

# Sistema de bajo costo para la autoproducción y análisis de vídeos de baloncesto: un caso de estudio para campus universitarios

Low-cost edge computing system for basketball video auto-production and analysis: a study-case for university campuses

Julián Quiroga<sup>1</sup>, Henry Carrillo<sup>1</sup>, Pedro Atencio<sup>1</sup>, Luis Zapata<sup>1</sup>,  
 Edisson Maldonado<sup>1</sup>, Manuel García<sup>1</sup>

<sup>1</sup> GeniusSports, Colombia

julian.quiroga@geniussports.com , henry.carrillo@geniussports.com ,  
 pedro.atencio@geniussports.com , luis.zapata@geniussports.com ,  
 edisson.maldonado@geniussports.com , manuel.garcia@geniussports.com

**RESUMEN.** El deporte al interior de las instituciones educativas usualmente se limita tradicionalmente al aspecto formativo y competitivo amateur. Esto junto a factores de financiamiento hacen que no resulte usual la transmisión de los partidos, o la aplicación de tecnologías disruptivas como el análisis de juego a partir de los videos. En este aspecto, proponemos el diseño de un sistema de bajo costo para la realización de producciones automáticas y transmisiones de video de baloncesto en competencias aficionadas y campus universitarios, los cuales pueden encontrar restrictivos los costos de soluciones de este tipo orientadas a ligas profesionales. Este sistema estará compuesto de unidades de cómputo en el borde capaces de realizar la transmisión y análisis del video, así como de los subsistemas para el control y gestión de estas. Consideramos que la implementación de esta propuesta en campus universitarios ayudaría a potenciar el desarrollo de los equipos de baloncesto universitario, tanto en competitividad como en sentido de pertenencia por parte de su comunidad.

**ABSTRACT.** Sports within educational institutions are usually limited to the formative and amateur competitive aspect. This, along with funding factors, makes it uncommon for games to be broadcast or for disruptive technologies such as video game analysis to be applied. In this regard, we propose the design of a low-cost system for the automatic production and transmission of basketball videos in amateur leagues and university campuses, which may find the costs of professional league-oriented solutions restrictive. This system will be composed of edge computing units capable of transmitting and analyzing video, as well as subsystems for their control and management. We believe that the implementation of this proposal on university campuses would help to enhance the development of university basketball teams, both in terms of competitiveness and sense of belonging within their community.

**PALABRAS CLAVE:** Arquitectura de sistemas, Cómputo en el borde, Análisis de vídeo.

**KEYWORDS:** System architecture, Edge computing, Video analysis.

## 1. Introducción

La transmisión en vivo de eventos deportivos usualmente está supeditada al dominio de grandes torneos y/o campeonatos con un alto número de espectadores, dejando por fuera ligas aficionadas y campeonatos de equipos universitarios. Una de las causas son los altos costos que acarrea actualmente la producción de video de tipo televisión, que involucra entre otros operadores de cámara, directores de transmisión y cámaras de alto costo.

Por otra parte, este hecho hace que la adquisición de nuevas tecnologías para el apoyo técnico y táctico para los entrenadores de estos equipos se vea limitada. Usualmente los equipos graban sus partidos mediante teléfonos desde las gradas con el objetivo de realizar un análisis posterior del juego y obtener retroalimentación, pero la calidad no es la mejor y depende mucho de la pericia de camarógrafo aficionado.

Debido al dominio visual de ambos problemas, estos pueden ser solucionados mediante la implementación de un sistema de transmisión y análisis de videos utilizando diversos avances tecnológicos tales como la visión por computador, el aprendizaje de máquina, y la utilización de dispositivos móviles como sistemas de cómputo en el borde.

La implementación de un sistema como este requeriría el desarrollo tecnológico en tres grandes frentes: 1) la implementación de un sistema para el control y monitoreo de las unidades de transmisión y análisis, 2) la producción automática y transmisión del partido de baloncesto y, 3) la generación de datos de interés táctico para los entrenadores a partir del análisis del video.

La producción de video es el proceso de generar un video, a partir de la captura, edición y unificación desde diversas fuentes visuales. Una fuente, en este contexto, puede entenderse como cualquier elemento que brinde datos visuales del objeto o evento que se está registrando, por ejemplo, una cámara o un stream de video.

En la producción de video se reconocen tres etapas principales: preproducción, producción y postproducción (Wales, 2017). Desde una perspectiva general la preproducción se centra en aspectos de planeación y logística del proceso de producción, por ejemplo, la escritura de guiones y logística. En la etapa de producción, las tareas se centran en la grabación de las modalidades requeridas, principalmente video, sin embargo, el audio suele abordarse de manera separada utilizando equipos especializados. Finalmente, en la etapa de postproducción se abordan tareas de edición para generar un video final que cuente una historia para un público objetivo.

Estas etapas sufrirán variaciones de mayor o menor complejidad dependiendo del contexto de aplicación. Particularmente en el caso de la producción de videos de alta calidad de eventos deportivos en vivo, la producción requiere un personal muy hábil capaz de responder ante las situaciones altamente cambiantes del deporte, lo cual implica un alto costo monetario (Owens, 2021). En este punto, la automatización del proceso de producción se presenta como una alternativa tecnológica en casos en los cuales no se puede permitir grandes costos.

En este artículo, se limita el trabajo inicialmente a la autoproducción y análisis de videos de baloncesto por razones de acceso hasta el momento de la publicación de este trabajo, pero consideramos que este mismo sistema aplicará a muchos más deportes tales como el fútbol, el voleibol y futsal por mencionar algunos.

## 2. Revisión de literatura

### 2.1. Producción automática de videos

La producción automática de video de eventos en vivo tiene sus primeros antecedentes en los años 90, con artículos como "Intelligent studios: Using computer vision to control TV cameras" (Pinhanez & Bobick, 1995),



y su principal aplicación fue grabar programas en estudio, por ejemplo, noticieros (Tsuda et al., 2008) o clases universitarias (Yokoi & Fujiyoshi, 2005). Recientemente el foco de aplicación ha pasado a la autoproducción de eventos deportivos de nicho (Quiroga et al., 2020).

Una forma de realizar la autoproducción de evento deportivos de manera automática es usando las denominadas cámaras virtuales. En las comunidades de computación gráfica y procesamiento multimedia se refiere usualmente a las cámaras virtuales como video re-targeting (Rachavarapu et al., 2018), virtual videography (Heck et al., 2007) o re-cinematography (Gleicher & Liu, 2008). Los desarrollos de este tipo tienen en común que se utilizan uno o varios sistemas de captura de video de alta resolución, en montajes fijos, y con vistas sobrepuestas, los cuales cubren la totalidad del escenario de interés, y la producción automática consiste en seleccionar la correcta porción de imagen a transmitir.



Figura 1. Marco seleccionado mediante cámara virtual para un partido de baloncesto. Fuente: Elaboración propia.

En la Figura 1 se puede observar un ejemplo de producción automática de video mediante cámara virtual. Nótese que la imagen completa no hace parte del video final sino una porción de esta.

En el caso de las cámaras virtuales existen algunos problemas como:

- **Ubicación y calibración de las cámaras:** Se debe asegurar que la vista total compuesta por la unión de las vistas individuales de cada cámara cubra de forma completa el escenario de interés, sus actores y elementos.
- **Modelamiento del escenario:** Este proceso consiste en la obtención de una correspondencia espacial entre el escenario real y el plano de la imagen. Por ejemplo, en el contexto deportivo, se desea modelar la correspondencia entre el sistema de coordenadas de la cancha y su proyección en la imagen capturada.
- **Problema de control y de selección:** El movimiento de la cámara y la elección de la cámara que debe salir al aire, en el caso de las cámaras virtuales se transforma en el problema de identificar la porción de imagen que se debe mirar, y a su vez en qué operaciones de traslado y escalado se aplican sobre dicha porción para mantener dentro de la misma la escena más relevante en el instante de tiempo, y preservando un sentido de estética visual (Gadde & Karlapalem, 2011). Por otra parte, las transiciones entre encuadres deben ser suaves de tal manera que no perturben ni cansen al usuario del video.

## 2.2. Análisis de videos deportivos

Huang-Chia Shih en (Shih, 2017) realiza una de las revisiones más detalladas sobre análisis de contenido de videos en deportes, que se encuentra en la literatura. Según el autor en el análisis del contenido de videos deportivos, se tienen tres modalidades: texto, audio y video. Por ejemplo, eventos relevantes de un partido de fútbol, tales como goles e infracciones, pueden ser identificados mediante el análisis del audio (Tjondronegoro & Chen, 2010). Desde un nivel semántico, los trabajos sobre análisis de video deportivo pueden agruparse en 2 categorías: detección de objetos y detección de actividades.

## Detección de objetos

En el contexto de videos deportivos, el concepto de objeto abarca desde los jugadores en la cancha, elementos del juego como pelotas, raquetas o bates, hasta posturas de jugadores en casos como el béisbol (H.-T. Chen et al., 2011). Por otra parte, la detección no se limita a la individualidad de los objetos, por lo que, en algunas aplicaciones, las dinámicas de interacción entre objetos brindan más información sobre el estado o eventos del juego. Luo et al. (2003) proponen la utilización de redes Bayesianas dinámicas (DBN por sus siglas en inglés) para caracterizar la naturaleza espaciotemporal de la semántica entre objetos en escenarios deportivos.

Usualmente, la detección de objetos se acompaña de un proceso de tracking o seguimiento. Particularmente en el análisis de contenido deportivo, el seguimiento es un proceso crucial que permite observar actividades continuas en el campo. Un problema abierto en la literatura es el seguimiento de múltiples objetos como un todo, y existen competencias abiertas como “Multiple Object Tracking Challenge” (Milan et al., 2016) las cuales permiten una adecuada comparación de desempeño entre diversos algoritmos de las cuales una evaluación exhaustiva puede ser encontrada en (Leal-Taixé et al., 2017).

## Detección de actividades

Si bien la literatura sobre reconocimiento de actividades en video es amplia, pocos trabajos se especializan en el ámbito deportivo. Shih et al. en (Shih, 2017) realiza una taxonomía de reconocimiento de actividades en videos deportivos y las clasifica en 3 categorías: a) actividades individuales, b) entre objetos, y c) entre grupos. La categoría de actividades individuales requiere de la estimación precisa de posturas del cuerpo humano con sus articulaciones, así como del modelado temporal de dichas posturas (Kazemi et al., 2013). La segunda categoría requiere de la detección de objetos de interés como bates, raquetas y pelota, y el modelamiento de las acciones entre jugadores y objetos (Yao & Fei-Fei, 2010 ; Bangpeng Yao & Li Fei-Fei, 2012). La última categoría estaría principalmente relacionada con deportes de equipo donde las dinámicas entre los grupos de jugadores se relacionan con actividades específicas del juego, por ejemplo en un tiro de esquina o un penalti en el caso del fútbol (C.-Y. Chen et al., 2007; Chung-Yuan Chao et al., 2005).

## 2.3. Computación en el borde (edge computing)

También conocida como computación de frontera, es un paradigma de computación distribuida que plantea transferir parte del proceso cómputo de un sistema a los subsistemas que están más cerca del fenómeno, a diferencia de la forma tradicional en la cual estos últimos solo sirven como sensores y envían los datos para ser procesados por algún sistema de procesamiento central.

Existen diversas alternativas de computadores de bajo costo que pueden ser integrados con diversos sensores para construir subsistemas de cómputo de frontera, por ejemplo, Raspberry, Intel NUC, NVidia Jetson o Google Coral, por mencionar algunos de los más conocidos.

### 2.3.1. Detección objetos en el borde

En la actualidad, los dispositivos móviles están adquiriendo relevancia en aplicaciones relacionadas con la visión por computadora. Uno de los más interesantes desafíos es lograr utilizar los recursos computacionales del dispositivo móvil sin perder la calidad entregada por dichos algoritmos. Cabe resaltar que el estado del arte en este campo nos plantea el uso de redes neuronales profundas, las cuales muestran excelentes resultados en diferentes conferencias en visión por computadora, pero suponen un desafío en cuanto a su uso en dispositivos móviles. Algunas arquitecturas han sido diseñadas para ser altamente eficientes al nivel de poder ser ejecutadas en dispositivos móviles en aplicaciones industriales y comerciales.

MobileNet V1 y V2 (Sandler et al., 2018) utiliza la idea de convoluciones separables profundas, las cuales permiten optimizar la profundidad de la red. Adicional a esto, la versión 2 plantea el uso de estrategias de



cuello de botella lineales, las cuales permiten codificar eficazmente la información transmitida en la red. EfficientDet se basa en una red piramidal de características bidireccional ponderada (BiFPN), que permite una fusión de características multi-escala de manera fácil y rápida. En (Guo et al., 2022) los autores proponen utilizar precisión mixta y generación de redes dispersas, combinado con métodos de detección de libres de ancla (anchor-free) para mejorar el entrenamiento del modelo y la velocidad de inferencia. Tiny-YoLo es una versión ligera del conocido detector YOLOV4. La principal diferencia radica en la cantidad de capas y parámetros que utiliza la versión tiny. Utilizando esta configuración, los autores confirman una velocidad de 244 FPS en una máquina con una sola GPU, con un modelo cuya huella de memoria es inferior a 50MB.

## 2.4. Sistemas comerciales

Existen numerosas opciones comerciales de sistemas automáticos de producción y transmisión de video de eventos deportivos. En la Tabla 1 se presentan algunas de las opciones comerciales más conocidas.

| Producto                | Deportes                                                | Características relevantes                                                                                                           |
|-------------------------|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Pixellot                | Fútbol, baloncesto, hockey, fútbol americano, vóleybol. | Creación de imagen panorámica 8K; identificación del estado del juego; seguimiento de jugadores; creación de resúmenes informativos. |
| PlaySight Smart Tracker | Fútbol, baloncesto, hockey y tenis.                     | Creación de imagen panorámica 4K; Cámaras independientes; seguimiento de jugadores.                                                  |
| VEO                     | Fútbol.                                                 | Creación de imagen panorámica 4K, Seguimiento del balón.                                                                             |

Tabla 1. Sistemas comerciales de autoproducción de videos de deportes. Fuente: Elaboración propia.

Estas opciones comerciales usualmente presentan precios restrictivos para clubes de ligas aficionadas y equipos universitarios, máxime si se tienen varios escenarios deportivos. El costo promedio es alrededor de 2000 USD por dispositivo y pagos mensuales desde 100 USD.

## 3. Desarrollo

En esta sección se abordará el desarrollo del sistema de autoproducción y análisis de videos de baloncesto, la estrategia de autoproducción y la plataforma de control y operación remota. Debido a la complejidad del sistema completo, abordaremos la generalidad de cada aspecto, su papel en el sistema, y su conexión con otros sub-sistemas.

### 3.1. Producción automática y análisis de videos mediante cómputo en el borde

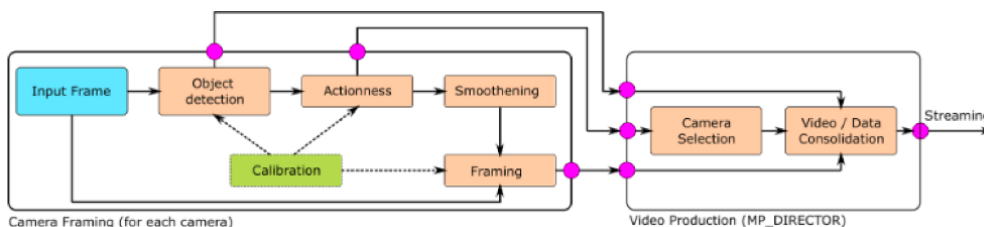


Figura 2. Estrategia propuesta para la autoproducción de partidos de baloncesto. Fuente: Elaboración propia.

La estrategia propuesta para la autoproducción y análisis de partidos de baloncesto está basada en la detección de jugadores y pelota, así como la detección de ciertos estados de juego. En la Figura 2 se puede observar un esquema con el flujo de datos de la estrategia propuesta.

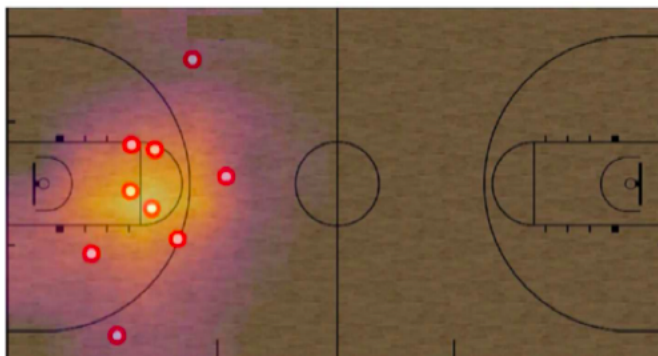


Figura 3. Distribución de los jugadores detectados en la cancha y mapa de calor del foco de interés. Fuente: Elaboración propia.

Primero, los jugadores y la pelota son detectados utilizando un modelo liviano de clasificación. Seguido se computa el foco de interés en la imagen a través de la distribución de los jugadores y la posición de la pelota dentro de la cancha (ver Figura 3). Este foco es suavizado a lo largo del video para evitar variaciones rápidas. Posteriormente se calcula una ventana de encuadre para la cámara virtual, tomando como centro el foco de interés, y finalmente se aplica una estrategia de control y selección de cámaras para el caso en que se tenga más de una unidad de producción en la cancha.

#### Estimación del encuadre



Figura 4. Ubicación del encuadre de la cámara virtual, basado en el foco de interés de la imagen. Fuente: Elaboración propia.

De manera general, la producción automática de video se realiza mediante un algoritmo de cámara virtual que controla el encuadre de la imagen capturada mediante operaciones simuladas de paneo y acercamiento de manera tal que dé al espectador la idea de que el video ha sido producido por una persona (ver Figura 4).

De manera específica, una vez ha sido obtenido el foco de interés en la imagen se activa el siguiente flujo:

- Se estima un encuadre objetivo  $F_O$  centrado en el foco de interés con una relación de aspecto de 16:9 y con un tamaño dependiente de una heurística basada en regiones de la cancha.
- A partir del encuadre actual  $F_C$  un nuevo encuadre  $F_t$  se estima como una transformación lineal entre  $F_C$  y  $F_O$ .

En otras palabras, el cómputo del encuadre de la cámara virtual puede ser entendido como:

$$F_t = (F_O - F_C) \cdot t + F_C \quad (1)$$

Donde  $t$  es un parámetro en el rango  $[0,1]$ . Nótese que, en la ecuación anterior, se generalizan tanto el tamaño como la posición del encuadre en una sola variable  $F$ .

## Heurística del encuadre

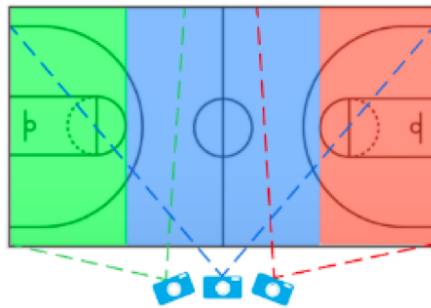


Figura 5. Regiones de la cancha para la heurística del encuadre de la cámara virtual. Fuente: Elaboración propia.

La cancha de baloncesto es dividida en tres regiones (centro, izquierda y derecha) en las que puede estar ubicado el foco de interés (ver Figura 5) y a partir de las cuales se define la siguiente heurística:

- Centro: Solo se permite paneo.
- Izquierda: Se permite paneo, acercamiento y alejamiento.
- Derecha: Se permite paneo, acercamiento y alejamiento.

Una vez el foco de interés para la imagen actual es ubicado dentro de una de las tres regiones anteriores utilizando la calibración de la cámara, se estima  $F_O$ , y un sistema de control se encarga de realizar la transformación  $F_C \rightarrow F_O$

## Controlador del encuadre

En la ecuación 1, el parámetro  $t$  debe ser controlado para ajustar la velocidad de transición de la transformación  $F_C \rightarrow F_O$ , lo cual en términos visuales se refiere a los siguientes movimientos de la cámara virtual:

- Paneo: Cuando la transformación  $F_C \rightarrow F_O$  aplica solo al centro del encuadre.
- Acercamiento: Cuando el tamaño de  $F_C$  es mayor que el de  $F_O$  por lo cual la transformación implica una ampliación de la imagen resultante.
- Alejamiento: Cuando el tamaño de  $F_C$  es menor que el de  $F_O$  por lo cual la transformación implica una reducción de la imagen resultante.

Para ello se implementó un controlador PID y se ajustaron sus parámetros para que las variaciones de  $t$  fueran lo suficientemente rápidas para seguir el juego, pero lo suficientemente lentas para no marear al espectador.

## Detección eficiente de entidades de juego

Cómo se mencionó anteriormente, la estrategia de autoproducción y análisis está basada en la detección eficiente de jugadores y pelota en videos en dispositivos de cómputo en el borde. A partir de ello se, se consideraron los siguientes requerimientos:

- La detección debe ejecutarse de manera eficiente dentro de un dispositivo móvil.
- La detección debe ejecutarse de manera eficiente utilizando la captura de video desde la cámara del dispositivo.
- Se deben detectar jugadores y pelota en la cancha.
- La detección debe ser robusta ante condiciones variables de iluminación y ubicación del sistema respecto a la cancha.

En este punto se decidió utilizar una aproximación basada en transferencia (transfer-learning) y ajuste (fine-tuning) sobre alguna arquitectura liviana de clasificación conocida. Para este propósito, se realizó una evaluación experimental de distintas arquitecturas livianas, entrenadas utilizando el dataset COCO (Lin et al., 2015), y considerando los requerimientos anteriores. Una vez fueron realizadas las iteraciones sobre las diferentes arquitecturas, fue seleccionada la que presentó un mejor balance entre tiempo de inferencia y número de parámetros. Finalmente, esta arquitectura fue ajustada y entrenada con los datos propios de para clasificar jugadores y pelota de baloncesto. En la sección de resultados se pueden consultar los resultados de la evaluación.

#### Unidad de cómputo en el borde

Al momento del desarrollo de este proyecto se definieron los siguientes requerimientos para la unidad:

- Debe permitir realizar transmisión de audio y video en vivo.
- Debe proveer calidad de imagen 720P a 30 FPS.
- Debe ser configurable de manera remota.
- Debe poderse monitorear de manera remota.
- Debe realizar producción automática y análisis de video.
- Debe poderse actualizar de versiones de manera remota.

Estos requerimientos conllevaron a utilizar un dispositivo móvil basado en Android como unidad de cómputo de frontera. Algunas de las razones para dicha elección fueron:

- Existen numerosas alternativas en el mercado de dispositivos de gama media / alta lo cual permite que el proyecto sea costo efectivo.
- Se encuentran alternativas con diferentes grados de protección a la intemperie.
- Se encuentran alternativas con alta calidad de captura de video, por ejemplo, grabación en 4K u 8K en algunos casos.
- Cuentan con interfaces de conectividad confiables.
- Se encuentran alternativas con alto poder de cómputo en CPU y GPU, capacidad de almacenamiento y memoria.
- El ecosistema de desarrollo Android cuenta con gran soporte por parte de la comunidad de visión por computador y aprendizaje de máquina.
- Se encuentran alternativas comerciales y libres de software para la gestión de dispositivos móviles (MDM por sus siglas en inglés).

#### Aplicación de Análisis de video y Transmisión

Para el dispositivo móvil fue desarrollada una aplicación de Android que permite la transmisión en tiempo real y baja latencia de autoproducciones de partidos de baloncesto, además de grabaciones locales de estos. La aplicación permite el procesamiento del video, la ejecución del modelo de detección, la generación de la autoproducción y su transmisión.

La aplicación no está pensada para la interacción con usuarios, sin embargo, se agregó una secuencia táctil que permite abrir la cámara principal (ver Figura 6), seguida de la cámara gran angular en caso de requerir visualizar lo que el teléfono está capturando al momento de realizar la instalación en el escenario deportivo.





Figura 6. Aplicación de autoproducción y análisis de video en el modo cámara principal. Fuente: Elaboración propia.

En la Figura 7 se muestra un diagrama de la interacción entre la aplicación desarrollada y el sistema de soporte (backend).

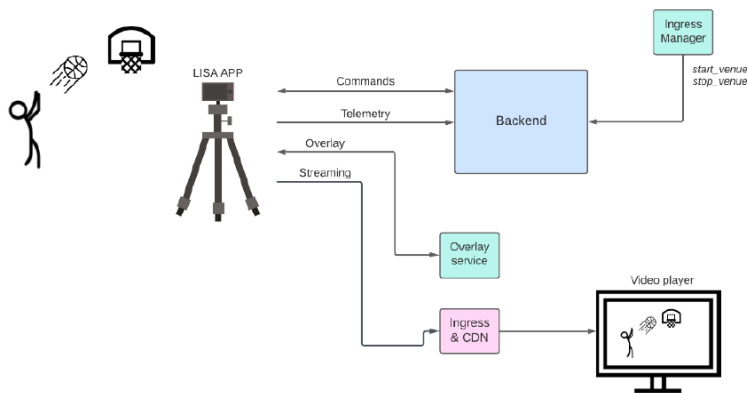


Figura 7. Diagrama general de la aplicación desarrollada y su interacción con el sistema de backend. Fuente: Elaboración propia.

En general la interacción sucede de la siguiente manera:

- i. Una vez la aplicación se ha abierto en el dispositivo móvil comienza a enviar datos de telemetría al backend de manera periódica, lo cual permite que el dispositivo pueda ser alcanzado y monitoreado.
- ii. Cuando se desea iniciar la autoproducción de un partido, el dispositivo recibe desde el backend un grupo de comandos de configuración e inicio del partido.
- iii. Por cada comando del grupo, el dispositivo ejecuta el comando interno respectivo y responde al backend con los resultados.
- iv. Una vez el dispositivo está configurado comienza la autoproducción y transmisión del video al servicio preferido.
- v. En caso de que se desee visualizar el puntaje del partido en el video, se puede consultar un servicio expuesto por la administración del coliseo.

### 3.2. Sistema para la gestión y el control de unidades

El backend está compuesto por una serie de subsistemas que permiten controlar, configurar y monitorear los dispositivos (ver Figura 8). Estos servicios cumplen tareas específicas enfocadas a la configuración y puesta en marcha de la autoproducción y análisis del partido de baloncesto.

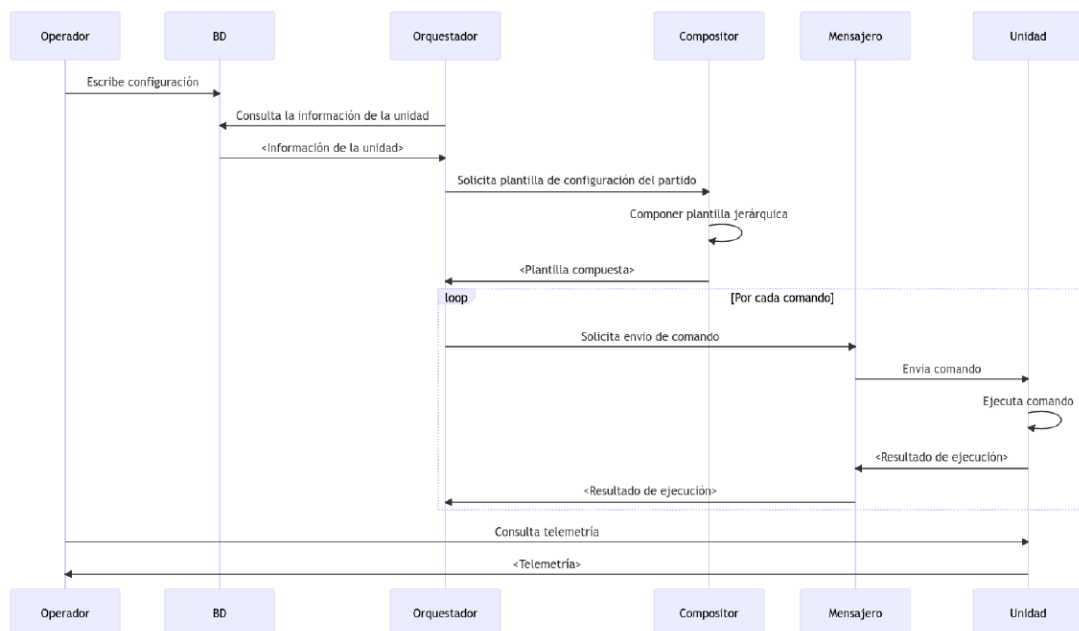


Figura 8. Diagrama de secuencia de las interacciones entre los subsistemas del sistema de autoproducción y análisis de videos. Fuente: Elaboración propia.

### Sistema de orquestación (orquestador)

Este sistema es el encargado de iniciar y detener la autoproducción y transmisión del video en el dispositivo, coordinando los subsistemas que participan. Para ello, el sistema cuenta con una interfaz de GraphQL que permite, entre otras, dos operaciones principales: `start_venue` y `stop_venue`.

La operación `start_venue` maneja toda la lógica requerida para comenzar la transmisión y la autoproducción para un coliseo, cancha o recinto deportivo (venue) en donde está instalada la unidad de autoproducción. La operación `stop_venue` realiza lo opuesto.

Durante una operación de `start_venue` o `stop_venue`, el sistema orquestador realiza lo siguiente:

- i. Recupera desde la base de datos la información de la unidad asociada al recinto deportivo.
- ii. Consulta la disponibilidad de la unidad a través de su telemetría.
- iii. Recupera la plantilla de configuración de autoproducción asociada específicamente a la unidad.
- iv. Genera la configuración final a través de una composición jerárquica de las plantillas.
- v. Compone los grupos de comandos que serán enviados a la unidad.
- vi. Solicita la ejecución de los comandos a través del subsistema de entrega de comandos.
- vii. Retorna los resultados de la ejecución de los comandos.

### Sistema de entrega de comandos (mensajero)

Este sistema es responsable enviar comandos a las unidades a través de un canal de Ably<sup>1</sup> en el cual cada unidad tiene un escucha. Este sistema cuenta con una interfaz de GraphQL o GraphQL IDE<sup>2</sup> que permite la interacción de usuario. Sin embargo, el principal uso de este sistema es a través de su interfaz GraphQL.

<sup>1</sup> <https://ably.com>

<sup>2</sup> <https://github.com/graphql/graphiql>



### Sistema de composición de configuración (compositor)

La principal función de este sistema es la de componer las plantillas de autoproducción para un partido en la siguiente jerarquía: deporte --> estadio --> partido. Esta configuración reside en unas plantillas preconfiguradas que determinan, por ejemplo, la resolución de captura, resolución de la transmisión, parámetros del detector, etc. La plantilla de configuración resultante contiene todos los parámetros y comandos que serán enviados a la unidad para iniciar o detener la transmisión de un partido. Estas plantillas son almacenadas en una base de datos PostgreSQL.

Este subsistema expone un editor web útil para dar soporte a un partido, por ejemplo, para cambiar parámetros de autoproducción.

### Sistema de configuración y monitorización de dispositivos (operador)

Este sistema constituye la principal herramienta operacional para el sistema de autoproducción. En este, se configuran y relacionan, los deportes, recintos deportivos y partidos. A través de una interfaz gráfica, el equipo de operaciones puede configurar qué unidades corresponden con cuál recinto deportivo, y que deporte se asocia a este, por mencionar algunos ejemplos.

Finalmente, a través de este sistema, también es posible visualizar la telemetría reportada por el teléfono en diagramas y hacer consultas sobre estos.

Si bien los 4 subsistemas anteriores componen el backend que soporta la autoproducción y las unidades, se requieran otras herramientas que soportan tareas alrededor, algunos de estas herramientas son:

- Herramienta de calibración: Se utiliza para calibrar la imagen de capturada, es decir, se encuentra la relación entre las coordenadas de los píxeles de la imagen y la cancha, de tal manera que es posible localizar en el espacio de la cancha a los jugadores y pelota detectados.
- Herramienta de anotación de videos: Esta herramienta se utiliza para anotar a mano jugadores y pelotas al interior de los videos grabados de partidos de baloncesto. Estos datos soportan el entrenamiento del modelo de detección y la evaluación de la calidad de la autoproducción.
- Mobile Device Management: Este sistema permite instalar y actualizar remotamente la aplicación de autoproducción en los teléfonos.

## 4. Resultados

A continuación, se presentan algunos resultados y evidencias del sistema propuesto de autoproducción y análisis de videos de baloncesto. Estos resultados se agrupan según los diferentes componentes del sistema.

### 4.1. Detección eficiente de entidades de juego

Para la comparación entre arquitecturas livianas clasificación fue utilizado el framework ncnn<sup>3</sup> de Tencent el cual es altamente optimizado para dispositivos móviles. Se realizaron mediciones de los tiempos de inferencia tanto en CPU como en GPU, y se probaron diferentes tamaños de imagen. El procedimiento para ejecutar la evaluación fue el siguiente:

- Todas las redes fueron pre-entrenadas con el dataset COCO.
- Por cada red tomó el tiempo promedio de inferencia para un grupo de 100 imágenes.
- Cada imagen del grupo contenía jugadores y pelota en diferentes deportes.
- Se realizaron iteraciones sobre el entorno de ejecución y el tamaño de la imagen.

En la Tabla 2 se pueden observar los resultados obtenidos.

<sup>3</sup> <https://github.com/Tencent/ncnn>

| Arquitectura      | Tamaño de la imagen | Entorno de ejecución | Número de parámetros | Tiempo promedio de detección |
|-------------------|---------------------|----------------------|----------------------|------------------------------|
| SSD-mobilenet     | 300 : 300           | CPU                  | 4.3 M                | 149 ms                       |
| SSD-mobilenet     | 300 : 300           | GPU                  | 4.3 M                | 54 ms                        |
| Yolo v5 s         | 640 : 640           | CPU                  | 7.3 M                | 658 ms                       |
| Yolo v5 s         | 640 : 640           | GPU                  | 7.3 M                | 174 ms                       |
| NanoDet-m         | 320 : 320           | CPU                  | 0.95M                | 25 ms                        |
| NanoDet-m         | 320 : 320           | GPU                  | 0.95M                | 77 ms                        |
| NanoDet-m         | 416 : 416           | CPU                  | 0.95M                | 37 ms                        |
| NanoDet-m         | 416 : 416           | GPU                  | 0.95M                | 103 ms                       |
| NanoDet-g         | 416 : 416           | CPU                  | 3.81 M               | 40 ms                        |
| NanoDet-g         | 416 : 416           | GPU                  | 3.81 M               | 140 ms                       |
| EfficientNet-Lite | 320 : 320           | CPU                  | 3.11M                | 35 ms                        |
| EfficientNet-Lite | 320 : 320           | GPU                  | 3.11M                | 98 ms                        |
| EfficientNet-Lite | 416 : 416           | CPU                  | 4.01M                | 72 ms                        |
| EfficientNet-Lite | 416 : 416           | GPU                  | 4.01M                | 166 ms                       |
| EfficientNet-Lite | 520 : 520           | CPU                  | 4.71 M               | 100 ms                       |

Tabla 2. Comparación experimental de distintas arquitecturas livianas de clasificación. Fuente: Elaboración propia.

A partir de los experimentos se seleccionó la arquitectura Nanodet<sup>4</sup> debido a su bajo número de parámetros, su bajo costo computacional, y debido a que obtuvo los mejores resultados para el entorno CPU y el segundo mejor resultado para GPU.

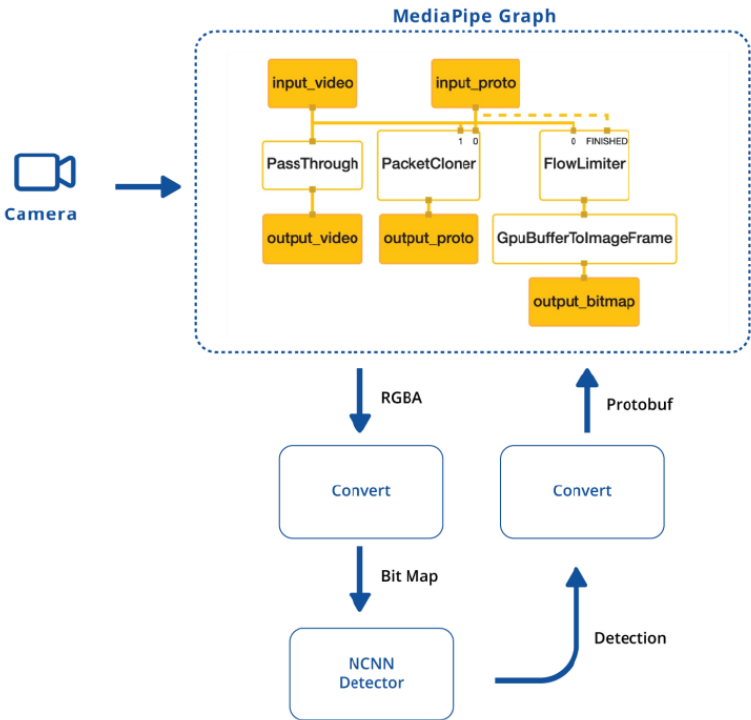


Figura 9. Diagrama de integración del modelo de clasificación en la aplicación de autoproducción. Fuente: Elaboración propia.

Una vez la arquitectura base fue seleccionada, se realizó un proceso de entrenamiento y ajuste para el problema de clasificación binario de jugadores y pelotas. El modelo entrenado es posteriormente enviado a un

<sup>4</sup> <https://github.com/RangiLyu/nanodet>



bucket de AWS-S3 desde el cual la aplicación lo descarga.

Al interior de la aplicación, el modelo es cargado utilizando el framework ncnn. Al momento del partido, se ejecuta un grafo de Mediapipe<sup>5</sup> el cual contiene la lógica de la autoproducción y desde este se capturan los fotogramas de la cámara del teléfono y se le envían al detector, el cual genera las inferencias y las devuelve al grafo continuar con el proceso de estimación del encuadre. En la Figura 9 se muestra un diagrama que ilustra esta interacción con el modelo de detección.

## 4.2. Subsistemas de soporte

A continuación, se presentan capturas de pantallas de los sub-sistemas de que soportan la autoproducción de los videos.

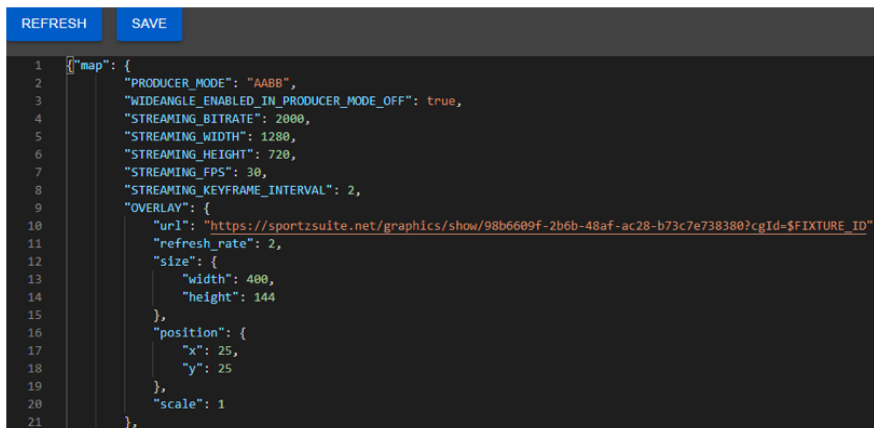


Figura 10. Captura de pantalla del editor del sistema compositor de configuraciones. Fuente: Elaboración propia.

Como se mencionó anteriormente, el sistema compositor de configuraciones cuenta con un editor que permite modificar valores de la autoproducción a mano. En la Figura 10 se puede observar una captura de pantalla de la plantilla de partido vista desde el editor del sistema compositor.

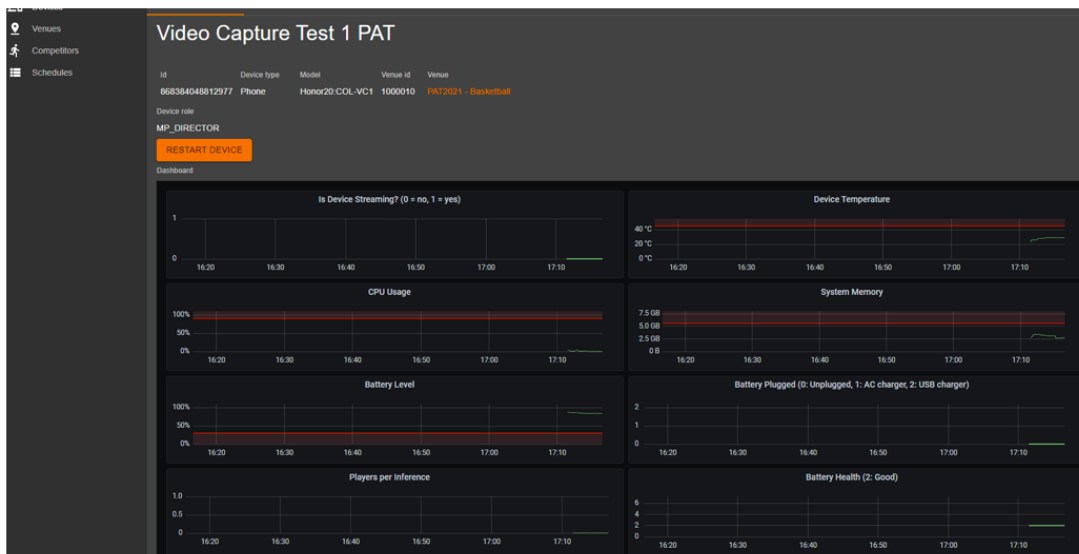


Figura 11. Captura de pantalla del sistema operador. Fuente: Elaboración propia.

<sup>5</sup> <https://mediapipe.dev>

La consulta de la telemetría que reportan las unidades de autoproducción es un aspecto vital a la hora de dar soporte a un sistema de esta naturaleza. Dentro del subsistema operador, se agregaron distintas visualizaciones utilizando paneles de Grafana<sup>6</sup> para varias métricas del teléfono tales como el estado de la carga de la batería, el estado de la transmisión, el número de inferencias por segundo del detector, y la temperatura del teléfono, entre otras. Un ejemplo de esto puede verse en la Figura 11.

### 4.3. Autoproducción y análisis de videos de partidos de baloncesto

Al inicio de un partido se captura remotamente una imagen de la cámara del teléfono instalado en el estadio. Con esta imagen se crea una tarea en la herramienta de calibración y se procede a referenciar la cancha. En la Figura 12 se muestra el resultado del proceso de calibración para un partido de prueba.



Figura 12. Calibración de la cancha de un partido de prueba de baloncesto. Fuente: Elaboración propia.

Nótese en la figura anterior, que resultado del proceso de calibración se estiman distintas cajas que son utilizadas en el proceso de autoproducción. Por ejemplo, en color amarillo se visualiza la caja que contiene la región que se será utilizada por el detector. Esto debido a que contiene principalmente la cancha y por ende no se satura el detector con información irrelevante. En verde claro y oscuro las cajas que definen las regiones izquierda y derecha para el sistema de control. En rojo, la caja que define el máximo alejamiento alcanzable para un encuadre de la autoproducción.

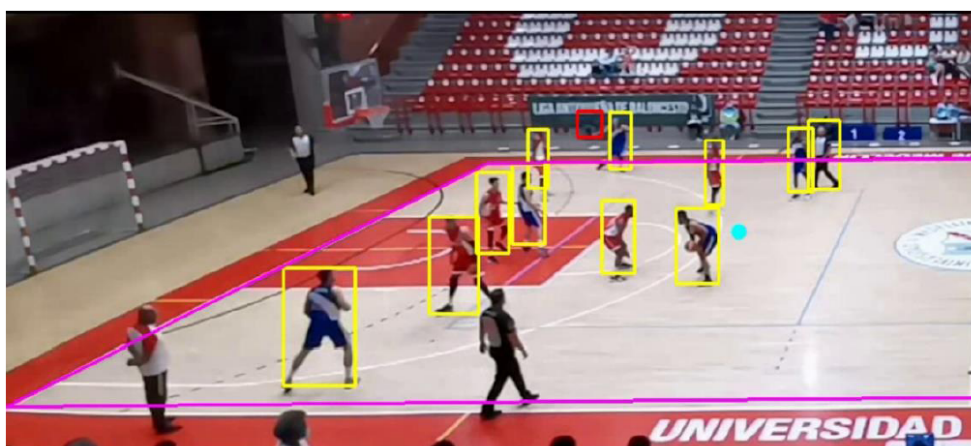


Figura 13. Detecciones en tiempo de juego desde el dispositivo móvil. Fuente: Elaboración propia.

<sup>6</sup> Grafana: The open observability platform | Grafana Labs --- <https://grafana.com/>

En la Figura 13 se visualizan las detecciones (amarillo) y el foco de interés (cian) estimados por el teléfono para un instante de tiempo de un partido de prueba. En rojo se encuentra una detección tipo falso-positivo para la pelota de baloncesto. Si bien las detecciones pueden presentar confusiones para ambas categorías (jugadores y pelota), la aproximación basada en el centroide de las distribuciones de las entidades en la cancha permite robustez ante datos atípicos.



Figura 14. Ejemplo de autoproducción transmitida (arriba) y autoproducción modo depuración. Fuente: Elaboración propia.

Al momento de transmitir la autoproducción, no se renderiza ningún elemento asociado a las detecciones. Sin embargo, la aplicación permite configurarse remotamente para visualizar las detecciones y el foco de interés de tal forma que un equipo de desarrollo pueda depurar el sistema (ver Figura 14).

## 5. Conclusión

El sistema de producción y transmisión automática de videos de baloncesto en ligas menores y campus universitarios propuesto, tiene el potencial de revolucionar la forma en que se transmiten y ven los deportes en las instituciones educativas. Al aprovechar el paradigma de computación de frontera, este sistema puede proporcionar una transmisión agradable de un partido de baloncesto mediante una cámara virtual y, por otra parte, las detecciones de jugadores y pelota y su localización en las coordenadas de la cancha es información que puede ser útil para los entrenadores y jugadores a la hora diseñar estrategias de juego y revisar su rendimiento. La adopción de este sistema al interior de las instituciones puede contribuir en fomentar un sentido de comunidad alrededor de su equipo universitario de baloncesto inicialmente, sin embargo, el sistema también puede ser extendido a otros deportes. Si bien lo abordado en este documento cubre solo la generalidad del sistema consideramos que incluye lo esencial para que un sistema de este tipo pueda ser replicado, implementado y ajustado a los requerimientos específicos de las instituciones que puedan interesarse. Finalmente, aspectos de infraestructura tecnológica tales como la seguridad de la información, redundancia de los datos, etc., quedaron por fuera de este documento y sin embargo deben ser abordados al momento de una implementación concreta. Esperamos que esta propuesta inspire más investigación y desarrollo en el campo de la tecnología deportiva y, en última instancia, conduzca a un enfoque más inclusivo

y basado en datos para la educación deportiva.

## Financiación

Este es el resultado de un proyecto de investigación financiado en parte por el gobierno Colombiano a través de Minciencias mediante la convocatoria de beneficios tributarios 2022 para empresas Colombianas.

### Cómo citar este artículo / How to cite this paper

Quiroga, J.; Carrillo, H.; Atencio, P.; Zapata, L.; Maldonado, E.; García, M. (2024). Sistema de bajo costo para la autoproducción y análisis de vídeos de baloncesto: un caso de estudio para campus universitarios. *Campus Virtuales*, 13(2), 89-105. <https://doi.org/10.54988/cv.2024.2.1383>

## Referencias

- Bangpeng, Y.; Li, F.-F. (2012). Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9), 1691-1703. doi:10.1109/TPAMI.2012.67.
- Chen, C.-Y.; Wang, J.-C.; Wang, J.-F.; Hu, Y.-H. (2007). Event-Based Segmentation of Sports Video Using Motion Entropy. In Ninth IEEE International Symposium on Multimedia (ISM 2007) (pp. 107-111). doi:10.1109/ISM.2007.4412363.
- Chen, H.-T.; Chou, C.-L.; Tsai, W.-J.; Lee, S.-Y.; Yu, J.-Y. (2011). Extraction and representation of human body for pitching style recognition in broadcast baseball video. In 2011 IEEE International Conference on Multimedia and Expo (pp. 1-4). doi:10.1109/ICME.2011.6011840.
- Chao, C.-Y.; Shih, H.-C.; Huang, C.-L. (2005). Semantics-Based Highlight Extraction of Soccer Program Using DBN. In Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing (pp. 1057-1060). doi:10.1109/ICASSP.2005.1415590.
- Gadde, R.; Karlapalem, K. (2011). Aesthetic Guideline Driven Photography by Robots. doi:10.5591/978-1-57735-516-8/IJCAI11-344.
- Gleicher, M. L.; Liu, F. (2008). Re-cinematography. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 5(1), 1-28. doi:10.1145/1404880.1404882.
- Guo, J.-M.; Yang, J.-S.; Seshathiri, S.; Wu, H.-W. (2022). A Light-Weight CNN for Object Detection with Sparse Model and Knowledge Distillation. *Electronics*, 11(4), 575. doi:10.3390/electronics11040575.
- Heck, R.; Wallick, M.; Gleicher, M. (2007). Virtual videography. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 3(1), 4. doi:10.1145/1198302.1198306.
- Kazemi, V.; Burenius, M.; Azizpour, H.; Sullivan, J. (2013). Multi-view Body Part Recognition with Random Forests. In Proceedings of the British Machine Vision Conference 2013 (pp. 48.1-48.11). doi:10.5244/C.27.48.
- Leal-Taixé, L.; Milan, A.; Schindler, K.; Cremers, D.; Reid, I.; Roth, S. (2017). Tracking the Trackers: An Analysis of the State of the Art in Multiple Object Tracking.
- Lin, T.; Maire, M.; Belongie, S.; Bourdev, L. (2015). Microsoft: COCO: Common Objects in Context. *Computer Vision*.
- Luo, Y.; Wu, T.-D.; Hwang, J.-N. (2003). Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks. *Computer Vision and Image Understanding*, 92(2-3), 196-216. doi:10.1016/j.cviu.2003.08.001.
- Matsui, K.; Iwase, M.; Agata, M.; Tanaka, T. T.; Ohnishi, N. (1998). Soccer image sequence computed by a virtual camera. In Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231) (pp. 860-865). doi:10.1109/CVPR.1998.698705.
- Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. (2016). MOT16: A Benchmark for Multi-Object Tracking.
- Owens, J. (2021). *Television Sports Production*. Routledge. (<https://www.routledge.com/Television-Sports-Production/Owens/p/book/9780367563738>).
- Pinhanez, C.; Bobick, A. F. (1995). Intelligent Studios: Using Computer Vision to Control TV Cameras.
- Rachavarapu, K. K.; Kumar, M.; Gandhi, V.; Subramanian, R. (2018). Watch to Edit: Video Retargeting using Gaze. *Computer Graphics Forum*, 37(2), 205-215. doi:10.1111/cgf.13354.
- Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4510-4520). doi:10.1109/CVPR.2018.00474.
- Shih, H. C. (2017). A Survey on Content-aware Video Analysis for Sports. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99), 1. doi:10.1109/TCSVT.2017.2655624.
- Tjondronegoro, D. W.; Chen, Y.-P. P. (2010). Knowledge-Discounted Event Detection in Sports Video. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 40(5), 1009-1024. doi:10.1109/TSMCA.2010.2046729.
- Tsuda, T.; Okuda, M.; Mutou, K.; Yanagisawa, H.; Kubo, N.; Date, Y. (2008). A High-Accuracy Image Composition System Using a Mobile Robotic Camera. *SMPTE Motion Imaging Journal*, 117(2), 30-37. doi:10.5594/J16001.
- Wales, L. (2017). *The Complete Guide to Film and Digital Production: The People and The Process* (3rd ed.). (<https://www.amazon.com/-/es/Lorene-M-Wales-ebook-dp->

B06WVF94V3/dp/B06WVF94V3/ref=mt\_other?\_encoding=UTF8&me=&qid=).

Yao, B.; Fei-Fei, L. (2010). Modeling mutual context of object and human pose in human-object interaction activities. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 17-24). doi:10.1109/CVPR.2010.5540235.

Yokoi, T.; Fujiyoshi, H. (2005). Virtual Camerawork for Generating Lecture Video from High Resolution Images. In 2005 IEEE International Conference on Multimedia and Expo (pp. 751-754). doi:10.1109/ICME.2005.1521532.