International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 12, No.1 (Jul-2022)

https://dx.doi.org/10.12785/ijcds/120114

Binary Heatmap Based High Speed Object Tracking in Racket Sports Using Semantic Image Segmentation

Manikandan G[1], Sayf Hussain Z[1] and Suryaa V S[1]

[1]*Department of Computer Science and Engineering, College of Engineering, Guindy, Anna University, Chennai - 25*

Received 12 Jun. 2021, Revised 9 Apr. 2022, Accepted 15 Jun. 2022, Published 1 Jul. 2022

Abstract: Ball trajectory data is a vital and important aspect when it comes to evaluating player performance and analyzing game strategies. It is a strenuous task to identify the position of a fast-moving tiny ball or cork accurately from any video. In this work, we employ image segmentation technique and propose a deep learning network consisting of both convolutional and deconvolutional networks to detect the trajectory of the cork or the ball in frames from broadcast videos. For experimental validation, we used tennis, Badminton and table tennis datasets, further the proposed model is compared with the standard state-of-the-art work. Based on the experimental results and comparisons, the proposed model provides better precision, recall, and F1score when compared to existing methods.

Keywords: Deep Learning, Computer Vision, Image Segmentation, Sports Analytics

1. Introduction

Trajectory is the path taken by an object in a given space over a particular period of time. Ball trajectory data is a vital and important aspect when it comes to evaluating player performance and analyzing game strategies. It is a strenuous task to identify the position of a fast-moving tiny ball or cork accurately from any video. The ball or cork used in racquet sports like tennis, table tennis, and badminton is small and can travel at extremely high speeds clocking hundreds of kilometers per hour, resulting in undetectable and hazy pictures. The fastest badminton smash ever recorded is at 426 km/hr which explains the magnitude of the arduous challenge posed. The other major challenge is that the ball is often occluded by the players or the net and tends to merge with the audience or any complicated background. Hence, the appearance, size, shape, and velocity of the ball changes irregularly over the frames. Video contains a large amount of information that makes processing and analytics on the data tedious. In the fields of image processing, extracting information from videos has become a popular study topic [1]. Modern games require high end quality analysis on the game and athlete training which requires enormous amounts of video data. Sophisticated and expensive cameras are used in professional sports to record high-frame rate and high-resolution videos, which are integrated with image processing to aid the match officials and also used for data collection. On the downside, these require adequate financial and technical resources which makes it infeasible for amateurs. Instead of running analysis on the entire lengthy voluminous video, analysis can be done on highlights by skipping the less fascinating parts of the video thus saving both time and money. Sports like cricket have had precise calculation of ball trajectory [2] using techniques based on background subtraction and the model generates trajectory of the ball using the Kalman filter [3]. This is attributed to the fact that the noise in the data is relatively less and the velocity of the travelling ball is comparatively low when compared with sports like tennis and badminton. Traditional background subtraction methods are ineffective at removing the vast majority of noise, and they frequently necessitate extra processes [4].

In order to simplify the process of ball tracking in racket sports by implementing image segmentation technique in which a Gaussian based kernel [5] is used to generate a heatmap for the frames. In this work, we propose a deep learning network using SegNet [6] to detect the small sized balls from broadcast videos of various racket sports in which the ball images are tiny, hazy and occasionally even invisible. The core idea behind this work is to build a semantic image segmentation model which can identify ball/cork from the provided images. The model has been trained on various frames which were segmented from broadcast videos of various racket sports like tennis, badminton and table tennis. The segmented frames were first manually labelled and the position of the ball/cork are identified. This information is used to create a binary heatmap for each training frame. This generated heatmap is used as the groundtruth (target) output for the model. The model is based on SegNet architecture which is used in

E-mail address: manikandangmsb@gmail.com; sayfhussain.10@gmail.com; surya.seran@gmail.com

Image segmentation. Three consecutive segmented frames are fed as input to the model with the target output being the generated heatmap. After the model has been trained, any suitable sport clip video can be used for analysis. The input video will be segmented into frames and three consecutive frames are fed to the model. For each input to the model, a binary heatmap is generated as the output. OpenCV is then used to detect circles on this generated heatmap using the Hough Gradient method [7]. If circles are detected, then the x and y coordinates of the detected circles are used to draw a circle (considered to be the ball) on the input image. This is repeated until all input frames are exhausted.

In this work, we propose a heatmap based deep learning network to recognize the ball in every frame and the frames are further processed and concatenated to create the output video. To explore the model extensibility, tennis, table tennis and badminton tracking are evaluated. The designed model can precisely position balls in tennis, table tennis and badminton on televised videos or videos captured without professional cameras. In this approach, the suggested model solves the problems of hazy pictures and can even identify occluded balls by studying their trajectories, and this framework may be used on other ball-based games such as baseball and golf without extending too much emphasis on financial considerations. Other than image segmentation techniques in deep learning, even sequential classifiers like Recurrent Neural Networks (RNN) and LSTMs are used in tracking the ball trajectory [8] [9].

The proposed paper flow is as follows. Section 2 provides the various related works in the field of object tracking and the proposed system model with detailed architecture design is discussed in section 3. Section 4 provides the experimental design, results and discussion of the proposed model. Further, section 5 concludes the work with future directions.

2. RELATED WORKS

This section provides various related works carried out in object tracking domain in deep learning. Yu-Chuan Huang et al [10] proposed a heatmap-based deep learning network, Tracknet, not only taught to detect the balls from a single frame, but it is also taught to remember flying patterns across multiple frames. To position the ball, TrackNet uses images with a resolution of 640 x 360 pixels to build a detection heatmap from a single frame or multiple successive frames, and it can achieve high precision even on publicly accessible videos. For this model, after application of 10-fold cross validation, the Precision value was recorded to be 95.3% while the recall was measured as 75.7%. Similarly, the F1-measure was observed as 84.3%. Archana et al [11] model can categorize ball tracking in the Australian Open with a precision of 90.64% and the Wimbledon Open with a precision of 90.32 %. To identify the ball, a logical AND operation is conducted on the produced backdrop and image the difference, after which the ball candidates are recognized and dilated using threshold values. Xinguo Yu et al [12] eliminates as many non-ball items from each frame as feasible by using sieves. The frame's ball candidate pixels are identified as the residual pixels from the frame. Candidate trajectories are generated from the candidate feature images, which present all the ball candidates of a given sequence together. All elements in a candidate trajectory are ball candidates, which is also considered to be a ball trajectory candidate. The work tries to distinguish the ball trajectories from the candidate trajectories on the basis of the confidence index of each candidate trajectory and from the understanding that there is at most one ball in each frame. Xiangzeng Zhou et al [13] have proposed a two-layered data association (TLDA) approach for tennis ball tracking. The approach to handle multiple object tracking is proposed as an extension in the work. Currently, the methodology uses graph approaches to solve the data association problem by attempting to reduce the overall cost.

Hnin Myint et al [14] offers an efficient and effective detection and tracking approach that uses stereo vision to track a table tennis ball from stereo films recorded by two low-cost single-view cameras. The suggested system uses the segmentation-detection-tracking methodology as well, but with a focus on improving segmentation and detection. A stereo vision system with an inter-view correcting mechanism is also proposed to overcome the occlusion problem. Xinchao Wang et al [15] tracks players and decides who is in possession of the ball and then utilizes players' trajectories to achieve reliable ball tracking. Yan et al [16] suggested a method for foreground blob classification that uses numerous visual properties, one of which is a novel feature for recognizing elliptical objects. The tennis candidates are tracked using a particle filter. The posterior density is used to draw samples. The trajectory is then refined using smoothing and observation origin identification to improve tracking accuracy. For moving object blob identification, Kamble et al [17] 2-stage buffer median filtering backdrop modelling is utilized, and the system can predict the ball position for a feasible duration after any player occludes the ball.

3. PROPOSED OBJECT TRACKING SYSTEM

In this section, initially we have provided the basic concepts of semantic image segmentation and then we explain the algorithmic description with explanation of the proposed heat map-based object tracking framework.

A. Semantic Image Segmentation

A collection of different pixels is represented as an image. Pixels that have similar attributes are grouped together and labelled as specific regions according to what's shown in the image using image segmentation techniques. That is, a pixel wise mask is created for the objects present in the picture. Image segmentation techniques are currently utilized in several domains such as Traffic control systems [18], self-driving cars [19], medical segmentation [20] and also for object identification tasks such as locating

and identifying objects in the satellite images [21]. Every pixel in the images is assigned to a particular class, either the background or the objects to be recognized. Image segmentation techniques can be broadly classified into two types. When all the pixels that belong to a particular class are represented by a single-colored mask it is an example of semantic image segmentation. When the different instances of the same object in an image are represented by masks of different colors it is an example of Instance segmentation [22].

The goal of semantic image segmentation (dense predictions) is to assign a class of what is being represented by each pixel in an image. The main idea is to only consider the category of each pixel, not to differentiate between different instances of a particular class or category. The employment of an encoder/decoder structure is a popular strategy for picture segmentation models, where the encoder part of the model diminishes the resolution of the input image to create an encoded feature map which is highly effective at discriminating between classes. The Decoder part of the model up-samples the intermediate representation into a segmentation map of full-resolution. Long et al [23] suggested the notion of employing an end-to-end trained FCN network for picture segmentation in 2014. The authors proposed adaptation of image classification networks as the encoder module, inclusive of a decoder module that was built with transpose convolutional layers for the purposes of up-sampling the coarse feature maps to a full-resolution segmentation map.

The work uses the semantic image segmentation technique to help distinguish the ball or the cork from the noisy background. It also serves the purpose of target output. In case, if we try to feed the model without segmenting and identifying the ball, the model is highly likely to be confused with the plethora of objects around the ball and would result in uninspiring results. The segmented images act as a target output towards which the model actively trains. The goal of the model would be to replicate the segmented images for newly fed frames from which we can identify the ball location using Hough gradient met.

B. Proposed Framework

Fig 1 shows the entire workflow for the work. The objective of the work is to track the trajectory of the ball from any given broadcast video of racket sports such as Tennis, Table Tennis and Badminton. For this purpose, a suitable dataset of broadcast videos is collected, relevant video portions are identified and segmented to generate frames. The frames are then annotated to obtain the ball center coordinates for each frame. A Heatmap is generated by applying a Gaussian Kernel centered around the ball center for each frame which acts as the ground truth image. A model based on SegNet is designed which takes in a triplet of consecutive frames as the input to predict the heatmap.

When analysis has to be performed on a new input video, the input video is segmented into frames and 3 consecutive frames are fed to the model as the input. The model predicts a heatmap which is processed to convert it into a black and white binary image by applying a threshold of 127. The work tries to model a binary heatmap which requires only two values either 0 or 255. Hence, we set the threshold as 127 as it is the median value. All the values below 127 is considered as 0 and above it as 255. Hough gradient method is used to detect circles from the predicted heatmap and if circles are found, the center coordinates of the circle are used to draw a circle on the original input frame. This frame is then appended to the output video path. The whole process is repeated until all frames are exhausted.

C. Preparation of the Dataset

The dataset consists of broadcast videos from racket sports such as tennis, badminton and table tennis. The videos for tennis are from various tennis games occurring between 2014 and 2017 captured at a resolution of 1280 x 720 pixels at 30 fps. The videos of Table Tennis are obtained from Liga Pro Men's games which were recorded at a resolution of 1920 x 1080 pixels at 120 fps. The videos for Badminton are obtained from different badminton games which were all recorded at 1920 x 1080 pixels at 30 fps. Irrelevant sections from the videos where the ball is not in play are identified and truncated to generate game clips for each video. The clips are all segmented to generate frames. The frames are all manually annotated with the visibility class of the ball and the ball position to generate a labelled dataset for each clip. The visibility class of the ball can take one of the following values: 0 for no ball in frame, 1 for easily identifiable ball in frame, 2 for hard identification of the ball in frame where the ball merges with the background and 3 for occluded ball in frame where the ball cannot be discerned due to obstruction by other objects. For visibility class 2 and 3 the position of the ball in the frame can be identified from the ball position in the neighboring frames. The labelled dataset file contains records for the frame name which is the frame number within the clip, the visibility class for the frame and the X and Y coordinate of the ball position. Fig 2 shows the segmented frames for each sport category. Table 1 shows the Visibility Class wise distribution for frames from each sport category. For Table Tennis, the lateral view of the game is recorded and hence we have 0 frames where the ball is occluded by the player (Visibility Class 3). Furthermore, due to the absence of background noise in the training data for Table Tennis, we have 0 frames belonging to the hard identification class (Visibility Class 2).

D. Image Segmentation and Training the model

The proposed model is based on SegNet, an image segmentation model in which a Convolutional Neural Network is stacked with a Deconvolutional Neural Network. The data format for the model has been set as channels first i.e, at any stage in the model the tensor is represented as (number of channels, height, width) instead of (height, width, number

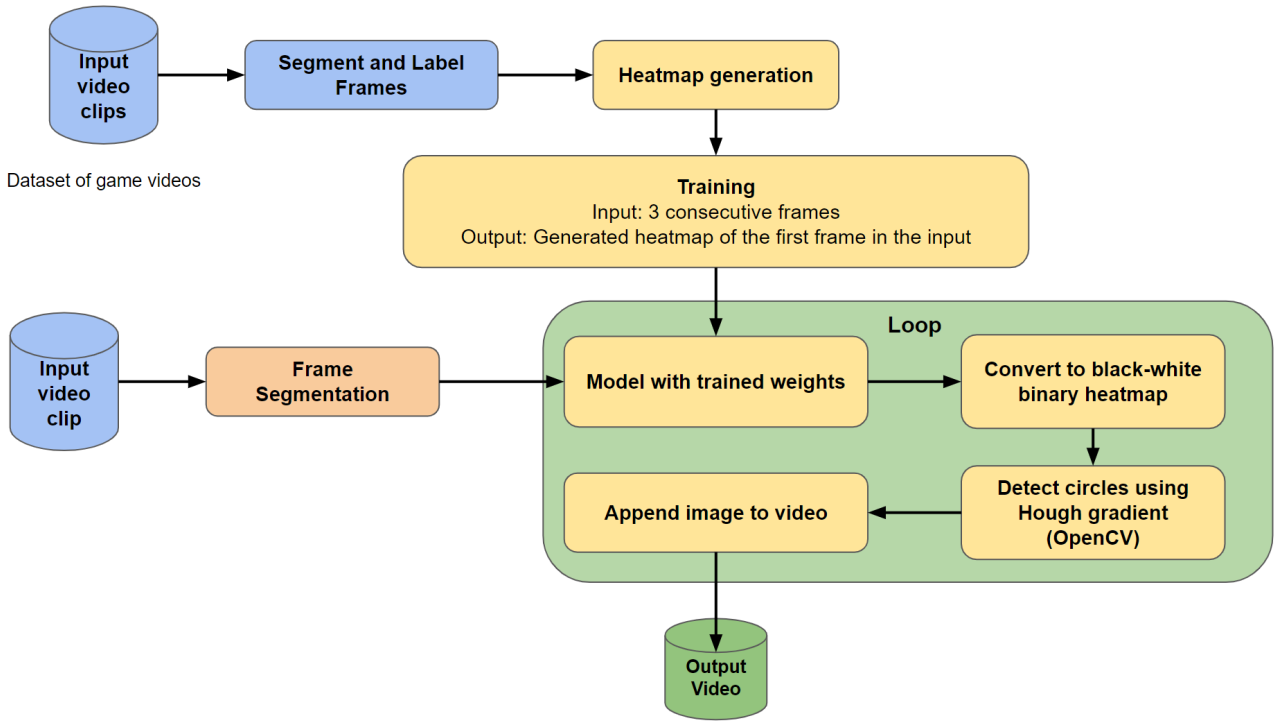


Figure 1. Heatmap based Object Tracking Framework

TABLE I. Sport-wise Visibility class split-up

Sport	VC 0	VC 1	VC 2	VC 3	Total frames
Tennis	718	20,202	2,336	8	23,264
TableTennis	183	12,816	-	-	12,999
Badminton	94	682	209	65	1,050

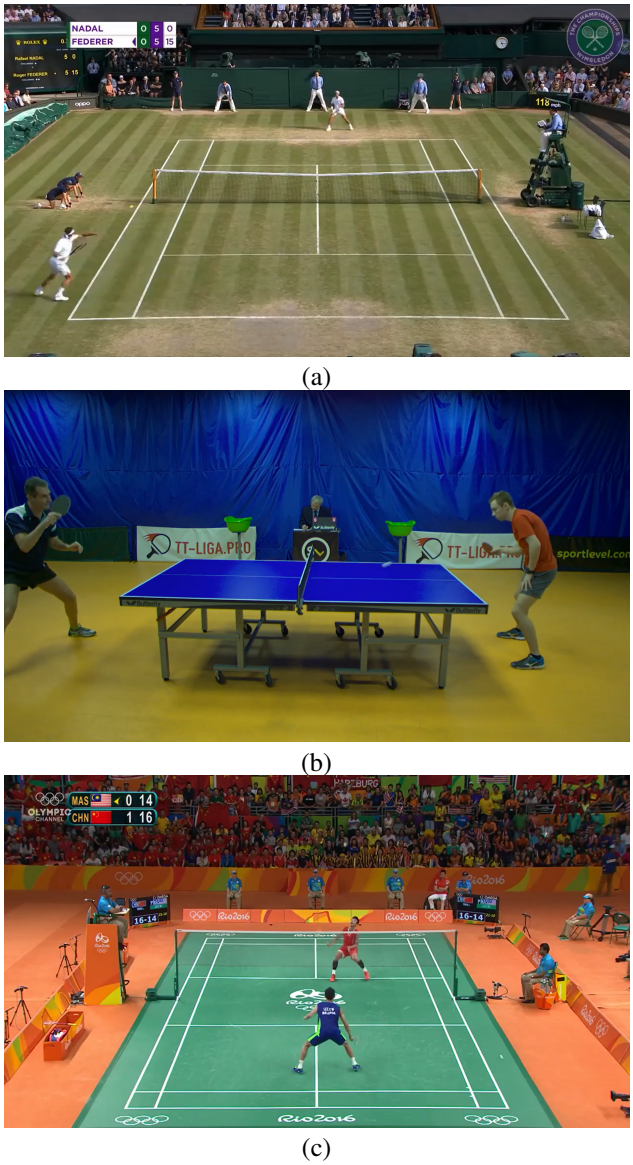
of channels). Since the model uses channels first approach the model has to be trained using a GPU.

The model takes 3 consecutive frames of size 360 x 640 as input to generate the heatmap. Since 3 images are concatenated to generate the input, the input dimensions for the model would be 9 x 360 x 640. As shown in Fig. 3 the first 13 layers of the model (Convolutional Neural Network) correspond to the down-sampling part of the model which creates an abstract representation of the input image and the last 13 layers (Deconvolutional Neural Network) correspond to the up-sampling part of the model where the abstract representation is up-sampled to make their spatial dimensions equal to the input image.

Each layer contains a varying number of filters of size 3 x 3, the input to each layer was padded so as to preserve the feature size. Layers 3, 6 and 10 are the max pooling layer which decreases the feature map dimensions across the model. Layers 17, 20 and 23 contain an up-sampling layer. The final convolutional layer predicts a feature map of dimensions 256 x 360 x 640. Additionally, a softmax layer is used to calculate the probability distribution for all pixels over all 256 channels. For any pixel p(i,j), the heatmap value h(i,j) is chosen as the value which corresponds to the highest probability across all 256 channels to finally generate an image of dimensions 1 x 360 x 640.

Each layer contains a varying number of filters of size 3 x 3, the input to each layer was padded so as to preserve the feature size. Layers 3, 6 and 10 are the max pooling layer which decreases the feature map dimensions across the model. Layers 17, 20 and 23 contain an up-sampling layer. The final convolutional layer predicts a feature map of dimensions 256 x 360 x 640. Additionally, a softmax layer is used to calculate the probability distribution for all pixels over all 256 channels. For any pixel p(i,j), the heatmap value h(i,j) is chosen as the value which corresponds to the highest probability across all 256 channels to finally generate an image of dimensions 1 x 360 x 640.

The information from the Labelled dataset for each clip is then used to generate the ground truth Heatmap. These are scaled 2D gaussian distributions for each frame centered at the ball center. For each frame the information from the labelled dataset is fed to the gaussian distribution function to create the ground truth black and white image where the ball is being highlighted in white and the background is black. Let the ball's center be (x1, y1) and $\sigma 2$ the variance, then the heatmap function is represented by



(a)

(b)

(c)

Figure 2. Segmented Frames (a) Tennis (b) Table Tennis (c) Badminton

A Gaussian Distribution Grid of size 40 x 40 centered around (20,20) is generated. The variance in the Gaussian distribution function is used to represent the size of the ball in the generated heat map which is set to 10. For each point in the grid its gaussian distribution value is calculated which is a value ranging from 0 to 1 depending upon to points affinity to the center. This value is multiplied by 255 to project its corresponding value on the color scale. The Gaussian distribution is shown in Fig 4.

In Groundtruth generator, for each row in the labelled dataset file, if the visibility is 0 then an image of the same dimensions as the input frame with all its pixel values as 0 is generated. For any other visibility, first an image of the same dimensions as the input frame with all pixel values as 0 is generated and then the ball center values from the labelled dataset file are obtained.

TABLE II. Learning Parameters

Parameter	Value
Optimizer	Adadelta
Learning Rate	0.01
Loss Function	Categorical Cross entropy
Batch size	2
Steps per epoch	200

A Gaussian Distribution grid of size 40 x 40 centered around (20,20) as shown in Fig 4 is superimposed on the generated black frame by aligning the ball center with the grid center.

After the ground truth images are generated, a new dataset file is created which contains a triplet of training images and their corresponding ground truth image. The input triplet consists of three consecutive frames, kth frame, (k-1)th frame and (k-2)th frame along with the kth ground truth heatmap. Three distinct models are created for the three rackets sports considered- Tennis, Table Tennis and Badminton. All three models have the same underlying architecture and differ from each other only in the dataset used for training the model. Table 2 shows the training parameters used while training all the 3 models.

Algorithm : Gaussian Distribution Grid

1. Set the size of the Gaussian Kernel block to 20 to create a 40 x 40 grid centered around (20,20).
2. Variance = 10
3. **for each** point in the grid (x,y) use the Gaussian kernel formula as

3.1 $g[x][y] = e^{-(((x^2-20^2)+(y^2-20^2))/(2*variance))}$

3.2 $g[x][y] = g[x][y] * 255$

4.**endfor**

5.*ReturnthegeneratedGaussianKernel*.

E. Execution

When an input video is given for analysis, the input video is segmented as frames depending on fps value. Three consecutive frames are concatenated, reshaped and then fed into the model to predict the heatmap. Three consecutive frames are fed to the model so that temporal features of the game are also taken into consideration. In some frames the ball might look blurry due to the high velocity at which it moves, by concatenating multiple consecutive frames we overcome this issue. The first 2 frames of the input video cannot be used for prediction as the first 2 frames cannot form triplets that can be fed to the model, So the first 2 frames are copied without any modification to the output video path. From the 3rd frame until all frames are exhausted the current frame is read, concatenated with

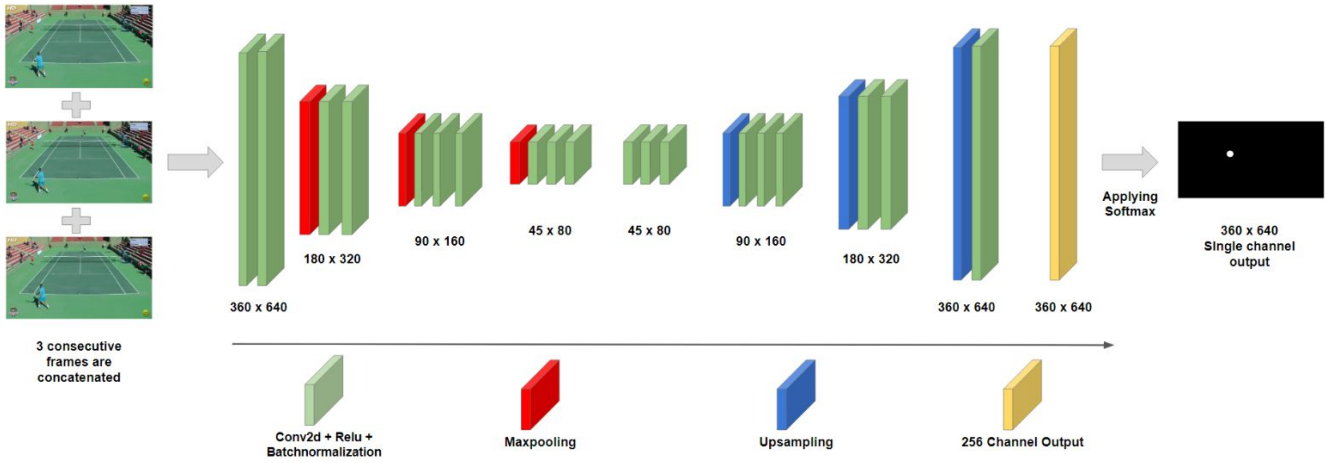


Figure 3. Proposed Deep Learning Framework

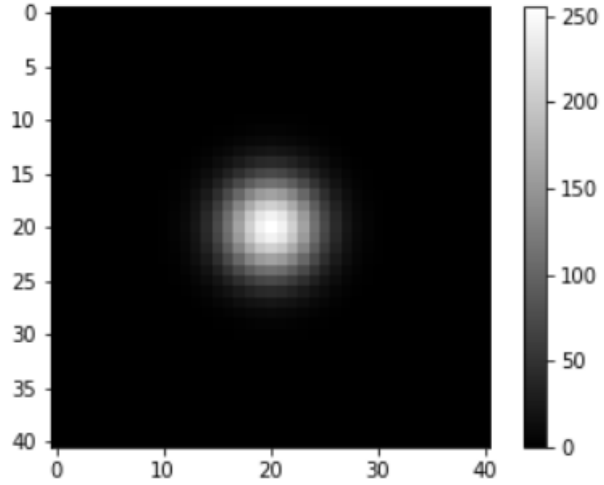


Figure 4. Gaussian Kernel centered at (20,20)visualized on 40x40 grid

the previous 2 frames and resized. The orientation of the concatenated frames is changed so as to make it channels first and it is then fed to the model for prediction. The generated heatmap is then reshaped to convert it back to channels last orientation. This heatmap is then processed pixel-wise to convert it into a grayscale heatmap. If the pixel reaches the threshold value of 127, then the pixel's intensity is set as 255 otherwise the pixel's intensity is assigned as 0. The generated heatmap is then resized to the original input frame dimensions. Hough Gradient method is used to identify and locate circles from the black-white binary heatmap. If circles are identified, then the center coordinates of the circle are used to draw a circle on the input frame. The information about the previous 10 frames ball location is used to draw circles on the current input frame so as to form a trajectory of ball movement across multiple frames. The frame is then added to the output video path. If no circle is identified, then the ball location from the previous 10 frames alone is used to draw circles on the current input frame and no new ball location is added. The input frame is then appended to the output video path. This process is repeated until all input frames are exhausted.

Algorithm : Ball Tracking

Input: Video for analysis.
Output: Video with the trajectory of the ball identified.

1. The input video is segmented into frames.
2. **for each** frame from the 2nd frame:
2.1 Three consecutive frames- current kth frame, (k-1)th frame and (k-2)th frame are concatenated.
2.2 Concatenated frames (channel first data format) fed as input to the model.
2.3 Binary Heatmap predicted as output.
2.4 Hough Gradients employed to detect position of the ball from heatmap.
2.5 **If** circles detected:
2.5.1 Draw circles on the original input frame with the detected circle(s) coordinates.
2.5.2 **If** (current frame -10) = 10: Ball location from previous 10 frames is marked on input frame.
2.5.3 **endif**
2.6 **endif**
2.7 Append input frame to the output video path.
3. **endfor**
4. The entire output video is generated once all input frames are exhausted

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

Table 3 shows the training time for the different models generated. The Tennis, Table tennis and badminton model have been trained for 500, 260 and 220 epochs respectively. The weights obtained by training the tennis model for 500 epochs are used as the starting weights to train the badminton model. The specifications of the machine on which the model was trained are a machine with intel core

TABLE III. Training Time

Model	Number of epochs	Training Time
Tennis	500	100 hours 39 minutes
Table Tennis	260	54 hours 36 minutes
Badminton	220	45 hours 29 minutes

TABLE IV. Complete Results

Sport	epochs	Precision	Recall
Tennis	500	96.12%	85.62%
Table Tennis	155	99.31%	96.68%
Badminton	220	70.14%	69.48%

i5 10th gen with 16GB of RAM and an Nvidia Tesla P100 GPU.

A. *Result Analysis*

Videos which were not a part of the dataset were downloaded and the model was used for analysis on those videos. The model has accurately identified the ball trajectory position from the videos and generated the analyzed video. Screenshots of the input and output videos are added in Fig 5, Fig 6 and Fig 7 to better illustrate the model.

The images in Fig 5 show the before and after result of execution for sample frames. Fig 5 (a) is the input to the model and Fig 5 (b) is the output respectively. We can notice that the ball trajectory has been traced with the previous 10 frame position being marked. We can infer the ball's dip and angle of travel, and we can see them in frames when the ball is obscured, hazy, or distorted. Even though the ball size is imperceptible, the trajectory can be used to understand the movement.

The images in Fig 6 show the before and after result of execution for sample frames. Fig 6(a) is the input to the model and Fig 6(b) is the output respectively.

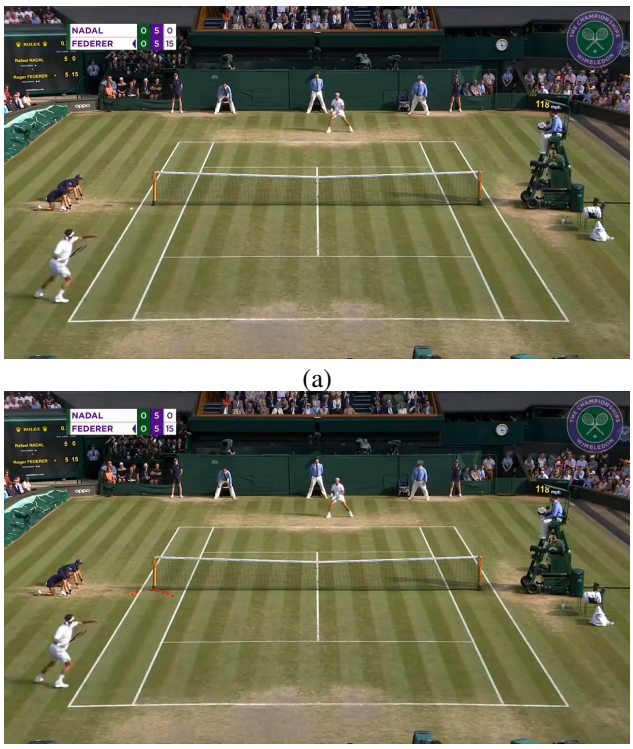
The images in Fig 7 show the before and after result of execution for sample frames. Fig 7(a)is the input to the model and Fig 7(b) is the output respectively. The cork travels at very high speed in badminton and the trajectory of the cork has still been traced by the model with significant precision.

From Fig 8, we can infer the precision and recall for the model's prediction on the test data for tennis, table tennis and badminton. For tennis, the best precision and recall are 96.12% and 85.623% respectively. The model for tennis was trained for a total of 400 epochs. The model for table tennis performed exceptionally well with a precision and recall of 99.319% and 96.688% respectively and was trained for 260 epochs over the training data. The model for badminton was trained on the already trained tennis model for 220 epochs and achieved a precision of 70.14% and recall of 69.48%. Table 4 shows the best results for the 3 models generated. So, the initial weights for badminton model training were not random, it was from the fully trained tennis model. The relatively lower performance in badminton can be caused by the fact that the dataset for badminton was limited and the cork speed was relatively higher than in other sports like tennis and table tennis. The cork speed in badminton occasionally reaches above 200 mph compared to the average speed of tennis serve ranging from 95 mph to 130 mph.

TABLE V. Tennis Model results compared with related works

Model	Precision	Recall	F1-measure
TrackNet[7]	95.3%	75.7%	84.3%
Archana's [8]	92.5%	74.5%	82.5%
Yan's [13]	88.4%	53.82%	66.91%
Zhou's [10]	84.39%	75.81%	79.87%
Proposed Model	96.12%	85.62%	90.568%

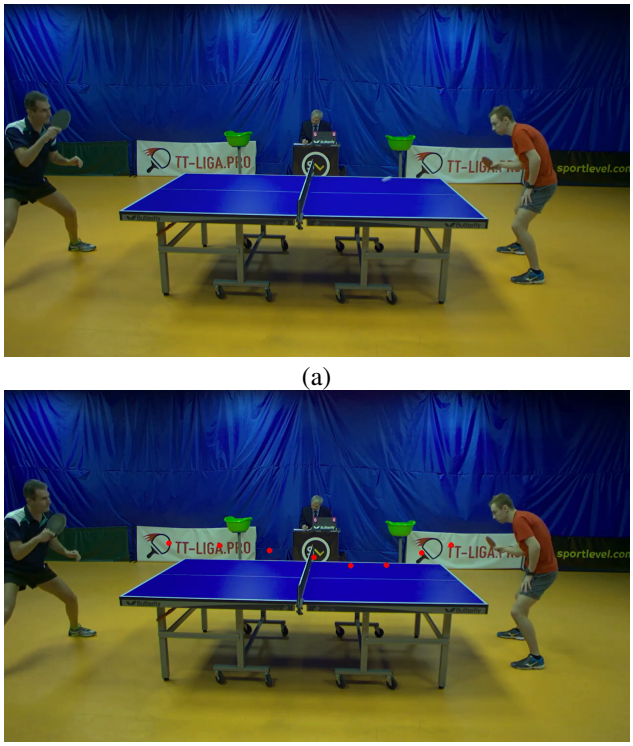
From Fig 9, we can also notice the positioning error in the case of tennis. The position error helps us understand the error in predicting the ball position with respect to the actual position of the ball. The percentage in the y-axis is the total percentage of the frames with a particular positioning error. The graph has been plotted for positioning error up to 5% and the remaining error has been accounted for together.



(a)

(b)

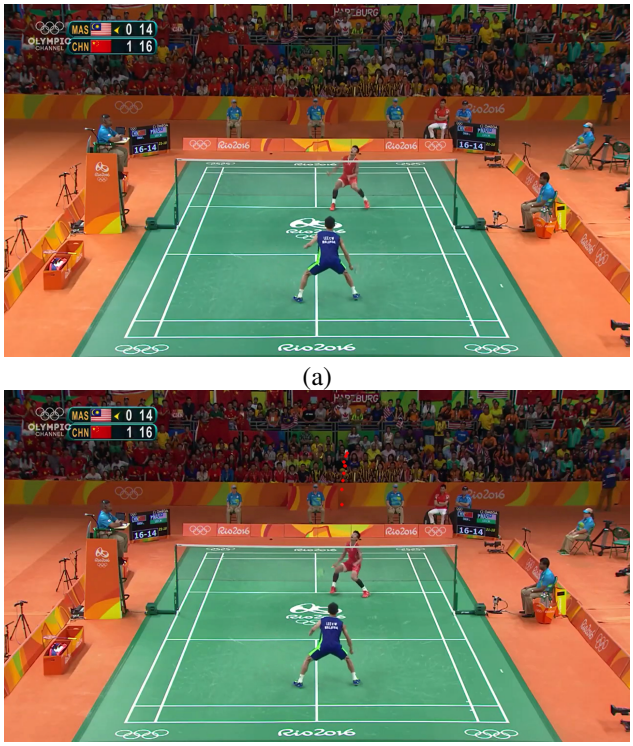
Figure 5. Input and Output Frames for different sports



(a)

(b)

Figure 6. Gaussian Kernel centered at (20,20)visualized on 40x40 grid



(a)

(b)

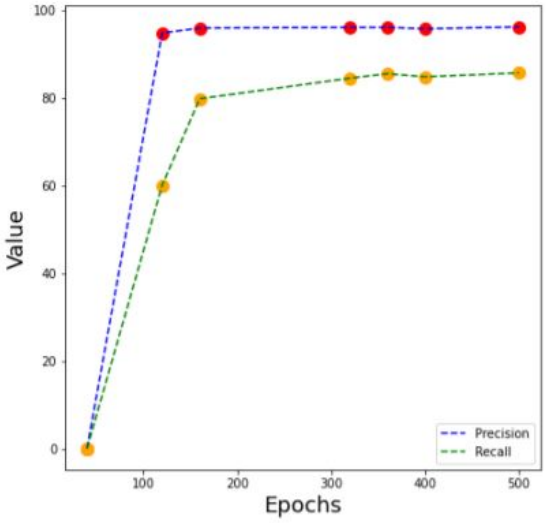
Figure 7. Gaussian Kernel centered at (20,20)visualized on 40x40 grid

B. Comparison with related works

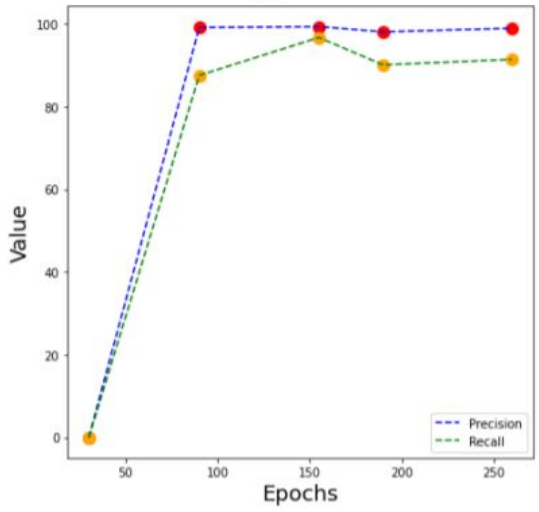
Observing Table 5, we infer that the proposed work outperforms other related work in the domain. TrackNet [7] observed a value of 95.3% for precision, 75.7% for recall and 84.3% F1-measure. These values have been calculated for a three-frame input model for the same dataset. Archana et al [8] has proposed a model with a precision of 92.5%, recall of 74.5% and f1-measure of 82.5%. Yan et al [13] model performed with a precision of 88.4%, recall of 53.82% and f1-measure of 66.91%. Zhou et al [10] model performed with a precision of 84.29%, recall of 75.81% and 79.87%. In the work done in table tennis, Hnin Mynit et al [11] have observed an accuracy of 91% in Sequence 3 background which is similar to the one used in our dataset. Our work outperforms the observed values considerably with 99.319% precision and 96.688% recall. Taking into consideration badminton, TrackNet [7] performs with precision of 85%, recall of 57.7% and f1-measure of 68.7% on the test data. The proposed work performs with 65.84% precision and 62.44% recall.

5. CONCLUSION

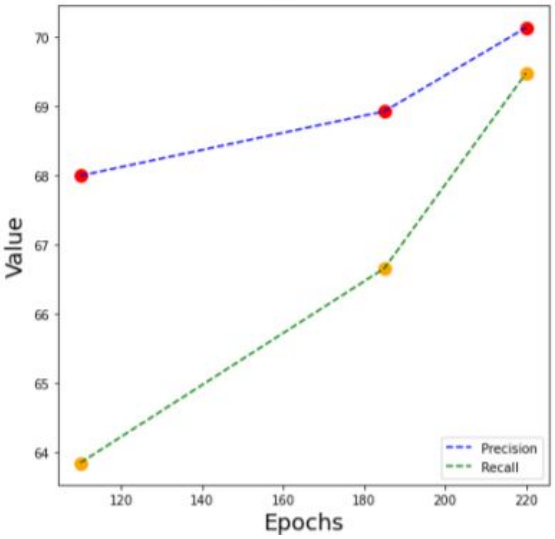
This paper proposes a heatmap-based deep learning network that accurately positions tiny fast-moving objects such as balls from broadcast videos of various racket sports. The model was built to take consecutive frames as input to improve its capacity to discern flight patterns of fast-moving objects. By employing the concept of merging adjacent frames, reliable predictions can be made on televised sports footage, without a high frame rate or quality, considerably lowering the expenses of recording and analyzing high-quality videos. The model performs with 96.12% precision and 85.623% recall in tennis in the proposed extensive method. Table tennis model performed better than badminton, with 99.319 % precision and 96.688 % recall, compared to 70.14% precision and 69.48% recall in badminton. More data can be utilized to train the badminton model to boost its performance even further. Other image segmentation models [24] [25]might be utilized as an extension of the work to compare performance and further examine the model. As the number of consecutive frames concatenated and used as input increases, the input size increases drastically with the trainable parameters. This in turn requires more RAM, computing space and also computational time. Hence, the work tried to maximize the frames fed to decrease temporal noise within the technical constraints. So, the model could be fed with more frames and checked for optimality in case of resource availability.



(a)

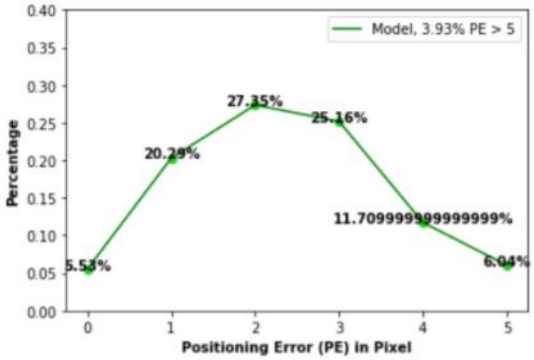


(b)



(c)

Figure 8. (a) Tennis, (b) Table Tennis, (c) Badminton



(a)

Figure 9. Positioning Error for Tennis Model at the best epoch value

References

[1] J. R. Wang and N. Parameswaran, "Analyzing tennis tactics from broadcasting tennis video clips," in *11th International Multimedia Modelling Conference*. IEEE, 2005, pp. 102–106.

[2] G. Singh, I. A. Mohammed, and S. Sasi, "Real-time boundary detection for cricket game," in *Proceedings of the 3rd Australasian Conference on Interactive Entertainment*, ser. IE '06. Murdoch, AUS: Murdoch University, 2006, p. 23–27.

[3] K. Saho, "Kalman filter for moving object tracking: Performance analysis and filter design," in *Kalman Filters*, G. L. de Oliveira Serra, Ed. Rijeka: IntechOpen, 2018, ch. 12. [Online]. Available: https://doi.org/10.5772/intechopen.71731

[4] J. Mao, D. Mould, and S. Subramanian, "Background subtraction for realtime tracking of a tennis ball." in *VISAPP (2)*. Citeseer, 2007, pp. 427–434.

[5] Z. Wang, Y. Zhou, F. Wang, S. Wang, and Z. Xu, "Sdgh-net: Ship detection in optical remote sensing images based on gaussian heatmap regression," *Remote Sensing*, vol. 13, no. 3, p. 499, 2021.

[6] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[7] T. D'Orazio, C. Guaragnella, M. Leo, and A. Distante, "A new algorithm for ball recognition using circle hough transform and neural classifier," *Pattern recognition*, vol. 37, no. 3, pp. 393–408, 2004.

[8] R. Shah and R. Romijnders, "Applying deep learning to basketball trajectories," *arXiv preprint arXiv:1608.03793*, 2016.

[9] F. Qiao, "Application of deep learning in automatic detection of technical and tactical indicators of table tennis," *PLOS ONE*, vol. 16, no. 3, pp. 1–16, 03 2021. [Online]. Available: https://doi.org/10.1371/journal.pone.0245259

[10] Y.-C. Huang, I.-N. Liao, C.-H. Chen, T.-U. İk, and W.-C. Peng, "Tracknet: a deep learning network for tracking high-speed and tiny objects in sports applications," in *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2019, pp. 1–8.

[11] M. Archana and M. K. Geetha, "Object detection and tracking based on trajectory in broadcast tennis video," *Procedia Computer Science*, vol. 58, pp. 225–232, 2015.

[12] X. Yu, C.-H. Sim, J. R. Wang, and L. F. Cheong, "A trajectory-based ball detection and tracking algorithm in broadcast tennis video," in *2004 International Conference on Image Processing, 2004. ICIP'04.*, vol. 2. IEEE, 2004, pp. 1049–1052.

[13] X. Zhou, L. Xie, Q. Huang, S. J. Cox, and Y. Zhang, "Tennis ball tracking using a two-layered data association approach," *IEEE Transactions on Multimedia*, vol. 17, no. 2, pp. 145–156, 2015.

[14] H. Myint, P. Wong, L. Dooley, and A. Hopgood, "Tracking a table tennis ball for umpiring purposes," in *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, 2015, pp. 170–173.

[15] X. Wang, V. Ablavsky, H. B. Shitrit, and P. Fua, "Take your eyes off the ball: Improving ball-tracking by focusing on team play," *Computer Vision and Image Understanding*, vol. 119, pp. 102–115, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314213002300

[16] F. Yan and J. Kittler, "A tennis ball tracking algorithm for automatic annotation of tennis match," *British Machine Vision Conference*, 01 2005.

[17] P. Kamble, A. Keskar, and K. Bhurchandi, "A deep learning ball tracking system in soccer videos," *Opto-Electronics Review*, vol. 27, no. 1, pp. 58–69, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S123034021830146X

[18] L. Li, B. Qian, J. Lian, W. Zheng, and Y. Zhou, "Traffic scene segmentation based on rgb-d image and deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1664–1669, 2018.

[19] Çağrı Kaymak and A. Uçar, "A brief survey and an application of semantic image segmentation for autonomous driving," 2018.

[20] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision (3DV)*, 2016, pp. 565–571.

[21] V. Khryashchev, L. Ivanovsky, V. Pavlov, A. Ostrovskaya, and A. Rubtsov, "Comparison of different convolutional neural network architectures for satellite image segmentation," in *2018 23rd Conference of Open Innovations Association (FRUCT)*, 2018, pp. 172–179.

[22] Y. Xie, H. Lu, J. Zhang, C. Shen, and Y. Xia, "Deep segmentation-emendation model for gland instance segmentation," 10 2019, pp. 469–477.

[23] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[24] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[25] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," 2017. [Online]. Available: https://arxiv.org/abs/1703.06870

Manikandan Govindaraju received his degree M.Tech. Information Technology in College of Engineering Guindy, Anna University. Currently, he is doing research in the field of big data Analytics in the department of Information Science and Technology and working as a Teaching Faculty in the Department of Computer Science and Engineering, College of Engineering, Anna University, Chennai. He has authored a few international journals and conferences papers and four book chapters. His current research field includes analyzing high dimensional datasets using computational intelligence and machine learning algorithms. Also, his research area focuses upon the use of Machine Learning, Image Processing, Statistical Analytics and Big Data Analytics.

Sayf Hussain Z is currently pursuing his BE in Computer Science and Engineering from College of Engineering, Guindy, Anna University, Chennai, India. His areas of interest include Machine learning, Deep Learning and Computer Vision.

Suryaa V S is currently pursuing his BE in Computer Science and Engineering from College of Engineering, Guindy, Anna University, Chennai, India. His areas of interest include Big Data Analytics and Computer Vision.