: Not applicable.

Conflict of interest: The authors declare no conflict of interest.

References

1. Guo R, Chen B, Li Y, et al. Deep Learning Methods to Analyze the Forces and Torques in Joints Motion. Applied Sciences. 2024; 14(15): 6846. doi: 10.3390/app14156846
2. Li J, Zhang X, Yang G. The Biomechanical Analysis on the Tennis Batting Angle Selection Under Deep Learning. IEEE Access. 2023; 11: 97758–97768. doi: 10.1109/access.2023.3313167
3. Zare E, Mohammadi SM, Abbasi Kesbi R. A Portable Motion Sensor to Measure the Movements of Runners for Biomechanics Analysis. Journal of Bioengineering Research. 2020; 2(4): 12–22. doi: 10.22034/jbr.2020.253869.1036
4. Hendry D, Leadbetter R, McKee K, et al. An Exploration of Machine-Learning Estimation of Ground Reaction Force from Wearable Sensor Data. Sensors. 2020; 20(3): 740. doi: 10.3390/s20030740
5. Brumann C, Kukuk M, Reinsberger C. Evaluation of Open-Source and Pre-Trained Deep Convolutional Neural Networks Suitable for Player Detection and Motion Analysis in Squash. Sensors. 2021; 21(13): 4550. doi: 10.3390/s21134550
6. Wen L, Nie M, Chen P, et al. Wearable multimode sensor with a seamless integrated structure for recognition of different joint motion states with the assistance of a deep learning algorithm. Microsystems & Nanoengineering. 2022; 8(1): 24. doi: 10.1038/s41378-022-00358-2
7. Lu S, Zhang X, Wang J, et al. An IoT-Based Motion Tracking System for Next-Generation Foot-Related Sports Training and Talent Selection. Chen CH, ed. Journal of Healthcare Engineering. 2021; 2021: 1–14. doi: 10.1155/2021/9958256

8. Edriss S, Romagnoli C, Caprioli L, et al. The Role of Emergent Technologies in the Dynamic and Kinematic Assessment of Human Movement in Sport and Clinical Applications. Applied Sciences. 2024; 14(3): 1012. doi: 10.3390/app14031012
9. Tedesco S, Crowe C, Ryan A, et al. Motion Sensors-Based Machine Learning Approach for the Identification of Anterior Cruciate Ligament Gait Patterns in On-the-Field Activities in Rugby Players. Sensors. 2020; 20(11): 3029. doi: 10.3390/s20113029
10. Giles B, Kovalchik S, Reid M. A machine learning approach for automatic detection and classification of changes of direction from player tracking data in professional tennis. Journal of Sports Sciences. 2020; 38(1): 106–113. doi: 10.1080/02640414.2019.1684132
11. Li Z. Feature Extraction and Data Analysis of Basketball Motion Postures: Acquisition With an Inertial Sensor. Journal of Engineering and Science in Medical Diagnostics and Therapy. 2021; 4(4): 041006. doi: 10.1115/1.4052311
12. Hu X, Mo S, Qu X. Basketball Activity Classification Based on Upper Body Kinematics and Dynamic Time Warping. International Journal of Sports Medicine. 2020; 41(04): 255–263. doi: 10.1055/a-1065-2044
13. Chen Z, Zhang G. CNN sensor based motion capture system application in basketball training and injury prevention. Preventive Medicine. 2023; 174: 107644. doi: 10.1016/j.ypmed.2023.107644
14. Guo X, Brown E, Chan PPK, et al. Skill Level Classification in Basketball Free-Throws Using a Single Inertial Sensor. Applied Sciences. 2023; 13(9): 5401. doi: 10.3390/app13095401
15. Zhao Y, Wang X, Li J, et al. Using IoT Smart Basketball and Wristband Motion Data to Quantitatively Evaluate Action Indicators for Basketball Shooting. Advanced Intelligent Systems. 2023; 5(12). doi: 10.1002/aisy.202300239
16. Guembe IP, Lopez-Iturri P, Astrain JJ, et al. Basketball Player On-Body Biophysical and Environmental Parameter Monitoring Based on Wireless Sensor Network Integration. IEEE Access. 2021; 9: 27051–27066. doi: 10.1109/access.2021.3054990
17. Taborri J, Molinaro L, Santospagnuolo A, et al. A Machine-Learning Approach to Measure the Anterior Cruciate Ligament Injury Risk in Female Basketball Players. Sensors. 2021; 21(9): 3141. doi: 10.3390/s21093141
18. Fan J, Bi S, Xu R, et al. Hybrid lightweight Deep-learning model for Sensor-fusion basketball Shooting-posture recognition. Measurement. 2022; 189: 110595. doi: 10.1016/j.measurement.2021.110595
19. Tedesco S, Crowe C, Ryan A, et al. Motion Sensors-Based Machine Learning Approach for the Identification of Anterior Cruciate Ligament Gait Patterns in On-the-Field Activities in Rugby Players. Sensors. 2020; 20(11): 3029. doi: 10.3390/s20113029
20. Zhang L. Applying Deep Learning-Based Human Motion Recognition System in Sports Competition. Frontiers in Neurorobotics. 2022; 16. doi: 10.3389/fnbot.2022.860981
21. Johnson WR, Mian A, Robinson MA, et al. Multidimensional Ground Reaction Forces and Moments From Wearable Sensor Accelerations via Deep Learning. IEEE Transactions on Biomedical Engineering. 2021; 68(1): 289–297. doi: 10.1109/tbme.2020.3006158
22. Derungs A, Amft O. Estimating wearable motion sensor performance from personal biomechanical models and sensor data synthesis. Scientific Reports. 2020; 10(1): 11450. doi: 10.1038/s41598-020-68225-6
23. Lisca G, Prodaniuc C, Grauschopf T, et al. Less Is More: Learning Insights From a Single Motion Sensor for Accurate and Explainable Soccer Goalkeeper Kinematics. IEEE Sensors Journal. 2021; 21(18): 20375–20387. doi: 10.1109/jsen.2021.3094929
24. Xie Q, Jin N, Lu S. Lightweight Football Motion Recognition and Intensity Analysis Using Low-Cost Wearable Sensors. In: Applied Bionics and Biomechanics. Wiley; 2023.
25. Xiao J, Tian W, Ding L. Basketball Action Recognition Method of Deep Neural Network Based on Dynamic Residual Attention Mechanism. Information. 2022; 14(1): 13. doi: 10.3390/info14010013