Badminton shot classification and pose extraction to inform better training for amateur players

Deepak Talwar, Pratap Kishore Desai, Rahul Devaragudde Gopala and Saurabh Shirur
Department of Computer Engineering, San José State University
San José, CA
Email: deepak.talwar@sjsu.edu, pratapkishore.desai@sjsu.edu, rahul.devaraguddegopala@sjsu.edu, saurabh.shirur@sjsu.edu

***Abstract*— The use of technology in the area of sports has been ever increasing, yet Badminton, which is one of the most popular sports in terms of participation in the world, has not seen the influx of technology in the same way as other sports have. While in other sports like American football and basketball, players use technology extensively to assist in coaching, badminton players have to rely on their coaches for proper guidance. This paper uses deep learning to create a system that helps amateur players in badminton coaching. The first step is to identify the type of shot played. This is performed by training a Convolutional Neural Network (CNN) that performs multi-class shot classification. The shots are classified into four categories - smash, defense, backhand and net-drop return. The data to train this CNN is collected through scrubbing of broadcast badminton matches of top-5 ranked male singles players uploaded to YouTube. Our model returns a 84% accuracy in shot detection. The second step is to learn the qualities of a good shots by learning features from the dataset. These features are then compared to the features of shots played by an amateur player to generate insights on the quality of their shots. This provides them with direct areas to improve on.**

I. INTRODUCTION

Badminton is the fastest racket sport in the world, with the record of fastest smash hit by a male player during a tournament standing at 417 kmph [1] hit by Lee Chong Wei of Malaysia during Daihatsu Yonex Japan Open in 2017 [2]. It is also the second most popular sport in terms of participation in the world, yet badminton does not seem to get the same amount of attention when compared to other sports [3]. The fast nature of badminton makes it a very technical sport, which makes giving and receiving quality training difficult.

Despite having a smaller court size, the average distance traveled by a badminton player in a singles game is 3.7 miles, compared to 1.8 miles traveled by a tennis player on average. Furthermore, an average badminton match lasts approximately 1 hour 15 minutes, while an average tennis match is about 3 hours long, meaning that in half the time, badminton players travel twice the distance compared to tennis players [3]. Accordingly, in addition to technique, stamina is also a major requirement for badminton players. This calls for the need to provide quality coaching and training for players when they are young at the grassroots level. Availability of quality coaching and infrastructure, however, is lagging, which is why the sport has not garnered the same amount of attention and sponsors despite being extremely popular [4].

Over the past two decades, technology has played a big role in the growth and development of various sports. Players have embraced technology and analytics to learn insights and gain an edge over their opponents. American Football is a prime example where technology has helped teams not only to understand their opponents' strategies better, but also help players train against robots to minimize injuries. Intel has installed 30 5K cameras in football stadiums that capture volumetric data to uniquely render 3D replays and create multi-perspective views of the plays [5]. Mobile Virtual Players (MVP) makes dummy training robots that players can tackle during practice to reduce the number of unnecessary impacts and minimize the chance of injury [6]. HomeCourt.ai is a new iOS app that uses machine learning and pose detection to track basketball players' shots and provides meaningful insights such as release time, release angle, speed and vertical, just by using the iPhone camera [7]. This technology can help amateur players keep track of their progress over time and set goals for themselves.

The use of technology in badminton, however, has been largely limited to usage on the court. Badminton World Federation (BWF) introduced Hawk-Eye for confirming line-judge's calls and checking the landing of the shuttle [8]. Previous works involving machine learning in badminton have implemented tracking of shuttles for auto-linesmen systems [9], stroke classifiers using histogram of oriented gradient (HOG) representation and support vector machines (SVM) [10][11]. [12] uses a SpatialCNN for stroke segmentation and create a system for analyzing badminton broadcast videos.

Use of technology for badminton training has been limited to using IMU equipped attachments to badminton rackets [14] [15]. Although such devices provide a detailed view of the motion of the racket, they add weight to the racket which is highly undesirable in the sport of badminton. In addition, these devices provide no feedback on the quality of the footwork and on the success or failure of any shot. In this work, we use deep learning to identify shots hit by players and provide insights on the quality of the shot by comparing them to similar shots hit by professional badminton players. Our goal is provide amateur players with insights and recommendations to improve their quality of play by learning from the pose of shots hit by professionals. We first use a trained CNN to detect the type of shot played, then use pose detection [16] to extract features (such as lunge

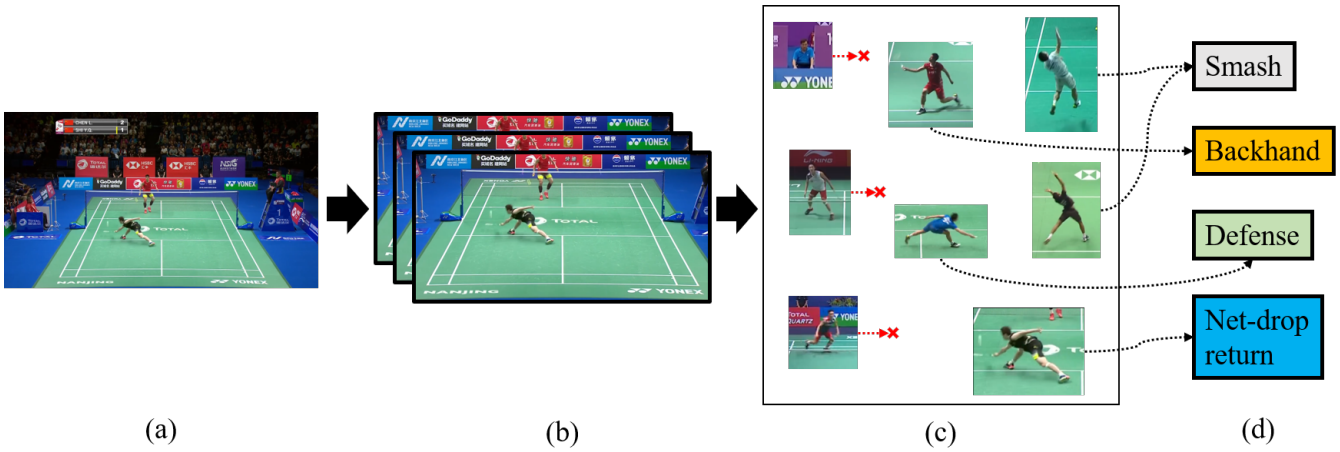

Fig. 1. This figure shows the dataset collection process. (a) YouTube Badminton matches videos were manually scrubbed and frames of interest were saved as .png files. (b) These frames were then cropped to remove the unnecessary parts of the frame, including the audience, the referee and parts surrounding the court. (c) These cropped frames are then run through TensorFlow's open source object detection API to detect humans in the frame [13]. All humans found are cropped and saved as individual .png files. This step would occasionally return other humans in the frame, such as the other player and the line judges. (d) Frames of interest were then manually sorted and labelled as one of the four shot categories. Unwanted frames were discarded.

length) and compare them to those of professional players and provide the player with recommendations to improve their performance.

The main contributions of this paper are (i) multi-class shot classifier with a trained CNN to classify shots among four categories - smash, defensive, backhand and net-drop return, (ii) automated procedure to extract player images from frames in broadcast badminton videos, (iii) modified pose estimation to return locations of all detected joints, (iv) extraction of characteristic features of shots hit by professional badminton players and (v) insight generation through comparison of characteristic features of shots hit my amateurs to those of shots hit by professional players.

II. DATA COLLECTION

The collection and pre-processing of data plays a vital role for all kinds of classification using various deep learning models. In our case, we dedicated ample amount of time and energy to build a manual dataset containing high resolution images of badminton shots from top 5 ranked male singles badminton players. The frames were extracted from videos of badminton matches uploaded on BadmintonWorld.tv YouTube channel [17]. Initially, we performed screen recording to capture video clips containing the shot of interest. This, however, resulted in degradation of video quality and was extremely slow to process. We then switched to using Adobe Premiere Pro [18] which allowed us to scrub through the videos frame by frame and save the frame of interest quickly and without any degradation in quality.

The frames were collected based on four kinds of shots which are smash shot, net-drop return shot, backhand shot and defense shot. The videos of badminton tournaments are usually captured from behind the court, therefore, showing the player playing the shot from behind or from the front.

Our manually generated dataset consists of 1097 image frames in total that were taken from scrubbing through 20 match videos. The process of data collection is shown in Fig 1. The original frames collected from the videos through scrubbing were cropped to remove unnecessary information. We then used TensorFlow open source Object Detection API to locate humans in the frame [13]. The players were padded with extra pixels, cropped and saved as individual image files. These files were then manually sorted through and classified into the correct category.

III. SHOT CLASSIFICATION

A. Problem Formulation

We wanted to build a highly accurate multiclass image classifier. The analysis of badminton shots was entirely dependent on correctly classifying the images to recognize a badminton shot. We choose Convolutional Neural Networks (CNN) to build a multi-class classifier as they perform well with data with spatial information such as images. CNNs can be trained to extract and identify useful features in images to successfully classify them into the correct category.

B. Using CNN

In this paper, we propose the use of CNN to classify different badminton shots. The models for the CNN was built using Keras [16] with a TensorFlow Backend [19]. We trained the CNN using two custom sequential models, The first model just flattened the input and fed it into two dense layers, the second model was built specifically for our application by fine-tuning the layers in the model to get the maximum prediction accuracy possible.

ReLU is the activation function used in both the models. ReLU was chosen as it is more efficient compared to the sigmoid activation function. ReLU reduces the likelihood of gradient vanishing. This is one of the reasons why ReLU can improve the learning speed of a CNN [20].

RMSprop optimizer was used to minimize the loss function for our model. RMSprop is similar to gradient descent

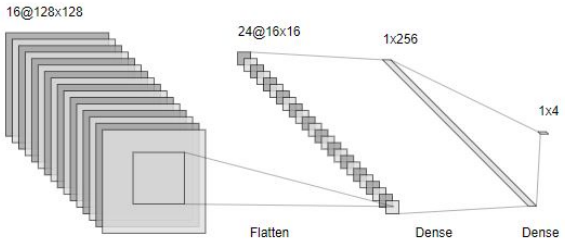

Fig. 2. Model 1 Architecture. The Architecture shows different layers of the models.

Fig. 3. Example output of pose detection on a net-drop return shot. Bones and joints of the skeleton are color coded to identify them.

algorithm and it restricts oscillations in the vertical direction. RMSprop optimizer was used for both our models.

Fig. 2 shows the architecture for Model 1. This model has a simple flatten layer and two fully connected dense layers. We use the flattening layer to insert our image into the Convolutional Neural Network.

Fig. 4 shows the architecture for Model 2. The model we propose consists of three convolutional layers, two dense layers, two max pooling layers and three dropout layers. The addition of convolution layer is to perform a simple convolution operation and pass the result to the next layer. A dense layer represents a fully connected layer. Max pooling layer was used repeatedly to reduce the number of parameters in the model and generalize the results from the convolutional filter. Max pooling layers in the model are approximately reducing the number of parameters by half. Dropout layers were added in the model to reduce overfitting by dropping randomly selected nodes. This allows the model to generalize better and reduce the out of sample error.

We used eighty percent of our data for training, validation and the remaining data for testing the Convolutional Neural Network.

The images were loaded onto the Convolutional Neural Network in a batch size of sixteen. We use thirty epochs to train the Convolutional Neural Network. An early stopping mechanism is used to stop the epochs when it detects overfitting. The validation errors are compared with the training error to detect overfitting. The batch size was one of the parameters we fine-tuned based on the training errors. We used larger batch sizes to improve the training speed of our models. With larger batch sizes we were able to generalize better on testing data.

C. Results

We compared the results of Model 1 and Model 2. In the simple Model 1, we got a maximum accuracy of 42%. Model 2 gave us a maximum accuracy of 84% after fine-tuning the model. By viewing the confusion matrix, we realized that our model was most confused between net-drop returns and backhands. It misclassified 14 backhand shots as net-drop return shots. This is likely as net-drop returns can be hit using both forehand side of the racket or backhand side of the racket. This can likely be improved by collecting more data.

IV. PLAYER POSE DETECTION

Pose Detection forms the basic step for feature extraction of valuable data. The extracted feature is used to calculate details and predict how good a shot is played. The pose detection technique used here is based on TensorFlow Open Pose model[19]. There were three pre-trained models available - Cmu, Dsconv and mobilenet. Cmu model was based on VGG pretrained network [21] and the weights were converted to Caffe format to use with TensorFlow. Dsconv has depth wise separable convolution and its architecture is similar to Cmu model. In Mobilenet model, 12 convolutional layers were used as feature-extraction layers. All these model were trained using coco dataset to detect humans. The technique used a sliding window to detect parts of human body and label them. Even though this model looked reliable it did not serve our purpose as this model failed to detect fast moving objects and also failed to recognize blurred areas in the image. To overcome the failure to detect blurred parts of the image we used the algorithm that converted Caffe [22] format of data to Keras [23] and trained these models with the new dataset generated by the Keras model [16]. This dataset helped the model to be pre-trained to detect fast moving objects as well as the images which had blurred wrist or arm movements. Fig 3 shows an example output of pose detection. All body parts are color coded for easy identification.

V. SHOT QUALITY ANALYSIS

Pose detection applied to shots played by professional players can be used to learn the correct skeleton configuration and movement to play a shot properly. For the purpose of this report, we will only be analyzing the net-drop return shot. The steps demonstrated in the following sections can be extended for analyzing other types of shots as well. The goal here is to learn the way professional players play these shots, and learn the ideal pose required to play a perfect shot. To prevent learning only a single player's style, we used data from all the players we collected data on.

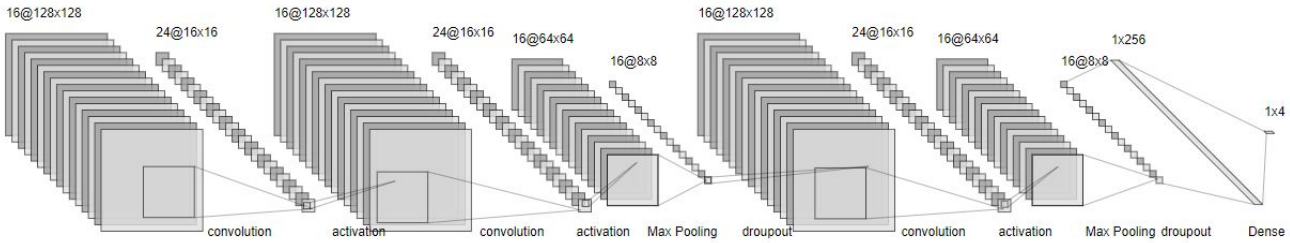

Fig. 4. Model 2 Architecture. The Architecture shows different layers of the models. The model takes in a 128x128 image as the input and classifies it as one of the 4 Badminton shots.

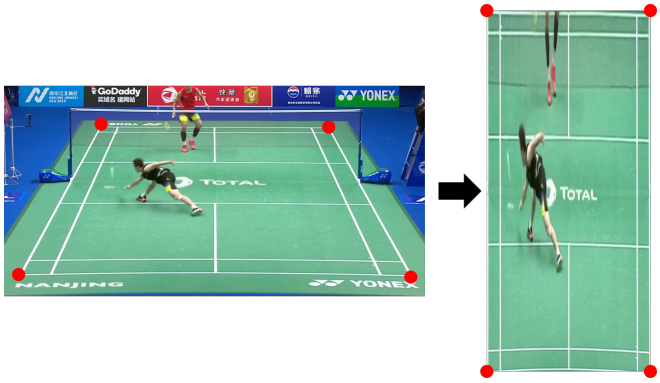

Fig. 5. Perspective is transformed using the four-corners of the badminton court. Pixels within these corners are warped to a destination image (right) with dimensions proportional to the physical dimensions of a badminton court. Pixel distance calculated in the warped image (right) corresponds to physical distance using the calculated conversion factor γ.

A. Identifying features to extract

The first step is to identify the relevant features that define a good net-drop return. According to [24], a good net-drop return has a long lunge distance and has both arms stretched over the legs providing balance and maximizing player's reach. Thus, we define the features to learn as:

1) Lunge distance: distance between the two ankles of player detected by pose detection in IV, and
2) Right elbow angle: angle between the right arm and forearm at the elbow detected by pose detection in IV.
3) Left elbow angle: angle between the left arm and forearm at the elbow detected by pose detection in IV.

Please note that for this report, we extract and learn the lunge distance, while analysis of right elbow angle and left elbow angle is left as a future extension.

B. Badminton court extraction

Various methods of detection and recognition of the court boundaries have been proposed in previous works. [10] uses a method described in [25] to detect tennis courts and modifies it to use it for badminton courts. [25] applies a Hough transform to extract court lines and then uses a combinatorial search to establish correspondence between the lines and a court model. Such a method would have worked well for the purpose of this research, however, given the shortage of time we were unable to implement it. In addition, the fact that the camera perspective does not change for matches of a single tournament, and that for this research, identifying the corners of the court manually identifying the corners of the courts for each tournament was the simpler and faster approach.

Corner points of the badminton court were then used to perform a four-point perspective transform. The destination output image size was set to 2235×1016 pixels, which is proportional to the size of a badminton court - $13.411m \times 6.096m$. This allows us to calculate the conversion factor γ as follows:

$$\gamma = \frac{\text{length of court in meters}}{\text{height in pixels}} = \frac{13.411}{2235} = 0.006 \text{ m/px} \tag{1}$$

The process of perspective transform allows us to use γ to convert pixel values to actual distance values. This is possible because the perspective transform shows us the "bird's-eye view" of the court, allowing us to map the known dimensions of the court on to an image of proportional dimensions. The perspective view is calculated using a transformation matrix that maintains the relationship between the camera image and the warped image. This matrix is called the homography matrix (H) and it transforms point (x, y) in the original image to point (x', y') through the following relationship:

$$\begin{bmatrix} \alpha x' \\ \alpha y' \\ \alpha \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

where, H is a 3×3 matrix and α is a constant for that pixel.

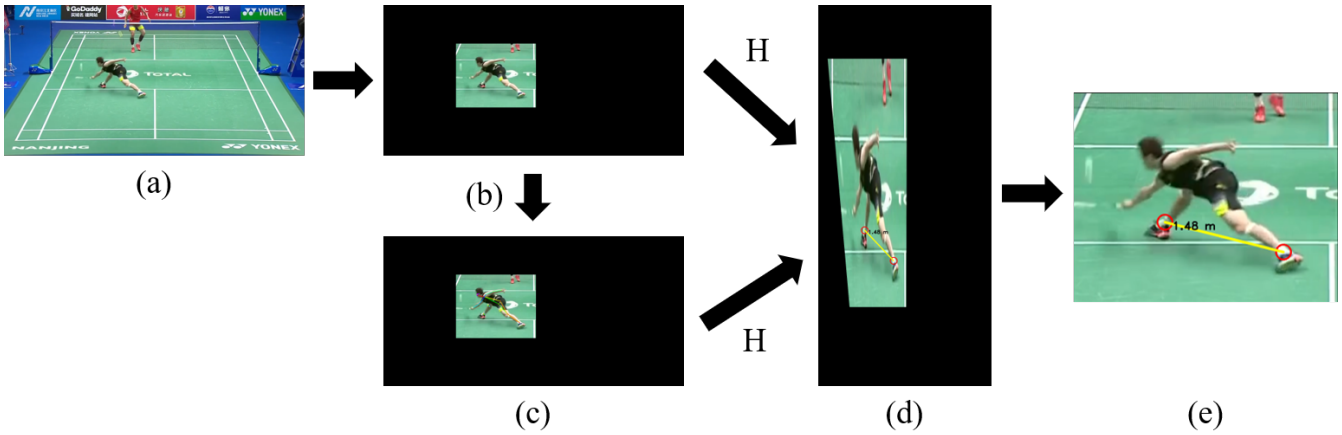

Fig. 6. This figure shows the procedure of calculating lunge lengths from camera frames. (a) Shows the cropped frames collected during the dataset generation. (b) These frames are then passed through the model for detecting humans. Pixels without the human were set to $[0, 0, 0]$ to ease the pose detection process discussed in IV, but the frame was not cropped to maintain the original dimensions. (c) Pose detection was performed. (d) Using the homography matrix H, calculated in V-B, output of (b) is transformed to match the dimensions of the court. The two ankles in (c) are transformed using H and plotted on the warped image. These coordinates are then used to calculate the distance in meters using γ. (e) Calculated distance is then plotted back to original frame (here, $1.48m$).

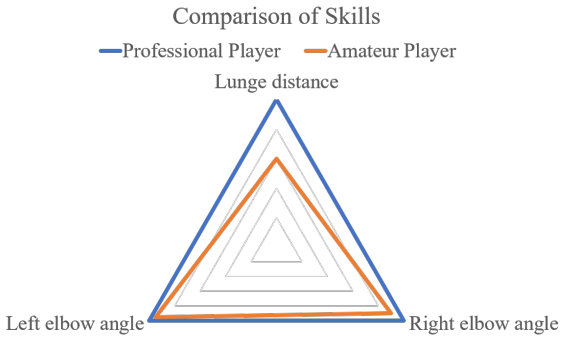

Fig. 7. Feature comparison example

C. Lunge distance calculation

Fig 6 shows the procedure to calculate the lunge distance for every frame. Please note that we only consider the shots played by the player on the near-side of the court for this analysis. This resulted in 228 frames of players playing net-drop return in the near-side of the court.

The pre-processing of frames for lunge distance calculation was significantly different from the pre-processing required for shot classification. Instead of cropping the player out of the main frame (shown in Fig 6 (a)), we blacked out, i.e, replaced the pixel RGB values with $[0, 0, 0]$, all the pixels that did not include the player playing the shot. This was done to help quicken the pose detection process, and prevent other humans' skeletons being detected. Moreover, we did not crop the player out to maintain the dimension of the original frame with the entire view of the court. This is needed as for the perspective transform (using H) to work correctly, we need the correct source points (corners of the court) in the original frame. We modified the pose detection algorithm to output a text file of all the joints and their pixel locations. Using the pixel locations for the ankles in the original perspective (shown in Fig 6 (c)), and the homography matrix H, we calculated the transformed locations of the ankles and plotted them on the image with the transformed perspective (shown in Fig 6 (d)). We then calculated the euclidean distance between the two ankles in pixels and used the conversion factor γ, to calculate the distance in meters. This distance was then plotted back on the original image. The average lunge distance calculated from these frames is **1.36 m**.

VI. EXTENSION TO SHOT TRAINING

Analyzing and learning from professional players' poses can help create models for how shots must be played. Extracted features from these shots can help with coaching of amateur players. An amateur player who may be struggling to improve their technique, can compare the features extracted from their play to those learned from professional players' play. This can guide quickly point out the best practices to them, and areas to focus and improve on.

An amateur player's shots can be analyzed using a similar framework as outlined in Fig 6. This framework will output the lunge distance and can be further expanded to output other recognized meaningful features. These features, when compared to those of models learned from professional players' play, can generate meaningful insights that can help the player improve and keep track of their progress.

VII. FUTURE EXTENSIONS

Currently the shot predictions will only predict 4 different shots played by right handed badminton players. This model can be extended by adding more shots. We can add one more layer to the CNN and by training with images of left handed badminton players we can predict left handed players shots.

The algorithm can even be extended for women badminton players. In future we can have an application that would allow users to just capture one of their badminton shot picture and by uploading the picture they could get to know how good their shots is and also get inputs which could help them to learn and get better.

VIII. CONCLUSION

In this paper, we present a system for helping amateur badminton players learn to play shots the way professional badminton players play shots. The first step was to create a multi-class shot classifier with a trained CNN to classify shots among four categories - smash, defensive, backhand and net-drop return using a dataset which was manually created for this research. We achieved accuracy of 84% in shot classification. Once the shot is classified correctly, the second step was to use pose estimation to extract characteristic features of shots hit by professional badminton players and learn from them. The third step was to repeat the process on shots hit by amateur players to extract the characteristic features and generate insights on improvement by comparing these features to those of professional players.

IX. TASK ALLOCATION

- Data collection: Rahul, Deepak
- Data pre-processing: Rahul, Deepak
- Multi-class shot classifier: Saurabh, Pratap, Deepak
- Pose estimation: Saurabh, Pratap
- Feature extraction: Deepak

X. ACKNOWLEDGMENT

We would like to thank Prof. Wencen Wu for her guidance and support throughout the semester and for providing us with an opportunity to work on a project like this through which we were able to gain ample amount of industry knowledge by learning and implementing concepts in deep learning technology.

REFERENCES

[1] Fastest badminton hit in competition (male) — Guinness World Records. [Online]. Available: http://www.guinnessworldrecords.com/world-records/fastest-badminton-hit-in-competition-(male)/

[2] Results — DAIHATSU YONEX Japan Open 2017. [Online]. Available: https://bwfbadminton.com/results/2669/daihatsu-yonex-japan-open/podium

[3] Badminton second to soccer in participation worldwide. [Online]. Available: http://www.espn.com/olympics/summer04/badminton/news/story?id=1845228

[4] PV Sindhu Says India Needs More Badminton Coaches Because Not Everyone Can Afford The Pullela Gopichand Academy. [Online]. Available: https://www.indiatimes.com/sports/pv-sindhu-says-india-needs-more-badminton-coaches-because-not-everyone-can-afford-the-pullela-gopichand-academy-331095.html

[5] Intel True View. [Online]. Available: https://www.intel.com/content/www/us/en/sports/technology/true-view.html

[6] MVP Mobile Dummies. [Online]. Available: http://www.mobilevirtualplayer.com/

[7] HomeCourt. [Online]. Available: https://www.homecourt.ai/

[8] Hawk-Eye. [Online]. Available: https://www.hawkeyeinnovations.com/sports/badminton

[9] F. Chen and H. Wang, "Automatic Linesman System for Badminton Games," 2016. [Online]. Available: https://web.stanford.edu/class/cs231a/prev_projects_2016/cs231a_final_report_Feiyu_He.pdf/

[10] W.-T. Chu and S. Situmeang, "Badminton Video Analysis Based on Spatiotemporal and Stroke Features," in *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, ser. ICMR '17. New York, NY, USA: ACM, 2017, pp. 448–451. [Online]. Available: http://doi.acm.org/10.1145/3078971.3079032

[11] S. Ramasinghe, M. Chathuramali, and R. Rodrigo, "Recognition of Badminton Strokes Using Dense Trajectories," 12 2014.

[12] A. Ghosh, S. Singh, and C. V. Jawahar, "Towards structured analysis of broadcast badminton videos," 12 2017.

[13] TensorFlow Models. [Online]. Available: https://github.com/tensorflow/models/tree/master/research/object_detection

[14] Badminton Sensor Review–Coollang Smart Badminton Sensor — BadmintonCentral. [Online]. Available: https://www.badmintoncentral.com/forums/index.php?threads/badminton-sensor-review-coollang-smart-badminton-sensor.172984/

[15] J. Lin, C. Chang, C. Wang, H. Chi, C. Yi, Y. Tseng, and C. Wang, "Design and implement a mobile badminton stroke classification system," in *2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Sept 2017, pp. 235–238.

[16] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," *CoRR*, vol. abs/1611.08050, 2016. [Online]. Available: http://arxiv.org/abs/1611.08050

[17] BadmintonWorld.tv - YouTube. [Online]. Available: https://www.youtube.com/user/bwf

[18] Buy Adobe Premiere Pro CC — Video editing and production software. [Online]. Available: https://www.adobe.com/products/premiere.html

[19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[20] H. Ide and T. Kurita, "Improvement of learning for CNN with ReLU activation by sparse regularization," 2017.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[22] Caffe — deep learning framework. [Online]. Available: http://caffe.berkeleyvision.org/

[23] Home - Keras Documentation. [Online]. Available: https://keras.io/

[24] Lunge technique — Badminton Bible. [Online]. Available: https://www.badmintonbible.com/articles/footwork/movement-elements/lunge-technique

[25] P. H. N. d. W. W. E. Dirk Farin, Susanne Krabbe, "Robust camera calibration for sport videos using court models," vol. 5307, 2003, pp. 5307 – 5307 – 12. [Online]. Available: https://doi.org/10.1117/12.526813