Bowling action recognition using computer vision

Dhruv Shindhe S[1], Tejas S Koundinya[2], Subhas B S[3] and S N Omkar[4][0000-0002-0806-8339]

[1]ARTPARK, Indian Institute of Science, Bengaluru, India
[2]BNMIT dept. of Electronics and communication, Bangalore, India
[3]RGIT dept. of Computer Science, Bangalore, India
[4]IISc dept. of Aerospace, Bangalore, India
omkar@iisc.ac.in

Abstract. In the realm of computer vision with respect to cricket domain, object tracking with precise pose prediction has proven to be difficult and hence less work is found in this region. Object detection is different from such applications, as used for identifying the type of object present, while the task of tracking and identifying the unique identity of the specific object which is detected is one of the main objectives. Object tracking technology has a wide spectrum of applications in the sports industry and can be used to identify different types of athletes based on their unique characteristics of action performed while bowling. This study focuses on identifying cricket players and distinguishing between bowling and non-bowling actions based on the various poses they adopt while playing the game. Yolo V3 is the object detection system, here it is used to detect cricket players while MoveNet lightning is used for the detection of keypoints of the individual. The CNN acts as a classification model, which is designed for the classification of action, which is identifying each athletes pose and enabling more accurate classification of bowling or non-bowling action.

Keywords: Move-Net Lightning, Yolo v3, Bowling and Non-Bowling action, Single-Pose Detection, Convolutional Neural Network (CNN)

1 Introduction

Video footage can be analyzed w.r.t each frame, as they add up to form a video more information about how things change over time can be achieved with this preprocessing technique. When action recognition is sought, this could prove to be crucial and would demand more store space and computational capacity than simple object detection. Computer vision with Object tracking has attracted a lot of research due to its numerous uses, including traffic flow monitoring, robotic vision, autonomous vehicle control, medical diagnostics, and action identification.

When it comes to action detection in sports footage, some actions can be better represented with knowledge of the object's trajectory. Tracking objects is done to execute valuable information such as object extraction, object recognition, and tracking, or to offer information about the activity that has carried out.

There are many variables that affect how well tracking works and what parameters affect the trackers robustness, including changes in lighting, rotation w.r.t out-plane and in-plane, abrupt changes in object trajectory, objects out-of-view, deformation, and motion blur. All of the aforementioned issues should typically be resolved co currently, which increases complexity and provides a general idea of the difficulties a tracker must handle it.

The aforementioned problems are too complex for computer vision to solve, while others, such as illumination and blur, are addressed by hardware that is always getting better. Occlusion, scale, and even previously mentioned problems like out-of-plane and out-of-view rotation could be handled by employing many camera sources from various different point of view, but the goal of this research is to identify a solution using just one camera. The issues, which writers have just partially examined, are mostly connected to the size and occlusion of the tracked objects. The depth dimension can affect item size by making it too small to be detected or too huge, which then enters the domain of occlusion, obscuring other intriguing objects and making them difficult to track.

Finding the boundaries of the desired object is the first step in the tracking process. Although this can be done manually, recent studies on object detection suggest that performing selection automatically is more practical. ML algorithms have been created with the goal of extracting crucial data for the detestation of specific object classes, such as faces or pedestrians. Trackers based on Histogram of Oriented Gradients (HOG) features, for example, perform better than those based on Haar-like features. In fact, the feature extraction has a significant impact on tracking. Convolutional neural networks (CNN) are a deep learning technique that has recently had more success in feature extraction applied to object detection without the need for specialized class-dependent hand-coded features. It is logical to assume that these outcomes will get even better as neural networks continue to advance.

2 Related work

[11] claims that the visual tracking method is divided into two parts: a motion model and an observation model. Based on the description of the previous state, a motion model predicts the state in which the object will be. These filters include Kalman [12] and particle [13] and [14] filters, for instance. The appearance information for the monitored item is represented by the observation model, which validates predictions for each frame [15]. According to [10] study, the observation model is more important for visual tracking than the motion model. The two main types of observation models now in use are generative and discriminative approaches 11. While discriminative approaches are primarily concerned with the classification and trying towards the separation of the object from background, generative methods are more

focused on finding ROI that are similar to the desired item through template matching [12][16][17][18]. Which approach to choose relies on the object trackers intended use, For instance, discriminative CNN trackers are typically more accurate but faster than generative CNN trackers [11].

Deep visual tracking is used with more than just CNNs. Recurrent neural networks are well suited for sequence modelling because they can establish temporal connections between states and store memories of previous ones. Some authors suggest combining RNNs with correlation filters [25] or even adding features produced by RNN into CNN to achieve a robust feature representation [26] because of the success in the fields of handwriting [21][22] and speech recognition [23][24].

Deep neural networks can be used for the selection of the best candidate towards tracking object in addition to extracting their features. In this instance, feature extraction networks (FENs) and end-to-end networks are the two types of networks (EENs). In contrast to EENs, which handle both tasks, FENs have the advantage of acquiring high-quality features, which are then used by conventional methods to learn model appearance. As a result, EENs can provide an object location in the form of a bounding box or segmentation representation, or, as a complete solution, a probability map [27][28].

Due to the nature of the sport, writers have chosen to concentrate on solutions that will offer quick solutions with the highest level of precision in situations when quick and fast detection is required with constant multiple object movement in different directions. To handle object detection and simple online and real-time tracking, Yolo [29] was used. The identified item from Yolo is then cropped and passed to MoveNet for pose prediction of the cricket players to decide whether they are bowler or not based on their pose estimation.

3 Description

3.1 Yolo V3 and Move net lightning

Yolo V3 is an objection detection algorithm used in real time classification of data through computer vision. Developed by applying 1 x 1 detection with kernels into the feature maps of 3 different sizes at 3 different places in the network yolo v3 is achieved.

In this study Yolo V3 has been used for person detection, the video is passed frame to frame. Each frame is analyzed by yolo and provides the bounding box for each person present in it. The bounding box coordinates are used to crop the images and pass them for image pre-processing where a black image with the same image dimensions is created.

Move-net lightning is a pose detection algorithm used for obtaining 17 key points of a human. Here as the cropped image of the person is sent individually this helps us avoid the algorithm to consider key-point other individuals in the frame. The 17 key points are noted and are plotted in the black image that was previously generated. The 17 key points are nose: 0, left eye: 1, right eye: 2, left ear: 3, right ear: 4, left shoulder: 5, right shoulder: 6, left elbow: 7, right elbow: 8, left wrist: 9, right wrist: 10, left hip: 11, right hip: 12, left knee: 13, right knee: 14, left ankle: 15, right ankle: 16.

3.2 Data Generation

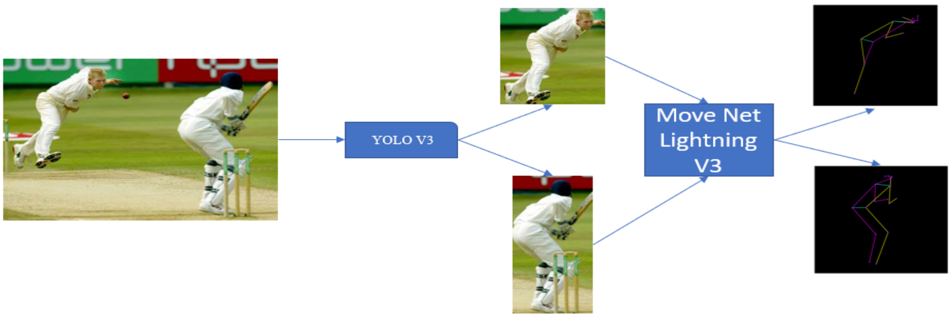

Figure 1: Data Generation process

The image is passed to yolo v3 where each person is cropped. The cropped images are then passed along to move net lightning v3 intended for real time applications, while Thunder is intended for applications that requires a high accurate prediction. After passing through it we get the 17 key points ear, eye, nose, neck and so on as mentioned above these values are stored and then plotted by mat plot library on a black image of same dimension as that of the input and saved in .png formate. These data will later be used for the training of the CNN.

3.3 Data description

While video is taken as an input the standard that we have found for the collected data is about 30 frames per second (FPS) we analyze about 3 frames for each 1 second of the video hence help us achieve a faster classification and a real time deployment.
Each individual player is detected via yoloV3 and passed to move net lightning and the 17 keypoints which is used for plotting in a black image of the same dimension as that of the input as shown in figure1.

Our dataset includes 2 classes consisting of the pre-processed data of a total 5588 such images out of which 3912 are used for training and 1676 files for validation. The 3912 images are used for training of CNN.

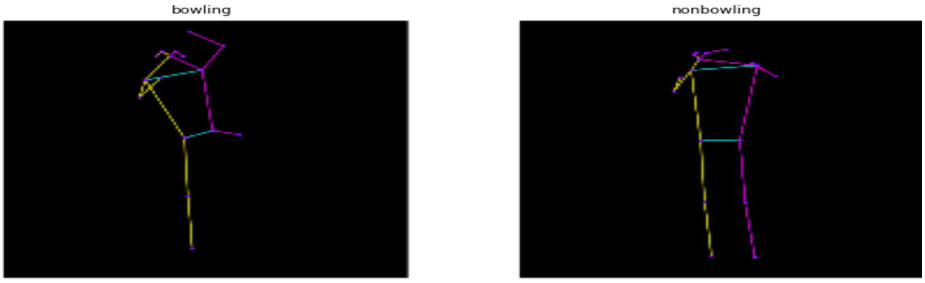

Figure 2: Data generated w.r.t 17 key-points

The total data generated between non-bowling and bowling action is 2985 and 2603. The bowling action is considered for those images where the action of bowling is observed moving one arm above the shoulder level all these images are grouped together forming the database for bowling and then the once with batting, keeping and umpire position are classified as non-bowling actions.

4 Methodology

4.1 Data Flow and transformation

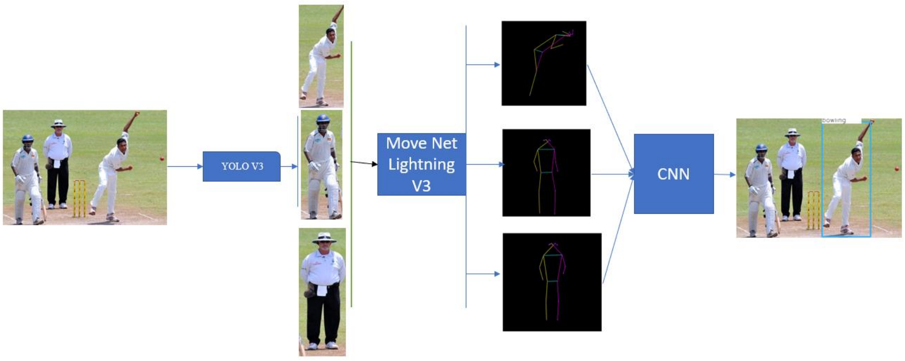

Figure 3: Data flow for ROI (region of interest)

The image from the video is passed to yolo v3 where each person is detected and each frame is passed with its bounding box coordinates to move net lightning where their key point is detected and passed to a black image where it is plotted.

Once obtained the pre-processed image it is rescaled to 180x180, normalized and later passed to CNN where it provides us with a result of bowling or non-bowling. These results are passed for each frame with their bounding box with the original frame. Only the bounding box which meets the criteria for the bowling action is taken into account for display. This result is later printed onto the bounding box having bowling action w.r.t original image. Each frame is processed individually one after another which is helpful for us to obtain ROI (region of interest). As seen in figure 3 each person is detected and passed to move net individually as for multi-pose detection the results are shown that in move net their key-points are not accurate for the body pose hence we pass each frame individually through single-pose detection.

4.2 CNN Architecture

The Convolution Neural Network (CNN) is based on the Rectifier Linear Unit (ReLU) as its activation function found to have time efficiency for both training and testing. ReLU is a type of activation function which yields the input as output if the input is positive, else it results in a null value. The input image is resized to 180x180 for the processing in CNN these resizing of images are done.

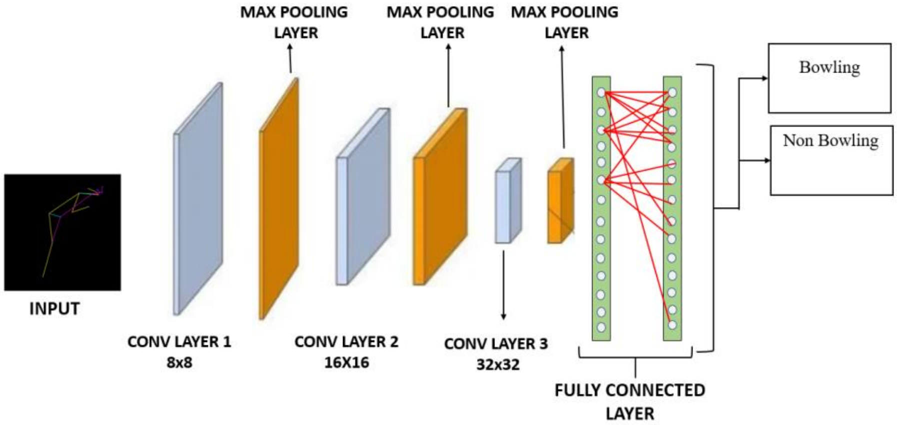

Figure 4: CNN architecture

The CNN is designed to have 3 convolution layers along with 3 max pooling layers and one fully connected or dense layer. There are 2 output classifications bowling and non-bowling action. The image generated by drawing the key points on a black image is the input to the CNN. The CNN is used for classifying the action is of bowling or of non-bowling.

If classified as bowling with a confidence level of the model is less than 65% is considered as non-bowling action this is done to reduce the false positive. Total number of parameters present in the network are 992,146 out of which the first convolution layer is of the size 8x8 (conv layer 1) convolution filter followed by size 16x16 (conv layer 2) and 32x32 (conv layer 3). Parameters that are present in the dense layer are 991296. The architecture of the convolutional neural network that is in use is as mentioned in Figure 4.

5 Result and discussion

The methodology was developed to obtain an accurate result but was found that would require to process each individual frame which increased the time taken to analyze each frame. Thus, allowing us to implement pipe-lining for real time deployment which can deal with is multi-processing task for each individual frames.

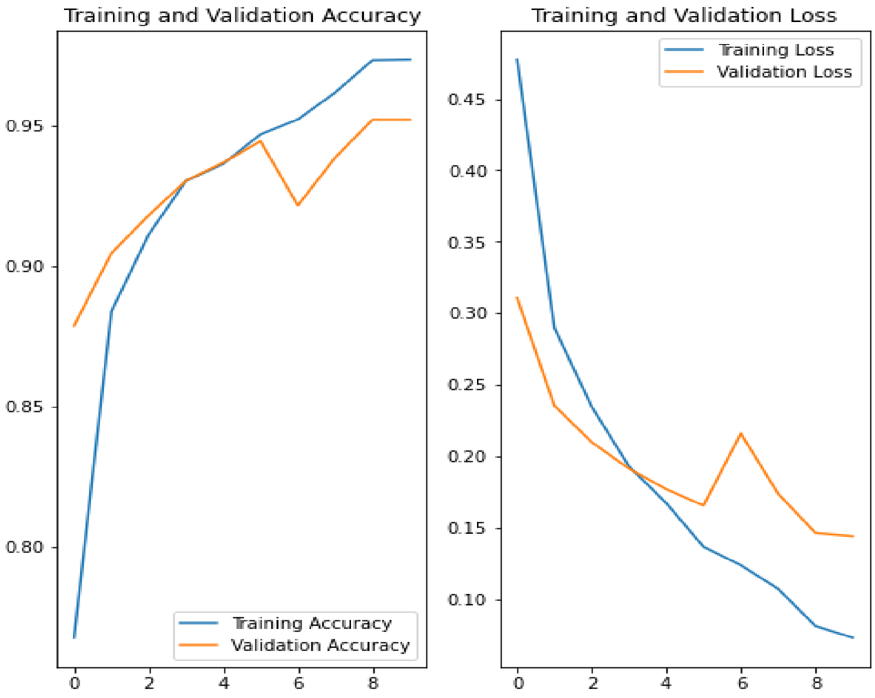

Figure 5: Accuracy and Loss

The accuracy while training was observed to reach 97.34 and that of validation accuracy was 95.19. The validation accuracy is observed to be 95.19 as the pose of bowling is action is similar throughout the different styles. The validation loss was 0.144 and training loss was 0.0729 this result was recorded for 10 number of epoch and a batch size of 64. The accuracy was found to be close in 3 last stages hence the model was terminated at the 10th epoch. This accuracy may have a different effect if the target size was set to vary from bowling, keeping and batting. One of the major challenges are of analyzing the data during tournament matches which has a huge content of information being generated.

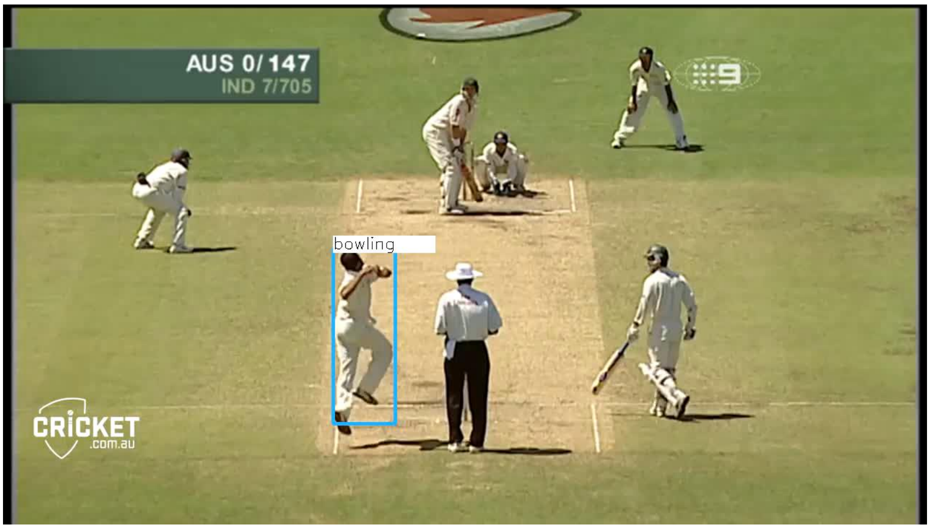

Figure 6:Tournament data analysis

6 Conclusion

Existing methods of detection are subjective and are highly reliant on complex methodology. Deep Learning has also made an impact on sport and its functionality. This can further be worked on knowing a player's development in their bowling action throughout the season. The actions which have the best result can also be considered as future work in this domain. The main application of such models is during tournaments helping the analyst to examine the bowling strategies for the game and learn how to develop their game plan accordingly as seen in figure 5. Hence, we can conclude saying that each player has a specific action to perform during the game and can be classified with the help of computer vision and deep learning algorithms which opens a complete domain of work in cricket and its advancement.

References

1. Y. Wang, J. F. Doherty, and R. E. Van Dyck, "Moving object tracking in video," in Proceedings - Applied Imagery Pattern Recognition Workshop, 2000, vol. 2000-January, pp. 95–101.
2. B. Tian, Q. Yao, Y. Gu, K. Wang, and Y. Li, "Video processing techniques for traffic flow monitoring: A survey," in IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2011, pp. 1103–1108.
3. J. M. B. Oñate, D. J. M. Chipantasi, and N. D. R. V. Erazo, "Tracking objects using Artificial Neural Networks and wireless connection for robotics," J. Telecommun. Electron. Comput. Eng., vol. 9, no. 1–3, pp. 161–164, 2017.
4. M. Brown, J. Funke, S. Erlien, and J. C. Gerdes, "Safe driving envelopes for path tracking in autonomous vehicles," Control Eng. Pract., vol. 61, pp. 307–316, 2017.
5. V. A. Laurense, J. Y. Goh, and J. C. Gerdes, "Path-tracking for autonomous vehicles at the limit of friction," in Proceedings of the American Control Conference, 2017, pp. 5586–5591.
6. S. Walker et al., "Systems and methods for localizing, tracking and/or controlling medical instruments," 2017.

7. M. Buric, M. Pobar, and M. Ivasic-Kos, “An overview of action recognition in videos,” 2017 40th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2017 - Proc., pp. 1098–1103, 2017.
8. P. Viola and M. Jones, “Managing work role performance: Challenges for twenty-first century organizations and their employees.,” Rapid Object Detect. using a Boost. Cascade Simple Featur., 2001.
9. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 2005.
10. N. Wang, J. Shi, D. Yeung, J. J.-P. of the IEEE, and undefined 2015, “Understanding and diagnosing visual tracking systems,” openaccess.thecvf.com.
11. P. Li, D. Wang, L. Wang, and H. Lu, “Deep visual tracking: Review and experimental comparison,” Pattern Recognit., vol. 76, pp. 323–338, 2018.
12. A. Heidari and P. Aarabi, “Real-time object tracking on iPhone,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2011, vol. 6938 LNCS, no. PART 1, pp. 768–777.
13. P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, “Color-based probabilistic tracking,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2002, vol. 2350, pp. 661–675.
14. Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, “Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 10, pp. 1728–1740, 2008.
15. X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. Van Den Hengel, “A survey of appearance models in visual object tracking,” ACM Transactions on Intelligent Systems and Technology, vol. 4, no. 4. 2013.
16. J. Kwon and K. M. Lee, “Tracking by sampling and integrating multiple trackers,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 7, pp. 1428–1441, 2014.
17. X. Mei and H. Ling, “Robust visual tracking and vehicle classification via sparse representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 11, pp. 2259–2272, 2011.
18. A. Adam, E. Rivlin, and I. Shimshoni, “Robust fragments-based tracking using the integral histogram,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, vol. 1, pp. 798–805.
19. C. Xu, W. Tao, Z. Meng, and Z. Feng, “Robust visual tracking via online multiple instance learning with Fisher information,” Pattern Recognit., vol. 48, no. 12, pp. 3917–3926, 2015.
20. L. Zhang and P. N. Suganthan, “Robust visual tracking via co-trained Kernelized correlation filters,” Pattern Recognit., vol. 69, pp. 82–93, 2017.
21. D. Cireşan and U. Meier, “Multi-Column Deep Neural Networks for offline handwritten Chinese character classification,” in Proceedings of the International Joint Conference on Neural Networks, 2015, vol. 2015-Septe.
22. D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, “Convolutional neural network committees for handwritten character classification,” in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2011, pp. 1135–1139.
23. Z. Wu, C. Valentini-Botinhao, O. Watts, and S. King, “Deep neural networks employing Multi-Task Learning and stacked bottleneck features for speech synthesis,” in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2015, vol. 2015-Augus, pp. 4460–4464.
24. S. Kim, T. Hori, S. W.-2017 I. International, and U. 2017, “Joint CTC-attention based end-to-end speech recognition using multi-task learning,” ieeexplore.ieee.org.
25. Z. Cui, S. Xiao, J. Feng, S. Y.-P. of the IEEE, and U. 2016, “Recurrently target-attending tracking,” openaccess.thecvf.com.
26. H. Fan and H. Ling, “SANet: Structure-Aware Network for Visual Tracking,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2017, vol. 2017-July, pp. 2217–2224.
27. D. Held, S. Thrun, and S. Savarese, “Learning to track at 100 FPS with deep regression networks,” in Lecture Notes in Computer Science, 2016, vol. 9905 LNCS, pp. 749–765.
28. G. Ning et al., “Spatially supervised recurrent convolutional neural networks for visual object tracking,” in Proceedings - IEEE International Symposium on Circuits and Systems, 2017.
29. Redmon, J. and Farhadi, A. 2017. “YOLO9000: better, faster, stronger,” arXiv preprint.

30. A. Bewley, Z. Ge, L. Ott, et. al., 2016, “Simple online and realtime tracking,” ieeexplore.ieee.org.
31. N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” in Proceedings - International Conference on Image Processing, ICIP, 2018, vol. 2017-Septe, pp. 3645–3649.
32. H. W. Kuhn, “http://dx.doi.org/10.1002/nav.3800020109The Hungarian method for the assignment problem,” Nav. Res. Logist. Q., vol. 2, no. 1–2, pp. 83–97, Mar. 1955.
33. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2017.