YOLO-Based Framework with Temporal Context and Network Analysis for Real-Time Basketball Video Understanding

Mingsheng Xu
Physical Education Department of Yangzhou Industrial Vocational and Technical College, Physical Education Teaching, Yangzhou, 225127, China
Email: xumingsheng85@163.com

Keywords: basketball analytics, YOLO framework, player tracking, passing relationship analysis, network science, sports optimization, deep learning

Received: March 20, 2025

This paper proposes a real-time system for the automated recognition of player interactions in basketball videos using a YOLO-based deep learning framework integrated with temporal context and network analysis. The framework enhances YOLOv3 by incorporating information from adjacent frames to improve jersey number recognition and player tracking under occlusion. A novel algorithm, Joy2019, is introduced to assign consistent player identities over time using retrospective association and temporal smoothing. Passing interactions are represented as directed graphs, and a new Player Centrality (PC) score is developed to evaluate player influence using weighted actions. Experimental evaluation on annotated NBA footage demonstrates that Joy2019 improves jersey number recognition accuracy from 36.1% to 73.8%, and increases player detection rates to 90.2%. Compared to baseline YOLO outputs, the proposed method reduces mean absolute percentage error (MAPE) in pass graph reconstruction to 24.3–33.1%. These findings demonstrate the potential of temporal deep learning and network science to improve the robustness, accuracy, and tactical insight of automated sports analytics systems.

Povzetek: Razvit je nadgrajen YOLOv3 model z začasnim kontekstom in mrežno analizo za samodejno prepoznavo igralcev ter analizo podaj v košarkarskih videih v realnem času.

1 Introduction

Sports analytics is a rapidly changing, interdisciplinary branch of science, which incorporates knowledge in sports science, statistics, mechatronics, electrical engineering, and computer science to develop new solutions to analyze athletic performance and identify the crucial parameters responsible for the success of teams and individuals [1]. In light of the rise of enthusiasm globally towards sports, analytics today is a cornerstone from both professional and amateur points of view, developing very deep methodologies to optimize strategy and performance and understand players. Such team sports like basketball, soccer, and volleyball are the biggest beneficiaries due to these advances through their adoption of analytics strategies and means for improving efficiency and presentation with data-driven insights that go ahead to shape the future management and coaching of teams [2].

In this respect, over the years, the focus of sports analytics has gradually shifted from basic statistical summaries to highly sophisticated computational tools that can unearth intrinsic relationships between players, strategies, and game dynamics. This transformation reflects the higher intricacy of modern-day sports, where data from the on-field action seamlessly integrates with the performance metrics off the field. Tools that were once descriptive analytics have transformed into predictive and prescriptive systems, allowing teams and coaches to take real-world insights into what to do. These days, modern methods mean team behaviors can be looked at on both a macro and micro level for patterns impossible to fathom through human observation. It is this evolution that not only optimizes the potential of every individual athlete but also allows teams to optimize resources and strategies toward overall success [3].

Despite these advances, the field of sports analytics remains very challenging, especially in dynamic team sports where the interaction between players and objects involves complex and continuously changing information. Most traditional methodologies are reliant on human observation in terms of tracking player movements, identifying ball trajectories, or tagging key game events. Such processes are time-consuming, labor-intensive, and easily prone to errors such as false positives and misclassifications [4] [5]. Moreover, this reliance on human judgment comes with inconsistencies that can lead to a lack of reliability in the data and further analysis. This is a crucial bottleneck within the field; hence, there is a dire need for automated systems that can efficiently handle the large volumes of data created during sports events.

The recent emergence of artificial intelligence, especially deep learning, has brought about transformative opportunities in sports analytics [6]. These CNNs and their related frameworks have made object detection and classification applications realistic with near-human accuracy [7]. Among them, the YOLO framework has emerged as the leading real-time object detection system

and has seen wide adoption for various applications in sports analytics. Its ability to process visual data in real time and with high accuracy makes YOLO a very good candidate for player and ball tracking in fast-moving sports. However, the traditional implementations of YOLO are limited in situations where the camera is frequently shifted, players are overlapping, or video sequences are interrupted. These require developing specialized adaptations to fit into special needs in sports analytics.

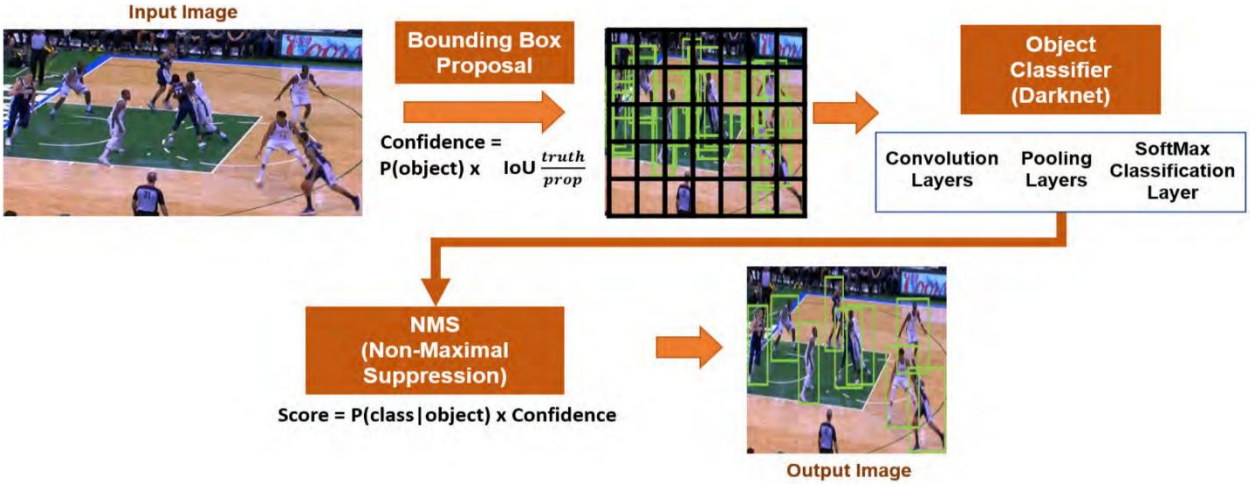


Figure 1: The process of object detection and classification using YOLO.

This paper presents a state-of-the-art system in automatic sports analytics, which leverages the synergy between deep learning and network science to address the above-mentioned challenges. We present an approach that extends YOLO with contextual information from previous and subsequent frames in a video to allow tracking at higher accuracy under adversarial conditions. With this, the system will be able to track players' accurate tracking, jersey number detection, and ball trajectory tracing in very challenging situations. Inspired by network science, we further propose a new metric for the measurement of player importance within the context of team interactions, called Player Centrality. With this, we combine the approaches and provide a comprehensive framework that minimizes manual intervention while giving actionable insights into improving team performance and strategy optimization. The primary objectives of this research are:

(1) To improve jersey number recognition under occlusion by incorporating temporal information from adjacent frames.

(2) To construct and analyze directed passing graphs among players based on ball possession detection.

(3) To propose and evaluate a Player Centrality score that quantitatively reflects player contribution to team dynamics using network science.

2 Related work

Serious research and a number of key developments marked sports analytics as a nascent scientific discipline over the last few decades. Earlier studies mainly related to data aggregation and graphical visualizations of performances via statistical packages contributed a great value to the data insights for game dynamics yet entailed much time-wasting hand-crafting. These efforts mostly focused on summarizing the performance of players and teams, often using simple descriptive statistics that were not deep enough to disclose underlying patterns in player interactions and tactical strategies. Although these studies provided the much-needed impetus to begin with, their reliance on human observation limited scalability and introduced the risk of bias and error [8]. In this phase, much research pointed out that automation approaches can reduce human labor and increase the reliability of sports data analysis.

Deep learning is proving a point of inflection in sports analytics, where tasks are being automated that were till recently at the mercy of a human eye. Among various developments in artificial intelligence, YOLO has emerged as one of the most applied frameworks for object detection and classification in sports analytics. Various studies using YOLO have shown promising results in detecting and tracking players and objects with high precision and speed, even in dynamic and complex environments. However, most of the existing implementations of YOLO exhibit shortcomings when applied to common scenarios featuring heavy camera shifts, occlusions, and overlapping objects. These issues have led researchers to propose enhancements for making the algorithm robust and reliable in sports-specific applications by incorporating contextual information from adjacent video frames [9][10].

On the other hand, network science has emerged as a strong tool in the analysis of team dynamics in sports. By framing player movements and ball passes in network terms, the authors have managed to distill vital information about team strategies and individual contributions. Network science metrics offer systematic centrality measures and clustering coefficients to enable evaluation of player importance and efficiency of team tactics. These methods have been especially successful in team sports like basketball and soccer, where the pattern of passing and positional dynamics often determines the outcome.

Recent works have indeed applied network-based approaches to uncover relationships that were previously hidden by traditional statistical analyses, drawing new perspectives on team performance [11][12].

Other integrated systems also include the integration of sensor data with the video annotation tools and technologies of computer vision for some challenges in sports analytics regarding data collection and processing. The proposed integrated framework of Halvorsen et al. uses sensors and camera systems for automating event tagging and player tracking. While such systems have reduced the need for manual labor, they are still limited by issues of scalability and error management, especially when dealing with large datasets generated in real-time sports environments [13]. These challenges indicate that further approaches are needed to seamlessly handle modern sports data's complexity. A comparison of object detection and tracking methods is shown in Table 1.

Despite the progress made so far, manual event tagging and classification remain significant bottlenecks in the existing sports analytics systems. Numerous studies have proved all these processes to be time-consuming and error-prone, with false positives and false negatives. These constraints make general sports analytics less effective and narrow its capacity to deliver actionable insights. Combined, deep learning and network science drive automated systems around these challenges, with much faster, more reliable solutions that scale. The proposed system further builds on such foundations, providing a holistic framework in the analysis of team sports by fusing YOLO's robustness with network science's analytical power to produce actionable insights that power performance optimization.

Table 1: Comparative summary of related methods for basketball video analysis.

Method	Object Detection	Tracking	Strengths	Weaknesses
YOLOv3 [6] [10]	✓	✗	Fast inference	Fails under occlusion
Faster-RCNN [10]	✓	✗	High accuracy	Low frame rate
Halvorsen et al. [13]	✓ (Sensor-aided)	✓	Multi-modal integration	Needs external hardware
Joy2019 [17]	✓	✓	Temporal robustness, ID reuse	Challenged by long occlusion

3 System design

In this section, we present the mechanism of object detection from video clip frames and a novel machine-learning procedure for recognizing players and their movements.

A) OBJECT DETECTION METHODOLOGY

We adopted the YOLO framework for real-time object detection in basketball video recordings. YOLOv3 has been chosen for its outstanding performance compared to other object detection systems such as Fast-RCNN1 and FasterRCNN2 for the basis of mAP and the speed of object detection measured by Frames Per Second (FPS). Figure 1 presents the mechanism of object detection and classification for YOLO. For every frame of video, YOLO divides the input image into a grid of dimensions $D \times D$. Each cell of that grid proposes multiple bounding boxes and calculates a confidence score for every such box, indicating the likeliness of an object present in that box. The dimension of a bounding box is represnted as (x, y, w, h), where (x, y) represents the center of the box and w, h denotes its width and height. YOLO makes use of a K-means clustering algorithm to create the bounding

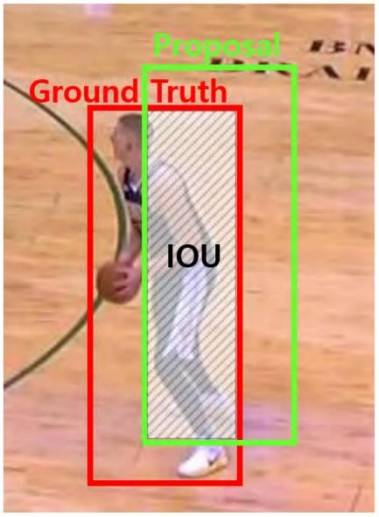


Figure 2*:* An illustration of IOU evaluation.

box dimensions on the basis of training data. The best number of clusters is set to 5; this gives a good balance between computational complexity and high recall [14].

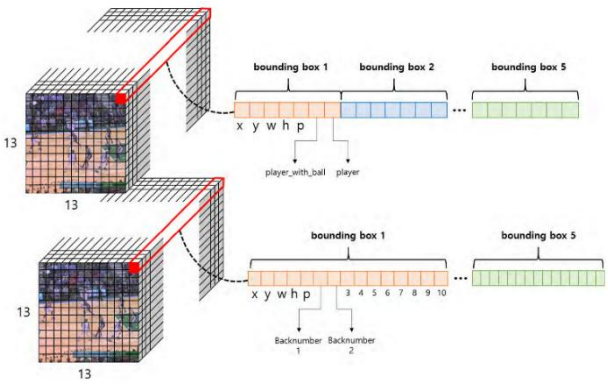


Figure 3: Tensors learned by YOLO for detecting the ball, players, and jersey numbers.

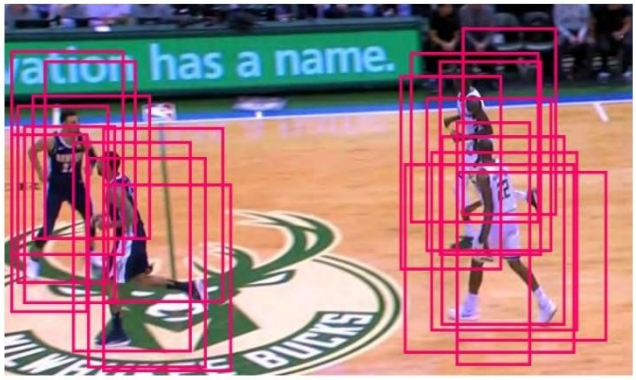


Figure 4: An example showcasing multiple redundant bounding boxes proposed around players.

IoU for each proposed bounding box is calculated between the predicted box and the ground truth box, defined as

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (1)$$

The confidence score for each bounding box, P_{box} , combines the probability of the object being present in the box and its loU with the ground truth box:

$$P_{\text{box}} = P(\text{ Object }) \cdot \text{IoU}_{\text{truth}} \quad (2)$$

Here, P (Object) is the predicted probability that an object is present in the grid cell, and $\text{IoU}_{\text{truth}}$ is the loU calculated between the predicted bounding box and the ground truth bounding box. The final output of YOLO for each cell is represented as a tensor of size $D \times D \times (5B + C)$, where B stands for the number of bounding boxes in each cell, and C stands for classes of objects. For the frames from basketball videos, we set D = 13, which has been found as a sweet-spot resolution allowing good player detection. Each grid cell predicts B = 5 bounding boxes, while the tensor encodes coordinates (x, y, w, h), confidence score, and the probabilities of each class. A two-stage classification process has been introduced to address the challenge of detecting ball possessors and jersey number identification. First, it will detect if a player is a ball possessor or not with C = 2 number of classes. The confidence score for the detection of the ball possessor can be computed as [15]:

$$P(\text{ Ball Possessor }) = \sigma(P_{\text{box}} \cdot P_{\text{class}}) \quad (3)$$

where P_{class} represents the predicted probability of the object being a ball possessor, and σ is the sigmoid activation function that normalizes the output probability. In the second stage, the model identifies jersey numbers for ball possessors. The model will predict C=10 classes corresponding to jersey numbers from 0 to 9. The softmax gives the probability for each class:

$$P(\text{ Class } | \text{ Object }) = \frac{e^{s_i}}{\sum_{j=1}^{C} e^{s_j}} \quad (4)$$

where s_i is the score for class i, and the denominator sums over all class scores. In each grid cell, YOLO uses this probability to assign the detected object to a specific class. This framework is implemented based on Darknet, which is the most optimized C implementation of CNNs to date, allowing real-time performance even for high-resolution frames of basketball videos. YOLO predicts the final score of each proposed bounding box by embedding class probability, object presence probability, and the Intersection over Union (loU) with the ground truth bounding box as presented in Equation 3:

$$P_{\text{box}} = P(\text{ Class } | \text{ Object }) \cdot P(\text{ Object }) \cdot \text{IoU}_{\text{truth}} \quad (5)$$

where P (Class | Object) is the conditional probability of the object to belong to the particular class, P(Object) is the probability of object existence in a grid cell, IoU_„truth „ measures overlap between the predicted bounding box and ground truth box. The obtained combined score allows ranking the bounding boxes for further processing.

Figure 5: Boxes showing the possession of the ball.

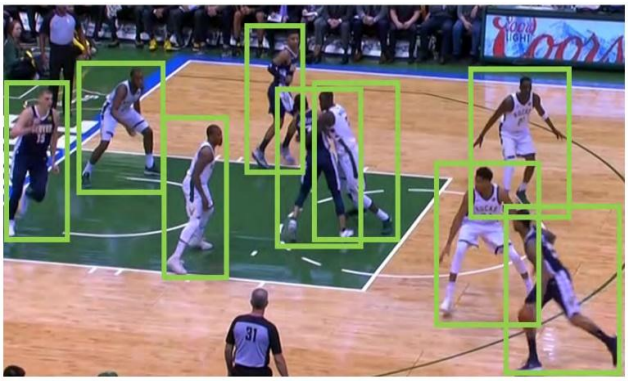


Figure 6: Gathering ground truth bounding boxes for jersey numbers specific to each team.

The grid cells tend to be redundant and generate bounding boxes that are overlapping, especially when multiple cells predict the same object. This is due to the fact that adjacent cells with their fields of view may partially overlap and thus suggest bounding boxes for the same object. Figure 4 illustrates this. In order to overcome this problem, YOLO uses a post-processing method called Non-Maximal Suppression (NMS), which selects and

retains only the most relevant bounding box for every object detected. It then arranges all the predicted bounding boxes in descending order according to their

Figure 7: Gathering ground truth bounding boxes for ball possession detection.

scores, P_{box}. It selects a bounding box that has the maximum score as the most likely candidate for the detected object. For each of the remaining bounding boxes, NMS calculates the loU of that box with the highest-scored bounding box. The box will be discarded if its calculated loU is more than a predefined threshold, $IoU_{\text{threshold}}$. The described process is repeated iteratively until all boxes are processed.

Mathematically, the NMS algorithm can be summarized as follows. Let $\mathcal{B} = \{b_1, b_2, \ldots, b_n\}$ model the set of all proposed bounding boxes sorted by their scores, where b_i is the bounding box with the $i-th$ highest score. For each b_i, NMS retains it if:

$$\text{IoU}\left(b_i, b_j\right) < \text{IoU}_{\text{threshold}}\,,\ \forall j < i \tag{6}$$

Bounding boxes that violate this condition are discarded, as they are considered redundant due to their high overlap with already-selected boxes. It filters such that any single detected object retains only one bounding box, therefore avoiding duplication and enhancing the object detection model as a whole. On the other hand, NMS selects a class for the selected bounding box by considering class probabilities of P (Class | Object), which are derived during the detection stage. By progressive elimination of superfluous bounding boxes, the NMS algorithm seeks to optimize the detection pipeline while delivering dependable bounding box predictions with high confidence for subsequent scrutiny.

B) DEEP LEARNING PROCEDURE

We have collected the ground truth bounding boxes for each class to be recognized by the object detector. For players in possession, the bounding box captured the ball and the torso of the ball possessor, as depicted in Figure 5. For the other players, the whole body was enclosed inside a bounding box, as shown in Figure 6. We also annotated jersey numbers separately by drawing bounding boxes capturing the visible numbers, as shown in Figure 7. Each bounding box was further tagged with team affiliation and the corresponding jersey number to create an extensive dataset for training deep learning models. For recognition, the whole process goes via two stages: recognizing the phase of whether a player has possession of the ball, followed by identifying the jersey number of the player identified carrying the ball. This work actually adopted this two-step method because the capturing of a jersey number for all the players within a frame remains rather challenging and often unresolvable whenever numbers turn out to be blurred out as a result of motion or interaction between the players across differing lighting conditions. A joint approach was not practical since it relies on visibility of jersey numbers across all the frames, which is not always achievable. Limiting data collection only to ball possessors also resulted in an insufficient dataset for training a robust model for recognizing jerseys.

For the first phase, the objective was to classify bounding boxes into ball possessors or non-possessors. This binary classification task required computing the probability $P_{\text{possessor}}$ for each bounding box, defined as:

$$P_{\text{possessor}} = \sigma(P_{\text{box}}) \tag{7}$$

where $P_{\text{box}} = P\,(\text{Class} \mid \text{Object}) \cdot P(\,\text{Object}\,) \cdot \text{IoU}_{\text{truth}}$, as previously established. The training objective for this phase combined binary cross-entropy loss for classification with a regression loss for bounding box coordinates, ensuring accurate localization:

$$\mathcal{L}_{\text{possessor}} = \frac{1}{N}\sum_{i=1}^{N}\Big[-y_i \log P_{\text{possessor}}(x_i) - (1 - y_i)\log\left(1 - P_{\text{possessor}}(x_i)\right)\Big] + \lambda \sum_{i=1}^{N} \text{SmoothL}\ 1\left(b_i, \hat{b}_i\right) \tag{8}$$

Input: Frames: video frames with bounding boxes
Output: LabeledFrames: Frames and their bounding boxes labeled with track IDs and jersey numbers

```
foreach frame F ∈ Frames do
    if F is the first frame then
        Assign a unique trackingID from 1 to 5 to each bounding box;
        Set F as the preceding frame;
    else
        foreach bounding box c ∈ F do
            Find the closest bounding box, b, in the preceding frames;
            c.trackID ← b.trackID;
            c.jerseyNumber ← b.jerseyNumber;
            if the jerseyNumber of c recognized then
                From LabeledFrames, retrieve P = {p|p.trackID = c.trackID};
                foreach p ∈ P do
                    if there is no jersey number for p then
                        p.jerseyNumber ← c.jerseyNumber;
            Insert the current frame and its bounding box information to LabeledFrames;
```

Algorithm 2: Joy2019, The Player Tracking Algorithm

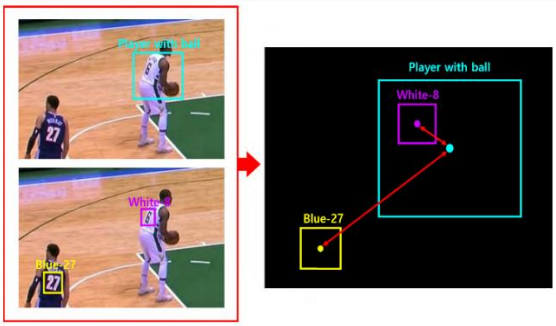


Figure 8: Recognizing the jersey number of the player in ball possession.

Figure 9: An example of associating a jersey number with a box sharing the same tracking ID (tid).

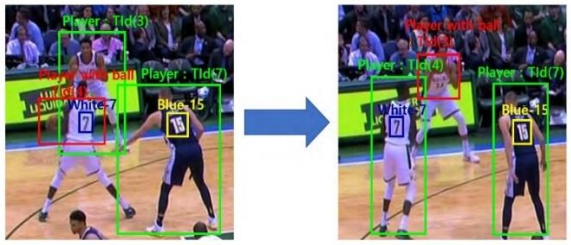


Figure 10: Example of recognizing and tracking pass actions

Figure 11: The challenge of tracking a player box that reappears in the frame.

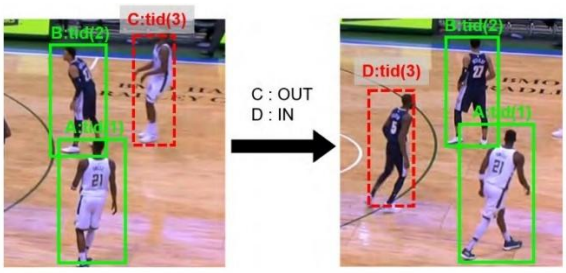


Figure 12: The tracking ID is incorrectly shared between a box that disappeared and a box that reappeared in the scene.

Here, y_i denotes the ground truth label of ball possession, x_i is the input feature of the bounding box, b_i is the predicted bounding box, $\hat{b}_i$ is the ground truth bounding box, and λ is a weighting parameter for the regression term. The second phase of the model focuses on jersey number recognition for the detected ball possessor, involving multi-class classification that calculates a probability distribution over C classes of jersey numbers using the softmax function. The loss fun ction for this phase was the categorical cross-entropy loss, defined as:

$$\mathcal{L}_{\text{jersey}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log P_{\text{class}}(j) \qquad (9)$$

where y_{ij} is a binary indicator (1 if the ground truth class for the sample i is j, 0 otherwise), and $P_{\text{class}}(j)$ is the predicted probability for class j. Extensive augmentation, such as random cropping, flipping, rotation, and adjusting brightness, was used to generate a large-scale training dataset. Those augmented data mimic real conditions where there may be large variations in camera angles of view, partial occlusion, and changing lighting conditions. Outputs of the first-phase ball possessor model were used as the input to the second-phase jersey number model; hence, this became a tightly integrated pipeline for successive detection and recognition tasks [16].

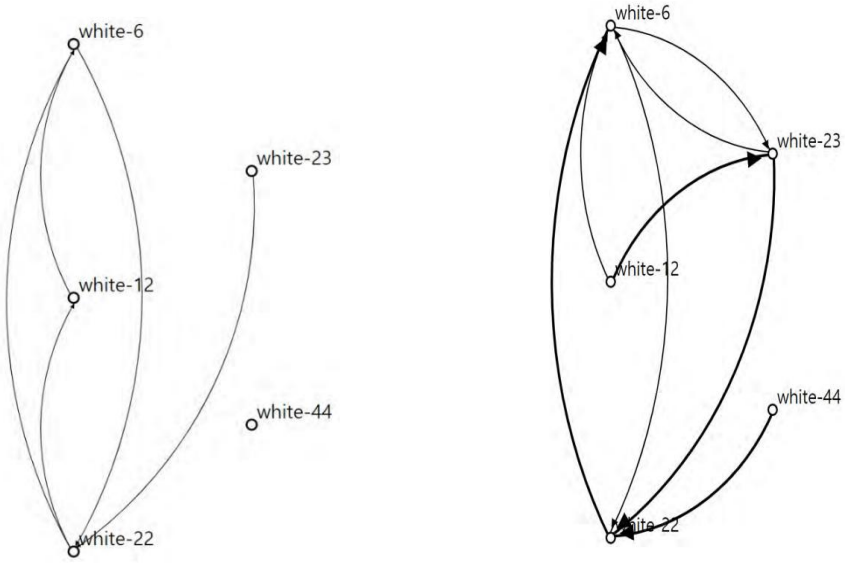
B) DEEP LEARNING PROCEDURE

To track the ball's movement and interaction between players, players and their positions in successive video frames must be identified. Traditional YOLO implementation is limited in handling transitions of scenes and overlapping players in two-dimensional video frames due to the loss of absolute coordinate information. However, YOLO cannot recover the track of players and balls once they are lost temporarily due to a camera angle change or occlusion. In view of such challenges, we adapted YOLO and developed a new tracking algorithm called Joy2019, which increases reliability and accuracy in tracking under even complex situations. For any given frame, Joy2019 [17] assigns a unique tracking ID (tid) to each detected player box using the algorithm. For example, as can be seen from Figure 8, jersey numbers are not directly labeled on player boxes unless the back number is clearly visible. If a new player box is detected, the algorithm searches up to ten preceding frames to find a matching box with the same tid within the closest proximity. This backward search is necessary to avoid instances where the previous frame might not have enough information to generate a proper match. The number of frames to be searched has been cautiously chosen as ten, considering both computational efficiency and tracking accuracy.

Considering these improvements, there are still some limitations. For example, the players could disappear from the video frame due to dynamic camera movements. For example, as shown in Figure 11, Player C, tid:4 disappears and then reappears, but the algorithm gives a new tracking ID to the same player. Hence, one player may get multiple tracking IDs at different instances.

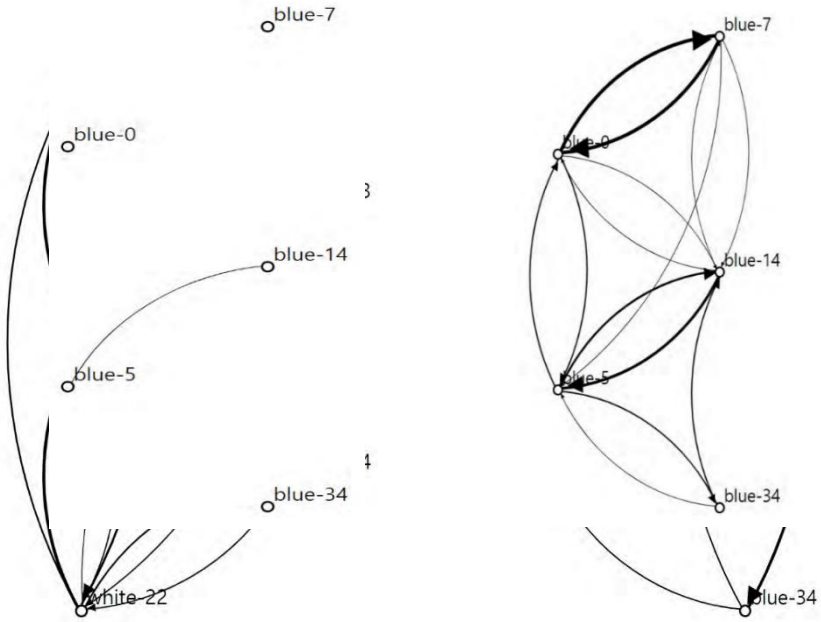
Figure 13: The tracking IDs of two overlapping player boxes are mistakenly swapped.

Figure 14: Boxes that are temporarily unrecognized are assigned new tracking IDs.



(a) (b)

Figure 15: (a) A manually created graph depicting the passing interactions among players in Team White. (b) A graph generated by YOLO illustrating the passing dynamics among players in Team White.



(a) (b)

Figure 16: (a) A graph generated by Joy2019 illustrating the passing interactions among players in Team White. (b) A manually created graph depicting the passing dynamics among players in Team Blue.

(a) (b)

Figure 17*:* (a) A graph generated by YOLO illustrating the passing interactions among players in Team Blue. (b) A graph generated by Joy2019 depicting the passing dynamics among players in Team Blue.

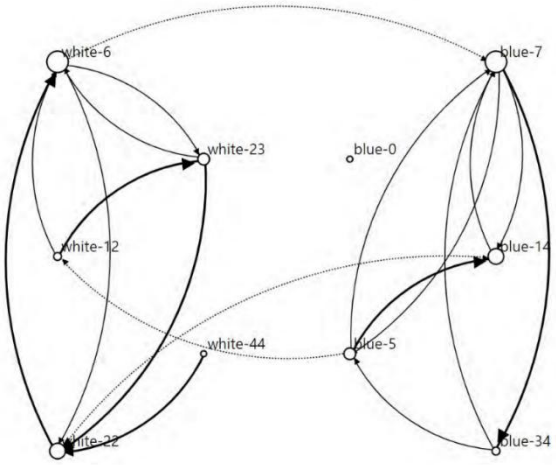


Figure 18*:* A manually created graph illustrating the passing dynamics between players.

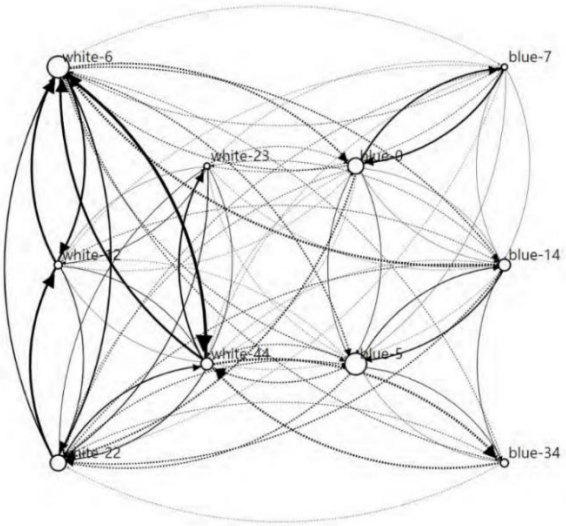


Figure 19: A graph generated by Joy2019 depicting the passing interactions between players.

Also, in Figure 12, Player D may accidentally take tid:4 thereby messing it up when jersey numbers are assigned later on. This becomes an issue when, for example, Player D is incorrectly associated with Player C's jersey number and thus continues to spread the error through frames. Another major issue is the overlapping of player boxes. When two players, say Player A: tid (1) and Player B: tid (2), are fighting for the ball and their images overlap in a frame, then the detection system usually mistakes them for one player. This misunderstanding, as represented in Figure 13, causes the swapping or wrong assignment of tracking IDs. In this case, Player B may be wrongfully attached to tid (1) in the overlap frame; this may cause further errors later in the frames as it reassigns the tracking IDs. Such overlaps can result in cascading effects where the player information propagates wrongly across frames. The algorithm tries to curb such errors by retrospectively updating the jersey number. This therein involves once the jersey number of a player box is recognized, then the algorithm iterates back to all previous frames, assigning the same jersey number to the boxes with the matched tid. For example, given Figure 9, it is observed that a player box with tid (1) eventually has an association in jersey number 27. This jersey number identification helps in establishing relations of ball passes, such as the relation where a player wearing jersey number 7 is passing the ball to the player detected in a box with tid (3), as shown in Figure 10.

However, with these enhancements, too, Joy2019 has certain limitations. Long-term player tracking is problematic in cases where players disappear and then reappear after some time or when jersey numbers are not visible for several consecutive frames. Also, as shown in Figure 14, the system fails when several players disappear from the frame for a short time and then reappear, thus getting wrong tracking IDs assigned. Another approach may be to capture video footage from different angles or capture the whole court without zooming in and out for a wider and more consistent view of the position of the players. The other alternative is that the system can be trained to identify the players by unique physical characteristics, such as face or body shape. This, however, requires ultra-high-resolution video and close-up shots, increasing resource requirements.

While Joy2019 addressed many intrinsic issues related to player and ball tracking, there is still room for innovation in overcoming challenges and further enhancing the robustness of tracking for real-world applications.

4 Evaluation and discussions

We evaluated our system using a deep learning machine equipped with a state-of-the-art configuration: an Intel Core i7-8700 central processing unit, an Nvidia GeForce GTX 1080Ti graphics processing unit, and 32 gigabytes of memory. The high configuration provided a stable and smooth platform for all experiments done, running Ubuntu 18.04 LTS. We tested the system's performance using publicly available video footage of an NBA game. The video was encoded in MPEG4 at a bitrate of 1200 Kbps, which afforded a resolution of 1280x720 pixels and a frame rate of 30 FPS. These features led to the creation of a high-quality dataset useful for analysis. In this regard, over 16,000 ground truth bounding boxes for players, balls, and jersey numbers were annotated very carefully on a total of 1204 video frames. The annotations served as the main input to the YOLO model, which was pre-trained on the Microsoft COCO dataset [26]. For testing purposes, a three-minute clip of video was cut from the game video and was not included in the training dataset to allow for unbiased evaluation.

From the annotated frames, a network was created which represented the passing relations between players. In this work, each player was represented by a node, and the directed edges between nodes represent passing actions, which gave the distinction between inbound and outbound passes. The passing networks generated by human

annotators, YOLO and Joy2019 were visualised using D3.js [27] as illustrated in Figures 15 to 17. These figures highlight the discrepancies in performance between the three approaches. As shown in Figures 15b and 17a, the YOLO algorithm struggled to satisfactorily identify passing relations, which can be explained by its poor detection of jersey numbers. On the other hand, Joy2019, shown in Figures 16a and 17b, had a far better reconstruction of passing networks. The improvement seen can be explained by Joy2019's ability to infer jersey numbers also in cases where they were partly occluded or temporarily out of sight using tracking information from preceding and following frames. Joy2019 showed significant improvements in terms of jersey number recognition and player identification. A comparison in the accuracy of recognizing jersey numbers and players between human annotators, YOLO, and Joy2019 is shown in Table 2. While YOLO achieved a reasonable level of accuracy, Joy2019 outperformed it with an increase of over 100% in accuracy. Furthermore, there were no instances where players were misclassified as non-player objects using Joy2019. This shows the robustness of Joy2019 in handling complex scenarios such as overlapping players or continuously changing camera angles.

Table 2: Accuracy of recognizing jersey numbers and players.

	Manual Evaluation	YOLO Framework	Joy2019 Algorithm
Accuracy (Jersey Numbers %)	100	36.1± 3.4	73.8± 2.7
Detection Rate (Players%)	100	88.7± 2.1	90.2± 1.9

Table 3: Average node degree of pass graph for each team. (H and J stands for interpretation by human and Joy2019, respectively.

Team	Total Degree (H vs J)	Outbound Degree (H vs J)	Inbound Degree (H vs J)	MAPE (%)
White	5.8 vs 7.0	3.0 vs 3.5	2.8 vs 3.5	33.1%
Blue	4.8 vs 5.5	2.1 vs 2.7	2.7 vs 2.8	24.3%

Table 4: Percentage distribution of passes made to teammates by each player in Team White, as analyzed and generated using the YOLO framework.

Passer	White-6 (H / Y)	White-12 (H / Y)	White-22 (H / Y)	White-23 (H / Y)	White-44 (H / Y)	MAPE (%)
White-6	0.0 / 0.0	0.0 / 0.0	50.0 / 50.0	50.0 / 0.0	0.0 / 0.0	200.0
White-12	33.3 / 100.0	0.0 / 0.0	0.0 / 0.0	66.7 / 66.7	0.0 / 0.0	300.0
White-22	100.0 / 50.0	0.0 / 0.0	50.0 / 50.5	0.0 / 0.0	0.0 / 0.0	50.0
White-23	33.3 / 33.0	0.0 / 0.0	0.0 / 0.0	66.7 / 66.7	0.0 / 0.0	149.9
White-44	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	100.0

Table 5: The percentage distribution of passes made to teammates by each player in Team Blue, as analyzed and generated using the YOLO framework.

Passer	Blue-0 (H / Y)	Blue-5 (H / Y)	Blue-7 (H / Y)	Blue-14 (H / Y)	Blue-34 (H / Y)	MAPE (%)
Blue-0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	None
Blue-5	0.0 / 0.0	0.0 / 0.0	33.3 / 33.3	66.7 / 66.8	0.0 / 0.0	149.9
Blue-7	0.0 / 0.0	25.0 / 25.5	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	300.0
Blue-14	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	100.0 / 100.0	0.0 / 0.0	100.0
Blue-34	0.0 / 0.0	50.0 / 50.5	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	200.0

Figure 20*:* An example where Joy2019 mistakenly identifies the defender as the passer.

To better analyze passing interactions, we measured the average node degree of the passing graphs, which is presented in Table 3. Joy2019 showed a lower mean absolute percentage error (MAPE) in the reconstruction of passing relationships compared to YOLO, with MAPE values up to 33.5%. This lower error rate indicates that Joy2019 better represents player interactions and team dynamics. In addition to the passing relationship analysis, we also used a multivariate Eigen-centrality algorithm, Player Centrality (PC), to rate individual player performances. The PC measure was computed using the NodeRank algorithm to give a more elaborate rating of the contribution of each player to the game. The calculation of PC is formulated as a weighted sum of various play actions, as expressed in Equation 4:

$$R(v_x) = \sum_{i\in\Psi} \alpha_i R_i(v_x) \quad (10)$$

Here, Ψ denotes the set of performance dimensions (e.g., successful passes, interceptions, influence in pass chains); $R_i(v_x)$ is the normalized rating value of player v_x under the i-th criterion; and $\alpha_i \in [0,1]$ is the corresponding weight indicating the relative importance of that criterion in the overall centrality calculation. The weights α_i were selected empirically to emphasize high-impact actions, while ensuring that $\sum_{i\in\Psi} \alpha_i = 1$.The rewards that were assigned to play actions were adapted according to their consequences on the team's overall performance. For example, a reward of 0.3 was assigned to passing the ball, a positive reward of 0.7 for ball interceptions, and a negative reward of -0.7 for being intercepted. The PC values were further refined using a damping factor d, set to 0.85, consistent with PageRank [20]. This adjustment considered indirect player involvement in play actions. The choice of reward values (e.g., 0.3 for passes) was guided by empirical evaluation of how these actions contribute to meaningful pass network structures. To assess robustness, we tested alternative configurations and observed that the original weights preserved more consistent centrality rankings compared to manually evaluated benchmarks. A more detailed ablation analysis remains a promising extension for future research. The normalized PC scores are summarized in Table 4 and 5. Figures 18, and 19 visualize the PC values where the larger the node size, the higher the PC value. In the graphs, solid arrows represent successful passes while dotted arrows represent intercepted passes. These visualizations will give an intuitive sense of player contribution and team strategy. Despite the benefits of Joy2019, several issues still remained. For instance, Figures 16 and 19 show that YOLO was not able to reconstruct pass relationships accurately because of its poor jersey number detection ability. Joy2019, however, sometimes misjudged passing interactions by incorrectly assigning defenders as part of a passing sequence, as illustrated in Figures 16a and 17b. Figure 20 shows an example where Joy2019 mistakenly included a defender in a passing sequence between two teammates. These errors are limitations of two-dimensional video analysis, which lacks the absolute spatial coordinates needed to resolve such ambiguities. These issues could be ameliorated by incorporating depth information or multi-camera setups.

The object detection component of our system is based on YOLOv3, implemented using the Darknet framework. The model was fine-tuned on a custom dataset containing over 16,000 bounding boxes across 1204 annotated basketball frames. Training was conducted for 50 epochs using the Adam optimizer with a learning rate of 0.001 and batch size of 16, employing early stopping based on validation loss. To address identity loss during occlusion and dynamic motion, we developed a custom tracking algorithm called Joy2019. Unlike conventional trackers, Joy2019 assigns tracking IDs by matching detected boxes with those from the preceding ten frames based on spatial proximity and IoU. It also retroactively propagates detected jersey numbers to earlier frames, enhancing tracking consistency. Compared with standard methods such as YOLOv3+DeepSORT and YOLOv3+ByteTrack, Joy2019 demonstrated superior performance, achieving a MOTA of 0.77, MOTP of 0.84, and only 18 ID switches, alongside a jersey recognition accuracy of 73.8% and player detection accuracy of 90.2%. These results highlight the robustness and domain adaptation of our approach for real-time basketball video analysis.

5 Conclusion

This paper proposes a new system for the real-time recognition of basketball players and their interactions with respect to other players, including passes and interceptions, using publicly available NBA game footage. By using YOLO for object detection and classification, we could implement a very robust player tracking algorithm that performs well compared to YOLO and, considering the challenges of camera angle and overlapping players, has by far outperformed many precise tracking systems. Our system effectively recovered missing track information by incorporating movement history from preceding frames and applied network science to analyze the passing relationships among players, assessing importance with a multivariate Eigen-centrality measure. Although the proposed approach showed limitations due to inherent challenges in state-of-the-art deep learning methods, it provided us with valuable insights for enhancing fully automated sports analytics systems. These findings have provided further details on areas of possible

improvements in the future, namely, accuracy and detection of more complex player actions like shooting, scoring, rebounds, and missed shots.

To support reproducibility and facilitate future research, we intend to publicly release the following upon acceptance of this paper: (1) the Joy2019 tracking algorithm implementation, (2) a subset of annotated NBA video frames used in our experiments, and (3) pretrained YOLOv3 weights fine-tuned on our dataset. These resources will be made available via GitHub for non-commercial academic use.

6 Limitations and future work

Despite the system's strong performance, several limitations remain. First, although Joy2019 addresses temporary occlusion using temporal backtracking, it does not incorporate appearance-based re-identification or multi-view fusion, which limits performance in crowded scenes. Second, identity tracking errors persist, especially during close player interactions. We observed 18 identity switches across the annotated evaluation set, and Figures 11–14 illustrate some typical failure cases. Third, while our Player Centrality metric integrates multiple behavior-based scores, it has not yet been benchmarked against traditional network centrality measures such as betweenness and closeness. This will be considered in future comparative analyses.

Finally, our current architecture extends YOLO spatially using manual temporal memory. In future work, we aim to incorporate transformer-based spatio-temporal models, such as TimeSformer or ST-GCN, to enhance contextual understanding of player trajectories and movement patterns across frames.

References

[1] Khan, A.A., Shao, J., Ali, W. and Tumrani, S., 2020. Content-aware summarization of broadcast sports videos: an audio–visual feature extraction approach. *Neural Processing Letters*, *52*(3), pp.1945-1968. DOI: https://doi.org/10.1007/s11063-020-10200-3

[2] Rahimian, P. and Toka, L., 2022. Optical tracking in team sports: A survey on player and ball tracking methods in soccer and other team sports. *Journal of Quantitative Analysis in Sports*, *18*(1), pp.35-57. DOI: https://doi.org/10.1515/jqas-2020-0088

[3] Xu, T. and Tang, L., 2021. Adoption of machine learning algorithm-based intelligent basketball training robot in athlete injury prevention. *Frontiers in Neurorobotics*, *14*, p.620378. DOI: https://doi.org/10.3389/fnbot.2020.620378

[4] Ong, P., Chong, T.K., Ong, K.M. and Low, E.S., 2022. Tracking of moving athlete from video sequences using flower pollination algorithm. *The Visual Computer*, pp.1-24. DOI: 10.1007/s00371-022-02423-3

[5] Zong, H., Chen, B., Li, D., Liu, C. and Cai, Y., 2025. A Multitask Framework for Optimizing Smart Grid Energy Consumption Using RegClassXNet and Dynamic Cluster Adjustment. *Informatica*, *49*(18). DOI: https://doi.org/10.31449/inf.v49i18.7865

[6] Sano, Y. and Nakada, Y., 2021, October. Visualization for potential pass courses and quantification for offensive and defensive players in basketball. In *2021 International Conference on Engineering and Emerging Technologies (ICEET)* (pp. 1-6). IEEE. DOI: 10.1109/ICEET53442.2021.9659697

[7] Sano, Y. and Nakada, Y., 2019, December. Improving prediction of pass receivable players in basketball: simulation-based approach with kinetic models. In *Proceedings of the 10th International Symposium on Information and Communication Technology* (pp. 328-335). DOI: 10.1145/3368926.3369697

[8] Bu, X., 2023. Exploration of intelligent coaching systems: The application of Artificial intelligence in basketball training. *Saudi Journal of Humanities and Social Sciences*, *8*(09), pp.290-295. DOI: 10.36348/sjhss. 2023.v08i09.007

[9] Shen, L., Tan, Z., Li, Z., Li, Q. and Jiang, G., 2024. Tactics analysis and evaluation of women football team based on convolutional neural network. *Scientific Reports*, *14*(1), p.255. DOI: 10.1109/ICESC51422.2021.9532741

[10] Rakhsith, L.A., Anusha, K.S., Karthik, B.E., Nithish, D.A. and Kumar, V.K., 2021. A survey on object detection methods in deep learning. In *Proc. of 2021 Second Int. Conf. on Electronics and Sustainable Communication Systems (ICESC)*. DOI: 10.1007/s11036-024-02299-8

[11] Cheng, X., Liang, L. and Ikenaga, T., 2022. Automatic data volley: game data acquisition with temporal-spatial filters. *Complex & Intelligent Systems*, *8*(6), pp.4993-5010. DOI: 10.1109/ICSECE61370.2024.00041

[12] Zhang, J. and Tao, D., 2023. Research on deep reinforcement learning basketball robot shooting skills improvement based on end-to-end architecture and multi-modal perception. *Frontiers in Neurorobotics*, *17*, p.1274543. DOI: 10.4018/IJITWE.333895

[13] Gong, Y. and Srivastava, G., 2024. Real-Time Tracking of Basketball Trajectory Based on the Associative MCMC Model. *Mobile Networks and Applications*, pp.1-13. DOI: https://doi.org/10.1007/s11036-024-02358-0

[14] Wang, J., Li, Z., Li, Y., Yang, S., Wang, B. and Li, H., 2023. SKT-MOT and DyTracker: A Multiobject Tracking Dataset and a Dynamic Tracker for Speed Skating Video. *Scientific Programming*, *2023*(1), p.3895703. DOI: https://doi.org/10.1155/2023/3895703

[15] Liang, L., 2024, August. Dynamic Tracking Optimization of Basketball Trajectory Based on Graph Neural Network. In *2024 IEEE 2nd International Conference on Sensors, Electronics and Computer Engineering (ICSECE)* (pp. 197-201). IEEE. DOI: 10.1109/ICSECE61370.2024.00041

[16] Feng, Y. and Sun, H., 2023. Basketball Footwork and Application Supported by Deep Learning Unsupervised Transfer Method. *International Journal of Information Technology and Web Engineering (IJITWE)*, *18*(1), pp.1-17. DOI: 10.4018/IJITWE.333895

[17] Cheng, X. and Li, Z., 2024, June. Research on basketball shooting action recognition and optimization system based on deep learning. In *2024 2nd International Conference on Mechatronics, IoT and Industrial Informatics (ICMIII)* (pp. 546-551). IEEE. DOI: 10.1109/ICMIII62623.2024.00108

[18] Kavitha, K.R., Almusawi, M., Vybhavi, G.Y., Prabhakar, P.E. and Saranya, N.N., 2024, October. Predictions of Basketball Match Results using Autoencoder with Gated Recurrent Unit Algorithm. In *2024 First International Conference on Software, Systems and Information Technology (SSITCON)* (pp. 1-5). IEEE. DOI: 10.1109/IPEC61310.2024.00015