Proceedings of the 5th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/108/2025.21197

Movement Detection of Tennis Players Based on Yolo and Human Skeleton Recognition Technology

Mengle He[1,a,*,†], Mingyu Fu[2,b,†], Deyang Cao[3,c,†]

[1]Maynooth International Engineering College, Fuzhou University,
[2]School of Beijing Foreign Study, University International Curriculum Center,
[3]School of Beijing Foreign Study, University International Curriculum Center
a. 3261994752@qq.com, b. CaspianFu@outlook.com, c. caodeyang2025@outlook.com
**corresponding author*
[†]These authors contributed to the work equally and should be regarded as co-first authors

Abstract: This study aims to develop a monitoring model for the action states of athletes (such as standing, moving, and striking) during tennis matches. The model is based on the YOLO (You Only Look Once) architecture and is trained with the 3dResNet50 to achieve automatic recognition of athletes' actions. Additionally, we utilized the MediaPipe model to recognize the human skeletal structure, further enhancing the accuracy of action recognition. Tested on real tennis match video data, the model demonstrated efficient action recognition capabilities and good real-time performance, providing strong technical support for sports training and competition analysis. This research not only extends the application of computer vision in the field of sports but also lays a foundation for further advancements in motion analysis technology.

Keywords: Behavior Recognition, YOLO, 3DResNet50, Media-pipe, STGCN model, Frame Compare

1. Introduction

In modern sports science, accurately monitoring and analyzing the action states of athletes is crucial for enhancing training outcomes and competitive performance. This is particularly significant in tennis, where the player's actions, such as standing, moving, and striking, directly affect the outcome of the match. Traditional methods of action analysis, which rely on manual observation and recording, are time-consuming and susceptible to subjective biases. Thus, developing an efficient and accurate automated action recognition system has become a key direction in sports science research.

In recent years, deep learning technology has made significant advances in the field of computer vision, especially with object detection algorithms like YOLO (You Only Look Once), which have opened new possibilities for real-time action recognition. YOLO achieves rapid and precise object detection by predicting the location and category of targets in a single pass. However, relying solely on YOLO for action recognition presents certain limitations when dealing with complex movements and subtle differences.

Early research in the field of action recognition exhibited several shortcomings, including limited diversity and low annotation quality of datasets, an over-reliance on handcrafted features with high computational complexity, and a lack of real-time applicability and suitability for real-world

DOI: 10.54254/2755-2721/108/2025.21197

scenarios. Additionally, these studies often struggled to accurately handle complex movements and athlete interactions, and they lacked interdisciplinary approaches that integrate knowledge from related fields. Furthermore, many of these studies did not undergo large-scale validation and overlooked the needs of actual users, which constrained their effectiveness and generalizability in practical applications.

To date, deep learning methods, particularly those involving computer vision techniques such as YOLO and ResNet, have been widely applied to various sports for tasks like player tracking, action recognition, and performance analysis. These methods have proven effective in sports like soccer, basketball, and athletics, where they are used to monitor player movements, analyze strategies, and even assist in real-time decision-making during games. By automatically detecting and classifying actions, these techniques have significantly improved the accuracy and efficiency of sports analysis, providing valuable insights for coaches and analysts.

Despite the success of these methods in other sports, they have not yet been extensively applied to the analysis of tennis matches. The unique dynamics of tennis, including the fast-paced exchanges, frequent player movements, and the involvement of both the player and the ball, present distinct challenges that have yet to be fully addressed by current deep learning techniques. This gap indicates a significant opportunity for research and development in applying these advanced methods to tennis, potentially leading to new tools and insights that could revolutionize how the sport is analyzed and understood.

The objective of this research is to develop a system capable of automatically identifying and classifying the action states of athletes during tennis matches. We begin with preliminary target detection using the YOLO model, followed by training convolutional neural networks to classify action states. Additionally, the MediaPipe model is incorporated to precisely locate human skeletal structures, thus enhancing the overall precision of recognition. The model was tested on actual tennis match video data, and the results demonstrated excellent accuracy and real-time performance.

This research not only provides new technological means for the analysis of tennis actions but also explores new directions for the application of deep learning in the sports field. The subsequent sections of this paper will detail our research methods, experimental design, results analysis, and conclusions.

2. Related Work

YOLO (You Only Look Once) is a popular real-time object detection system developed by Joseph Redmon and others[1]. Its core idea is to use a single neural network to predict object bounding boxes and class probabilities, achieving fast and accurate object detection in images. YOLO is characterized by its speed and ability to process video streams in real-time, making it highly suitable for applications that require immediate response, such as surveillance and autonomous driving.

Regarding the research of YOLO in the field of sports science and game monitoring, several papers have demonstrated its application results in motion recognition and motion tracking. For example, one study used YOLO and Deep SORT technology to improve accuracy in soccer multi-detection tasks, achieving 95 percent ball detection accuracy through a semi-supervised learning system, which is significantly better than previous methods [2]. In addition, for basketball games, the researchers improved the ability to track and detect basketball games by integrating multi-source motion features and hybrid YOLO-T2LSTM networks [3]. Moreover, a study in the sport of squash evaluated multiple open sources, pre-trained deep convolutional neural networks suitable for detecting athlete movements from single-camera video to help coaches and players optimize training and competition strategies. These studies show that YOLO can not only track athletes and objects in real-time but also support the assessment of tactical placement and physical status by analyzing the dynamic position and movement patterns of athletes.

DOI: 10.54254/2755-2721/108/2025.21197

If we want to achieve situational awareness of the entire tennis game, player behavior detection is quite an important problem that should be solved. In most cases, we conclude that players have three behaviors, which are moving, standing, and hitting. We can do some work base on the player's behavior. Two methods came out. The first is to detect the player's motion straightforward base on computer vision, and we concluded that the training dataset is a segment of the video. The second method is detecting the behavior base player's pose, which training datasets are the details of the pose, including relative angle, absolute position vector, etc.

Obviously, we cannot just display the player's status straightly on screen, so Yolov5 has been introduced, first bracket the player, and write the state near the bracket.

Mediapipe is a cross-platform open-source framework widely used for building pipelines for multimodal data, such as visual and audio data. For tennis player action recognition, Mediapipe can be used to capture the skeletal key points of the players (i.e., pose estimation). The pose estimation module of Mediapipe can detect the key points of the player in each frame of the video, including joints, limb positions, and more. These key points are then extracted to form a time-series data sequence, serving as input for further processing.

3. Methodology

3.1. Main steps

The combination of YOLO and skeletal recognition is typically achieved through the following steps: First, the YOLO model is used to detect and locate athletes in the video, generating bounding boxes to separate the athletes from the background. Then, the image regions containing the athletes are cropped and fed into a skeletal recognition model, such as Mediapipe or a ResNet-based pose estimation model, to identify the key joints of the athletes. Finally, the results of skeletal recognition are combined with YOLO's detection outputs to achieve more accurate action classification, such as standing, moving, or hitting. This approach effectively combines YOLO's [1] object detection capabilities with skeletal recognition for action analysis, providing a powerful tool for automating the recognition and analysis of complex movements. These time-series data (sequential graph structures) are input into the ST-GCN model, where ST-GCN uses spatiotemporal convolution operations to extract spatial and temporal features. The extracted spatiotemporal features are then fed into a classifier, typically a fully connected network, to perform action classification. Ultimately, the system can identify specific actions being performed by the tennis player, such as swinging or running

These time-series data (sequential graph structures) are input into the ST-GCN model, where ST-GCN uses spatiotemporal convolution operations to extract spatial and temporal features. The extracted spatiotemporal features are then fed into a classifier, typically a fully connected network, to perform action classification. Ultimately, the system can identify specific actions being performed by the tennis player, such as swinging or running

3.2. Data Preparing

For Resnet50, First, take any tennis match video data online before cutting off non-related parts and only save the downside of each data. Make sure the video is 30 frames per second. Next, make a category of video (1 and 0 represent hitting or other, respectively), convert it from video to picture by using cv2, and group it into 2 different files. Finally, use Yolo to extract the player into two 1(hitting) and 0(other) directories. To overcome oversampling, take one sample frame by 10 frames. For 3dResenet50, it is mostly the same, but the training dataset should be 5-second video; use a video that is 60 frames per second that can help 3dResent50 "connect" each frame more accurately while the processing speeds may reduced. For the third method, by using the STGCN model, it should first use Mediapipe to convert labeled data into skeleton parameter files (.npy). The skeleton parameter

DOI: 10.54254/2755-2721/108/2025.21197

includes the relative angle between arms and body and the joint vector of shoulder, hand, leg, and barycenter. However, it is important to collect at least 1080p resolution of video data to train the opponent player, or the 3dResnet50 model may be divergent, as the camera angle is fixed, and the opponent player occupies a small number of pixels. Before sending data into different models, data enhancement is needed, especially on opponent training. Data enhancement can improve the universality of a model; also, because of the lack of training data, data enhancement can increase the amount of training data. Meanwhile, without data enhancement, the model cannot well recognize the opponent player's status.

3.3. Train Process

The whole periods start with after the data enhancement. We are aiming to make the name of the directory become the label 1 hitting and 0 other, but there are some uncertainties in single frames, as when cutting off some of the frames, human reaction time should be calculated. So, remove the front and back frames before putting them into the model. We applied a data enhancement method for Resent50, which is resize to height 224 and width 224. For three color channels, we normalize it as a mean equal to 0.485, 0.456, and 0.406, with standard deviations of 0.229, 0.224, and 0.225, respectively.[4]

3.4. Resnet50

The whole periods start with after the data enhancement. We are aiming to make the name of the directory become the label 1 hitting and 0 other, but there are some uncertainties in single frames, as when cutting off some of the frames, human reaction time should be calculated. So, remove the front and back frames before putting them into the model. We applied a data enhancement method for Resent50, which is resize to height 224 and width 224. For three color channels, we normalize it as a mean equal to 0.485, 0.456, 0.406, standard deviation 0.229, 0.224, and 0.225, respectively, and use a confusion matrix to generate the result [5]. Figure 1 shows how data has been processed by using resnet50 methods.

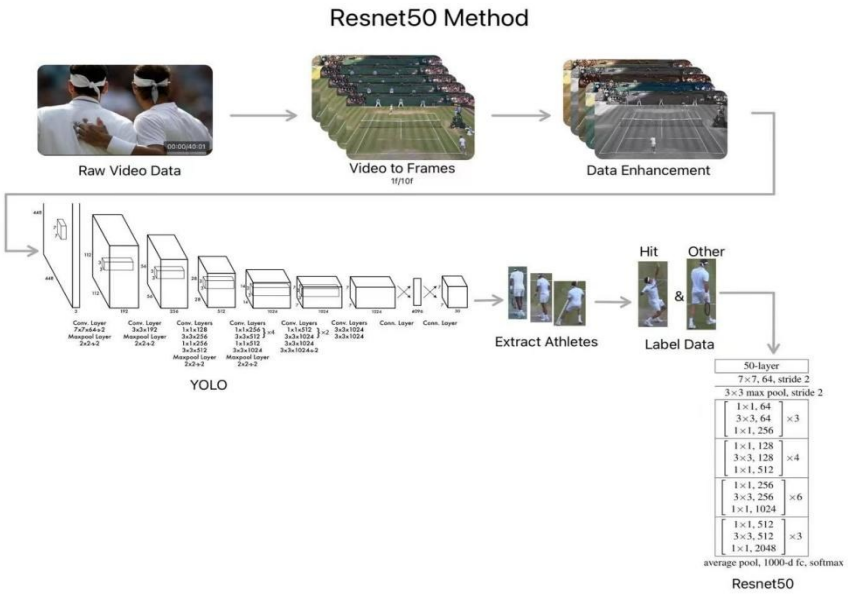

Figure 1: Shows the process of resnet50 methods to detect player

DOI: 10.54254/2755-2721/108/2025.21197

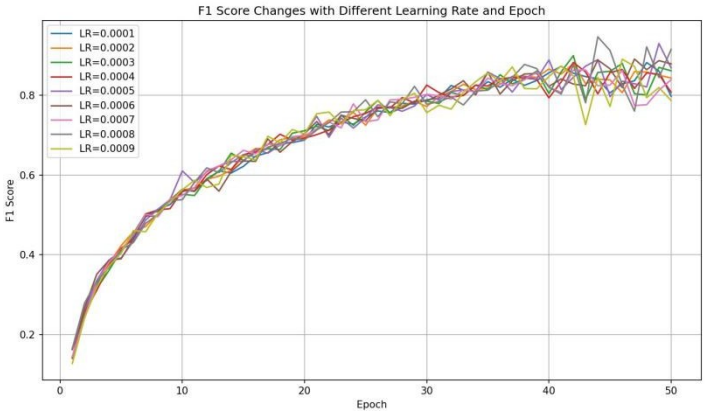

Figure 2: Shows the relationship between F1 score and number of epoch

3.5. 3dResnet50

Use download 3dResnet50 model [6] and change the last full connection layer to binary output. Use method of data enhancement to increase the model universality (especially on color enhancement). Use SGD optimizer [7] function and cross entropy loss function. SGD optimizer function will first random pick a data from training datasets and calculate the F1 Score Changes with different learning rate and epoch gradient of that data, then it changes the parameter by reference to the gradient and learning rate, until convergent. As figure 2 shows the relationship between f1 score and epoch number to help us chose the number of epoch wisely. SGD algorithms have a very low cost in calculating due to it only pick one sample to update the model parameter, compare to other algorithms such as BGD (optimizer function that use whole sample to calculate next iteration). While, because of SGD are randomly pick data so it overcome model sink into local minimum and randomness. But it sometimes not stable, as this optimizer only pick on data to calculate gradient, so its next update may have a considerable effect by the noise of data. To overcome that problem, it needs an appropriate learning rate for SGD. If learning rate is too big, it may lead to vibration and never convergent, or if learning rate is small, it speeds of convergent will be very slow. Meanwhile, it may never sink into the local minimum and couldn't jump out. So, we adopt various methods to solve those problems, to start with, use momentum, add an inertia. When there is a big change in gradient, its momentum is big, and inertia will make it move further. As a result, it helps the parameter have a bit offset, avoid sink into local minimum. Learning rate decay was also a method to overcome divergent. As it helps loss function are steadily convergent when it is near the minimum value of loss function. Our loss function uses cross entropy loss function. As figure 3 shows how a data been process by using 3dresnet50 methods.

It is very suitable to solve binary classification problems. Finally, draw the confusion matrices. Fourth, store the model in each epoch. Then use print out the validation lost and accuracy, use the lowest validation lost model to training next model. weight decay=1e-3, learning rate=0.0001. Make sure not over fitting. For Resnet50 its process is very similar with 3dResent50, but Resent50 are likely to recognize different status by single picture.

DOI: 10.54254/2755-2721/108/2025.21197

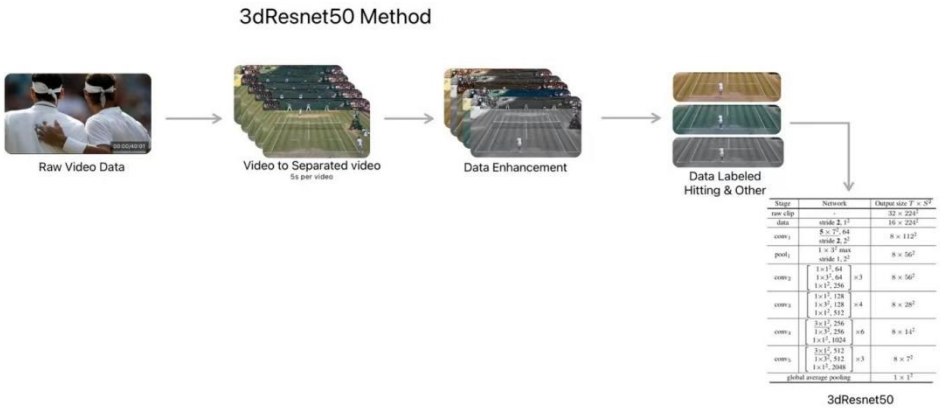

Figure 3: Shows the process of 3dresnet50 methods to detect player

3.6. STGCN

ST-GCN is a deep learning model designed to handle extemporization data, [8]specifically tailored for analyzing graph-structured data in sequences. Figure 4 shows the principles of the STGCN network and implies the different layers in the STGCN model. Key points can be viewed as nodes in the graph structure, while the connections between joints can be represented as edges. The combination of YOLO and skeletal recognition is typically achieved through the following steps: First, the YOLO model is used to detect and locate athletes in the video, generating bounding boxes to separate the athletes from the background. Then, the image regions containing the athletes are cropped and fed into a skeletal recognition model, such as Mediapipe or a ResNet-based pose estimation model, to identify the key joints of the athletes. Finally, the results of skeletal recognition are combined with YOLO's detection outputs to achieve more accurate action classification, such as standing, moving, or hitting. This approach effectively combines YOLO's object detection capabilities with skeletal recognition for action analysis, providing a powerful tool for automating the recognition and analysis of complex movements. These time-series data (sequential graph structures) are input into the ST-GCN model, where ST-GCN uses extemporization convolution operations to extract spatial and temporal features. The extracted extemporization features are then fed into a classifier, typically a fully connected network, to perform action classification. Ultimately, the system can identify specific actions being performed by the tennis player, such as swinging or running

3.7. Result Compare

However, these three different models are adopting different way to recognize the tennis player status. But each have their benefit and drawbacks. As table 1 shows the performance and different part of processed time for each model.

DOI: 10.54254/2755-2721/108/2025.21197

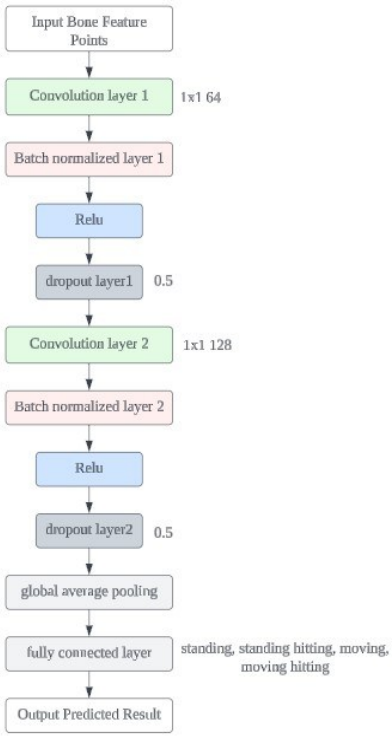

Figure 4: Shows STGCN network

Table 1: Shows the performance and speed for a different model

Model	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy	F1 Score	Avg Inference Time (5s Video)	Avg Inference Time (Frame)	Init Time (Read Video)	Init Time (YOLO Processing)	Init Time (Write In)	InitTime (Mediapipe)
3DResNet50 (Down Layer)	0.18973628	0.972819	0.25673829	0.900000	0.895	0.0205	-	6.2278	13.2910	13.3320	-
3DResNet50 (Up Layer)	0.20563295	0.8653215	0.356495	0.845316	0.757	0.0214	-	5.6225	11.2432	11.7395	-
ResNet50 (Previous)	0.17287654	0.950984	0.41987987	0.868563498	0.705	-	0.0221	4.5718	12.8953	12.4138	-
STGCN	0.18973628	0.972819	0.25673829	0.900000	0.608	-	0.0273	5.0718	16.4453	10.1318	32.9175

4. Experimentation

We downloaded some videos from the 2023 Wimbledon matches from various online video platforms to create the base datasets. We then clipped these videos into several short segments based on the three actions we want the model to recognize. Using YOLO, we identified and cropped the target athletes, creating a training set for the subsequent model training. You can access these clips through the following web page link: https://b23.tv/CrzmImT. Here are some comparisons between our model and the traditional model: previously resent50 model:

Training lost: 0.17287654 Traing accuracy: 0.95098398128 Validation lost: 0.41987987 Validation accuracy:0.868563498 Most recent 3dresnet50 model:

Training lost:0.18973628 Traning accuracy: 0.972818943232

Validation lost: 0.25673829 Validation accuracy:0.9000000000

In our study, we identified a major challenge: most tennis match videos are filmed from a fixed camera angle, resulting in our datasets consisting primarily of actions captured from a single perspective. This lack of diversity in viewpoints may limit the model's ability to learn the subtle differences between actions. Although we incorporated a skeletal recognition model to better analyze

DOI: 10.54254/2755-2721/108/2025.21197

the movements of tennis players, there remain some shortcomings in both recognition speed and accuracy. Additionally, when athletes perform complex movements or overlap with other players, the model may struggle to accurately distinguish the boundaries between different actions. Occlusion can further impact the model's recognition capability, particularly when key body parts, such as arms or legs, are partially obscured.

To address these issues, we recommend that future research focus on improving the model's generalization ability. This could be achieved by expanding the scale and diversity of the dataset, including videos from matches played under different weather conditions, on various court types (such as grass, hard, and clay courts), and featuring athletes of different genders and age groups. Data augmentation techniques (such as rotation, flipping, and adding noise) can also be employed to increase the diversity of the data, thereby enhancing the model's training outcomes. Additionally, exploring lightweight deep learning models could improve the model's robustness in situations with limited datasets and computational resources.

5. Conclusion

Our work presents a comprehensive study aimed at developing a monitoring model for tennis players' action states, such as standing, moving, and striking, during matches. The model is built on the YOLO architecture, combined with 3DResNet50 and MediaPipe for enhanced action recognition and skeletal structure analysis. Despite successfully applying these methods to real tennis match videos, the study identified challenges, particularly due to the fixed camera angles [9]in the dataset, which limited the model's ability to generalize across different perspectives. The report highlights the need for future research to focus on expanding the dataset's diversity and improving the model's generalization and robustness. The study also emphasizes the potential of combining YOLO with skeletal recognition for more accurate and automated analysis of complex movements in sports, offering valuable insights for training and competition analysis. In the future, we may apply ball tracking. [2,3]

References

[1] *Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 779–788, 2016.*

[2] *Juan A. Vega-Márquez, Mikel Martínez-Otzeta, Aitor Gómez-Arriola, and Víctor Hernández-Herrero. Semisupervised deep learning for soccer ball detection and tracking. Sensors, 23(9):4688, 2023.*

[3] *Xiaofei Li, Ronghua Luo, and Faiz Ul Islam. Tracking and detection of basketball movements using multi-feature data fusion and hybrid yolo-t2lstm network. Soft Computing, 28:1653–1667, 2024.*

[4] *Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 770–778, 2016.*

[5] *Jochen Görtler, Fred Hohman, Dominik Moritz, Kanit Wongsuphasawat, Donghao Ren, Rahul Nair, Marc Kirchner, and Kayur Patel. Neo: Generalizing confusion matrix visualization to hierarchical and multi-output labels. In CHI Conference on Human Factors in Computing Systems, pages 1–13. ACM, 2022.*

[6] *Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks for action recognition in videos. In ICCV, 2015.*

[7] *Andrew C Wilson, Rebecca Roelofs, Mitchell Stern, Nathan Srebro, and Benjamin Recht. On empirical comparisons of optimizers for deep learning. arXiv preprint arXiv:1710.06451, 2017.*

[8] *Yu-Chuan Huang, I-No Liao, Ching-Hsuan Chen, Tsì-Uí Ik, and Wen-Chih Peng. Tracknet: A deep learning network for concurrent ball tracking and pose estimation during sports events. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 4610–4619, 2018.*

[9] *Xingkui Zhu, Shuchang Lyu, Xu Wang, and Qi Zhao. Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios. arXiv preprint arXiv:2204.08621, 2022.*