EasyChair Preprint

№ 8982

A Deep Learning Based Framework for Badminton Rally Outcome Prediction

Yong En Tan, John See, Junaidi Abdullah, Lai Kuan Wong and Kar Weng Ban

October 4, 2022

A deep learning based framework for badminton rally outcome prediction

Yong En Tan
Faculty of Computing & Informatics
Multimedia University
Cyberjaya, Selangor, Malaysia
yongen_@outlook.com

John See
School of Mathematical and Computer Sciences
Heriot-Watt University, Malaysia
Putrajaya, W.P., Malaysia
j.see@hw.ac.uk

Junaidi Abdullah
Faculty of Computing & Informatics
Multimedia University
Cyberjaya, Selangor, Malaysia
junaidi.abdullah@mmu.edu.my

Lai Kuan Wang
Faculty of Computing & Informatics
Multimedia University
Cyberjaya, Selangor, Malaysia
lkwong@mmu.edu.my

Kar Weng Ban
Faculty of Computing & Informatics
Multimedia University
Cyberjaya, Selangor, Malaysia
kwban@mmu.edu.my

***Abstract*—Badminton is a fast-paced net-based sport in which players' actions and strategies in-game determine their chances of winning. With sports analytics gaining popularity due to its capability in providing valuable information for players and coaches to counter opponents with tactics, some recent research works attempted to perform stroke recognition. However, there has been little research into using stroke sequences for sports analytics. In this paper, we propose a player-independent framework to investigate the relationship between strokes and rally outcome in badminton games. To classify the rally outcome, strokes are represented by deep features extracted using CNN and fitted into LSTM. Experiments with various variants of GRU and LSTM models demonstrate that Bidirectional LSTM gives the best prediction performance, with ResNet-18 as the feature extractor. Additional experiments were performed to study different features that represent the stroke as plain text and player's pose, as well as methods to augment a small sequential dataset.**

***Index Terms*—sports analytics, badminton analytics, rally outcome prediction**

I. Introduction

Sports videos are available on platforms like YouTube. Channels such as Badminton World Federation, FIBA, and International Tennis Table Federation share the most recent full-game recordings. Many researchers are taking advantage of the opportunity to access these free videos to conduct different aspects of research: sport analytic [24], match outcome prediction [1], coaching assistant [25], and dataset building [9], [22]. Sports analytics is a useful application that provides valuable information for players and coaches to counter opponents with tactics or improve the necessary items during training. Sport analytics is gaining more advantages to process large amount of videos with the help of deep learning techniques, reducing the need for human labour in annotating the videos [7].

In badminton, existing analytic works that use deep learning are done in the following aspects: general strategy classification [5], strokes recognition and shuttlecock tracking task [15], [18], match-level outcome prediction [23]. However, studies on the relationship between the sequence of strokes (actions) and rally outcome are rare. We believe every in-game action is crucial for players to win their games. In this paper, we introduce a framework to predict the rally outcome for a single badminton match based on action sequences performed by both badminton players. We experimented with different features and ways to augment a small dataset. Experimental results show that deep features extracted using ResNet-18 have higher representation power than HoG and n-gram features. In addition, experiments with various variants of GRU and LSTM models demonstrate that Bidirectional LSTM performs the best in predicting the rally outcome.

II. Related Works

Several badminton analytic works [5], [27], [28] uses machine learning and computer vision techniques to analyse the broadcast videos. In general, Weeratunga et al. [27], [28] profile the players by classifying the players' movement based on players' location and shuttlecock trajectory. Chu and Situmeang [5] extend the prior works with additional tasks, such as player tracking and stroke classification. This work also classifies strategies in badminton into either offensive or defensive strategies, based on prior results. An attempt is made by Ghosh et al. [10] to automate the process of analysing badminton games. The automated tasks include game section segmentation, player detection, and tracking, and stroke classification. The outputs of these tasks are being used as metrics (player dominance status, the average speed of players, average reaction of players, etc) for analysis purposes. Wang et al.'s work [26] has a similar goal to our work in classifying the rally outcome based on the sequence of strokes, but with a different dataset that consists of different types of strokes. This dataset is annotated by using their proposed badminton language, BLSR (Badminton Language from Shot to Rally), with input from domain experts. The classification

is performed with Bi-GRU network that takes in different encoded inputs (strokes, location, etc.).

In relation to racket sports analytics, various works about predicting tennis match outcomes [4], [8], [12], [17] have also been published. The works are similar in that the match outcome prediction is based on the players' past performance data (aces, scored, successful first serves etc.), situational data (tournament level, court surface type, etc.) and players' basic demographic (seed, height, and etc.) information about the players. Makino. et al [16] believe features used by previous works are not sufficient to provide strategic advice to the players. Thus, they broaden the study by including the frequency of various shot patterns in rallies.

III. Proposed Approach

A. Overview of framework

In a badminton singles game, a *rally* is played with a sequence of *strokes* (actions) $R_j = \{s_1, ..., s_{K_j}\}$ taken by both players, where K_j is the number of strokes in the j-th rally. Each rally can be won by one of the two players (hence, the player scores a 'point'), which is a binary state *i.e.* $y_R = 1$ if a particular referenced player wins, and $y_R = 0$ if the player loses. In *rally outcome prediction*, the aim is to classify the outcome of a rally y given a sequence of strokes R. In this paper, we propose a framework for classifying a rally as won or lost based on the last ten strokes. Following the illustrated activities in Fig.1:

1) A frame f_s is selected from the time period where the stroke is being performed, to represent a stroke. The players in the frame will be detected and cropped out using Faster R-CNN [20].
2) The two cropped player frames constitute the frames where a player performs the stroke, $f'_{s,Player_1}$ and the opponent waits for it, $f'_{s,Player_2}$.
3) Using ResNet18 [14], deep features are generated to represent the frames of both cropped players. The extracted features of both players, $fv_{s,Player_1}$ and $fv_{s,Player_2}$, are then being vertically concatenated to form an input x to the BiLSTM for a time step.
4) Finally, each rally is presented as $R_j = \{x_{s_1}, ..., x_{s_{10}}\}$ and fit into a two-layer BiLSTM.

B. Dataset

All the ten annotated single-player badminton (five females & five males) videos from Ghosh's work [10] are used in our work. The videos are from the 2012 Summer Olympics [9]. Under Ghosh's annotation, a player's stroke can be categorised as one of the five types of stroke: *serve*, *lob*, *smash*, *backhand*, and *forehand*. The annotation includes the time period of strokes and the number of rallies won by both players. The performed strokes are further split into *top* or *bottom* based on players' location. In total, there are 754 rallies annotated in this dataset.

1) Data Pre-processing: The pre-processing steps are taken to prepare the dataset for prediction:

- *Different numbers of stroke:* In handling different lengths of each rally R_j, the last 10 strokes of the sequence are used. Rallies with number of strokes less than three are being omitted as it is too short to provide useful information about a rally and a coach would treat three strokes as a pattern [26]. Rallies with three to nine strokes will be pre-padded to make it 10 strokes.
- *Scene with different angles of view:* Different angles of view (side view, player close focus view, etc) from the match will be captured by the broadcasters. To maintain frame consistency, only rallies with all strokes that have a far court-end view will be kept, leaving 498 rallies.
- *Rally labelling:* If Player 1 wins a rally, it will be labelled as 0, otherwise, it is labelled as 1. In defining Player 1 and Player 2, we select the player who began the particular game first as Player 1 and the other player as Player 2.

2) Preparing the features for each stroke: Despite the fact that the videos are available at 25 fps, a player can perform a stroke in less than a second because badminton is a fast-paced sport. Thus, each stroke has a different total number of frames. Due to this condition, only a single frame is selected to represent a stroke. Specifically, the frame located in the middle of its sequence of frames $f_1, ..., f_N$, is selected to represent a stroke. The selected frame will be passed into Faster R-CNN to crop out the players' regions, $f'_{s,Player_1}, f'_{s,Player_2}$. The two cropped player frames are resized to a dimension of 224 x 224. Deep features for both resized frames are extracted from the output of the fully connected layer in ResNet-18. These two deep feature vectors with 512-D are concatenated vertically to form a 1024-D of feature x_s to represent the input to BiLSTM at a time step.

C. Rally Outcome Prediction

In the dataset D, each entry constitutes a rally R_j and a ground-truth label y_j, where $y \in C = \{0, 1\}$. Each rally is a sequence of 10 strokes, with each stroke represented by the concatenated ResNet-18 features, $R_j = \{x_{s_1}, ..., x_{s_{10}}\}$. Given the dataset $D = \{R_j, y_j\}$, we fit D into our selected model M, a BiLSTM network that classifies the outcome of a rally as 0 or 1: $M(P) = \{0, 1\}$.

1) Bidirectional Long Short-Term Memory (BiLSTM) network: BiLSTM [11] consist of two LSTMs [21], where the inputs flow in two directions; the forward and the backward direction. It maps the rally sequence $x_j = \{x_{s_1}, ..., x_{s_T}\}$ into a H function to generate the hidden vector sequence $h_j = \{h_{s_1}, ..., h_{s_T}\}$ for $t = 1$ to T time steps, where T is set to 10. Each hidden vector h at t time step is computed for both directions:

$$\begin{aligned} \overleftarrow{h_t} &= H(\overleftarrow{W_{x_s}} x_{s_t} + \overleftarrow{W_{h_s}} h_{s_{t-1}} + \overleftarrow{b_{s_t}}) \\ \overrightarrow{h_t} &= H(\overrightarrow{W_{x_s}} x_{s_t} + \overrightarrow{W_{h_s}} h_{s_{t+1}} + \overrightarrow{b_{s_t}}) \end{aligned} \tag{1}$$

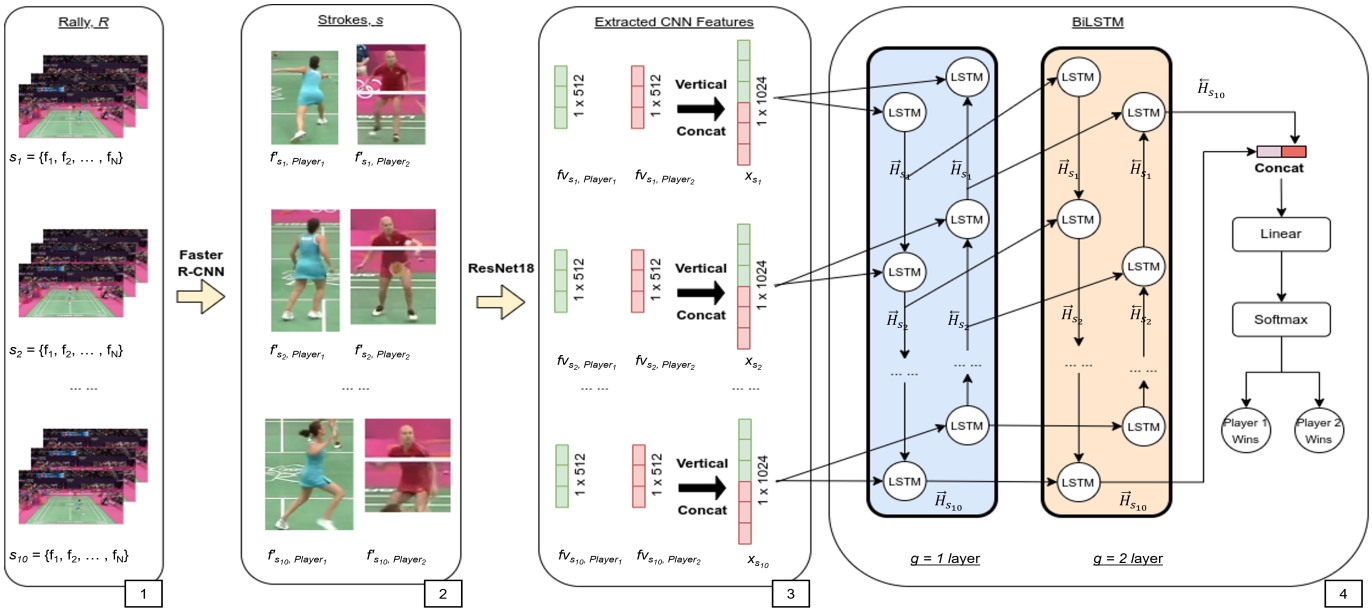

Fig. 1: Overview of framework.

where W and b are the weight matrices and bias vectors respectively. The H function in LSTM is defined as:

$$
\begin{aligned}
i_{s_t} &= \sigma(W_{x_s i} x_{s_t} + W_{h_s i} h_{s_{t-1}} + W_{c_s i} c_{s_{t-1}} + b_{i_{s_t}}) \\
f_{s_t} &= \sigma(W_{x_s f} x_{s_t} + W_{h_s f} h_{s_{t-1}} + W_{c_s f} c_{s_{t-1}} + b_{f_{s_t}}) \\
o_{s_t} &= \sigma(W_{x_s o} x_{s_t} + W_{h_s o} h_{s_{t-1}} + W_{c_s o} c_{s_t} + b_{o_{s_t}}) \\
c_{s_t} &= f_{s_t} \odot c_{s_{t-1}} + \\
&\qquad i_{s_t} \odot \tanh(W_{x_s c_s} x_{s_t} + W_{h_s c_s} h_{s_{t-1}} + b_{c_{s_t}}) \\
h_{s_t} &= o_{s_t} \odot \tan h(c_{s_t})
\end{aligned}
\tag{2}
$$

where i, f, o, c, σ, and $\odot$ are input gate, forget gate, output gate, cell activation vector, logistic sigmoid function, and element-wise product of vectors respectively. This process is performed for two layers, $g = 1$ and $g = 2$. To predict the point outcome $\hat{y}_j$ of a rally R_j, the probability distribution of $Pr(y_j = c)$ is computed by concatenating $\overleftarrow{h_{s_{10}}}$ and $\overrightarrow{h_{s_{10}}}$ in layer $g = 2$, and connecting it to a linear layer $\hat{y}_j = W_{\overleftarrow{h}_{s_{10}}} \overleftarrow{h_{s_{10}}} + W_{\overrightarrow{h}_{s_{10}}} \overrightarrow{h_{s_{10}}} + b_{h_{s_{10}}}$, followed by applying the softmax function $softmax(\hat{y}_j) = \frac{\exp(\hat{y}_j)}{\sum_0^{c=1} \exp(\hat{y}_c)}$.

2) Network Training: In training the network, the error between y and $\hat{y}$ is calculated by the cross-entropy loss, $CELoss(y_j, \hat{y}_j) = -\sum_{j=1}^{C} y_j \log \hat{y}_j$ and optimized using Adam [19]. The process is monitored by early stopping to avoid overfitting.

IV. Experiments and Discussion

For all experiments, the pre-processed dataset in Section III-B1 is partitioned into two sets – a training set (398 rallies) and a test set (100 rallies) based on stratified sampling. Accuracy, Area Under Receiver Operating Characteristics (AUROC) curve, and brier score [3] are the metrics used to evaluate model performance. The brier score is defined as $\frac{1}{D} \sum_j (Pr(y_j = c) - y_j)^2$.

A. Evaluation of sequential models

We ran experiments on several sequential architectures including LSTM, GRU and Bayesian LSTM [2] using ResNet-18 and HoG features [6] for the purpose of comparisons. HoG features are generated for the cropped regions of both players, $f'_{st,Player1}$ and $f'_{st,Player2}$, to form two 3780-D feature vectors, which are then concatenated vertically. The results are shown in Table I. BiLSTM with ResNet-18 features outperforms all other models, with an accuracy of 0.70, an AUROC of 0.61, and a brier score of 0.30. In comparing ResNet-18 and HoG features, ResNet-18 performs better than HoG for most models, except GRU and LSTM. This could be due to the ability of convolution layers to capture image features in much greater detail. Even though HoG performs better in these two models, it is less efficient compared to ResNet-18, due to HoG having a significantly larger feature size than ResNet-18.

Notably, because there are insufficient datasets disclosed and different types of strokes are annotated, we are unable to compare our work with Wang et al's [26], which is the closest to ours.

B. Evaluation of different features

Other than the ResNet-18 and HoG features, we have also experimented with other features:

- *n-gram*: A n-gram is a contiguous sequence of combined n words generated from the original sentence, where the n indicates the number of words. In our context, the words are the strokes. We experimented with three sets of n-gram features, specifically unigram, bigram and trigram.
- *Keypoint-RCNN*: KeyPoint-RCNN [13] is a modified Mask R-CNN originally designed to classify key joints

TABLE I: Models Performance between HoG & ResNet-18

Models	Accuracy		AUROC		Brier score	
	HoG	ResNet-18	HoG	ResNet-18	HoG	ResNet-18
GRU	0.66	0.62	0.61	0.57	0.34	0.38
BiGRU	0.60	0.63	0.57	0.60	0.40	0.37
Bayesian LSTM	0.60	0.64	0.49	0.54	0.40	0.36
LSTM	0.68	0.61	0.57	0.53	0.32	0.39
BiLSTM	0.62	**0.70**	0.59	**0.61**	0.38	**0.30**

TABLE II: Performance of different features being used in BiLSTM

Features	Accuracy	AUROC	Brier Score
Unigram	0.54	0.52	0.42
Bigram	0.54	0.52	0.41
Trigram	0.54	0.51	0.43
KeyPoint-RCNN	0.67	0.57	0.33
HoG	0.62	0.59	0.38
ResNet-50	0.63	0.58	0.37
ResNet-101	0.62	0.54	0.38
ResNet-18	**0.70**	**0.61**	**0.30**

of the human body. The features for both cropped player regions, $f'_{st,Player1}$ and $f'_{st,Player2}$, are extracted from the last convolutional layer of KeyPoint R-CNN and average pooling is applied to obtain the stroke features.

The different feature sets have varying sizes. Following the order in Table II, the size of different feature sets are: 5, 10, 15, 1024, 7560, 4096, 4096, and 1024 respectively. The comparative results of the different features with BiLSTM is tabulated in Table II. The result shows that ResNet-18 has the best performance compared to all other features. Between text (n-gram) and image (HoG, KeyPoint-RCNN, ResNet) features, our results demonstrate that BiLSTM works better with image features. We have also experimented with deeper ResNet variants but results show that the performance does not improve with features from these variants.

V. Conclusion and Future Works

In this work, we propose a framework to predict badminton the rally outcome based on the last 10 strokes. The framework uses ResNet-18 to extract deep features that represent both players' stroke sequences, which are then passed to BiLSTM for point prediction. In the future, we would like to explore other available annotated datasets due to the limited type of strokes in the current dataset. In addition, we plan to investigate transformer-based models and ensemble methods to improve the prediction performance.

References

[1] Ryan James Beal, Stuart Middleton, Timothy Norman, and Sarvapali Ramchurn. Combining machine learning and human experts to predict match outcomes in football: A baseline model. 2021.

[2] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015.

[3] Glenn W Brier et al. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

[4] Vincenzo Candila and Lucio Palazzo. Neural networks and betting strategies for tennis. *Risks*, 8(3):68, 2020.

[5] Wei-Ta Chu and Samuel Situmeang. Badminton video analysis based on spatiotemporal and stroke features. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pages 448–451, 2017.

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[7] Dazhen Deng, Jiang Wu, Jiachen Wang, Yihong Wu, Xiao Xie, Zheng Zhou, Hui Zhang, Xiaolong Zhang, and Yingcai Wu. Eventanchor: Reducing human interactions in event annotation of racket sports videos. *arXiv preprint arXiv:2101.04954*, 2021.

[8] Zijian Gao and Amanda Kowalczyk. Random forest model identifies serve strength as a key predictor of tennis match outcome. *arXiv preprint arXiv:1910.03203*, 2019.

[9] Anurag Ghosh, Suriya Singh, and CV Jawahar. Badminton videos at london 2012 olympics.

[10] Anurag Ghosh, Suriya Singh, and CV Jawahar. Towards structured analysis of broadcast badminton videos. pages 296–304, 2018.

[11] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pages 273–278. IEEE, 2013.

[12] Wei Gu and Thomas L Saaty. Predicting the outcome of a tennis tournament: Based on both data and judgments. *Journal of Systems Science and Systems Engineering*, 28(3):317–343, 2019.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Tzu-Han Hsu, Ching-Hsuan Chen, Nyan Ping Jut, Tsì-Uí İk, Wen-Chih Peng, Yu-Shuen Wang, Yu-Chee Tseng, Jiun-Long Huang, Yu-Tai Ching, Chih-Chuan Wang, et al. Coachai: A project for microscopic badminton match data collection and tactical analysis. In *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pages 1–4. IEEE, 2019.

[16] M Makino, Tomohiro Odaka, Jousuke Kuroiwa, Izumi Suwa, and Hideyuki Shirai. Feature selection to win the point of atp tennis players using rally information. *International Journal of Computer Science in Sport*, 19(1):37–50, 2020.

[17] Pierre-Etienne Martin, Jenny Benois-Pineau, Renaud Péteri, and Julien Morlier. Sport action recognition with siamese spatio-temporal cnns: Application to table tennis. In *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*, pages 1–6. IEEE, 2018.

[18] NA Rahmad and MA As'ari. The new convolutional neural network (cnn) local feature extractor for automated badminton action recognition on vision based data. In *Journal of Physics: Conference Series*, volume 1529, page 022021. IOP Publishing, 2020.

[19] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[21] Haşim Sak, Andrew Senior, and Françoise Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, 2014.

[22] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2616–2625, 2020.

[23] Manoj Sharma, Naresh Kumar, Pardeep Kumar, et al. Badminton match outcome prediction model using naïve bayes and feature weighting technique. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15, 2020.

[24] Manuel Stein, Halldor Janetzko, Andreas Lamprecht, Thorsten Breitkreutz, Philipp Zimmermann, Bastian Goldlücke, Tobias Schreck, Gennady Andrienko, Michael Grossniklaus, and Daniel A Keim. Bring it to the pitch: Combining video and movement data to enhance team sport analysis. *IEEE transactions on visualization and computer graphics*, 24(1):13–22, 2017.

[25] Jianbo Wang, Kai Qiu, Houwen Peng, Jianlong Fu, and Jianke Zhu. Ai coach: Deep human pose estimation and analysis for personalized athletic training assistance. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 374–382, 2019.

[26] Wei-Yao Wang, Teng-Fong Chan, Hui-Kuo Yang, Chih-Chuan Wang, Yao-Chung Fan, and Wen-Chih Peng. Exploring the long short-term dependencies to infer shot influence in badminton matches. *arXiv preprint arXiv:2109.06431*, 2021.

[27] Kokum Weeratunga, Anuja Dharmaratne, and Khoo Boon How. Application of computer vision and vector space model for tactical movement classification in badminton. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 76–82, 2017.

[28] Kokum Weeratunga, Khoo Boon How, Anuja Dharmaratne, and Chris Messom. Application of computer vision to automate notation for tactical analysis of badminton. In *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, pages 340–345. IEEE, 2014.