UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

Grado en Ingeniería Informática

Visión por Computador

Curso Académico 2025-2026

Trabajo de Fin de Curso

Sistema Automático de Cámara Inteligente

Presentado por:

Miguel Ángel Rodríguez Ruano

Alberto José Rodríguez Ruano

Fecha de entrega: 9 de enero de 2026

Índice

1 Documento de Memoria

1.1 Motivación/Argumentación del Trabajo

La producción audiovisual profesional tradicionalmente requiere equipos técnicos especializados y operadores de cámara experimentados para lograr encuadres cinematográficos de calidad. Un camarógrafo profesional debe dominar no solo los aspectos técnicos del equipo, sino también comprender profundamente los principios de composición visual, narrativa cinematográfica y lenguaje audiovisual. Este conocimiento se adquiere tras años de formación y práctica, lo que representa una barrera significativa de entrada para creadores de contenido, estudiantes de cine, productores independientes y profesionales que trabajan en entornos con recursos limitados.

En el contexto actual, la democratización de la creación de contenido audiovisual ha experimentado un crecimiento exponencial. Plataformas de streaming, redes sociales, educación en línea y comunicación corporativa demandan constantemente contenido de video de calidad profesional. Sin embargo, existe una brecha notable entre las aspiraciones creativas de los productores de contenido y su capacidad técnica para ejecutar movimientos de cámara y encuadres cinematográficos sofisticados. Esta problemática se agrava en situaciones donde una sola persona debe simultanear múltiples roles: presentador, director y operador de cámara.

La visión por computador, como disciplina dentro de la inteligencia artificial, ha alcanzado en los últimos años niveles de madurez que permiten la detección y seguimiento de personas en tiempo real con precisión milimétrica. Tecnologías como MediaPipe, desarrollada por Google Research, han democratizado el acceso a modelos de detección de pose humana pre-entrenados que funcionan eficientemente incluso en hardware convencional. Estas herramientas abren la posibilidad de crear sistemas inteligentes capaces de automatizar tareas que tradicionalmente requerían operadores humanos especializados.

Nuestro trabajo surge de la convergencia entre estas necesidades prácticas y las capacidades tecnológicas actuales. Identificamos la oportunidad de desarrollar un sistema que aplique principios cinematográficos profesionales de manera automática, utilizando exclusivamente técnicas de visión por computador sin requerir entrenamiento de modelos personalizados. La motivación principal es proporcionar una herramienta que permita a cualquier persona obtener resultados visuales de calidad profesional sin necesidad de conocimientos técnicos avanzados en cinematografía o equipamiento costoso.

Además, este proyecto representa una exploración innovadora de cómo la inteligencia artificial puede comprender y replicar decisiones creativas humanas. Al traducir conceptos

subjetivos como la composición visual y el lenguaje cinematográfico en algoritmos computacionales, estamos explorando los límites de la automatización inteligente en dominios tradicionalmente considerados artísticos y exclusivamente humanos.

1.2 Objetivo de la Propuesta

El objetivo principal de este trabajo es diseñar, implementar y evaluar un sistema automático de cámara inteligente capaz de realizar seguimiento de sujetos en tiempo real y aplicar encuadres cinematográficos profesionales de manera autónoma, utilizando exclusivamente tecnologías de visión por computador basadas en modelos pre-entrenados de MediaPipe.

Para alcanzar este objetivo general, establecemos los siguientes objetivos específicos que guían el desarrollo del proyecto:

- **Implementar un sistema de detección multimodal en tiempo real** que integre simultáneamente tres capacidades de MediaPipe: detección de pose corporal con 33 puntos de referencia (landmarks), reconocimiento de gestos manuales con detección de hasta dos manos simultáneamente, y detección facial para mejorar la precisión del seguimiento en planos cerrados. Esta integración multimodal debe funcionar con fluidez suficiente para procesar video en tiempo real, manteniendo tasas de refresco superiores a 20 fotogramas por segundo en hardware convencional.

- **Desarrollar un clasificador automático de planos cinematográficos** basado en métricas geométricas derivadas de los landmarks corporales detectados. Este clasificador debe ser capaz de determinar de manera inteligente cuál de los doce tipos de planos cinematográficos implementados (desde Plano General Extremo hasta Primerísimo Plano, incluyendo planos especiales como Sobre el Hombro, Vista de Espaldas, y ángulos Picado y Contrapicado) es más apropiado según la posición, distancia y orientación del sujeto respecto a la cámara. La clasificación debe basarse en estándares cinematográficos profesionales documentados en la literatura especializada.

- **Implementar un sistema de control manual mediante gestos** que permita al usuario sobrescribir las decisiones automáticas del sistema mediante poses específicas de las manos. Este sistema debe reconocer de manera robusta ocho gestos distintos (rock and roll, pulgar abajo, puño cerrado, gesto de paz, uno, tres, cuatro y cinco dedos extendidos) y mapearlos de forma intuitiva a planos cinematográficos específicos, proporcionando al usuario control creativo directo sobre el encuadre sin necesidad de interrumpir la grabación.

- **Diseñar e implementar un sistema de encuadre cinematográfico con transiciones suaves** que no solo determine qué plano aplicar, sino que también calcule dinámicamente el

punto de interés visual según el tipo de plano (siguiendo la cara para primeros planos, el torso para planos medios, o el centro de masa corporal para planos generales), aplique factores de zoom apropiados y ajustes de posición vertical específicos para cada tipo de plano, y realice transiciones suaves entre diferentes encuadres utilizando interpolación exponencial para evitar movimientos bruscos que degraden la calidad visual.

- **Incorporar capacidades de detección de orientación corporal tridimensional** que permitan al sistema identificar si el sujeto está de frente, de perfil o de espaldas a la cámara, así como detectar inclinaciones corporales y posiciones en los bordes del cuadro. Esta información debe utilizarse para activar automáticamente planos especiales como Vista de Espaldas cuando el sujeto se gira, o ángulos Picado y Contrapicado cuando se detectan posiciones verticales extremas en el frame.

- **Desarrollar un sistema multi-cámara** que permita la integración y alternancia entre múltiples fuentes de video conectadas simultáneamente, manteniendo estados de encuadre independientes para cada cámara y permitiendo transiciones fluidas entre diferentes puntos de vista durante la grabación.

- **Crear una interfaz de visualización profesional** que presente simultáneamente tres vistas distintas: un panel de control con información detallada del sistema (modo activo, plano actual, gestos detectados, orientación corporal, controles disponibles), una vista de detección que muestre visualmente los landmarks y puntos de referencia detectados para fines de depuración y comprensión del funcionamiento interno, y una vista de resultado que presente el frame final con el encuadre cinematográfico aplicado y guías visuales de composición opcionales como la regla de tercios.

- **Implementar funcionalidades avanzadas de control** que incluyan un modo de bloqueo (hold) para congelar el plano actual independientemente de los movimientos del sujeto, capacidad de captura de screenshots con metadata del plano activo, y controles para activar/desactivar elementos visuales como la regla de tercios según las necesidades del usuario.

El sistema completo debe funcionar sin requerir conexión a internet, ejecutarse en tiempo real en hardware convencional (sin necesidad de GPUs dedicadas de alta gama), y no requerir ningún proceso de entrenamiento o calibración previa, funcionando inmediatamente tras su inicialización. Además, debe ser robusto ante condiciones variables de iluminación, fondos diversos y múltiples tipos de sujetos.

1.3 Descripción Técnica del Trabajo Realizado

El sistema desarrollado se estructura en una arquitectura modular compuesta por cinco componentes principales que trabajan de manera integrada y sincronizada para lograr el seguimiento y encuadre cinematográfico automático en tiempo real.

1.3.1 Arquitectura General del Sistema

La arquitectura del sistema sigue un pipeline de procesamiento secuencial donde cada módulo recibe información del anterior y genera salidas que alimentan los módulos subsecuentes. El flujo comienza con la captura de frames desde las cámaras conectadas, continúa con el procesamiento mediante MediaPipe para extracción de características, pasa por los módulos de clasificación y decisión, y finaliza con la aplicación del encuadre y renderizado de las vistas de salida.

El sistema opera en un bucle principal de procesamiento que se ejecuta continuamente. En cada iteración, se captura un frame de la cámara activa, se convierte del espacio de color BGR (utilizado por OpenCV) al espacio RGB (requerido por MediaPipe), se procesa mediante los tres modelos de MediaPipe simultáneamente en paralelo, y se ejecutan los algoritmos de clasificación y encuadre. Este diseño permite mantener latencias bajas y tasas de refresco elevadas, fundamentales para la percepción de fluidez en el seguimiento.

1.3.2 Módulo de Detección Multimodal con MediaPipe

El primer componente del sistema implementa la detección simultánea de tres modalidades diferentes utilizando las soluciones pre-entrenadas de MediaPipe. Esta aproximación multimodal proporciona información complementaria que enriquece significativamente las capacidades del sistema.

Para la detección de pose corporal, utilizamos MediaPipe Pose configurado con complejidad de modelo intermedia (`model_complexity=1`), que proporciona un balance óptimo entre precisión y velocidad de procesamiento. La configuración incluye suavizado de landmarks activado (`smooth_landmarks=True`) para reducir el jitter entre frames consecutivos, y umbrales de confianza ajustados a 0.5 para detección inicial y 0.7 para seguimiento continuo. Estos valores fueron determinados experimentalmente para maximizar la estabilidad del seguimiento manteniendo capacidad de re-detección rápida cuando el sujeto sale y vuelve a entrar en el encuadre.

MediaPipe Pose proporciona 33 landmarks tridimensionales que representan puntos anatómicos clave del cuerpo humano: nariz, ojos, orejas, hombros, codos, muñecas, caderas, rodillas, tobillos, y puntos adicionales de las manos y pies. Cada landmark incluye coordenadas X e

Y normalizadas (valores entre 0 y 1 relativos al tamaño del frame), coordenada Z que representa profundidad relativa, y un valor de visibilidad que indica la confianza del modelo en que ese punto específico es realmente visible en la imagen (no ocluido). Esta información tridimensional es fundamental para nuestros algoritmos de clasificación y orientación.

Para el reconocimiento de gestos manuales, implementamos MediaPipe Hands configurado para detectar hasta dos manos simultáneamente (`max_num_hands=2`). Los umbrales de confianza fueron incrementados respecto a los valores por defecto (0.7 para detección y 0.6 para seguimiento) para reducir falsos positivos que podrían causar cambios de plano no intencionales. MediaPipe Hands proporciona 21 landmarks por cada mano detectada, representando la muñeca y las articulaciones de cada dedo. Utilizamos estos landmarks para implementar funciones de reconocimiento de gestos específicos mediante análisis geométrico de las posiciones relativas de las puntas y articulaciones de los dedos.

El tercer componente de detección es MediaPipe Face Detection, configurado con un umbral de confianza de 0.5. Aunque este módulo no es crítico para el funcionamiento básico del sistema, proporciona información adicional valiosa para refinar el seguimiento en planos cerrados donde la precisión del centrado facial es especialmente importante para la calidad visual del resultado.

Los tres modelos de MediaPipe se ejecutan en paralelo sobre el mismo frame RGB en cada iteración del bucle principal. Esta arquitectura paralela aprovecha eficientemente la capacidad de procesamiento disponible, ya que MediaPipe implementa optimizaciones internas que permiten ejecución concurrente de sus modelos. El tiempo total de procesamiento de los tres modelos simultáneamente es típicamente inferior a 30 milisegundos en hardware moderno, permitiendo mantener tasas de refresco superiores a 30 fotogramas por segundo.

1.3.3 Sistema de Clasificación Automática de Planos

El módulo de clasificación automática de planos cinematográficos representa el núcleo inteligente del sistema. Su función es analizar los landmarks corporales detectados y determinar qué tipo de plano cinematográfico es más apropiado según la posición, distancia y orientación del sujeto.

El algoritmo de clasificación implementado en la clase `BodyPositionDetector` se basa en la extracción y análisis de métricas geométricas derivadas de los landmarks. La métrica principal es el ancho de hombros (`shoulder_width`), calculado como la distancia euclidiana entre los landmarks 11 y 12 (hombros izquierdo y derecho) en coordenadas normalizadas. Esta métrica es inversamente proporcional a la distancia del sujeto a la cámara: cuando el sujeto está cerca, sus hombros ocupan una proporción mayor del ancho del frame; cuando está lejos, aparecen más estrechos.

Definimos rangos de `shoulder_width` que corresponden a cada tipo de plano cinematográfico basándonos en observación empírica y estándares de la industria. Valores superiores a 0.50 indican que el sujeto está muy cerca de la cámara, activando el Primerísimo Plano. Rangos entre 0.38 y 0.50 corresponden al Primer Plano. Valores entre 0.28 y 0.38 activan planos medios, diferenciando entre Plano Medio Corto y Plano Medio según la visibilidad de las caderas. Rangos inferiores a 0.15 indican que el sujeto está lejos, activando Plano General o Plano General Extremo.

Complementariamente al ancho de hombros, el algoritmo analiza la visibilidad de diferentes partes del cuerpo para refinar la clasificación. Evaluamos los valores de visibilidad de caderas (`hips_vis`), rodillas (`knees_vis`) y tobillos (`ankles_vis`) para determinar qué proporción del cuerpo es visible en el frame. Por ejemplo, si las rodillas son visibles (`knees_vis >0.3`) pero los tobillos no, clasificamos como Plano Americano. Si los tobillos son visibles (`ankles_vis >0.3`), clasificamos como Plano Entero.

El algoritmo también calcula la posición del centro de masa del sujeto dentro del frame. Cuando el sujeto se posiciona en los bordes laterales (coordenada X menor a 0.15 o mayor a 0.85), el sistema activa automáticamente el Plano General Extremo, independientemente del ancho de hombros, siguiendo el principio cinematográfico de proporcionar más contexto espacial cuando el sujeto no está centrado. Similarmente, posiciones verticales extremas (coordenada Y menor a 0.2 o mayor a 0.8) activan ángulos Picado o Contrapicado respectivamente.

Un aspecto crítico del sistema es la detección de orientación corporal, implementada mediante análisis de las coordenadas Z de los landmarks. Comparamos la coordenada Z de la nariz con el promedio de las coordenadas Z de ambos hombros. Si la nariz está significativamente más cerca de la cámara que los hombros (diferencia mayor a 0.1 en coordenadas normalizadas), clasificamos la orientación como frontal. Si los hombros están más cerca que la nariz, el sujeto está de espaldas, activando el plano especial Vista de Espaldas. Valores intermedios indican orientación de perfil, que puede activar el plano Sobre el Hombro si se cumplen condiciones adicionales de asimetría lateral.

Para garantizar estabilidad temporal y evitar cambios de plano erráticos causados por fluctuaciones momentáneas en la detección, implementamos un sistema de suavizado temporal mediante una cola de decisiones recientes (`deque` con capacidad para 3 elementos). El plano final emitido por el clasificador es el más frecuente entre las últimas tres decisiones, implementando efectivamente un filtro de moda temporal que elimina cambios espurios mientras mantiene capacidad de respuesta rápida ante cambios genuinos.

1.3.4 Sistema de Control Manual por Gestos

Paralelamente al sistema de clasificación automática, implementamos un modo de control manual que permite al usuario especificar explícitamente el plano deseado mediante gestos de mano. Este sistema proporciona flexibilidad creativa y control directo sobre el encuadre cuando el usuario desea sobrescribir las decisiones automáticas del sistema.

El reconocimiento de gestos se basa en análisis geométrico de las posiciones relativas de los landmarks de la mano proporcionados por MediaPipe Hands. Implementamos funciones especializadas para detectar cada gesto específico mediante combinaciones de condiciones sobre las posiciones de las puntas y articulaciones de los dedos.

La función `contar_dedos` implementa un algoritmo que determina cuántos dedos están extendidos comparando las coordenadas Y de las puntas de los dedos con sus articulaciones correspondientes. Para el pulgar, que se mueve lateralmente, comparamos coordenadas X en lugar de Y. Esta función proporciona la base para reconocer gestos de uno, tres, cuatro y cinco dedos extendidos.

Para el gesto rock and roll (índice y meñique extendidos, medio y anular cerrados), implementamos la función `detectar_rock_and_roll` que verifica que el índice y el meñique estén extendidos (coordenadas Y de las puntas por encima de las articulaciones) mientras que el medio y el anular estén cerrados. Este gesto característico se mapea al Primerísimo Plano, proporcionando una forma intuitiva y memorable de activar el plano más cercano.

Para el pulgar abajo, implementamos una función dedicada que verifica no solo que el pulgar esté en posición inferior (comparación de coordenadas Y), sino también que los demás cuatro dedos estén cerrados (puntas de dedos por debajo de sus articulaciones medias). Esta combinación de condiciones asegura que el gesto sea inequívoco y minimiza falsos positivos. El pulgar abajo se mapea al Plano General Extremo, el plano más lejano.

El gesto de paz (señal de victoria con índice y medio extendidos) requiere una detección más sofisticada. Verificamos que el índice y el dedo medio estén extendidos (puntas por encima de articulaciones) mientras que el anular y el meñique estén cerrados (puntas por debajo de articulaciones). Esta combinación específica solo se satisface con el gesto de paz, evitando confusión con otros gestos.

El mapeo entre gestos y planos fue diseñado intentando cierta intuitividad nemotécnica: el rock and roll (gesto enérgico) se mapea al plano más cercano (Primerísimo Plano), mientras que el pulgar abajo se mapea al plano más lejano (Plano General Extremo). El puño cerrado (compacto) se mapea a Plano General, mientras que la mano completamente abierta (cinco dedos) se mapea a Plano Medio Corto. Los gestos de uno, tres y cuatro dedos se mapean a Plano Entero, Plano Medio y Plano Americano respectivamente.

El sistema permite alternar entre modo automático y modo manual mediante la tecla 'm'.

En modo automático, el clasificador de planos determina continuamente el encuadre apropiado. En modo manual, el último gesto detectado establece el plano activo, que se mantiene hasta que se detecte un nuevo gesto o se vuelva a modo automático. Esta dualidad proporciona flexibilidad: el usuario puede permitir que el sistema funcione autónomamente la mayor parte del tiempo, interviniendo manualmente solo cuando desea un encuadre específico para propósitos creativos o narrativos particulares.

1.3.5 Sistema de Encuadre y Seguimiento

Una vez determinado el plano cinematográfico apropiado (ya sea automáticamente o por gesto manual), el sistema de encuadre debe calcular cómo transformar el frame capturado para lograr el efecto visual deseado. Este proceso involucra tres aspectos fundamentales: determinar el punto de interés visual a seguir, calcular el factor de zoom apropiado, y aplicar transiciones suaves entre diferentes estados de encuadre.

El cálculo del punto de seguimiento varía según el tipo de plano activo, siguiendo principios cinematográficos establecidos. Para planos cerrados (Primer Plano y Primerísimo Plano), el punto de interés es el rostro del sujeto, calculado como el centroide entre la nariz y el punto medio entre las comisuras de los labios. Esta posición asegura que la cara esté centrada y correctamente encuadrada. Para planos medios (Plano Medio y Plano Medio Corto), calculamos un punto ponderado entre la nariz y el centro de los hombros, priorizando ligeramente la parte superior del torso. Para planos más abiertos, utilizamos el centro de masa de múltiples landmarks clave incluyendo cabeza, hombros y caderas.

Cada tipo de plano tiene asociado un factor de zoom específico que determina qué porción del frame original se utilizará. El Plano General Extremo utiliza un zoom de 0.7 (mostrando más del 100 % del frame original mediante adición de contexto), mientras que el Primerísimo Plano utiliza un zoom de 2.8 (mostrando aproximadamente un tercio del ancho original). Estos valores fueron calibrados para cumplir con las proporciones estándar de cada tipo de plano según la literatura cinematográfica consultada.

Adicionalmente, cada plano incluye un ajuste vertical (`y_offset`) que desplaza el punto de seguimiento hacia arriba o hacia abajo en el frame. Los planos cerrados típicamente tienen offsets negativos para proporcionar más "espacio de cabeza" (headroom) siguiendo la convención cinematográfica de no centrar la cabeza exactamente en el medio del frame sino ligeramente por encima del centro. Los planos generales tienen offsets positivos o neutros para mostrar más contexto inferior.

El aspecto más crítico para la calidad visual del sistema es la implementación de transiciones suaves entre diferentes estados de encuadre. Cambios abruptos de zoom o posición son visualmente desagradables y revelan inmediatamente la naturaleza automatizada del sis-

tema. Para resolver esto, implementamos la clase `SmoothFramer` que mantiene un estado actual de encuadre (posición X, posición Y, y zoom) y un estado objetivo. En cada frame, el estado actual se mueve hacia el estado objetivo mediante interpolación exponencial con un factor de suavizado de 0.15.

Matemáticamente, la actualización se expresa como:

$$\text{estado\_actual} \leftarrow \text{estado\_actual} + 0{,}15 \times (\text{estado\_objetivo} - \text{estado\_actual})$$

Este tipo de interpolación produce movimientos que aceleran inicialmente y desaceleran al acercarse al objetivo, creando transiciones que se perciben orgánicas y profesionales. El factor 0.15 fue ajustado experimentalmente: valores menores producen transiciones más suaves pero lentas, mientras que valores mayores producen transiciones más rápidas pero potencialmente bruscas.

La aplicación física del encuadre se realiza mediante operaciones de recorte (crop) y redimensionamiento. Calculamos las dimensiones del recorte en píxeles según el factor de zoom actual, determinamos las coordenadas del rectángulo de recorte centrado en el punto de seguimiento (con validación de límites para asegurar que no excedemos los bordes del frame original), extraemos la sub-región correspondiente, y la redimensionamos al tamaño completo del frame de salida mediante interpolación bilineal.

1.3.6 Funcionalidades Avanzadas de Control

El sistema implementa varias funcionalidades adicionales que enriquecen significativamente la experiencia del usuario y las capacidades prácticas del sistema.

Modo Hold (Bloqueo de Plano): Activado mediante la tecla 'h', este modo congela el plano actual independientemente de los movimientos del sujeto o detecciones automáticas. Cuando está activo, ni el clasificador automático ni los gestos manuales pueden cambiar el plano, permitiendo al usuario mantener un encuadre específico durante el tiempo deseado. Esto es especialmente útil en situaciones donde se requiere consistencia visual absoluta o cuando el usuario ha encontrado el encuadre perfecto y no desea arriesgarse a cambios no intencionales. El panel de control muestra claramente el estado del modo Hold mediante indicadores visuales y cambios de color.

Sistema de Captura de Screenshots: Mediante la tecla 's', el usuario puede capturar instantáneamente el frame actual de la ventana de resultado. Las capturas se guardan automáticamente en una carpeta `screenshots/` con nombres descriptivos que incluyen timestamp (año, mes, día, hora, minuto, segundo) y el nombre del plano cinematográfico activo en el

momento de la captura. Por ejemplo: `shot_20260109_143025_Primer_Plano.png`. Esta funcionalidad facilita la documentación del trabajo, permite comparar diferentes encuadres, y proporciona material de referencia para futuras producciones.

Toggle de Regla de Tercios: La tecla 'g' permite activar o desactivar la visualización de la regla de tercios en la ventana de resultado. Esta regla se renderiza como líneas grises tenues que dividen el frame en una cuadrícula de 3x3, proporcionando guías visuales para evaluar la composición según principios clásicos de fotografía y cinematografía. El sistema inicia por defecto con la regla activada, pero el usuario puede desactivarla cuando desee una visualización completamente limpia del resultado final. El estado actual del grid se muestra en el panel de control.

Sistema Multi-Cámara: El sistema soporta conexión simultánea de múltiples cámaras y permite alternar entre ellas durante la ejecución mediante la tecla 'c'. En la inicialización, el sistema intenta abrir los dispositivos de captura en índices 0 y 1 (típicamente cámara integrada y cámara USB externa en sistemas portátiles). Las cámaras que se abren exitosamente se añaden a una lista de cámaras disponibles, cada una con su propio estado de encuadre independiente. Cada cámara tiene asociada una instancia independiente de `SmoothFramer`, lo que asegura que al alternar entre cámaras, cada una mantiene su estado de transición actual. Esto previene saltos visuales cuando se vuelve a una cámara previamente utilizada: el encuadre continúa desde donde estaba, no desde un estado inicial.

Interfaz de Visualización Profesional: La arquitectura de visualización del sistema presenta simultáneamente tres ventanas que muestran diferentes aspectos del procesamiento:

- **Ventana "CONTROL":** Presenta un panel de información detallada con fondo negro donde se renderiza texto mostrando el modo activo (automático o manual) con indicación visual si el modo Hold está activado, el plano cinematográfico actual con su descripción, el plano que el sistema automático detectaría (útil cuando se está en modo manual para ver qué decisión tomaría el sistema), la métrica de distancia (shoulder_width) para propósitos de debugging, la orientación corporal en tres ejes (yaw, pitch, roll) cuando está disponible, el estado de las funcionalidades (Hold Lock ON/OFF, Grid ON/OFF), controles de teclado disponibles con descripción de cada tecla, y referencia visual completa de los gestos manuales con sus planos correspondientes.
- **Ventana "DETECCION":** Muestra el frame original con los landmarks detectados superpuestos visualmente. Utilizamos las funciones de dibujado proporcionadas por MediaPipe que renderizan los puntos de landmarks como círculos y las conexiones entre puntos como líneas, utilizando esquemas de color predefinidos que distinguen visualmente diferentes partes del cuerpo. Los landmarks de pose corporal se muestran en verde y azul, mientras

que los landmarks de las manos se muestran en rojo. Esta visualización es invaluable para comprender qué está detectando el sistema en tiempo real y para depuración cuando el comportamiento no es el esperado.

- **Ventana "RESULTADO":** Presenta el frame final después de aplicar el encuadre cinematográfico en formato completamente limpio, sin overlays de texto que distraigan de la evaluación visual del encuadre. El único elemento opcional es la regla de tercios (cuando está activada), dibujada como líneas grises tenues que dividen el frame en una cuadrícula de 3x3. Esta presentación minimalista permite al usuario enfocarse completamente en la calidad compositiva del encuadre y evaluar si el resultado cumple con estándares cinematográficos profesionales.

El sistema calcula y mantiene un promedio móvil de la tasa de frames por segundo utilizando una cola de las últimas 30 mediciones. Este promedio suavizado proporciona una métrica más estable y representativa del rendimiento del sistema que mediciones instantáneas que podrían fluctuar significativamente. El FPS se muestra prominentemente en el panel de control con código de color verde para indicar rendimiento saludable.

1.4 Fuentes y Tecnologías Utilizadas

El desarrollo de este proyecto se fundamenta en un ecosistema de tecnologías de código abierto y documentación especializada que abarca tanto el ámbito técnico de la visión por computador como el conocimiento cinematográfico profesional.

1.4.1 Tecnologías de Visión por Computador

La columna vertebral tecnológica del sistema es **MediaPipe**, una plataforma de código abierto desarrollada por Google Research para construir pipelines de machine learning aplicados a procesamiento de medios. Específicamente, utilizamos tres soluciones de MediaPipe:

- **MediaPipe Pose:** Proporciona detección de pose humana en tiempo real mediante un modelo de machine learning que identifica 33 landmarks corporales tridimensionales. El modelo utiliza una arquitectura en dos etapas: primero un detector ligero localiza la región del cuerpo en la imagen, luego una red más compleja predice las posiciones precisas de los landmarks dentro de esa región. Esta arquitectura permite procesamiento eficiente manteniendo alta precisión. MediaPipe Pose fue entrenado en datasets masivos que incluyen diversas etnias, tipos corporales, vestimentas e iluminaciones, proporcionando robustez ante variabilidad real. La documentación oficial y guías de implementación de MediaPipe Pose fueron recursos fundamentales durante el desarrollo.

- **MediaPipe Hands:** Implementa detección y seguimiento de manos con 21 landmarks por mano detectada. Utiliza una arquitectura similar de detector + predictor de landmarks, optimizada específicamente para las características geométricas de las manos humanas. La capacidad de detectar hasta dos manos simultáneamente fue crucial para nuestro sistema de gestos manuales. MediaPipe Hands incorpora seguimiento temporal que aprovecha la posición de las manos en frames anteriores para mejorar eficiencia y estabilidad.

- **MediaPipe Face Detection:** Proporciona detección rápida de rostros con información de bounding box y puntos clave faciales. Aunque menos central para nuestro sistema que los dos anteriores, contribuye al refinamiento del seguimiento en planos cerrados.

Los tres modelos de MediaPipe están pre-entrenados y se distribuyen con la biblioteca, eliminando completamente la necesidad de recolección de datos, etiquetado, entrenamiento o ajuste fino de modelos. Esta característica fue determinante en la viabilidad del proyecto: pudimos enfocarnos completamente en la lógica de aplicación cinematográfica sin invertir recursos en el desarrollo de capacidades de detección básicas.

Complementariamente a MediaPipe, utilizamos las siguientes bibliotecas fundamentales:

- **OpenCV (Open Source Computer Vision Library):** Biblioteca fundamental para procesamiento de imágenes y video. Utilizamos OpenCV para captura de frames desde las cámaras (`cv2.VideoCapture`), manipulación de espacios de color (conversión BGR-RGB), operaciones geométricas (recorte y redimensionamiento de frames), renderizado de elementos visuales (texto, líneas, figuras), y gestión de ventanas de visualización. OpenCV proporciona implementaciones altamente optimizadas de estas operaciones, muchas de ellas con aceleración por hardware cuando está disponible.

- **NumPy:** Biblioteca de computación numérica en Python que sustenta todo el procesamiento matemático del sistema. Utilizamos NumPy para cálculos vectoriales (distancias entre puntos, centroides de conjuntos de puntos), operaciones de álgebra lineal (vectores para cálculo de orientación), funciones estadísticas (medias, promedios móviles), y manipulación eficiente de arrays multidimensionales que representan imágenes y coordenadas. La implementación en C de las operaciones de NumPy proporciona rendimiento crítico para mantener procesamiento en tiempo real.

1.4.2 Recursos de Conocimiento Cinematográfico

El diseño del sistema de clasificación de planos y las decisiones sobre parámetros de encuadre se fundamentan en documentación especializada sobre lenguaje cinematográfico y composición visual profesional consultada en diversos portales educativos de cinematografía.

El estudio de estas fuentes permitió comprender la taxonomía completa de tipos de plano cinematográfico, desde planos generales que establecen contexto espacial hasta primeros planos que capturan emociones íntimas. Se analizaron las proporciones específicas del cuerpo humano que deben ser visibles en cada tipo de plano según estándares de la industria: el Plano Americano corta a la altura de las rodillas, el Plano Medio a la cintura, el Plano Entero muestra de pies a cabeza, y el Primer Plano se centra en rostro y hombros. Estas especificaciones anatómicas fueron directamente traducidas a los umbrales de visibilidad de landmarks corporales implementados en el algoritmo de clasificación automática.

Adicionalmente, la documentación consultada proporcionó conocimiento sobre clasificación funcional de planos en tres categorías: descriptivos (contexto espacial), narrativos (acción y diálogo), y expresivos (emoción), lo cual informó las decisiones sobre cuándo activar automáticamente cada tipo de plano. También se estudiaron aspectos técnicos de composición visual profesional como el espacio de cabeza (headroom) apropiado para cada plano, la regla de los tercios como principio compositivo fundamental, y la progresión lógica de encuadres desde planos abiertos a cerrados para mantener coherencia narrativa.

Los enlaces completos a todos los recursos cinematográficos consultados se incluyen en la sección de créditos al final de este documento.

1.4.3 Entorno de Desarrollo

El sistema fue desarrollado completamente en **Python 3.10**, aprovechando su ecosistema maduro de bibliotecas científicas y de visión por computador. Python proporciona el balance ideal entre facilidad de desarrollo y rendimiento suficiente para aplicaciones de tiempo real cuando se utilizan bibliotecas optimizadas como NumPy y OpenCV.

El entorno de desarrollo se gestionó mediante **Anaconda**, una distribución de Python especializada en computación científica que facilita la gestión de dependencias y entornos virtuales aislados. Anaconda permitió crear un entorno dedicado para el proyecto con versiones específicas de todas las bibliotecas necesarias, evitando conflictos de dependencias y garantizando reproducibilidad del entorno de ejecución en diferentes máquinas.

Durante las fases iniciales de prototipado y experimentación se utilizaron **Jupyter Notebooks**, que proporcionaron un entorno interactivo ideal para explorar capacidades de MediaPipe, ajustar parámetros de detección, y visualizar resultados intermedios de manera iterativa. Los notebooks permitieron documentar el proceso de experimentación combinando código ejecutable, visualizaciones y anotaciones explicativas en un mismo documento.

El desarrollo del sistema final se realizó utilizando **Visual Studio Code** como entorno de desarrollo integrado, aprovechando sus capacidades avanzadas de depuración de Python, autocompletado inteligente, y extensiones especializadas para desarrollo con OpenCV y visualización de estructuras de datos. El control de versiones se gestionó mediante **Git**, permitiendo experimentación con diferentes enfoques mediante ramas de desarrollo y reversión a estados estables cuando fue necesario.

Utilizamos estructuras de datos especializadas de Python para implementar funcionalidad específica: `deque` de la biblioteca `collections` para implementar colas de capacidad limitada con operaciones eficientes, fundamentales para nuestros sistemas de suavizado temporal; `dataclass` para definir estructuras de datos simples como `FrameTarget`; y `Enum` para definir tipos enumerados como `TipoPlano` que proporcionan seguridad de tipos y claridad en el código.

Las pruebas se realizaron en múltiples configuraciones de hardware: computadoras portátiles con procesadores Intel Core i5 y i7 de diferentes generaciones, sistemas de escritorio con y sin GPU dedicada, y diferentes modelos de cámaras web (integradas y USB externas). Esta diversidad de hardware de prueba aseguró que el sistema fuera robusto y funcionara en equipos convencionales sin requerir hardware especializado.

1.5 Conclusiones y Propuestas de Ampliación

El desarrollo de este Sistema Automático de Cámara Inteligente ha demostrado la viabilidad de aplicar principios cinematográficos profesionales mediante técnicas de visión por computador sin requerir entrenamiento de modelos personalizados. Los objetivos planteados inicialmente se han cumplido satisfactoriamente, resultando en un sistema funcional capaz de realizar seguimiento de sujetos y aplicar encuadres cinematográficos en tiempo real con calidad visual profesional.

1.5.1 Logros Principales

La integración exitosa de múltiples soluciones de MediaPipe (Pose, Hands, Face Detection) en un pipeline unificado ha proporcionado capacidades de percepción robustas que funcionan en condiciones variables de iluminación, fondos y tipos de sujetos. El sistema de clasificación automática de planos basado en métricas geométricas derivadas de landmarks corporales ha demostrado ser sorprendentemente preciso, tomando decisiones que en la mayoría de los casos coinciden con lo que un camarógrafo humano elegiría en situaciones similares.

El sistema de control manual mediante gestos complementa efectivamente la operación automática, proporcionando flexibilidad creativa sin sacrificar la fluidez de la experiencia.

La incorporación del gesto rock and roll para el Primerísimo Plano ha demostrado ser intuitivo y memorable, facilitando el control manual sin ambigüedad. La capacidad de alternar instantáneamente entre modo automático y manual permite workflows híbridos donde el sistema opera autónomamente la mayor parte del tiempo, con intervenciones manuales puntuales cuando se requiere un encuadre específico por razones narrativas o estéticas.

Las transiciones suaves implementadas mediante interpolación exponencial han resultado ser efectivas para crear movimientos de cámara que se perciben naturales y profesionales. La diferencia entre un sistema con y sin este suavizado es dramática: sin él, los cambios de encuadre son abruptos y revelan inmediatamente la naturaleza automatizada del sistema; con él, las transiciones fluyen orgánicamente de manera que el espectador casual puede no darse cuenta de que está observando un sistema automático.

El modo Hold (bloqueo de plano) ha probado ser especialmente valioso en situaciones de producción real donde se requiere mantener un encuadre específico a pesar de movimientos del sujeto. Esta funcionalidad proporciona el control preciso necesario para situaciones donde la consistencia visual es crítica, como entrevistas formales o presentaciones estructuradas.

La funcionalidad de captura de screenshots con metadata automática facilita significativamente la documentación del trabajo y permite análisis posterior de la efectividad de diferentes encuadres. El sistema de nomenclatura con timestamp y nombre de plano hace que las capturas sean autoexplicativas y fáciles de organizar.

La arquitectura multi-cámara ha probado ser valiosa en escenarios de producción donde se desea capturar simultáneamente desde múltiples ángulos. La capacidad de cada cámara de mantener su propio estado de encuadre independiente permite crear configuraciones donde diferentes cámaras siguen diferentes estrategias (por ejemplo, una en modo automático para seguimiento general y otra en modo manual manteniendo un plano fijo específico).

1.5.2 Limitaciones Identificadas

A pesar de los logros, el desarrollo también ha revelado limitaciones que representan oportunidades de mejora. La detección de landmarks mediante MediaPipe, aunque robusta en condiciones normales, puede degradarse significativamente en situaciones de iluminación extremadamente baja o contraluz severo. En estos escenarios, la pérdida intermitente de detección causa fluctuaciones en el encuadre que, aunque suavizadas por el sistema de interpolación, son perceptibles.

El sistema actual está optimizado para seguimiento de un único sujeto. En escenarios con

múltiples personas en el frame, MediaPipe Pose puede alternar entre detectar diferentes personas o proporcionar detecciones parciales de varias personas simultáneamente, causando comportamiento errático del encuadre. Una ampliación deseable sería implementar seguimiento multi-persona con lógica para decidir a quién seguir basándose en criterios como cercanía, centralidad, o incluso reconocimiento del sujeto principal.

La latencia del sistema, aunque suficientemente baja para la mayoría de aplicaciones (típicamente 30-50ms), puede ser perceptible en escenarios donde el sujeto realiza movimientos muy rápidos. El retraso inherente entre el movimiento físico y la reacción del encuadre es una limitación fundamental de cualquier sistema reactivo. Técnicas de predicción de movimiento podrían potencialmente reducir esta latencia aparente anticipando trayectorias.

El sistema de gestos manuales, aunque funcional, requiere poses de mano relativamente precisas y estables para evitar falsos positivos. En situaciones donde el usuario está realizando otras actividades con las manos, el sistema puede interpretar erróneamente gestos no intencionales como comandos. El modo Hold mitiga parcialmente este problema permitiendo al usuario bloquear el plano cuando no desea cambios, pero una solución más robusta sería deseable.

1.5.3 Propuestas de Ampliación

Basándonos en la experiencia de desarrollo y las limitaciones identificadas, proponemos varias direcciones para futuras ampliaciones del sistema:

- **Seguimiento multi-persona inteligente:** Implementar capacidades para detectar y seguir múltiples personas simultáneamente, con lógica para decidir dinámicamente a quién encuadrar o cómo componer el encuadre para incluir múltiples sujetos. Esto podría incluir detección de interacciones entre personas (por ejemplo, dos personas conversando) para activar automáticamente planos especiales como plano-contraplano o planos de dos.

- **Reconocimiento de contexto y escena:** Integrar capacidades de detección de objetos y segmentación semántica para que el sistema comprenda el contexto espacial más allá de la detección de personas. Por ejemplo, detectar muebles, puertas, ventanas, y utilizar esta información para tomar decisiones de encuadre más informadas que respeten la composición espacial de la escena.

- **Análisis de contenido semántico:** Incorporar análisis del contenido verbal (mediante reconocimiento de voz) o visual (mediante reconocimiento de acciones) para ajustar el encuadre según lo que está ocurriendo. Por ejemplo, acercarse automáticamente durante momentos de diálogo emocional o alejarse durante momentos de acción física amplia.

- **Perfiles de estilo cinematográfico:** Implementar diferentes perfiles que emulen estilos cinematográficos de directores o géneros específicos. Por ejemplo, un perfil "Wes Anderson" que priorice composiciones simétricas y centradas, versus un perfil "documental" que mantenga encuadres más distantes y estables, versus un perfil "acción" que utilice encuadres más dinámicos y cerrados.

- **Integración con sistemas de iluminación inteligente:** Desarrollar comunicación con sistemas de iluminación automatizada para que no solo el encuadre sino también la iluminación se ajuste según el plano activo, creando una solución integral de producción automatizada.

- **Capacidades de post-procesamiento:** Implementar análisis retrospectivo de las grabaciones para identificar momentos donde el encuadre automático no fue óptimo y sugerir o aplicar automáticamente correcciones, aprovechando que en post-producción se dispone de información temporal completa (pasado y futuro) que no está disponible durante grabación en tiempo real.

- **Exportación de metadatos temporales:** Generar archivos de metadatos que documenten qué planos estaban activos en qué momentos de la grabación, facilitando edición posterior y permitiendo recrear o ajustar decisiones de encuadre en post-producción.

- **Interfaz web para control remoto:** Desarrollar una interfaz web que permita controlar el sistema remotamente desde dispositivos móviles o tabletas, permitiendo que un asistente monitoree y ajuste el encuadre sin estar físicamente en la ubicación de la cámara.

- **Machine learning para personalización:** Implementar un sistema que aprenda de las correcciones manuales del usuario (cuándo interviene manualmente para cambiar planos o activa el modo Hold) para ajustar progresivamente los umbrales y preferencias del sistema a las preferencias estéticas específicas de cada usuario.

- **Modo de grabación con buffer temporal:** Implementar un buffer circular que permita al sistema grabar continuamente los últimos 30-60 segundos, permitiendo al usuario "guardar" retroactivamente momentos que ya ocurrieron cuando decide que el encuadre fue particularmente exitoso.

1.5.4 Reflexión Final

Este proyecto ha demostrado que la intersección entre visión por computador y arte cinematográfico es un campo fértil para innovación. La automatización inteligente no necesariamente disminuye la calidad artística; cuando se implementa con comprensión profunda de los principios subyacentes, puede democratizar capacidades que tradicionalmente requerían

años de experiencia especializada.

La experiencia de traducir conceptos subjetivos y artísticos como "composición visual" o "narrativa cinematográfica" en algoritmos concretos y métricas cuantificables ha sido intelectualmente desafiante y reveladora. Ha requerido no solo competencias técnicas en programación y visión por computador, sino también estudio profundo de teoría cinematográfica y composición visual.

El resultado es un sistema que, si bien no reemplaza la creatividad y juicio de un camarógrafo humano experto en situaciones complejas, proporciona capacidades muy superiores a las que tendría una persona sin experiencia operando una cámara manualmente. Las funcionalidades adicionales implementadas (modo Hold, captura de screenshots, toggle de grid) demuestran cómo pequeños refinamientos orientados al usuario pueden transformar un prototipo técnico en una herramienta práctica de producción.

En este sentido, cumple su objetivo fundamental de democratizar la producción audiovisual de calidad, proporcionando a creadores individuales herramientas que antes solo estaban disponibles para producciones con equipos técnicos completos.

1.6 Indicación de Herramientas/Tecnologías con las que les Hubiera Gustado Contar

Aunque el ecosistema de herramientas utilizado fue suficiente para lograr los objetivos del proyecto, el proceso de desarrollo identificó varias tecnologías adicionales que habrían facilitado o enriquecido ciertos aspectos del trabajo.

1.6.1 Herramientas de Hardware

- **Cámaras PTZ (Pan-Tilt-Zoom) motorizadas:** Habrían permitido implementar seguimiento físico real en lugar de seguimiento digital mediante recorte y redimensionamiento de imagen. Las cámaras PTZ profesionales pueden controlarse programáticamente mediante protocolos como VISCA o NDI, permitiendo que el sistema moviera físicamente la cámara y ajustara el zoom óptico real. Esto eliminaría la pérdida de resolución inherente al zoom digital queutiliza nuestro sistema actual y proporcionaría mayor flexibilidad en el rango de movimiento.

- **Sistema de iluminación inteligente controlable:** Luces LED programables con control de intensidad, temperatura de color y dirección habrían permitido crear un sistema verdaderamente integral donde no solo el encuadre sino también la iluminación se ajustara automáticamente según el tipo de plano. Por ejemplo, planos cerrados podrían beneficiarse de iluminación más suave y difusa, mientras que planos generales podrían utilizar iluminación más amplia y uniforme.

- **Sensores de profundidad dedicados (LiDAR o Time-of-Flight):** Aunque MediaPipe

proporciona información de profundidad relativa mediante la coordenada Z de los landmarks, sensores de profundidad dedicados como los utilizados en dispositivos como el iPhone Pro o Microsoft Kinect habrían proporcionado información tridimensional más precisa y densa, mejorando la detección de orientación corporal y permitiendo efectos más sofisticados como desenfoques de fondo controlados (bokeh artificial) más realistas.

- **Cámaras de mayor calidad y múltiples ángulos:** Cámaras 4K o superiores con lentes intercambiables habrían proporcionado mayor resolución base, permitiendo zoom digital más agresivo sin degradación perceptible de calidad. Un setup multi-cámara profesional con 3-4 cámaras fijas en diferentes ángulos habría permitido explorar sistemas de selección automática de cámara basados en la acción y composición.

- **Sistema de rieles motorizados (slider motorizado):** Un slider o dolly motorizado controlable programáticamente habría permitido implementar movimientos de cámara horizontales y verticales adicionales a los movimientos de encuadre digital, añadiendo dimensión cinemática real al sistema.

- **Micrófonos direccionales automatizados:** Micrófonos con capacidad de enfoque direccional controlable habrían permitido crear un sistema audiovisual integral donde tanto la imagen como el audio se ajustaran automáticamente para seguir al sujeto.

1.6.2 Herramientas de Software y Bibliotecas

- **Modelos de detección multi-persona más avanzados:** Frameworks como YOLO (You Only Look Once) v8 o Detectron2 de Facebook AI habrían proporcionado capacidades superiores de detección y seguimiento de múltiples personas simultáneamente, con identificación persistente de individuos entre frames (tracking de identidad) que MediaPipe no proporciona nativamente.

- **Bibliotecas de análisis de escena semántica:** Herramientas como DeepLab para segmentación semántica habrían permitido al sistema comprender el contexto espacial completo de la escena (detectar muebles, paredes, objetos) y tomar decisiones de encuadre más informadas que respeten la composición espacial del entorno.

- **Sistemas de predicción de movimiento:** Frameworks especializados en predicción de trayectorias humanas como Social-LSTM o Trajectron++ habrían permitido anticipar movimientos futuros del sujeto, reduciendo la latencia aparente del sistema y permitiendo encuadres más proactivos que reactivos.

- **Frameworks de reconocimiento de actividades:** Bibliotecas especializadas en reconocimiento de acciones humanas (como I3D, SlowFast, o TSM) habrían permitido que el

sistema comprenda qué actividad está realizando el sujeto (caminando, sentándose, gesticulando) y ajuste el encuadre según el contexto de la acción.

- **Motores de procesamiento de video en tiempo real más avanzados:** Frameworks como GStreamer o FFmpeg con aceleración GPU completa habrían proporcionado mayor rendimiento y capacidad de procesamiento de múltiples streams de video simultáneamente en resoluciones más altas.

- **Herramientas de análisis de composición visual:** Algoritecas especializados en evaluación de composición fotográfica que pudieran calcular métricas de calidad compositiva (balance, simetría, puntos de interés según regla de tercios) habrían permitido que el sistema auto-evalúe y optimice sus decisiones de encuadre.

- **Sistemas de reconocimiento de emociones faciales:** Bibliotecas como FER (Facial Emotion Recognition) o AffectNet habrían permitido que el sistema detecte el estado emocional del sujeto y ajuste el encuadre apropiadamente (por ejemplo, acercándose automáticamente durante momentos de emoción intensa).

- **APIs de procesamiento de lenguaje natural para control por voz:** Integración con sistemas como Whisper de OpenAI para reconocimiento de voz habría permitido control del sistema mediante comandos verbales, haciendo innecesario el uso de gestos manuales que pueden ser incómodos durante presentaciones.

1.6.3 Infraestructura de Desarrollo y Pruebas

- **GPUs dedicadas de alta gama:** Aunque el sistema funciona en hardware convencional, GPUs como NVIDIA RTX 4090 o A100 habrían permitido experimentar con modelos más complejos y procesamiento de mayor resolución sin comprometer la tasa de frames.

- **Entorno de pruebas profesional:** Un espacio dedicado con iluminación controlada, fondos profesionales, y múltiples posiciones de cámara preconfiguradas habría facilitado pruebas sistemáticas y reproducibles del sistema en condiciones estandarizadas.

- **Software de análisis de calidad de video profesional:** Herramientas como DaVinci Resolve Studio o Adobe Premiere Pro con acceso a plugins de análisis de composición habrían facilitado la evaluación objetiva de la calidad de los encuadres generados por el sistema.

- **Plataforma de anotación y evaluación de datos:** Herramientas como Label Studio o CVAT habrían facilitado la creación de datasets de evaluación con encuadres marcados por expertos en cinematografía, permitiendo evaluación cuantitativa del sistema comparando sus decisiones con las de profesionales.

- **Sistema de integración continua con pruebas automatizadas:** Infraestructura CI/CD como GitHub Actions con runners equipados con cámaras y capacidad de procesamiento de video habría permitido pruebas automatizadas del sistema ante cada cambio de código.

1.6.4 Conocimiento y Recursos Educativos

- **Acceso a consultoría con directores de fotografía profesionales:** Aunque la documentación online fue valiosa, tener acceso directo a profesionales de la cinematografía que pudieran evaluar las decisiones del sistema y proporcionar feedback habría acelerado significativamente el refinamiento de los algoritmos de clasificación y encuadre.

- **Datasets anotados de cinematografía:** Colecciones de clips de películas profesionales con anotaciones detalladas sobre tipos de plano, composición, y decisiones de cámara habrían proporcionado material invaluable para entrenar y evaluar el sistema.

- **Cursos especializados en cinematografía técnica:** Formación formal en operación de cámara, lenguaje cinematográfico y dirección de fotografía habría proporcionado comprensión más profunda de aspectos sutiles que son difíciles de capturar mediante lectura de documentación solamente.

A pesar de no disponer de estas herramientas y recursos adicionales, el proyecto logró sus objetivos utilizando tecnologías de código abierto accesibles y documentación pública. Esta limitación de recursos, en retrospectiva, puede considerarse una fortaleza: demuestra que es posible crear sistemas sofisticados de visión por computador con aplicaciones prácticas reales utilizando únicamente herramientas gratuitas y hardware convencional, verdaderamente democratizando el acceso a estas tecnologías.

1.7 Créditos Materiales no Originales del Grupo

Este proyecto utiliza exclusivamente tecnologías de código abierto y recursos educativos de acceso público. A continuación detallamos todos los materiales de terceros utilizados:

1.7.1 Recursos Educativos y Documentación

- **Documentación oficial de MediaPipe:** Guías oficiales y tutoriales disponibles en `https://ai.google.dev/edge/mediapipe/` fueron fundamentales para comprender las capacidades y limitaciones de las soluciones de detección utilizadas.

- **Aprender Cine:** Taxonomía de tipos de plano cinematográfico disponible en `https://aprendercine.com/tipos-de-plano/`

- **Catt's Camera Blog:** Guía completa de planos en cine disponible en `https://blog.cattscamera.com/guia-tipos-de-plano-en-cine/`
- **Escuela de Artesanía:** Análisis de composición y encuadre cinematográfico disponible en `https://escueladeartesania.com/tipos-de-plano/`
- **Casanova Foto:** Guía de planos fotográficos y cinematográficos disponible en `https://casanovafoto.com/tipos-de-plano/`
- **TAIARTS:** Taxonomía detallada de planos de cámara disponible en `https://taiarts.com/planos-de-camara/`
- **Wikipedia - Plano cinematográfico:** Referencia general sobre tipos de plano y su función narrativa disponible en `https://es.wikipedia.org/wiki/Plano_cinematogrfico`

1.7.2 Declaración de Originalidad

Todo el código de aplicación (clasificación de planos, reconocimiento de gestos, sistema de encuadre, visualización, funcionalidades de control avanzado) fue desarrollado completamente por los autores de este trabajo sin utilizar código de terceros más allá de las bibliotecas estándar mencionadas. No se utilizaron tutoriales de código completos ni implementaciones existentes de sistemas similares.

Las decisiones de diseño, algoritmos de clasificación, umbrales de detección, parámetros de encuadre, y funcionalidades como el modo Hold, sistema de screenshots y toggle de grid fueron determinados mediante experimentación propia basada en comprensión de principios teóricos, no por copia de implementaciones existentes.

1.8 Diario de Reuniones del Grupo

1.8.1 Reunión 1 - 18 de noviembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 2 horas

Agenda: Brainstorming inicial del proyecto, definición del alcance, exploración de tecnologías candidatas.

Decisiones tomadas: Acordamos enfocarnos en un sistema de cámara inteligente después de considerar varias propuestas alternativas. La motivación principal fue combinar nuestro interés en visión por computador con aplicación práctica directa. Decidimos investigar MediaPipe como tecnología base después de descartar opciones que requerirían entrenamiento extensivo de modelos personalizados.

Tareas asignadas: Miguel Ángel - investigar capacidades de MediaPipe Pose y preparar prototipos de detección básica. Alberto José - investigar teoría cinematográfica y compilar taxonomía de tipos de plano con sus características.

1.8.2 Reunión 2 - 27 de noviembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 3 horas

Agenda: Revisión de prototipos iniciales, integración de MediaPipe Hands, discusión de arquitectura del sistema.

Decisiones tomadas: Confirmamos que MediaPipe proporciona detección suficientemente robusta para nuestros propósitos. Decidimos implementar sistema dual de control automático + manual mediante gestos. Definimos arquitectura modular con separación clara entre detección, clasificación, y encuadre.

Desafíos identificados: La detección de landmarks tiene jitter notable; necesitamos implementar suavizado temporal. Los cambios de plano abruptos son visualmente desagradables; necesitamos sistema de transiciones.

Tareas asignadas: Miguel Ángel - implementar módulo de clasificación automática de planos basado en métricas geométricas. Alberto José - implementar sistema de reconocimiento de gestos manuales y mapeo a planos.

1.8.3 Reunión 3 - 2 de diciembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 4 horas (sesión de integración)

Agenda: Integración de módulos desarrollados independientemente, pruebas conjuntas, ajuste de parámetros.

Decisiones tomadas: Integramos exitosamente clasificación automática y control manual en un sistema unificado. Implementamos toggle entre modos mediante tecla 'm'. Ajustamos umbrales de shoulder_width después de pruebas extensivas con diferentes distancias.

Problemas resueltos: Inicialmente el clasificador automático era demasiado sensible, cambiando de plano constantemente. Implementamos sistema de suavizado temporal con deque que resolvió el problema. Los gestos manuales generaban falsos positivos; incrementamos umbrales de confianza de MediaPipe Hands.

Tareas asignadas: Miguel Ángel - implementar sistema de encuadre con transiciones suaves. Alberto José - desarrollar sistema de visualización multi-ventana.

1.8.4 Reunión 4 - 10 de diciembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 2.5 horas

Agenda: Revisión de implementación de encuadre, optimización de rendimiento, pruebas de sistema completo.

Decisiones tomadas: El sistema de interpolación exponencial con factor 0.15 proporciona transiciones visualmente agradables. Implementamos cálculo adaptativo del punto de seguimiento según tipo de plano. Añadimos offsets verticales específicos para cada plano.

Desafíos identificados: El rendimiento en hardware menos potente era insuficiente. Identificamos que llamadas redundantes a funciones de MediaPipe estaban degradando performance.

Optimizaciones implementadas: Reducimos resolución de captura a 1280x720. Optimizamos el bucle principal para minimizar conversiones de espacio de color redundantes. Utilizamos configuración de MediaPipe con model_complexity=1 en lugar de 2.

Tareas asignadas: Miguel Ángel - implementar soporte multi-cámara. Alberto José - refinar panel de control y añadir información de debug.

1.8.5 Reunión 5 - 17 de diciembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 3 horas

Agenda: Integración de capacidades multi-cámara, detección de orientación corporal, planos especiales.

Decisiones tomadas: Implementamos detección de orientación mediante análisis de coordenadas Z de landmarks. Añadimos planos especiales: Vista de Espaldas, Sobre el Hombro, Picado, Contrapicado. Cada cámara mantiene estado de encuadre independiente.

Resultados positivos: La detección de orientación funciona sorprendentemente bien usando simplemente comparación de Z entre nariz y hombros. Los planos especiales añaden variedad significativa sin comprometer rendimiento.

Tareas asignadas: Ambos - pruebas exhaustivas con diferentes escenarios, preparación de material de presentación.

1.8.6 Reunión 6 - 27 de diciembre de 2025

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 2.5 horas

Agenda: Implementación de funcionalidades avanzadas de control, pulido final del sistema.

Decisiones tomadas: Implementamos el modo Hold (bloqueo de plano) mediante la tecla 'h', que permite congelar el encuadre actual independientemente de movimientos o detecciones. Añadimos sistema de captura de screenshots con la tecla 's' que guarda automáticamente en carpeta screenshots/ con timestamp y nombre de plano. Implementamos toggle de regla

de tercios con la tecla 'g' para activar/desactivar la visualización del grid en la ventana de resultado.

Refinamientos implementados: Actualizamos el panel de control para mostrar claramente el estado de Hold Lock y Grid (ON/OFF) con código de colores. Modificamos la ventana RESULTADO para presentar un frame completamente limpio sin overlays de texto, mostrando solo el grid opcional. Reemplazamos el gesto "pulgar arriba"por el gesto rock and roll"(índice y meñique extendidos) para el Primerísimo Plano, resultando más distintivo y memorable.

Pruebas realizadas: Verificamos que el modo Hold funciona correctamente en ambos modos (AUTO y MANUAL), impidiendo cambios de plano hasta que se desactiva. Probamos el sistema de screenshots capturando múltiples imágenes y verificando que los archivos se guardan con nombres correctos y metadata apropiada. Confirmamos que el toggle de grid funciona fluidamente sin afectar el rendimiento.

Tareas asignadas: Miguel Ángel - pulir interfaz visual del panel de control y asegurar que toda la información relevante esté visible. Alberto José - verificar robustez del reconocimiento del nuevo gesto rock and roll en diferentes condiciones.

1.8.7 Reunión 7 - 4 de enero de 2026

Asistentes: Miguel Ángel Rodríguez Ruano, Alberto José Rodríguez Ruano

Duración: 3 horas

Agenda: Revisión final de documentación, preparación de material de entrega, ensayo de presentación.

Actividades realizadas: Revisión completa de la memoria técnica actualizando todas las secciones para reflejar las funcionalidades finales implementadas (modo Hold, screenshots, toggle de grid, gesto rock and roll). Verificación de que todos los componentes del proyecto están correctamente documentados. Preparación del material audiovisual de demostración mostrando las diferentes funcionalidades en acción. Ensayo de la presentación oral con cronometraje.

Ajustes finales al código: Limpiamos código eliminando funciones de debug no utilizadas. Añadimos comentarios explicativos en secciones críticas. Verificamos que todas las teclas de control están documentadas en el panel de control y en la función print_resumen_planos(). Confirmamos que el sistema inicia con configuración por defecto apropiada (modo AUTO, grid activado, hold desactivado).

Reflexiones finales: El proyecto cumplió todos los objetivos planteados inicialmente y añadió funcionalidades adicionales que mejoran significativamente la usabilidad práctica del sistema. La colaboración entre ambos miembros del equipo fue efectiva, con buena distribución de tareas y comunicación constante. Las decisiones técnicas tomadas demostraron ser acertadas, especialmente la adopción de MediaPipe como base tecnológica y la arquitectura modular del sistema.

Lecciones aprendidas: La importancia de iteración rápida y pruebas continuas con usuarios reales. El valor de separar claramente detección (MediaPipe) de lógica de aplicación (nuestro código). La necesidad de suavizado temporal en cualquier sistema de visión por computador que interactúe con usuarios en tiempo real. La importancia de funcionalidades aparentemente simples (como el modo Hold) que tienen impacto desproporcionado en usabilidad práctica. El valor de una interfaz visual clara que muestre el estado del sistema de manera inequívoca.

1.9 Créditos Materiales no Originales del Grupo

Este proyecto utiliza exclusivamente tecnologías de código abierto y recursos educativos de acceso público. A continuación detallamos todos los materiales de terceros utilizados:

1.9.1 Recursos Educativos y Documentación

- **Documentación oficial de MediaPipe:** Guías oficiales y tutoriales disponibles en `https://ai.google.dev/edge/mediapipe/` fueron fundamentales para comprender las capacidades y limitaciones de las soluciones de detección utilizadas.
- **Aprender Cine:** Taxonomía de tipos de plano cinematográfico disponible en `https://aprendercine.com/tipos-de-plano/`
- **Catt's Camera Blog:** Guía completa de planos en cine disponible en `https://blog.cattscamera.com/guia-tipos-de-plano-en-cine/`
- **Escuela de Artesanía:** Análisis de composición y encuadre cinematográfico disponible en `https://escueladeartesania.com/tipos-de-plano/`
- **Casanova Foto:** Guía de planos fotográficos y cinematográficos disponible en `https://casanovafoto.com/tipos-de-plano/`
- **TAIARTS:** Taxonomía detallada de planos de cámara disponible en `https://taiarts.com/planos-de-camara/`
- **Wikipedia - Plano cinematográfico:** Referencia general sobre tipos de plano y su función narrativa disponible en `https://es.wikipedia.org/wiki/Plano_cinematogrfico`
- **Documentación de OpenCV:** Referencia técnica de funciones de procesamiento de imágenes disponible en `https://docs.opencv.org/`
- **Documentación de NumPy:** Guías de operaciones numéricas y manipulación de arrays disponible en `https://numpy.org/doc/`

- **Documentación de Python:** Referencias oficiales del lenguaje y bibliotecas estándar disponible en `https://docs.python.org/3/`

1.9.2 Declaración de Originalidad

Todo el código de aplicación (clasificación de planos, reconocimiento de gestos, sistema de encuadre, visualización, funcionalidades de control avanzado) fue desarrollado completamente por los autores de este trabajo sin utilizar código de terceros más allá de las bibliotecas estándar mencionadas. No se utilizaron tutoriales de código completos ni implementaciones existentes de sistemas similares.

Las decisiones de diseño, algoritmos de clasificación, umbrales de detección, parámetros de encuadre, y funcionalidades como el modo Hold, sistema de screenshots y toggle de grid fueron determinados mediante experimentación propia basada en comprensión de principios teóricos, no por copia de implementaciones existentes.

El sistema completo, desde la arquitectura general hasta los detalles de implementación, representa trabajo original que aplica creativamente tecnologías existentes (MediaPipe, OpenCV, NumPy) para resolver un problema novedoso en el dominio de producción audiovisual automatizada.

2 Uso de Herramientas de Inteligencia Artificial

Durante el desarrollo de este proyecto se utilizaron herramientas de inteligencia artificial para optimizar aspectos específicos del trabajo. A continuación se documenta de manera transparente su uso:

2.1 Herramientas Utilizadas

2.1.1 Asistencia en Redacción

Se utilizó **ChatGPT** para:

- Revisión y mejora de redacción de la memoria técnica para claridad y coherencia
- Estructuración de secciones complejas y organización del contenido

Todos los conceptos técnicos, argumentaciones y razonamientos fueron originales de los autores. La IA solo asistió en la expresión escrita.

2.1.2 Depuración y Optimización de Código

Se utilizaron **GitHub Copilot** y **ChatGPT** para:

- Identificación de errores lógicos y optimizaciones de rendimiento
- Sugerencias de refactorización del código Python
- Generación de plantillas para funciones repetitivas (detección de gestos)
- Creación de comentarios y documentación del código

Todos los algoritmos principales (clasificación de planos, encuadre, transiciones suaves, detección de orientación) fueron diseñados y desarrollados completamente por los autores.

2.1.3 Generación de Contenido Visual

Se utilizó **ChatGPT** para:

- Generación de imágenes conceptuales e ilustrativas para la presentación
- Creación de diagramas visuales de arquitectura del sistema
- Diseño de gráficos comparativos y material de apoyo visual

2.1.4 Video Promocional

Se utilizó **HeyGen** para:

- Creación del video de pitch para inversores/clientes (2 minutos)
- Generación automática de locución y avatares virtuales
- Edición y montaje automatizado del video promocional

El guion, estructura narrativa y contenido del video fueron diseñados por los autores.

2.2 Aspectos NO Generados por IA

Los siguientes componentes fueron completamente originales y desarrollados sin asistencia de IA:

- Toda la lógica de programación y algoritmos del sistema
- Decisiones arquitectónicas y diseño modular
- Cálculos matemáticos (shoulder_width, interpolación exponencial, orientación 3D)
- Integración de MediaPipe, OpenCV y NumPy con ajuste de parámetros
- Conceptualización completa del proyecto y su enfoque
- Pruebas, validación y experimentación del sistema
- Implementación de funcionalidades (modo Hold, screenshots, toggle de grid)

2.3 Uso Responsable

El equipo utilizó herramientas de IA como apoyo para: - Mejorar la calidad de documentación y presentación - Acelerar tareas repetitivas de edición y formato - Validar código mediante análisis automatizado

Sin embargo, el núcleo técnico del proyecto —visión por computador, algoritmos cinematográficos, arquitectura del sistema— es completamente original y representa el verdadero trabajo de los autores.