Multi-Modal Hit Detection and Positional Analysis in Padel Competitions

Robbe Decorte, Martin Paré, Jelle Vanhaeverbeke, Joachim Taelman,
Maarten Slembrouck, Steven Verstockt
Ghent University - imec, IDLab
{firstname}.{lastname}@ugent.be

Abstract

Padel is a rapidly growing racquet sport and has gained popularity globally due to its accessibility and exciting gameplay dynamics. Effective coordination between teammates hinges on maintaining an appropriate distance, allowing for seamless transitions between offensive and defensive maneuvers. A balanced inter-player distance and distance to the net not only facilitates efficient communication but also enhances the team's ability to exploit openings in the opponent's defense while minimizing vulnerabilities. We introduce a new open dataset of padel rallies with annotations for hits and player-ball interactions, a predictive model for detecting hits based on audio signals, a re-identification algorithm for pose tracking, and a framework for calculating inter-player and player-net distances during rallies. Our predictive model achieves an average F1-score of 92% for hit detection, demonstrating robust performance across different match conditions. Furthermore, we develop a system for accurately assigning hits to individual players, achieving an overall accuracy of 83.70% for player-specific assignment and 86.83% for team-based assignment.

1. Introduction

Padel, a fast-growing racquet sport, has gained remarkable popularity worldwide due to its accessibility, dynamic gameplay, and social appeal. Combining elements of tennis and squash, padel is played on an enclosed court with glass walls and a perforated surface, allowing for exciting rallies and strategic shot-making. The sport's surge in popularity has sparked interest among researchers and sports scientists, leading to a growing body of literature focused on understanding player dynamics, performance metrics, and tactical strategies in padel.

A lot of insights can be obtained by measuring the inter player distances between teams (in regards to defensive field coverage) as well as the distance to the net. For example, when defending players push their rivals to the back of the court, they tend to get closer to the net in order to gain a better attacking position and control over the court. But how close should they get? The editors of padeltrainer.com [4] suggest around 4 meters. The actual optimal value of distance to the net and between players is not further investigated in this work, but we present a way to objectively measure these distances at important timestamps in the rally (when the ball is hit). These insights could help the coaching staff with more in depth point-by-point analysis such as traveled distance, player movement patterns and field coverage (on top of the previously mentioned distance metrics).

This research paper aims to contribute to the field of sports analytics by exploring player tracking and measuring distances between players in padel. The core contributions of this publication are as follows: a new open dataset of padel rallies with annotations for hits (full) and which player hits the ball (partial), a predictive model used to detect hits based on the audio signal of a rally, a re-identification algorithm based on the pose tracks appear and disappear locations and finally, a framework based on all previous to calculate inter-player and player-net distance at relevant timestamps in the rally.

The remainder of this paper is organized as: Section 2 introduces relevant literature to this work. In Section 3 introduces a new padel-specific, public and annotated dataset. Our method regarding ball hit detection and player related metrics are described in Section 4 and Section 5. In Section 6, the results and analysis of the implemented techniques are presented. And finally, the core contributions and general findings regarding the core contributions are summarized in Section 7.

2. Related work

2.1. Position tracking of the ball and players

Analysis of player movements that includes the trajectory data of the ball have become more prevalent to enhance sports analytics systems [3, 12]. With the advent of machine learning, particularly deep learning, there has been a paradigm shift towards convolutional-driven approaches for player/ball detection in sports. Researchers have adapted

convolutional neural network (CNN) architectures such as Faster R-CNN [21] and YOLO [20] for real-time detection in various sports scenarios. Additionally, TrackNet [8], a specialized neural network architecture designed for object tracking, has gained attention in sports-related applications. TrackNet leverages temporal information to track objects across consecutive video frames, offering improved robustness and accuracy in ball tracking tasks. In tennis, Rocha et al. [22] introduced a TrackNet-based approach for real-time ball tracking, demonstrating improved performance in challenging scenarios such as occlusions and rapid motion compared to other state-of-the-art methods.

The counterpart of the ball position required for game analysis is the tracking of the players' movements on the court. Compared to plain tracking, Human pose estimation (HPE) offers several advantages due to its ability to provide more detailed insights into athletes' movements and biomechanics. With access to the player's wrist and ankle positions we are able to correct some false detections of our system (see Section 5). The applicability of HPE models in sports related contexts has already been shown by several studies [9, 28], and specifically in padel by Javadiha et al. [11]. In addition, large-scale datasets designed to train a pose estimator that adequately captures the challenging and dynamic nature of sports movements, have started to emerge [10].

2.2. Sound Event Detection in Sports

Sound classification and sound event detection have emerged as essential components in sports analytics, enabling comprehensive analysis and understanding of athletic performances. Sound event detection (SED) in sports involves the automatic detection and localization of specific sound events within sports audio recordings, facilitating tasks such as action recognition and play-by-play analysis. Since it predicts the onset and offset time as well as the category of the sound, it can be seen as an extension of sound classification. In many cases, sound event detection tasks require more fine-grained temporal analysis compared to sound classification, as the goal is to precisely locate and identify individual sound events within a longer audio recording [19].

Traditional approaches to SED often rely on handcrafted feature extraction followed by pattern recognition techniques such as Mel-Frequency Cepstral Coefficients (MFCC) or the spectral envelope. For example, Heittola et al. [7] proposed a method based on a Hidden Markov Model (HMMs) classifier that uses context-dependant representations from MFCC. Part of their dataset originates from basketball games but showed poor robustness due to noise interference from the polyphonic nature of these environments.

In recent years, data-driven approaches leveraging deep learning architectures have gained prominence in SED. CNNs have been adapted for SED tasks by treating audio spectrograms as images and applying 2D convolutions to capture temporal and frequency features. For instance, Baughman et al. [2] employed CNNs for detecting important events in tennis matches, including hits, announcers, and crowd reactions, achieving robust performance across different match conditions. Furthermore, attention mechanisms and recurrent neural networks (RNNs) have been integrated into deep learning models for SED to enhance the model's ability to capture temporal dependencies and focus on relevant audio segments [6, 15, 26]. This means that consecutive outputs are no longer independent of eachother, which is the case for repeated windowed classification. For example, Cakir et al. [29] have shown how the capabilities of a CNN to learn local translation invariant filters can be combined with an RNN's capability to model short and long term temporal dependencies in a so called Convolutional Recurrent Neural Network (CRNN) classifier for polyphonic SED tasks. The main limitation of such network is the dependency on a large amount of annotated data. As with other domains where CNNs are used, they suggest to utilize transfer learning to overcome the limitation imposed by small datasets by fine-tuning the final layers of a pretrained model on a smaller dataset [27].

3. Dataset

Thanks to the cooperation of Play Sports, we are able to publicly release 5 hours and 28 minutes of padel matches. The dataset consists of the video summaries of the highlights of padel matches as well as two full length matches which are all recorded from a fixed camera view. For the highlight video's, we have annotated specific start and end frames of each rally in the dataset to avoid analyzing replays, rallies on switching camera views and moving cameras. Although these non-fixed camera views are more appealing to the broadcast audience, they are much harder to analyse as they usually zoom in on the action and therefore fail to frame all 4 players in the video at any given time. In order to determine the player positions, we need to observe all players on the field. In total, the video summaries contain 99 rallies from 11 tournaments, which are further explored in the design and performance analysis of the experiments of this paper. For each rally it includes annotations of the hit windows as well as, for a smaller subsection (319/2377), which player hit the ball. The usage of these annotations or validation purposes is further discussed in Sections 4 and 6.

The dataset to reproduce the experiments and the unprocessed commentary matches are made publicly available at https://cloud.ilabt.imec.be/index.php/s/TFimLDWno6W9ED3.

4. Ball hit detection

Detecting and tracking the ball at relative low framerate (25 fps) is a hard problem, even for humans and since the ball does not remain inside the field of view of the camera at all times and might be occluded by the players' bodies. Hence, false positive ball positions are to be expected. Therefore, we propose an alternative strategy by detecting ball hits using audio instead of video and perform ball detection only on video frames close to the time intervals that the ball was hit.

4.1. Audio analysis

Whenever a player hits the ball, it generally produces a distinct sound that is captured by the microphone of the broadcasting setup. Analysing the audio signal of the video allows the system to find the exact moments that hits occur during the rally and propagate those timing windows to the video pipeline for further analysis. Figure 1 (a) shows the audio signal of such rally. Most peaks in this spectrogram correspond to ball hits. Although most hits cause a similar sound, this is not always the case. Slice movements or when the ball is hit in the same direction it was traveling towards (e.g., hitting the ball after it rebounds from the back wall) may cause a different sound profile compared to a smash. We must also consider the interference caused by shouting as well as when the racket hits the ground or a player who runs into the plexiglass. The influence of spectators as they are expected to be quiet during the rally to not disturb the players is limited, but it is not uncommon for loud gasps/encouragements during exciting plays.

4.2. Sound classification

As preparation of the predictive model that indicates where in an audio file hits occur, we have annotated the onset and offset times of the hits for each rally in the dataset. All labels were created using an audio specific labeling interface through Label Studio [23]. In total, 2377 hits were annotated with the labeling process taking 9 hours and 53 minutes. An overview of the distribution between tournaments, rallies and hits, is shown in Table 1. The annotations will be publicly available alongside the dataset.

As discussed in section 4.1, ball hits are not the only events that can be observed in the audio track. To achieve player-level statistics, we must only extract the hit timestamps. Other sound origins are currently irrelevant, reducing this to a binary problem.

4.2.1 Deep learning based hit detection

The base architecture of the hit detection model has been adapted from SED-net [1]. Considering the scale and difficulty of the problem/dataset it was originally created for, we have modified its structure such that it better fits to the scope of this work and dataset size. Compared to the original architecture we reduced the depth of the CNN layers from 128 to 64, reduced the number of units to 16 in the second GRU layer, kept only the first and last time distributed dense layer and replaced the loss function with binary focal cross-entropy [14]. These changes have reduced the number of trainable parameters from 362,829 to 109,159.

Table 2 shows the adapted SED-net model dimensions. The network input features are 40 log-Mel energy bins in the range 0-42 kHz calculated with 50% overlap and an FFT window length of 2,048, split into sequences of 256 frames. The proposed network is trained using the Adam optimizer [13] with a learning rate of 0.001 and a batch size of 128 sequences. It should be noted that infrasound and ultrasound frequencies are unavailable as the raw footage of the provided videos is unavailable and those ranges were largely discarded by the compression algorithm of the video container.

Tournament	Rallies	Hits	Setting
20230903_FINLAND	10	320	Indoor
20230528_VIGO	17	315	Outdoor
20230927_MADRID	11	296	Indoor
20230604_BRUSSEL	12	250	Outdoor
20231008_DUITSLAND	7	227	Indoor
20230806_MALAGA	10	201	Indoor
20231015_AMSTERDAM	9	190	Indoor
20231102_MENORCA	6	187	Indoor
20231112_MALMO	7	173	Indoor
20230702_VALLADOLID	6	114	Outdoor
20230709_VALENCIA	4	104	Indoor
Totals: 11	99	2377	

Table 1. Distribution of rallies, hits and environment for each tournament's highlights.

Layer type	Hyperparameters	Activation
conv2D	$1 \rightarrow 64$, (3x3), (1x1)	ReLU
maxpool	(1x5)	-
conv2D	$64 \rightarrow 64$, (3x3), (1x1)	ReLU
maxpool	(1x2)	-
conv2D	$64 \rightarrow 64$, (3x3), (1x1)	ReLU
maxpool	(1x2)	-
reshape	(12x64) $\rightarrow$ (256x3)	-
bidirectional GRU	(256x3) $\rightarrow$ (256x32)	tanh
bidirectional GRU	(256x32) $\rightarrow$ (256x16)	tanh
time distributed dense	(256x16) $\rightarrow$ (256x16)	-
time distributed dense	(256x16) $\rightarrow$ (256x1)	sigmoid

Table 2. Adapted SED-net layer configurations for audio-based hit detection. Conv layers defined by inChannels $\rightarrow$ outChannels, kernelSize, stride. Others by inShape $\rightarrow$ outShape.

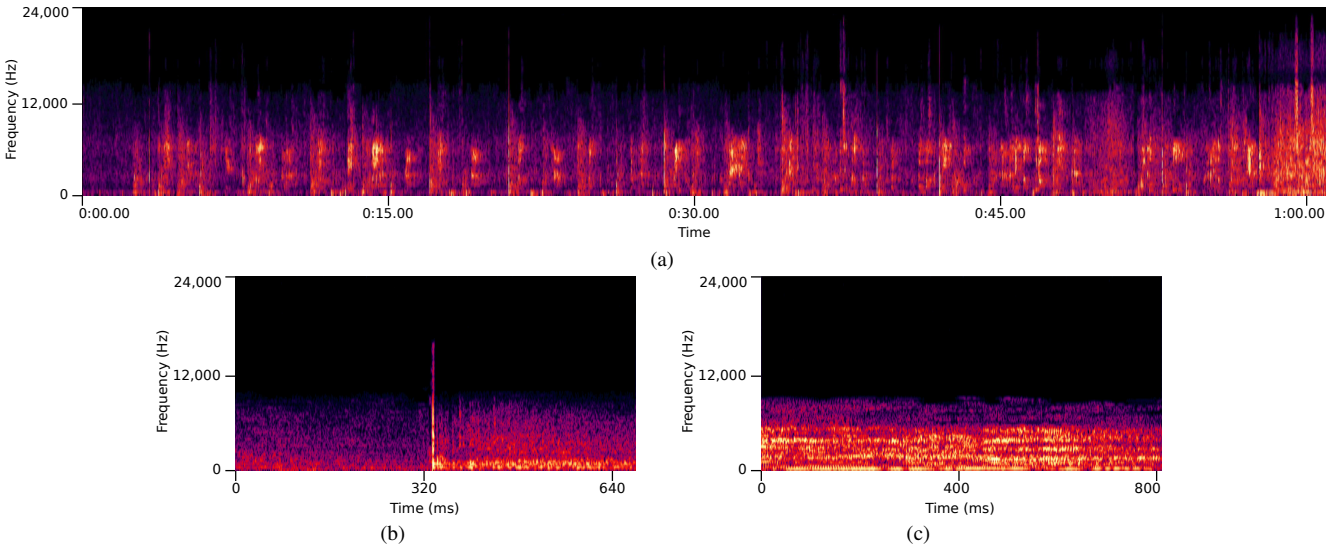


Figure 1. Example rally and two common events: Spectrogram of a padel rally, 61s (a). Its gradient gives an indication of the presence of a hit, but also contains noise artefacts from e.g. footsteps, players speaking, crowd cheers (see end of spectrogram). Spectrogram of a smash, 678ms (b). Spectrogram of one of the players speaking without a hit occurrence, 809ms (c).

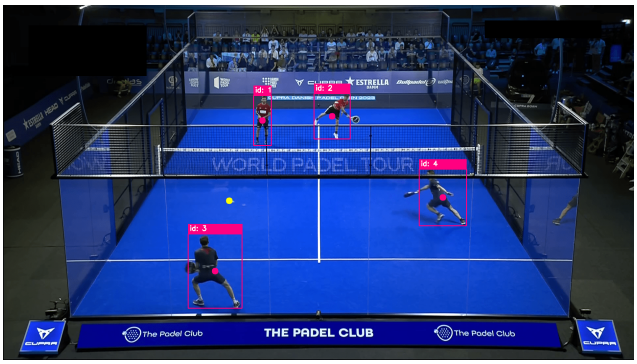


Figure 2. Detection overview at the start of a hit window. Shows the detected players and their corresponding tracking ID according to the rally start positions. Location of the ball is indicated by the yellow circle.

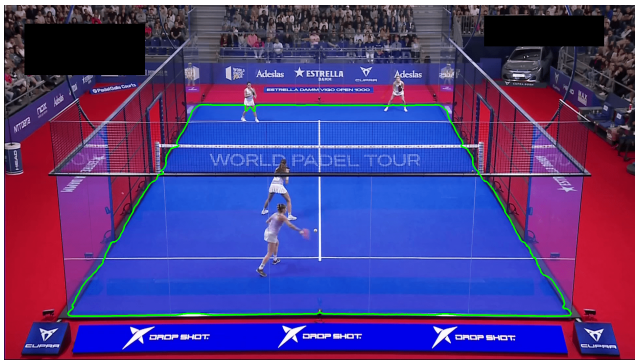


Figure 3. Color-based field extraction for approximating corner positions. Interference from the environment causes inaccurate corner approximations along the contour of the court (green line).

4.3. Ball detection

Knowing when the ball is hit, is crucial, but in order to perform a meaningful analysis on the player positions', it is also key to know which of the 4 players is hitting the ball. Although the inter-player analysis only requires knowledge of which side is attacking, such that the opposite side can be evaluated, it may be useful for other types of analysis to know which specific player hit the ball as well. Therefore the frames before and after a detected hit are taken in consideration regarding the location of the ball.

To obtain its position in the hit window, we utilized a pretrained TrackNet [8] model trained on tennis data. Which closely resembles padel as the balls in both sports look the same, the padel court is similar to a tennis hardcourt and a similar static camera viewpoint is used to film both sports. An example of the output of the ball detection is shown in Figure 2. The ball predictions are further processed to remove outliers to avoid ball teleportation, and interpolates missing values in smaller subsequences of the prediction output.

5. Inter-player distance

5.1. Court detection

The camera used to film the sloped viewpoint does not move during the tournament. Consequently, a single homography can be used for every rally in that tournament (calculated with manually selected and accurate points). Automatic

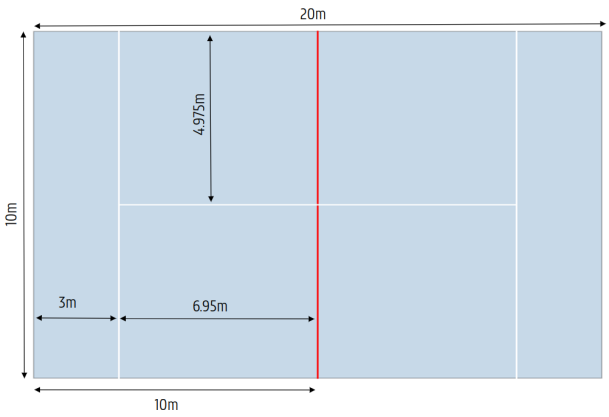


Figure 4. Dimensions of a standard padel court. Measurements up until the edge of the line. Lines have a width of 5cm.

methods for selecting corresponding points may be susceptible to noise, occlusions, or irrelevant features, which can lead to inaccuracies in the estimated homography. This is illustrated in Figure 3. Here a color-based approach was utilized to find the contour of the field and approximate its corners. Using color features of a padel field is generally discouraged as they are not standardized across tournaments. The possibility of using the perpendicular divider lines inside the court was previously researched by Wennerblom and Arronet [24], although the authors report similar inconsistencies as before. Manual selection of points allows users to precisely choose corresponding corners of the field in the images that best represent the desired transformation (ignoring the effect of noisy or occluded regions). This level of control ensures that the selected points accurately capture the geometric transformation between the images, resulting in a more accurate homography and subsequently more accurate distances.

Finally, the transformation matrix of the homography is calculated using the selected corners of the court and their corresponding point the court plane, which is defined according to the dimensions of a padel field (see Figure 4).

5.2. Player detection and tracklet generation

Player bounding box detections and their corresponding poses and tracks are obtained through the YOLO framework provided by Ultralytics (similarly like YOLO-POSE [17]). The detections are further processed to differentiate between players and spectators. By applying a mask on each frame based on the court's coordinates and a padding region, we obtain a trapezoid region by which the bounding box of the pose must overlap to be included (see Figure 5).

Players are assigned a number based on their starting position in the rally. From the camera perspective, the top-side (resp. bottom-side) team is assigned numbers 1 and 2 (resp. 3 and 4) from left to right. Note that the annotations provided in our dataset follow the same assignment procedure. To calculate the initial position of each player we reduce all available poses from the first three seconds of the rally into four average center-coordinates of the bounding boxes. These are then assigned number 1-4 by sorting them based on the x-coordinate into the leftmost and rightmost points. From both categories we take the first two elements sort them by the y-coordinate to differentiate between 1 and 3 in the leftmost points, and 2 and 4 in the rightmost points.

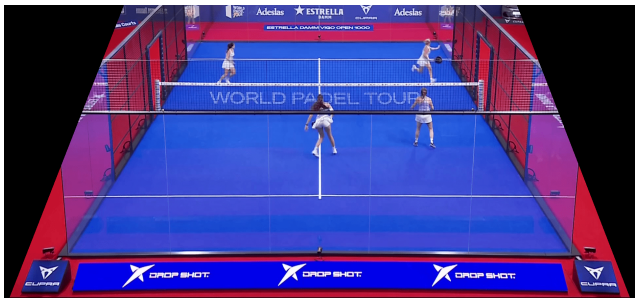
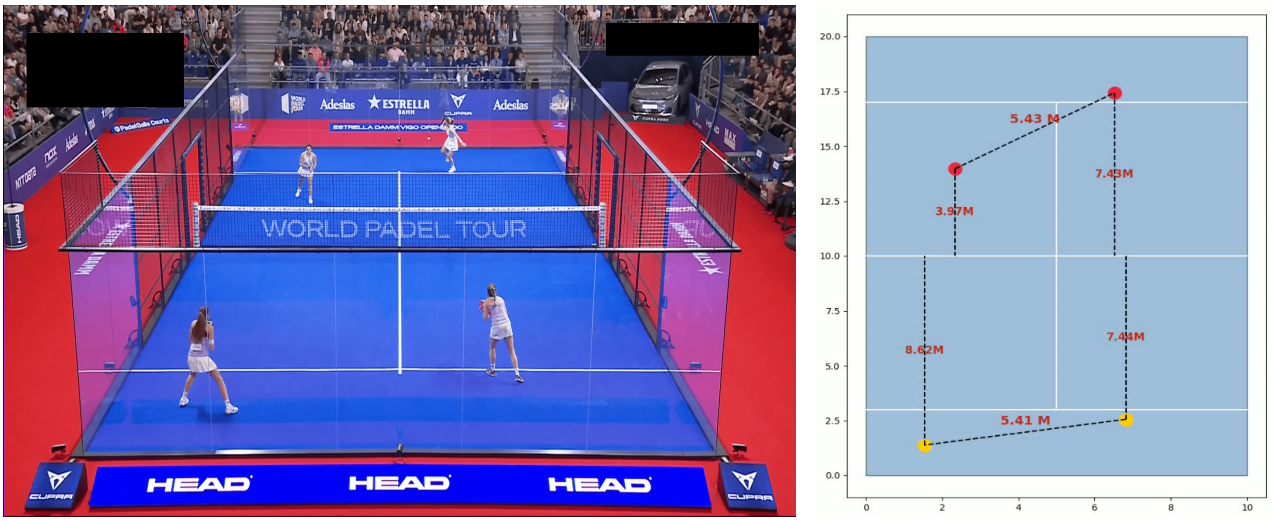


Figure 5. Region-of-interest focused on the padel court. The corresponding mask (based on the court corners) is used to differentiate player poses from those from the audience.

5.2.1 Re-identification and merging tracking IDs

Players are free to roam inside and outside the court during the rally. This means that poses might be (temporarily) occluded. Whenever the player is detected again, it will be tracked under a new ID. In this case it is required that the newly introduced track ID is merged into the main tracklets of one of the four players. It is important that the four main track IDs correspond to the assigned positional ID as explained in the previous section. Such re-identification of poses in multi-object tracking (MOT) is commonly obtained through a visual association metric (e.g., DeepSORT [25] and StrongSORT [5]). A more in depth literature review of MOT techniques has been published by Luo et al. [16]. The visual metrics often work well in the wild as tracked subjects have a lot more variance in their appearance, but in team sports this is more often than not an issue as players wear the same outfit (especially on the same team). In the case of padel we have chosen for an alternative technique using the appear and disappear positions of the tracklets. Because the number of unique tracks we are interested in is known beforehand (4), and we assume that it is very unlikely that both players of a side will be invisible at the same time, we use the points where a pose is lost and where a new one occurs. Using this information we have developed an algorithm (without a predictive model) to deduce to which of the main track collections the newly tracked pose corresponds. The algorithm assigns each tracklet to a player by logic reasoning. For instance



(a) (b)

Figure 6. Video still of a rally (a). Results of the inter player and net distance calculation pipeline (b).

if a tracklet ends for player 1 and the other players remain tracked, a newly started tracklet will be assigned to player 1.

5.3. Player hit assignment and positioning

The first part of this section will describe how hits are assigned to a player/team. If the predicted hit window is long enough, we use the boundaries provided by the model prediction. If not, or if using a ground truth prediction from the video (singular values), the window is padded on both sides until it is 500ms long (corresponds to 12 frames at 25 FPS). For each selected frame we check the number of detected bounding boxes and if the ball's location was predicted. Utilizing multiple frames increase the robustness of the approach as it is still able to assign a player even though some frames have missing or inaccurate information. For frames which include a ball detection we calculate its distance to each of the four players if possible. The distances between the ball and each player are used to obtain the closest player ID in a weighted majority vote. The weights ($W = w_1, w_2, ..., w_N$) are determined by the standardized euclidian distance:

$$w_i = 1 - \frac{d_i - d_{min} + 0.1}{d_{max} - d_{min}} \quad (1)$$

In case of a tie, the player with the closest euclidean distance over all frames is selected. To which point of the player this distance is measured depends on a few conditions:

- If no bounding box is available for the current frame, we calculate the average bounding box for the current player in a symmetrical window of 2 seconds around the current frame timestamp. If the larger interval still has no available poses, there is no distance added for the current player in the voting mechanism, else the center coordinate of the bounding box is used.
- If the bounding box was predicted with an absent pose detection, the center coordinate of the bounding box is used.
- If the bounding box and the pose are detected, the distance of both wrists is calculated with respect to the ball position. Only the smallest distance is kept. In case the pose has no prediction for the wrists, the player's coordinate falls back to the center of the bounding box.

A secondary sweep is performed to fill gaps where there was not enough pose information to assign a team such that the alternating nature of which team hits the ball is not violated. The results are then mapped to a string that indicates the side and player ID.

To calculate the distance measurements, the pose for each player is further reduced to a single point on the court. Before the transformation to the court plane, we calculate the central hip point and use its x-coordinate. The y-coordinate is obtained by averaging the y-coordinate of both ankles. In case the pose is absent but the bounding box is not, we use the bottom middle of the bounding box. Using the four reduced representations, we calculate the distance between both players of each team and for each individual, the distance to the net (centerline of the court). The result of this procedure is shown in Figure 6.

6. Results

The audio-based hit detection model has been validated through a standardized evaluation framework, called sed_eval [18]. It combines class accuracy with temporal

position checks at instance level by evaluating the onset and offset conditions, also called event-based evaluation. Events are either classified as: (i) true positive (TP) if it has a temporal position overlapping with the temporal position of an event with the same label in the ground truth. (ii) false positive (FP) if there is no correspondence to an event with the same label in the ground truth. (iii) false negative (FN) if an event in the ground truth that has no correspondence to an event with same label in the predicted output. All temporal checks must lie within the collar window (in ms). The two metrics used to evaluate the system are: firstly, the F1-score, and secondly, the error rate (ER), which is the ratio of the sum of substituted (S, the number of events with correct temporal position but incorrect class label, irrelevant for binary SED), inserted (I, the number of predicted events unaccounted for as correct or substituted), and deleted (D, the number of ground truth events unaccounted for as correct or substituted) labels compared to the ground truth and the total number of labels (N):

$$ER = \frac{S + D + I}{N} \tag{2}$$

The dataset was divided in four splits without overlapping files (full rallies were assigned to splits) with for each a 70%-30% split between training and testing, or approximately 570 hits. This was evaluated using the event-based metrics with a collar size of 250ms or a minimum overlap of 50% and achieved an average F1-score of 92% ($\sigma = 1.6\%$). The model trained on the first split is used for inference and reported an error rate of 0.16 (0.13 deletion rate, 0.03 insertion rate) on its validation split.

As indicated by the deletion rate, most of the errors are of type two. The most common cause of false negatives were from slice movements which produce little to no sound and are also underrepresented in the dataset as this type of shot is less common then a direct hit. Causes of false positives that we've noticed were racquet hits against the metal frame of the field, players running into the plexiglass or stomping hard on the ground and racquets hitting the floor just after a hit.

In regards to the hit assignment task, a subset of 319 hits of two tournaments of the original dataset is further annotated for evaluation purposes. It includes for each hit window, the timestamp of the closest video frame where the hit is visible and the ID of the player that hit the ball (according to the assignment scheme in Section 5.2). As the system output can be used to assign to one of the individual players as well as a team, we will report metrics for both assignment types. Our mapped output corresponds to the labels in the ground truth. This yields an accuracy for each evaluated rally that is weighted accordingly to its total number of hits before calculating the global accuracy. This achieved an accuracy of 83.70% for player specific assignment and

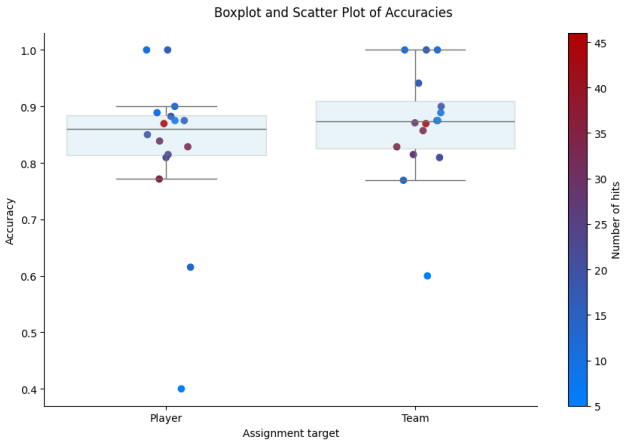


Figure 7. Boxplot and scattered accuracy values of hit assignments for each rally. Compares both player and team specific assignment.

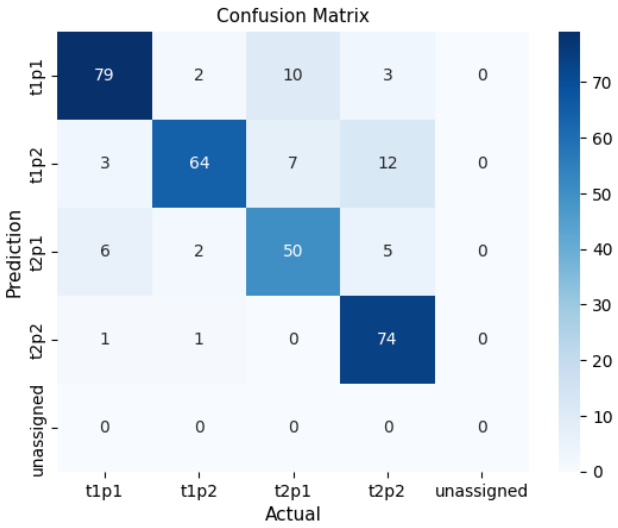


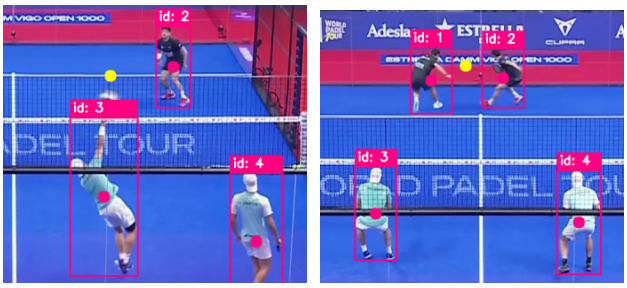
Figure 8. Confusion matrix of the hit assignments. Unassigned may occur if the hit window has no frames for which both ball location and enough player detections are known.

86.83% if only the team indicator is considered. The accuracy distribution of the rallies is visualized in Figure 7. Further investigation showed that rallies with a considerably lower accuracy contain less hits. Therefore has a single missed hit a large impact on the overall rally accuracy. Discrepancies between both assignment types, as shown in Table 3, are caused by missing poses by 1 or 2 players but still recoverable such that the correct side is determined.

From the confusion matrix in Figure 8 we can see that most errors originate between opposing players (1 with 3 and 2 with 4). This is due to the limitations of a single camera viewpoint. At some point, when the player on the far-

Rally ID	P2P Acc. (%)	Team Acc. (%)	Hits
1	87.50	87.50	16
2	83.87	87.10	31
3	88.89	88.89	9
4	61.54	76.92	13
5	87.50	87.50	8
6	100.00	100.00	10
7	81.48	81.48	27
8	88.24	94.12	17
9	100.00	100.00	16
10	77.14	82.86	35
11	90.00	100.00	10
12	86.96	86.96	46
13	85.00	90.00	20
14	80.95	80.95	21
15	82.86	85.71	35
16	40.00	60.00	5
	$\mu = 83.70$	$\mu = 86.83$	319

Table 3. Overview of individual rally accuracies for player-to-player (P2P) and team assignment. With μ calculated as the weighted average using the number of hits.



(a) (b)

Figure 9. Examples of assignment failure: Distance between the ball and players 1 and 3 is very similar due to the limitations of a 2D representation (a). Defending players both try to hit the ball such that they are both very close to the ball (b).

side moves closer to the net, their detection start to overlap with the opposite player due to the depth perception problem and the limitations of working with a 2D representation. A similar circumstance is shown in Figure 9a. Player 3 is about to smash the ball but due to the camera viewpoint, the distance from the ball to player 2 is close to that of the ball and player 3, possibly causing an incorrect assignment. Less common but also possible, is when players of the same side move towards the ball to try and hit it, causing ambiguity in the assignment algorithm. This is illustrated in Figure 9b.

Implemented procedures like the majority voting or multi-frame assignments are used to make the system more robust against these types of errors. But in situations where there is simply not enough information for prolonged periods of time, the system will not be able to recover until the it becomes available again later in the rally.

7. Conclusion

In this paper, we propose a multi-modal method to detect and analyse padel hits using audio and video features. For this purpose, we introduce a new dataset consisting of videos of padel rallies of high level tournaments which will remain public for others to work with. The padel dataset comprises of 2377 hit annotations and a subset of 319 also indicate which player has hit the ball.

These methods can be used to automatically identify and track the four players in the rally and calculate the inter-player distance and their placement with respect to the net. This is valuable information which can be used by coaches to analyse field coverage and attacking strategies. The developed predictive model for hit detection based on audio signals demonstrates robust performance, with an average F1-score of 92%, highlighting its effectiveness in identifying the most interesting frames during rallies. Additionally, our system employs a custom re-identification and voting algorithm to assign hits to individual players. This system achieves an overall accuracy of 83.70% for player-specific assignment and 86.83% for team-based assignment, providing valuable insights into player movements and interactions.

Overall, this research contributes to a deeper understanding of padel gameplay dynamics and offers practical implications for coaching staff and players to optimize performance and strategic decision-making. Future research directions may include further refining the predictive models into a single unified model, exploring additional performance metrics utilized by coaches to quantify attacking/defensive quality of the rally, and analyzing player movement patterns to provide more comprehensive insights into padel gameplay.

References

[1] Sharath Adavanne, Pasi Pertilä, and Tuomas Virtanen. Sound event detection using spatial features and convolutional recurrent neural network, 2017. 3

[2] Aaron Baughman, Eduardo Morales, Gary Reiss, Nancy Greco, Stephen Hammer, and Shiqiang Wang. Detection of tennis events from acoustic data. pages 91–99, 2019. 2

[3] Hua-Tsung Chen, Wen-Jiin Tsai, Suh-Yin Lee, and Jen-Yu Yu. Ball tracking and 3d trajectory approximation with applications to tactics analysis from single-camera volleyball sequences. *Multimedia Tools and Applications*, 60(3):641–667, 2012. 1

[4] Daniel Dios. Padel tactics: How close should players play from the net? https://padeltrainer.com/padel-tactics-how-close-should-players-play-from-the-net, 2018. Online; accessed 10/03/2024. 1

[5] Yunhao Du, Zhicheng Zhao, Yang Song, Yanyun Zhao, Fei Su, Tao Gong, and Hongying Meng. Strongsort: Make deepsort great again, 2023. 5

[6] Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, Takaaki Hori, Jonathan Le Roux, and Kazuya Takeda. Duration-controlled lstm for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(11):2059–2070, 2017. 2

[7] Toni Heittola, Annamaria Mesaros, Antti Eronen, and Tuomas Virtanen. Context-dependent sound event detection. *EURASIP Journal on Audio, Speech, and Music Processing*, 2013(1):1, 2013. 2

[8] Yu-Chuan Huang, I-No Liao, Ching-Hsuan Chen, Tsi-Ui Ik, and Wen-Chih Peng. Tracknet: A deep learning network for tracking high-speed and tiny objects in sports applications. *CoRR*, abs/1907.03698, 2019. 2, 4

[9] Jihye Hwang, Sungheon Park, and Nojun Kwak. Athlete pose estimation by a global-local network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 114–121, 2017. 2

[10] C. Ingwersen, C. Moller Mikkelstrup, J. Jensen, M. Rieger Hannemose, and A. Dahl. Sportspose - a dynamic 3d sports pose dataset. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 5219–5228, Los Alamitos, CA, USA, 2023. IEEE Computer Society. 2

[11] Mohammadreza Javadiha, Carlos Andujar, Enrique Lacasa, Angel Ric, and Antonio Susin. Estimating player positions from padel high-angle videos: Accuracy comparison of recent computer vision methods. *Sensors*, 21(10), 2021. 2

[12] Paresh R. Kamble, Avinash G. Keskar, and Kishor M. Bhurchandi. Ball tracking in sports: a survey. *Artificial Intelligence Review*, 52(3):1655–1705, 2019. 1

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 3

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018. 3

[15] Rui Lu, Zhiyao Duan, and Changshui Zhang. Multi-scale recurrent neural network for sound event detection. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 131–135, 2018. 2

[16] Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, and Tae-Kyun Kim. Multiple object tracking: A literature review. *Artificial Intelligence*, 293:103448, 2021. 5

[17] Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. pages 2636–2645, 2022. 5

[18] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. Metrics for polyphonic sound event detection. *Applied Sciences*, 6(6), 2016. 6

[19] Annamaria Mesaros, Toni Heittola, Tuomas Virtanen, and Mark D. Plumbley. Sound event detection: A tutorial. *IEEE Signal Processing Magazine*, 38(5):67–83, 2021. 2

[20] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2016. 2

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2015. 2

[22] Nayara M. S. Rocha, Milena F. Pinto, Iago Z. Biundini, Aurelio G. Melo, and André L. M. Marcato. Analysis of tennis games using tracknet-based neural network and applying morphological operations to the match videos. *Signal, Image and Video Processing*, 17(4):1133–1141, 2023. 2

[23] Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2022. Open source software available from https://github.com/heartexlabs/label-studio. 3

[24] David Wennerblom and Andrey Arronet. Padel court detection system, 2023. Available at https://www.diva-portal.org/smash/get/diva2:1770685/FULLTEXT01.pdf (accessed on 05/03/2024). 5

[25] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017. 5

[26] Chen Yihan, Guo Min, and Li Zhiqiang. Sound event detection based on bidirectional temporal convolutional network and gated recurrent unit. In *2021 20th International Conference on Ubiquitous Computing and Communications (IUCC/CIT/DSCI/SmartCNS)*, pages 445–450, 2021. 2

[27] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3320–3328, Cambridge, MA, USA, 2014. MIT Press. 2

[28] Dan Zecha, Moritz Einfalt, Christian Eggert, and Rainer Lienhart. Kinematic pose rectification for performance analysis and retrieval in sports. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1872–18728, 2018. 2

[29] Emre Çakır, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(6):1291–1303, 2017. 2