International Journal of Innovative Computing, Information and Control
Volume **22**, Number **1**, February **2026**

ICIC International ©2026 ISSN 1349-4198
pp. **109–128**

ANTHROPOMETRIC BODY ZONE-BASED SOCCER PASS DETECTION AND TEAM CLASSIFICATION WITH INTEGRATED COMPUTER VISION MODELS

Namjildorj Batbaatar[1,2], Stephen Karungaru[1], Kenji Terada[1]
Munkhbat Gantumur[1] and Altangerel Ayush[2]

[1]Department of Information Science and Intelligent Systems
Faculty of Engineering
Tokushima University
2-1 Minami-josanjima, Tokushima 770-8506, Japan
{ c612345003; c612345001 }@tokushima-u.ac.jp; { karunga; terada }@is.tokushima-u.ac.jp

[2]Department of Information Technology
Mongolian University of Science and Technology
22th Khoroo, Bayanzurkh 13340, Mongolia
a.altangerel@must.edu.mn

Received June 2025; revised October 2025

Abstract. *Real-time detection and classification of player actions in soccer matches remains challenging due to complex player interactions, frequent occlusions, and the need for accurate team-specific possession analysis. This paper introduces a novel soccer event detection system that achieves 84.51% accuracy at 18.33 FPS through the integration of anthropometric body zone analysis with multiple computer vision models. Our approach combines YOLOv8 for multi-object detection, ByteTrack for tracking, and an innovative unsupervised team classification method using SigLIP embeddings with UMAP-accelerated dimensionality reduction and K-means clustering. To resolve ambiguities in ball-player association, we develop a principled anthropometric approach that segments player bounding boxes into upper and foot zones based on human biomechanical ratios, supplemented by ResNet-based CNN classifiers for complex occlusion scenarios. The system detects comprehensive soccer events including standard passes (84.36% accuracy), headed passes (86.05% accuracy), throw-ins (87.5% accuracy), and chest control interactions – capabilities not addressed in previous related works. Experimental validation on diverse soccer footage demonstrates superior performance compared to existing methods, with improvements of 2.41-10.81% over state-of-the-art approaches while maintaining real-time processing capabilities. The anthropometric body zone analysis represents a novel application of human biomechanics to sports video analysis, enabling robust ball-player association across varying camera angles and player poses. Our GPU-optimized pipeline, featuring caching mechanisms, achieves computational efficiency suitable for live match analysis, processing standard broadcast video with minimal real-time latency on consumer hardware.*

Keywords: Soccer game, Object detection, Team classification, Pass detection, Event detection, Soccer analysis

1. **Introduction.** At any given moment, a sporting event is happening somewhere in the world. Soccer matches, in particular, are lengthy and continuous, making it uncommon for viewers to watch an entire game through to the end. As a result, highlight summaries – featuring goals, bookings (cards), shots on goal, and penalties – are often prepared. Automating the identification of key events, such as the start of an attack, the number

DOI: 10.24507/ijicic.22.01.109

of passes in a play, goal attempts, and team possession statistics, can significantly reduce the time and cost involved in manual annotation, thereby improving the accuracy and consistency of soccer analytics. These statistics are valuable not only for coaches and teams in planning future tactics but also for enhancing the overall value and utility of the data. However, detecting soccer events in broadcast footage remains challenging due to varying camera angles, frequent occlusions, and the complexity of interpreting temporal context.

Earlier research focused on highlight detection using broadcast cues or basic audio-visual heuristics. One of the most common and strategically significant events in soccer is a pass, which immediately affects ball possession statistics – a measure highly related to game results. A pass is considered valid if it is made between two different players on the same team. To determine the percentage of possession, valid passes for each team are manually tallied [2, 3, 15]. Recently, a growing body of research has been using computer vision and artificial intelligence (AI) algorithms in football analysis. Some examples include soccer ball detection, tracking, and team classification [1, 5, 6, 7, 8]. There is research on the detection of soccer events, such as red and yellow cards [10], dribbling, kicking, running, and walking [11], as well as goals, shots, corners, free kicks, yellow cards, fouls, and corner kicks [4]. These studies detected events, but did not focus on the detailed detection and counting of all team passes. Sarkar et al. [11, 12, 13, 14] have conducted several studies on pass detection. These studies used reinforcement learning to detect passes and possession without specific team labels.

However, their approach requires extensive training, and some models and functions are challenging to implement due to their black-box policies [11, 12, 13, 14]. We seek an alternative that leverages explicit computer vision modules for transparency and is easier to implement. In this paper, we provide a real-time soccer pass detection and possession analysis system that combines high-level game semantics with object-level visual perception. Our system detects players, ball, and referees in each frame, classifies players into teams, identifies which team has possession of the ball and all types of passes (standard passes, headed passes, throw-ins and chest control), showing information about successful passes for each team, and is designed to run on broadcast video.

The remainder of this paper is organized into distinct sections as follows. Section 2 presents a summary of relevant research. Section 3 provides a detailed introduction to the functions, models, and algorithms included in the proposed method, as well as information on the various soccer match datasets used for training. Section 4 describes the experiments and comparisons with existing research. Section 5 introduces the error analysis and possible improvements. The last section gives a summary of the entire study.

2. **Related Work.**

Soccer video analysis research encompasses a range of topics, from recognizing actions in short clips to detecting, recording, and classifying long-form events. The statistical information and data obtained are crucial to optimizing team tactics, offensive strategies, and substitutions. We have divided the research related to our proposed approach into two parts.

2.1. **Player classification and team assigner.**

Recognizing and classifying players in untrained video is crucial for tactical analysis. Ivankovic et al. [23] divided players into two teams according to the color of their jerseys using a combination of object contour analysis, morphological filtering, and background subtraction. Lu et al. [6] proposed a method that uses lightweight CNNs and large labeled datasets for efficient detection. In contrast, Koshkina et al. [5] proposed a different learning method for unsupervised

player classification in sports videos. The process utilizes self-supervised learning techniques to classify players without requiring labeled data, demonstrating high accuracy in player identification and classification. Their results show that the learned representation method, which incorporates both color and spatial information, achieves better team classification results than the pure color method. To perform team classification from soccer matches, Gadde and Jawahar [7] proposed a combination of a pre-trained detection model on domain-specific data, a k-means clustering algorithm to validate player classification, and a transductive model to tune detection parameters. Istasse et al. [8] proposed embedding vectors for predicting the feature vector of each pixel of a player's jersey, which are trained on a convolutional neural network (CNN). In this way, similar vectors are assigned to pixels where players on the same team are located, while dissimilar vectors are assigned to pairs of pixels corresponding to different teams. Lopes and Machado [16] demonstrated the effectiveness of implementing UMAP metric reduction, clustering, and visualization tools to process large amounts of complex data, such as images and videos, on low-performance devices. Namjildorj et al. [29] applied unsupervised learning techniques, combined with pixel segmentation and an improved k-means clustering algorithm, to assigning teams in soccer videos without any labeled data.

2.2. **Event detection and sport analytics.** Traditionally, events during soccer matches are detected using rule-based and learning-based techniques. Event detection depends heavily on the position, speed, and direction of the player and the ball, the occlusion of the players and the orientation of the player's body. Manafifard et al. [1] provided a comprehensive overview of player detection and tracking approaches, which form the backbone of many event-based systems. Opta Sports [2, 3] uses the number of passes by teams to calculate the ball possession statistics. They manually count the number of successful passes by each team during a match. This is a time-consuming and resource-intensive task. Detection has been enhanced by utilizing computer vision and deep learning methods for event detection in soccer videos. To identify events during a soccer match, Karimi et al. [10] proposed a deep learning method that focuses on correctly identifying photos of specific events from other images and differentiating between images of red and yellow cards. Yu et al. [4] proposed a method using a CNN and LSTM based on VGG-16 to identify key events in soccer matches, such as goals, shots, and fouls. They introduced a method to transform the video footage into a sequence of frames and train the network using annotated event labels.

Bose et al. proposed a lightweight student network trained to imitate the teacher's behavior using only RGB inputs, and a high-performance teacher network that processes both RGB frames and optical flow to develop a rich spatiotemporal description. However, it ignores fine-grained tasks like player-level possession and pass detection [11], although it determines the interaction energy based on the proximity, spatial location, and speed of the player and the ball, and detects the action using a CNN-based model, the processing time is 21.8 seconds, which is difficult to implement in real time [12], dual-agent reinforcement learning framework where first agent ("Watch") identifies possession events and the second ("Act") refines these predictions by evaluating match condition [13], players and the ball tracking results are modeled as a minimum-cost flow (MCF) network to formulate the graph-theoretic formulation of possession tasks [14]. Link and Hoernig presented a technique for calculating individual-level possession based on players' closeness and movement patterns in relation to the ball [15].

Recent approaches to sports video analysis have focused on improving temporal precision in action and event spotting. Rongved and Stige [9] implemented a machine learning method for event detection (also called spotting) and event classification of soccer match

events by combining audio and visual features extracted from soccer videos. Some researchers have proposed real-time event detection with low latency. Cioppa et al. [25] proposed a context-aware loss function (CALF) that considers explicitly the temporal context around each action to detect soccer events temporal spotting and validated it on large datasets such as SoccerNet. Hicks et al. [26, 27] introduced a combination of 3D convolutional neural networks and sliding window methods to detect events such as goals, yellow/red cards, and substitutions. To improve the accuracy, adaptability, and universality of sports action recognition and evaluation, and to increase the possibility of providing technical support for athletes' training, Hu and Liu developed a method combining DTW and KNN algorithms [32]. Fan and Sun introduced a novel fitness coaching method that combines Kinect technology with K-means and DTW to recognize and detect fitness qigong movements, recognize gymnastic posture characteristics, and evaluate the correct performance of exercises. The Static K-means algorithm (SK-m) in this study classified some movements with 95-100% accuracy [33]. Liu et al. developed a novel convolutional neural network (CNN) based on the VGG16 architecture to detect pedestrian standing, squatting, bending, and walking actions from infrared camera data. The method was able to identify human actions in outdoor environments at night, even under varying ambient temperatures and lighting conditions [34].

Transformer architectures have been adopted in recent SoccerNet challenges to detect soccer action spotting. ASTRA (An Action Spotting Transformer for Soccer Videos), proposed by Xarles et al., tackles the challenge of accurately localizing soccer events, such as goals, fouls, throw-ins, clearances, kick-offs, penalties, and substitutions. This approach employs a transformer encoder-decoder architecture similar to DETR. It incorporates various innovations to address common challenges in action detection, including long-tail data distribution, label noise, and occluded events. With an average mAP (density) of 70.21% on the SoccerNet challenge set, it secured the third position in the SoccerNet 2023 Action Spotting Challenge [35]. Put forth by Xarles et al., the T-DEED (Temporal-Discriminability Enhancer Encoder-Decoder) framework presents a new approach for accurately localizing events in sports videos. T-DEED enables more precise event detection than traditional approaches by enhancing temporal discriminability through the use of an encoder-decoder structure. The model effectively addresses challenges such as ambiguous temporal boundaries and overlapping actions, which are prevalent in fast-paced sports like soccer and basketball. This research secured first place in the 2024 SoccerNet Ball Action Spotting Challenge [36]. Deniz et al. presented COMEDIAN [37], which enhances temporal transformers through self-supervised pretraining and knowledge distillation, following this approach. On SoccerNet-v2, COMEDIAN detects actions with an average-mAP of 73.1%. Although such transformer-based methods yield high accuracy, their computational cost limits their use in real-time applications, motivating more lightweight alternatives.

To summarize, numerous researchers have developed various models and conducted studies focused on detecting and assessing actions in sports activities, including soccer matches. However, related research still has some shortcomings. For example, the accuracy of existing methods for detecting fine-grained passes still needs improvement. Furthermore, most of them require large amounts of memory and annotated data to be used in practical applications. Since soccer teams wear different jerseys for each match, preparing labeled data for every game can be time-consuming and labor-intensive. Therefore, this study proposes a method for computing passing statistics with low latency in soccer matches, combining YOLOv8 object detection, ByteTrack, K-means unsupervised team classification, UMAP, SigLIP, ResNet, and CNN, which can be implemented without requiring large memory or annotated data.

3. Methodology.

3.1. **System overview and pipeline architecture.** Our system comprises several modules that aim to sequentially process input video frames for soccer pass detection, tracking, and statistics collection with minimal real-time loss. As illustrated in Figure 1, the pipeline functions as follows: for each frame, players, the ball, the goalkeeper, and the referee are detected and tracked over time. Using an unsupervised classification module, team affiliation is determined based on the color of the players' jerseys. Subsequently, the player possessing the ball is identified, and successful passes are detected and aggregated for each team. Each module in the pipeline is described in detail in the following subsections.

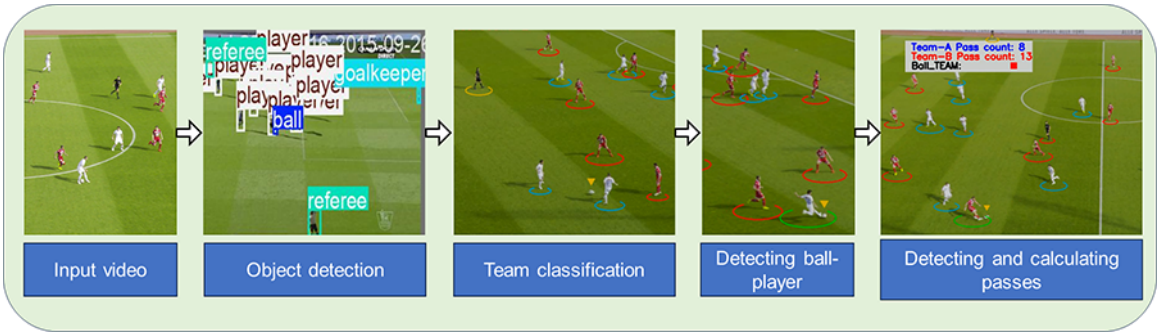

FIGURE 1. A general diagram of the proposed method

3.2. **Multi-object detection and tracking.** Accurately detecting and tracking players, balls, referees, and goalkeepers on the field is a crucial prerequisite for identifying soccer events. The YOLO model, first introduced by Redmon et al. [28], demonstrated that this approach significantly improves object localization and classification accuracy. We apply the YOLOv8 model to detecting objects in the soccer match and the ByteTrack [20] integration to track their positions. By combining YOLO detection with ByteTrack tracking, we can create continuous player trajectories, which are essential for later team classification and ball-player interaction analysis. We utilized images from the open-source Roboflow [45, 46] to train our model. To increase the data size, we selected videos from SoccerNet and Bundesliga and annotated them using the Roboflow tool, resulting in a total of 9300 images. Our training data consists of 4 classes: ball, player, goalkeeper, and referee. The model was trained using the Yolov8x version, with a batch size of 8, for 1000 epochs, an image resolution of 640 pixels, and the training data was divided into 70% for Training, 15% for Testing, and 15% for Validation. The accuracy of our multi-object detection model is illustrated in Figure 2.

3.3. **Unsupervised team classification.** Accurately classifying players on the field is essential for extracting team-related statistics, such as passing networks, attacking patterns, and possession percentages. High-resolution images consume substantial storage and memory resources, slowing down processing. Reducing image size while preserving essential visual features can improve efficiency in image processing tasks. We utilized a pre-trained image embedding model [21] to convert player jersey image vectors into compact input vectors that retain key features. SigLIP embeddings are typically 512-768 in size, which can be too high-dimensional for simple clustering algorithms like K-means. Therefore, these vectors are then subjected to dimensionality reduction via UMAP [16, 18], followed by clustering with K-means [5, 8, 10, 29], classifying players into two distinct

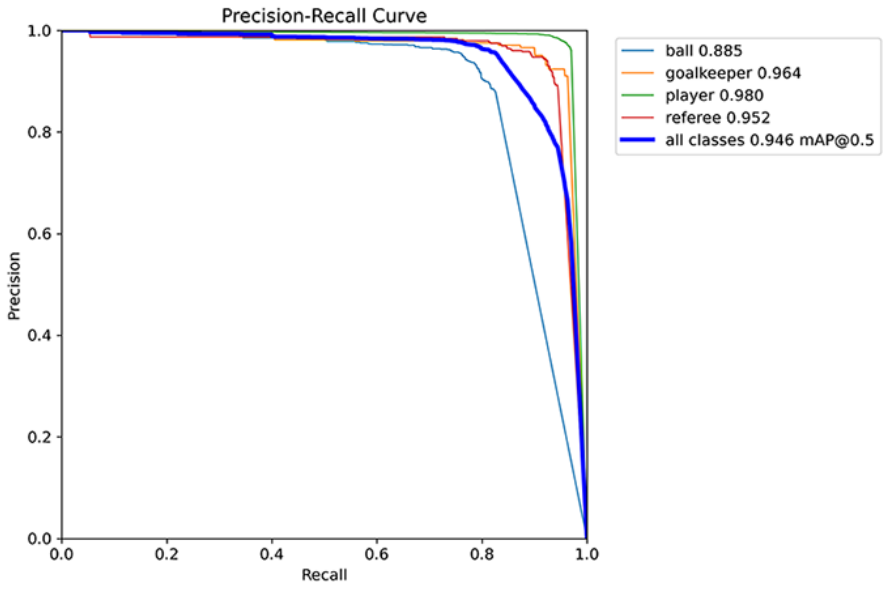

FIGURE 2. (color online) Model = Yolov8x, batch = 8, epoch = 1000

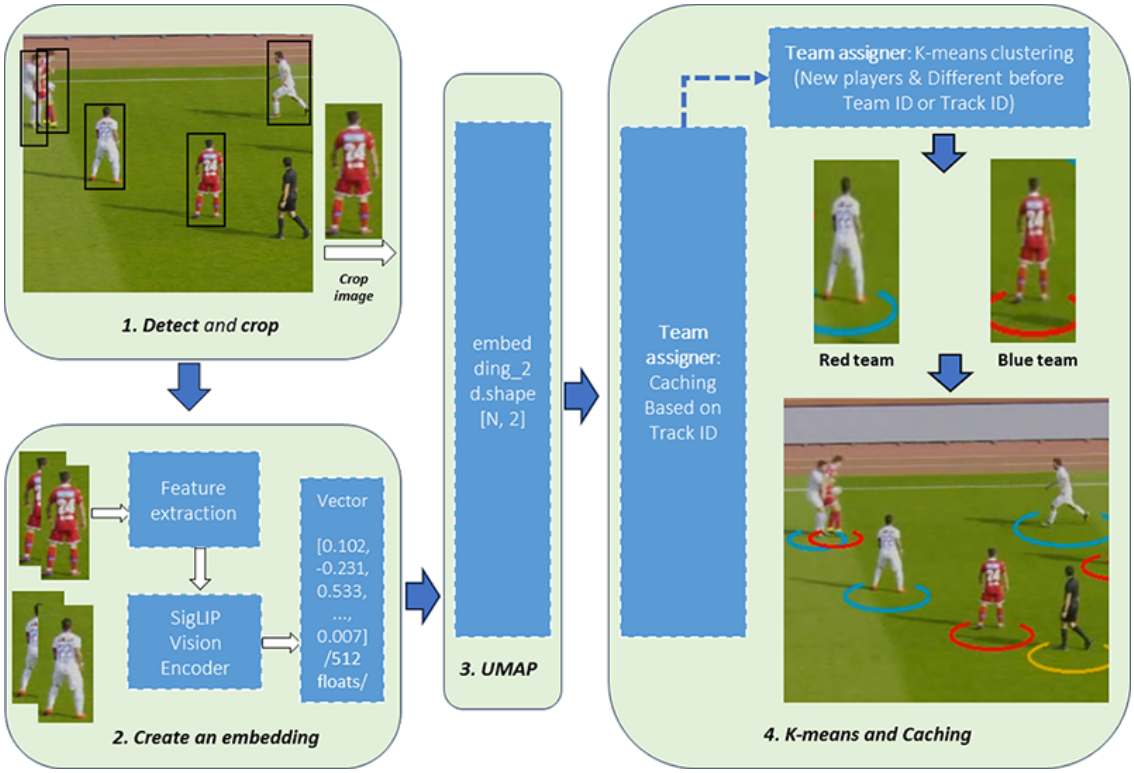

FIGURE 3. Structure of object detection team assignment method

clusters corresponding to their respective teams. The structure of the proposed object detection and team assignment method is shown in Figure 3.

First, the YOLOv8 model detects players and crops each player's image using their bounding box coordinates. Second, each cropped player image is passed through the SigLIP image encoder, producing a 512-dimensional embedding vector that represents the visual features of the player's jersey. Next, embeddings are reduced to two dimensions

using the UMAP algorithm. Finally, players are classified into two teams based on visual similarity using K-means Caching. The feature embedding is expressed as Equation (1). Each input image x is first encoded into a high-dimensional representation using the SigLIP encoder:

$$z = f_\theta(x), \quad z \in \mathbb{R}^{512}, \tag{1}$$

where f_θ denotes the SigLIP model parameters, and z is the resulting 512-dimensional embedding vector. The dimensionality reduction is expressed in Equation (2) and Equation (3). UMAP constructs a weighted graph based on neighborhood relationships:

$$w_{ij} = \exp\left(-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}\right), \tag{2}$$

where $d(x_i, x_j)$ is the Euclidean distance between embeddings, and ρ_i, σ_i control local connectivity. The low-dimensional embedding $y_i \in \mathbb{R}^2$ is obtained by minimizing the cross-entropy loss (3):

$$L = \sum_{i \neq j} \left[w_{ij} \log \frac{1}{1 + \|y_i - y_j\|^2} + (1 - w_{ij}) \log \left(1 - \frac{1}{1 + \|y_i - y_j\|^2}\right)\right]. \tag{3}$$

The team clustering using K-means is expressed as Equations (4) and (5). The reduced embeddings y_i are clustered into two groups using K-means ($k = 2$). Each sample is assigned to the nearest cluster center:

$$c_i = \arg\min_{j \in \{1,2\}} \|y_i - \mu_j\|, \tag{4}$$

where c_i denotes the cluster label of sample i, and μ_j is the centroid of cluster j. The centroids are updated iteratively as

$$\mu_j = \frac{1}{|C_j|} \sum_{i \in C_j} y_i. \tag{5}$$

Moreover, as a team's attack typically begins with the goalkeeper, we adopted a method in [31] for assigning goalkeepers to teams.

3.4. **Ball-player association.** During a soccer match, accurately identifying the player controlling the ball is one of the difficult challenges. We suggest a method that employs a machine learning model to address this challenge. This model makes decisions based on the distance between the ball and the player, as well as the zone of the player's body in contact with the ball. The method is divided into Anthropometric body region analysis, Distance-based initial task, and CNN-based occlusion solution.

3.4.1. *Anthropometric body region analysis.* We propose a method to identify the player possessing the ball by measuring the distance between the ball's bounding box, detected by YOLOv8, and specific zones of the player's bounding box, namely the upper zone and the foot zone. Since a player typically interacts with the ball using two primary body zones – such as the head, hands, chest (grouped as the upper zone), and foot – we divide the player's bounding box into the upper zone and foot zone using the anthropometric body region division [30].

Each player is represented by a bounding box with the top coordinates $BB[y]$ and height $BB[h]$. This formulation ensures that the ball's vertical position relative to the player's body is consistent with human body proportions, thus allowing reliable classification of ball possession zones. Figure 4 illustrates the algorithm used to determine the player's

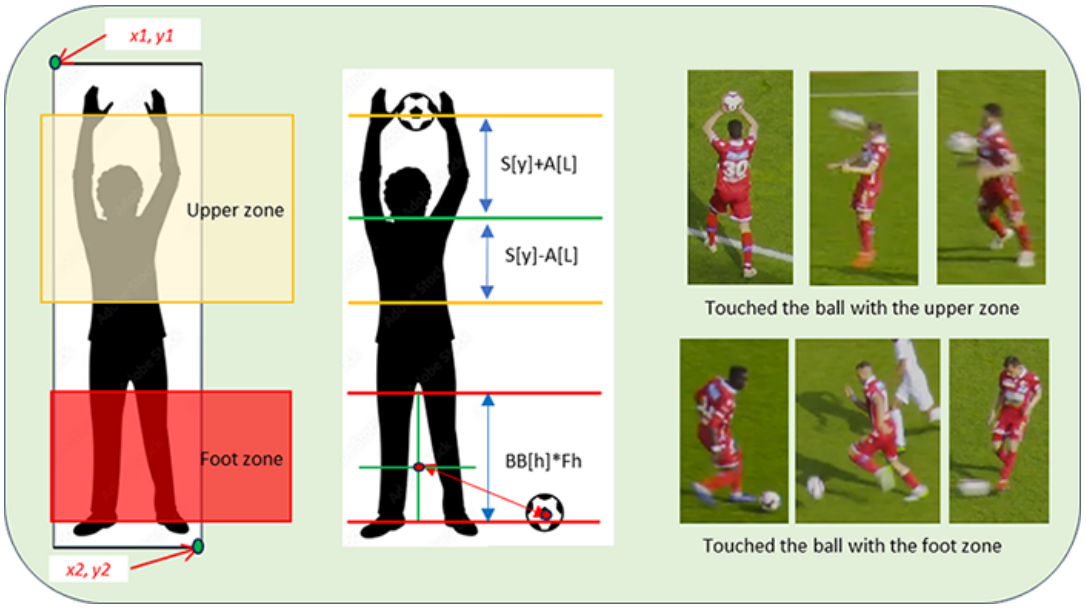

FIGURE 4. Structure of a player's body is divided into upper and foot zone body regions using the anthropometric body region division method.

body zones. The definition of the upper and foot zones is expressed in Equations (6) and (7). Let the bounding box of a detected player be represented as

$$BoundBox(x_1, y_1, x_2, y_2) = players\_detections.xyxy[idx],$$

where (x_1, y_1) and (x_2, y_2) denote the top-left and bottom-right coordinates of the bounding box. The player height in the image is calculated as

$$BB[h] = y_2 - y_1.$$

The approximate shoulder height is given by

$$S[y] = y_1 + (1 - 0.818) \cdot BB[h],$$

where $0.2 \cdot BB[h]$ corresponds to the anthropometric proportion of shoulder size relative to body height. The arm length is estimated as

$$A[L] = 0.4 \cdot BB[h].$$

The upper zone is then defined by

$$S[y] - A[L] \leq y_b \leq S[y] + A[L]. \tag{6}$$

0.818 is the normalized shoulder height ratio with respect to body height average arm length being approximately 40% of body height [30]. The foot zone is defined relative to the bottom of the bounding box as

$$Foot\_zone = y_2 - BB[h] \cdot F_h, \tag{7}$$

where $F_h \in [0.25, 0.40]$ represents the proportional range (25-40%) above the player's foot zone.

3.4.2. *Distance-based initial task.* Accurately calculating the distance between the ball and the player is crucial for identifying which player is in possession. We use the Euclidean distance in Equation (8), which measures the straight-line distance between two points, to compute the distance from the player to the ball. Let B denote the ball position and P the player position.

$$D = \|P - B\|_2. \tag{8}$$

We assessed three distance thresholds (40, 50, and 60 pixels) for attributing ball possession to the nearest player. With the actual number of ball-possession instances being $N = 150$, let TP represent the count of accurate detections and Pred signify all detected instances (true positives and false positives combined). Precision, Recall, and the balanced F1-score are computed as per Equation (9):

$$Precision = \frac{TP}{Pred}, \quad Recall = \frac{TP}{N}, \quad F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \tag{9}$$

TABLE 1. Results across distance thresholds (pixels), $N = 150$

Distance (px)	All detected	Correct	Precision	Recall	F1
40	105	52	0.495	0.347	0.408
50	118	78	0.661	0.520	**0.582**
60	125	56	0.448	0.373	0.407

As shown in Table 1, the threshold of **50 pixels** achieves the highest F1-score (0.582), indicating the best balance between Precision and Recall among the experimented settings. Therefore, we select **50 pixels** as the distance threshold for ball-to-player assignment.

3.4.3. *CNN-based occlusion solution.* Player actions such as kicking, headed passes, and blocking the ball occur within very short timeframes, often just a few frames, making them difficult to detect with the naked eye. The algorithm for determining who has possession of the ball first added the player to the list of close players (Equation (11)) if the distance from the ball to the player's foot or upper body in each frame is less than a predefined distance value (Equation (10)). When only one player is close ($N = 1$), that player is immediately assigned possession (Equation (12)). A situation in which multiple players are near the ball is termed "players occlusion". This concept aligns with Folgado et al. [31] description that "several potential duel areas will arise" when multiple players approach each other closely around the ball. Since the clustering of players ($N > 1$) leads to uncertainty in the distribution of possession, our algorithm employs a machine learning model (Equations (13) and (14)) we developed earlier to determine who has possession of the ball.

Specifically, we developed two classifiers based on the ResNet architecture, tailored to identify possession at different body zones – the upper zone and the foot zone. To train the model, collect several short video clips and images featuring multiple players competing for the ball and manually label them as either "ball-player" (in possession) or "player" (not in possession). Using this dataset, we trained a ResNet-18 classifier on cropped images containing the ball and nearby players. We trained the model to identify ambiguous possession scenarios, such as those near a player's feet, between two players' feet, contested possessions, and when the ball is in the air. Unlike simple distance-based heuristics, these CNN classifiers analyze visual features such as jersey shape, player posture, and the apparent control of the ball. The distance from the B center of the ball to player P_i, $D = 50$ pixels is the threshold distance value:

$$d_i = \|P_i - B\|_2. \tag{10}$$

Close players as

$$\mathcal{N} = \{i \mid d_i \leq D\}. \tag{11}$$

If only one close player

$$|\mathcal{N}| = 1 \Rightarrow p^* = \mathcal{N}[1]. \tag{12}$$

If multiple players are within the threshold ($|\mathcal{N}| > 1$):

- *upper zone*:

$$s_i^{\text{upper}} = f_\theta^{\text{upper}}(P_i, B), \quad p^* = \arg\max_{i \in \mathcal{N}} s_i^{\text{upper}}. \tag{13}$$

- *foot zone*:

$$s_i^{\text{foot}} = f_\theta^{\text{foot}}(P_i, B), \quad p^* = \arg\max_{i \in \mathcal{N}} s_i^{\text{foot}}. \tag{14}$$

To ensure repeatability and transparency, we describe the detailed training settings, hyperparameters, and pseudocode (Algorithm 1) for the CNN models used in this work as follows. We set the random seed for deterministic behavior:

- torch.manual_seed(42),
- np.randomṡeed(42),
- randomṡeed(42).

These are the settings and pseudocode for training our CNN model:

- Architecture: ResNet-18 (pretrained weights: ImageNet).
- Final layer: fully connected with 2 classes (ball_player vs. player).
- Input size: 224×224; normalization with ImageNet mean/std.
- Data augmentation: random rotation ($\pm 5°$).
- Loss: Cross-entropy loss.
- Optimizer: Adam with learning rate = 0.001.
- Hyperparameters:
  - batch size = 16,
  - epochs = 100,
  - weight decay = 0 (default Adam).

Algorithm 1 Training loop for neural network

```
for epoch in 1 ... N_epochs do
    for (images, labels) in train_loader do
        Move (images, labels) to device
        optimizer.zero_grad()
        outputs ← model(images)
        loss ← criterion(outputs, labels)
        loss.backward()
        optimizer.step()
    end for
    validate()
    save_model()
end for
```

The principle of operation of the algorithm for determining the player in possession of the ball is shown in Figure 5. Specifically, the upper-zone CNN model was trained on 750 original images, augmented to 2,250 images, and the foot region model was trained by increasing the original 3,167 images to 9,501 images by rotating them by -5 and $+5$ degrees.

3.5. **Pass detection and statistics generation.** Our proposed method detects a successful team pass if the player currently in possession of the ball and the player who previously had the ball belong to the same team. If the previous player regains possession of the ball, the opposing team intercepts the ball, goes out of bounds, or is passed directly to another team player, it is not counted as a successful pass. A valid pass for team-A is

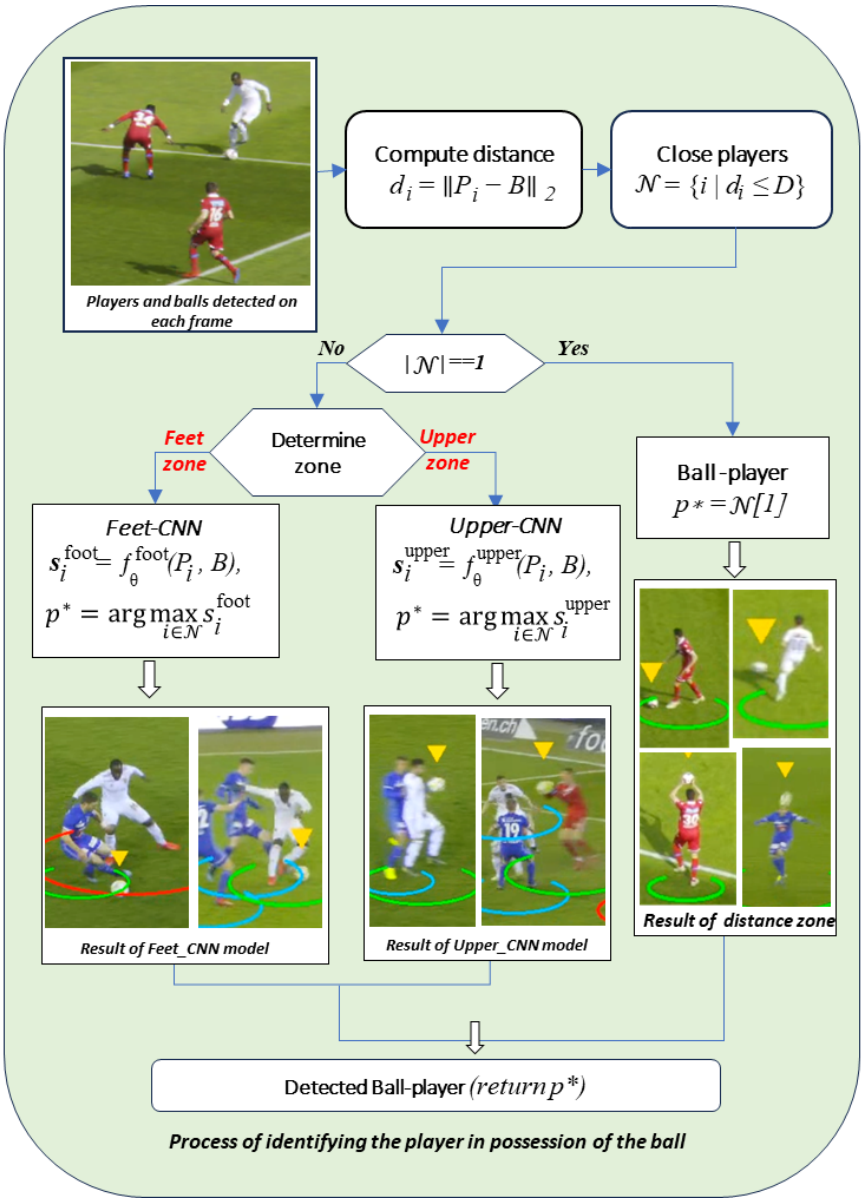

FIGURE 5. Algorithm for determining the player in possession of the ball

explicitly defined as the transfer of ball possession between two players of the same team [12]. The team's successful pass is calculated by Equation (15). Let P_{pre} and P_{curr} denote the player identifiers for the previous and current ball possessors, respectively. T_{pre} and T_{curr} denote the teams to which the previous and current player belong.

$$PassSuccess(t) = \begin{cases} 1, & \text{if } P_{pre} \neq P_{curr} \wedge T_{pre} = T_{curr}, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

After identifying the current ball possessor in a given frame, this player is assigned as the previous possessor (P_{pre}) for the subsequent frame. This step ensures the continuity of ball ownership tracking and provides information about the player and team that previously held possession. Processing each frame sequentially minimizes real-time latency and reduces computational overhead. Upon completing the processing of each frame, we provide live statistics indicating the team currently in possession and the cumulative number of successful passes for both teams. Algorithm 2 outlines the procedure used to compute passing statistics.

Algorithm 2 Generate statistics on successful team passes and ball possession of a team

Input: Soccer video
Output: Count of passes for each team and show current ball possession status.
Set $passCounts = (\text{“}Team - A\text{”} : 0, \text{“}Team - B\text{”} : 0)$
Set $D = Distance\ value$
Set $T_{pre} = \text{None}$, $T_{curr} = \text{None}$
Set $P_{pre} = \text{None}$
Set $B_{team} = \text{None}$
for each frame in video **do**
    $P_{curr} = \text{FunctionBallPlayer}(P[], B, P_{pre}, D)$
    if $P_{pre} \neq \text{None}$ and $P_{curr} \neq \text{None}$ **then**
        if $T_{curr} == T_{pre}$ **then**
            if $P_{pre} \neq P_{curr}$ **then**
                $passCounts[T_{curr}] += 1$
            end if
        else
            $B_{team} = T_{curr}$
        end if
    else
        $B_{team} = \text{None}$
    end if
    $P_{pre} = P_{curr}$
end for

4. **Experiment and Results.** In this section, we present the results of the proposed system and its evaluation across diverse soccer video scenarios to assess the accuracy of its detection, tracking, and event recognition components, as well as its runtime performance. We describe the datasets and test matches used, the baseline (ground-truth) data for evaluation, the applied metrics, and the experimental results. We compiled various broadcast soccer video clips from publicly available sources for evaluation purposes. We experimented with our method on a range of diverse 10 short video clips, each 1.5-8 minutes long, under different conditions: single-camera matches from the SoccerNet dataset [38] (Video1-Video4), dual-camera footage from the 2018 FIFA World Cup [39, 40] (Video5 and Video6), a drone-recorded match [41] (Video7), a match with visually similar jerseys [42] (Video8), a match played under snowy weather conditions [43] (Video9), and an additional video from [44] (Video10). Importantly, none of these clips were used to train the detection or possession CNN models, ensuring fair and unbiased evaluation. We used videos excluded from the test dataset to train our ResNet-based CNN classifiers for ball possession (upper zone and foot zone).

4.1. **Team assigner.** Our unsupervised team assignment method performed strongly in classifying players into their respective teams across the test dataset. The process achieved high accuracy even in challenging scenarios where teams wore visually similar jerseys (e.g., one team in light blue and the other in blue). Misclassifications were avoided entirely due to our goalkeeper-specific annotation strategy, which assigns goalkeepers to their respective teams independently of the clustering process. The classification results, visualized in our figures, validate the accuracy and robustness of our approach across all test videos. We present the experimental results of the UMAP+KMeans and UMAP+KMeans Caching

TABLE 2. Classification results of our UMAP+KMeans and UMAP+ KMeans Caching methods

Videos	**Ground truth**	**UMAP+ KMeans Correct**	**UMAP+ KMeans Accuracy**	**UMAP+KMeans Caching Correct**	**UMAP+KMeans Caching Accuracy**
Video1	16166	15630	96.68%	15599	96.49%
Video2	12476	12181	97.64%	12145	97.35%
Video3	14990	14775	96.93%	14720	98.20%
Video4	16031	15760	95.75%	15716	98.04%
Video7	10528	9956	94.57%	9890	93.94%
Video8	8660	8065	93.13%	8125	93.82%
Video9	772	626	81.09%	620	80.31%
Average	–	–	**93.68%**	–	**94.02%**

TABLE 3. FPS of UMAP+KMeans vs UMAP+KMeans Caching

Videos	**UMAP+KMeans**	**UMAP+KMeans Caching (GPU cuML)**
Video1	7.35 FPS	19.78 FPS
Video2	8.68 FPS	17.89 FPS
Video3	8.51 FPS	16.79 FPS
Video4	8.26 FPS	18.89 FPS

methods on 7 videos with different colors of athletes' jerseys and different environments in Table 2.

Since our system is designed for low-latency real-time operation, evaluating the time performance of our proposed team classification methods is crucial. To this end, we measured the number of frames processed per second (FPS) for each method, as reported in Table 3. The results of both methods were compared using the Benefit-Time Efficiency Ratio (16) method. Specifically, this ratio considers the higher classification accuracy (Accuracy) in conjunction with the system's frame processing rate (FPS), allowing for a balanced assessment of both computational performance and classification effectiveness.

Benefit-Time Efficiency Ratio:

$$\mathit{Efficiency} = \frac{\mathit{Accuracy}}{\mathit{Time}}. \tag{16}$$

The efficiency ratio of the UMAP+KMeans method is

$$\mathit{Efficiency}_{\mathit{UMAP+KMeans}} = \frac{0.9764}{0.110} = 8.87. \tag{17}$$

The efficiency ratio of the UMAP+KMeans Caching method is

$$\mathit{Efficiency}_{\mathit{UMAP+KMeans\ Caching}} = \frac{0.9820}{0.054} = 18.18. \tag{18}$$

The efficiency-time ratio of the UMAP+KMeans (17) method and the UMAP+ KMeans Caching (18) method is calculated. It can be seen that the average accuracy, processing speed, and efficiency of the UMAP+KMeans Caching method (18.33 FPS) are superior to those of the UMAP+KMeans method (8.2 FPS). The processing speed of the UMAP+KMeans Caching method is clearly superior to that of the 18.33 FPS (0.054 seconds per frame), which is about 2.2 times faster. Given the negligible trade-off in accuracy and the substantial gain in processing efficiency, we selected the UMAP+KMeans Caching method for real-time soccer match analysis, where minimizing latency is critical.

4.2. **Detecting ball-player interactions.** To validate our method, we constructed a ground truth dataset that includes annotations of players in ball possession. Unlike prior studies, which often focus solely on foot interactions, our approach accounts for more subtle forms of ball control such as headers, throw-ins, and chest traps. As such, we designed experiments to measure the accuracy of our method in detecting various types of ball-player interactions. As such, we designed experiments to measure the performance of our method in detecting ball-player interactions. Two evaluation pipelines were used: one that relies purely on geometric heuristics (without machine learning), and the other that incorporates CNN-based classification models. To provide a more comprehensive evaluation, we adopted standard classification metrics, including Precision, Recall, F1-score, and Overall Accuracy. A true positive (TP) was recorded when the system correctly identified a player's foot making contact with the ball. A false positive (FP) was logged when the system incorrectly identified ball contact by the player's foot when no such interaction occurred. A false negative (FN) represented missed detections where a ball-player interaction was present in the ground truth but not detected by the system. Table 4 summarizes the results for seven evaluation videos. On average, our method achieved a Precision of 94.82%, a Recall of 88.45%, an F1-score of 91.50%, and an average of 84.36% using the ground truth as a reference, demonstrating that the CNN-based approach substantially improved the reliability of ball-player interaction detection compared to the heuristic-only ground truth.

TABLE 4. Results of the CNN model during the standard passes. Precision, Recall, F1, and Accuracy are in %.

Videos	**GT**	**TP**	**FP**	**FN**	**Ball_player**			**Player**			**Acc.**
					Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	
Video1	52	46	3	6	93.88	88.46	91.09	93.88	88.46	91.09	83.64
Video2	78	68	4	10	94.44	87.18	90.67	94.44	87.18	90.67	82.93
Video3	125	105	5	20	95.45	84.00	89.36	95.45	84.00	89.36	80.77
Video4	95	84	4	11	95.45	88.42	91.80	95.45	88.42	91.80	84.85
Video5	92	82	4	10	95.35	89.13	92.13	95.35	89.13	92.13	85.42
Video6	119	105	5	14	95.45	88.24	91.70	95.45	88.24	91.70	84.68
Video10	16	15	1	1	93.75	93.75	93.75	93.75	93.75	93.75	88.24
Average	–	–	–	–	**94.82**	**88.45**	**91.50**	**94.82**	**88.45**	**91.50**	**84.36**

A key advantage of our method over existing approaches is its ability to detect a broader range of ball-player interactions beyond traditional foot passes. Specifically, our system successfully identifies complex soccer events such as headed passes, throw-ins, and chest control, in addition to standard foot passes. This comprehensive detection capability lets us capture a more complete picture of in-game actions and player involvement. Table 5 presents the number of such events included in our experimental dataset and the corresponding detection results produced by our method.

The **Weighted Average Precision** is calculated using the following equation:

$$\textit{Weighted Average Precision} = \frac{\sum_{i=1}^{N} P_i \times w_i}{\sum_{i=1}^{N} w_i} \tag{19}$$

Our method demonstrates higher accuracy than previous approaches by detecting headed passes, throw-ins, and chest control, which were not accounted for in earlier research. These types of actions represent approximately 8-11% of all passes in matches involving highly skilled players.

TABLE 5. Accuracy of all events

Events	Ground truth	Correct	Accuracy correct (P_i)	Percentage for all events (w_i)
Headed passes	43	37	86.05%	0.0665
Throws	16	14	87.50%	0.0247
Chest control	11	9	81.82%	0.0170
Foot passes	577	505	84.36%	0.8918

To fairly evaluate overall performance, we computed the average detection accuracy using the Weighted Average Accuracy method (19), which considers the relative frequency of each type of pass in the dataset, to calculate the accuracy of our method, which is 84.51%.

4.3. **Comparison to other methods.** Our method achieves a performance of 84.51%, representing an improvement of 3.01% compared to the proposed method in [13], a 2.41% improvement over [12], and a 10.81% improvement over [14]. The average of the throw-ins, shots, penalties, and goals detected by the [35] is 4.93% lower than the average accuracy of our method. Moreover, our approach assigns all players to teams, enabling more comprehensive analysis while maintaining GPU runtimes closer to real-time performance. These gains highlight the effectiveness of our approach in capturing both conventional and nuanced passing actions. Additionally, Table 6 presents a comparative summary of the pass detection accuracy and per-frame GPU processing time across different methods.

TABLE 6. Comparison of the accuracy of the calculation of pass detection and processing time by different methods

Methods	Detected events	Accuracy	Processing time
[13]	Standard pass	81.5%	0.05 second
[12]	Standard pass	82.1%	21.8 second
[14]	Standard pass	73.7%	6.86 second
[35]	Average of throw-ins, penalties, shots, and goal	79.58%	unspecified
Our method	Standard pass, head pass, throw-ins, and chest control	84.51%	0.054 second

Although the GPU processing time of our method is comparable to that of [13], it offers high accuracy in pass detection. It uniquely detects previously undetected events, such as headers and shots. Additionally, while [13] only identifies the team of the player in possession, our method assigns all players to their respective teams, enabling more detailed team-based analysis. Our training and experimentation were conducted on an NVIDIA GeForce RTX 4070 GPU with 12 GB of memory, CUDA 12.6, and cuDNN 9. The experiments were conducted on a system equipped with an Intel i5 13th-generation 2.5 GHz processor, 64 GB of RAM, and Ubuntu 20.04, utilizing Python 3.10.

5. Discussion.

5.1. **Error analysis.** We performed an error analysis to identify the primary sources of failure. One source of failure is the occlusion of players. In this case, the ball is partially obscured, causing the system to determine the ball owner incorrectly. A second failure happens while the ball is airborne. Due to perspective overlap, the ball in the air visually appears to match the player behind it, which may lead the system to attribute possession

TABLE 7. Summary of detection errors across all evaluation videos

Error type	Count	Proportion (%)
Total errors (FP + FN)	98	100.0
Occlusion (player congestion)	42	42.9
Airborne ball / Visual overlap	56	57.1
False Positives (FP)	26	26.5
False Negatives (FN)	72	73.5

to that player mistakenly. We performed a thorough error analysis, categorizing all incorrect detections into two primary types, to gain a clearer insight into the limitations of our approach. Table 7 provides a summary of the overall error distribution in the evaluation videos. Of the 98 errors, 26 were false positives, incorrectly assigning ball possession to a player who did not possess it, while 72 were false negatives, in which the system overlooked actual possession events.

These error cases demonstrate that relying solely on the distance between the ball and the player, along with a CNN classifier, is not a comprehensive solution. However, it does result in a slight loss of real-time performance. Our comprehensive error analysis reveals specific scenarios where the current approach faces limitations:
Occlusion-related failures (42.9% of errors):

- Dense player clustering during corner kicks and free kicks
- Ball temporarily hidden behind player bodies during dribbling
- Multiple players competing for aerial balls

Airborne ball challenges (57.1% of errors):

- Perspective alignment causing false possession attribution
- Fast ball movement creating motion blur in detection
- Headers occurring at frame boundaries causing temporal misalignment

Visual examples of these failure modes are illustrated in Figure 6, demonstrating both the challenging nature of these scenarios and the system's general robustness in standard gameplay situations.

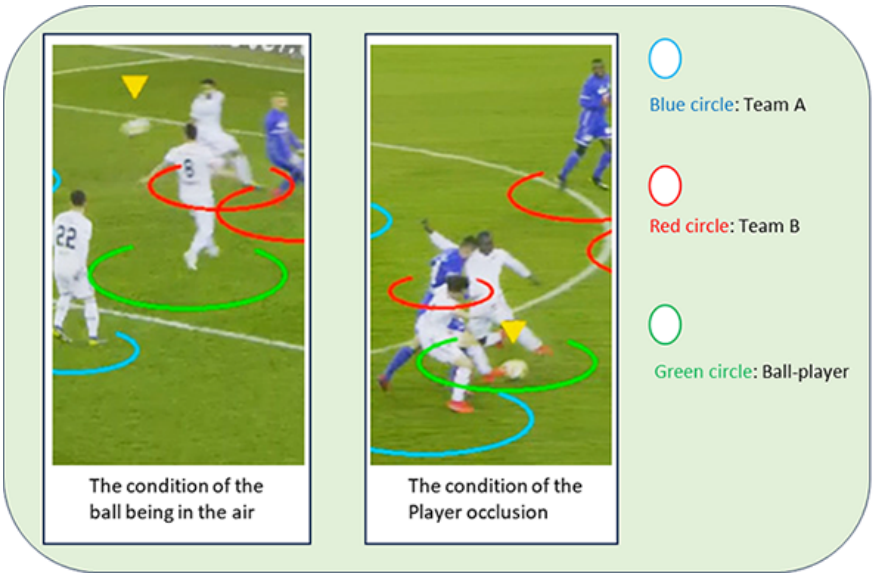

FIGURE 6. (color online) Cases of incorrect detection of possession of the ball

5.2. **Possible improvements.** Future work should not only improve the accuracy of detection, but also take account of real-time constraints. LSTM and 3D convolutional architectures offer temporal modeling capabilities, but their computational cost may impede their use in live match situations. Another promising approach is to apply keypoints on the pitch field to adjusting the spatial coordinates of the player and the ball. Camera geometry and player/ball positions can be determined in normal field coordinates by defining features like corners, sidelines, and goal areas. This will reduce mistakes caused by ball obstructions and aerial balls, as the distance between the player and the ball in field space can be measured more precisely than with raw image coordinates.

6. **Conclusion.** In this study, we proposed an efficient method that combines computer vision and machine learning techniques to recognize soccer events and automatically identify a team's successful pass statistics from soccer videos. Our approach can accurately identify players possessing the ball even during occlusion by segmenting the player's body into upper and foot zones and leveraging specialized CNN models in crowded situations. For team classification, we utilize SigLIP embedding, UMAP reduction, and KMeans-Caching, enabling us to classify teams in the absence of annotated data. The pass detection performance increased to 84.51% and the processing time increased to 0.054 seconds, confirming that it has higher accuracy and near-real-time processing than other methods. Unlike prior work, our system detects a wide range of passes, including headed passes, throw-ins, chest control, and foot passes. Our future work is to address the highlighted limitations in error analysis and possible improvements without compromising real-time performance.

REFERENCES

[1] M. Manafifard, H. Ebadi and H. A. Moghaddam, A survey on player tracking in soccer videos, *Computer Vision and Image Understanding*, vol.159, pp.19-46, 2017.

[2] Jr. R. Pielke, *How Ball Possession is Measured in Football*, 2012, http://leastthing.blogspot.com/2012/02/how-ball-possession-is-measured-in.html, Accessed in April, 2025.

[3] H. Glasser, *The Problem with Possession, The Inside Story of Soccer's Most Controversial Stat*, 2014, https://slate.com/culture/2014/06/soccer-possession-the-inside-story-of-the-games-most-controversial-stat.html, Accessed in April, 2025.

[4] J. Yu, A. Lei and Y. Hu, Soccer video event detection based on deep learning, *MultiMedia Modeling, MMM 2019*, pp.377-389, 2019.

[5] M. Koshkina, H. Pidaparthy and J. H. Elder, Contrastive learning for sports video: Unsupervised player classification, *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2021.

[6] K. Lu, J. Chen, J. J. Little and H. He, Lightweight convolutional neural networks for player detection and classification, *Computer Vision and Image Understanding*, vol.172, pp.77-87, 2018.

[7] C. A. Gadde and C. V. Jawahar, Transductive weakly-supervised player detection using soccer broadcast videos, *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2021.

[8] M. Istasse, J. Moreau and C. De Vleeschouwer, Associative embedding for team discrimination, *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[9] O. A. N. Rongved and M. Stige, Automated event detection and classification in soccer: The potential of using multiple modalities, *Machine Learning and Knowledge Extraction MDPI*, pp.1030-1054, 2021.

[10] A. Karimi, R. Toosi and M. A. Akhaee, Soccer event detection using deep learning, *arXiv Preprint*, arXiv: 2102.04331v1, 2021.

[11] S. Bose, S. Sarkar and A. Chakrabarti, SoccerKDNet: A knowledge distillation framework for action recognition in soccer videos, *arXiv Preprint*, arXiv: 2307.07768v1, 2023.

[12] S. Sarkar, A. Chakrabarti and D. P. Mukherjee, Estimation of ball possession statistics in soccer video, *ICVGIP 2018*, Hyderabad, India, 2018.

[13] S. Sarkar, D. P. Mukherjee and A. Chakrabarti, Watch and act: Dual interacting agents for automatic generation of possession statistics in soccer, *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022.
[14] S. Sarkar, D. P. Mukherjee and A. Chakrabarti, Generation of ball possession statistics in soccer using minimum-cost flow network, *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.
[15] D. Link and M. Hoernig, Individual ball possession in soccer, *PloS One*, 2017.
[16] A. M. Lopes and J. A. T. Machado, Uniform manifold approximation and projection analysis of soccer players, *Entropy*, 2021.
[17] J. Huang, N. Li, T. Zhang, G. Li, T. Huang and W. Gao, SAP: Self-adaptive proposal model for temporal action detection based on reinforcement learning, *The 32nd AAAI Conference on Artificial Intelligence*, 2018.
[18] L. McInnes, J. Healy and J. Melville, UMAP: Uniform manifold approximation and projection for dimension reduction, *arXiv Preprint*, arXiv: 1802.03426, 2018.
[19] I. T. Jolliffe and J. Cadima, Principal component analysis: A review and recent developments, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, DOI: 10.1098/rsta.2015.0202, 2016.
[20] Y. Zhang, P. Sun and Y. Jiang, ByteTrack: Multi-object tracking by associating every detection box, *Computer Vision – ECCV 2022: The 17th European Conference*, pp.23-37, 2022.
[21] X. Zhai, B. Mustafa, A. Kolesnikov and L. Beyer, Sigmoid loss for language image pre-training, *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
[22] D. Sorano, F. Carrara, P. Cintia, F. Falchi and L. Pappalardo, Automatic pass annotation from soccer video streams based on object detection and LSTM, *arXiv Preprint*, arXiv: 2007.06475v1, 2020.
[23] Z. Ivankovic, M. Rackovic and M. Ivkovic, Automatic player position detection in basketball games, *Multimedia Tools and Applications*, vol.72, pp.2741-2767, 2014.
[24] I. Uchida, H. Shishido and Y. Kameda, Automated offside detection by spatio-temporal analysis of football videos, *MMSports 21, Analyses in Team Sports*, Virtual Event, China, 2021.
[25] A. Cioppa, A. Deliege and S. Giancola, A context-aware loss function for action spotting in soccer videos, *Open Access Version, Provided by the Computer Vision Foundation (CVPR)*, 2020.
[26] S. A. Hicks, V. Thambawita, H. K. Stensland, E. Zouganeli, D. Johansen, M. A. Riegler and P. Halvorsen, Real-time detection of events in soccer videos using 3D convolutional neural networks, *Proc. of the SMEEE International Symposium on Multimedia (ISM)*, pp.135-144, 2020.
[27] Rongved, O. Hicks, S. Thambawita, V. Stensland, H. Zouganeli, E. Johansen, D. Midoglu, C. Riegler and M. Halvorsen, Using 3D convolutional neural networks for real-time detection of soccer events, *IEEE J. Sel. Top. Signal Process*, vol.15, pp.161-187, 2021.
[28] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, You Only Look Once: Unified, real-time object detection, *IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp.779-788, 2016.
[29] B. Namjildorj, S. Karungaru, K. Terada and A. Ayush, Detection of players on a soccer team using unsupervised methods, *The 3rd International Conference on Electrical Facilities and Information Technologies*, Ulaanbaatar, Mongolia, pp.141-144, 2024.
[30] C. C. Gordon et al., Anthropometric survey of U.S. personnel, *Summary Statistics, Interim Report*, 1989.
[31] H. Folgado, K. Lemmink, W. Frencken and J. Sampaio, Length, width, and centroid distance as measures of teams' tactical performance in youth football, *European College of Sport Science*, DOI: 10.1080/17461391.2012.730060, 2012.
[32] Y. Hu and D. Liu, Design of sports action recognition and evaluation based on improved DTW algorithm, *International Journal of Innovative Computing, Information and Control*, vol.21, no.1, pp.37-52, DOI: 10.24507/ijicic.21.01.37, 2025.
[33] Z. Fan and K. Sun, Kinect based recognition and detection of fitness Qigong movements, *International Journal of Innovative Computing, Information and Control*, vol.20, no.6, pp.1837-1850, DOI: 10.24507/ijicic.20.06.1837, 2024.
[34] Y. Liu, K. Matsui, Y. Kageyama, H. Shirai and C. Ishizawa, A CNN-based method for human action analysis using nighttime infrared images, *International Journal of Innovative Computing, Information and Control*, vol.19, no.6, pp.1861-1875, DOI: 10.24507/ijicic.19.06.1861, 2023.

[35] A. Xarles, S. Escalera, T. B. Moeslund and A. Clapes, ASTRA: An Action Spotting TRAnsformer for soccer videos, *Proc. of the 6th International Workshop on Multimedia Content Analysis in Sports (MMSports'23)*, pp.93-102, DOI: 10.1145/3606038.3616153, 2023.
[36] A. Xarles, S. Escalera, T. B. Moeslund and A. Clapes, T-DEED: Temporal-Discriminability Enhancer Encoder-Decoder for precise event spotting in sports videos, *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp.3350-3359, DOI: 10.1109/CVPRW63382.2024.00345, 2024.
[37] J. Deniz, M. Liashuh, J. Rabarisoa, A. Orcesi and R. Herault, COMEDIAN: Self-supervised learning and knowledge distillation for action spotting using transformers, *Proc. of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACV Workshops)*, 2024.
[38] Video1-4, https://www.soccer-net.org/tasks/camera-calibration#h.p7raxm186dqc, 2021.
[39] Video5, *2018 FIFA World Cup: France v Argentina*, https://www.youtube.com/watch?v=Cbij3MKhdOY, Accessed on September 22, 2022.
[40] Video6, *2018 FIFA World Cup: Portugal v Spain*, https://www.youtube.com/watch?v=OFbyNU6UQQs&=1127s, Accessed on August 25, 2022.
[41] Video7, *2018 FIFA World Cup: France v Argentina*, https://www.youtube.com/watch?v=Cbij3MKhdOY&t=2s, Accessed on September 22, 2022.
[42] Video8, *Matt Ervin (MIDrone)*, https://www.youtube.com/watch?v=1JPchA7LZ5U, Accessed on October 12, 2016.
[43] Video9, *Premier League Football in the SNOW*, https://www.youtube.com/watch?v=V6H9U_EiMPI, Accessed on January 10, 2024.
[44] Video10, *Premier League | Flashback – Man City v Man United*, https://www.youtube.com/watch?v=l20F3_Bcwuc, Accessed on November 11, 2018.
[45] Roboflow Open Data, https://universe.roboflow.com/roboflow-jvuqo/football-players-detection-3zvbc, Accessed on January 10, 2023.
[46] Roboflow Open Data, https://universe.roboflow.com/label-gdoal/soccer-oilql/dataset/3, Accessed on February 28, 2025.

Author Biography

Namjildorj Batbaatar received a master's degree in Computer Science from the Mongolian University of Science and Technology, Mongolia, in 2008. He has been working as a teacher in the Department of Information Technology at Mongolian University of Science and Technology since 2008. He is currently pursuing a Ph.D. degree in Information Science and Intelligent Systems at Tokushima University, Japan. His research interests include image processing, real-time image/video processing, object and event detection, and computer vision.

Stephen Karungaru graduated from the Department of Electronics, Information and Communication Technology at MOI University, Kenya, in 1993. He received a Ph.D. degree in Information System Design from the Department of Information Science and Intelligent Systems, Tokushima University, Japan, in 2004. He is currently an associate professor at the same university. His research interests are in pattern recognition for artificial intelligence. He has published 2 book chapters and numerous journal papers.

Kenji Terada received a Ph.D. degree from Keio University, Japan, in 1995. In 2009, he became a professor in the Department of Information Science and Intelligent Systems, Tokushima University, Japan. His research interests are in computer vision and image processing. He is a member of the IEEE, IEEJ, and IEICE.

Munkhbat Gantumur received a master's degree in Information Technology from the Mongolian University of Science and Technology, Mongolia, in 2015. He is studying for a doctoral course at the Department of Information Science and Intelligent Systems, Tokushima University, Japan. His research interests include image processing, object and event detection, and computer vision.

Altangerel Ayush received a doctorate in Computer Science from the Mongolian University of Science and Technology, Mongolia, in 2011. In 2014, he became a professor in the Department of Information Technology, School of Information and Communication Technology, Mongolian University of Science and Technology, Mongolia. His research interests are in neural networks, machine learning and image processing.