

A REVIEW OF SMART TECHNOLOGIES IN NET SPORTS

Karthigai Lakshmi S¹, Raja Rajeshwari G², Thirisha T³, Vasuki K⁴, Thaarani S⁵

Professor and Head, Department of Electronics and Communication Engineering, SSM Institute of Engineering and Technology, Dindigul, India¹

Assistant Professor, Department of Electronics and Communication Engineering, SSM Institute of Engineering and Technology, Dindigul, India²

UG Students, Department of Electronics and Communication Engineering, SSM Institute of Engineering and Technology, Dindigul, India³⁻⁵

Abstract: Net based sports are a fast-paced sport where the ball or shuttle moves at very high speed, making manual analysis and player performance analysis difficult and often inaccurate. Traditional methods rely on human observation, which may not capture precise details during high-speed gameplay and player Movements . With the advancement of sensor technology and Artificial Intelligence, it is now possible to automatically monitor physical movements during sports activities. Such technologies help in improving performance analysis, and supporting better training methods for players [6].

Keywords: Net Sports, Performance Analysis, Real time Monitoring , Portability

I. LITERATURE REVIEW

1) Shuttlecock Detection Using Residual U-Net

Research Objective:

The main objective of this study is to develop an accurate shuttlecock detection system for badminton matches using deep learning techniques. Detecting the shuttlecock is difficult because it moves very fast, appears very small in video frames, and is affected by complex backgrounds and lighting conditions. To address this problem, the researchers proposed a Residual U-Net architecture that combines U-Net with residual connections to improve feature learning and detection accuracy[1]. The model processes video frames from badminton matches and learns to identify the shuttlecock even in challenging conditions. This approach improves detection performance and helps in automatic shuttlecock tracking and match analysis.

Residual U-Net Architecture:

Encoder:

The encoder extracts important features from the input image by applying multiple convolution layers and pooling operations.

Decoder:

The decoder reconstructs the image features and identifies the location of the shuttlecock in the frame. Residual connections are added between layers to improve the training process[1]. These connections help the network retain important feature information and avoid problems such as gradient vanishing.

Shuttlecock Detection Process:

The proposed system follows several steps for shuttlecock detection. First, video frames are extracted from badminton match recordings and preprocessed using techniques such as resizing, normalization, and noise reduction to improve image quality. Next, the preprocessed frames are given as input to the Residual U-Net model, which analyzes important spatial features and identifies the shuttlecock in each frame. The encoder extracts useful features while the decoder helps

locate the shuttlecock position. Finally, the system outputs the detected shuttlecock location with good accuracy, even when the shuttlecock appears blurred or partially hidden[1].

Advantages of the Proposed Method:

The residual U-Net model offers several advantages for shuttlecock detection. It improves feature extraction capability, allowing the model to detect small objects like shuttlecocks more accurately[1]. The residual connections help improve the stability of the training process and reduce information loss during deep network training[1].

The system also performs well in complex backgrounds, making it suitable for real badminton match environments.

Equation	Purpose
$y=F(x,W_i)+xy = F(x,W_{-i}) + xy=F(x,W_i)+x[1]$	Residual learning in neural network[1]
$\text{Precision}=\frac{TP}{TP+FP}$ $\text{Precision}=\frac{TP}{TP+FP}[1]$	Detection accuracy measurement[1]
$F1=2PRP+RF1 = \frac{2PR}{P+R}$ $F1=P+R2PR[1]$	Balanced performance evaluation[1]

2) AUTOMATED HIT-FRAME DETECTION FOR BADMINTON MATCH ANALYSIS

Research Objective:

The goal of this research is to automatically detect hit frames during badminton matches. A hit frame refers to the exact moment when the racket makes contact with the shuttlecock[2]. Identifying this moment is important for analyzing player performance and rally sequences. Detecting hit frames manually from match videos is difficult and time-consuming because the shuttlecock moves very fast and the contact moment occurs within a very short time. Therefore, automated methods are used to improve accuracy and reduce human effort. In this research, video analysis techniques are applied to examine frame-by-frame changes in the shuttlecock’s motion and the interaction between the racket and the shuttlecock. By analyzing these visual changes, the system can identify the exact frame where the hit occurs. This information can be used for further analysis such as shot classification, rally analysis, and performance evaluation of players during badminton matches.

Hit-Frame Detection Concept:

In badminton gameplay, when a player hits the shuttlecock, its direction and speed suddenly change. This sudden variation in motion can be used to detect hit events. The proposed system analyzes the shuttlecock trajectory across multiple video frames to observe its movement pattern during play[2]. By continuously tracking the shuttlecock position, the system compares its motion between consecutive frames. When a significant change in speed or direction is detected, the system identifies that moment as a possible hit event. This method helps in locating the exact frame where the racket makes contact with the shuttlecock. Such detection can support further analysis of rally sequences, shot timing, and overall match performance.

Convolution Operation (CNN Feature Extraction):

$$H_{conv}(l) = \sum_{c=1}^C \text{Out}(l-1) \sum_{i=1}^I \sum_{j=1}^J H_{out,c,i,j}(l-1) W_{c,i,j}(l) + b(l) H^{\{l\}}_{\{conv\}} = \sum_{c=1}^C \{C^{\{l-1\}}_{out}\} \sum_{i=1}^I \sum_{j=1}^J H^{\{l-1\}}_{out,c,i,j} W^{\{l\}}_{c,i,j} + b^{\{l\}} H_{conv}(l) = c=1 \sum C_{out}(l-1) i=1 \sum 3 j=1 \sum 3 H_{out,c,i,j}(l-1) W_{c,i,j}(l) + b(l) \dots [2]$$

Purpose

- Extracts spatial features from video frames.
- Used in the **SA-CNN model for shot angle recognition**.

ReLU Activation Function:

$$H_{out}^{(l)} = \text{ReLU}(H_{BN}^{(l)}) = \max\{0, H_{BN}^{(l)}\} \quad H_{out}^{(l)} = \text{ReLU}(H_{BN}^{(l)}) = \max(0, H_{BN}^{(l)})$$

Purpose

- Introduces non-linearity in the CNN.
- Helps the neural network learn complex patterns.

Cross-Entropy Loss Function (Main Training Equation):

$$L = -\log \left(\frac{e^{\hat{y}_{n,c}}}{\sum_{c=1}^C e^{\hat{y}_{n,c}}} \right) \quad L = -\log \left(\sum_{c=1}^C e^{\hat{y}_{n,c}} \right)$$

Total loss:

$$L = \sum_{n=1}^N L_n = \sum_{n=1}^N \left(-\log \left(\frac{e^{\hat{y}_{n,c}}}{\sum_{c=1}^C e^{\hat{y}_{n,c}}} \right) \right) \dots [2]$$

Purpose

- Measures difference between predicted class and true class.
- Used to **train the SA-CNN model**.

Transformer Prediction Equation:

$$y^t = \text{Transformer}(X_t) \quad y^t = \text{Transformer}(X_t)$$

Where

$$X_t = (x_{1t}, x_{2t}, \dots, x_{Nt}) \quad X_t = (x_{1t}, x_{2t}, \dots, x_{Nt}) \dots [2]$$

Purpose

- Predicts **shuttlecock flying direction sequence** using player keypoints.

Temporal Video Analysis:

The method uses temporal analysis techniques to study the relationship between consecutive video frames[2]. Instead of analyzing each frame independently, the system observes how the shuttlecock moves over time across multiple frames. By examining changes in shuttlecock velocity, trajectory direction, and player movement, the system can detect sudden motion variations that occur when the racket hits the shuttlecock. These changes help the system determine the exact frame where the hit occurs. Compared to simple frame-by-frame analysis, this temporal approach provides better accuracy and reliability in identifying hit events during badminton matches.

Applications:

The automated hit-frame detection system can be used in several applications such as match statistics generation, coaching analysis, and performance evaluation. By automatically identifying the exact moment of shuttlecock contact, the system helps analysts understand the sequence of shots during a rally. It also provides useful information about shot timing,

frequency of hits, and player response during gameplay. This data can support coaches in studying player strategies, improving training methods, and evaluating overall performance more effectively[2].

3) **BADMINTON HAWKEYE CHALLENGE A LIGHTWEIGHT BADMINTON**

Research Objective:

This research focuses on developing a lightweight neural network capable of detecting shuttlecocks and player actions in real time. Traditional deep learning models require high computational power and may not be suitable for real-time sports analysis systems.

Feature Pyramid Network:

The proposed detection system uses a feature pyramid network to detect objects at multiple scales[3].

Shuttlecocks are extremely small compared to other objects in the scene[3]. The feature pyramid structure allows the network to analyze features at different resolutions, improving the detection of small objects.

Lightweight Architecture:

The model is designed to be computationally efficient by reducing the number of parameters and layers while still maintaining good detection accuracy[3]. This lightweight architecture decreases the computational complexity and memory usage of the system. As a result, the model can process video frames faster and perform detection in a shorter time. This design allows the system to run on devices with limited computational resources, such as embedded systems or low-power hardware, making it suitable for real-time sports analysis and practical applications.

Final Combined Loss Function :

$$\text{Loss} = R, \text{Loss}_{\{\text{NWD}\}} + (1-R), \text{Loss}_{\{\text{IoU}\}} \dots [3]$$

Purpose

- Combines **Normalized Wasserstein Distance (NWD) loss** and **IoU loss**.
- Improves **small object detection accuracy** such as a badminton shuttlecock.

Wasserstein Distance Equation:

$$W_2(A, B) = \|\mu_A - \mu_B\|_2 + \|\Sigma_A^{1/2} - \Sigma_B^{1/2}\|_F \quad W_2^2(A, B) = \|\mu_A - \mu_B\|_2^2 + \|\Sigma_A^{1/2} - \Sigma_B^{1/2}\|_F^2 \quad W_2(A, B) = \|\mu_A - \mu_B\|_2 + \|\Sigma_A^{1/2} - \Sigma_B^{1/2}\|_F \dots [3]$$

Purpose

- Measures the **distance between predicted and actual bounding box distributions**.
- Helps the model detect **tiny objects more accurately**.

Frame Difference Motion Detection:

$$D_n(x, y) = |f_n(x, y) - f_{n-1}(x, y)| \quad D_n(x, y) = |f_n(x, y) - f_{n-1}(x, y)| \dots [3]$$

Purpose

- Detects **movement between two video frames**.
- Used to identify the **shuttlecock motion in the video**.

Real-Time Detection Capability:

Because of its efficient design, the proposed model can perform real-time detection during badminton matches. This allows the system to analyze video frames quickly and identify important events without significant delay. Real-time detection makes the system useful in several practical applications. In sports broadcasting, it can help highlight key moments such as important shots and rally sequences. In automated refereeing systems, it can assist in identifying contact events and supporting decision-making. Additionally, in training analysis platforms, the system can provide instant feedback to players and coaches by analyzing shot timing and gameplay patterns, which helps in improving performance and strategy development[3].

4) Monocular Camera-Based Shuttlecock Tracking**Research Objective:**

The purpose of this study is to track shuttlecock movement and detect hit frames using a monocular camera system. Unlike multi-camera systems, this approach uses only a single camera to monitor the badminton court and capture the entire gameplay[4]. The main objective is to develop a reliable and efficient method for tracking the shuttlecock and identifying important events during a match. Using a single camera reduces the cost and complexity of the system compared to multi-camera setups[4]. However, tracking the shuttlecock with one camera is challenging because the shuttlecock moves very fast and appears small in video frames. To address these challenges, image processing and tracking techniques are applied to detect the shuttlecock and follow its movement across consecutive frames.

Camera-Based Tracking Method:

The system captures video frames from a monocular camera placed near the badminton court and processes them using computer vision algorithms. Each frame is analyzed to detect the shuttlecock using image processing techniques such as object detection and feature extraction. These techniques help distinguish the shuttlecock from the background and other moving objects in the scene.

Once the shuttlecock is detected, the system tracks its position across consecutive frames. By comparing the shuttlecock's location in successive frames, the system calculates its trajectory and movement pattern. This trajectory information helps in understanding the direction and speed of the shuttlecock during gameplay and supports further analysis such as hit detection and rally tracking[4].

Hit Detection Mechanism:

The algorithm monitors the shuttlecock's movement across consecutive video frames and compares its speed and direction over time. When a significant change in these motion patterns is observed, the system identifies that moment as a hit event. By locating the frame where this sudden change occurs, the system can determine the exact moment of contact between the racket and the shuttlecock. This method helps in accurately detecting hit frames and supports further analysis of rally sequences and player performance.

Temporal Intersection over Union:

$$t\text{-IoU} = \frac{HM \cap GT}{HM \cup GT} \quad t\text{-IoU} = \frac{HM \cap GT}{HM \cup GT} \dots [4]$$

Precision:

$$\text{Precision} = \frac{\text{True Detections}}{\text{All Detections}} \quad \text{Precision} = \frac{\text{True Detections}}{\text{All Detections}} \dots [4]$$

Advantages of Monocular Systems:

Using a single camera simplifies the hardware setup and reduces system cost. It also makes the system easier to install in training environments.

Despite its simplicity, the system can still provide reliable tracking and hit detection performance.

5) AI Based Badminton Smash Detection Using Residual Shuffle Network

Research Objective:

The goal of this research is to automatically classify different badminton shots using deep learning techniques. Shot classification helps in analyzing player strategies, understanding gameplay patterns, and evaluating performance during matches. Different types of shots such as smash, drop, clear, and drive have unique motion patterns that can be identified from video data. In this approach, a deep learning model called the Residual-Shuffle Network is used to analyze video frames and extract important features related to player movement and shuttlecock motion[5]. The network learns these patterns during training and then classifies the type of shot performed by the player. This automated classification system helps coaches, analysts, and players study match strategies more effectively and improve training methods.

Residual-Shuffle Network:

The proposed model uses a Residual-Shuffle Network architecture for badminton shot classification. In this model, residual connections are used to improve feature learning by allowing information to pass directly between layers, which helps the network learn deeper and more meaningful features. At the same time, shuffle operations are applied to reorganize feature channels, which reduces computational complexity and improves the efficiency of the model[5]. By combining residual learning with channel shuffle operations, the network can process visual information more effectively while using fewer computational resources. This architecture enables the model to capture both spatial features and motion patterns from video frames, helping it accurately identify and classify different types of badminton shots.

Gaussian Filter Equation:

Used for **image smoothing and noise reduction** before detection.

$$G(a,b)=\frac{1}{2\pi\sigma^2}e^{-\frac{a^2+b^2}{2\sigma^2}} \quad G(a,b)=2\pi\sigma^2 e^{-2\sigma^2 a^2+b^2} \dots [5]$$

Where:

- a, b, a, b = pixel distance from center
- σ = standard deviation of Gaussian distribution

Image Convolution Equation:

Used to apply the **Gaussian filter to the image**.

$$I_{i,j}' = \sum_{u=-k}^k \sum_{v=-l}^l I_{i+u,j+v} \cdot G_{u,v} \quad I_{i,j}' = \sum_{u=-k}^k \sum_{v=-l}^l I_{i+u,j+v} \cdot G_{u,v} \dots [5]$$

Where:

- $I_{i+u,j+v}$ = input pixel value
- $G_{u,v}$ = Gaussian kernel value

Population Initialization Equation:

Used in the **Upgraded Pufferfish Optimizer** to initialize solutions.

$$y_{i,d} = lb_d + e(ub_d - lb_d) \quad y_{i,d} = lb_d + e(ub_d - lb_d) \dots [5]$$

Where:

- l_{dblb_dlbd} = lower bound
- u_{dbub_dubd} = upper bound
- eee = random value between 0 and 1

Mean Squared Error (MSE):

$$l(y_i, \hat{y}_i) = \frac{1}{2} (y_i - \hat{y}_i)^2$$

Used to **measure prediction error during training.**

Optimized Training with Pufferfish Optimizer:

The researchers used an upgraded Pufferfish Optimizer to improve the training process of the neural network. An optimizer plays an important role in adjusting the model parameters during training so that the network can learn the correct patterns from the data. In this method, the Pufferfish Optimizer helps the model update its weights more efficiently by searching for the best solution during the learning process. This improves the stability of training and reduces the chances of the model getting stuck in poor solutions. As a result, the network can achieve faster convergence and better classification accuracy when identifying different badminton shots[5].

Shot Classification Performance:

The system can classify various badminton shots such as smashes, clears, and drop shots by analyzing patterns in player movement and shuttlecock motion. During the training process, the model learns the unique characteristics of each type of shot from the video data. After training, the system can automatically identify and categorize these shots during match

Shot Classification Performance:

The system can classify various badminton shots such as smashes, clears, and drop shots by analyzing patterns in player movement and shuttlecock motion. During the training process, the model learns the unique characteristics of each type of shot from the video data. After training, the system can automatically identify and categorize these shots during match analysis. Experimental results show that the model achieves high classification accuracy and performs effectively in real badminton match scenarios. The system is capable of recognizing different shot types even when the gameplay is fast and complex. This performance demonstrates that the proposed approach is reliable for badminton match analysis and can support coaching, performance evaluation, and strategy analysis[5].

6) IoT-BASED AUTOMATED SPORTS DETECTION SYSTEM

Research Objective:

This research focuses on developing an IoT-based sports monitoring system using sensors and embedded hardware to automatically detect important events during sports activities. The system integrates sensors, microcontrollers, and communication modules to collect and process data in real time[6]. Although the proposed system is mainly designed for cricket stumping detection. This approach helps automate sports event detection and supports performance monitoring and training analysis.

Formula	Purpose
$(A = \sqrt{A_x^2 + A_y^2 + A_z^2})$ [6]	Detect stump disturbance
$(D > D_t)$ [6]	Batter outside crease
$(D \leq D_t)$ [6]	Batter inside crease
$(A > A_t \ \& \ D > D_t)$ [6]	Final decision: OUT

System Architecture:

The system detects Stumping in cricket using sensor Technology[6]. It senses the stump disturbance when the wicketkeeper removes the bails, while a Time-of-Flight Sensor measures the batter's position relative to the crease. When disturbance detected, the system checks the distance. If the batter is outside the crease, it declares as "OUT", otherwise, "NOT OUT" by LED indicators[6].

Advantages of IoT-Based Systems:

IoT-based sports monitoring systems are cost-effective and easy to deploy. They can provide real-time event detection and remote monitoring capabilities. Such systems can also be integrated with computer vision techniques to create hybrid sports analytics platforms[6].

- a. **Affordable & Scalable**
- b. **Live Data & Instant Alerts**
- c. **Training Without Borders (Remote)**
- d. **Precision Officiating**

II. CONCLUSION

A new setup uses sensors and AI technologies to track how hard and fast players hit the shuttlecock or ball during matches. One step beyond today's setup might be linking sensors to online logs that store workout details automatically and for players movement analysis. Instead of just showing real-time stats, information could flow into digital platforms where patterns emerge over weeks. Remote feedback may come easier when coaches access updated records from anywhere. Small upgrades over time might turn basic tools into full training helpers.

Table 1: Comparison Based on System Components

Reference	Sensors	Hardware Platform	Software / Framework	Communication / Processing
Ref 1 (2023)	Camera	Embedded / Edge devices	CNN, MobileNetV3	Lightweight edge processing
Ref 2 (2023)	Camera	Deep learning processing system	SA-CNN, R-CNN, Transformer	Sequential video analysis
Ref 3 (2024)	Camera	AI processing unit	TrackNet, YOLOv7, DensePose	Video analysis pipeline
Ref 4 (2024)	Camera	High-performance computing system	PyTorch	Neural network training
Ref 5 (2025)	Camera	GPU-based computing system	Deep Learning framework	Video frame processing
Ref 6 (2025)	ToF sensor, Accelerometer	Arduino UNO	Embedded C	ESP8266 Wi-Fi communication

Source: Compiled from [1][2][3][4][5][6]

Table 2: Comparison Based on Methodology

Reference	Detection Approach	Core Algorithm	Processing Technique	Analysis Method
Ref 1 (2023)	Lightweight detection	MobileNetV3 CNN	Feature pyramid detection	Embedded object detection
Ref 2 (2023)	Tactical match analysis	SA-CNN + Transformer	Rally trimming	Stroke recognition
Ref 3 (2024)	Multi-model analysis	TrackNet + YOLOv7	Pose estimation	Shot and player analysis
Ref 4 (2024)	Shuttlecock tracking	Residual U-Net	Deep learning segmentation	Object detection
Ref 5 (2025)	Video-based detection	ResNet with Residual-Shuffle	Frame classification	Smash recognition
Ref 6 (2025)	Sensor-based detection	Motion threshold analysis	Microcontroller processing	Real-time impact detection

Source: Compiled from [1][2][3][4][5][6]

Table 3: Comparison Based on Dataset / Input

Reference	Input Type	Dataset Size	Resolution / Format	Training Requirement
Ref 1 (2023)	Image dataset	Shuttlecock dataset	Image frames	CNN model training
Ref 2 (2023)	Match videos	Professional game dataset	Frame sequences	Multi-stage training
Ref 3 (2024)	Video	Match recordings	Single-camera video	Model training
Ref 4 (2024)	Video frames	55,000 images	512 × 288 resolution	Deep learning training
Ref 5 (2025)	Video frames	Badminton match videos	Frame sequences	Large dataset required
Ref 6 (2025)	Sensor data	Real-time collected data	Motion signals	Minimal training

Source: Compiled from [1][2][3][4][5][6]

Table 4: Comparison Based on Performance Analysis

Reference	Accuracy	Precision	Recall	Response Time
Ref 1 (2023)	82.8%	82.8%	Moderate	Fast
Ref 2 (2023)	96–99% depending on task	Very High	High	Moderate
Ref 3 (2024)	89.7%	High	91.3%	Moderate
Ref 4 (2024)	~94%	High	High	Moderate
Ref 5 (2025)	~95%	High	High	Moderate
Ref 6 (2025)	~95%	High	High	<0.5 seconds

Source: Compiled from [1][2][3][4][5][6]

Table 5: Comparison Based on Applications

Reference	Sports Analysis	Coaching Support	Real-Time Monitoring	Embedded System Support
Ref 1 (2023)	Yes	Limited	Yes	Yes
Ref 2 (2023)	Yes	Yes	Limited	No
Ref 3 (2024)	Yes	Yes	Limited	No
Ref 4 (2024)	Yes	Yes	No	No
Ref 5 (2025)	Yes	Yes	No	No
Ref 6 (2025)	Yes	Limited	Yes	Yes

Source: Compiled from [1][2][3][4][5][6]

Table 6: Comparison Based on Advantages and Limitations

Source: Compiled from [1][2][3][4][5][6]

Reference	Advantages	Limitations	Complexity Level	System Cost
Ref 1 (2023)	Lightweight model for mobile devices	Lower detection accuracy	Low	Low
Ref 2 (2023)	Detailed tactical insights	Complex architecture	Very High	High
Ref 3 (2024)	Multi-feature match analysis	Slightly lower accuracy	Medium	Medium
Ref 4 (2024)	Robust shuttlecock detection	Large dataset required	High	High
Ref 5 (2025)	High smash detection accuracy	High computation requirement	High	High
Ref 6 (2025)	Fast response and reliable detection	Requires physical sensors	Low	Low

III.FUTURE WORK

The proposed system can enhance its performance and usability. Future work may include integrating wireless communication and mobile applications. The use of cloud-based storage can enable long-term data analysis and performance tracking. Additionally, machine learning techniques can be applied to classify different shot types and analyze player behavior. Furthermore, future research can explore the integration of advanced data analytics and intelligent coaching support systems. By analyzing collected data over multiple training sessions, the system could provide personalized feedback and performance recommendations to players. This would assist coaches in identifying strengths and weaknesses, optimizing training strategies, and improving overall gameplay performance.

REFERENCES

- [1]. M. Abdul Haq, S. Tarashima, and N. Tagawa, "Shuttlecock detection using residual U-Net for fast-moving objects in complex backgrounds," *Journal of Sports Analytics*, 2024.
- [2]. Y.-H. Chien and F. Yu, "Automated hit-frame detection for badminton match analysis," *arXiv preprint arXiv:2307.16000v2*, 2023.
- [3]. H. Jin, Z. Qiang, J. Yan, X. Zhang, and C. Wang, "Lightweight badminton detection network with feature pyramid for fast shuttlecock and player action detection," *IEEE Access*, 2023.
- [4]. C.Hsuetal., "Monocularcamera-based shuttlecock tracking and hit-frame detection for player performance assessment," *Journal of Sports Technology*, 2024.
- [5]. X. Peng and L. Zheng, "Residual-shuffle network optimized with upgraded pufferfish optimizer for badminton shot classification," *Neural Computing and Applications*, 2025.
- [6]. S.A.Saifetal., "IoST: IoT-based automated stumping detection system using sensors and ESP8266," *International Journal of Sports Engineering and Technology*, 2025