Melih Kurtaran

11761 – Image and Video Analysis Project 2 Report

Introduction

Ball tracking technology is utilized in sports and other applications for monitoring the movement and location of a ball. The objective of this project is to perform an analysis of tennis matches by tracking the ball and player movements, counting the number of strikes, and determining the winner of the game. The methodology employed involves using advanced algorithms and sensors to track the ball in real-time and provide accurate data on its position and velocity. This information is then processed and analyzed to provide insights into the game's performance and outcome.

Design and Implementation

The design of the implementation encompasses several stages. The initial positions of the players are determined based on the video, and the filename is provided for the mean-shift algorithm. During the course of the game, various variables are recorded, including the scores of the players, whose turn it is, and whether the game has ended.

The implementation consists of three main functions. The first function, create_mask_and_hist, generates a mask and histogram for a specified region of interest within the frame. The second function, process_frame, performs background subtraction on the specified region of interest, followed by median blurring and morphological operations to eliminate noise and retain only the objects of interest for tracking. This function returns two variables: the track windows for the players after the mean-shift algorithm has been applied, and a list of contours, which are potential candidates for the ball. The third function, writeScore, writes the number of hits for each player but does not return any values.

Both Kalman filter and simple contour detection are utilized for ball tracking. The logic of the Kalman filter is not implemented from scratch, but rather, a publicly available class (KalmanFilter.py) [1] is employed for this purpose. Estimated, predicted, and measured positions of the result of Kalman filter is drawn in the videos. The resulted video with all players and ball tracking, number of hits and the announcement of the winner gets created after running the code.

Algorithm Pseudocode

1. Import necessary libraries and modules: OpenCV, NumPy, and KalmanFilter.
2. Define the ball tracking function
3. Convert the input frame to HSV color space using the "cv2.cvtColor" function.
4. Threshold the HSV frame to extract only the desired color range using the "cv2.inRange" function.
5. Perform morphological operations on the thresholded image to reduce noise and improve the tracking accuracy.
6. Find contours in the thresholded image and filter out contours that are too small or too large using the "cv2.findContours" and "cv2.contourArea" functions.
7. Find the largest contour in the filtered contours and compute its moments using the "cv2.moments" function.
8. Extract the ball's center coordinate and the radius using the "cv2.minEnclosingCircle" function.
9. Draw a circle around the ball's center and display the frame with ball tracking information using the "cv2.circle" and "cv2.imshow" functions.
10. Repeat steps 3 to 9 for every frame in the video stream until the stream ends or the user presses the "q" key.

Strong and weak points

The implementation of the project has successfully accomplished all the specified objectives. The movement and position of both the players and the ball are continuously monitored, with the exception of instances where the ball is in contact with a player or outside the defined observation region. The accurate calculation of the number of hits made by each player is consistently displayed, and the determination of the game winner is accurately predicted. The ball hitting count is updated 0.3 seconds after the impact with the racket, as the direction of the ball is determined by examining its previous locations in the last one third of a frame per second.

The background subtraction component of the implementation could be further optimized as, while a majority of background noise has been successfully removed, a limited amount of undesirable noise persists. This presents a challenging aspect as it involves balancing the preservation of the ball's real-time information with an adequate level of noise removal.

Another improvement could be tracking the ball even though it is interacted with the players. The current implementation employs a margin value of 5 pixels, which results in non-detection of the ball if it is situated closely to the player's contour. This is to minimize the detection of erroneous contours that may arise from player movements. Also, their interaction with the ball is predicted by their distance to the ball for determining the hit but it is possible that a player can get very close to the ball but still miss the shot. In such circumstances, the algorithm does not reflect a successful hit, as it takes into account the direction of the ball as well. However, this approach has limitations, as ball bounces along the y-axis may result in incorrect hit detections. This is due to the fact that the analysis of the video is performed in 2D, only considering the y-axis for determining the direction of the ball, despite the actual game being played in a 3D environment.

Discussion

The implementation of ball tracking technology in this project has proven to be effective in tracking the movement and position of a ball in tennis matches. The results have shown that the algorithm is able to continuously track both players and the ball, accurately count the number of hits, and determine the winner of the game.

The current implementation can be further improved by optimizing the background subtraction component and exploring other ball tracking algorithms that could lead to better performance and accuracy. Additionally, the project could be expanded to track more players, analyze more types of sports, or incorporate additional data sources to provide a more comprehensive analysis of the game. Another potential area of exploration is the integration of machine learning algorithms to automatically identify players, track the ball, and predict the outcome of the game, thereby reducing manual input and increasing accuracy.

Result

The accompanying figures one and two depict snapshots from the monitored tennis matches, wherein the movement and position of both the players and the ball are continually monitored. The scores are updated in real-time, reflecting the dynamic nature of the match.

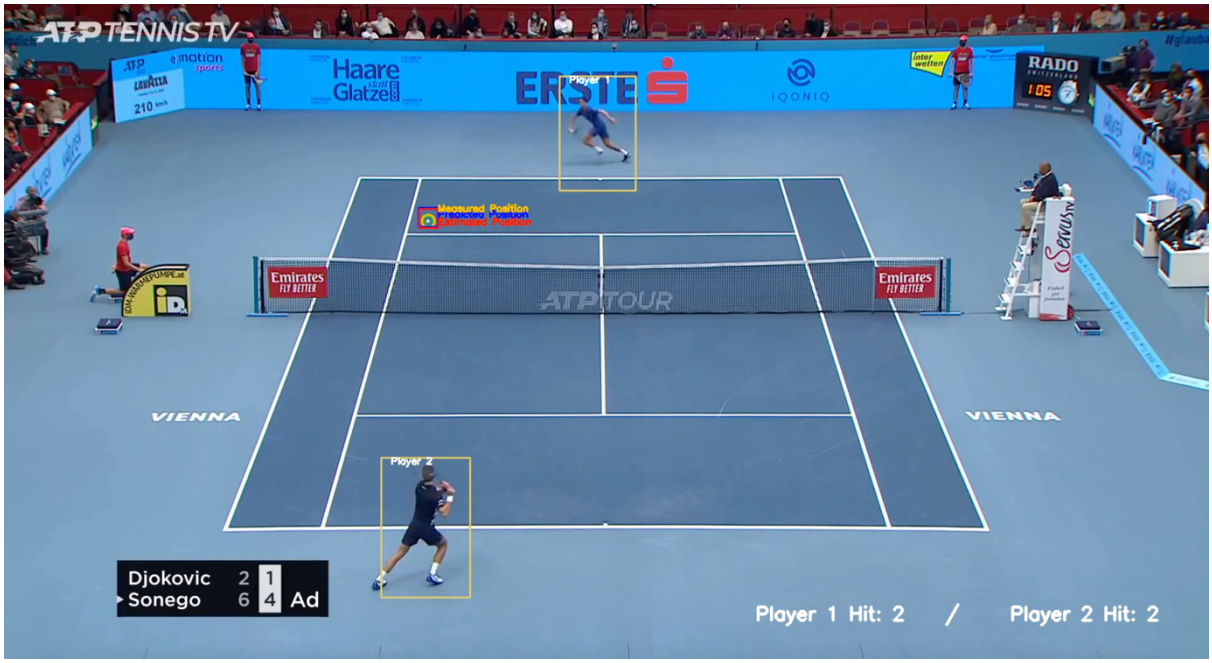

Figure I: The first tennis match

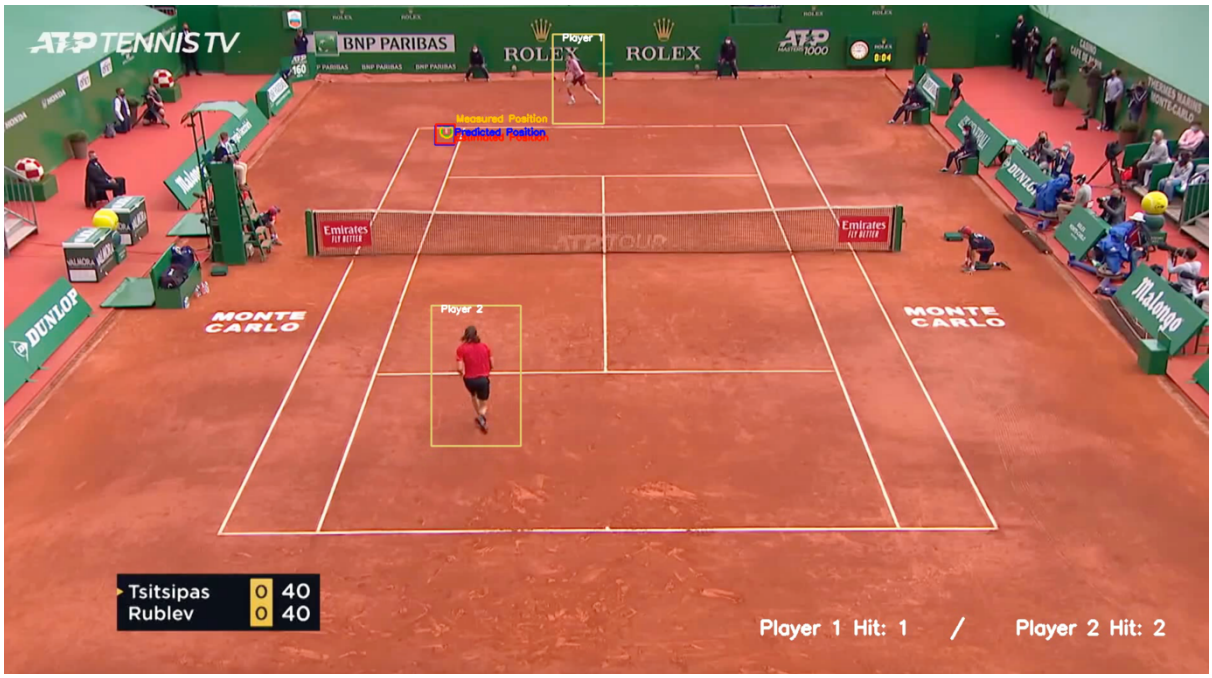

Figure II: The second tennis match

References

[1] Sadli, R. (n.d.). Rahmadsadli/2-D-kalman-filter: 2-D Kalman filter for tracking a randomly moving object. GitHub. Retrieved January 30, 2023, from https://github.com/RahmadSadli/2-D-Kalman-Filter