



## **TRABAJO DE FIN DE MÁSTER**

# **GENERACIÓN AUTOMATIZADA DE ESTADÍSTICAS DE FÚTBOL DESDE UNA ÚNICA CÁMARA DE TELEVISIÓN**

**Gerard Naharro López**

**Máster Universitario en Sistemas Inteligentes (MUSI)**

**Especialidad: Inteligencia Artificial**

**Centro de Estudios de Posgrado**

**Año Académico 2023-24**

# **GENERACIÓN AUTOMATIZADA DE ESTADÍSTICAS DE FÚTBOL DESDE UNA ÚNICA CÁMARA DE TELEVISIÓN**

**Gerard Naharro López**

**Trabajo de Fin de Máster  
Centro de Estudios de Posgrado  
Universidad de las Illes Balears**

**Año Académico 2023-24**

Palabras clave del trabajo:

Artificial Intelligence (AI), Football Analytics, YOLO

*Nombre Tutores del Trabajo: Gabriel Moyà Alcover y José María Buades Rubio*

# Generación Automatizada De Estadísticas De Fútbol Desde Una Única Cámara De Televisión

Gerard Naharro López

Tutores: Gabriel Moyà Alcover y José María Buades Rubio

Trabajo de fin de Máster Universitario en Sistemas Inteligentes (MUSI)

Universitat de les Illes Balears

07122 Palma, Illes Balears, Espanya

gerardnaharlopez@gmail.com

**Resumen**—Hoy en día, el fútbol es uno de los deportes más populares y seguidos a nivel mundial. En este proyecto, hemos desarrollado un sistema automatizado para analizar partidos de fútbol, con el objetivo de obtener métricas precisas de posesión del balón por equipos y zonas del campo, además de crear un minimapa que visualice el estado del terreno de juego en tiempo real. Utilizamos una red YOLOv8 pre-entrenada, optimizada para detectar jugadores, la pelota y los árbitros en cada fotograma del vídeo. Evaluamos la precisión de nuestro sistema mediante varios conjuntos de pruebas, cada uno compuesto por diversos vídeos de partidos. Además, discutimos posibles mejoras, como la implementación de algoritmos de seguimiento avanzados (BoTSORT o ByteTrack).

## ABSTRACT

Nowadays, football is one of the most popular and followed sports worldwide. In this project, we have developed an automated system to analyse football matches, with the aim of obtaining accurate ball possession metrics by teams and areas of the pitch, as well as creating a mini-map that visualises the state of the pitch in real time. We use a pre-trained YOLOv8 network, optimised to detect players, the ball and referees in every frame of the video. We evaluated the accuracy of our system through several test sets, each consisting of several match videos. In addition, we discussed possible improvements, such as the implementation of advanced tracking algorithms (BoTSORT or ByteTrack). We expect that this automated system has the potential to contribute significantly to the analysis and understanding of football matches, providing useful tools for both coaches and sports analysts.

**Index Terms**—Artificial Intelligence (AI), Football Analytics, YOLO

## I. INTRODUCCIÓN

El fútbol, como uno de los deportes más populares a nivel mundial, ha evolucionado más allá de ser simplemente un pasatiempo para convertirse en un fenómeno cultural que mueve millones de seguidores y cantidades importantes de dinero en la industria deportiva. Esta creciente relevancia ha llevado a una inevitable fusión con los campos punteros de la tecnología, donde la inteligencia artificial (IA) y la visión por computador han emergido como herramientas clave en el análisis deportivo. Estas, no solo ofrecen una manera innovadora de entender y mejorar el rendimiento en el campo, sino

que también abren nuevas posibilidades para la generación de estadísticas e información.

En el ámbito del análisis deportivo, el uso de técnicas de inteligencia artificial (IA) y visión por computador han revolucionado la forma en que se extraen estadísticas y se generan *insights* a partir de eventos deportivos. En este trabajo de fin de máster (TFM), se presenta un programa diseñado para analizar partidos de fútbol utilizando técnicas de IA y visión por computador.

El objetivo principal de este proyecto es desarrollar un sistema que, a partir de un único vídeo de un partido de fútbol capturado por una cámara de televisión estándar, pueda extraer estadísticas sobre la posesión del balón y proporcionar una representación visual en forma de minimapa de la ubicación de los jugadores y la pelota sobre el campo. Este enfoque presenta distintos desafíos, ya que estamos limitados por la disponibilidad de una única cámara y la necesidad de realizar un tratamiento secuencial de los fotogramas del vídeo, lo que implica que solo tenemos el fotograma actual y los fotogramas pasados.

El sistema desarrollado emplea varias técnicas de IA y visión por computador para lograr sus objetivos. Entre ellas, se utiliza una red neuronal convolucional, específicamente una implementación de YOLO (You Only Look Once) [13], para la detección de la pelota, los jugadores y los árbitros en cada fotograma del vídeo. Además, se aplican técnicas de filtrado por colores para determinar a qué equipo pertenece cada jugador.

En cuanto a la creación del minimapa, el sistema utiliza diversas redes neuronales para transformar la imagen del partido de fútbol en una representación que contiene únicamente las líneas del campo. Esta imagen simplificada se compara con una base de datos de pares de imágenes de líneas de campo y sus homografías correspondientes. Esto nos permite encontrar la correspondencia más similar y aplicar la homografía para transformar la perspectiva original a la vista del minimapa.

Es importante destacar que, debido a que la detección de la pelota no es perfecta en todos los casos, se implementa un método de predicción de su posición basado en la información de fotogramas anteriores.

Este trabajo representa una aproximación al análisis automatizado de partidos de fútbol a partir de vídeos de una sola cámara de televisión, aprovechando el potencial de la IA y de la visión por computador para proporcionar estadísticas sobre

el partido.

## II. ESTADO DEL ARTE

En este apartado, se presenta un análisis del panorama actual de la investigación relacionado con el análisis automatizado de partidos de fútbol. Se examinan los proyectos más relevantes que hagan uso de inteligencia artificial (IA) y visión por computador dentro de este contexto. Todo esto, proporciona información clave para comprender la contribución de este trabajo dentro del campo.

Uno de los proyectos más recientes que podemos encontrar se trata del presentado en 2022 por Naik *et al.* [9]. Este proyecto propone una metodología que hace uso de la red neuronal convolucional YOLOv4 para la detección rápida y precisa de jugadores, pelotas y árbitros en vídeos de partidos de fútbol. Además, utiliza un modelo de seguimiento en tiempo real (SORT) modificado para mejorar el seguimiento de objetos a lo largo de los fotogramas. Logrando muy buen rendimiento con una velocidad de detección de 23 fotogramas por segundo. En el caso de nuestro proyecto, no hacemos seguimiento de jugadores, sin embargo, proporcionamos estadísticas del partido y una visión de la distribución de los jugadores y la pelota sobre un mapa 2D.

Otro proyecto el cual debemos destacar debido a su calidad es el propuesto por Descoins y Marvid [6] llamado “Using AI to compute ball possession”. Propone una solución automatizada para calcular la posesión del balón en partidos de fútbol utilizando técnicas de aprendizaje profundo y visión por computador. Basando su proyecto en el seguimiento de objetos claves (Norfair). A diferencia de nuestro enfoque, que se centra en la extracción de estadísticas detalladas de la posesión y en la creación del minimapa, este proyecto se centra en determinar la posesión total por equipo, además de incluir métodos muy sofisticados para la clasificación por equipo y un detector de pases (ambos permitidos por el seguimiento de jugadores y balón).

Dentro del estado del arte, se pueden encontrar también otros proyectos interesantes como el propuesto por Sarkar *et al.* [11]. Este último tiene como objetivo automatizar el cálculo de estadísticas de posesión del balón en un vídeo de fútbol mediante un enfoque basado en un modelo de red de coste mínimo para detectar eventos de pase de balón. Se basa en un marco teórico de teoría de grafos, donde se modelan eventos de inicio y final de pase de balón así como componentes conectados que se separan o se fusionan entre fotogramas consecutivos. Las mayores diferencias con nuestro enfoque radican en la manera de detectar los jugadores, el desglose de la posesión de balón y en que nuestro proyecto incorpora la creación del minimapa con la localización de los jugadores sobre el campo en una vista de pájaro en dos dimensiones.

En otros proyectos como el propuesto por Tumtong *et al.* [15] se busca darle otro enfoque al problema de la detección de jugadores, alejado de la IA y centrándose en técnicas de visión por computador. Sin embargo aunque el enfoque para la resolución del problema sea muy distinto a lo que estamos acostumbrados últimamente en pleno apogeo de la IA, el ámbito de este proyecto se ve muy acotado, ya que busca solo solventar el problema de la detección de jugadores.

Manteniendo enfoques diferentes, pero en este caso sin salirse de la IA, encontramos el proyecto propuesto por Sarkar *et al.* [12]. El proyecto propone una técnica basada en aprendizaje por refuerzo (RL) para detectar pases en el vídeo de un partido de fútbol, mediante la cual se determina las estadísticas de posesión del balón. Una secuencia de fotogramas se mapea a una secuencia de estados, como el balón bajo control del equipo A o del equipo B, o el balón no poseído por ninguno de los equipos. Este enfoque se diferencia del nuestro mayoritariamente en las técnicas de IA empleadas para hacer las detecciones y en la manera de calcular la posesión de balón.

El proyecto más reciente lo encontramos de mano de Somers *et al.* [14], donde nos muestran un proyecto similar al nuestro, donde a pesar de no obtener métricas de la posesión del balón, si se realizan seguimientos de jugadores, se detectan sus dorsales y se crea un minimapa. Además, presenta una novedosa métrica de evaluación de seguimientos multi-objeto para su caso particular.

Tras examinar el estado del arte en el análisis automatizado de partidos de fútbol, se observa una diversidad de enfoques y técnicas utilizadas para abordar este desafío. Proyectos como el de Naik *et al.* [9] destacan por su enfoque en la detección rápida y precisa de objetos clave en el campo, mientras que iniciativas como la de Descoins y Marvid [6] resaltan la importancia de calcular la posesión del balón mediante métodos más avanzados.

Asimismo, proyectos como el de Sarkar *et al.* [12] y el de Tumtong *et al.* [15] muestran diferentes aproximaciones al problema, desde el uso de agentes de IA basados en aprendizaje por refuerzo hasta técnicas basadas en visión por computador. Cada uno de estos proyectos contribuye al avance del campo, ya sea mediante el desarrollo de nuevas técnicas de detección, seguimiento o análisis estadístico.

En el contexto de este trabajo, se destaca la importancia de integrar múltiples técnicas de IA y visión por computador para proporcionar un análisis de los partidos de fútbol. Aunque nuestro enfoque se diferencia en algunos aspectos a los vistos anteriormente, como en la creación de un minimapa y en la extracción de estadísticas detalladas sobre la posesión.

## III. METODOLOGÍA

En esta sección, se describen los métodos que nos sirven para llegar a la obtención de las estadísticas de posesión de la Figura 1 a partir de una secuencia de imágenes que conforman un partido de fútbol. Para lograr este proceso, el primer paso consiste en identificar a los jugadores, la pelota y los árbitros en el campo. En nuestro proyecto, hemos abordado esta tarea utilizando una red neuronal YOLOv8, que nos permite realizar una detección eficiente y precisa de estos elementos clave en cada fotograma del vídeo.

Una vez que hemos identificado los objetos de interés, se obtienen las coordenadas espaciales de cada uno de ellos en el campo de juego. A partir de las líneas del terreno de juego se obtiene la posición de la cámara para poder representar el terreno de juego y los objetos de interés en un minimapa. A partir de las líneas del terreno de juego se obtiene la

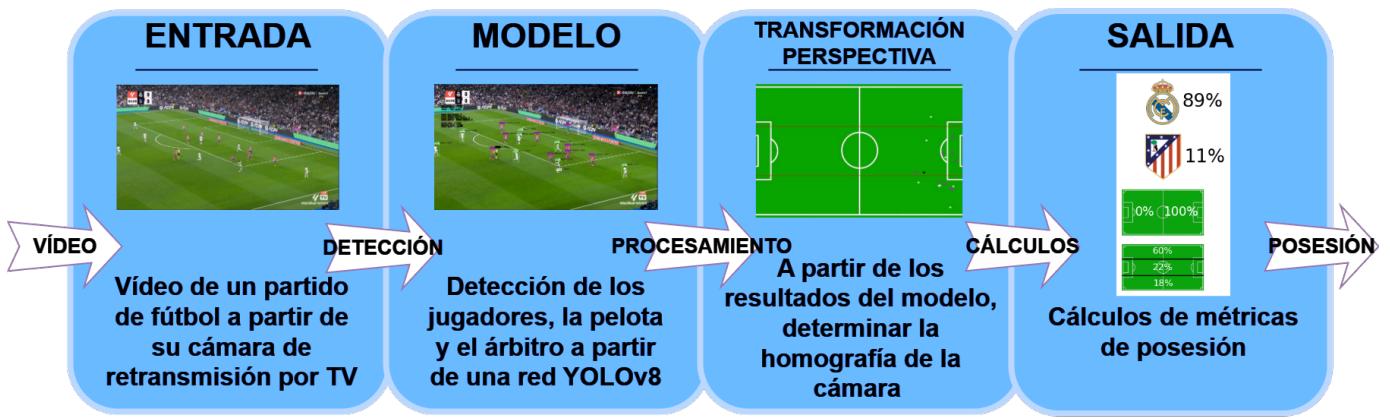


Figura 1. Esquema de las entradas y salidas del modelo, donde a partir de un vídeo de un partido de fútbol, obtenemos los resultados de posesión y el minimapa.

Model	size (pixels)	mAP <sub>val</sub> 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Figura 2. Características de los diferentes modelos de YOLO, ordenados en orden creciente de menor a mayor capacidad de detección.

homografía que nos permite relacionar las coordenadas imagen con coordenadas del minimapa, y así ubicarlos.

Con estos elementos se aplican algoritmos para determinar la ubicación del balón en el terreno de juego y la posesión. Estos algoritmos incluyen análisis de movimiento de la pelota y de los jugadores a lo largo del tiempo.

### III-A. YOLO

YOLO (You Only Look Once) es una arquitectura de detección de objetos que divide la imagen en una cuadrícula y realiza predicciones en cada una de estas regiones. Para cada celda de la cuadrícula, la red neuronal predice cuadros delimitadores, las probabilidades asociadas de que cada cuadro contenga un objeto y las probabilidades de dicho objeto. Estos cuadros se ponderan en función de las probabilidades predichas, permitiendo a YOLO detectar múltiples objetos en una imagen de manera eficiente y precisa en tiempo real[10, 19].

La decisión de utilizar YOLO sobre otros tipos de algoritmos o redes neuronales y más específicamente la selección de la v8 en lugar de otra versión, radica en las facilidades proporcionadas por ultralytics frente a otras alternativas similares, los creadores de YOLO, para el entrenamiento y el uso de la arquitectura YOLO en su versión 8 [16, 17].

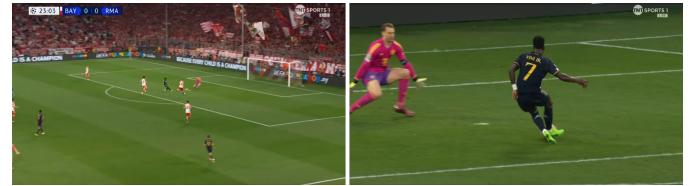


Figura 3. Ejemplo de tipo de cámara válido vs tipo de cámara no válido.

### III-B. Conjunto de datos de validación

Para validar el sistema se ha optado por crear una base de datos propia.

La base de datos está compuesta por secuencias de partidos de fútbol. Estas secuencias son extracciones de duración variable (entre 10 y 30 segundos) de partidos enteros, donde el punto de vista corresponde a la cámara principal de la retransmisión de un partido. Obviando repeticiones secuencias con repeticiones y cámaras a pie de campo o laterales más próximas al córner (ver Figura 3).

Disponemos de un total de 11 vídeos, todos de partidos diferentes. Los nombres de los vídeos se encuentran codificados de la siguiente manera: cada archivo de vídeo está nombrado utilizando las abreviaciones de los equipos participantes en mayúsculas, separadas por 'vs'. Por ejemplo, un partido entre Bayern de Múnich y Real Madrid se almacenaría como "BAYvsRMA.mp4". Esta nomenclatura se ha implementado para facilitar la identificación automatizada de los equipos involucrados e información adicional.

Adicionalmente a las secuencias de vídeos disponemos de información de los equipos: Nombre, abreviación, escudo y máscaras de color para las diferentes equipaciones, incluyendo a los porteros. Toda esta información adicional que se visualiza sobre los equipos se almacena localmente.

De esta manera podremos siempre saber qué máscaras aplicar para la detección de equipos y qué escudos utilizar para dar el resultado final de la posesión.

### III-C. Filtrado de detecciones

Al realizar la inferencia con YOLO sobre un fotograma de vídeo obtenemos diversos resultados, los cuales requieren de un filtrado previo para obtener el resultado deseado. La red YOLO únicamente clasifica los objetos en tres clases: jugador, árbitro y pelota. Es necesario determinar cada jugador a qué equipo pertenece y si se trata del portero.

Con el objetivo de eliminar duplicidades de la pelota, lo primero que realizamos es un filtrado de objetos clase ‘pelota’, quedándonos únicamente con el que ha obtenido mayor valor de confianza, suponiendo así, que será esa la pelota real entre las diversas detecciones posibles. Esto es debido a que la red puede detectar también balones que se encuentran fuera del campo, en el área de los recoge pelotas, y debido a que también puede detectar algunas botas blancas de jugadores como pelota.

Los objetos clase ‘jugador’ son etiquetados según el equipo al que pertenece (ver sección III-F), con esta información se determina la posesión del balón (ver sección III-I).

### III-D. Predicción de los movimientos de la pelota

En lo que respecta a la detección de la pelota, necesitamos explorar técnicas que nos permitan predecir la posición de la pelota para esos momentos donde el modelo de detección no detecte balón o lo haga con baja confianza.

La primera aproximación para resolver este problema consiste en aplicar un filtro de Kalman, un algoritmo de estimación que se utiliza para predecir el estado futuro de un sistema dinámico a partir de una serie de mediciones incompletas y ruidosas. Este filtro combina la información de las mediciones actuales con las predicciones del estado futuro del sistema, utilizando modelos probabilísticos tanto para el proceso dinámico subyacente como para las mediciones, proporcionando así una estimación del estado del sistema.

La segunda aproximación se trata de una distribución gaussiana multivariante. Con esta distribución, utilizando unos parámetros correctos, queremos modelar la proyección 2D del movimiento natural de una pelota. Para ello, la estimación en un fotograma  $t + 1$  vendrá dada por la colocación de una gaussiana en la posición de la pelota en el fotograma  $t$ . Este modelo es más preciso al disponer de una estimación probabilística en lugar de una estimación puntual.

Una distribución gaussiana multivariante se modela con los siguientes parámetros: media ( $\mu$ ) y la desviación típica ( $\sigma$ ). La media se ha modelado como un vector de dos componentes, correspondientes a la posición en la imagen, y la varianza es un único valor asumiendo misma varianza en ambos ejes. La media se modelará como la posición de la pelota en el fotograma anterior y la varianza será calculada a partir de las posiciones de la pelota en los 5 fotogramas anteriores. De esta manera, podemos modelar de forma correcta una distribución normal multivariante siempre que sea necesario para poder estimar las posiciones donde es más probable que se encuentre la pelota en el siguiente fotograma.

### III-E. Posición de la pelota

Como hemos discutido anteriormente, el modelo de detección no siempre nos proporciona una detección de la pelota y,

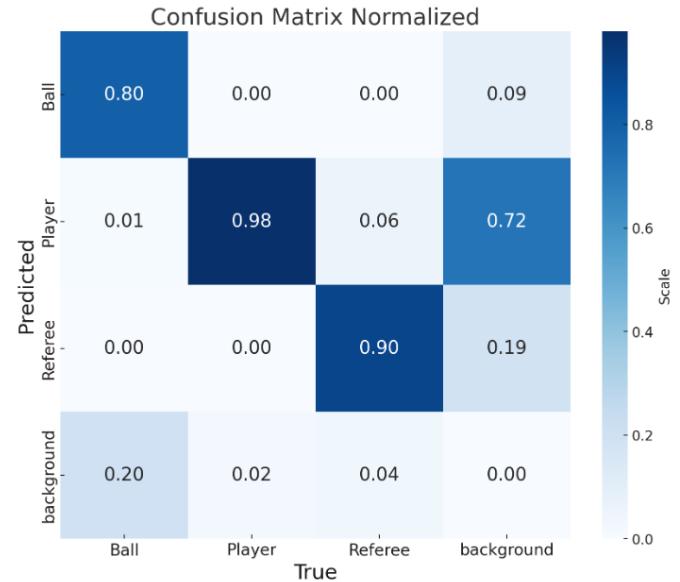


Figura 4. Matriz de confusión normalizada del entrenamiento de la arquitectura YOLO utilizada finalmente en el proyecto.

cuando nos la proporciona, no siempre es correcta, por lo cual hemos diseñado un método para predecir los movimientos del balón. Para decidir la fuente de la que seleccionar la posición del esférico (es decir, decidir si la posición actual viene determinada por la detección de la YOLO o por la estimación de la distribución normal), hemos diseñado un modelo basado en probabilidades, del cuál elegiremos la alternativa con mayor puntuación.

Este modelo está formado por dos alternativas diferentes: la posición detectada por YOLO y la predicción realizada por la gaussiana multivariante. La alternativa YOLO viene puntuada por  $\max(C_Y, P_p)$  donde  $C_Y$  es la confianza devuelta por YOLO y  $P_p$  es la probabilidad ponderada por la distribución gaussiana. La ponderación se determina como  $P_p = C_Y \cdot P_G \cdot \theta_1$ , donde  $P_G$  es la probabilidad de la distribución gaussiana en la posición predicha por YOLO, y  $\theta_1$  es un factor de corrección. De esta forma la puntuación de esta alternativa es alta si el valor de confianza de YOLO es alto, o esta alineada con la estimación de la gaussiana multivariante.

La segunda alternativa es el punto más probable de la distribución gaussiana multivariante cuya puntuación es  $\theta_2$ , este valor es constante y sirve para descartar detecciones de YOLO poco confiables.

La posición de la pelota se determina como la alternativa de mayor puntuación.

Los valores de  $\theta_1$  y  $\theta_2$  se obtienen empíricamente. Los resultados experimentales han dado como resultado  $\theta_1 = 22$   $\theta_2 = 0.5$ . Se determinan tras un estudio donde se ejecutan todas las combinaciones posibles entre 10 y 40 para  $\theta_1$  y entre 0.2 y 0.6 para  $\theta_2$ . Este estudio se realiza sobre un conjunto de 5 secuencias las cuales fueron previamente etiquetadas para obtener un *ground truth* de las posiciones de la pelota. Para cada una de las combinaciones se determina el número de fotogramas bien etiquetados (fotogramas donde la pelota se

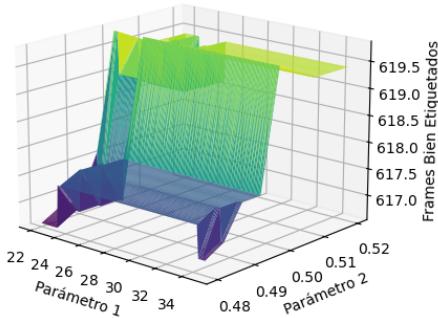


Figura 5. Resultados del estudio del valor de  $\theta_1$  y  $\theta_2$  (imagen con los valores acotados). El valor  $\theta_1$  se utiliza como factor de corrección en la probabilidad ponderada  $P_p$ , y el valor  $\theta_2$  es el valor umbral utilizado para descartar la predicción de YOLO.



Figura 6. Resultados de recortar el cuadro delimitador y aplicar dos máscaras diferentes.

detecta o se predice donde realmente se encuentra la pelota) y el número de fotogramas mal etiquetados (fotogramas donde la pelota se detecta o se predice donde no se encuentra la pelota). Se pueden ver los resultados en la Figura 5.

#### III-F. Distinción por equipos

YOLO detecta jugadores (ver Figura 4), sin embargo, esta detección es simplemente de jugador de fútbol, para poder clasificar cada jugador en un equipo diferente debemos aplicar técnicas adicionales de visión por computador.

Para cumplir este objetivo, decidimos utilizar máscaras de color, además, necesitamos también hacer uso de la base de datos.

Para cada objeto detectado por YOLO con la clase ‘jugador’ se debe determinar el equipo, para ello se analiza el color del cuadrado delimitador. A partir del cuadrado delimitador se usa la zona central del mismo (ver Figura 6). Esto reduce el área de análisis, focalizando la zona central del jugador, facilita el procesamiento y mejorando los resultados. El análisis se realiza en el espacio color HSV aplicando máscaras de color.

A través de una operación a nivel de bits (*AND*), se aplican las máscaras a la imagen original para aislar las partes que

coinciden con los colores de los equipos y porteros (ver Figura 6). Se calcula la suma de píxeles no cero en cada una de las áreas cubiertas por las máscaras, lo que proporciona una medida de la cantidad de colores de cada equipo presentes en la imagen. El equipo con la mayor cantidad de píxeles no cero se considera el equipo al que pertenece el jugador.

En caso de que la cantidad de píxeles no sea suficiente para clasificar al jugador en alguno de los equipos, se le clasifica como árbitro. Dado que puede ocurrir que YOLO detecte un árbitro como jugador.

#### III-G. Generación del minimapa

Para poder generar el minimapa a partir de la imagen obtenida de la retransmisión, necesitamos una homografía ya que esto nos permitirá mapear los puntos de la imagen capturada por la cámara (proyección 2D del mundo real) a la imagen del minimapa.

Para calcular la homografía, seguimos un proceso que consta de varias etapas [4, 5, 7]:

Primero, generamos imágenes de líneas del campo utilizando una red two-GAN (red generativa adversativa) compuesta por dos redes GANs concatenadas. *Pix2pix* es un tipo de GAN diseñada para la traducción de imágenes de un dominio a otro. Siguiendo la referencia [5], al concatenar dos *pix2pix* GANs, logramos, partiendo de una imagen normal del partido de fútbol, obtener una imagen donde todo está en negro excepto los bordes del campo (ver Figura 7).

A continuación, utilizando una red siamesa, buscamos la imagen más similar en la base de datos, lo que nos permite encontrar una correspondencia entre la imagen actual y las imágenes almacenadas.

Finalmente, una vez que encontramos la imagen más similar en la base de datos, para cada imagen de la base de datos se tiene asociada la homografía con la que se ha obtenido. Así, tenemos la homografía a utilizar. Aplicamos dicha homografía a la imagen para transformarla y obtener una vista en planta de la escena (ver Figura 9). Esta transformación nos permite realizar análisis adicionales y extraer información sobre la distribución de los objetos en la escena, como la posición de los jugadores y la pelota en el campo de juego.

Sin embargo, esta aproximación no es perfecta y comete fallos. De cara a mitigar al máximo posible estos fallos (y de reducir la carga computacional) se realiza el cálculo de la homografía cada 5 fotogramas, ya que la cámara en espacios de tiempo tan cortos no se mueve, o lo hace de forma muy sutil y el cómputo de la homografía es costoso. Además se complementa con la implementación de un sistema basado en el cálculo de la diferencia entre dos homografías diferentes. De esta manera, antes de hacer el cambio de una a otra se comprobará la diferencia entre ambas, si esta diferencia es demasiado grande (igual o superior a 0.2), mantendremos la homografía anterior (ya que se da por hecho que en 5 fotogramas no ha dado tiempo a haber cambios drásticos en la perspectiva de la cámara). En caso de encadenar demasiadas homografías dadas por “erróneas”, se supondrá que el error habrá estado en nuestras suposiciones y se utilizará la nueva.

Además, para mejorar el rendimiento del modelo, se implementa una técnica adicional. Esta técnica consiste en aplicar

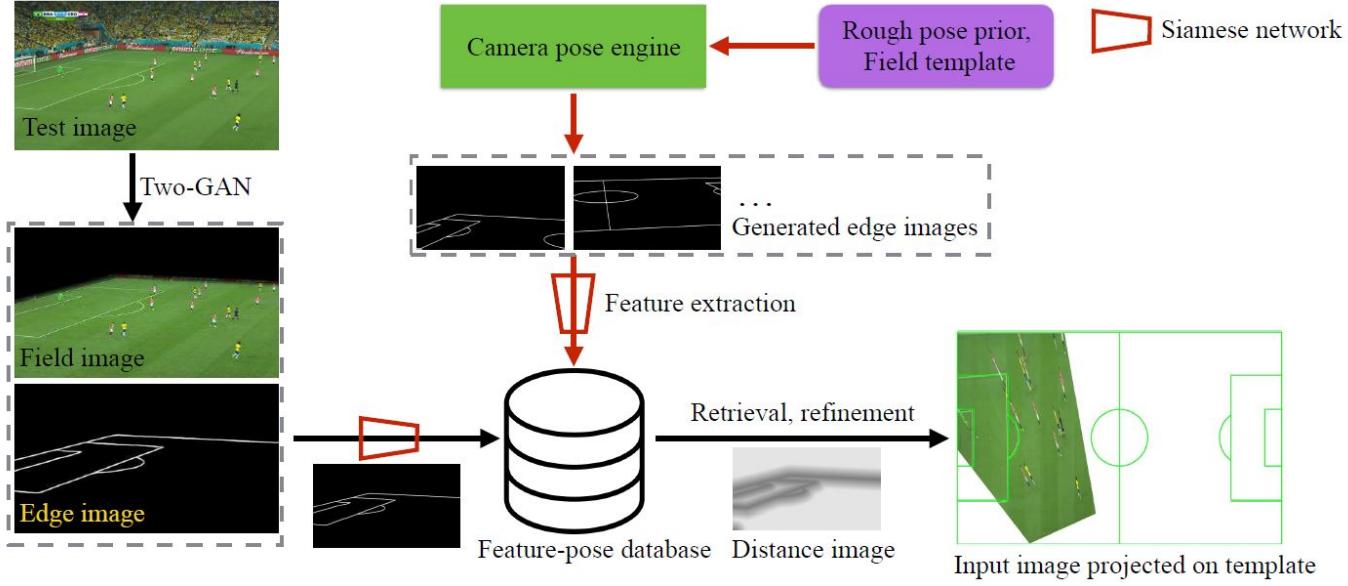


Figura 7. Esquema del proceso de creación del minimapa. [4]



Figura 8. En rojo la línea detectada por la transformada de Hough, en verde la línea extendida

filtros para eliminar el ruido en la imagen generada por la segunda GAN, la cual produce una imagen binaria que contiene únicamente las líneas del campo. Este proceso ayuda a eliminar posibles errores al comparar esta imagen con la base de datos en la búsqueda de la imagen más similar.

### III-H. Detección del lado del campo

Para determinar en qué mitad del campo se encuentra el balón, empleamos técnicas de visión por computador para detectar la línea central del campo. Estas técnicas incluyen el uso del algoritmo de Canny, operaciones de dilatación y la transformada de Hough.

En primer lugar, aplicamos máscaras de color para segmentar el terreno de juego, eliminando así elementos como las gradas y los jugadores. A continuación, utilizamos el algoritmo de Canny para identificar las líneas del campo. La imagen resultante se somete a una dilatación para mejorar la

detección de estas líneas, combinando aquellas que puedan ser parte de la misma pero que inicialmente se detectan como separadas. Finalmente, aplicamos la versión probabilística de la transformada de Hough.

Las líneas detectadas se filtran para conservar únicamente aquellas cuyos ángulos se encuentran en el rango de 85 a 95 grados, ya que la línea central del campo aparece completamente vertical o con una inclinación mínima. Si existen múltiples líneas que cumplen con estas características, seleccionamos la generada por el segmento más largo.

Una vez identificada la línea central, la extendemos para ocupar toda la imagen, calculando la pendiente y el ángulo de los extremos de la línea obtenida mediante la transformada de Hough (ver Figura 8).

Este enfoque es eficaz en la mayoría de los casos, excepto cuando el vídeo comienza enfocando una de las áreas del campo, donde la línea central no es visible. En tales situaciones, utilizamos un método auxiliar basado en la posición de la pelota en el minimapa, obtenida mediante la transformación de perspectiva. No utilizamos este método auxiliar como principal debido a que la proyección de la perspectiva sobre el minimapa puede contener errores significativos, especialmente cuando el balón está cerca de la línea central. Por lo tanto, mantenemos este método auxiliar como respaldo, utilizándolo únicamente cuando el principal, basado en visión por computador, no es aplicable y que coincide con el momento en que este resulta funcional.

### III-I. Cálculo de la posesión

Para llevar a cabo el cálculo de la posesión existen varias cuestiones a tener en cuenta, entre ellas cuándo consideramos que un equipo tiene la posesión y cuándo no.

Siguiendo las pautas utilizadas en el último Mundial disputado (Qatar 2022) [8] nos encontramos con que existen 4

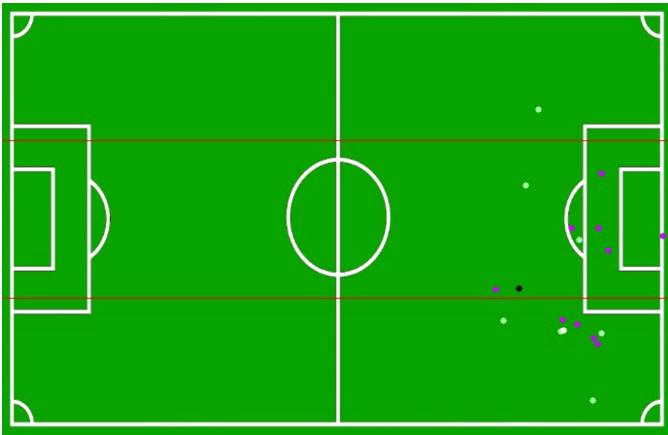


Figura 9. Ejemplo de minimapa y de división del campo en tercios.

posibles estados para determinar qué equipo tiene la posesión de balón (o si no la tiene nadie). Y estas son: el balón se encuentra en posesión del equipo A, el balón se encuentra en posesión del equipo B, el balón se encuentra en juego pero ningún equipo lo posee y, finalmente, el juego se encuentra interrumpido.

Siguiendo con estas pautas, podemos plantear el cálculo de posesión del balón por parte de un equipo X basándonos en el resultado obtenido de dividir el número de fotogramas que ese equipo ha tenido el balón entre el número de fotogramas en los que alguno de los dos equipos ha tenido la posesión.

Un equipo tendrá la posesión del balón si alguno de sus jugadores tiene el balón controlado en alguno de los dos pies. Para esto, debemos definir un valor umbral  $\theta_3$ , el cuál indicará la distancia a partir de la cual ya no consideramos que el balón se encuentra en posesión del jugador. Sin embargo, el valor de  $\theta_3$  debe ser dinámico y cambiar dependiendo del tercio del campo (dividiendo el campo en tercio en horizontal, ver Figura 9), ya que al encontrarse la cámara en perspectiva, un número dado de píxeles en la imagen no representan la misma distancia en el tercio más bajo del campo y en el tercio más alto del campo (siendo mayor distancia en un tercio superior que en uno inferior, producido por la perspectiva). Por tanto utilizaremos los conocimientos de la posición de la pelota sobre el minimapa creado con la homografía para determinar en qué tercio del campo nos encontramos y actualizar así de forma dinámica el  $\theta_3$ , reduciéndolo cuando nos encontramos en la zona más alta del campo y aumentándolo al encontrarnos en la zona más baja.

Una vez resuelto ese problema, basta con obtener las coordenadas inferior izquierda e inferior derecha de las *bounding boxes* de los jugadores (simulando así el pie izquierdo y el pie derecho) y realizar una búsqueda para encontrar el más cercano. Si el "pie" más cercano presenta una distancia igual o inferior a  $\theta_3$  consideraremos que ese jugador se encuentra en posesión del balón y, por tanto, el equipo al que pertenezca sumará un fotograma a su recuento de fotogramas con posesión del balón. En caso de que el balón no se encuentre en posesión de nadie no realizaremos ninguna acción.



Figura 10. Ejemplo de resultado final.

### III-J. Resultados finales

El resultado final de nuestro programa incluye una imagen con los escudos de los equipos que han participado en el clip analizado, junto a los porcentajes totales de posesión de cada uno de ellos, acompañados de dos imágenes del campo que nos muestran en qué mitad del campo se ha encontrado el balón durante más tiempo y en qué tercios (ver Figura 10).

Además de esto, si es deseado, se puede guardar también el clip con las anotaciones realizadas por YOLO y el minimapa

En esta sección, hemos detallado el proceso integral que nos lleva desde la captura de un partido de fútbol hasta la obtención de las estadísticas de posesión deseadas. Describiendo paso a paso el funcionamiento de cada eje dentro de la arquitectura del proyecto.

## IV. EXPERIMENTACIÓN

En el apartado de experimentación, nos enfocamos en la fase práctica de nuestro proyecto. Aquí, nuestro principal objetivo es poner a prueba la efectividad de nuestras técnicas y algoritmos.

Para ello, diseñamos experimentos específicos para evaluar el rendimiento de nuestro sistema en distintos escenarios.

### IV-A. Entrenamiento YOLO

El proceso de entrenamiento se lleva a cabo en un ordenador personal con las siguientes especificaciones: procesador Intel i5-7600k, GPU Nvidia GTX 1060 6GB y 16GB de RAM.

Durante este proceso, realizamos entrenamientos con modelos de YOLO de diferentes tamaños, incluyendo los tamaños 'S', 'M' y 'X' (ver Figura 2), evaluando cuál proporciona los mejores resultados en términos de precisión y velocidad de detección.

Después de diversos entrenamientos diferentes, seleccionamos el modelo que ofrece los mejores resultados en la detección de jugadores, árbitros y la pelota en el contexto de un partido de fútbol (ver Figura 4). Este enfoque nos permite obtener una red YOLOv8 finamente ajustada y optimizada para nuestra aplicación específica de análisis automatizado de partidos de fútbol.

Para entrenar nuestra red, utilizamos un modelo de tamaño S pre-entrenado sobre el conjunto de datos COCO (Common

Objects in Context) [1]. Este pre-entrenamiento inicializa los pesos de la red con conocimientos previos sobre la detección de una amplia variedad de objetos en imágenes generales. El tamaño S hace referencia a uno de los distintos tamaños de modelo disponibles para la red (ver Figura 2), en este caso, es una versión más pequeña y ligera del modelo YOLO, diseñada para ofrecer un equilibrio entre velocidad y precisión. La selección de este tamaño se debe a que es particularmente adecuado para aplicaciones en tiempo real donde los recursos computacionales son limitados, ya que consume menos memoria y es más rápido de ejecutar en comparación con modelos de mayor tamaño.

Respecto a los datos de entrenamiento, nos basamos en un conjunto de datos compuesto por imágenes de partidos de fútbol [3], donde solo nos interesan tres clases principales: jugador, pelota y árbitro. Esto fue crucial para centrarnos en los elementos esenciales para el análisis de un partido de fútbol y simplificar la detección de objetos en el campo. Este conjunto de datos está formado por una colección de fotogramas específicamente recopilados de partidos de fútbol. Incluye etiquetas para los jugadores, árbitros y la pelota, lo que permite a YOLO identificar estos elementos clave en cada fotograma del video.

Uno de los desafíos que enfrentamos durante este proceso es el de obtener detecciones adecuadas de la pelota en las imágenes. Descubrimos que esto se debe en parte al tamaño de las imágenes de entrenamiento. Al reducir demasiado el tamaño de las imágenes, la pelota se volvía demasiado pequeña para que el modelo de detección pudiera realizar detecciones precisas. Los objetos deben tener al menos un tamaño de 15x15 píxeles para obtener resultados óptimos.

Para abordar este problema, aumentamos el tamaño de las imágenes de entrada a 1216x1216 píxeles. Este tamaño cumple con los requisitos mínimos de tamaño para la detección de la pelota y también cumple con la restricción de YOLO de que el tamaño de la imagen debe ser un múltiplo de 32. Esta solución nos permite obtener detecciones más precisas y consistentes de la pelota en las imágenes de los partidos de fútbol (ver Figura 4).

#### IV-B. Predicción de la posición de la pelota

Inicialmente, la idea de predecir los movimientos del balón surge con la única finalidad de tener una posición del balón, incluso si es predicha, para los momentos en que el modelo de detección no detecte ningún objeto como pelota.

Por tanto, compararemos ambos métodos propuestos sobre un conjunto de vídeos de prueba, para ver cuál de los dos consigue un mayor número de fotogramas bien etiquetados.

#### IV-C. Generación del minimapa

Para lograr una implementación efectiva del modelo utilizado para generar una homografía a partir de una imagen de cámara de retransmisión de un partido de fútbol [4], es necesario llevar a cabo una serie extensa de experimentos.

Para obtener los mejores resultados posibles, realizamos pruebas para determinar la frecuencia óptima con la que se calculaba la homografía. Dado el riesgo de errores en la zona del campo captada por la cámara y el alto costo computacional

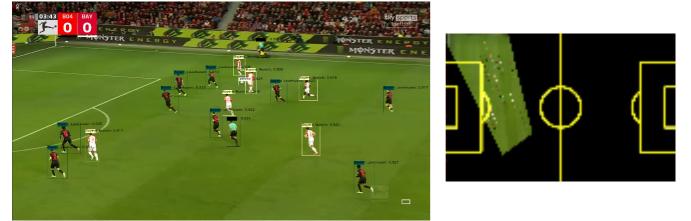


Figura 11. Ejemplo del uso de la homografía para deformar la imagen original, y su posterior transformación a su lugar correspondiente en el minimapa.

asociado, se estableció que el cálculo se llevaría a cabo cada 5 fotogramas de vídeo. Esta elección permite realizar los cálculos con la frecuencia adecuada (aproximadamente cada  $\frac{1}{5}$  de segundo) para evitar movimientos bruscos de la cámara, al mismo tiempo que se reduce el riesgo de errores graves.

Por otro lado, la selección del error máximo permitido entre las homografías para validar la nueva recién calculada se lleva a cabo de forma empírica. Dado que no se dispone de un *ground truth* para verificar la veracidad de las homografías, utilizamos un conjunto de 5 vídeos donde se calcula la homografía en cada fotograma y se compara con la anterior.

#### IV-D. Posesión

Uno de nuestros objetivos principales es evaluar la precisión de nuestro cálculo de posesión del balón. Para lograrlo, hemos definido diversos conjuntos de pruebas, cada uno compuesto por cinco vídeos. Aunque no todos los conjuntos contienen exactamente los mismos vídeos, cada conjunto está diseñado para proporcionar una muestra representativa de diferentes situaciones y escenarios en partidos de fútbol.

Para llevar a cabo esta evaluación, hemos generado un *ground truth* para cada vídeo de prueba. Esto se obtuvo etiquetando manualmente cada fotograma de los vídeos, determinando a quién pertenece la posesión del balón en cada momento. Este proceso de etiquetado manual garantiza que tenemos una referencia precisa y confiable contra la cual comparar los resultados generados por nuestro sistema automatizado.

Al comparar los resultados de nuestro cálculo de posesión con el *ground truth*, podemos determinar la precisión y fiabilidad de nuestro sistema en distintas situaciones.

## V. RESULTADOS

#### V-A. Tamaño YOLO

Después de la evaluación, determinamos que el tamaño de modelo ‘S’ se adaptaba mejor a nuestras necesidades. Este tamaño ofrecía el mejor rendimiento en términos de detección precisa de los objetos de interés, al tiempo que mantenía una carga computacional manejable. Como se puede ver en la Figura 4, el tamaño ‘S’ detectó la pelota de forma correcta en un 80 % de los casos. Sin embargo los modelos de tamaño ‘M’ y ‘X’ lo hicieron en un 40 % y 38 % respectivamente.

Cuadro I

COMPARATIVA DEL TOTAL DE FOTOGRAMAS BIEN ETIQUETADOS.

Nombre del clip	YOLO	YOLO Y PREDICCIÓN
ASMvsMCY	348	357
BMUvsBLV	328	364
BVBvsPSG	833	928
LVPvsCHE	916	1015
RMAvsSEV	405	435



Figura 12. Ejemplos de mala clasificación como árbitro y de occlusión.

#### V-B. Predicción de la posición de la pelota

Al comparar el número de fotogramas correctamente etiquetados por el filtro de Kalman con los del modelo que utilizamos finalmente, podemos observar que los resultados son significativamente mejores con nuestro modelo actual. Esta comparación nos permite concluir que el uso del filtro de Kalman es menos efectivo, lo que nos lleva a descartar su implementación en favor de nuestro enfoque basado en distribuciones normales.

#### V-C. Detección

Algo imprescindible de cara a poder realizar cálculos fieles de la posesión de balón y un buen cálculo de la posición de los jugadores y el balón sobre un minimapa es conseguir una buena detección de todos los elementos.

Por lo que respecta a los resultados finales de detección, podemos observar que disponemos de unos resultados bastante precisos a la hora de la correcta detección de los jugadores, árbitros y pelota. Asimismo la clasificación de los jugadores a sus respectivos equipos mediante la técnica de máscaras de color, también presenta resultados satisfactorios.

Sin embargo, satisfactorios no significan perfectos y en algunos momentos se pueden producir errores. El error más común ocurre cuando dos jugadores se superponen uno por delante del otro. Este hecho puede propiciar dos fallos. El primero consiste en YOLO detectándolos como un solo jugador debido a la superposición de ambos. El segundo se produce cuando ambos han sido correctamente identificados como jugadores independientes, pero al realizar los cálculos de la máscara de color, al haber uno por delante del otro, se produce una clasificación de equipo errónea (ver Figura 12). Algo similar ocurre si por algún tipo de cambio de iluminación o casuística similar, un jugador no cumple con el *threshold* mínimo de píxeles cuyo valor no sea cero al aplicar las máscaras, por tanto un jugador de campo acaba siendo identificado como árbitro (ver Figura 12).

Cuadro II

COMPARATIVA DEL TOTAL DE FOTOGRAMAS MAL ETIQUETADOS.

Nombre del clip	YOLO	YOLO Y PREDICCIÓN
ASMvsMCY	1	7
BMUvsBLV	1	86
BVBvsPSG	17	115
LVPvsCHE	5	65
RMAvsSEV	1	15

Cuadro III

RESULTADOS DE POSESIÓN POR EQUIPO.

Nombre del clip	Real	Predicción
ASMvsMCY	94 % vs 6 %	94 % vs 6 %
BMUvsBLV	84 % vs 16 %	85 % vs 15 %
LVPvsCHE	85 % vs 15 %	78 % vs 22 %
RMAvsATM	100 % vs 0 %	89 % vs 11 %
RMAvsSEV	100 % vs 0 %	100 % vs 0 %

En relación con la correcta detección de la pelota, podemos observar los datos del cuadro I donde vemos que al incluir la predicción de la posición de la pelota aumentamos, de media, 54 fotogramas bien etiquetados por vídeo. Por otro lado, podemos ver el error introducido a raíz de predecir la posición de la pelota en el cuadro II. Como era esperable, al introducir una predicción, se acierta pero también se erra, añadiendo de media 52 fotogramas mal etiquetados por vídeo.

Cabe destacar, que el cálculo para decidir si la predicción de la pelota es correcta o no se realiza de forma automatizada con un *threshold* bastante estricto y aún así obtenemos resultados favorables.

#### V-D. Posesión por equipos

Los resultados obtenidos por nuestro programa, comparados con los datos reales de posesión de balón por equipo, se pueden observar en el cuadro III.

De nuestro análisis, se desprende un error medio del 6 % en el cálculo de la posesión de balón por equipo, un margen de error bastante satisfactorio, considerando la complejidad inherente a la automatización de este tipo de cálculos mediante técnicas de visión por computador.

#### V-E. Posesión por zona del campo

Los resultados predichos por nuestro programa de posesión por zona del campo se presentan en el cuadro IV. Estos resultados muestran la distribución de la posesión en diferentes áreas del campo (izquierda, derecha, arriba, en medio y abajo) para cada uno de las secuencias de vídeo analizadas.

Para validar la precisión de estos resultados, los comparamos con los resultados reales de posesión por zona del campo, obtenidos a partir del etiquetado manual. Esto puede observarse en el cuadro V.

Al comparar los resultados predichos con los reales, podemos observar la capacidad de nuestro sistema para identificar la posesión del balón en diferentes zonas del campo. Aunque existen algunas diferencias entre las predicciones y los datos reales, en general, nuestro sistema ha demostrado ser bastante preciso en la mayoría de los casos. Esto es particularmente evidente en las zonas de posesión ‘izquierda’ y ‘derecha’, donde

Cuadro IV  
RESULTADOS PREDICHOS DE POSESIÓN POR ZONA DEL CAMPO.

Nombre del clip	Izquierda	Derecha	Arriba	En medio	Abajo
ASMvsMCY	33 %	67 %	0 %	21 %	79 %
BMUvsBLV	91 %	9 %	52 %	39 %	9 %
BVBvsPSG	56 %	44 %	58 %	35 %	7 %
LVPvsCHE	38 %	62 %	0 %	9 %	91 %
RMAvsSEV	36 %	64 %	12 %	37 %	51 %

Cuadro V  
RESULTADOS REALES DE POSESIÓN POR ZONA DEL CAMPO.

Nombre del clip	Izquierda	Derecha	Arriba	En medio	Abajo
ASMvsMCY	33 %	67 %	0 %	33 %	67 %
BMUvsBLV	100 %	0 %	26 %	74 %	0 %
BVBvsPSG	57 %	43 %	57 %	41 %	2 %
LVPvsCHE	40 %	60 %	0 %	15 %	85 %
RMAvsSEV	36 %	64 %	7 %	45 %	48 %

los porcentajes predichos y reales son bastante similares. Esto se debe a que determinar el lado del campo es algo muy exacto, ya que se hace a partir de la línea central del medio del campo, la cual es muy sencilla de detectar.

Sin embargo encontramos peores resultados a la hora de determinar el tercio del campo, ya que este cálculo depende de la transformación de perspectiva utilizada para crear el minimapa, y esto no es preciso.

## VI. MEJORAS

En este apartado, abordamos diversas estrategias y enfoques destinados a mejorar la precisión y eficiencia de nuestro sistema de detección y análisis de partidos de fútbol. A lo largo del desarrollo del proyecto, hemos identificado varias áreas donde se pueden implementar mejoras significativas para optimizar el rendimiento general y la exactitud de los resultados.

El primer aspecto en el cual podríamos aplicar una mejora es el seguimiento de objetos [18]. Hasta ahora, nos hemos centrado en la detección de jugadores, la pelota y los árbitros utilizando simplemente la inferencia de YOLO, sin embargo, la precisión y continuidad del análisis pueden incrementarse considerablemente al implementar algoritmos de seguimiento de objetos, como BoTSORT[2] o ByteTrack[20]. Estos algoritmos permitirían realizar un seguimiento continuo y preciso de los objetos detectados a lo largo del tiempo, mejorando la coherencia y exactitud de los datos recogidos, además de abrir la posibilidad de extraer nuevas métricas como podrían ser distancias recorridas por jugador o mapas de calor de jugadores.

Además de esto, se podrían implementar mejoras detección de la zona del campo en la que se está jugando. La identificación precisa de la zona es crucial para obtener una matriz de transformación precisa y, en consecuencia, una mejor transformación de perspectiva. Para conseguir la mejora, una idea bastante intuitiva sería aumentar la base de datos de correspondencias imagen-homografía. Al incrementar la cantidad y diversidad de ejemplos en la base de datos, el

sistema podría reconocer mejor la escena que se visualiza y ajustar mejor la homografía, reduciendo errores y mejorando la precisión general del modelo.

Otra área de mejora que añadiría riqueza al proyecto y que no se ha implementado por cuestiones de tiempo, es la detección de posesión de balón por equipo y por zona del campo. Actualmente, nuestro sistema puede calcular la posesión del balón por equipo o por zona, pero no ambas simultáneamente. Implementar una funcionalidad que combine ambas métricas proporcionaría un análisis más detallado y útil del partido. Esto permitiría, por ejemplo, determinar qué equipo tiene la posesión del balón en cada zona específica del campo, proporcionando datos tácticos más profundos y precisos.

## VII. CONCLUSIONES

El proyecto ha demostrado ser una herramienta útil para el análisis automatizado de partidos de fútbol, proporcionando métricas de posesión de balón y un minimapa interactivo que refleja la dinámica del juego. Todo esto ha sido posible a través de la implementación de técnicas avanzadas de visión por computador y aprendizaje profundo.

Uno de los principales logros del proyecto, si no el mayor, ha sido la creación de un sistema robusto para la detección de objetos utilizando YOLOv8, que ha mostrado una alta precisión en la identificación de los elementos clave del campo de juego. A pesar de los desafíos encontrados, como la dificultad en la detección de la pelota debido a su tamaño y las variaciones en las condiciones de iluminación y ángulos de cámara, hemos implementado soluciones efectivas, como el uso de imágenes de alta resolución y la predicción del balón para mejorar los resultados.

La obtención de métricas de posesión de balón no ha sido tan exitosa. Nuestros resultados indican un error promedio del 8 % en comparación con los datos reales, lo cual es bastante alto.

Sin embargo, el proyecto también ha revelado áreas de mejora. La implementación de algoritmos de seguimiento,

como BoTSORT o ByteTrack, podría mejorar la continuidad y precisión en el seguimiento de los objetos detectados. Además, aumentar la base de datos de correspondencias imagen-homografía mejoraría la precisión en la detección de la zona del campo, reduciendo el error en la estimación de la homografía y, por ende, la transformación de perspectiva.

En resumen, este proyecto ha servido para explorar el análisis automatizado de partidos de fútbol, logrando avances significativos en la detección de objetos, así como en la obtención de métricas de posesión de balón. Además, las mejoras propuestas apuntan a mejorar la precisión y utilidad del sistema.

El proyecto puede encontrarse en este repositorio.

## REFERENCIAS

- [1]
- [2] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking, 2022.
- [3] N. Chapre. Detect players dataset. <https://universe.roboflow.com/nikhil-chapre-xgndf/detect-players-dgxz0>, dec 2023. visited on 2024-05-14.
- [4] J. chen. Lood339/sccvds: Sports camera calibration via synthetic data.
- [5] J. Chen and J. J. Little. Sports camera calibration via synthetic data, 2018.
- [6] A. Descoins and D. Marvid. Automatically measuring soccer ball possession with ai and video analytics, Oct 2022.
- [7] A. Karimi and A. Kazerouni. Footballanalysis/footballanalysis.
- [8] A. Martín. Qué es la posesión en disputa, la nueva métrica implantada en el mundial de qatar 2022, Nov 2022.
- [9] B. T. Naik, M. F. Hashmi, Z. W. Geem, and N. D. Bokde. Deepplayer-track: Player and referee tracking with jersey color recognition in soccer. *IEEE Access*, 10:32494–32509, 2022.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection, 2016.
- [11] S. Sarkar, A. Chakrabarti, and D. P. Mukherjee. Generation of ball possession statistics in soccer using minimum-cost flow network. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2515–2523, 2019.
- [12] S. Sarkar, D. P. Mukherjee, and A. Chakrabarti. Reinforcement learning for pass detection and generation of possession statistics in soccer. *IEEE Transactions on Cognitive and Developmental Systems*, 15(2):914–924, 2023.
- [13] J. Solawetz and Francesco. What is yolov8? the ultimate guide. [2024], Apr 2024.
- [14] V. Somers, V. Joos, A. Cioppa, S. Giancola, S. A. Ghasemzadeh, F. Magera, B. Standaert, A. M. Mansourian, X. Zhou, S. Kasaei, B. Ghanem, A. Alahi, M. V. Droogenbroeck, and C. D. Vleeschouwer. Soccernet game state reconstruction: End-to-end athlete tracking and identification on a minimap, 2024.
- [15] P. Tumtong, P. Promvitittrakarn, P. Pattanaworapan, and T. Charoenpong. A method for football players detection on the soccer field by integrated image processing techniques. In *2023 Third International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*, pages 19–22, 2023.
- [16] Ultralytics. Predict, May 2024.
- [17] Ultralytics. Train, May 2024.
- [18] Ultralytics. Yolo tracking, May 2024.
- [19] R. Ultralytics. Home, Apr 2024.
- [20] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang. Bytetrack: Multi-object tracking by associating every detection box, 2022.