High-Speed Object Detection in Tennis Videos

Francois Gaston, Ippolito Martin and Fuster Marina

Instituto Tecnológico de Buenos Aires (ITBA)

Iguazú 341, Ciudad Autónoma de Buenos Aires, Argentina

Abstract. Today, where technology and data analysis have revolutionized multiple industries, sports is no exception. In both team and individual sports, sports statistics enable data-driven decision making, which has created a significant competitive advantage.

However, these metrics are still far removed from amateur athletes, who do not have access to reliable metrics, making it difficult to track their performance.

For all these reasons, the investigation of methods for detecting player, ball and court boundary movement is presented, with the aim of obtaining tennis match metrics using accessible equipment. In particular, as part of the integral solution, a method for high-speed object recognition is proposed, whose computational cost and performance is better than current proposals.

Keywords: Computer Vision, Machine Learning, Object Detection, Motion Analysis, Tennis.

Detección de objetos a alta velocidad en videos de tenis

Resumen. En la actualidad, donde la tecnología y el análisis de datos ha revolucionado múltiples industrias, el deporte no es la excepción. Tanto en deportes de equipo como en individuales, las estadísticas deportivas permiten tomar decisiones basadas en datos concretos, lo que ha generado una ventaja competitiva significativa.

Sin embargo, estas métricas aún están muy alejadas de los deportistas aficionados, quienes no cuentan con acceso a métricas de forma fiable, dificultando el seguimiento de su rendimiento.

Por todo esto, se presenta la investigación de métodos para detectar el movimiento de los jugadores, de la pelota y de los límites de la cancha, con el objetivo de obtener métricas de partidos de tenis utilizando equipamiento accesible. En particular, como parte de la solución integral, se propone un método para reconocimiento de objetos a alta velocidad, cuyo costo computacional y performance resulta mejor que propuestas actuales

Palabras clave: Visión por computadora, Aprendizaje Automático, Detección de objetos, Análisis de movimiento, Tenis.

1 *Métodos de detección*

Utilizar el tenis como deporte de estudio resulta una ventaja respecto a los demás deportes, ya que solamente con seguir al jugador, la pelota y la cancha se puede analizar un partido. Para tomar referencia de la manera de captura y poder explicar las detecciones usaremos un video de ejemplo[1].

1.1 Detección del jugador: *Pose estimation*

Mediante un modelo abierto y preentrenado de *Pose estimation* (Bazarevsky, 2021) se obtienen 33 puntos de la persona por cuadro. Se elige el modelo en particular por contar con una buena relación entre tiempo de inferencia y cantidad de puntos de detección[2].

1.2 Detección de la pelota: *Método propio "XOR"*

Tomando 3 cuadros consecutivos A, B y C y utilizando la librería OpenCV (Bradski, 2000) se computa la diferencia entre cuadros, es decir, se busca mantener solamente los píxeles que van a cambiar en el próximo cuadro. Para tal motivo se aplica el método XOR entre A y B, entre A y C y entre las dos diferencias, expresado matemáticamente tenemos que para cada píxel del cuadro aplicamos *AB*= XOR(A, B); luego *AC* = XOR(A, C) y finalmente *cuadro_final* = XOR(AB, AC).

$$XOR(A,B,X,Y) = \begin{cases} RGB(A,X,Y) & \text{si } RGB(A,X,Y) \neq RGB(B,X,Y) \\ [0,0,0] & \text{si } RGB(A,X,Y) = RGB(B,X,Y) \end{cases}$$

De este proceso se obtienen los píxeles de A que van a cambiar en B, es decir, los píxeles que están en movimiento. Técnicas similares resultan habituales en sistemas de detección de movimiento, pero resulta una innovación para el seguimiento de objetos[3]. (Husein, 2017)

Tomando solamente los píxeles en movimiento se procede a realizar una clusterización mediante DBSCAN tomando como datos de entrada la posición X e Y de cada píxel. Como resultado de la clusterización se logra diferenciar cada jugador, la pelota y el resto de píxeles en movimiento[4].

Finalizado este proceso se busca mantener solo el clúster correspondiente a la pelota, por lo que se itera por todos los *clusters* y se mantienen solo los que cumplan 3 condiciones. Primero el color promedio del clúster debe encontrarse dentro de un rango RGB, luego al menos un porcentaje de los píxeles se debe encontrar en el rango

[1] https://bit.ly/JAIIO-VIDEO-ORIGINAL
[2] https://bit.ly/JAIIO-TENNIS-POSE
[3] https://bit.ly/JAIIO-XOR-FRAMES
[4] https://bit.ly/JAIIO-XOR-DBSCAN

RGB y finalmente el desvío estándar debe ser menor que un umbral, este último ya que la variación de color en el clúster de los jugadores es mucho mayor que la variación en el clúster de la pelota[5].

Como dato final a guardar se conservan todos los píxeles de los *clusters* que hayan pasado por todos los filtros en formato de lista de píxeles.

Una posible variación al método recae en no clusterizar y aplicar filtros RGB a los píxeles por separado que surgen del método XOR. En este caso el método resulta mejor en términos de tiempo pero errático en términos de rendimiento, ya que ante cualquier objeto en el rango de RGB de la pelota que se encuentre en movimiento, no se lo podrá filtrar y perjudicará la detección puntual de la pelota.

1.3 Detección de la cancha: *Transformada Hough*

Mediante un modelo de detección de objetos ajustado a la detección de canchas de tenis (YOLO, 2025), se limita la imagen solo al rectángulo de la detección eliminando el ruido externo a la cancha. Luego, se pasa la nueva imagen a escala de grises y mediante OpenCV se obtiene la representación Canny. Una vez aquí se aplica la transformada de Hough probabilística (OpenCV 2025) y se obtiene un conjunto de líneas. Para sintetizar estas líneas, se utiliza DBSCAN, representando cada línea como su pendiente y ordenada al origen.

Las líneas resultantes se clasifican según su pendiente para determinar las líneas de la cancha. Finalmente, los vértices de la cancha se obtienen como intersecciones entre estas líneas, constituyendo la salida del proceso[6].

2 Resultados

2.1 Elaboración del conjunto de datos

Se construyó un conjunto de datos propio, compuesto por 3 videos con la misma disposición y resolución. Estos videos fueron elegidos con el fin de abarcar diferentes superficies (polvo de ladrillo y cemento de dos colores) e iluminaciones (luz natural y luz artificial). Posteriormente se les hizo un etiquetado manual donde se tomó la posición de la pelota (P1), del pie izquierdo y derecho del jugador (M1 y M2) y de los vértices de la cancha (C1, C2, C3 y C4),

2.2 Detección de la pelota

Para evaluar la detección de la pelota, se calcula en cada cuadro de video i el promedio de las coordenadas de los píxeles detectados, obteniendo un único punto de

[5] https://bit.ly/JAIIO-XOR-FILTRO
[6] https://bit.ly/TENIS_CANCHA

referencia Di que se compara con la posición etiquetada manualmente Pi. Se considera válida la detección si la distancia euclidiana entre ambos es menor a 10 píxeles. La tasa de aciertos se define como:

$$\text{Tasa de Aciertos} = \frac{1}{N}\sum_{i=1}^{N} 1\left(\|Di - Pi\| < 10\right)$$

Donde N es la cantidad de cuadros de video etiquetados. Se descartan aquellos cuadros donde la pelota no se puede detectar o cuando se encuentra del otro lado de la cancha, por tales motivos solo el 32.24% de los cuadros de video son tenidos en cuenta.

En cuanto a los resultados, en la *Tabla 1* se expone el promedio de la tasa de aciertos. Sobre estos resultados, notamos no solamente un rendimiento superior en detección sino también un rendimiento equivalente en términos de tiempos de cómputo.

Tabla 1. Resultados de la detección de la pelota.

Método	Tasa de aciertos	Tiempo (Seg)	Tiempo por cuadro (Seg)
YOLO Object detection[7]	0.30	119 ± 4	0.19 ± 0.06
Xor	0.40 ± 0.32	51 ± 4	0.067 ± 0.03
Xor DBSCAN	0.77 ± 0.01	117 ± 3	0.19 ± 0.08

2.3 Detección del jugador

Al evaluar la detección del jugador, se calcula en cada cuadro de video i el promedio de las posiciones de los pies detectados (Di) y de los pies etiquetados manualmente (Mi). Se considera válida la detección si la distancia euclidiana entre ambos es menor a 60 píxeles. La tasa de aciertos se define como:

$$\text{Tasa de Aciertos} = \frac{1}{N}\sum_{i=1}^{N} 1\left(\|Di - Mi\| < 10\right)$$

Donde N es la cantidad de cuadros de video etiquetados. Utilizando este método contamos con una tasa de aciertos variable entre videos de 0.5 ± 0.1.

2.4 Detección de la cancha

A la hora de analizar la detección de la cancha no se puede recurrir a los métodos utilizados previamente, ya que no se puede resumir a un solo punto. Motivados por el estado del arte en detección de objetos, realizamos para 1 de cada 50 cuadros de video el cálculo de IoU (*Intersection over Union*), el cual consiste en el cociente entre la interacción entre el área detectada y el área etiquetada sobre la unión de ambas áreas. Utilizando este método contamos con un resultado promedio de IoU de 0.96 ± 0.03.

[7] Se considera un modelo de detección de objetos cómo referencia del estado del arte actual (YOLO, 2025)

3 Trabajo Futuro

En términos de trabajo futuro, se buscará ampliar el conjunto de videos evaluados para mejorar la robustez de los resultados y analizar el rendimiento en distintos escenarios. También se podrán incorporar pruebas con ruido para evaluar la estabilidad de los métodos ante condiciones adversas. Además, se continuará trabajando en el ajuste de parámetros para reducir detecciones erróneas o poco fiables. Como parte de estas mejoras, será fundamental establecer una línea de base clara y documentada, que permita comparar objetivamente el desempeño actual con futuras versiones del sistema.

References

1. Bazarevsky V., Grishchenko I., Bazavan E. G. (2021), BlazePose: On-device Real-time Body Pose tracking, CVPR Workshop on Computer Vision for Augmented and Virtual Reality.

2. Bradski, G. (2000). La biblioteca OpenCV. Revista de herramientas de software del Dr. Dobb .

3. Husein, A. M., Calvin, D., Halim, D., Leo, R., & William. (2017). Motion detect application with frame difference method on a surveillance camera.

4. YOLO. (s.f.). YOLO Homepage. Recuperado el 31 de marzo de 2025 de https://yolov11.com/.

5. Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise*. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining.

6. OpenCV. (12 de abril de 2025). Feature detection — OpenCV documentation. OpenCV. https://docs.opencv.org/3.4/dd/d1a/group__imgproc__feature.html#ga8618180a5948286384e3b7ca02f6feeb