Action Detection in Badminton Courts Using AVA Dataset and MMAction2 Architecture with Slow Fast Model

Yuhan Chen *

Department of Computer and Mathematical Sciences, University of Toronto Scarborough, Toronto, Canada

* Corresponding Author Email: hanky.chen@mail.utoronto.ca

Abstract. As a matter of fact, action detection technologies have emerged as powerful tools for a multitude of applications, from surveillance and security to healthcare and sports analytics. Badminton courts, especially those that are highly frequented, present unique challenges owing to the concentration of people engaged in diverse activities. Incidents such as unintended collisions between players, or unauthorized walking across the courts, are not uncommon and necessitate efficient monitoring for risk mitigation. This study addresses these issues by employing the MMAction2 architecture and the Slow Fast model for action detection in badminton courts. The author uses a dataset collected from multiple badminton facilities and leverages the AVA dataset for training and validation. The results are promising, with the model showing high levels of accuracy in identifying various types of actions: playing badminton, sitting, walking across the court, falling, and watching the game. The implications of this research are significant for badminton court management and safety.

Keywords: AVA Dataset, MMAction2 Architecture, action detection technologies.

1. Introduction

The area of action detection within computer vision has experienced considerable growth, driven by the increasing complexity of video data and the need for intelligent interpretation of such data [1]. It not only contributes to the safety and security in public spaces but also plays an integral role in optimizing operational efficiency and human behavior analysis [2]. As the prevalence of surveillance cameras continues to rise, action detection algorithms are being used in increasingly innovative ways, offering vital applications that extend beyond conventional uses in security [3, 4]. In sports analytics, real-time action detection offers valuable insights into player performance, enabling coaches and players to optimize strategies and improve in-game decision-making [5]. Moreover, it provides a valuable safety net for preventing accidents and ensuring the smooth conduct of various sporting activities. Given these practical applications and the rapid advancements in computer vision technologies, action detection has become a field of paramount importance for researchers and practitioners alike.

One model that has significantly influenced recent research in action detection is the Slow Fast [6], altering how video data is processed for action recognition tasks fundamentally. In traditional video processing, frames are often uniformly sampled or down sampled to mitigate computational demands [7]. While this approach simplifies the analysis and storage requirements, it risks overlooking high-frequency actions that manifest over shorter time spans, thereby compromising the ability to capture nuanced dynamics within the video. However, the Slow Fast model elegantly solves this problem by using two parallel pathways: a "slow" pathway and a "fast" pathway to capture the broader and more abstract temporal features. In contrast, the "fast" pathway processes video frames at the original frame rate to capture fine-grained, rapid temporal variations. Both pathways are then integrated to produce a more comprehensive representation of the actions taking place, resulting in significantly higher levels of accuracy. Additionally, Slow Fast employs a straightforward convolutional layer to integrate motion data from the fast pathway into the slow pathway following each stage [8].

While the Slow Fast model is highly effective in capturing a wide array of temporal features in video data, it is not without its drawbacks. Though PyTorch provides extensive flexibility, the out-of-the-box implementation of SlowFast can be computationally intensive [9] and somewhat challenging to fine-tune for specific use-cases. These challenges can be especially pronounced when adapting the model for real-time action detection in specific environments, like a badminton court, where a delicate balance between speed and accuracy is required. The usage of MMAction2 framework [10], which offers a flexible platform for SlowFast action detection. This framework is chosen not only for its ease of implementation but also for its versatility, enabling a range of customizations that are well-suited for tackling the unique challenges of action detection in badminton courts.

The motivation for this research stems from the increasing prevalence of badminton as a sport and the accompanying challenges in ensuring safety and operational efficiency in badminton courts. Despite the advanced state of action detection in various domains, there is a notable lack of research focusing on its application in badminton environments. This study aims to fill this gap by employing advanced action detection technologies like the SlowFast model to identify and analyze different types of human activities occurring in badminton courts. The goal is to facilitate better court management and enhance the safety measures, thereby providing both players and staff with a more secure and efficient environment.

2. Data and Method

2.1. Data Collection

In the model construction and performance section, several adjustments were made to optimize both the data and the computational efficiency for action detection in badminton courts. To maintain data richness, the author chose venues where badminton activities were frequently held, ensuring a diverse dataset that could effectively train the model to recognize a wide array of activities. First, each video was trimmed to only the first 15 seconds to optimize computational efficiency, as preliminary analyses indicated that most significant actions in badminton occur within this timeframe. Then, snapshots were captured every 30 frames based on pilot tests showing this frequency sufficiently captured most actions without being computationally expensive. The VGG Image Annotator (VIA) was selected for its intuitive interface and customization capabilities, to annotate a total of 276 snapshots obtained for the study. Annotations focused on five key action categories relevant to badminton: Playing, Sitting, Walking, Falling, and Watching. Multiple annotations per image were allowed for capturing instances where more than one action could be happening simultaneously. To tackle the limitations of a small and imbalance dataset in *Fig.1*, the author adapted the original via2ava code to partition the data into training and validation subsets using a 10:1 ratio. Additionally, the author employed a shuffle function to randomize the dataset's order, thereby facilitating a more robust validation process during training [11-13]. Fig. 1 provides a visual representation of the distribution of different action categories in the training dataset. Each bar in the chart represents one of the five key action categories relevant to badminton, i.e., Playing, Sitting, Walking, Falling, and Watching. The height of the bars indicates the number of instances for each category, offering insights into small and imbalance.

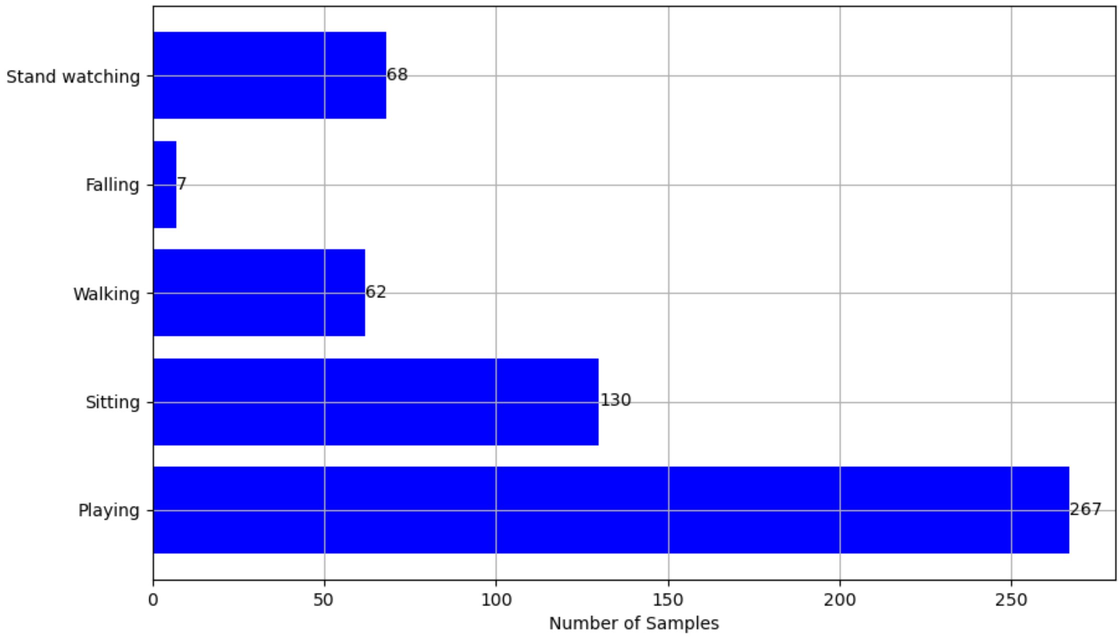


Figure 1. Data Distribution of the Training Dataset (Photo/Picture credit: Original).

2.2. Model and Configuration

For this study, the MMAction2 framework was chosen to implement the "Slowfast kinetics r50" model. This model employs a Fast CNN architecture integrated with a "ResNet3dSlowFast" backbone. Several factors informed the specific parameter choices for the backbone and pathways [14]. Resample_rate, Speed_ratio, and Channel_ratio was set to 8 in the backbone, with the intent to balance computational efficiency with accurate feature extraction. These settings assist in aligning the spatial and temporal dimensions of the slow and fast pathways, enabling the model to capture both high-level and low-level features efficiently. The depth of 50 in both slow and fast pathways was chosen due to its optimal balance between computational costs and model performance. In the slow pathway, a convolution kernel of (1, 7, 7) was selected to capture finer temporal information while maintaining a stride of 1 in the temporal dimension. Similarly, a stride of 1 was employed to maintain the spatial resolution in the pooling layer. These settings make the slow pathway adept at capturing long-term temporal dependencies. In contrast, the fast pathway features a base channel of 8 and a convolution kernel of (5, 7, 7). The choice of a larger kernel in the fast pathway aims to capture more localized spatial information, while the reduced base channel keeps the computational load manageable. This pathway is designed to recognize high-frequency changes, which are essential for understanding fast-paced activities like badminton rallies. This meticulously crafted configuration serves to maximize the efficacy and adaptability of the SlowFast model for the unique demands of action detection in badminton courts. The combination of these settings, including the adoption of a modified loss function to manage data imbalance, creates a powerful tool for accurate and efficient action detection. Codes are available on: https://github.com/Hank-coder/mmaction2-slowfast-badminton-detection.

3. Results and Discussion

3.1. Feature Engineering

This process begins with a series of transformations, each designed to refine the raw video frames into a format suitable for feeding into the model. To further improve the model's performance, especially in terms of adaptability to varying conditions common in real-world badminton courts, additional data augmentation techniques were incorporated into the MMAction2 framework's pipeline. These new transformations were meticulously chosen for their potential to enhance the model's resilience to various environmental factors such as lighting and contrast. Specifically, the following augmentations were introduced:

• Random Brightness: This transformation adjusts the brightness of each frame within a range of 0.75 to 1.25. The goal here is to make the model resilient to lighting fluctuations, a common variable in real-world settings like badminton courts.

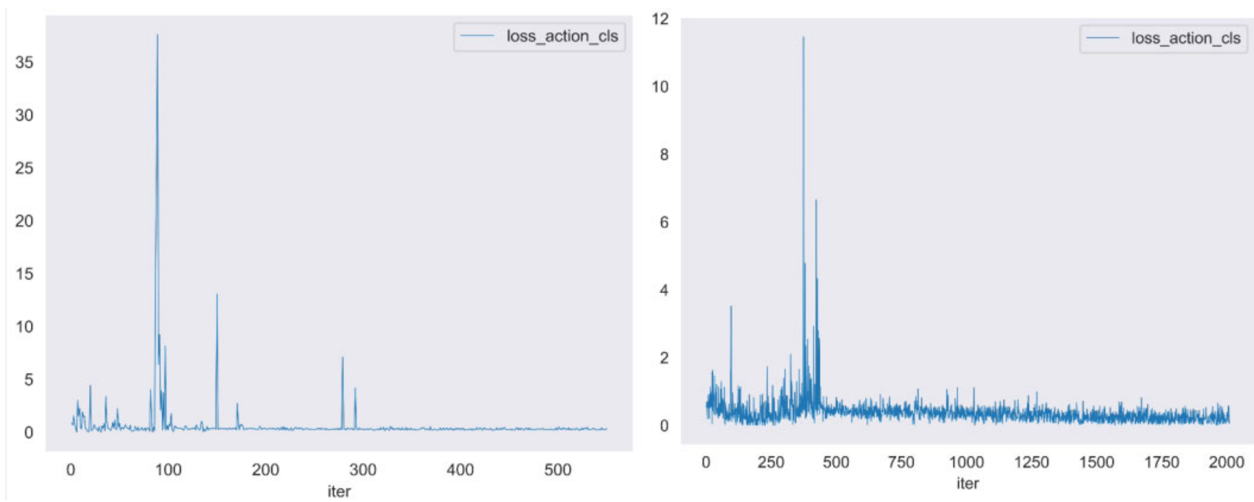
• Random Contrast: This transformation alters the contrast levels of each frame within a range of 0.8 to 1.2. The inclusion of this technique aims to amplify the model's ability to discern different actions even when visual conditions are not ideal.

• RandomGaussianBlur: A Gaussian blur is applied to each frame, with a sigma value set at 0.15. This feature is particularly beneficial for noise reduction, making the model less susceptible to minor and irrelevant fluctuations in the video data.

The incorporation of these additional data transformations into the pipeline was not just a methodological decision but a strategic one. It expanded the model's ability to generalize better by exposing it to a broader range of variations. These augmentations also served to further enrich the dataset, adding complexity to the feature set, and aiding the model in learning a more diverse set of representations. In sum, these feature engineering strategies had a salutary effect on the model's performance metrics. They equipped the model with the robustness and versatility needed to tackle the complexities of action detection in varying conditions often encountered in badminton courts.

3.2. Model Construction and Performance

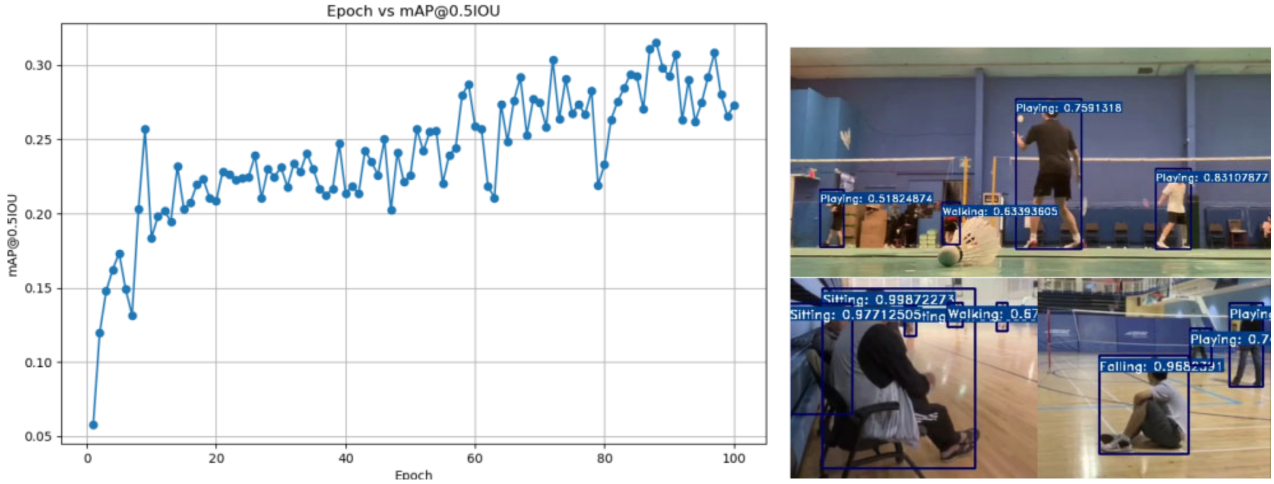
One such critical adjustment was in the configuration of the roi_head, specifically the bbox_head section which plays an instrumental role in action detection. Initially, the model struggled with the issue of data imbalance in Fig. 1, a prevalent challenge in action detection tasks. To address this, the original loss function was swapped out for binary cross-entropy with logits loss function which is well-suited for multi-label classification problems and effective for imbalanced datasets particularly. This loss function computes the binary cross-entropy between the target and the sigmoid of the predicted logits. Given the multi-label nature of action categories in badminton, ranging from 'playing' to 'falling', this loss function proved to be beneficial. However, during the training phase, it was observed that the loss fluctuated considerably in Fig. 2 (a), raising concerns about the model's stability and convergence. To mitigate this issue, a Batch Normalization (BN) layer was integrated into the loss function. This layer normalizes the classification scores (cls_score), aiming to stabilize and speed up the learning process by minimizing the internal covariate shift. By ensuring a smoother loss landscape, the BN layer contributed to more stable convergence in Fig. 2 (b), thus enhancing the model's overall performance.



(a) (b)

Figure 2. Comparison of classification loss: (a) the original settings prior to any adjustments and the modified configuration, highlighting changes (Photo/Picture credit: Original).

In the final evaluation, as depicted in Fig. 3(b), the model performed notably well in differentiating among 'Falling', 'Sitting', and 'Playing' actions. However, it showed occasional inaccuracies in distinguishing between 'Walking' on the court and 'Playing' badminton. Despite this limitation, the extensive training and validation strategy underpinned by the MMAction2 framework proved successful in elevating the model's overall performance. To elaborate, the training was conducted using MMAction2's built-in train utility, and the model's performance was assessed using the spatiotemporal_det utility for testing. A validation step was incorporated at the end of each epoch to facilitate a comprehensive evaluation of the model's ongoing performance. Initially, 100 epochs were run to evaluate the model and fine-tune its parameters. At this stage, as shown in Fig. 3(a), the model's mean average precision (mAP) at a 50% Intersection Over Union (IOU) threshold already exceeded 30%. Encouraged by these promising early results, the training was then extended to a total of 500 epochs, after which the mAP reached nearly 50%. This rigorous, multi-phase training and validation process successfully validated the effectiveness of the custom modifications and augmentations, setting a promising path for further refinements and real-world applications.



(a) (b)

Figure 3. Evaluation and Result of (a) Epoch vs. MAP at 50% IOU threshold and (b) test set performance on badminton actions (Photo/Picture credit: Original).

3.3. Explanation and Implication

The application of the binary cross entropy loss coupled with a Batch Normalization layer, showcases the model's efficiency in handling data imbalance, which is crucial for accurately identifying less frequent yet important anomalous activities. The author fine-tuned the model to perform effectively on limited data, as demonstrated by employing a 10:1 ratio for the training-to-validation data split and by randomizing the dataset order. Additionally, custom data augmentation techniques were introduced to enhance the model's adaptability to varying lighting and visual conditions. These features are particularly vital for real-world applications where inconsistent lighting could otherwise compromise the model's ability to reliably detect abnormal behaviors. In essence, the model is geared towards being a reliable tool for identifying abnormal behaviors in badminton courts, thus potentially enhancing the safety and security measures of these environments.

4. Limitations & Future Outlooks

This study faces several limitations that warrant discussion despite its advancements in the field of action detection within badminton courts. Primarily, the fluctuating loss metrics during the training phase present a challenge for model stability. This might be directly linked to both the limited size and potentially inconsistent quality of our dataset. Secondly, the model's capacity to recognize only five key actions due to data constraints significantly limits its practical application. Actions like 'Smash,' 'Drop Shot,' or 'Clear Shot,' which are quintessential to badminton, remain beyond its current detection capabilities. Furthermore, computational efficiency also presents a limitation. Though strategies were employed to minimize computational burden, the model still does not reach real-time processing speed, which is crucial for monitoring and immediate intervention in cases of unusual behavior detection. Lastly, the model has not been sufficiently tested for its adaptability to different lighting and environmental conditions. While some data augmentations were used, these are no substitute for real-world variability. This limitation potentially affects the model's ability to generalize its detections across different badminton courts or during different times of the day.

Addressing these limitations provides multiple avenues for further research and improvements. For starters, enriching the dataset with more diverse actions and environmental conditions can be a primary focus. Increasing the volume and quality of data can not only improve the model's accuracy but also stabilize the loss metrics, enabling more robust training. Regarding real-time applications, future developments could explore the incorporation of more efficient object detection algorithms. The thought of combining this model with a "Yolov5+SlowFast" real-time behavior detection algorithm is particularly enticing [12]. Replacing Faster R-CNN with YOLO could drastically increase processing speeds to near real-time, opening doors for practical, real-time monitoring. Another compelling avenue could be the inclusion of temporal context by employing object tracking algorithms. This would enable the model to maintain a context-rich understanding of ongoing actions, making the behavioral classification less discrete and more continuous. This could particularly benefit the monitoring of more complex actions or sequences of actions that take place over multiple frames or seconds. Moreover, thorough testing in various environmental settings should be a priority. Evaluating how well the model generalizes across different lighting conditions, crowd sizes, or even different types of badminton courts can provide insights into the model's adaptability. While this study has laid some groundwork in the domain of action detection in badminton, much remains to be done. Refining the recent model's architecture, expanding the dataset, and incorporating real-time processing capabilities are just some of the next steps that could significantly elevate this research from being a mere proof of concept to a robust, real-world application.

5. Conclusion

To sum up, the core intent of this research was to proficiently identify abnormal behaviors on badminton courts, with the overarching objective of bolstering safety and streamlining court management. In pursuit of this aim, the author embarked on constructing a proprietary dataset tailored to the unique nuances of the badminton environment, was meticulously refined to optimize data distribution and was subsequently processed using a specialized conversion method to make it compatible with the chosen architecture. Venturing further into refining the model, the inherent loss function was substituted with the more adept binary cross entropy. In an astute move to counteract the oscillations in the loss during training sessions, a Batch Normalization layer was annexed to the architecture. The culmination of these methodical enhancements was a robust model, finely tuned to accurately discern and categorize five pivotal actions commonly observed within a badminton courts. This endeavor not only showcases the profound impact of tailored modifications but also underscores the significance of custom datasets in shaping machine learning outcomes. Despite its success, the model faces limitations related to data volume and fluctuating loss, areas that could be improved in future research. The current study serves as a valuable stepping stone for future work aimed at real-time, multi-action detection in sports environments, demonstrating the efficacy of these modifications in tackling specific, real-world challenges. The significance of this research lies in its potential to improve the safety and management of badminton courts by providing a foundational framework for action detection. Despite its limitations, this study paves the way for more advanced systems capable of real-time monitoring and complex action recognition, thereby adding a new layer of sophistication to sports analytics and security.

References

[1] F. Camarena, M. Gonzalez-Mendoza, L. Chang and R. Cuevas-Ascencio, Math. & Comp. Appl., 28 (2), 61 (2023).

B. Degardin and H. Proença, Appl. Sci., 11 (18), 8324 (2021).

[2] C. B. Jin, S. Li and H. Kim, arXiv preprint arXiv:1710.03383 (2017).

[3] A. Karbalaie, F. Abtahi and M. Sjöström, Multi. Tools & Appl., 81, 35463 – 35501 (2022).

[4] S. Wang, Y. Song, K. Chen, et al. *LCTS: Longest Continuous Temporal Sequences for Action* Detection, LCTS Report, (2015).

[5] C. Feichtenhofer, H. Fan, J. Malik, and K. He, *Slowfast networks for video recognition* In Proceedings of the IEEE/CVF international conference on computer vision (2019) pp. 6202 - 6211.

[6] E. P. Bennett, and L. McMillan, ACM Trans. on Graphics, 26 (3), 102 – es (2007).

[7] Wei, D., Tian, Y., Wei, L., Zhong, H., Chen, S., Pu, S., & Lu, H. Efficient dual attention SlowFast networks for video action recognition. Comp. Vision & Image Under. 222, 103484 (2022).

[8] V. Saxena, Slow-Fast PyTorch Implementation [Computer software] (2020), Retrieved from: https: //github.com/vaib-saxena/Slow-Fast-pytorch-implementation.

[9] MMAction2 Contributors, OpenMMLab's Next Generation Video Understanding Toolbox and Benchmark [Computer software] (2020), Retrieved from: https://github.com/open-mmlab/mmaction2.

[10] W. Zhu, JN-OpenLib-mmaction2 [Computer software] (2020), Retrieved from: https: //github.com/Wenhai-Zhu/JN-OpenLib-mmaction2.

[11] Wufan-tb, Yolo_slowfast [Software] GitHub (2021), Retrieved from: https://github.com/wufan-tb/yolo_slowfast.

[12] H. Jiang, *Theme: ava format data transformer*, Computer software (2022, March 14).

[13] S. Ren, K. He, R. Girshick, and J. Sun, F*aster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*, In Advances in Neural Information Processing Systems (NIPS) (2015) pp. 17 - 28.