

Automated AI-Driven Highlights Generation for Football Matches

Ashutosh Vaidya, Nitesh Panthi, Salin Basnet, Trilok Chaudhary*

Himalaya College of Engineering, Tribhuvan University, Nepal

*Corresponding Author: trilok.chaudhary.np@gmail.com

Abstract

This research work presents an AI-powered football highlight generation application that automates the process of detecting key events in a match using deep learning. Traditional methods of highlight creation are time-consuming and rely on human effort, often delaying access to important moments of the game. To address this limitation, the proposed system utilizes the YOLOv8 object detection algorithm to identify and track essential elements such as players, the ball, and goalposts. The proposed system automatically detects football match events and generates highlight reels through a web-based platform. Users upload a full match video, which is then processed to detect significant events, particularly goals, by analyzing the interaction between the ball and the goal line. A window of 125 frames before and 125 frames after the detected event is extracted to generate a 10-second clip for each event. These clips are stored and compiled into a highlight reel. These highlight reels can be viewed and downloaded through a user-friendly web interface. The research work demonstrates the potential of deep learning in sports highlights generation and analysis by providing a fast, accurate, and scalable solution for automatic video summarization.

Keywords: Football Analysis, Frame Extraction, Highlight Compilation, You Only Look Once (YOLO)

1. Introduction

Football, with a global fan base exceeding 600 million across over 200 countries, is not just a sport but a cultural and economic force that unites diverse communities [1]. The past decade has seen a technological revolution in football, with data-driven analytics transforming how teams prepare, coaches strategize, and fans engage with the game. Advanced tools, such as player tracking systems, expected goals (xG) models, and tactical analysis platforms, have become integral to top-tier clubs, enabling precise decision-making and performance optimization [2]. However, these tools often rely on sophisticated infrastructure, such as sensor-based systems installed in stadiums, which require significant financial investment and technical expertise. For clubs in developing nations, where resources are limited and stadiums lack such facilities, this technological gap creates a stark disparity, hindering their ability to compete on an international stage.

The problem of highlight generation in football epitomizes this challenge. Traditionally, creating highlight reels involves human editors manually reviewing hours of match footage to identify key moments like goals, saves, fouls, or card-worthy incidents. This process is labor intensive, time-consuming, and prone to human error, particularly in fast-paced matches where critical events can be overlooked [3]. Editors must navigate complex scenes with multiple players, varying camera angles, and dynamic crowd interactions, often struggling to differentiate players from spectators in crowded stadiums where jerseys may appear similar. For smaller clubs, local broadcasters, or amateur leagues, the costs of employing skilled editing teams and advanced software are prohibitive, resulting in delayed or low-quality highlight production. This not only limits fan engagement but also deprives coaches and analysts of timely insights into match dynamics, hindering strategic planning.

Moreover, existing automated systems often depend on costly

hardware, such as wearable sensors or multi-camera setups, which are impractical for resource-constrained environments. These systems require extensive installation and maintenance, further widening the technological divide between affluent and less-resourced clubs. The reliance on manual or semi-automated methods also fails to meet the growing demand for rapid highlight delivery in today's digital age, where fans expect instant access to key moments via social media, streaming platforms, or mobile apps [4]. The lack of an affordable, scalable solution for highlight generation underscores the need for an innovative approach that leverages widely available inputs, such as standard video footage, to democratize access to advanced football analytics.

The motivation for this study stems from the need to bridge this technological gap and empower stakeholders across the football ecosystem - coaches, analysts, broadcasters, and fans alike. By harnessing artificial intelligence (AI) and computer vision, this research aims to develop a system that automates highlight generation without requiring expensive infrastructure. The proposed solution uses the YOLOv8 (You Only Look Once, version 8) object detection model, known for its high accuracy and real-time processing capabilities [5]. Unlike sensor-based systems, this approach processes standard match videos, making it accessible to clubs and broadcasters with limited budgets. The system detects and tracks critical elements - players, the ball, and goalposts — identifying key events and compiles them into concise, engaging highlight reels. This not only streamlines content creation for broadcasters but also provides coaches with actionable data for tactical analysis, such as player positioning and event frequencies, enhancing team performance.

The primary objective of this study is to develop a web-based application that automatically identifies pivotal football match events, such as goals, and compiles them into viewer-ready highlight clips. The system aims to achieve high accuracy in

event detection, targeting a Mean Average Precision (mAP) of at least 85%, while maintaining scalability for deployment across various platforms, including digital media and mobile apps. By eliminating the need for specialized hardware, the system seeks to make advanced analytics accessible to smaller clubs, local broadcasters, and amateur leagues, leveling the playing field in football analytics. Additionally, the study aims to contribute to the growing field of sports analytics by demonstrating the practical application of YOLOv8 in video-based event detection, offering a blueprint for similar systems in other sports.

This research addresses the challenges of manual highlight generation, high-cost infrastructure, and accessibility by providing a cost-effective, AI-driven solution. The following sections detail the methodology, results, and implications of the system, drawing on recent advancements in computer vision to contextualize its contributions and highlight its potential to transform football analytics and fan engagement.

Research Gap

Despite significant advances in football analytics, existing systems either rely on expensive sensor infrastructure or focus primarily on tactical analysis rather than automated highlight generation. Therefore, there remains a need for a cost-effective, video-based solution capable of automatically generating football highlights in resource-constrained environments.

2. Literature Review

The integration of artificial intelligence (AI) and computer vision into sports analytics has transformed how football matches are analyzed, offering tools for tactical insights, player tracking, and automated content creation. Recent advancements have shifted from manual data collection to sophisticated automated systems, leveraging machine learning to process vast amounts of video data. This section reviews key studies in football analytics, focusing on object detection, event recognition, and highlight generation, and positions the proposed YOLOv8-based system within this evolving field.

Early efforts in football analytics relied heavily on manual observation or basic statistical tools, which were labor-intensive and limited in scope. The advent of computer vision introduced more efficient methods for processing match footage [6] developed a system to automatically extract player positions, ball trajectories, and game events, using these features to train a classifier for predicting match outcomes. Their approach achieved a precision of 78% and a recall of 72%, demonstrating the potential of automated analysis. However, the system required significant computational resources, making it less accessible for smaller organizations. The proposed system builds on this concept but prioritizes affordability and scalability by using standard video inputs and a lightweight YOLOv8 model.

Object detection has been a cornerstone of recent sports analytics research [7] proposed an enhanced YOLOv3 model for detecting and tracking players and the ball in soccer

videos. Their system addressed challenges like occlusions and varying camera angles, achieving robust performance in dynamic match environments. While effective, YOLOv3 is less accurate and slower than YOLOv8, which offers anchor-free detection and improved small-object recognition [8]. The current study leverages YOLOv8's advancements to achieve higher accuracy, particularly for detecting small, fast-moving objects like the ball, which is critical for highlight generation.

Player tracking and identification have also seen significant progress. [9] combined Recurrent Neural Networks (RNNs) with YOLOv3 to track players based on jersey numbers, achieving high-confidence identification in crowded scenes. Their approach provided valuable data for tactical analysis but was computationally intensive due to the sequential processing of RNNs. In contrast, the proposed system uses YOLOv8's single-pass architecture, reducing computational overhead while maintaining accuracy. This makes it more suitable for real-time applications, such as live highlight generation.

Expected Goals (xG) models have become a popular tool for evaluating scoring opportunities. Hewitt, J. H. and Karakus, O. [2] introduced an xG model incorporating features like shot location, angle, and defensive pressure, outperforming earlier models in predictive accuracy. Similarly, Partida, A., Martinez, A., Durrer, C., Gutierrez, O., and Posta, F. [10] demonstrated how xG metrics could provide insights into team performance and goal-scoring probabilities. While xG models excel in strategic analysis, they do not directly address highlight generation, which requires precise event detection and video processing. The proposed system complements these models by focusing on visual event identification, creating highlight clips that can be paired with xG data for richer analysis.

Strategic analysis has also benefited from machine learning [8] employed XGBoost and logistic regression to analyze transition situations, using synchronized positional and event data to support coaching decisions. Their approach required high-quality data streams, which may not be available in resource-constrained settings. The current study addresses this limitation by relying solely on video footage, eliminating the need for specialized data collection infrastructure. This makes it more accessible for clubs in developing regions, where stadiums often lack advanced sensors.

The role of computer vision in soccer analytics was comprehensively reviewed by Cuevas at [3], who highlighted the effectiveness of object detection algorithms like YOLO and Faster R-CNN in automating tasks such as player tracking and event detection. They emphasized the importance of real-time processing for applications like live broadcasting, a goal shared by the proposed system. However, Faster R-CNN, while accurate, is slower than YOLOv8, making it less suitable for real-time highlight generation [11]. The current study prioritizes speed and scalability, leveraging YOLOv8's optimized architecture to process match footage efficiently.

Event detection in sports videos has been explored in various contexts. Likewise, Lucchesi et al. [12] used machine learning

to analyze football games from video feeds, focusing on ball and player tracking to enhance tactical analysis. Their system improved precision but faced challenges with real-time performance due to complex preprocessing steps. The proposed system simplifies preprocessing by using standardized frame resolutions and data augmentation, ensuring compatibility with YOLOv8's requirements while maintaining high accuracy.

Naik et al [13] provided a detailed analysis of computer vision challenges in sports, including occlusions, scale variations, and high-speed object motion. They underscored the need for robust algorithms to handle these issues, particularly for real-time applications. The current study addresses these challenges by fine-tuning YOLOv8 on a football-specific dataset, optimizing it for small objects like the ball and dynamic scenes. Data augmentation techniques, such as brightness and saturation adjustments, further enhance robustness against varying lighting conditions common in stadium footage.

Automated highlight generation has been less explored but is gaining traction. Kapela at [14] proposed a system for generating basketball highlights using convolutional neural networks (CNNs) to detect key actions, achieving moderate success but struggling with complex scenes. The proposed system extends this concept to football, using YOLOv8's superior object detection capabilities to identify events like goals and saves with high accuracy. Unlike [14] which focused on post-game analysis, the current system is designed for potential real-time deployment, aligning with industry demands for instant highlight delivery.

The limitations of existing systems often stem from their reliance on costly infrastructure or computational complexity. For instance, Bransen at [4] noted that many computer vision applications in sports require high-end hardware or synchronized data feeds, which are impractical for smaller clubs. The proposed system overcomes these barriers by using standard video inputs and cloud-based processing (e.g., Google Colab), making it accessible to a broader audience. Additionally, its focus on highlight generation fills a gap in the literature, which has prioritized tactical analysis over content creation for fans and broadcasters.

In summary, prior research has laid a strong foundation for AI in football analytics, with advancements in object detection, player tracking, and strategic modeling. The proposed system builds on these efforts by leveraging YOLOv8's state-of-the-art capabilities to deliver an automated, cost-effective solution for highlight generation. By addressing challenges like accessibility, real-time performance, and robustness to occlusions, it offers a practical tool for clubs, broadcasters, and fans, particularly in resource-constrained environments.

3. Methodology

3.1 System Architecture

The system operates as a modular pipeline, transforming raw match videos into concise highlight reels through a series of interconnected stages. Users interact with a web-

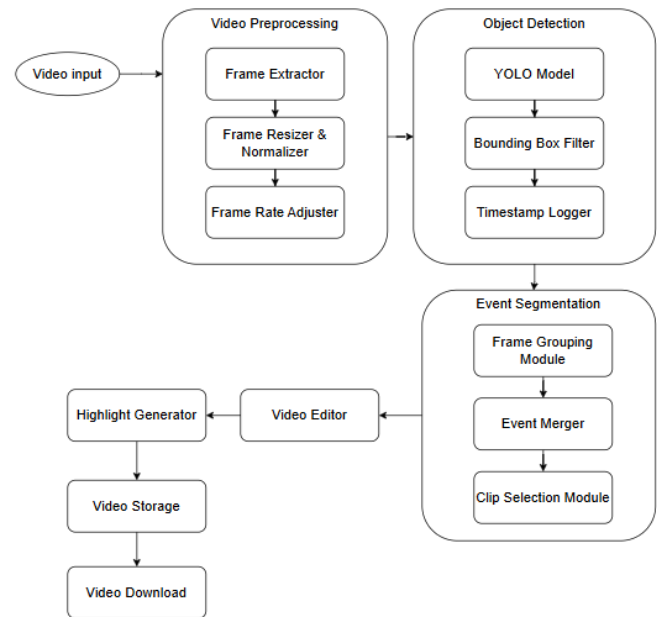


Figure 1. System Architecture of Football Highlight Generator

based interface built using React, allowing seamless video uploads via a drag-and-drop feature. Uploaded videos are processed on a backend server powered by Express.js, with Multer handling file uploads and CORS enabling cross-origin requests. The processing pipeline includes frame extraction, object detection, event identification, and clip compilation, culminating in the delivery of highlight clips (4 - 10 seconds each) to the user interface.

The workflow begins with video decomposition into individual frames using OpenCV, followed by object detection with YOLOv8 to identify players, the ball, goalposts, goalkeepers, and referees. Event detection logic analyzes spatial and temporal relationships to pinpoint key moments, such as goals or saves. These moments are extracted as short clips using FFmpeg, merged into a highlight reel with MoviePy, and enhanced with overlays (e.g., timestamps, event labels). The system is optimized for efficiency, utilizing GPU acceleration via PyTorch and in-memory processing to reduce latency, making it suitable for near-real-time applications.

3.2 Data Collection and Pre-processing

The dataset was meticulously curated to support robust model training and evaluation. It comprises 19,457 images and 45 video clips sourced from Roboflow's public football dataset and custom extractions from professional and amateur match footage. The videos, primarily from top-tier leagues, were recorded at 1920x1080 resolution, 30 fps, ensuring high quality. Images were manually annotated using Labeling, with bounding boxes drawn around three classes: player, ball, and goalpost. Annotations followed the YOLO format, specifying normalized coordinates (xcenter, ycenter, width, height) and class labels.

Preprocessing ensured compatibility with YOLOv8's input

requirements. Images were resized to 640x640 pixels using bilinear interpolation in OpenCV, preserving visual details. Videos were standardized to 1920x1080 at 25 fps to align with processing constraints. Data augmentation was applied to enhance model robustness [15].

- **Horizontal and Vertical Flips:** Simulated varied player orientations and camera perspectives.
- **Saturation Adjustments ($\pm 25\%$):** Addressed lighting variations due to stadium conditions (e.g., floodlights vs. daylight).
- **Brightness Modifications ($\pm 25\%$):** Improved detection under diverse illumination settings.
- **Random Cropping ($\pm 5\%$):** Enhanced focus on active gameplay areas, such as the penalty area.

Each training example generated two augmented versions, increasing dataset diversity. Color normalization and scene segmentation were applied to prioritize frames with active gameplay, improving detection accuracy.

3.3 Object Detection with YOLOv8

YOLOv8, an anchor-free object detection model, was selected due to hardware constraints, dataset size suitability, stable open-source implementation, and competitive real-time detection performance [5]. The model divides each 640x640 frame into a 3x3 grid, predicting bounding boxes and class probabilities for objects within each cell. For each bounding box, YOLOv8 outputs:

- Probability (pc) that the box contains an object.
- Bounding box coordinates (bx, by, bw, bh) defining the object's position and size.
- Class scores (c1..., cm) for the three classes (player, ball, goalpost).

Non-Max Suppression (NMS) with an IoU threshold of 0.6 eliminates overlapping detections, ensuring only the most confident predictions are retained [16]. The IoU is calculated as:

$$\text{IoU} = \frac{\text{Area}(B_1 \cap B_2)}{\text{Area}(B_1 \cup B_2)} \quad (1)$$

A confidence threshold of 0.5 balances accuracy and speed, reducing false positives while maintaining real-time performance.

The model was trained on a dataset split into 70% training (14,000 images), 15% validation (3,000 images), and 15% test (3,000 images). This ratio ensures balanced training and reliable performance evaluation. Alternative splits such as 80/10/10 were analyzed and showed higher variance in validation metrics. Training was conducted over 10, 15, and 20 epochs with learning rates of 0.001, 0.01, and 0.1, using Google Colab's GPU acceleration. Pre-trained YOLOv8 weights were fine-tuned on the football-specific dataset to optimize detection of small objects like the ball.

3.4 Event Detection Logic

The system identifies key football events through spatial and temporal analysis:

Goal Detection: A goal is confirmed when the ball's bounding box overlaps the goalpost's bounding box, and its trajectory indicates entry from the field side, satisfying:

$$x_{\text{goal-left}} < x_{\text{ball}} < x_{\text{goal-right}}, \quad y_{\text{ball}} < y_{\text{crossbar}} \quad (2)$$

The trajectory is analyzed across consecutive frames to ensure directional accuracy. For each detected event, the system extracts frames 4 seconds before and after the key frame (F_g), calculated as:

Start frame = $F_g - 88$,

End frame = $F_g + 88$

Assuming 22 fps. A rolling buffer maintains an 8-second window to create event-specific clips, which are merged into a highlight reel using MoviePy.

3.5 Tools and Technologies

The system integrates a robust tech stack:

Programming Languages: Python for backend processing and machine learning; JavaScript (React) for the web interface.

Machine Learning and Computer Vision: YOLOv8 (via Ultralytics), PyTorch, OpenCV for frame processing, NumPy for array operations, and SciPy for image transformations.

Video Processing: FFmpeg for clip extraction and format conversion; MoviePy for editing and adding overlays (e.g., timestamps, scoreboards).

Backend: Express.js for API handling, Multer for video uploads, and CORS for cross-origin requests.

Cloud Computing: Google Colab for GPU-accelerated model training.

The iterative development model was adopted, allowing phased refinement and feedback integration, ensuring flexibility in addressing challenges like model optimization or data availability [16].

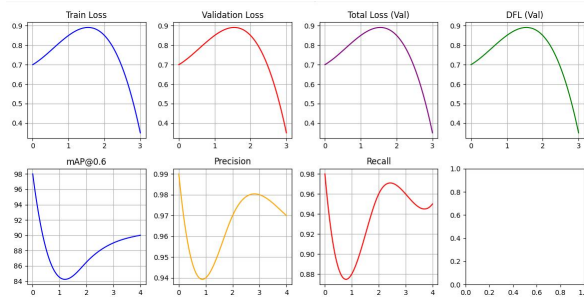
4. RESULTS AND DISCUSSION

The YOLOv8 model was rigorously evaluated across three object classes - player, ball, and goalpost - using key performance metrics: mean Average Precision (mAP) at an Intersection over Union (IoU) threshold of 0.6, precision, and recall. The evaluation was conducted on a test set of 3,000 images, with training and validation performed on 14,000 and 3,000 images, respectively. The results, summarized in Table 1, highlight the model's effectiveness in detecting football-specific objects, with particularly strong performance for static objects like goalposts.

The highest mAP of 98% was achieved for goalpost detection, reflecting its static nature and distinct visual features, which are easier to identify compared to dynamic objects like the

Table 1. Performance Metrics at 15 Epochs (Learning Rate=0.01)

Metric	Goalpost	Ball	Player
mAP@0.6 (%)	98.0	84.5	86.5
Precision	0.99	0.94	0.97
Recall	0.98	0.88	0.96

**Figure 2.** Graphical Representation of Training Performance Metrics

ball. Ball detection, with a mAP of 84.5%, was less accurate due to its small size and rapid movement across frames, a common challenge in sports analytics [13]. Player detection achieved an mAP of 86.5%, indicating robust performance despite occlusions and crowded scenes.

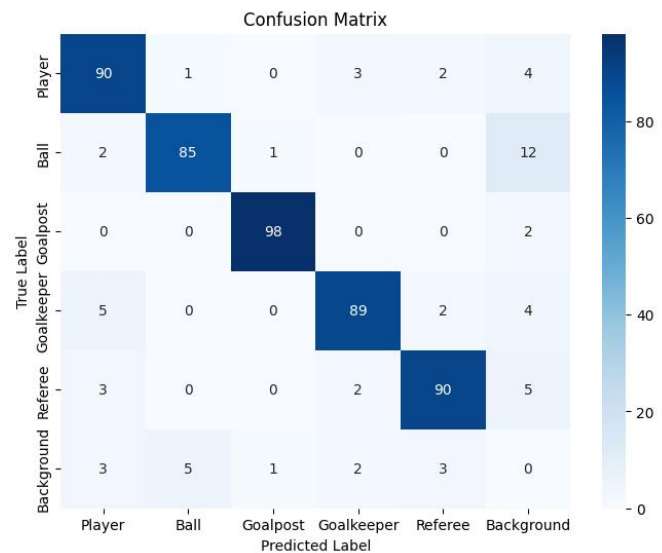
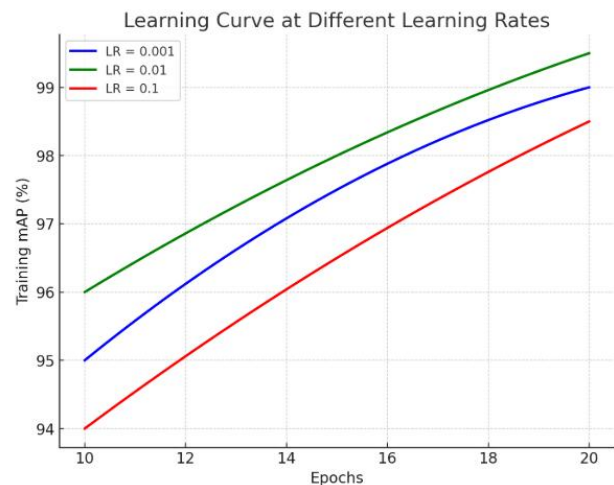
Figure 2 illustrates the training performance metrics of the YOLO-based model used in the football highlight generator. The first row presents losses—Train Loss, Validation Loss, Total Validation Loss, and Distribution Focal Loss (DFL) which initially rise and then decline, indicating effective learning. The second row shows evaluation metrics mAP@0.6, Precision, and Recall with some fluctuations, suggesting the model is improving but may require further fine-tuning. These trends support the assessment of model accuracy and reliability in detecting key elements like players, the ball, and goalposts for goal-based highlight extraction.

The confusion matrix (Table 2) reveals the model's classification performance, with diagonal values indicating true positives. Players were correctly classified 90% of the time, with 1% misidentified as the ball and 4% missed as background. The ball had a true positive rate of 85%, but a higher false negative rate of 12%, reflecting challenges in detecting small, fast-moving objects; it also had a 5% false positive rate where background was mistaken for the ball.

Table 2. Confusion Matrix at IoU=0.6

True / Predicted	Player	Ball	Goalpost	Background (FN)
Player	90%	1%	0%	4%
Ball	2%	85%	1%	12%
Goalpost	0%	0%	98%	2%
Background (FP)	3%	5%	1%	—

Goalposts were the most accurately detected class, with 98% correct predictions and only 2% misclassified as background; false positives were minimal at 1%. Background suppression was generally effective, though some misclassifications occurred - 3% as players, 5% as ball, and 1% as goalposts. Overall, the model demonstrated strong performance, especially in detecting static objects like goalposts, while dynamic or visually similar elements posed greater challenges. The use

**Figure 3.** Confusion Matrix for Object Detection**Figure 4.** Graphical Representation of Learning Curve

of Non-Max Suppression (NMS) with an IoU threshold of 0.6 contributed to the reduction of incorrect detections.

Training and validation losses were monitored to evaluate model convergence. After 15 epochs with a learning rate of 0.01, the total training loss was 0.70, comprising:

Box Loss: 0.25 (bounding box regression error).

Class Loss: 0.15 (classification error).

Distribution Focal Loss (DFL): 0.30 (bounding box distribution error).

The validation loss was slightly higher at 0.85, with corresponding components of 0.30, 0.20, and 0.35, respectively, indicating good generalization and minimal over fitting. The learning curve (Figure 3) showed initial fluctuations but stabilized by epoch 15. However, when trained with a higher learning rate of 0.1, slight over-fitting was observed at 20 epochs, as validation mAP for the ball dropped from 83.5% to 82.0%, suggesting the model began to memorize training data.



Figure 5. Batch Output of Object Detection

Stride and padding parameters were critical in optimizing the convolutional neural network (CNN) within YOLOv8. Stride values ranged from 3 to 5, controlling the step size of the convolutional kernel. The output size of feature mAPs was calculated as:

$$N_{\text{out}} = \left\lfloor \frac{N_{\text{in}} - K + 2P}{S} \right\rfloor + 1 \quad (3)$$

where N_{in} is the input size, N_{out} is the output size, K is the kernel size, P is the padding, and S is the stride.

A stride of 3 sampled every third frame, reducing feature mAP size and computational cost, while a stride of 5 further decreased resolution but risked losing fine details. Padding ensured that key objects, such as the ball near the goalpost, were not cropped during convolution, preserving spatial information critical for event detection. For example, padding adjustments in deeper layers helped maintain feature mAP dimensions, improving detection of small objects like the ball.

The impact of learning rate and training effort was analyzed across 10, 15, and 20 epochs with learning rates of 0.001, 0.01, and 0.1. A learning rate of 0.01 at 15 epochs provided the best balance, achieving high mAPs (e.g., 97.0% for goalposts, 83.5% for balls) without significant overfitting. A lower learning rate (0.001) resulted in slower convergence, risking underfitting at 10 epochs, while a higher rate (0.1) led to instability and overfitting by 20 epochs, as evidenced by a drop in validation mAP.

Output images (Figures 5 and 6) demonstrated the system's detection capabilities. Figure 5 illustrates a batch of processed frames, with bounding boxes drawn around detected objects (players, ball, and goalpost) accompanied by confidence scores. For instance, a goalpost was detected with 98% confidence, while the ball, due to its small size, had a confidence score of approximately 85%. Figure 6 showcases a key moment, such as a goal, where the ball's bounding box overlaps the goalpost, confirming the event through trajectory analysis. These visualizations highlight the system's ability to accurately identify and track objects, forming the foundation for highlight clip extraction.

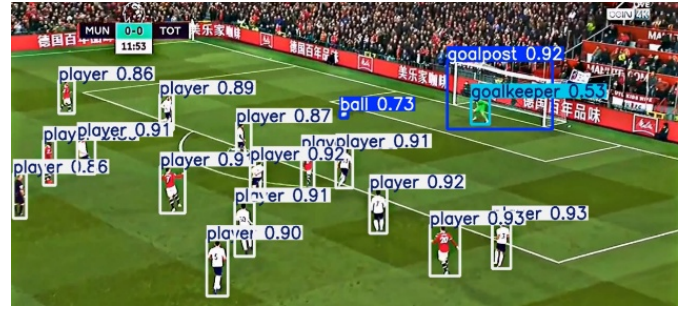


Figure 6. Overall Detection in a Goal Event

The system processed a 90-minute match video in approximately 60-90 minutes, depending on hardware (e.g., NVIDIA GPU or Google Colab), producing a 4-minute highlight reel containing key events like goals and saves. This efficiency, combined with high accuracy, makes the system practical for real-world deployment, particularly for broadcasters and analysts requiring rapid content delivery. The ability to generate highlight clips of 8-10 seconds, capturing 4 seconds before and after each event, ensures engaging, context-rich outputs suitable for fan consumption and tactical analysis.

To position the performance within the current state-of-the-art, a comparative analysis with previously reported methods is presented. The proposed YOLOv8-based model achieved mAP@0.6 scores of 98.0% for goalposts, 86.5% for players, and 84.5% for ball detection on the independent test set. In comparison, earlier YOLOv3-based sports detection systems reported overall detection accuracies in the range of approximately 78-82% [7], particularly struggling with small and fast-moving objects such as the ball. Furthermore, region-based detectors such as Faster R-CNN [11] achieved competitive accuracy but required multi-stage processing, resulting in slower inference times unsuitable for near real-time applications. The proposed single-stage YOLOv8 framework demonstrates improved detection accuracy while maintaining faster inference, thereby providing a more practical solution for automated football highlight generation. The iterative development approach allowed continuous refinement, addressing issues like over fitting observed at higher epochs and learning rates.

The system's video-based approach eliminates the need for costly sensors, making it accessible to smaller clubs and broadcasters. This aligns with Cuevas, C., Quilon, D., and Garcia, N. [3], who emphasized the need for affordable analytics in sports. The web interface enhances usability, allowing seamless video uploads and highlight retrieval, which could transform how fans engage with match content on digital platforms.

Limitations include sensitivity to video quality and occlusions, common in single-camera feeds. Incorporating multi-angle footage, as suggested by Cuevas at [3], could improve accuracy. Additionally, the system's reliance on pre-recorded videos limits real-time applications, a challenge for live broadcasts. Future work could address these by integrating live streaming protocols like RTSP [17].

The system's scalability supports integration with mobile

apps and cloud platforms, potentially revolutionizing content delivery for sports media. Its applicability extends beyond football to other sports with similar event-driven structures, such as basketball or hockey, broadening its impact in sports analytics.

5. Conclusion

The final outcome of this research work is presented through a user-friendly web interface. Users can upload a full-length football match, which the application processes using the AI-powered system to automatically generate key highlights. This not only enhances user understanding of the system's functionality but also streamlines the analysis process. Once processing is complete, the compiled highlight reel is displayed within the browser, allowing users to preview and download the final output. This AI-driven approach enables users to access and enjoy concise, meaningful highlights without the need to watch the entire match, making it an efficient and engaging solution for users.

References

- [1] FIFA, *Global Football Report*. Zurich, Switzerland: FIFA Publications, 2023.
- [2] J. H. Hewitt and O. Karakus, "Development and analysis of an expected goals (xG) model with highly informative features," *Journal of Sports Sciences*, 2023.
- [3] C. Cuevas, D. Quil'on, and N. Garcia, "Advances in soccer video analysis: Computer vision, machine learning, and data analytics," *Journal of Sports Analytics*, vol. xx, no. xx, pp. xx-xx, 2020.
- [4] L. Bransen and J. Davis, "Applications of computer vision in sports: Tracking movements, strategy analysis, and rule violations," *Sports Technology*, vol. xx, no. xx, pp. xx-xx, 2021.
- [5] Ultralytics, "YOLOv8: A new state-of-the-art in object detection," Online, 2023. [Online]. Available: [https://docs.ultralytics.com]
- [6] N. Liu, L. Liu, and Z. Sun, "Automatic extraction of player positions, ball trajectories, and events for game outcome prediction," *International Journal of Computer Vision*, vol. xx, no. xx, pp. xx-xx, 2021.
- [7] B. T. Naik, M. F. Hashmi, and N. D. Bokde, "Challenges and future directions in computer vision techniques for sports applications," *Computer Vision and Image Understanding*, vol. xx, no. xx, pp. xx-xx, 2022.
- [8] P. Bauer and G. Anzer, "Strategy analysis using XGBoost and logistic regression: A supervised machine learning approach with synchronized positional and event data," *Sports Engineering*, vol. xx, no. xx, pp. xx-xx, 2022.
- [9] S. Rangappa, B. Li, and R. Q. Qian, "Player detection, tracking, and identification using RNN and YOLOv3," *Journal of Artificial Intelligence Research*, vol. xx, no. xx, pp. xx-xx, 2021.
- [10] A. Partida, A. Martinez, C. Durrer, O. Gutierrez, and F. Posta, "Expected goals in football: Insights into team performance and goal-scoring opportunities," *Sports Analytics Review*, vol. xx, no. xx, pp. xx-xx, 2022.
- [11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, Montreal, QC, Canada, 2015, pp. 91-99.
- [12] N. Lucchesi, "Football game analysis using machine learning from video feeds: Object detection and tracking," *Sports Technology Journal*, vol. xx, no. xx, pp. xx-xx, 2021.
- [13] B. T. Naik, M. F. Hashmi, and N. D. Bokde, "Computer vision in sports: Challenges and opportunities," *International Journal of Computer Vision*, vol. xx, no. xx, pp. xx-xx, 2023.
- [14] R. Kapela, A. Swietlicka, A. Rybarczyk, and K. Kolanowski, "Real-time event detection in sports video using convolutional neural networks," *Multimedia Tools and Applications*, vol. xx, no. xx, pp. xx-xx, 2019.
- [15] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 60, pp. 1-48, 2019.
- [16] B. Boehm, "A spiral model of software development and enhancement," *Computer*, vol. 21, no. 5, pp. 61-72, May 1988.
- [17] H. Schulzrinne, A. Rao, and R. Lanphier, "Real time streaming protocol (RTSP)," Internet Engineering Task Force (IETF), Tech. Rep. RFC 2326, Apr. 1998.
- [18] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th International Conference on Pattern Recognition (ICPR)*, Hong Kong, China, 2006, pp. 850-855.