

# Herramienta de apoyo en el diagnóstico motriz y rehabilitación en veterinaria

Proyecto Fin de Carrera

27/06/2016

Universidad de Las Palmas de Gran Canaria

Deriman Franco García



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Escuela de Ingeniería Informática



# Proyecto Fin de Carrera

**Título:** Herramienta de apoyo en el diagnóstico motriz y rehabilitación en veterinaria

**Apellidos y nombre del alumno:** Franco García, Deriman

**Tutor:** Quesada Arencibia, Alexis

**Cotutor:** Vilar Guereño, José Manuel

### *Agradecimientos*

La realización de este proyecto supone el final de una etapa en mi carrera universitaria y no hubiera podido llegar hasta aquí sin la ayuda de algunas personas:

En primer lugar agradecer a mis tutores Alexis y Jose Manuel, la realización de este proyecto sencillamente no habría sido posible sin su guía, ayuda y consejo.

A continuación, agradecer a mi familia que me ha apoyado durante todos estos años, no solamente durante la realización del proyecto, sino durante toda la carrera.

A mis amigos, que me han ayudado habiendo problemas o no, siempre preparados para echar una mano.

Y a la Universidad de Las Palmas de Gran Canaria y todo su personal universitario, en especial a los departamentos de Informática y Veterinaria, a quienes respeto por todo el trabajo que realizan a diario.

## Índice

1.	Introducción .....	9
1.1.	Prefacio .....	9
1.2.	Estructura del documento .....	10
1.3.	Objetivos.....	12
2.	Estado del arte.....	13
3.	Metodología .....	17
3.1.	Proceso de desarrollo iterativo e incremental .....	17
3.2.	Ventajas del desarrollo iterativo .....	19
3.3.	Debilidades de este modelo de desarrollo .....	19
4.	Planificación temporal.....	21
4.1.	Introducción.....	21
4.2.	Plan de trabajo.....	21
4.2.1.	Planificación temporal estimada .....	21
4.2.2.	Temporización de Gantt .....	23
4.2.3.	Dedicación real .....	24
4.2.4.	Tabla comparativa de resultados .....	26
4.2.5.	Conclusiones.....	27
5.	Estudio de tecnologías y herramientas .....	28
5.1.	Introducción.....	28
5.2.	Listado de herramientas.....	28
5.3.	WxWidgets.....	31
5.3.1.	Estructura de una aplicación .....	32
5.4.	OpenCV .....	33
5.4.1.	Media I/O.....	33
5.4.2.	Image processing.....	35

5.4.3. Video.....	36
5.4.4. High-level GUI.....	37
6. Introducción al flujo óptico .....	38
6.1. Calculo del flujo óptico basado en el gradiente .....	38
6.2. Soluciones al problema de apertura.....	39
6.3. Algoritmo de Lucas-Kanade .....	40
6.4. Aplicación del flujo óptico .....	41
7. Análisis.....	43
7.1. Definición del dominio del problema .....	43
7.1.1. Determinación del alcance del dominio.....	43
7.1.2. Identificación del Entorno Tecnológico.....	43
7.1.3. Identificación de Usuarios Participantes .....	43
7.1.4. Detalles y normas a tener en cuenta.....	43
7.2. Establecimiento de Requisitos.....	44
7.2.1. Requisitos de usuario .....	44
7.2.2. Especificación de requisitos del software .....	45
7.3. Casos de uso .....	47
7.3.1. Lista de Actores .....	47
7.3.2. Listado de casos de uso de nivel general .....	47
7.3.3. Diagramas de casos de uso.....	48
8. Desarrollo iterativo.....	52
8.1. Filtrado del parpadeo .....	52
8.2. Calibrado de Shi-Tomasi y Lucas Kanade.....	52
8.2.1. Algoritmo de Shi-Tomasi .....	53
8.2.2. Algoritmo de Lucas-Kanade.....	53
8.3. Primera Iteración .....	54

8.3.1.	El diseño de la Interfaz .....	54
8.3.2.	La ejecución multihilo.....	55
8.3.3.	La estructura de almacenamiento.....	57
8.3.4.	Pruebas .....	58
8.4.	Segunda Iteración .....	61
8.4.1.	Actualización de la Interfaz .....	61
8.4.2.	Gráfica.....	61
8.4.3.	Exportación de datos.....	62
8.4.4.	La barra de búsqueda .....	62
8.4.5.	El ángulo conjugado .....	63
8.4.6.	Pruebas .....	63
8.5.	Tercera iteración.....	66
8.5.1.	Actualización de la Interfaz .....	66
8.5.2.	El cuaderno de notas .....	67
8.5.3.	Nuevas herramientas .....	67
8.5.4.	Actualización de la gráfica .....	68
8.5.5.	La estructura de almacenamiento.....	68
8.5.6.	Pruebas .....	69
8.6.	Componentes de la aplicación.....	72
8.7.	Técnicas de pruebas .....	73
9.	Resultados y conclusiones.....	75
9.1.	Resultados obtenidos .....	75
9.2.	Conclusiones .....	76
10.	Trabajo Futuro .....	77
11.	Bibliografía.....	78
12.	Anexo I.....	80

12.1.	Manual de instalación del proyecto .....	80
12.1.1.	Instalación de java y eclipse .....	80
12.1.2.	Instalación de TortoiseSVN y Subversion .....	80
12.1.3.	Instalación de wxWidgets.....	81
12.1.4.	Instalación de OpenCV .....	81
13.	Anexo II.....	82
13.1.	Manual de usuario .....	82
13.1.1.	Guía de instalación de la aplicación .....	82
13.1.2.	Reproducir archivo de video.....	82
13.1.3.	Convertir secuencia de frames en video .....	83
13.1.4.	Calcular ángulos durante la reproducción.....	84
13.1.5.	Calcular distancias durante la reproducción .....	84
14.	Anexo III.....	88
14.1.	Listado de Casos de Uso completos.....	88

## Índice de Figuras

Figura 2.1 Tabla comparativa de primera categoría .....	15
Figura 2.2 Tabla comparativa de segunda categoría .....	16
Figura 2.3 Tabla comparativa de tercera y cuarta categoría .....	16
Figura 4.1 Tabla de temporización de Gantt.....	24
Figura 4.2 Tabla Comparativa de Resultados.....	26
Figura 4.3 Planificación estimada.....	26
Figura 4.4 Dedicación real.....	26
Figura 5.1 Ejemplo de apertura de un objeto con VideoCapture .....	34
Figura 5.2 Ejemplo de lectura y escritura de atributos de un objeto VideoCapture	34
Figura 5.3 Ejemplo de captura de un fotograma .....	34
Figura 5.4 Ejemplo de creación de un objeto VideoWriter .....	34
Figura 5.5 Ejemplo de uso de un objeto VideoWriter.....	34
Figura 5.6 Cabecera de la función Shi-Tomasi .....	35
Figura 5.7 Cabecera de la función Lucas-Kanade.....	36
Figura 5.8 Ejemplo de uso del módulo High-Level GUI.....	37
Figura 6.1 Algoritmo piramidal de Lucas Kanade.....	40
Figura 6.2 Algoritmo Shi-Tomasi aplicado a un fotograma.....	41
Figura 6.3 Algoritmo de Lucas-Kanade aplicado al siguiente fotograma.....	42
Figura 7.1 Modelo de dominio.....	45
Figura 7.2 Modelo Conceptual.....	46
Figura 7.3 Listado de casos de uso de nivel general .....	47
Figura 7.4 Caso de uso de apertura .....	48
Figura 7.5 Caso de uso de manejo de la reproducción .....	49
Figura 7.6 Caso de uso de manejo de marcadores .....	50
Figura 7.7 Caso de uso del manejo de notas .....	50

Figura 7.8 Caso de uso de manejo de la gráfica.....	51
Figura 7.9 Caso de uso de guardar y salir.....	51
Figura 8.1 Esquema de desarrollo por iteraciones .....	52
Figura 8.2 Dependencias en el diseño del patrón Fachada. ....	54
Figura 8.3 Diseño de la interfaz en la primera iteración.....	55
Figura 8.4 Esquema de ejecución del hilo secundario .....	56
Figura 8.5 Modelo de diseño de datos para los puntos.....	57
Figura 8.6 Implementación de la estructura valor3puntos .....	57
Figura 8.7 Falso positivo debido a una iluminación pobre .....	59
Figura 8.8 Captura de la interfaz en la primera iteración .....	60
Figura 8.9 Diseño de la interfaz en la segunda iteración .....	61
Figura 8.10 Ángulos conjugados .....	63
Figura 8.11 Datos originales y exportados a Excel y CSV .....	65
Figura 8.12 Captura de la interfaz en la segunda iteración .....	65
Figura 8.13 Diseño de la interfaz en la tercera iteración .....	66
Figura 8.14 Botones para medir las distancias .....	68
Figura 8.15 Botones para medir segmentos de tiempo.....	68
Figura 8.16 Modificación en la implementación de valor3puntos .....	68
Figura 8.17 Falso positivo debido a la iluminación .....	69
Figura 8.18 Visualización de distancia en el gráfico.....	70
Figura 8.19 Captura de la aplicación en la tercera iteración .....	71

## 1. INTRODUCCIÓN

### 1.1. PREFACIO

Vivimos en tiempos interesantes, una época en la que con pulsar un botón en nuestro teléfono móvil podemos ponernos en contacto con un amigo al otro lado del mundo, localizar nuestra ubicación en un mapa si nos perdemos, o recomendarnos un buen restaurante cercano si tenemos hambre. Una época en la que los hospitales trabajan a diario usando tecnología tan avanzada que es capaz de visualizar el interior del cerebro de una persona en tiempo real, o realizar impresiones tridimensionales del feto de una mujer embarazada.

En el campo de la veterinaria clínica actual, el diagnóstico y evaluación del movimiento en los animales para detectar de forma prematura enfermedades tales como la displasia de cadera, o para comprobar cómo se desarrolla su rehabilitación en caso de accidente o herida, se realiza manualmente con fotografías tomadas con una cámara de alta velocidad y su posterior análisis, sin ninguna ayuda informatizada.

Nos encontramos en un punto de la historia en el que las ciencias de la salud han de ir necesariamente de la mano de las ciencias computacionales para dar con la información más precisa posible y, en consecuencia, hacer el mejor uso de los conocimientos médicos y veterinarios por la salud de las personas.

Para mejorar estas funcionalidades, se llevará a cabo en este proyecto un estudio de la aplicación de técnicas de visión por computador en el campo del reconocimiento de movimiento para automatizar la generación de datos relevantes a la hora del diagnóstico.

Durante este estudio se desarrollará una aplicación como solución informática que permita la captación, análisis y representación de la información obtenida.

## 1.2. ESTRUCTURA DEL DOCUMENTO

El documento se divide en 12 partes.

La primera parte es la **introducción**. La introducción contiene información preliminar sobre el contenido del proyecto, su estructura y sus objetivos. Leyendo esta introducción el lector puede hacerse una idea de lo que va a encontrar en el resto del documento.

En la segunda parte se detalla el **estado del arte**, donde se comenta y compara una selección de las tecnologías actuales más relevantes y que mejor reúnen las condiciones necesarias para la consecución de los objetivos del proyecto.

La tercera parte trata sobre las **metodologías** de desarrollo usadas en el proyecto, en qué consisten, sus ventajas y sus debilidades.

La cuarta parte abarca la **planificación temporal** del proyecto, donde se describe la organización de sus actividades e hitos y actúa como indicador del desarrollo del trabajo.

La quinta parte consiste en el **estudio de tecnologías y herramientas**, en ella se describe qué herramientas y tecnologías se utilizarán en la elaboración de la solución.

La sexta parte es una **introducción al flujo óptico**, en la cual se explica brevemente los conceptos más importantes sobre las técnicas de flujo óptico y su impacto en el desarrollo de la solución.

La séptima parte es el **análisis**, donde se define el **dominio del problema**, que explica el problema a solucionar y su entorno; los **requisitos de usuario**, a partir de los cuales se desarrollará una terminología común que ayudará en la elaboración posterior de los **casos de uso**, los cuales afectarán directamente el diseño y desarrollo de la solución.

La octava parte trata del **desarrollo iterativo** de la aplicación. Este capítulo está dividido en las diferentes etapas del desarrollo de la aplicación y sus respectivas fases de diseño, implementación y pruebas, siguiendo el método iterativo e incremental.

La novena parte son las **conclusiones** finales a las que se ha llegado tras completar y evaluar el desarrollo del proyecto.

La décima parte es una sección dedicada a plantear posibles líneas de **trabajo futuro**.

La decimoprimer parte es la recopilación **bibliográfica** del proyecto. Desde libros y páginas web consultadas, hasta la documentación de las librerías software utilizadas.

La duodécima parte es el **Anexo** que contiene los manuales del proyecto y el listado de casos de uso completos.

Esta memoria se acompaña de un CD que contiene:

- Esta memoria en formato digital.
- Todo el código del proyecto necesario para su ejecución.

### 1.3. OBJETIVOS

El objetivo principal es obtener una herramienta de trabajo que incluya un sistema de detección de movimiento en una secuencia de video y que sea capaz de realizar mediciones automáticas sobre los datos que se observen o se marquen en el reproductor. Para ello deben abordarse diversos objetivos intermedios:

- Estudio de viabilidad del proyecto utilizando como base la toma de video recogida de una cámara Fastec Imaging, y de forma secundaria la toma de video de otras fuentes.
- Obtención de experiencia en el campo de la visión por computador, en particular para la detección de movimiento en secuencias de video.
- Estudio de las distintas librerías y herramientas existentes sobre el punto anterior para poder aplicar las elegidas en la aplicación final.
- Estudio de las distintas librerías y herramientas existentes para visualización y exposición de los datos obtenidos al usuario final.

El aprendizaje en técnicas de detección de movimiento para esta aplicación tiene como principal función el reconocimiento y seguimiento de determinados puntos marcados directamente sobre el pelaje del animal como preparación previa a la grabación, de forma que se agilice el procesado de videos en tiempo real.

Aunque existen diferentes técnicas de detección de movimiento, desde la sustracción de fondos (Background Substraction), al emparejamiento de bloques (Block Matching); este estudio se centrará en el uso de las técnicas de detección por Flujo Óptico (Optical Flow) a la hora de abordar el problema por ser una solución que proporciona un equilibrio entre dos parámetros fundamentales en la solución del problema: buena precisión y bajo coste computacional.

Como objetivo secundario se pretende que la aplicación sea multiplataforma para su uso tanto en Windows, como en Linux y Mac.

## 2. ESTADO DEL ARTE

Dentro del marco tecnológico del estado del arte se encuentran diversas aplicaciones que reúnen las condiciones que busca el usuario final. Puesto que cada una de ellas puede usarse a distintos niveles, se ha separado en cuatro grupos para posteriormente compararlas adecuadamente.

Por un lado tenemos aplicaciones que realizan la detección de movimiento de determinados puntos de interés en una secuencia de video, y que disponen de las herramientas de análisis necesarias para su uso. Ejemplos claros de este caso son OnTrack Equine, UCOTrack, Simi y el Ariel Performance Analysis System (APAS). Suele ocurrir que en estos casos el sistema en sí se comercializa tanto con las cámaras como el software necesario para su uso.

En un segundo bloque encontramos aplicaciones que superponen visualmente herramientas de análisis sobre una capa de reproducción de video. Un ejemplo de este tipo de herramientas es el Tracker, que se utiliza para impartir física mecánica; el Metric, que se utiliza en la industria para realizar mediciones de precisión; y herramientas especializadas para deportistas como el SkillSpector, el Kinovea, y el Dartfish.

En tercer lugar tenemos dos herramientas de análisis que se especializan en el análisis cinético tridimensional, estas herramientas son el Visual3D de C-Motion y la herramienta gratuita de código abierto Mokka.

Finalmente tenemos aplicaciones que se usan en la captura de movimiento, pero no realizan análisis alguno. Se trata de herramientas potentes, como Vicon y OptiTrack, que podrían usarse en conjunto con Visual3D o Mokka para realizar análisis.

Veamos a continuación una breve descripción de cada una de ellas:

- **OnTrack Equine de Lameness Solutions[07]** . Se trata de una aplicación destinada a la industria equina diseñado como herramienta de trabajo para la práctica veterinaria, fisioterapia, la enseñanza y para entrenamiento tanto del jinete como del caballo.
- **UCOTrack ©[08]**. Es un sistema óptico de captura y análisis de movimiento en 3D desarrollado en la universidad de Córdoba para su uso en cualquier investigación de índole biomecánica. Especialmente en el área clínica, deportiva, veterinaria y de I+D+I.
- **Simi[09]**. Simi es una empresa que se dedica a elaborar sistemas de análisis y captura de movimiento de gama alta. Sus servicios se prestan principalmente a campos tales como la medicina, deportes e industria, teniendo una línea de productos lo suficientemente amplia como para ajustarse a diversos presupuestos dependiendo de la complejidad y precisión de los datos.
- **Ariel Performance Analysis System (APAS)[10]**. Se trata de un sistema de análisis de movimiento en 3D. Es capaz de grabar desde varias cámaras simultáneamente y realizar el análisis biomecánico automáticamente utilizando toda la información entrante, ya sea video, electromiografías y plataformas de fuerza.

- **Tracker. Video Analysis and Modeling Tool[11].** Se trata de una herramienta gratuita para el modelado y análisis de video centrada en la enseñanza de las asignaturas de Física.
- **Metric. Video measurement technology for the PC[12].** Se trata de una herramienta destinada a las compañías industriales para realizar repeticiones de toda clase de medidas (ángulos, distancias, radio de un círculo, etc.) automáticamente sobre un video en vivo, generalmente a través de un microscopio. De esta forma se automatiza la comprobación de una producción en serie.
- **SkillSpector[13].** SkillSpector de video4coach es una herramienta gratuita de análisis de habilidad y movimiento para Windows. SkillSpector permite el análisis en 2 y 3 dimensiones, calculo automático de la inercia y el análisis avanzado de datos cinemáticos tanto lineales como angulares.
- **Kinovea[14].** Se trata de un reproductor de video para entusiastas del deporte. Con él se puede observar (añadiendo flechas y descripciones al video), medir (tanto distancias como ángulos y tiempos, pudiendo además seguir los puntos y cálculos), comparar (visualizar dos videos lado a lado sincronizados por un mismo evento) y exportar la información obtenida en el video.
- **Dartfish[15].** O más concretamente, la aplicación Dartfish Express para iPhone y iPad. Esta aplicación permite analizar imágenes y videos, añadiendo comentarios y dibujos para remarcar zonas de la pantalla.
- **Visual3D de C-Motion[16] (Qualisys[17]).** Se trata de un software para el análisis y modelado biomecánico en 3D disponible para investigadores y médicos clínicos que estudian rehabilitación, neurociencia, ingeniería, veterinaria, deportes, y muchos más. El software es capaz de procesar datos procedentes de cualquier sistema de captura de movimientos siempre que esté en formato C3D estándar o se reproduzca en tiempo real.
- **Mokka. Motion kinematic & kinetic analyzer[18].** Se trata de una herramienta multiplataforma y de código abierto para analizar fácilmente datos biomecánicos. Mokka es capaz de leer y escribir en archivos C3D, además de visualizar en 2D y 3D las trayectorias de marcadores, plataformas de fuerza, segmentos, así como los ángulos de las articulaciones, fuerza, momento y señales analógicas como los EMG.
- **Vicon, de Vicon Motion Systems Ltd[19].** Se trata de un Sistema de captura de movimiento tanto para la industria del entretenimiento, como en ingeniería, o en ciencias de la salud.
- **OptiTrack, de NaturalPoint[20].** Al igual que Vicon, OptiTrack es un sistema de captura de movimiento que puede usarse tanto en simulaciones de realidad virtual, como en robótica, animación, y para el análisis biomecánico en diversos campos de ciencias de la salud.

**Tabla comparativa de la primera categoría:**

	OnTrack Equine	UCOTrack	Simi	APAS
Detección de movimiento en 2D	Sí	Sí	Sí	Sí
Detección de movimiento en 3D	No	Sí	Sí	Sí
Herramientas de análisis	Sí	Sí	Sí	Sí
¿A quién va dirigido?	Industria equina	Investigación clínica, Deportiva, Veterinaria, I+D+I	Medicina, deportes, industria, animación, investigación celular	Deportistas, entorno académico y de investigación
Compatibilidad	iPad y iPhone	Consultar	Windows	Windows
Precio	9.99€	A consultar*	1,000€ a 150,000€ *	6,300€ a 12,600€

\* Varía según la complejidad del producto. Desde simplemente el software para realizar el análisis hasta comprar un laboratorio completo con cámaras y el resto del equipo.

Figura 2.1 Tabla comparativa de primera categoría

**Tabla comparativa de la segunda categoría:**

	Metric	Kinovea	Tracker	SkillSpector	Dartfish Express
Detección de movimiento en 2D	No	Sí	Sí	No	No
Detección de movimiento en 3D	No	No	No	No	No

Herramientas de análisis	Sí	Sí	Sí	Sí	Sí
¿A quién va dirigido?	Industria	Deportistas	Enseñanza de Física	Deportistas	Deportistas
Compatibilidad	Windows	Windows	Windows/ Mac OS X/ Linux	Windows	iOS
Precio	250€ a 1,650€	Gratuito	Gratuito	Gratuito	6.99€

Figura 2.2 Tabla comparativa de segunda categoría

### Tabla comparativa de la tercera y cuarta categoría:

	Mokka	Visual3D	Vicon	OptiTrack
Detección de movimiento en 2D	No	No	No	No
Detección de movimiento en 3D	No	No	Sí	Sí
Herramientas de análisis	Sí	Sí	No	No
¿A quién va dirigido?	Investigación	Investigación clínica, Deportiva, Veterinaria, ingeniería.	Industria del entretenimiento, ingeniería, ciencias de la salud	Industria del entretenimiento, ingeniería, ciencias de la salud
Compatibilidad	Windows/ Mac OS X/ Linux	Windows	Windows	Windows
Precio	Gratuito	663€ (estudiantes) 11,500€ (4 licencias)	A consultar	A consultar

Figura 2.3 Tabla comparativa de tercera y cuarta categoría

### 3. METODOLOGÍA

#### 3.1. PROCESO DE DESARROLLO ITERATIVO E INCREMENTAL

En el campo del desarrollo de software, se conoce como metodología al conjunto de conceptos, prácticas y criterios que se usan para estructurar, planear y controlar el proceso de desarrollo para lograr un producto bueno y de calidad.

Estas metodologías comienzan a surgir a finales de la década de los 60 con el propósito de redefinir y describir el ciclo de vida del software, es decir, el ciclo por el cual el desarrollo del software pasa desde su definición, a su diseño, implementación, verificación, validación, entrega y acaba con su operación y mantenimiento. A estas metodologías se las conoce como metodologías tradicionales, y su principal característica es la forma en la que, de manera estructurada y metódica, reiteran en cada una de las etapas del ciclo de vida.

Para la realización de este proyecto se ha optado por utilizar el **proceso de desarrollo iterativo e incremental**, uno de los modelos tradicionales que nace para tratar de resolver los problemas del modelo de desarrollo en cascada aplicando ideas del modelo de prototipos.

Mientras que la metodología del modelo en cascada es fácil de implementar y entender, promoviendo una filosofía de trabajo efectiva que consiste en definir antes de diseñar, y diseñar antes de codificar. Tiene el defecto de aumentar de manera inusitada los costos de desarrollo cuando se encuentran problemas de diseño durante las fases posteriores de desarrollo, obligando a rediseñar y volver a implementar los cambios.

Lo que hace el proceso de desarrollo iterativo e incremental es tomar los aspectos positivos del modelo en cascada y aplicarlos de forma iterada a un desarrollo escalado del producto. Esto significa que el resultado de las pruebas de cada iteración vuelve a ser analizado y contrastado con los requisitos de la aplicación, y se prepara para implementar nuevas funcionalidades durante la siguiente iteración. En definitiva, las iteraciones irán refinando el sistema hasta dar con un producto que satisfaga al cliente.

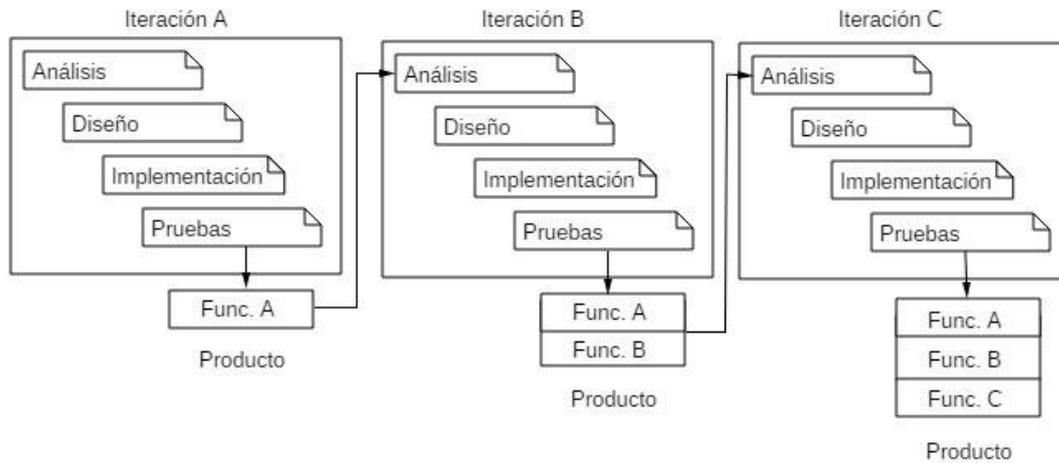


Figura 3.1 Ciclo de vida en el proceso de desarrollo iterativo e incremental

Este modelo favorece el desarrollo cuando se da una situación en la que o bien el usuario no sabe bien lo que necesita para satisfacer sus necesidades, o bien cuando los procesos tienden a cambiar durante el desarrollo.

El proceso en sí mismo consiste en una etapa de inicialización, una etapa de iteración y una lista de control de proyecto.

En la **etapa de inicialización** se crea una versión del producto con el que el usuario pueda interactuar, de esta forma puede comentar y criticar el producto durante su desarrollo. Este producto debe contener una muestra de los problemas y proveer una solución lo suficientemente simple para ser comprendida e implementada rápidamente. Para guiar el proceso de iteración se crea durante esta etapa una **lista de control de proyecto** que contiene un historial de todas las tareas a realizar, esto incluye funcionalidades a implementar y áreas de rediseño de la solución ya existente. Esta lista de control se revisa periódicamente durante la fase de análisis.

Durante la **etapa de iteración** se analiza, diseña, implementa y prueba una tarea de la lista de control de proyecto. Estos cuatro pasos conforman las fases por las que pasa el proceso de desarrollo. Durante la primera etapa de iteración por la que pasa el proceso, el producto implementado durante la etapa de inicialización se procesa para su análisis y determinar si son necesarios cambios en su diseño e implementación. En posteriores iteraciones, el producto sometido al proceso es siempre la última versión resultante de la iteración anterior.

La **fase de análisis** de la iteración se basa en las críticas y comentarios del usuario y en el análisis de las funcionalidades implementadas. Este análisis tiene como objetivo comprobar que se cumplan las metas, e involucra un análisis de la estructura, modularidad, usabilidad, confiabilidad, eficiencia y eficacia. La fase de análisis termina con un plan de especificaciones y requisitos sobre el que se trabajará en fases posteriores.

La **fase de diseño** consiste en el diseño de los componentes del sistema que dan respuesta a las funcionalidades descritas en el plan de especificaciones y requisitos. Generalmente se realiza en base a diagramas que permitan describir la arquitectura del software, las interacciones entre las estructuras, la representación de su interfaz y sus algoritmos.

La **fase de implementación** lleva a cabo la codificación planeada en la fase de diseño, depurando y solventando cualquier problema que pueda surgir en esta fase.

Para facilitar el proceso de rediseño de la etapa o de una tarea añadida a la lista de control, el diseño e implementación de la iteración ha de ser simple, directo y modular.

La **fase de pruebas** se corresponde con el testeo del producto para comprobar que realice correctamente las tareas implementadas. El objetivo de esta fase es identificar posibles fallos de implementación, calidad o usabilidad del sistema, así como presentar el producto final de la iteración al usuario para comprobar que satisface los requisitos propuestos de funcionalidad y usabilidad. Por medio de estas pruebas podemos verificar y revelar la calidad del producto.

### 3.2. VENTAJAS DEL DESARROLLO ITERATIVO

- La crítica y opinión de los usuarios se tiene en cuenta desde las fases más tempranas del proceso.
- Se reduce la complejidad del proyecto al separar y escalar su desarrollo en iteraciones.
- El producto es consistente y puntual en el desarrollo.
- Gracias a la constante realización de pruebas en cada iteración, los productos desarrollados con este modelo tienen una menor probabilidad de fallar.
- Se obtiene un aprendizaje en cada iteración que es aplicado en el desarrollo del producto y aumenta las experiencias para próximos proyectos.

### 3.3. DEBILIDADES DE ESTE MODELO DE DESARROLLO

- Los sistemas creados durante las etapas tempranas del proyecto pueden percibirse como demasiado simples y monótonas a ojos del usuario.
- Debido a que la mayoría de los incrementos se hacen en base a las necesidades de los usuarios, hay que tener en cuenta cómo se maneja el producto para ver si necesita de cambios extras. Este problema no se suele percibir porque la mayoría de las veces los incrementos estipulados antes de la entrega del proyecto satisfacen al usuario.
- La idea de los incrementos es agregar accesorios al sistema funcional, para que este tenga más de una forma de desenvolverse en su tarea; de modo que llenar

excesivamente los incrementos provocaría que se perdiera objetividad de lo que se trata el desarrollo incremental.

- Requiere de un cliente involucrado durante todo el curso del proyecto. Hay clientes que simplemente no estarán dispuestos a invertir el tiempo necesario.
- El trato con el cliente debe basarse en principios éticos y colaboración mutua, más que trabajar cada parte independientemente, defendiendo sólo su propio beneficio.
- La entrega de un programa que es parcial pero funcional puede hacer vulnerable al sistema debido a la falta de robustez en su sistema, provocando que agentes ajenos puedan interferir con el correcto funcionamiento del programa en sí.
- Infunde responsabilidad en el equipo de desarrollo al trabajar directamente con el cliente, requiriendo de profesionales sobre el promedio.
- Sufre fuertes penalizaciones en proyectos en los cuales los requerimientos están previamente definidos, o para proyectos "todo/nada" en los cuales se requiere que se completen en un 100% el producto para ser implementado (por ejemplo, licitaciones) otro punto muy importante es asegurarnos de que el trabajo se pueda cumplir tomando en cuenta los costos que podamos usar en nuestros propios recursos.

## 4. PLANIFICACIÓN TEMPORAL

### 4.1. INTRODUCCIÓN

La planificación temporal de un proyecto software es una actividad que permite la organización del esfuerzo a la hora de identificar, definir y programar las actividades e hitos que se requieren para llevar el proyecto. La planificación temporal es, además, susceptible a cambios debido a la naturaleza evolutiva del mismo. Esto significa que durante la planificación pueden ocurrir tanto adelantos como retrasos en las actividades que pueden desplazar la agenda.

Dada esta definición de planificación temporal, tenemos que en ella:

Se definen todas las tareas del proyecto, asignando tiempos y recursos a las mismas, así como los hitos que se alcanzan una vez la documentación producida como parte de la tarea ha sido revisada correctamente.

Se definen cuáles son las tareas críticas entre todas las tareas del proyecto, estas tareas críticas serán las encargadas de identificar el camino crítico por el que la secuencia de ejecución tenga un tiempo de desarrollo mínimo.

Finalmente, se realizará un seguimiento a las tareas críticas para detectar retrasos de forma inmediata. Estos retrasos afectarán al plan de trabajo, de forma que evolucione con el tiempo.

### 4.2. PLAN DE TRABAJO

En la propuesta del proyecto (PFC-1) se realizó un plan de trabajo que contenía la siguiente planificación temporal. Desglosando de manera detallada el trabajo que se realizaría en cada una de las actividades, junto a una estimación aproximada de horas, sumando un total de 930 horas de trabajo. En este apartado se expone tanto la planificación temporal como la dedicación real, dando una explicación al final del mismo sobre las desviaciones en las que se ha incurrido.

#### 4.2.1. PLANIFICACIÓN TEMPORAL ESTIMADA

##### Fase 1: Análisis

- Actividad 1.1 Análisis de requisitos de usuario (40h).
  - Realización encuestas y entrevistas.
  - Adquisición de información.
  - Estudio del estado del arte en el dominio tecnológico de la solución.
  - Análisis de requerimientos de usuario.
  - Análisis de viabilidad de requerimientos de usuario.

- Generación de documentación de análisis de requisitos de usuario.
- Actividad 1.2 Análisis de requisitos de software (40h).
  - Análisis de requerimientos de software.
  - Estudio de la viabilidad de las cámaras de alta velocidad.
  - Análisis de viabilidad de requerimientos de software.
  - Generación de documentación de análisis de requisitos de software.

#### Fase 2: Diseño

- Actividad 2.1: Estudio del estado del arte en Visión por Computador, y más concretamente en el reconocimiento de movimiento. (100 horas)
  - Estudio del aspecto teórico de la Visión por Computador.
  - Estudio de viabilidad de diversas herramientas.
  - Generación de documentación de herramientas para la Visión por Computador.
- Actividad 2.2: Análisis y estudio de librerías para tratamiento de imágenes. (150 horas)
  - Estudio del aspecto teórico del tratamiento de imágenes.
  - Estudio de viabilidad de diversas herramientas.
  - Generación de documentación de herramientas para el tratamiento de imágenes.
- Actividad 2.3: Diseño prototipo de la aplicación. (60 horas)
  - Diseño del módulo de herramientas.
  - Diseño del módulo de Interfaz.
  - Generación de documentación del diseño prototipo.

#### Fase 3: Implementación

- Actividad 3.1: Pruebas de detección de movimiento. (140 horas)
  - Implementación del módulo de Flujo Óptico.
  - Realización de pruebas del Flujo Óptico
  - Generación de documentación del módulo de Flujo Óptico
- Actividad 3.2: Pruebas de visualización de datos. (80 horas)
  - Implementación del módulo para la visualización de datos.
  - Realización de pruebas para la visualización de datos
  - Generación de documentación del módulo para la visualización de datos
- Actividad 3.3: Prototipo final integrado. (100 horas)
  - Implementación de todos los módulos en el prototipo final.
  - Realización de pruebas para su correcto funcionamiento.
  - Generación de documentación del prototipo final integrado.
- Actividad 3.4: Optimización. (60 horas)
  - Proceso iterado del desarrollo.

- Realización de pruebas.
- Generación de documentación del prototipo optimizado.

#### Fase 4: Validación y entrega

- Actividad 4.1: Test de validación. (80 horas)
  - Definición de test de validación.
  - Aplicación de los test de validación.
  - Análisis de resultados de los test de validación.
  - Generación de documentación del test de validación
- Actividad 4.2: Validación del usuario final (80 horas)
  - Entrevista para Validación del cliente.
  - Fase de test de Validación del cliente.
  - Análisis de resultados de los test de Validación del cliente.
  - Generación de documentación del test de Validación
- Actividad 4.3: Memoria del proyecto. (100 horas)
  - Recopilación de pruebas y datos de todo el proyecto
  - Generación de documentos

#### 4.2.2. TEMPORIZACIÓN DE GANTT

Temporización																			
Fases y actividades	Meses														Horas				
	1	2	3	4	5	6	7												
Fase 1: Análisis	■	■	■																80
Análisis de requisitos de usuario	■	■																	40
Análisis de requisitos de software	■	■																	40
Fase 2: Diseño		■	■	■	■	■	■												210
Estudio del estado del arte en Visión de Computador		■	■	■	■														100
Estudio de tratamientos de imágenes		■	■	■	■	■													150
Diseño prototipo				■	■	■													60
Fase 3: Implementación				■	■	■	■	■	■	■	■	■	■	■	■	■			300
Pruebas de detección de movimiento				■	■	■	■												140
Pruebas de visualización de datos				■	■	■													80



- Generación de documentación de herramientas para el tratamiento de imágenes.
- Actividad 2.3: Diseño prototipo de la aplicación. (60 horas)
  - Diseño del módulo de herramientas.
  - Diseño del módulo de Interfaz.
  - Generación de documentación del diseño prototipo.

### Fase 3: Implementación

- Actividad 3.1: Pruebas de detección de movimiento. (120 horas)
  - Implementación del módulo de Flujo Óptico.
  - Realización de pruebas del Flujo Óptico
  - Generación de documentación del módulo de Flujo Óptico
- Actividad 3.2: Pruebas de visualización de datos. (80 horas)
  - Implementación del módulo para la visualización de datos.
  - Realización de pruebas para la visualización de datos
  - Generación de documentación del módulo para la visualización de datos
- Actividad 3.3: Prototipo final integrado. (60 horas)
  - Implementación de todos los módulos en el prototipo final.
  - Realización de pruebas para su correcto funcionamiento.
  - Generación de documentación del prototipo final integrado.
- Actividad 3.4: Optimización. (240 horas)
  - Proceso iterado del desarrollo.
  - Realización de pruebas.
  - Generación de documentación del prototipo optimizado.

### Fase 4: Validación y entrega

- Actividad 4.1: Test de validación. (60 horas)
  - Definición de test de validación.
  - Aplicación de los test de validación.
  - Análisis de resultados de los test de validación.
  - Generación de documentación del test de validación
- Actividad 4.2: Validación del usuario final (60 horas)
  - Entrevista para Validación del cliente.
  - Fase de test de Validación del cliente.
  - Análisis de resultados de los test de Validación del cliente.
  - Generación de documentación del test de Validación
- Actividad 4.3: Memoria del proyecto. (120 horas)
  - Recopilación de pruebas y datos de todo el proyecto
  - Generación de documentos

## 4.2.4. TABLA COMPARATIVA DE RESULTADOS

	Planificación estimada	Dedicación real
Análisis	80	80
Diseño	210	200
Implementación	380	500
Validación y entrega	260	240
Total	930	1020

Figura 4.2 Tabla Comparativa de Resultados

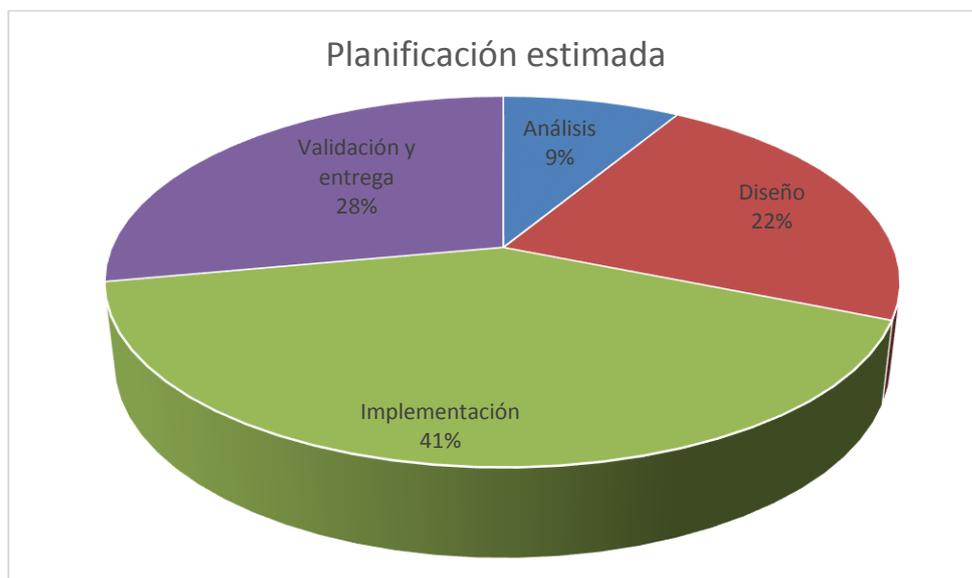


Figura 4.3 Planificación estimada

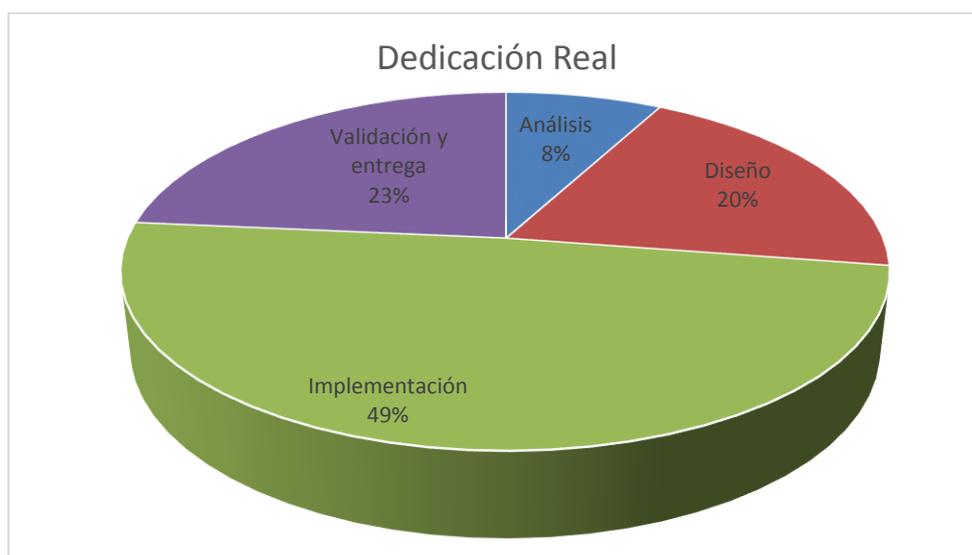


Figura 4.4 Dedicación real

#### 4.2.5. CONCLUSIONES

De la evolución de la planificación temporal puede observarse una diferencia notable entre la estimación inicial y el esfuerzo real dedicado. A continuación se describirán las circunstancias que se han dado para estas diferencias, teniendo sobre ellas un impacto notable la inexperiencia del autor del proyecto a la hora de realizar la planificación de un proyecto de esta índole.

Comenzando por el análisis, las estimaciones han resultado acertadas y no hubo contratiempos durante las entrevistas y el estudio del dominio del problema. El análisis del estado del arte arrojó luz acerca de las tendencias en el campo de los análisis clínicos que aportó una guía de referencia a la hora de dirigir el diseño.

El tiempo dedicado al estudio del tratamiento de imágenes y el estado del arte en la Visión por Computador se ha visto enormemente reducido gracias a que buena parte de ello se había estudiado anteriormente como parte de la formación del autor del proyecto. El resto del tiempo durante la etapa de diseño ha transcurrido de manera esperada y sin contratiempos.

Lo contrario ha ocurrido con el tiempo dedicado a la Implementación. Comenzando con diversos problemas de incompatibilidad entre las librerías de tratamiento de imágenes, el entorno de desarrollo, y el sistema operativo que retrasaron el inicio de la fase hasta que se halló como alternativa viable una versión ya compilada de las librerías\*.

A continuación, las pruebas realizadas para la detección de movimiento permitieron marcar límites respecto a la calidad de los videos que es capaz de analizar. Debido a esto el tiempo dedicado a estas pruebas se centró en el calibrado de los videos que si era capaz de analizar correctamente, recortando mínimamente el tiempo estimado.

La sorpresa vendría en la etapa de Optimización de la aplicación, debido a la naturaleza de la metodología se subestimó el tiempo dedicado a esta fase, dando como resultado que la optimización fuese la actividad que más tiempo necesita.

Finalmente tenemos unas pequeñas variaciones durante la cuarta y última fase de la planificación. En primer lugar, gracias a la revisión habitual de la aplicación por parte del usuario tenemos que se acorta el test de validación y la validación del cliente. Mientras que la realización de la memoria ha resultado un proceso más largo del esperado.

---

\* Afortunadamente, las versiones más recientes de las librerías no presentan problemas a la hora de compilar la aplicación.

## 5. ESTUDIO DE TECNOLOGÍAS Y HERRAMIENTAS

### 5.1. INTRODUCCIÓN

Se ha realizado un estudio de las tecnologías y herramientas óptimas para desarrollar con eficacia el proyecto teniendo en cuenta tanto la experiencia previa del alumno con las mismas como el nivel de prestaciones que ofrecen.

Inicialmente, se tenía pensado implementar el proyecto en C# usando la última versión del Visual Studio, y las librerías de *Microsoft Media Foundation* para tratar los archivos de video. Sin embargo esta idea se desechó cuando se planteó dejar las opciones abiertas a la implementación multiplataforma de la aplicación como una futura ampliación.

La elección del lenguaje como primer paso facilita posteriormente la elección de las herramientas, acotando las opciones disponibles para no sobrecargar el estudio de todas las posibilidades. De esta forma tenemos que el lenguaje elegido debe ser uno capaz de ser llevado a múltiples plataformas sin muchos problemas y a ser posible un lenguaje con el que el desarrollador ya tuviese experiencia previa.

Por tanto, el lenguaje de programación seleccionado para la realización de la aplicación es **C++**, tanto por comodidad para el desarrollador, al ser el lenguaje que mejor conoce; como por la propia flexibilidad del lenguaje. La cantidad de documentación disponible en Internet por si surgen dudas o problemas es también un punto a favor en la elección de este lenguaje.

### 5.2. LISTADO DE HERRAMIENTAS

A continuación se describen las herramientas software escogidas, y posteriormente se discutirán algunas de ellas en mayor profundidad:

**Microsoft Windows 7 de 32bits.** El proyecto se inició con la versión del sistema operativo *Microsoft Windows 7*. Esta versión está diseñada para uso en PC, incluyendo equipos de escritorio en hogares y oficinas, equipos portátiles, tablet PC, netbooks y equipos media center. Posee un amplio abanico de aplicaciones y librerías disponibles, así como compatibilidad con sus versiones anteriores.

Durante las etapas finales del proyecto, el sistema se actualizó a la versión **Windows 8.1 de 64bits**.

**Eclipse.** De la Fundación Eclipse, se trata de un entorno de desarrollo integrado de código abierto multiplataforma, utilizada para desarrollar entornos de desarrollo integrados como el *Java Development Toolkit* y el compilador *ECJ*, pero también ampliamente utilizada para desarrollar las llamadas “*Aplicaciones de Cliente Enriquecido*”.

Eclipse emplea módulos para proporcionar toda la funcionalidad al frente de la plataforma de cliente enriquecido, proporcionando funcionalidad según el usuario las vaya necesitando. Un aspecto clave a la hora de abordar el desarrollo de la solución del problema, de modo que aunque el SDK de Eclipse solamente incluye las herramientas necesarias para el desarrollo de Java, mediante diversos plugins se puede disponer de lenguajes como C/C++ (Eclipse CDT).

Aunque existan otros entornos de desarrollo más profesionales de cara a la programación multiplataforma, la experiencia en el uso de la herramienta ha sido un factor determinante a la hora de su elección, así como la facilidad con la que se pueden instalar toda clase de módulos para ayudar en el desarrollo de la aplicación.

**Compilador MinGW y MSYS.** Un entorno de desarrollo minimalista para aplicaciones nativas de Microsoft Windows. Las herramientas de programación que provee *MinGW* son apropiadas para el desarrollo sin depender de librerías de terceros. Además, sus compiladores permiten la funcionalidad de las aplicaciones en tiempo de ejecución de Microsoft C. Sin embargo, la aproximación minimalista de *MinGW* implica que nunca proporcionará un entorno de ejecución POSIX.

Por otro lado, MSYS es un intérprete de línea de comandos alternativo al `cmd.exe` de Microsoft. Este intérprete es particularmente adecuado para trabajar con *MinGW*, ya que incluye una serie de herramientas Unix elegidas para facilitar este objetivo.

**wxWidgets.** Inicialmente un proyecto para crear aplicaciones portables entre Unix y Windows, creció para dar soporte a Mac, WinCE y otras plataformas. Docenas de desarrolladores contribuyen activamente al proyecto y el kit de herramientas tiene una fuerte base de usuarios que incluye tanto desarrolladores particulares como empresas (por ejemplo AOL). *wxWidgets* proporciona una API simple y fácil de usar para escribir la interfaz gráfica de las aplicaciones en múltiples plataformas que aún usan el control nativo y las utilidades de la plataforma.

**wxJSON.** Para hablar de *wxJSON* antes hay que explicar que es *JSON*. *JSON* son las siglas de *JavaScript Object Notation*, y es un formato sencillo y ligero para el intercambio de datos, completamente independiente del lenguaje pero que usa convenciones similares a los lenguajes de la familia de C. Es sencillo de leer y escribir para humanos, e igual de sencillo de analizar y construir para las máquinas. *wxJSON* es, por tanto, una librería para *wxWidgets* que implementa los objetos JSON en el ecosistema *wxWidgets*, convirtiendo y usando las ristras `wxString` en el objeto de manera intuitiva. *wxJSON* forma parte del proyecto **wxCode**, el cual ofrece multitud de librerías relacionadas con *wxWidgets*.

**wxMathPlot.** Se trata de una librería que agrega la funcionalidad de representar gráficas científicas en 2D en el entorno de *wxWidgets*. Muy sencilla, ligera y fácil de implementar sea cual sea la versión de *wxWidgets*.

**OpenCV.** *Open Source Computer Vision Library* es una librería de software de aprendizaje construida para proporcionar una infraestructura común a las aplicaciones de visión por computador y acelerar el uso de la percepción de las máquinas en productos comerciales. La librería contiene más de 2500 algoritmos optimizados, que incluyen tanto algoritmos clásicos como actualizados de visión por computador y algoritmos de aprendizaje. Estos algoritmos pueden ser utilizados para detectar y reconocer caras, identificar objetos, clasificar acciones humanas en videos, seguir movimientos de cámara, seguir objetos en movimiento, extraer modelos tridimensionales de objetos, producir nubes de puntos tridimensionales de estéreo-cámaras, etc .

La detección de movimiento forma una parte importante de la aplicación, por lo que se ha optado por aplicar la técnica de flujo óptico usando esta librería gratuita que es potente y un referente en el campo de la visión por computador.

**FFmpeg.** Framework multimedia capaz de decodificar, codificar, transcodificar, filtrar y reproducir prácticamente cualquier cosa. Soporta desde los formatos más antiguos hasta los más novedosos. Sin importar si fueron diseñados por un comité, una comunidad o una empresa. Contiene las *librerías libavcodec, libavutil, libavformat, libavdevice, libswscale y libswresample* que pueden ser usadas por las aplicaciones. También *ffmpeg, ffmpegserver, ffmpegplay y ffmpegprobe* que pueden usarse por usuarios finales para transcodificar, streaming y reproducir.

**TortoiseSVN.** Se trata de un control de versiones para Windows muy fácil de usar basado en *Apache Subversion (SVN)*. Ha sido desarrollada bajo licencia GPL, lo que significa que es de uso completamente gratuito, incluso en un entorno comercial, sin ninguna restricción. Tampoco se ha integrado en ningún IDE específico, por lo que se puede utilizar junto a cualquier herramienta que se desee.

**Subversive.** El proyecto *Subversive* integra el control de versiones de *Subversion (SVN)* con la plataforma Eclipse. Usando el plug-in *Subversive* uno puede trabajar directamente desde Eclipse en proyectos almacenados en los repositorios *Subversion* tal y como puede hacerse con otros control de versiones para Eclipse, como CVS o Git.

En nuestro caso utilizaremos este plug-in para coordinar e integrar el control de versiones entre *TortoiseSVN* y *Eclipse*.

### 5.3. WXWIDGETS

wxWidgets es una librería de clases que permite compilar programas gráficos en C++ en multitud de plataformas. wxWidgets define una API común para todas ellas, pero usa la interfaz nativa de cada plataforma para general los gráficos. Por lo que el programa que se implemente tendrá aspecto de ser nativo para el sistema.

Mientras que las aplicaciones de interfaces son construidas en su mayor parte de forma codificada, existen varios editores que permiten construir las interfaces de manera atractiva. Poniendo un ejemplo comercial, *DialogBlocks* de *Anthemion Software*, aunque hay varios otros como *wxFormBuilder*, *wxDev-C++*, y *Code::Blocks*.

Internamente, el código de *wxWidgets* se divide en 6 niveles:

1. **Código común**, que es usado en todas las versiones. Esto incluye clases de datos, información de tiempo de ejecución, clases bases como la *wxWindowBase*, que se utilizan para factorizar el código común a todas las implementaciones de un clase.
2. **Código genérico**, para implementar los widgets, independiente de la plataforma, permitiendo emular los controles y otras funciones que no estén presentes en una plataforma determinada. *wxWizard* y *wxCalendarCtrl* son ejemplos de controles genéricos.
3. **wxUniversal** es un conjunto de widgets básico para aquellas plataformas que no posean sus propios widgets, como *X11* o *MGL*.
4. **Código específico de la plataforma** para implementar las clases de funcionalidades nativas de cada plataforma. Un ejemplo de código específico de plataforma es la clase *wxTextCtrl* en *wxMSW* para implementar el control de edición en *Win32*.
5. **Código aportado** existe en una jerarquía independiente llamada *contrib* e incluye clases no esenciales, pero útiles como *wxStyledTextCtrl*.
6. **Código de terceros** comprende las bibliotecas que se han desarrollado de forma independiente de *wxWidgets*, pero se utilizan para implementar características importantes. Ejemplos de código de terceros son las bibliotecas para *JPEG*, *Zlib*, *PNG* y *Expat*.

### 5.3.1. ESTRUCTURA DE UNA APLICACIÓN

Para preparar la ejecución de una aplicación *wxWidgets*, lo primero es derivar la clase *wxApp* y sobrescribir *wxApp::OnInit*.

Una aplicación *wxWidgets* debe tener como más alto nivel una ventana *wxFrame* o una ventana *wxDialog*. Cada ventana puede contener una o más instancias de clases como *wxPanel*, *wxSplitterWindow* u otras ventanas y controles.

Una ventana puede tener un *wxMenuBar*, un *wxToolBar*, un *wxStatusBar*, y un *wxIcon*. Cada una de estas clases se corresponde, respectivamente, con la barra de menú, la barra de herramientas, la barra de estado, y el icono de la aplicación.

*wxPanel* se usa para colocar controles (clases derivadas de *wxControl*) que se usa para interactuar con el usuario. Ejemplos de estos controles son *wxButton*, *wxCheckBox*, *wxChoice*, *wxListBox*, *wxRadioBox* y *wxSlider*.

Una instancia de *wxDialog* también puede usarse para los controles y tiene la ventaja de no requerir crear por separado una ventana.

En lugar de crear una caja de dialogo y llenarla de objetos, es posible escoger una de las clases de dialogo más comunes como *wxMessageDialog* y *wxFileDialog*.

Nunca se dibuja directamente sobre la ventana, en su lugar se usa un Dispositivo de Contexto (DC). *wxDC* es la base para *wxClientDC*, *wxPaintDC*, *wxMemoryDC*, *wxPostScriptDC*, *wxMemoryDC*, *wxMetafileDC* y *wxPrinterDC*. Si cualquiera de las funciones de dibujado tiene *wxDC* como parámetro, se le puede pasar cualquiera de estas instancias a la función y poder usar el mismo código para dibujar sobre distintos dispositivos. Se puede dibujar usando las funciones miembro de *wxDC*, tales como *wxDC::Drawline* y *wxDC::DrawText*, así como controlar el color en la ventana (*wxColour*) con pinceles (*wxBrush*) y lápices (*wxPen*).

Para interceptar eventos, añades un macro *DECLARE\_EVENT\_TABLE* en la declaración de clase de la ventana, y se pone un nuevo bloque *BEGIN\_EVENT\_TABLE...END\_EVENT\_TABLE* en el archivo de implementación. Entre estos macros se añaden nuevos macros que mapean el evento a una función miembro. Estos pueden sustituir los manejadores de eventos predefinidos como *wxKeyEvent* y *wxMouseEvent*.

Además, las aplicaciones más modernas tendrán un sistema de ayuda online, para ello se necesita hacer uso de *wxHelp* y la clase *wxHelpController* que lo controla.

## 5.4. OPENCV

*OpenCV* es una enorme librería que incluye varios cientos de algoritmos y se encuentra dirigida fundamentalmente a la visión por computador en tiempo real. Esta librería proporcionaría un marco de trabajo de nivel medio-alto que ayudaría al personal docente e investigador a desarrollar formas de interactuar con los ordenadores.

Entre sus muchas áreas de aplicación destacarían: interacción hombre-máquina; segmentación y reconocimiento de objetos; reconocimiento de gestos; seguimiento del movimiento; estructura del movimiento; y robots móviles.

Algunas de las características más importantes de esta librería son:

- Su uso es libre tanto para su uso comercial como no comercial, de acuerdo con sus licencias BSD, MIT y Open Source Definition.
- No utiliza librerías numéricas externas, aunque puede hacer uso de alguna de ellas, si están disponibles, en tiempo de ejecución.
- Es compatible con *The Intel® Processing Library* (IPL) y utiliza *The Intel® Integrated Performance Primitives* (IPP) para mejorar su rendimiento, si se encuentran disponibles en el sistema.
- *OpenCV* se ha escrito en C++ y es su interfaz primaria, aunque aún retiene gran parte de su interfaz original en C. Sin embargo dispone de multitud de interfaces para otros lenguajes y entornos, desde Matlab/Octave, Java y Python hasta wrappers para C#, Perl y Ruby.

En esta sección solamente se comentará el uso de aquellos aspectos y algoritmos que vayan a ser usados en el desarrollo de este proyecto.

De entrada cabe mencionar que los algoritmos aquí descritos harán uso de la *API OpenCV 2.x*, que hace referencia a la API que usa esta librería para el lenguaje C++.

### 5.4.1. MEDIA I/O

El primer módulo que se usa es el *Media I/O*. Este módulo permite capturar video a partir de archivos, secuencias de imágenes, o cámaras; así como también permite la creación de archivos de video.

La clase *VideoCapture*, como indica su nombre, se ocupa de capturar y leer los archivos de video. El constructor de la clase enlaza el objeto con la ruta al archivo en cuestión, mientras que la función miembro `isOpened()` comprueba si el constructor ha tenido éxito. A la hora de cerrar el programa, el propio destructor se encarga de cerrar la conexión del objeto con el archivo, de forma que no es necesario utilizar la función `release()`.

Como ejemplo de uso tenemos:

```
VideoCapture cap("Video.avi");
if(!cap.isOpened())
    return -1;
```

Figura 5.1 Ejemplo de apertura de un objeto con VideoCapture

A partir del objeto VideoCapture se puede usar la función miembro Get() para extraer información del video tal como la velocidad de reproducción, el tamaño de la imagen, el códec, la posición actual y la duración del video en fotogramas. Al contrario, la función Set() sirve para cambiar el valor de una propiedad en el video. Para usar correctamente esta función es necesario usar el identificador adecuado según la propiedad que se quiera conocer. Continuando con el ejemplo:

```
Int totalframes = cap.get(CAP_PROP_FRAME_COUNT );
Int width = cap.get(CAP_PROP_FRAME_WIDTH);
Int height = cap.get(CAP_PROP_FRAME_HEIGHT);
cap.set(CAP_PROP_FPS, 30);
```

Figura 5.2 Ejemplo de lectura y escritura de atributos de un objeto VideoCapture

La función miembro read() y el operador >> realizan la misma función. Capturar, descifrar y devolver el siguiente fotograma en el video. La función devuelve una estructura Mat que comprende la matriz de la imagen. La sencillez de su aplicación se muestra en el siguiente ejemplo:

```
Mat frame;
Cap >> frame;
```

Figura 5.3 Ejemplo de captura de un fotograma

La clase VideoWriter, al contrario que VideoCapture, permite construir archivos de video a partir de imágenes. Para ello se puede usar la función open() en un objeto VideoWriter, o definir las propiedades del mismo en el constructor para luego ir añadiendo fotograma a fotograma cada imagen que forma el archivo de video. La cabecera de la clase es la siguiente.

```
VideoWriter::VideoWriter(const string& filename, int fourcc,
double fps, Size frameSize, bool isColor=true)
```

Figura 5.4 Ejemplo de creación de un objeto VideoWriter

Y de forma similar a la clase anterior, usando las funciones write() o << se añaden los nuevos fotogramas al objeto. Un ejemplo:

```
VideoWriter NewVideo("video.avi", fourcc, fps, frameSize);
if(!NewVideo.isOpened())
    return -1;
NewVideo << newframe;
```

Figura 5.5 Ejemplo de uso de un objeto VideoWriter

#### 5.4.2. IMAGE PROCESSING

Este módulo contiene toda una serie de algoritmos relacionados con el procesamiento de imágenes. Desde filtros y transformaciones de imágenes, a histogramas, análisis de movimiento y detección de características.

Precisamente uno de los algoritmos de detección de características incluidos en este módulo es el detector de esquinas de Shi-Tomasi. Una versión mejorada del detector de esquinas de Harris que nos permite identificar bordes por medio de los autovectores.

La función `goodFeaturesToTrack` es una aplicación del detector de Shi y Tomasi, cuya forma puede verse debajo, y sirve perfectamente a nuestro propósito, sin embargo hay que tener en cuenta que la imagen de entrada ha de ser una imagen de un solo canal, por lo que hay que transformar la imagen a escala de grises.

```
void cv::goodFeaturesToTrack ( InputArray image,
                             OutputArray corners,
                             Int maxCorners,
                             double qualityLevel,
                             double minDistance,
                             InputArray mask = noArray(),
                             Int blockSize = 3,
                             Bool useHarrisDetector = false
                             double k = 0.04
                             )
```

Figura 5.6 Cabecera de la función Shi-Tomasi

El resultado de aplicar el algoritmo de Shi-Tomasi a una imagen es un vector con los bordes más destacables de la imagen. Detectar estos bordes facilitará en gran medida el proceso posterior en la aplicación del algoritmo de flujo óptico.

### 5.4.3. VIDEO

Este módulo trata de algoritmos para el análisis de video, concretamente para el análisis de movimiento y el seguimiento de objetos en una secuencia de fotogramas. Aquí encontraremos el algoritmo que utiliza el método iterativo de Lucas-Kanade para el cálculo de flujo óptico.

La función que implementa dicho algoritmo, `calcOpticalFlowPyrLK()`, precisa tanto de la imagen anterior y posterior, como de los puntos que se desean observar durante el desarrollo. Además de toda una serie de criterios de búsqueda como el tamaño de la ventana de búsqueda y el número de niveles de la pirámide de búsqueda.

```
void cv::calcOpticalFlowPyrLK (   InputArray prevImg,
                                InputArray nextImg,
                                InputArray prevPts,
                                InputOutputArray    nextPts,
                                OutputArray    status,
                                OutputArray    err,
                                Size winSize = Size(21, 21),
                                int  maxLevel = 3,
                                TermCriteria  criteria          =
                                TermCriteria(TermCriteria::COUNT+Term
                                Criteria::EPS, 30, 0.01),
                                int  flags = 0,
                                double    minEigThreshold = 1e-4
                                )
```

Figura 5.7 Cabecera de la función Lucas-Kanade

Tal y como se explicará posteriormente, el algoritmo iterativo de Lucas-Kanade aplica técnicas de flujo óptico esparcido en lugar de flujo óptico denso. Esto significa que para su correcto funcionamiento se precisa de una serie de regiones previamente definidas como buenas para su rastreo. Es por esto que aplicar el algoritmo de Shi-Tomasi a la imagen antes de aplicar el Lucas-Kanade es tan importante. De otro modo, tendríamos que utilizar técnicas para el cálculo de flujo óptico denso, lo cual se notaría en el tiempo de computación de la aplicación.

Pese a que no se hará uso del mismo en este proyecto, cabe mencionar que se puede encontrar en esta librería el algoritmo de Gunnar Farneback para el cálculo de flujo óptico denso en la función `calcOpticalFlowFarneback()`.

#### 5.4.4. HIGH-LEVEL GUI

Aunque no vaya a utilizarse en la versión definitiva del producto, el módulo de interfaz de alto nivel que proporciona OpenCV resulta útil a la hora de probar las propias funciones de la librería sin influencia externa por parte de la interfaz de wxWidgets que se usará en la versión final. Este módulo gira en torno a 3 funciones elementales:

- La primera función, `imread`, carga la imagen que se desee desde un archivo a un objeto `Mat`.
- La segunda función, `namedWindow`, crea una ventana del tamaño deseado.
- La tercera función, `imshow`, muestra la imagen contenida en un objeto `Mat` en una ventana.

A continuación, un ejemplo de código que hace uso de todas estas funciones para mostrar una imagen en pantalla.

```
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>
#include <iostream>

using namespace cv;
using namespace std;

int main( int argc, char** argv )
{
    if( argc != 2 ) {
        cout << " Usage: display_image ImageToLoadAndDisplay" << endl;
        return -1;
    }

    Mat image;
    image = imread(argv[1], CV_LOAD_IMAGE_COLOR);    // Read the file

    if(! image.data ) {                               // Check for invalid input
        cout << "Could not open or find the image" << std::endl ;
        return -1;
    }

    namedWindow( "Display window", WINDOW_AUTOSIZE );// Create a window for
display.
    imshow( "Display window", image );                // Show our image inside it.

    waitKey(0);                                       // Wait for a keystroke in the window
    return 0;
}
```

Figura 5.8 Ejemplo de uso del módulo High-Level GUI

## 6. INTRODUCCIÓN AL FLUJO ÓPTICO

En el campo de la visión por computador se conoce como *flujo óptico* al movimiento aparente de los píxeles de una imagen de un cuadro a otro dentro de una secuencia de video. En tres dimensiones el flujo óptico es definido como una medida de cómo se mueven los puntos en el espacio. El cálculo del flujo óptico no es un problema sencillo y computarlo para tomas realizadas en un ambiente real puede llegar a ser muy complicado. Por eso los algoritmos existentes se basan en determinadas hipótesis (intensidad constante, rigidez de los objetos, coherencia espacial, entre otras) que generalmente no se cumplen en escenarios reales.

### 6.1. CALCULO DEL FLUJO ÓPTICO BASADO EN EL GRADIENTE

La principal hipótesis detrás del cálculo del flujo óptico es la conservación de la intensidad de un píxel en el tiempo a pesar del posible cambio de posición. Matemáticamente es conocida como la ecuación de restricción de flujo óptico y podemos calcularla tomando la ecuación:

$$f(x+\Delta x, y+\Delta y, t+\Delta t) \approx f(x, y, t) \quad (1)$$

donde  $f(x, y, t)$  es la intensidad de la imagen en la posición  $(x, y)$  en el instante de tiempo  $t$  y  $\Delta x, \Delta y$  son el cambio de posición y  $\Delta t$  es el cambio en el tiempo. Aplicando el desarrollo en *Serie de Taylor* a la parte izquierda obtenemos:

$$f(x+\Delta x; y+\Delta y; t+\Delta t) = f(x, y, t) + \partial f \partial x \Delta x + \partial f \partial y \Delta y + \Delta f \Delta t + o(f) \quad (2)$$

donde  $\partial f \partial x, \partial f \partial y, \Delta f \Delta t$  son las derivadas parciales de la imagen en función de  $x, y, t$  ignoramos los términos restantes de la serie porque estamos trabajando con vecindades suficientemente pequeñas. Sustituyendo la ecuación (1) en (2) y desechando el término  $o(f)$  nos queda la ecuación de la conservación de la intensidad.

$$\partial f \partial x \Delta x + \partial f \partial y \Delta y + \Delta f \Delta t + o(f) = 0 \quad (3)$$

$$\text{Renombrando } I_x = \partial f \partial x, I_y = \partial f \partial y, V = (u, v) = (\Delta x, \Delta y)$$

$$\Delta I * V + I_t = 0 \quad (4)$$

Esta ecuación no brinda en todos los casos una solución única, pues tiene dos incógnitas. Esto es conocido como **el problema de apertura**. El problema de computar la velocidad total de la imagen se transforma entonces en encontrar otra restricción que permita despejar la incógnita restante.

## 6.2. SOLUCIONES AL PROBLEMA DE APERTURA

Con el objetivo de encontrar una restricción adicional a la ecuación existen tres técnicas generales:

- *Diferenciales*

Consiste en computar la velocidad a partir de la *Derivada* espacio-temporal de la intensidad de la imagen o de alguna versión filtrada de la misma.

- *Frecuencia o energía*

Utilizan la *Transformada de Fourier* para calcular el flujo a través del dominio de la *Frecuencia*. Estas suelen ser adecuadas para extraer el movimiento de objetos que son difíciles de capturar para los métodos de correspondencia de bloques y diferenciales, como puede ser el caso de puntos aleatorios.

- *Basadas en la correlación*

Se basan en la búsqueda de correspondencias utilizando ventanas o patrones alrededor de cada píxel. La ventaja que tienen con respecto a los anteriores, es que se utiliza información de los vecinos que en muchos casos hacen que la búsqueda de la correspondencia sea más efectiva. Lo que se busca con estos métodos es asociar píxeles a través de regiones similares en las imágenes que se obtienen por maximización de alguna medida de similaridad.

Independientemente de la técnica que implementen, por los resultados que producen, los métodos pueden clasificarse en dos grandes grupos, algoritmos de flujo óptico esparcido y algoritmos de flujo óptico denso.

- *Flujo óptico esparcido*

Como su nombre sugiere, el flujo óptico esparcido no computa los vectores de velocidad para cada píxel de la imagen. El primer paso es seleccionar una serie de regiones de la imagen que sean buenas para rastrear (bordes, esquinas, manchas, etc.). Luego para ese conjunto reducido de puntos se calcula el flujo óptico para toda la imagen. Uno de los primeros algoritmos que uso este enfoque fue el de Lucas-Kanade propuesto en 1981.

- *Flujo óptico denso*

El flujo óptico denso, a diferencia del esparcido, se caracteriza por computar el vector de velocidad para cada píxel de la imagen. Generalmente este es un proceso computacionalmente costoso por lo que no es factible hacerlo en tiempo real, aunque con el desarrollo de las tarjetas de video y los algoritmos recientes se han

logrado grandes avances en este sentido. Un exponente clásico de este grupo es el algoritmo de Horn y Schunck que data de 1981.

### 6.3. ALGORITMO DE LUCAS-KANADE

Continuando con la descripción del método de flujo óptico, este se sostiene sobre dos hipótesis. Primera, que la intensidad de los puntos no varía entre fotogramas. Y segunda, se asume que el movimiento es constante sobre una región, de forma que todos los pixeles circundantes a un punto se mueven de forma similar.

Lo que hace el algoritmo de Lucas-Kanade es tomar una ventana de 3x3 puntos, y asume que estos 9 puntos se mueven en la misma dirección, por lo que todos los elementos de la ventana mantienen el mismo gradiente. Esto da como resultado un sistema sobredeterminado, por lo que se aplica la solución de mínimos cuadrados, que denota las esquinas como los mejores puntos para el seguimiento.

Sin embargo, este método solo trata con movimientos pequeños. Así que cuando falla porque hay un movimiento mayor, el problema se resuelve con una implementación piramidal del algoritmo escalando la imagen, lo que permite eliminar los movimientos a distancias cortas y convirtiendo así los movimientos largos en movimientos cortos. De esta forma los movimientos se escalan y permite aplicar la solución de Lucas-Kanade.

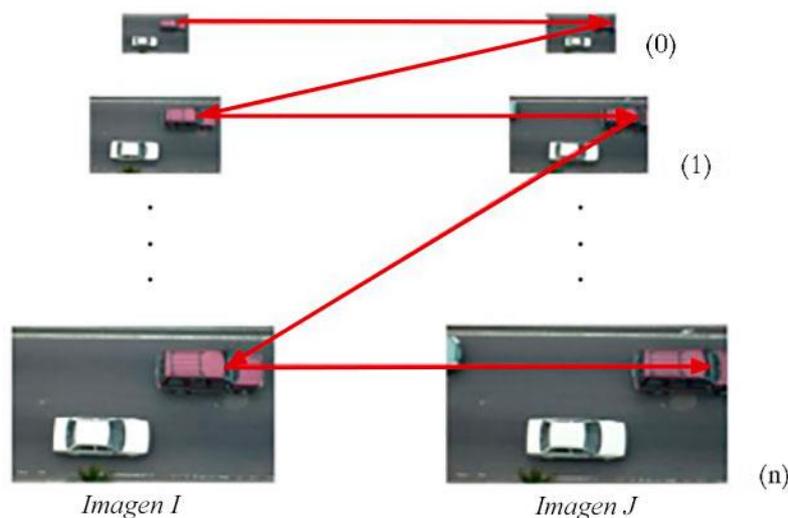


Figura 6.1 Algoritmo piramidal de Lucas Kanade

En la figura tenemos que se estima primero el flujo óptico entre ambas imágenes en el nivel 0, encontrando el desplazamiento de un punto específico sobre la imagen a baja resolución en una ventana pequeña. Luego ubica el mismo punto en el nivel 1 a mayor resolución, y estima su flujo óptico para el desplazamiento entre ambas imágenes. Estos pasos se repiten hasta alcanzar el nivel n, obteniendo de esta manera el desplazamiento total del punto.

#### 6.4. APLICACIÓN DEL FLUJO ÓPTICO

Podemos ver una representación de los resultados de estos métodos aplicando los algoritmos implementados en la librería *OpenCV*. Para ello se ha realizado una pequeña aplicación usando el módulo High level GUI para mostrar los resultados.

La aplicación se dividirá en 3 partes. La primera de ellas pedirá al usuario la ruta del video a tratar y comprobará que el video pueda tratarse correctamente. La segunda parte ejecutará el algoritmo de Shi-Tomasi en el primer fotograma de la secuencia de video, para posteriormente, en el tercer paso, aplicar el algoritmo de Lucas-Kanade al resto de fotogramas consecutivamente y seguir el movimiento en la imagen por la técnica de flujo óptico.

El algoritmo de Shi-Tomasi busca bordes en la imagen y selecciona aquellos puntos más destacables de su entorno para definir un vector de puntos significativos. Solapando la imagen original con este vector obtenemos la siguiente imagen en la que dichos puntos aparecen resaltados en verde.



Figura 6.2 Algoritmo Shi-Tomasi aplicado a un fotograma

Cabe destacar que, efectivamente, los pixeles verdes señalan bordes en la imagen, pero más importante es que cada uno de los marcadores que se han puesto en las articulaciones del animal han sido detectadas con claridad por el algoritmo de Shi-Tomasi. Esto es una buena prueba que nos servirá de referencia durante la implementación de estos algoritmos posteriormente en el proyecto, ya que detectar estos marcadores será clave para el análisis biomecánico del animal.

Tras tomar los puntos encontrados por el algoritmo Shi-Tomasi y aplicarles el algoritmo de Lucas-Kanade, la aplicación nos devuelve la nueva ubicación de los puntos hallados en el nuevo fotograma. Nótese que mientras que el algoritmo Shi-Tomasi busca los bordes y nos los señala en un vector, el algoritmo de Lucas-Kanade realiza una búsqueda de estos puntos del fotograma anterior, en el nuevo fotograma en el que los puntos han podido desplazarse.



Figura 6.3 Algoritmo de Lucas-Kanade aplicado al siguiente fotograma

De nuevo, en verde se encuentran marcados los puntos estáticos que no han cambiado de posición entre fotogramas. Por otro lado, en rojo podemos apreciar el movimiento de diversos puntos entre ambos.

Sin embargo, se ha de tener en cuenta que debido a la calidad del video, el ruido en la imagen puede interferir en la correcta apreciación del mismo. Prueba de esto son las líneas que aparecen en rojo en la parte inferior de la imagen. Por ello es preciso que la calidad de la imagen sea lo más nítida posible, realizando la grabación en un lugar bien iluminado y teniendo presente una correcta velocidad del obturador en la cámara.

## 7. ANÁLISIS

### 7.1. DEFINICIÓN DEL DOMINIO DEL PROBLEMA

El procedimiento por el cual se evalúa al animal consiste en hacerlo caminar sobre una cinta andadora, en una sala bien iluminada, y grabarlo con una cámara de alta velocidad enfocando el lado del animal que se desea evaluar.

En los preparativos para el procedimiento se marca con llamativos marcadores de colores las articulaciones del animal cuyo movimiento se desea seguir posteriormente en la grabación de video.

Tras realizar las grabaciones necesarias se vuelcan los archivos de video de la cámara en el ordenador, donde se utilizará un software especializado para extraer los distintos frames de la grabación y convertirlas en capturas individuales con las que se trabajará por medios tradicionales.

#### 7.1.1. DETERMINACIÓN DEL ALCANCE DEL DOMINIO

Durante la evaluación motriz del animal se marcan ciertos puntos de referencia sobre la imagen para determinar y medir el movimiento de las articulaciones del animal, la distancia de su zancada, ángulo de movimiento de las articulaciones y el tiempo que tarda en completar el movimiento, medidos en frames y posteriormente en segundos.

#### 7.1.2. IDENTIFICACIÓN DEL ENTORNO TECNOLÓGICO

Las herramientas disponibles son las siguientes:

- Ordenador con el sistema operativo Windows 7.
- Cámara de alta velocidad, capaz de grabar en formato AVI.
- Lápiz y papel.

#### 7.1.3. IDENTIFICACIÓN DE USUARIOS PARTICIPANTES

Los siguientes son los usuarios participantes en el dominio del problema:

- *Usuario del análisis de vídeo.* Utiliza la aplicación destinada al análisis de vídeo.

#### 7.1.4. DETALLES Y NORMAS A TENER EN CUENTA

A la hora de buscar una solución al problema hace falta que se mantengan las siguientes directrices:

- Manual de usuario en la aplicación.
- Tiempo de respuesta bajo.
- Interfaz intuitiva y atractiva.
- Capacidad de reproducción de archivos AVI.

## 7.2. ESTABLECIMIENTO DE REQUISITOS

### 7.2.1. REQUISITOS DE USUARIO

#### **Descripción.**

El proyecto consiste en ofrecer de forma sencilla e inmediata información contextual en la reproducción de video. Lo habitual en dicha reproducción será una grabación de un animal en movimiento con diversos puntos marcados de forma claramente visible en sus articulaciones y extremidades. La información contextual a la que se refiere es la distancia recorrida por la zancada del animal y el ángulo en movimiento de sus articulaciones, así como cualquier otra medida de distancia. Además, el proyecto también ofrecerá de forma intuitiva los datos tomados en forma de gráfica y una sección para guardar anotaciones y comentarios relativos al video.

En resumen:

- Capacidad de cargar archivos de video, concretamente en formato AVI, en el programa.
- Aplicaciones habituales en la reproducción de video (reproducir, pausar, rebobinar, adelantar), además de algunos más relevantes al caso (avance y retroceso de fotogramas).
- División en *frames* automática.
- Trabajar sobre dichos *frames* para el cálculo automático de las distancias respecto a la medida de referencia y los ángulos.
- Cálculo del tiempo transcurrido entre dos *frames*.
- Seguimiento automático de los marcadores en el animal.
- Comparación unificada de las líneas marcadas en las articulaciones respecto a un punto fijo.

#### **Glosario de Conceptos.**

- *Usuario*: Es un agente que utiliza la aplicación.
- *Frame*: Es un fotograma de la reproducción de video
- *Marcador*: Una de las marcas puestas en las articulaciones del animal para facilitar la detección de estas por parte de la aplicación.
- *Video*: Archivo multimedia que se compone de varias imágenes, o *frames*, en secuencia.
- *Gráfica*: Resultado visual de los datos analizados durante la reproducción del video.
- *Nota*: Campo de entrada/salida de la aplicación en el que se pueden guardar anotaciones relativas a un *frame* en concreto.

### Modelo de dominio.

El modelo de dominio que se describe a continuación resume de forma generalizada las relaciones entre los distintos elementos que componen el proyecto.

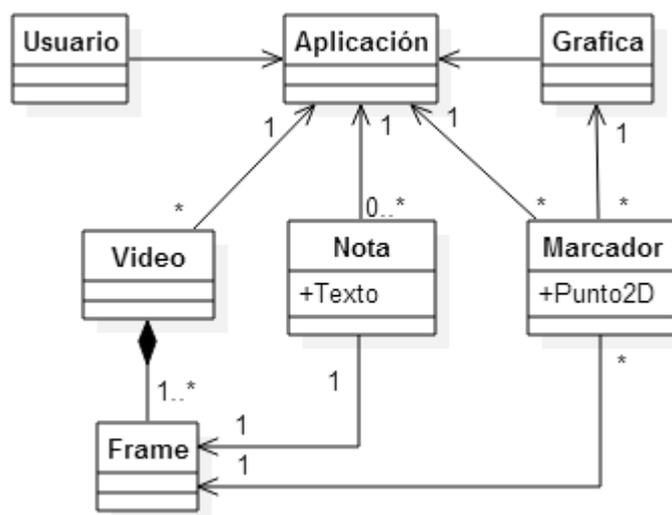


Figura 7.1 Modelo de dominio

## 7.2.2. ESPECIFICACIÓN DE REQUISITOS DEL SOFTWARE

### Descripción

La aplicación está dividida en dos partes, por un lado se encuentra la interfaz de video y por otro una interfaz de datos. La interfaz de video tiene la capacidad de abrir, reproducir, parar y avanzar *frame* a *frame* el video, además puede convertir un conjunto de imágenes consecutivas en un video para su reproducción. Otra capacidad de la interfaz de video es la de marcar ciertos puntos en un *frame* y seguir su movimiento durante su reproducción por medio de técnicas de flujo óptico.

La interfaz de datos se ocupa de mostrar la información obtenida en la interfaz de video de forma organizada: gráficas con el progreso de los datos según los *frames*, sección para tomar notas y guardarlas en la aplicación, detalle de los puntos marcados en el video.

### Glosario de Conceptos

- *Usuario*: Es un agente que utiliza la aplicación.
- *Interfaz de video*: Parte de la aplicación que se ocupa de la interacción entre el usuario y el motor de video de la aplicación.
- *Motor de video*: Parte de la aplicación que procesa las acciones en la interfaz de video y envía la imagen resultante de nuevo a la Interfaz.
- *Interfaz de datos*: Parte de la aplicación que se ocupa de la interacción entre el usuario y el motor de datos de la aplicación.

- *Motor de datos*: Parte de la aplicación que procesa las acciones en la interfaz de video y envía la imagen resultante de nuevo a la Interfaz.
- *Frame*: Es un fotograma de la reproducción de video
- *Marcador*: Una de las marcas puestas en las articulaciones del animal para facilitar la detección de estas por parte de la aplicación.
- *Video*: Archivo multimedia que se compone de varias imágenes, o *frames*, en secuencia.
- *Gráfica*: Resultado visual de los datos analizados durante la reproducción del video.
- *Nota*: Campo de entrada/salida de la aplicación en el que se pueden guardar anotaciones relativas a un *frame* en concreto.
- *Barra de herramientas*: La barra de herramientas se compone de varios botones para manipular la Interfaz de video.

### Modelo Conceptual

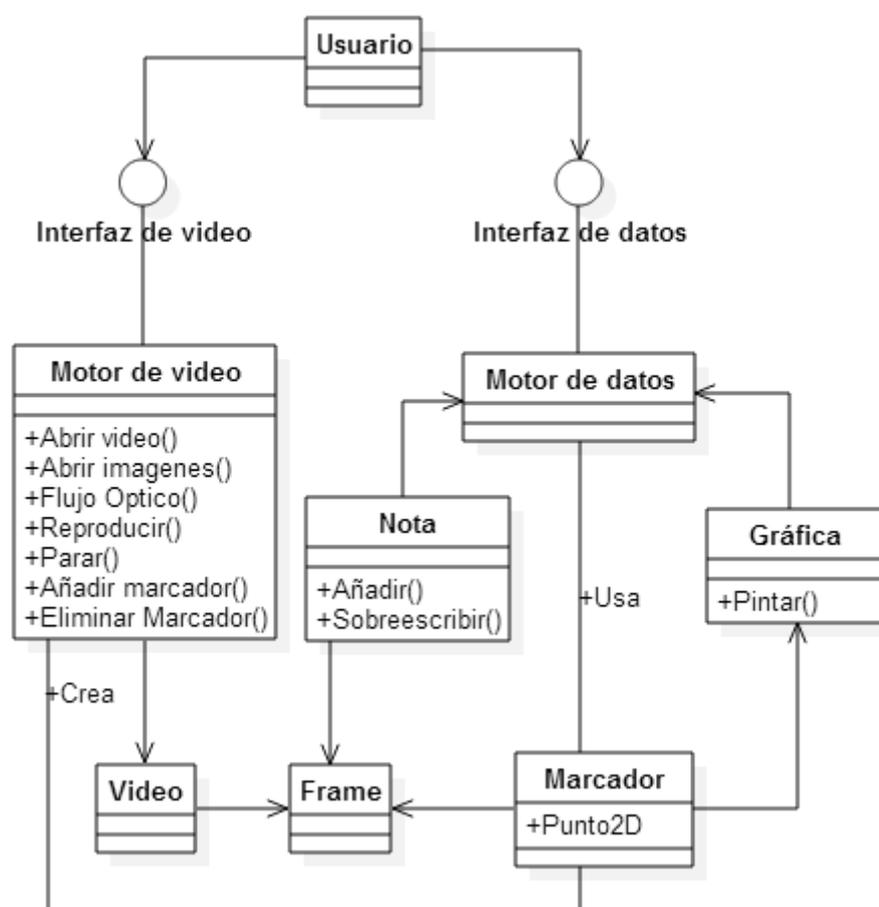


Figura 7.2 Modelo Conceptual

## 7.3. CASOS DE USO

### 7.3.1. LISTA DE ACTORES

**Usuario.** El usuario se ocupa del manejo de las sesiones.

### 7.3.2. LISTADO DE CASOS DE USO DE NIVEL GENERAL

Actor Principal	Casos de Uso
Usuario	1. Abrir archivo de video.
	2. Abrir secuencia de imágenes.
	3. Reproducir archivo de video.
	4. Pausar archivo de video.
	5. Modo búsqueda.
	6. Avanzar frame.
	7. Retroceder frame.
	8. Modificación de los FPS.
	9. Marcar punto en el frame actual.
	10. Calculo de la distancia entre dos puntos de la imagen.
	11. Calculo del ángulo entre tres puntos de la imagen.
	12. Calculo de Flujo óptico para los puntos marcados.
	13. Desplazamiento de los marcadores respecto a un punto
	14. Desplazar gráfica.
	15. Crear nueva anotación.
	16. Sobrescribir anotación.
	17. Guardar datos.
	18. Salir del programa.

Figura 7.3 Listado de casos de uso de nivel general

### 7.3.3. DIAGRAMAS DE CASOS DE USO

#### Casos de apertura:

Lo primero que ha de hacer el usuario es cargar el video o secuencia de imágenes que se piensa procesar. Para ello se selecciona la opción adecuada en el menú. Este paso se puede realizar en cualquier momento, tanto si ya se había cargado un video previamente, como si no.

En el caso de abrir un video, se abrirá una ventana de dialogo en la que se podrá seleccionar el video a reproducir.

En el caso de abrir una secuencia de imágenes, la aplicación procederá a generar un nuevo archivo de video. Para ello se abrirán tres ventanas de dialogo pidiendo ordenadamente:

1. Seleccionar la primera imagen de la secuencia.
2. Seleccionar la última imagen de la secuencia.
3. Un nombre y ubicación con el que nombrar a este nuevo archivo de video. Esto puede sobrescribir un archivo de video en caso de que ya exista.

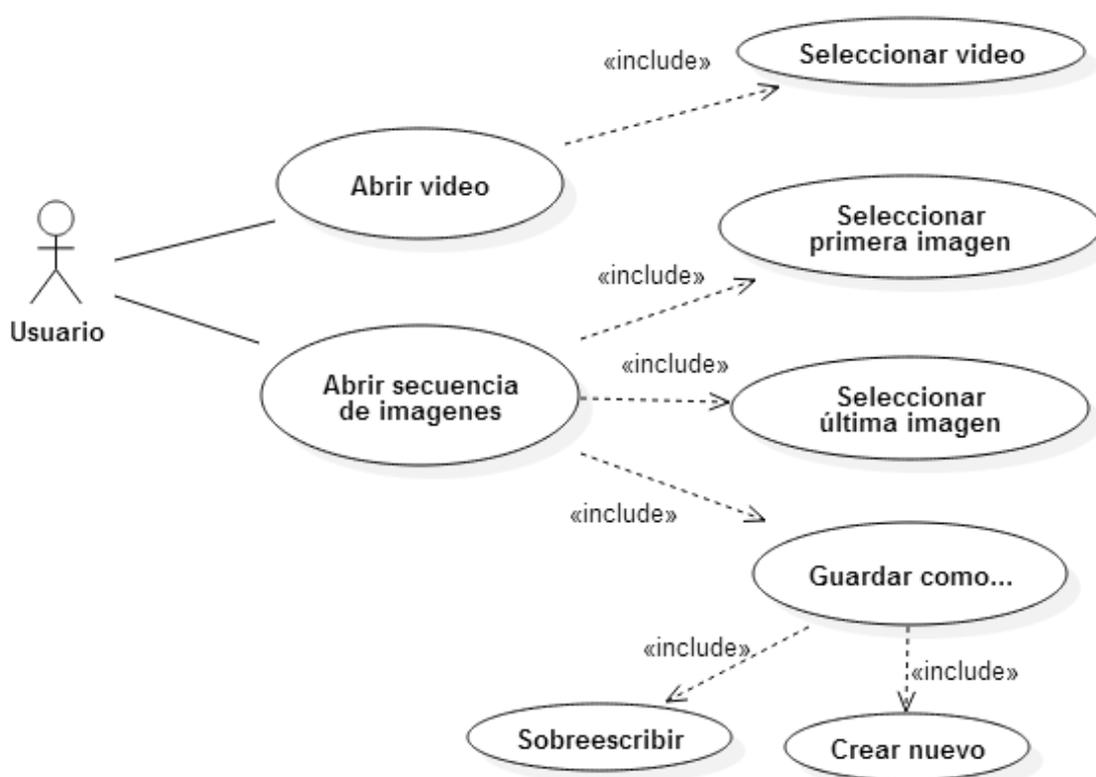


Figura 7.4 Caso de uso de apertura

### Casos de manejo de la reproducción:

Con el objetivo de manipular a voluntad la visualización correcta del video, el usuario dispone de herramientas que le permite reproducir, pausar y parar el video. Para un manejo más especializado también dispone de las herramientas para avanzar y retroceder la reproducción paso a paso (*frame a frame*) y finalmente una barra de búsqueda para situarse en cualquier punto de la reproducción.

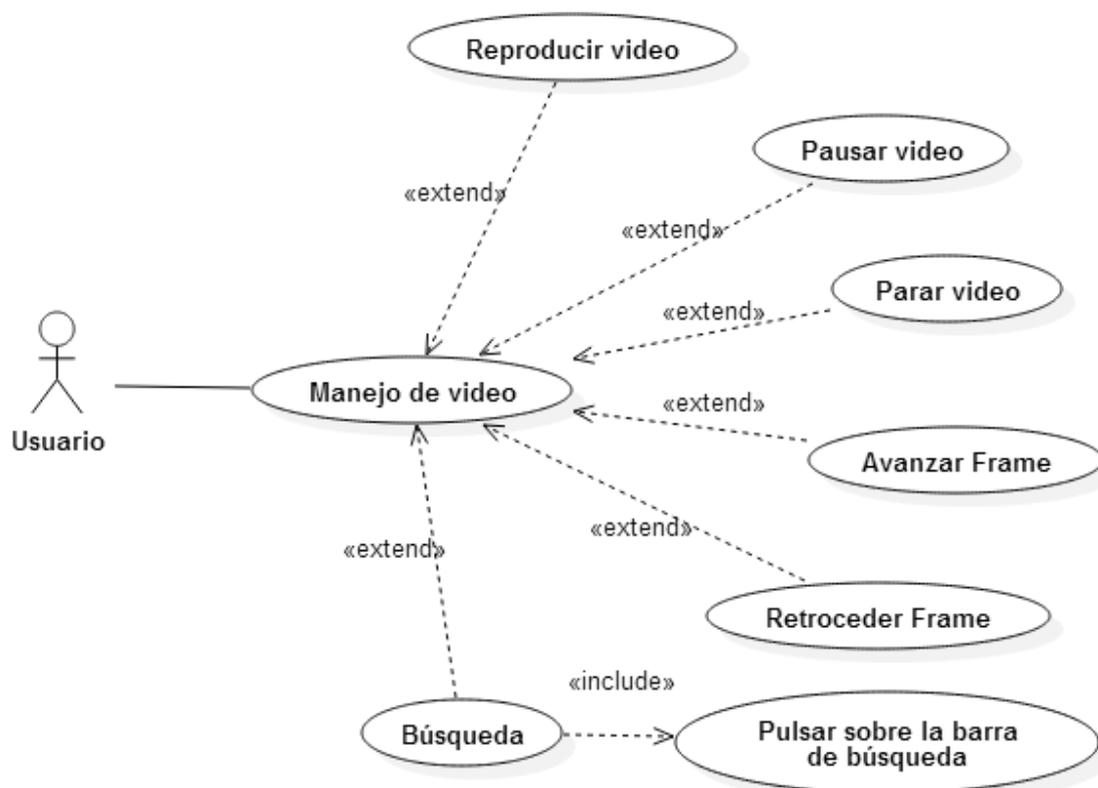


Figura 7.5 Caso de uso de manejo de la reproducción

### Casos de manejo de marcadores:

Los marcadores son una parte importante de la aplicación. Para poder calcular los ángulos y distancias, para poder aplicar las técnicas de flujo óptico, y para poder dibujar las gráficas con los datos; primero que nada el usuario necesita marcar los puntos destacables en la secuencia de video.

A partir de estos puntos marcados en un *frame* de la secuencia de video, el usuario puede relacionarlos entre ellos para hallar su ángulo. O puede marcar uno de ellos para seguir su movimiento y calcular la distancia que recorre. Y finalmente puede activar la opción de seguir el movimiento de todos los puntos por medio de técnicas de flujo óptico.

Los resultados obtenidos por medio de estas técnicas se visualizarán tanto en el cuadro de video como en la gráfica.

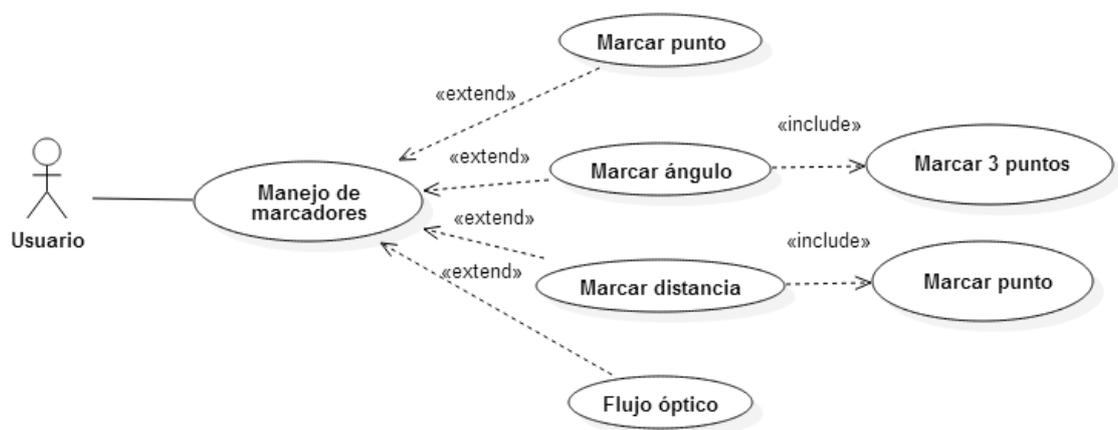


Figura 7.6 Caso de uso de manejo de marcadores

### Casos de manejo de notas:

La aplicación también es capaz de almacenar las notas que escriba el usuario, relacionándolas con un *frame* determinado.

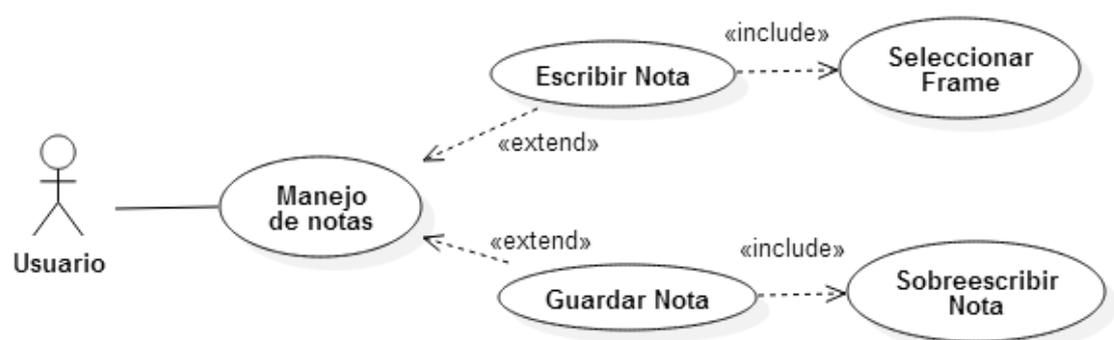


Figura 7.7 Caso de uso del manejo de notas

### Casos de manejo de la gráfica:

La información obtenida por los marcadores se muestra en la aplicación en forma de gráficas y tablas. Para facilitar la visualización de la gráfica esta dispone de opciones que le permiten al usuario desplazar, acercar, alejar y ajustar la gráfica.

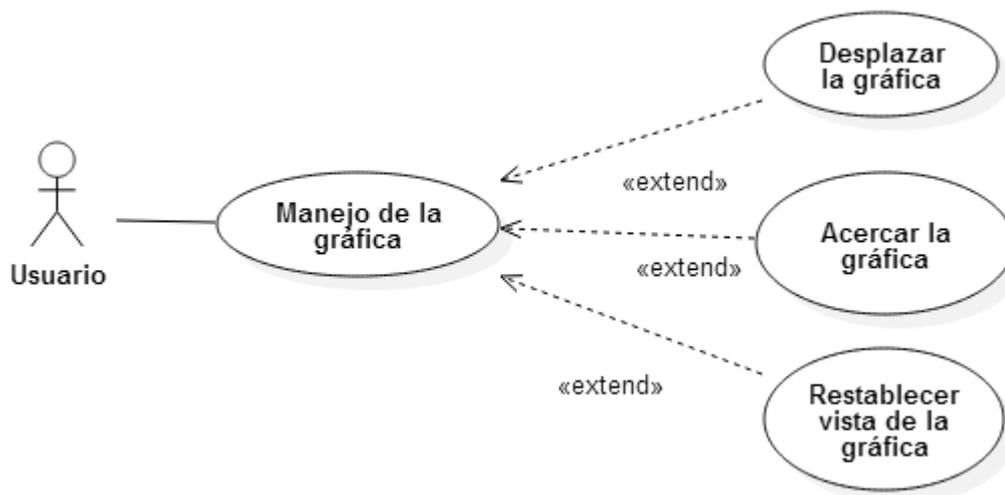


Figura 7.8 Caso de uso de manejo de la gráfica

### Caso para guardar los datos y salir de la aplicación:

Con el fin de no perder toda la nueva información que se ha obtenido durante el uso de la aplicación a la hora de cerrar la aplicación, el usuario es capaz de guardar los datos (marcadores, ángulos, distancias y notas) en un archivo.

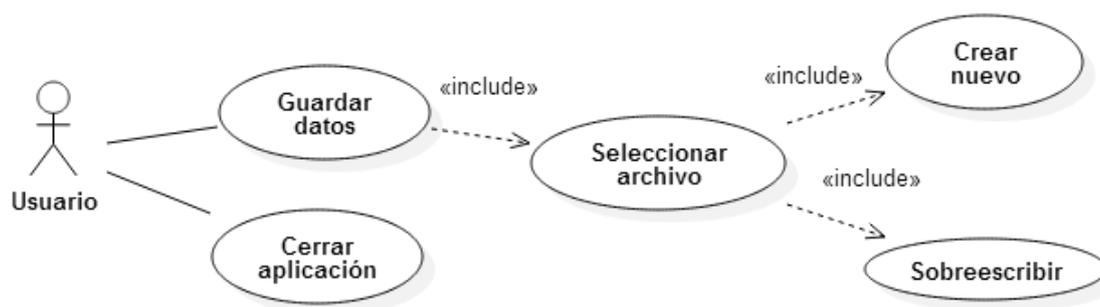


Figura 7.9 Caso de uso de guardar y salir

### Casos de uso completos:

El listado de casos de uso completos se encuentra en el Apéndice A.

## 8. DESARROLLO ITERATIVO

Para cumplir con los requisitos y casos de uso obtenidos durante la fase de análisis, el proceso de desarrollo de software se realizará en base al modelo de desarrollo iterativo e incremental, el cual consiste en planificar un proyecto en distintos bloques temporales denominados iteraciones que acercan el proceso de trabajo a un producto final. Cada iteración se define por los requisitos que debe cumplir el producto al final de dicha iteración, añadiendo nuevos requisitos en cada nueva iteración.

Esta etapa involucra el rediseño e implementación de una tarea de la lista de control de proyecto, y el análisis de la versión más reciente del sistema. La meta del diseño e implementación de cualquier iteración es ser simple, directa y modular, para poder soportar el rediseño de la etapa o como una tarea añadida a la lista de control de proyecto.

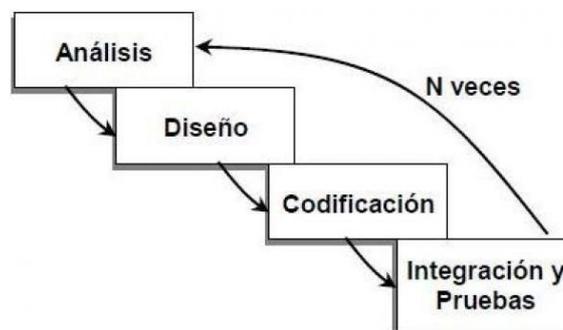


Figura 8.1 Esquema de desarrollo por iteraciones

Debido a que cada iteración contiene sus propias fases de análisis, diseño, implementación y pruebas en base a los requisitos de cada iteración, este capítulo divide cada una de estas fases en las distintas iteraciones que se han llevado a cabo.

### 8.1. FILTRADO DEL PARPADEO

Tal y como se señala durante la explicación teórica del flujo óptico, el algoritmo Lucas Kanade necesita de ciertas condiciones estables antes de poder aplicar su algoritmo. El obturador de la cámara de alta velocidad, junto a la iluminación de la sala donde se grabaron los videos, producía un cierto parpadeo en los tonos del video, lo cual imposibilitaba el uso del algoritmo. Debido a ello, previo al desarrollo es necesario realizar un filtrado a todos los videos para tratar de igualar y eliminar dicho parpadeo. Para ello se ha hecho uso de las herramientas *VirtualDub* y un filtro gratuito hecho con este propósito llamado *Deflicker*.

### 8.2. CALIBRADO DE SHI-TOMASI Y LUCAS KANADE

Antes de comenzar las fases de diseño e implementación de la aplicación se ha realizado un estudio previo sobre los algoritmos que se utilizarán durante su desarrollo. Usando el módulo High-Level GUI de OpenCV para aislar las pruebas de la librería wxWidgets,

se ha realizado una batería de experimentos para hallar los valores óptimos sobre los que trabajar con los algoritmos de Shi-Tomasi y de Lucas Kanade en este caso concreto.

Basándose en las muestras de video adquiridas para la realización de pruebas de este proyecto, la batería de pruebas consistía en consultar los distintos valores para los cuales ambos algoritmos respondiesen de manera favorable a los distintos videos de prueba.

---

### 8.2.1. ALGORITMO DE SHI-TOMASI

De esta forma se ha realizado en primer lugar un conjunto de pruebas para el algoritmo de Shi-Tomasi con el fin de encontrar los bordes más precisos en función de los videos y las variables. Los resultados de dicha batería de pruebas varían en función de 3 variables que se describen a continuación:

- La cantidad máxima de bordes que se pueden encontrar.
  - Se establecieron 500 bordes como máximo.
- La calidad del nivel, que determina el umbral por el que se realizará el corte de calidad en la detección del borde. Se determina por un valor entre 0 y 1.
  - Debido a que la calidad de los videos es muy variable, se ha optado por aplicar un valor de 0.01.
- Y la distancia euclídea mínima entre dos bordes encontrados.
  - 10 pixeles supone una distancia razonable para encontrar los bordes que se precisan en las pruebas.

Los resultados obtenidos se tomarán como valores generales para su uso en todos los videos. De manera que incluso si la calidad de la imagen es baja, la aplicación aún pueda encontrar los mejores bordes.

---

### 8.2.2. ALGORITMO DE LUCAS-KANADE

La batería de pruebas realizadas para el algoritmo de Lucas-Kanade ha requerido una toma de decisiones más compleja que para el algoritmo de Shi-Tomasi. Debido a que los resultados al variar los parámetros han sido más dispares de lo previsto. Se ha optado por tomar unos valores intermedios que funcionen para la mayoría de los videos en la prueba. En el algoritmo de Lucas-Kanade los resultados varían en función de 3 variables:

- El tamaño de la ventana de búsqueda en cada nivel de la pirámide.
  - 11.
- El número de niveles de la pirámide de muestreo.
  - 5.
- Y el valor de criterio para acabar la búsqueda iterativa.
  - maxCount: 20 iteraciones
  - Epsilon (tamaño mínimo de ventana): 0.31

### 8.3. PRIMERA ITERACIÓN

Para la primera iteración los requisitos a cumplir fueron aquellos que componen el núcleo de la aplicación:

- La reproducción de archivos de video.
- El seguimiento de los marcadores durante la reproducción del video.
- Capacidad de cálculo de ángulos y distancias.
- Capacidad para almacenar y restablecer los puntos marcados.

A la hora de abordar el diseño de la aplicación se ha optado por hacer uso del patrón de diseño Fachada. Permitiendo que la Fachada de la aplicación actúe como punto de entrada de las peticiones del usuario, mientras delega las acciones apropiadas a una serie de subsistemas.

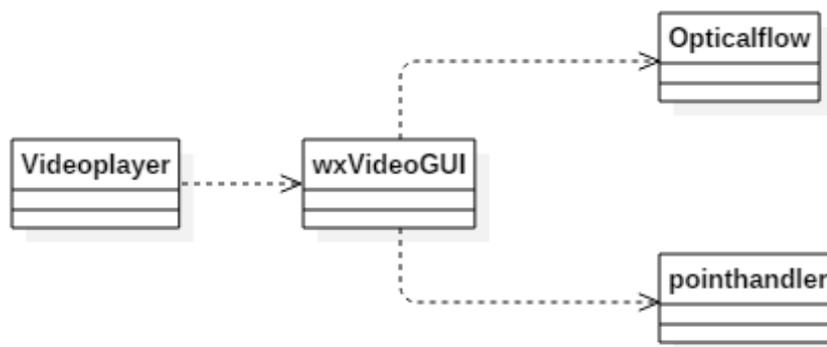


Figura 8.2 Dependencias en el diseño del patrón Fachada.

#### 8.3.1. EL DISEÑO DE LA INTERFAZ

El diseño de la interfaz es uno de los elementos claves en la realización de la aplicación. Se puede definir la interfaz como:

*“El conjunto de imágenes y objetos gráficos para representar la información, así como el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que verá y escuchará en cada momento, y las acciones que realizará, así como las respuestas que el sistema le dará.”*

Para la interfaz se ha buscado la comodidad de un modelo de interfaz conocido por todos los usuarios: el de una aplicación de reproducción de videos. De esta forma tenemos que en la barra superior de menú podemos encontrar las opciones habituales en este tipo de casos, y en particular la opción de carga de video, que mostrará un dialogo de búsqueda para encontrar el video en el sistema de archivos del ordenador.

En medio, y ocupando el 80% de la ventana, tenemos el cuadro donde se mostrará al usuario el archivo de video.

Inmediatamente debajo tenemos la barra de herramientas con todas las opciones habituales en la reproducción de un video, más las opciones y utilidades que caracterizan el propósito de la creación de esta aplicación.

Visualmente, el diseño de la interfaz quedaría del siguiente modo:

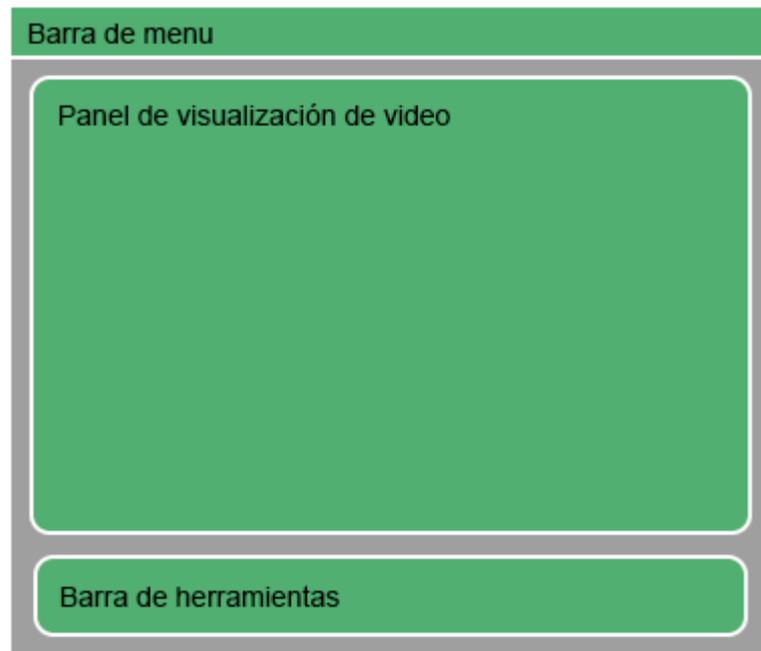


Figura 8.3 Diseño de la interfaz en la primera iteración

### 8.3.2. LA EJECUCIÓN MULTITHILO

Para la reproducción de archivos de video se ha hecho uso de la librería *OpenCV*. Siguiendo varios ejemplos incluidos en la misma, se optó por plasmar las imágenes generadas por la librería en un panel de la librería *wxWidgets* que forma parte de la ventana de la aplicación.

Así mismo, se ha incluido la ejecución de un segundo hilo de ejecución (*Entry thread*) que busca la siguiente imagen a mostrar en el panel de visualización, mientras el primer hilo se ocupa de la ejecución de los eventos.

También se incluyen botones para controlar la reproducción de distintas maneras: pausar, reproducir, avanzar y retroceder un solo *frame*. Estos botones utilizan la tabla de eventos de *wxWidgets* para modificar el funcionamiento normal del segundo hilo.

La ejecución del hilo secundario comienza cuando el usuario lanza el evento para cargar el archivo de video. Tras lo cual un cuadro de diálogo pedirá la ruta del archivo y lo

cargará en la clase `VideoCapture` de `OpenCV`. Esta clase se ocupará de manejar los pormenores de la operación mientras el hilo se ocupa de encontrar el *frame* apropiado a capturar según la operación que estemos realizando con los botones de la barra de herramientas. El hilo siempre permanece en un continuo bucle, incluso cuando la reproducción del video se pausa por el motivo que sea.

Tras la captura del fotograma tiene lugar el tratamiento de flujo óptico por parte de la clase `OpticalFlow`, que se ha implementado como parte de esta aplicación. Se procederá a dicho tratamiento solamente si la aplicación cumple dos condiciones: La primera es que el botón para aplicar la función de Flujo óptico esté activada en la barra de herramientas. Y la segunda es que haya algún punto marcado en la imagen sobre el que aplicar la función.

La función de flujo óptico, tal y como se explica en el capítulo 4, consiste en destacar elementos remarcables en el *frame* anterior y buscar coincidencias en el *frame* actual. Tras lo cual, se devuelve la nueva localización de los puntos para el *frame* actual y se almacena en su estructura correspondiente.

Hecho esto, se almacena la imagen obtenida del *frame* con un doble propósito. El primero es servir de medio para que la función que se ocupa de actualizar el panel de visualización con el *frame* correspondiente obtenga la imagen. El segundo propósito es servir como referencia a la función de flujo óptico durante la siguiente iteración. De esta forma la información no se pierde entre iteraciones.

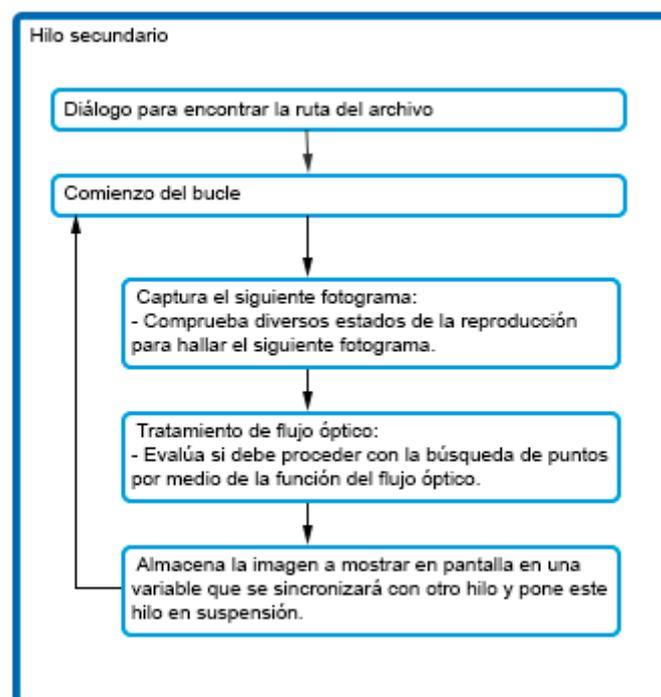


Figura 8.4 Esquema de ejecución del hilo secundario

Mientras que la ejecución del hilo secundario es sencilla y lineal, tal y como se muestra en la figura anterior; la ejecución del hilo principal está dominada por los eventos que se lanzan desde la tabla de eventos de *wxWidgets*.

### 8.3.3. LA ESTRUCTURA DE ALMACENAMIENTO

El seguimiento de las marcas puestas en el animal se realiza aplicando la función de flujo óptico, incluida como parte de la librería *OpenCV*. Pero antes se ha creado una nueva clase *pointhandler* para almacenar y manipular todos los puntos que el usuario marca en la pantalla. Para ello se ha dispuesto del siguiente diseño de datos.

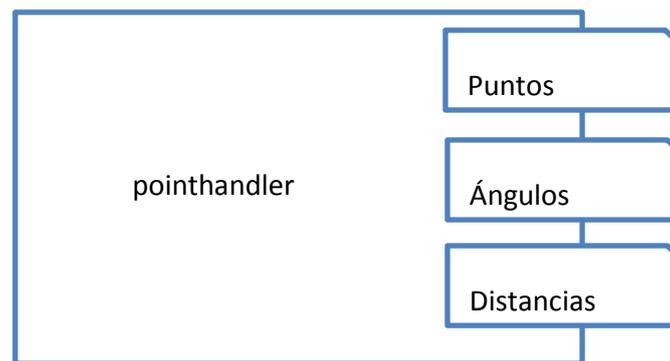


Figura 8.5 Modelo de diseño de datos para los puntos

Estos puntos se almacenan en una estructura *multimap*, la cual es una estructura binaria de datos incluida en la librería *standard de C++*. Dicha estructura consta de una clave por la que se realizan búsquedas y un tipo variable de datos. En el caso particular que nos ocupa la Clave de un punto en concreto será el *frame* de la reproducción en el que se capturó el punto. La estructura *multimap* devolverá todos los puntos asociados a la clave cuando se realiza una búsqueda, lo cual facilita la interacción a la hora de mostrar los puntos de un *frame* en pantalla. El segundo elemento de la estructura *multimap* consiste en una instancia de una estructura *valor3puntos* que contiene toda la información relevante a ese punto. Por ejemplo, si se trata de un vértice entre dos segmentos marcará los puntos con los que está relacionado, así como el ángulo entre ambos segmentos.

```

typedef struct valor3puntos{
    wxPoint pto;
    float valor;
    wxPoint ant;
    wxPoint post;
    int modo;
}
  
```

Figura 8.6 Implementación de la estructura *valor3puntos*

Los campos `pto`, `ant` y `post` son valores de tipo `wxPoint` que marcan un punto en la imagen. Siendo `pto` el punto central, y en el caso de tratarse de un vértice los puntos `ant` y `post` señalan los segmentos a los que está conectado.

El campo `valor` indica el valor dado en ese punto. Cuando se trata de un vértice este valor será el ángulo resultante entre los segmentos **anterior (pto-ant)** y **posterior (pto-post)**.

El campo `modo` puede tomar dos valores, 0 o 1, cuya utilidad consiste en mostrar u ocultar el punto durante la reproducción sin eliminarlo.

La capacidad para almacenar todos estos puntos se ha añadido en el menú superior. Se ha usado la librería `wxJSON` junto a las librerías nativas de `wxWidgets` para almacenar los datos en un archivo de texto. Un ejemplo de dicha estructura:

```
{
  "segmentos" : { "puntos" : [{"x" : 492, "frame" : 11, "y" : 65}],
  "cantidad" : 1},
  "angulos" : { "coordenadas" : [
    {
      "frame" : 11,
      "modo" : 1,
      "punto" : {"x" : 498, "y" : 144},
      "anterior" : {"x" : 492, "y" : 65},
      "posterior" : {"x" : 573, "y" : 226}
    }
  ]},
  "cantidad" : 1
},
"razon" : 1
}
```

---

#### 8.3.4. PRUEBAS

La implementación de todos estos aspectos de la aplicación se realiza de manera modular y progresiva, comprobando con cada nuevo elemento que todos funcionan de forma correcta.

Durante las pruebas de caja negra realizadas en esta fase surgieron varios problemas. El primero de ellos, que la librería `OpenCV` no era capaz de reproducir todos los tipos de archivos de video. Algo que, tras muchas pruebas para tratar de conciliar la librería con los archivos problema, se ha tener en cuenta a la hora usar la herramienta.

El segundo problema surge a raíz de la variabilidad en la calidad de video durante la reproducción, produciendo falsos positivos en la detección de puntos o, en ocasiones, haciéndolos desaparecer. Este problema ya se comentó en las pruebas realizadas anteriormente para la explicación del algoritmo de flujo óptico. La calidad de imagen de los

videos, la iluminación y lo remarcable que sean los marcadores resultan cruciales a la hora de realizar el seguimiento del movimiento.

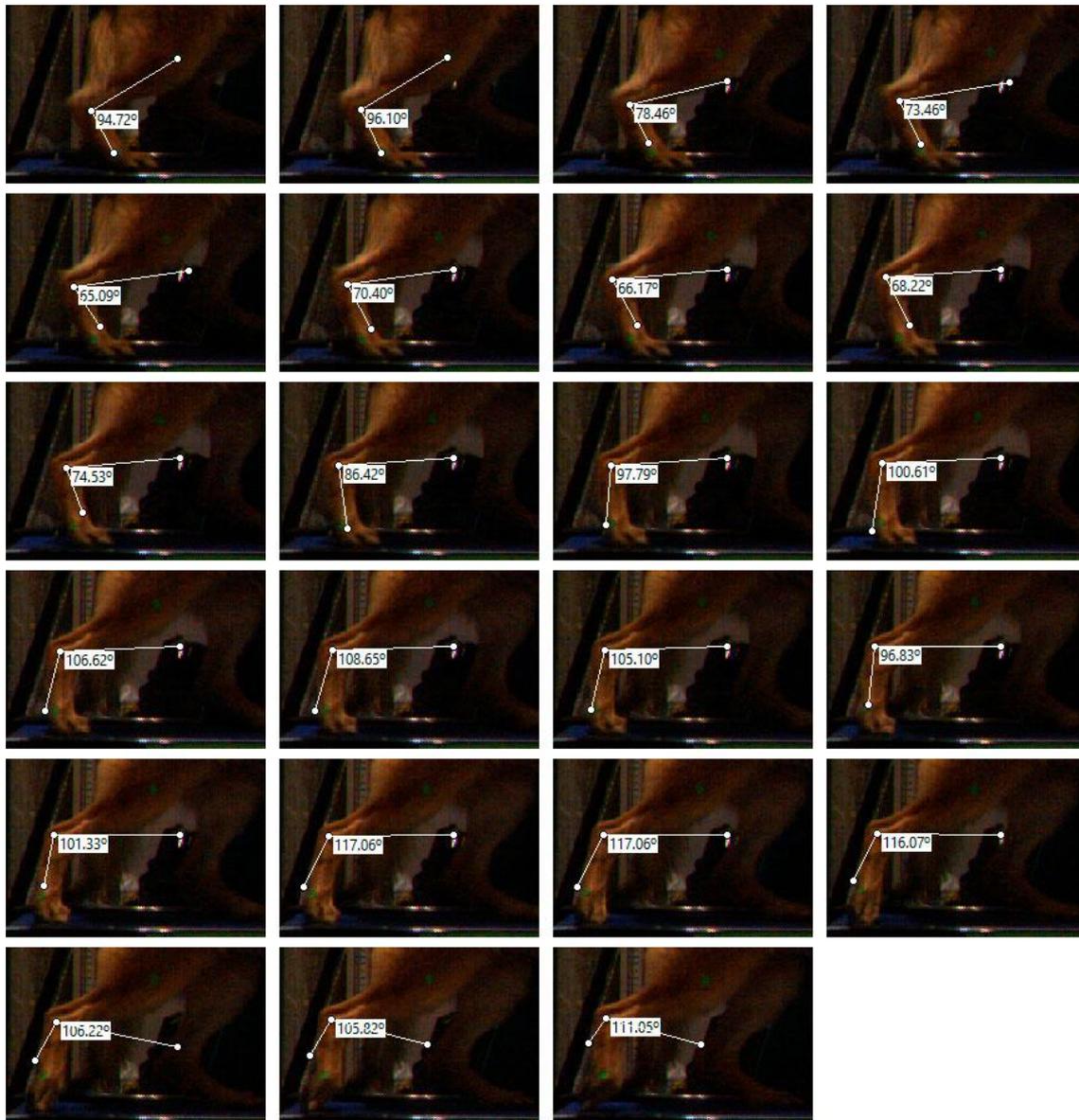


Figura 8.7 Falso positivo debido a una iluminación pobre

El tercer problema es que la aplicación no se encontraba correctamente implementada de forma que mantuviese separados el hilo de la interfaz del hilo de la reproducción del video. Es decir, durante la reproducción de un archivo de video, la interfaz permanecía inoperativa a los comandos del usuario. Para solucionar este problema se optó por hacer uso de las pruebas de caja blanca para hallar el problema exacto.

Además, dado lo importante de esta primera iteración para el resto del desarrollo, se ha realizado un seguimiento profundo de todas las variables críticas durante la ejecución multihilo de la aplicación. De esta forma se adquiere un conocimiento preciso de cómo

reacciona la aplicación durante su ejecución a los cambios que se le vayan a aplicar en futuras iteraciones.

La prueba de caja blanca reveló un mal uso en la aplicación de la librería multihilo de wxWidgets, en el que una vez el hilo de video comenzaba a reproducirse, no devolvía el control a la interfaz al final de cada fotograma. La solución del problema pasa por usar un buffer con el que mantener las imágenes que se vayan leyendo y la visualización de la imagen a través de un hilo distinto.

Finalmente, la interfaz de la aplicación quedaría en la ventana de la siguiente forma.

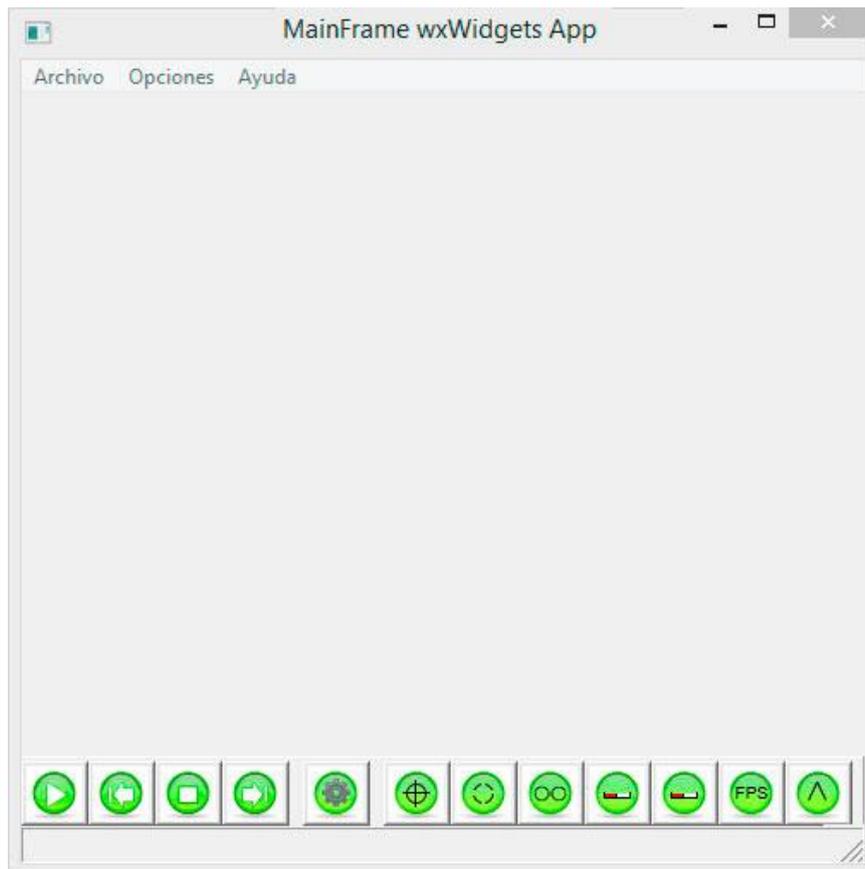


Figura 8.8 Captura de la interfaz en la primera iteración

## 8.4. SEGUNDA ITERACIÓN

Para la segunda iteración los objetivos marcados consistieron, además de solventar cualquier problema encontrado durante la iteración anterior, en la inclusión de medios por los que el usuario pudiera observar los datos mostrados en la aplicación. Por ello se propuso esta actualización en los objetivos:

- Gráficas para mostrar la evolución en el tiempo de los datos.
- Barra de búsqueda para el video.
- Cambiar un ángulo por su ángulo conjugado.
- Exportar la información a formato Excel y CSV.
- Guardar la gráfica como imagen.

### 8.4.1. ACTUALIZACIÓN DE LA INTERFAZ

La inclusión de nuevos elementos obliga a replantear la manera en la que se visualizan estos nuevos elementos. Se consideró el uso de distintas ventanas emergentes y flexibles en las que se mostrase la información, de forma que el usuario pueda acomodar la información en la pantalla como le resulte más cómodo.

Sin embargo, se ha optado por una solución visualmente más sólida, de forma que el usuario tenga en la ventana todo lo necesario para trabajar. Para ello se ha optado por realizar cambios drásticos en la interfaz, añadiendo los nuevos elementos junto a los anteriores. El diseño de la interfaz, finalmente, quedaría de la siguiente manera. En verde se resaltan las novedades, mientras que los cuadros de puntos indican modificaciones.



Figura 8.9 Diseño de la interfaz en la segunda iteración

### 8.4.2. GRÁFICA

Dado que uno de los objetivos es mostrar la información pertinente sobre los datos que se están analizando en la reproducción, se ha dispuesto que el usuario pueda visualizar

estos datos, de medida de ángulos y distancias, en una gráfica que también muestra el avance de la reproducción.

Para dejar más clara la información que se muestra en la gráfica también se han añadido sendas tablas con los datos contenidos de las variables de ángulos y distancias, en función del *frame* de la reproducción.

Para las gráficas se ha hecho uso de la librería *wxMathPlot*, una librería que añade funcionalidad de trazado científico, estadístico y matemático a *wxWidgets*.

---

#### 8.4.3. EXPORTACIÓN DE DATOS

Para facilitar el uso de la información obtenida por la aplicación, se ofrece al usuario la opción para exportar datos a *Microsoft Excel* y al formato *CSV*, un tipo de documento en formato abierto sencillo para representar los datos en una tabla en las que las columnas se separan por comas (o punto y coma) y las filas por salto de línea.

La librería *wxWidgets* ofrece las herramientas necesarias para crear dichas funcionalidades en la aplicación. La parte de la librería dedicada al *wxAutomation* resulta bastante directa a la hora de crear los objetos y las instancias necesarias, así como para evaluar las distintas propiedades que se han de manejar para exportar a Excel.

Por otra parte, la librería *wxWidgets* también proporciona las funciones para tratar con archivos binarios y de texto, de manera que la creación de un archivo estructurado para el formato *CSV* resulta muy sencilla.

Además de exportar datos a un formato distinto, también se ha implementado la opción de capturar la imagen de la gráfica y guardarla como un archivo de imagen. Implementar esta función resulta sencilla puesto que la propia librería *wxMathPlot* ya posee una función para realizar esa acción.

---

#### 8.4.4. LA BARRA DE BÚSQUEDA

Con el objetivo de facilitar el cambio en la posición en la que se encuentra la reproducción de video se ha buscado implementar una barra de búsqueda tal y como se ven en los reproductores de video normales. De esta forma, y de manera ya intuitiva, el usuario puede situarse en el *frame* concreto en el que desea pausar o comenzar la reproducción.

La barra de búsqueda se ha implementado usando el elemento *wxSlider* de *wxWidgets*. Se ha añadido a ambos extremos unas etiquetas para indicar el *frame* actual de la reproducción y el número total de *frames* que tiene el video.

#### 8.4.5. EL ÁNGULO CONJUGADO

Son ángulos conjugados aquellos cuyas medidas suman  $360^\circ$ . Esto significa que cuando hablamos de un ángulo conjugado hablamos de un ángulo al que se le ha restado  $360^\circ$ .

La implementación del ángulo conjugado tuvo una complicación adicional, ya que de entrada la intención fue incorporarlo de manera intuitiva y directa por medio de un menú contextual. Pero por complicaciones con la librería *wxWidgets* se ha optado por un botón con el que cambiar cualquier punto en el que haya una medida angular por su correspondiente conjugado. A consecuencia de esto se ha añadido un tercer modo a *valor3puntos*. Este tercer modo (indicado con el número 2) permite leer desde la aplicación cuando el ángulo que se busca es el conjugado y realiza el correspondiente cálculo.

#### 8.4.6. PRUEBAS

No se encontraron problemas con la interfaz de la aplicación durante la fase de pruebas de caja negra. Sin embargo, se propuso cambiar el diseño de los botones para que fuesen más claros en sus funciones.

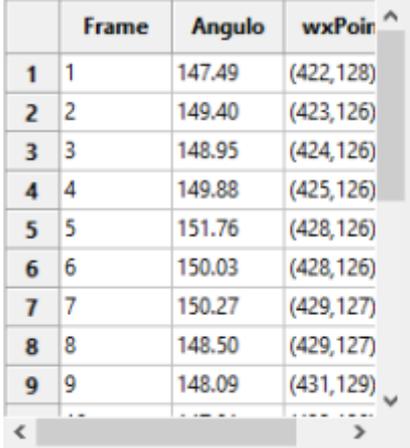
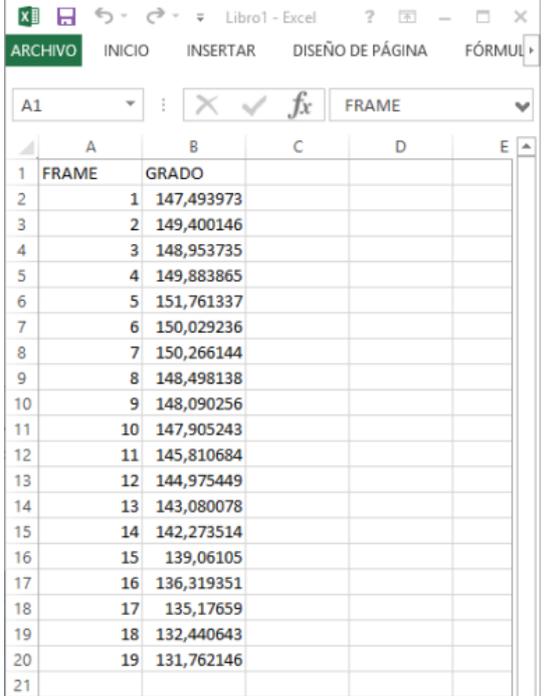
Las pruebas del ángulo conjugado dieron resultados satisfactorios en cuanto a su funcionalidad, aunque el problema de querer implementarlo como menú contextual, y por ende más intuitivo, persiste.



Figura 8.10 Ángulos conjugados

Una vez marcado el ángulo como conjugado, basta con permitir a la aplicación que rastree de manera normal el ángulo para que todos los resultados a partir del fotograma en el que se ha marcado sean también conjugados.

Durante las pruebas para la exportación de datos tanto a *Microsoft Excel* como a *CSV* no surgieron problemas. A continuación se muestra un ejemplo de cómo los datos se visualizan en la aplicación, y como su exportación a los distintos documentos proporcionan los datos necesarios para trabajar en ellos desde otras herramientas.

<p>Datos visualizados en la aplicación</p>	 <table border="1"> <thead> <tr> <th></th> <th>Frame</th> <th>Angulo</th> <th>wxPoin</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>147.49</td><td>(422,128)</td></tr> <tr><td>2</td><td>2</td><td>149.40</td><td>(423,126)</td></tr> <tr><td>3</td><td>3</td><td>148.95</td><td>(424,126)</td></tr> <tr><td>4</td><td>4</td><td>149.88</td><td>(425,126)</td></tr> <tr><td>5</td><td>5</td><td>151.76</td><td>(428,126)</td></tr> <tr><td>6</td><td>6</td><td>150.03</td><td>(428,126)</td></tr> <tr><td>7</td><td>7</td><td>150.27</td><td>(429,127)</td></tr> <tr><td>8</td><td>8</td><td>148.50</td><td>(429,127)</td></tr> <tr><td>9</td><td>9</td><td>148.09</td><td>(431,129)</td></tr> </tbody> </table>		Frame	Angulo	wxPoin	1	1	147.49	(422,128)	2	2	149.40	(423,126)	3	3	148.95	(424,126)	4	4	149.88	(425,126)	5	5	151.76	(428,126)	6	6	150.03	(428,126)	7	7	150.27	(429,127)	8	8	148.50	(429,127)	9	9	148.09	(431,129)																				
	Frame	Angulo	wxPoin																																																										
1	1	147.49	(422,128)																																																										
2	2	149.40	(423,126)																																																										
3	3	148.95	(424,126)																																																										
4	4	149.88	(425,126)																																																										
5	5	151.76	(428,126)																																																										
6	6	150.03	(428,126)																																																										
7	7	150.27	(429,127)																																																										
8	8	148.50	(429,127)																																																										
9	9	148.09	(431,129)																																																										
<p>Datos exportados a un documento <i>Microsoft Excel</i></p>	 <table border="1"> <thead> <tr> <th></th> <th>FRAME</th> <th>GRADO</th> </tr> </thead> <tbody> <tr><td>2</td><td>1</td><td>147,493973</td></tr> <tr><td>3</td><td>2</td><td>149,400146</td></tr> <tr><td>4</td><td>3</td><td>148,953735</td></tr> <tr><td>5</td><td>4</td><td>149,883865</td></tr> <tr><td>6</td><td>5</td><td>151,761337</td></tr> <tr><td>7</td><td>6</td><td>150,029236</td></tr> <tr><td>8</td><td>7</td><td>150,266144</td></tr> <tr><td>9</td><td>8</td><td>148,498138</td></tr> <tr><td>10</td><td>9</td><td>148,090256</td></tr> <tr><td>11</td><td>10</td><td>147,905243</td></tr> <tr><td>12</td><td>11</td><td>145,810684</td></tr> <tr><td>13</td><td>12</td><td>144,975449</td></tr> <tr><td>14</td><td>13</td><td>143,080078</td></tr> <tr><td>15</td><td>14</td><td>142,273514</td></tr> <tr><td>16</td><td>15</td><td>139,06105</td></tr> <tr><td>17</td><td>16</td><td>136,319351</td></tr> <tr><td>18</td><td>17</td><td>135,17659</td></tr> <tr><td>19</td><td>18</td><td>132,440643</td></tr> <tr><td>20</td><td>19</td><td>131,762146</td></tr> </tbody> </table>		FRAME	GRADO	2	1	147,493973	3	2	149,400146	4	3	148,953735	5	4	149,883865	6	5	151,761337	7	6	150,029236	8	7	150,266144	9	8	148,498138	10	9	148,090256	11	10	147,905243	12	11	145,810684	13	12	144,975449	14	13	143,080078	15	14	142,273514	16	15	139,06105	17	16	136,319351	18	17	135,17659	19	18	132,440643	20	19	131,762146
	FRAME	GRADO																																																											
2	1	147,493973																																																											
3	2	149,400146																																																											
4	3	148,953735																																																											
5	4	149,883865																																																											
6	5	151,761337																																																											
7	6	150,029236																																																											
8	7	150,266144																																																											
9	8	148,498138																																																											
10	9	148,090256																																																											
11	10	147,905243																																																											
12	11	145,810684																																																											
13	12	144,975449																																																											
14	13	143,080078																																																											
15	14	142,273514																																																											
16	15	139,06105																																																											
17	16	136,319351																																																											
18	17	135,17659																																																											
19	18	132,440643																																																											
20	19	131,762146																																																											

Datos exportados como documento CSV	<pre> FRAME;ANGULO 1;147.493973 2;149.400146 3;148.953735 4;149.883865 5;151.761337 6;150.029236 7;150.266144 8;148.498138 9;148.090256 10;147.905243 11;145.810684 12;144.975449 13;143.080078 14;142.273514 15;139.061050 16;136.319351 17;135.176590 18;132.440643 19;131.762146                     </pre>
-------------------------------------	--

Figura 8.11 Datos originales y exportados a Excel y CSV

Durante la revisión con el usuario salió a la luz que la herramienta utilizada para el cálculo de la zancada del animal requería una actualización en eficiencia para que fuera realmente útil a la hora de trabajar con ella. Además, se planteó la posibilidad de seguir varios ángulos al mismo tiempo.

La implementación de la interfaz daría como resultado la siguiente ventana de trabajo, puede observarse a la izquierda la interfaz ya conocida en la iteración anterior, mientras que a la derecha se encuentran los nuevos añadidos de la gráfica y las tablas.

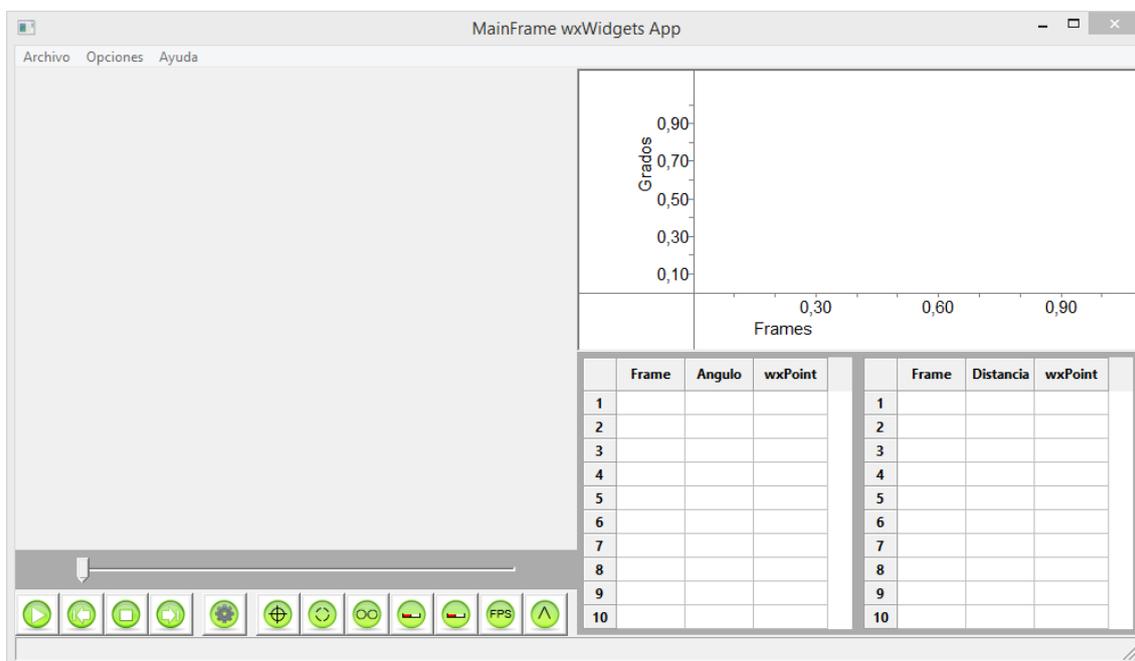


Figura 8.12 Captura de la interfaz en la segunda iteración

## 8.5. TERCERA ITERACIÓN

En la tercera iteración finalmente se introducen los últimos conceptos dispuestos para el usuario, además de mejorar el cálculo de las zancadas del animal tal y como se concluyó en las pruebas de la iteración anterior:

- Crear un archivo de video a partir de imágenes en secuencia.
- Una caja de texto sincronizada con el video para anotar información importante sobre el video en reproducción.
- Una herramienta para ayudar en el cálculo de la zancada del animal, conectado junto a las gráficas para mostrar la evolución del paso.
- Un mecanismo por el que se pueda calcular el tiempo exacto entre dos momentos de la reproducción.
- Seguir varios ángulos al mismo tiempo.
- Cambiar el color del trazado en la gráfica para seguir dichos ángulos.

### 8.5.1. ACTUALIZACIÓN DE LA INTERFAZ

La nueva iteración requiere, nuevamente, un cambio bastante importante en la interfaz. De entrada las notas pasarán a formar parte del cuadro donde antes se almacenaba la tabla de distancia, mientras que la gráfica y tabla cambiarán alternándose usando un mecanismo de pestañas donde se seleccionará la gráfica y tabla pertinentes.

Además, se han añadido las nuevas herramientas a la barra de menú y barra de herramientas además de un rediseño en los botones de la barra de herramientas.

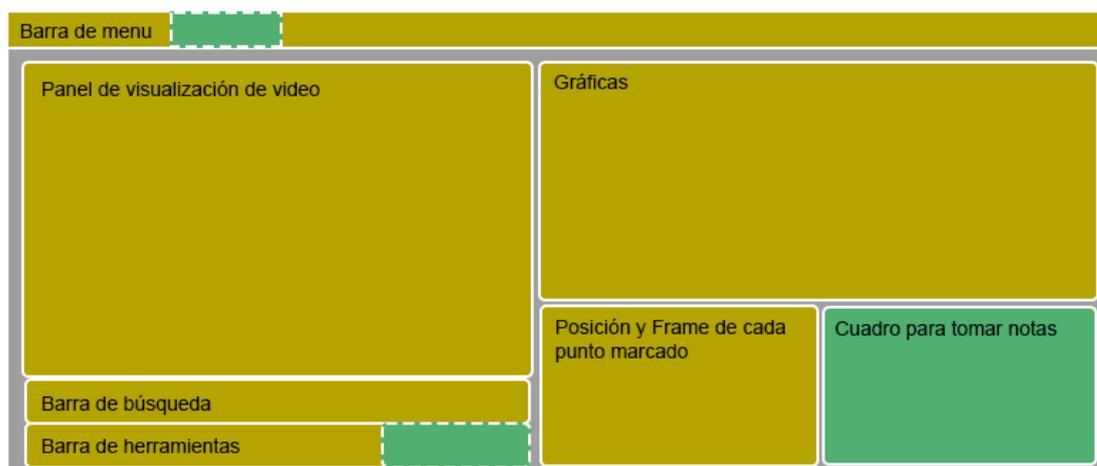


Figura 8.13 Diseño de la interfaz en la tercera iteración

---

### 8.5.2. MONTAJE DE VIDEO A PARTIR DE IMÁGENES EN SECUENCIA

En el caso de tener una serie de imágenes en secuencia, por ejemplo a consecuencia de separar un archivo de video problemático, se ha implementado la funcionalidad de crear un archivo de video a partir de ellas. Para ellos se ha hecho uso de la librería *VideoWriter*.

Su función consiste en un bucle que encuentra consecutivamente todos los archivos a partir de la imagen inicial, y los une aplicando la codificación indicada por el usuario. El resultado es un nuevo archivo de video formado por los fotogramas seleccionados.

La cantidad de códecs disponibles para la aplicación depende completamente de la máquina en la que se está ejecutando la aplicación, esto puede variar en gran medida desde tener solamente los códecs esenciales incluidos en el sistema operativo, hasta tener multitud de ellos por trabajar de manera frecuente con programas de edición de video.

---

### 8.5.3. EL CUADERNO DE NOTAS

Uno de los nuevos añadidos es el cuaderno de notas para que el usuario pueda escribir y almacenar notas relativas al *frame* en uso.

Este cuaderno de notas se ha implementado añadiendo una caja de texto con un selector que marca el *frame* en el que se ha tomado la nota, y un vector de ristas para almacenar todas las notas. Estos datos se almacenarán pertinentemente en el campo adecuado a la hora de guardar la información.

Además, al pulsar sobre una nota en concreto, la reproducción nos llevará hasta el fotograma donde se ha guardado la anotación.

---

### 8.5.4. NUEVAS HERRAMIENTAS

La herramienta para calcular la zancada del animal se realiza de dos maneras y se divide en dos partes. El cálculo puede realizarse de forma automática o manual.

La forma manual sencillamente consiste en marcar en la imagen el punto y *frame* en el que el animal empieza y acaba la zancada. La información se calcula automáticamente y se muestra en una gráfica correspondiendo a las zancadas del animal.

En la forma automática se establece con un punto la pata del animal a rastrear y con una línea el umbral por el que se considera que ese punto está levantando o apoyando la pata. Esta información se calcula automáticamente, igual que en el proceso manual, y se muestra en la gráfica correspondiente.

En la siguiente figura se muestran los botones para activar las distintas herramientas. De izquierda a derecha: Botón para la forma manual, Botón para marcar el

umbral de la forma automática, y por último un botón que permite medir distancias pulsando y arrastrando a modo de regla.



Figura 8.14 Botones para medir las distancias

Para calcular el tiempo transcurrido entre dos puntos de la reproducción se han añadido 3 botones a la barra de herramientas. Dos para marcar el principio y fin del tiempo transcurrido y uno para mostrar cuanto tiempo es. Previamente hace falta calibrar los *frames* por segundo en el menú de opciones.



Figura 8.15 Botones para medir segmentos de tiempo

#### 8.5.5. ACTUALIZACIÓN DE LA GRÁFICA

Con el objetivo de permitir el trazado de varios puntos en la misma gráfica y distinguirlos entre ellas se ha añadido un botón en la barra de herramientas para personalizar el color con el que se ven los trazos en el panel de gráficas. De forma que si se tienen varios valores en seguimiento al mismo tiempo, puedan diferenciarse en la gráfica.

Este trazado en la librería *wxMathPlot* requería un cambio en la forma en la que se almacenaban los puntos hasta ahora. A la estructura `valor3puntos` se le ha añadido un nuevo elemento:

```
typedef struct valor3puntos{
    ...
    int grupo;
}
```

Figura 8.16 Modificación en la implementación de `valor3puntos`

El valor `grupo` es un identificador que caracteriza a una serie de puntos durante la reproducción del video como el mismo punto en movimiento, de forma que sea posible discernir un punto concreto del resto dentro de la aplicación.

#### 8.5.6. LA ESTRUCTURA DE ALMACENAMIENTO

La inclusión de las notas se refleja en el almacenamiento de la estructura del archivo *JSON* tal y como se muestra en el siguiente ejemplo, como un nuevo tipo de dato a almacenar, junto a los ángulos y la distancia.

```
{
  [...]
```

```

"notas" : { "entrada" : [{"frame" : 11, "texto" : "Comienza\n"}],
            "cantidad" : 1},
[...]
}

```

### 8.5.7. PRUEBAS

Las pruebas realizadas para la creación de videos a partir de una secuencia de imágenes son exitosas. Siendo capaz de crear videos a partir de mapas de bits hasta imágenes en formato *jpeg*. Esta funcionalidad resulta especialmente útil a la hora de convertir videos en formatos que la aplicación ha tenido problemas a la hora de reproducir.

Al igual que ocurría en la iteración anterior. La iluminación intermitente causa problemas a la hora de seguir los puntos, creando falsos positivos y localizando los puntos de seguimiento incluso fuera del fotograma. En el siguiente ejemplo se hizo un seguimiento de varios puntos marcados a lo largo del lomo y la pata del animal. Puede observarse como la luz cambia drásticamente y provoca que los puntos se dispersen por todo el fotograma.

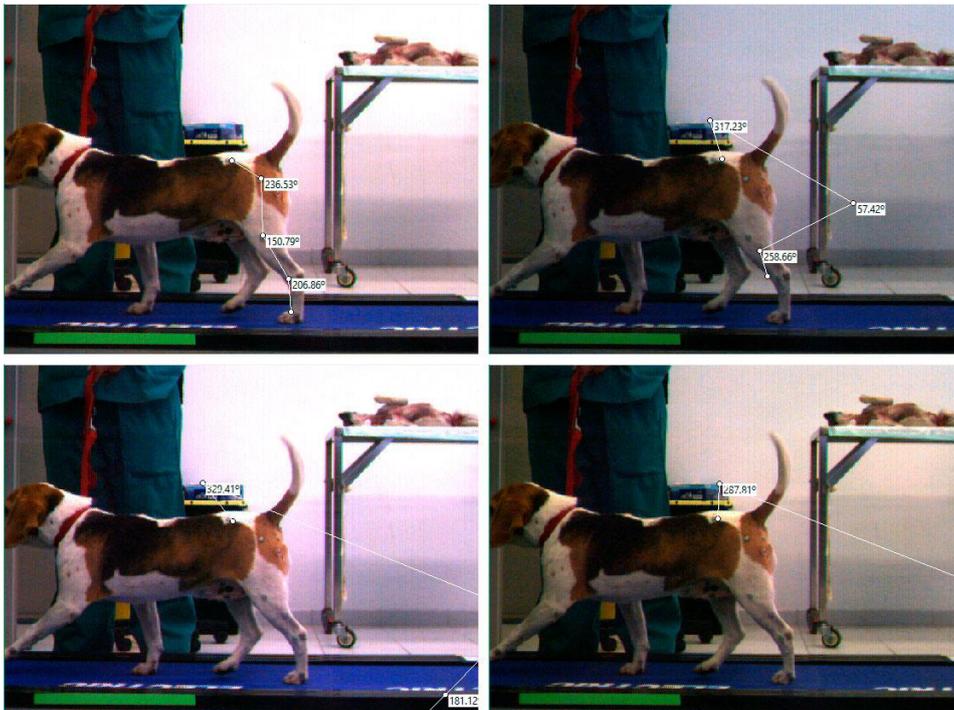


Figura 8.17 Falso positivo debido a la iluminación

Por otro lado, las pruebas realizadas para calcular la distancia de zancada del animal resultan más sencillas de seguir. Mientras que la efectividad de la utilidad automática para el cálculo de la zancada depende igualmente del algoritmo de Lucas Kanade, su implementación manual deja en manos del usuario el saber cuándo el animal está apoyando su peso sobre la pata para determinar cuándo empieza y acaba la zancada. Dando como

resultado el siguiente ejemplo en el cual se ha calculado la zancada del animal en tres ocasiones.

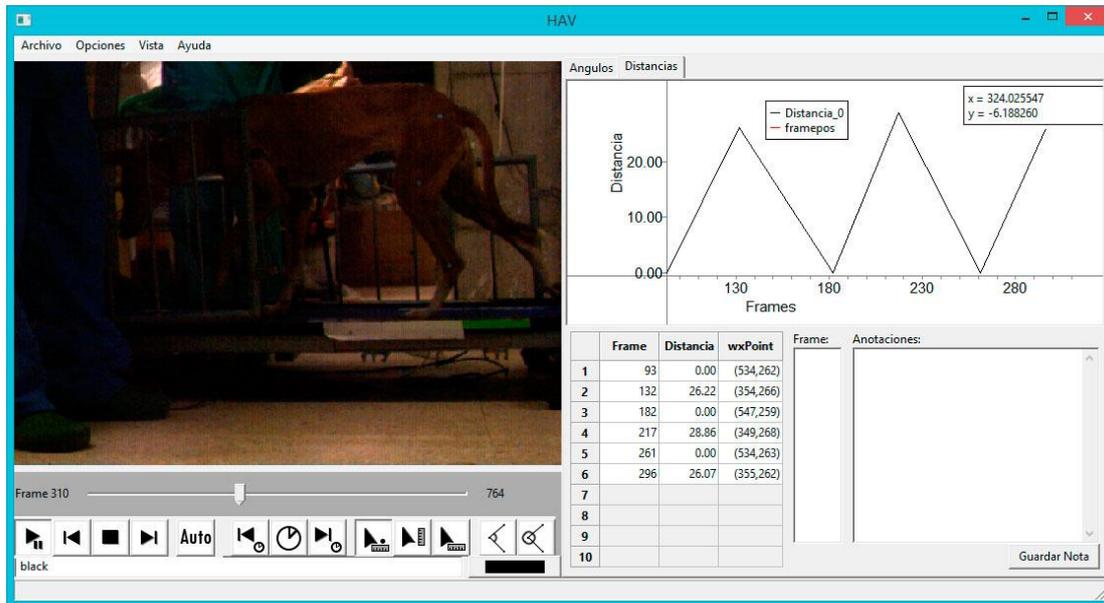


Figura 8.18 Visualización de distancia en el gráfico

En la captura anterior se muestran los últimos cambios realizados en la interfaz. Puede observarse que ambas pestañas para cambiar entre Ángulos y Distancias se encuentran en la mitad derecha de la aplicación, mientras que en la mitad izquierda se ha renovado el estilo gráfico de los botones, además de la inclusión de las nuevas herramientas ya mencionadas.

Las pruebas realizadas para medir el tiempo de diferencia entre dos fotogramas resultaron igual de sencillas que la evaluación de las distancias. En ambos casos es necesario calibrar previamente la medida de referencia para realizar el cálculo. Una vez calibrado y marcados ambos fotogramas de principio y fin del segmento, se obtiene por medio de una sencilla operación matemática cuanto es la duración, en segundos, del segmento.

Finalmente, como ejemplo de la aplicación en pleno funcionamiento, aquí se muestra una nueva captura con algunas anotaciones, ángulos dibujados en la gráfica, junto a un video pausado durante su reproducción. Las anotaciones procuran un acceso inmediato al fotograma relevante de forma que facilita en gran medida la navegación por el video.

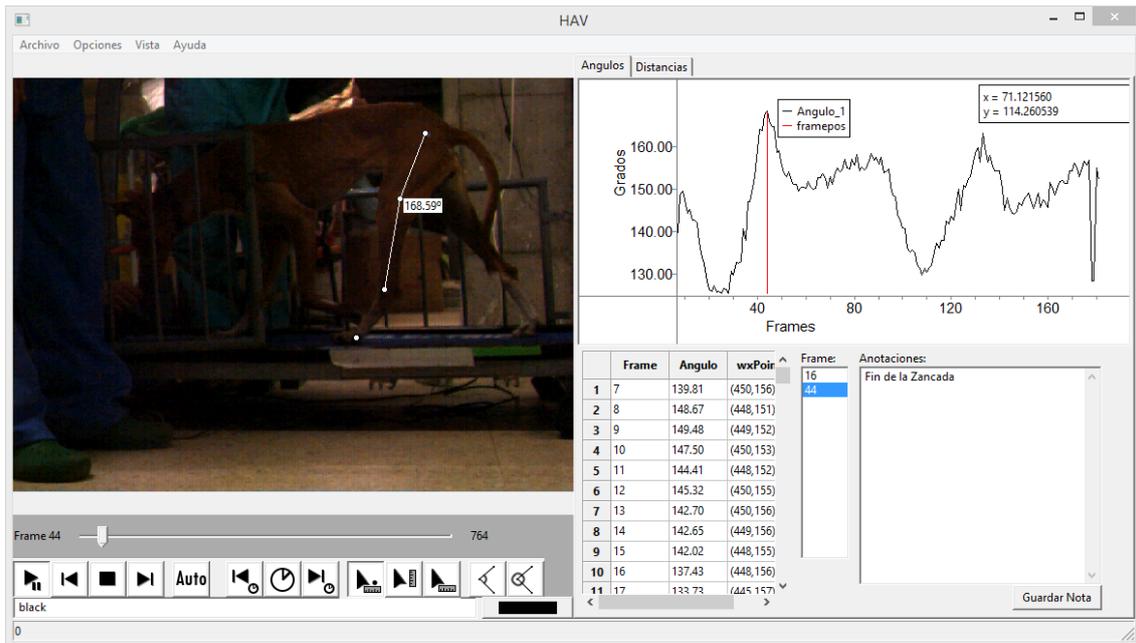


Figura 8.19 Captura de la aplicación en la tercera iteración

## 8.6. COMPONENTES DE LA APLICACIÓN

A continuación se lista la organización de los archivos de la aplicación, junto a una breve descripción de la funcionalidad de cada uno de los elementos.

EntryThread.cpp	La función Entry se ejecuta en bucle para capturar cada fotograma de video de forma consecutiva, y procesar con la función de flujo óptico los puntos marcados en cada fotograma.
Event_types.cpp	Tabla de eventos para <i>wxWidgets</i> . Los eventos son el principal medio de control en las aplicaciones de <i>WxWidgets</i> .
Filltable.cpp	La función Filltable se ocupa de actualizar las tablas con la información procesada en el bucle principal.
OnMenuItems.cpp	Contiene las funciones que se ejecutan a partir de las opciones en el menú de la parte superior de la aplicación.
OnSlider.cpp	Contiene las funciones que actualizan la barra de búsqueda durante la reproducción del video.
OnToolbox.cpp	Contiene todas las funciones que se ejecutan a partir de los botones de la caja de herramientas, situada debajo del panel de video.
OnVideoUpdate.cpp	Llamada desde Entrythread, OnVideoUpdate actualiza la imagen visualizada en el panel de video.
Opticalflow.cpp Opticalflow.h	Pequeña librería que contiene las funciones para el manejo del flujo óptico.
Pointhandler.cpp Pointhandler.h	Librería que almacena, manipula y elimina la información sobre los puntos, ángulos y distancias para su visualización en la reproducción.
wxVideoGui.cpp wxVideoGUI.h wxVideoPlayer.cpp	Librería para la interfaz de usuario de la herramienta, realizada en <i>wxWidgets</i> .

## 8.7. TÉCNICAS DE PRUEBAS

Para cumplir con el objetivo de las pruebas se utilizaron diversas técnicas que ponen a prueba distintos aspectos del software implementado. En concreto las conocidas técnicas de la caja negra y la caja blanca. A continuación se hará una breve descripción de en qué consiste cada tipo de prueba y se realizará una recapitulación de todas las pruebas realizadas durante el desarrollo de la aplicación.

La caja negra es una técnica que evalúa los requisitos funcionales del software. Permite encontrar casos en los que el módulo no se atiene a su especificación, es decir, no produce los resultados que se esperan. Para ello se realizan pruebas funcionales en las que el probador únicamente suministra los datos de entrada y estudia la salida que se obtiene sin preocuparse por lo que pueda ocurrir dentro del módulo.

Los errores que intentan encontrar las pruebas de la caja negra pertenecen a alguna de las siguientes categorías: funciones incorrectas o ausentes, errores de interfaz, errores en estructuras de datos o en accesos a bases de datos externas, errores de rendimiento, errores de inicialización, y errores de terminación.

Tanto el estudio previo como las pruebas de la primera iteración dieron como resultado que el principal problema al que se enfrenta la aplicación procede de mantener las condiciones necesarias para que el algoritmo de Lucas Kanade pueda aplicarse al video. Otro resultado de las pruebas de la primera iteración consiste en la incompatibilidad de ciertos formatos de video con el módulo de la librería OpenCV.

Por la parte técnica, la prueba de caja negra sacó a la luz un problema con la ejecución de hilos que dejaba la aplicación inoperable mientras un video se encontraba en reproducción. Problema que se resolvió realizando la prueba de caja blanca.

Durante la segunda iteración se resaltó la importancia de como el diseño de los botones en la interfaz no reflejaban de forma clara su función, por lo que se propuso su cambio por unos más apropiados. Además, se encontró que la función para calcular la zancada no estaba implementada de forma que ayudase, por lo que también requería una revisión.

Las pruebas de la tercera iteración reflejaron lo que ya sabíamos sobre los prerrequisitos para el video, pero también lo útil que han resultado las notas para marcar los inicios y finales de segmento y navegar así por la reproducción a los puntos relevantes.

La caja blanca es una técnica centrada en los detalles procedimentales del software. Consiste en escoger distintos valores de entrada para examinarlos en cada uno de los posibles flujos de ejecución de la aplicación y comprobar que devuelven los valores de salida adecuados. Dicho de otra forma, esta prueba garantiza que se recorran todos los caminos independientes de cada módulo: Cada decisión lógica, cada bucle y su límite operacional, cada estructura interna de datos.

Debido a lo exhaustivo de esta prueba, por lo general sólo se realiza en caso de que surjan errores durante las pruebas de la caja negra.

Las pruebas de caja blanca se realizaron solamente durante la primera iteración para tener un conocimiento más profundo de la aplicación y hallar donde se encontraba el problema de ejecución con los hilos. Gracias a estas pruebas las siguientes iteraciones sucedieron sin percances mayores a la hora de diseñar, implementar y testear el resto de funcionalidades y de su integración con el resto de la unidad.

Dicha prueba reveló que el hilo de reproducción de video dejaba inoperativa la interfaz de la aplicación a causa de un mal uso de las librerías multihilo de wxWidgets. Por lo que pudo buscarse una solución adecuada para su implementación durante la siguiente iteración.

## 9. RESULTADOS Y CONCLUSIONES

Las conclusiones a este proyecto de fin de carrera son el resultado de llevar a su término final las fases de estudio, análisis, diseño y desarrollo sobre el concepto que se había propuesto en un principio: El crear una aplicación que facilite la labor de los doctores en veterinaria a la hora de diagnosticar a los animales, tanto en el campo de la investigación y estudios clínicos, como de apoyo en la rehabilitación de los mismos.

Con el objetivo de analizar los resultados se ha dividido este apartado desde distintos puntos de vista.

### 9.1. RESULTADOS OBTENIDOS

Por supuesto, antes de hablar de las conclusiones es inevitable comenzar por comprobar si se han cumplido los objetivos marcados inicialmente: Una aplicación con un sistema de detección de movimiento para una secuencia de video capaz de realizar mediciones automáticas sobre los datos que se observen o se marquen en el reproductor.

En efecto se ha cumplido con estos objetivos, las librerías *OpenCV* y *wxWidgets* proporcionan las herramientas necesarias tanto para la interfaz, como para la reproducción de archivos multimedia y la integración de los métodos de flujo óptico; mientras que haciendo uso de librerías como *wxMathPlot* y *wxJSON*, así como de las librerías *standard de C++*, se ha logrado mantener el objetivo de la realización de los cálculos automáticos.

Además de cumplir con estos objetivos se ha trabajado en la medida de lo posible en proporcionar herramientas adicionales que complementen el propósito de la aplicación. De esta forma tenemos una aplicación que además de cumplir con los objetivos iniciales, también es capaz de almacenar y recuperar la sesión de trabajo, con notas, gráficas y ser capaz de exportar toda esta información a otros formatos, en caso de que sea necesario.

El estado actual de la aplicación significa que, bajo condiciones controladas, ya es posible realizar evaluaciones de animales en un entorno profesional, habiéndose implementado y probado durante las pruebas. Por tanto ya se podría utilizar esta aplicación para organizar experiencias piloto en los distintos campos que se tenía previsto para su uso, como por ejemplo en estudios clínicos de animales.

Pese a todo, este software no puede considerarse un producto final, sino como una base a partir de la cual se puedan desarrollar nuevas líneas de trabajo e investigación. Durante su realización siempre se dieron casos en los que la aplicación podía ampliarse, o surgían funcionalidades que podían implementarse para mejorar la experiencia del usuario. Muchas de estas ideas se han anotado para su inclusión en el apartado de Trabajo futuro.

## 9.2. CONCLUSIONES

El desarrollo de este proyecto de fin de carrera ha tenido como proceso de aprendizaje no sólo el perfeccionamiento del lenguaje de programación utilizado, sino también el aprendizaje de nuevas metodologías, estrategias y estructuras que se llevaron a la práctica durante este periodo, resolviendo de esta manera muchos de los obstáculos en el camino.

El uso de la librería *OpenCV* ha supuesto una experiencia muy educativa, poniendo en práctica de manera visual toda la teoría sobre los algoritmos de Shi-Tomasi y Lucas-Kanade de forma sencilla e intuitiva. La inclusión de un módulo independiente de toda interfaz visual ha sido una gran ayuda a la hora de la realización de pruebas sin la intervención del resto de librerías.

Aunque en este contexto resulta imperfecto debido a la calidad variable de los videos utilizados, el algoritmo Lucas-Kanade demuestra una gran flexibilidad en toda clase de proyectos basados en el seguimiento de movimiento. Por ello creo que vale la pena seguir investigando el uso del algoritmo en esta dirección.

Tratar con la librería de *wxWidgets* por el contrario ha supuesto un reto. Su diseño orientado a objetos no es el mejor que uno pueda ver. En ocasiones abusa de la utilización de macros para realizar operaciones, como la tabla de eventos, en favor de facilitar la codificación. También complica la labor de depuración, incluso para los propios desarrolladores de la librería.

Sin embargo, a pesar del esfuerzo por reunir las condiciones para facilitar la creación de una aplicación multiplataforma con las herramientas empleadas, lo cierto es que ninguna aplicación multiplataforma será tan eficiente como sus correspondientes aplicaciones desarrolladas de forma “nativa” para el sistema objetivo. Por este motivo creo que la aplicación se beneficiaría de su recreación en *C#* y el framework *.Net*.

Por otro lado, cuando se mira al gran esfuerzo y dedicación que ha requerido desarrollar una herramienta de estas características, resulta inevitable sentirse recompensado por la satisfacción de facilitar la labor de todos aquellos trabajadores en el campo de la veterinaria (doctores, investigadores, profesores, etc.).

## 10. TRABAJO FUTURO

Al comienzo del apartado anterior se habla de haber llevado el desarrollo de la aplicación a su término final, para posteriormente corregirse y definir el estado actual de la aplicación como tan solo un hito sobre el que seguir explorando y perfeccionando la herramienta en un futuro.

Ya durante el propio desarrollo del proyecto fueron surgiendo ideas que acabarían formando parte de una lista de posibles ampliaciones. Algunas de ellas se implementaron durante la realización del proyecto. Otras permanecieron anotadas en la lista como posibles elementos de trabajo futuro.

A continuación se proponen algunas ideas para este propósito.

**Funcionalidades añadidas:** La funcionalidad de la aplicación puede extenderse de diversas maneras. Por ejemplo:

- Es posible añadir una función para cortar y almacenar escenas por si mismas a partir de archivos de video más grandes, de forma que sea posible compartimentar la información.
- En la misma línea, ser capaz de relacionar distintas sesiones de video para el mismo animal, para luego posteriormente comparar la información obtenida en un mismo documento *Excel*.
- Calcular la amplitud y longitud, así como la velocidad y aceleración angular a la hora de calcular la zancada de forma automática como punto central durante la grabación de videos de larga duración.
- Añadir la posibilidad de que el propio usuario de la aplicación sea capaz de alterar los parámetros de precisión para los algoritmos de Shi-Tomasi, y de Lucas-Kanade.

**Mejoras de la interfaz:** La interfaz construida con *wxWidgets* tiene limitaciones en cuanto a la representación de la información. Por lo que sería recomendable su reconstrucción usando otro módulo, como por ejemplo *Qt*; o realizarlo de forma nativa para sistemas *Windows* con *C#*.

**Multiplataforma:** Ciertos aspectos de la aplicación tuvieron que dejarse de lado para concentrar el esfuerzo en la creación de una aplicación viable para el sistema usado actualmente. Como trabajo futuro, se podría continuar con el trabajo de esta aplicación para completar las condiciones que forman las características de una aplicación multiplataforma en los entornos de *Linux* y *OS X*.

**Sistema experto:** Con el objetivo de mejorar la asistencia que proporciona esta aplicación, se recomienda el desarrollo de un sistema experto que correlacione conocimientos en veterinaria con los datos que provee la aplicación.

## 11. BIBLIOGRAFÍA

- [FWE01] David J. Fleet and Yair Weiss (2006). "Optical Flow Estimation". In Paragios; et al. *Handbook of Mathematical Models in Computer Vision* (PDF). Springer.
  - ❖ <http://www.cs.toronto.edu/~fleet/research/Papers/flowChapter05.pdf>
- [BFB01] John L. Barron, David J. Fleet, and Steven Beauchemin (1994). "Performance of optical flow techniques" (PDF). *International Journal of Computer Vision* (Springer) **12**: 43–77.
  - ❖ <http://www.cs.toronto.edu/~fleet/research/Papers/ijcv-94.pdf>
- [AGA01] V.M. Arévalo, J. Gonzáles, G. Ambrosio (2004). "La librería de visión artificial OpenCV, aplicación a la docencia e investigación" (PDF).
  - ❖ <http://mapir.isa.uma.es/varevalo/drafts/arevalo2004lva1.pdf>
- [JKS01] Jae Kyu Shur, Computer Vision Lab (2009). "Kanade-Lucas-Tomasi (KL) Feature Tracker." (PDF).
  - ❖ <http://web.yonsei.ac.kr/jksuhr/articles/Kanade-Lucas-Tomasi%20Tracker.pdf>
- [CAS01] Luis Castellanos (2004). "Sistemas Operativos: Una guía de estudios" (PDF).
  - ❖ <https://luiscastellanos.files.wordpress.com/2014/01/sistemas-operativos-luis-castellanos.pdf>
- [01] Wikipedia sobre el Flujo Óptico. (2016)
  - ❖ [https://en.wikipedia.org/wiki/Optical\\_flow](https://en.wikipedia.org/wiki/Optical_flow)
- [02] Documentación de referencia para C++ standard. (2014)
  - ❖ <http://www.cplusplus.com/>
- [03] Documentación de referencia para wxWidgets. (2016)
  - ❖ <http://www.wxwidgets.org/docs/>
- [04] Documentación de referencia para OpenCV. (2016)
  - ❖ <http://opencv.org/>
  - ❖ <http://docs.opencv.org/3.1.0/>
  - ❖ <http://docs.opencv.org/2.4.2/>
- [05] Documentación de referencia para wxMathPlot. (2014)
  - ❖ <http://wxmathplot.sourceforge.net/>
- [06] Documentación de referencia para wxJSON. (2014)
  - ❖ <http://wxcode.sourceforge.net/docs/wxjson/>
- [07] OnTrack Equine de Lameness Solutions. (2006)
  - ❖ <http://www.ontrackequine.com>
- [08] UCOTRACK. (2015)
  - ❖ <http://www.ucotrack.es/>
- [09] Simi motions. (2015)
  - ❖ <http://www.simi.com/en/>
- [10] Ariel Performance Analysis System (APAS). (2012)

- ❖ <http://www.arielnet.com/apas>
- [11] Tracker. Video Analysis and Modeling Tool. (2015)
  - ❖ <https://www.cabrillo.edu/~dbrown/tracker/>
- [12] Metric. Video measurement technology for the PC. (2003)
  - ❖ [http://www.eurolupe.com/Lampe/software/Metric\\_L\\_e.htm](http://www.eurolupe.com/Lampe/software/Metric_L_e.htm)
  - ❖ [http://www.m-service.de/seiten/e/e\\_software/](http://www.m-service.de/seiten/e/e_software/)
- [13] SkillSpector. (2009)
  - ❖ <http://video4coach.com/>
- [14] Kinovea. (2011)
  - ❖ <http://www.kinovea.org/>
- [15] Dartfish. (2015)
  - ❖ <http://www.dartfish.com/Express>
- [16] Visual3D. (2015)
  - ❖ <http://www.c-motion.com/products/visual3d/>
- [17] Qualisys. (2015)
  - ❖ <http://www.qualisys.com/>
- [18] Mokka. (2013)
  - ❖ <https://b-tk.googlecode.com/svn/web/mokka/index.html>
- [19] Vicon. (2015)
  - ❖ <http://www.vicon.com/>
- [20] OptiTrack. (2015)
  - ❖ <https://www.naturalpoint.com/optitrack/>

## 12. ANEXO I

### 12.1. MANUAL DE INSTALACIÓN DEL PROYECTO

La instalación de este proyecto requiere descargar toda esta serie de herramientas y librerías antes de seguir los pasos que se detallan a continuación:

Herramienta	Versión	Enlace URL
Java Runtime Environment	jre_8u45_windows_x64	<a href="https://www.java.com">https://www.java.com</a>
Eclipse IDE for C/C++ Developers	eclipse_cpp_luna_SR2_win32_x86_64	<a href="https://eclipse.org">https://eclipse.org</a>
TortoiseSVN	Versión 1.8 (r24944)	<a href="https://tortoisesvn.net">https://tortoisesvn.net</a>
Subversion		Desde Eclipse
MinGW+MSYS	GCC 3.4.5	<a href="http://www.mingw.org/">http://www.mingw.org/</a>
wxWidgets	wxWidgets-3.0.2	<a href="https://wxwidgets.org">https://wxwidgets.org</a>
OpenCV	opencv-3.0.0-rc1	<a href="https://opencv.org">https://opencv.org</a>

#### 12.1.1. INSTALACIÓN DE JAVA Y ECLIPSE

- Instalar Java Runtime Environment. Versión jre\_8u45\_windows\_x64.
- Instalar Eclipse IDE for C/C++ Developers. Versión eclipse\_cpp\_luna\_SR2\_win32\_x86\_64.

#### 12.1.2. INSTALACIÓN DE TORTOISESVN Y SUBVERSION

Los pasos que se describen a continuación son solamente necesarios para retomar los datos del proyecto alojados en Dropbox.

- Instalar TortoiseSVN.
- Instalar Subversion desde el controlador de addons en Eclipse.
- Instalar Dropbox, descargar el directorio SVN en una carpeta local.
- Abrir Eclipse.
  - Entrar en el Explorador de repositorios SVN.
    - (SVN Repository Explorer)
  - Seleccionar como nueva localización para el repositorio el nuevo directorio SVN que nos hemos descargado de Dropbox.
    - (New Repository Location->SVN directory)

- Cambiar a la vista C/C++.
- Crear nuevo proyecto desde SVN.
  - (New->Project...->SVN->Project from SVN)

### 12.1.3. INSTALACIÓN DE WXWIDGETS

- Instalar MinGW y MSYS.
- Descomprimir wxWidgets en C:/MinGW.
  - Por ejemplo C:/MinGW/wx3.
- Crear el directorio “/mingw” en “/msys/1.0”.
- Crear un archivo de texto sin formato en “/msys/1.0/etc” llamado fstab con el texto:

#Win32_Path	Mount_Point
C:/mingw	/mingw

- A continuación se abre la consola de comandos de MSYS y se ejecutan los siguientes comandos para habilitar las opciones debug:
  - `cd c/wxwidgets-3.0.2`
  - `mkdir build_debug`
  - `./configure --prefix=/mingw --disable-shared --enable-debug`
    - La opción `--disable-shared` habilitará el enlace estático.
    - La opción `--enable-debug` habilitará las opciones de debug.
  - `make && make install`
- Para la versión final, se desactivarán las opciones de debug.
  - `mkdir build-release`
  - `cd build-release`
  - `./configure --prefix=/mingw --disable-shared`
  - `make && make install`
- Añadir `c:/mingw/bin` al path de Windows.

### 12.1.4. INSTALACIÓN DE OPENCV

- Descomprimir los archivos fuente de OpenCV.
- Instalar CMake.
- Usar CMake para generar las cabeceras para la compilación de los archivos fuente de la librería OpenCV.
- Abrir `cmd.exe`, y situarse en el directorio con los archivos generados por CMake.
- Ejecutar `mingw32-make`.
- Añadir el directorio con las librerías generadas al Path de Windows.
- Añadir las librerías necesarias generadas al proyecto, listas para su uso.

## 13. ANEXO II

### 13.1. MANUAL DE USUARIO

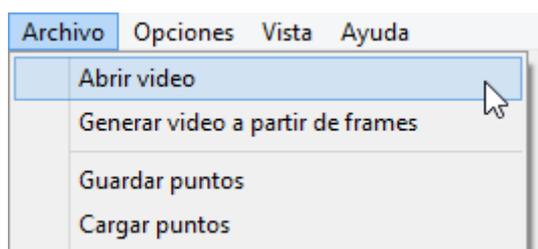
#### 13.1.1. GUÍA DE INSTALACIÓN DE LA APLICACIÓN

La aplicación debería estar contenida en un solo archivo comprimido, de forma que al descomprimir el archivo todas las carpetas y archivos estén localizados en su posición correspondiente y la aplicación lista para usar.

Res/	Res/	player_angle.bmp
HAV.exe	Res/	detect.bmp
libgcc_s_dw2-1.dll	Res/	detect1.bmp
libopencv_core300.dll	Res/	detect2.bmp
libopencv_highgui300.dll	Res/	detect3.bmp
libopencv_imgcodecs300.dll	Res/	end.bmp
libopencv_imgproc300.dll	Res/	erase.bmp
libopencv_video300.dll	Res/	erase1.bmp
libopencv_videoio300.dll	Res/	fps.bmp
libstdc++-6.dll	Res/	gear.bmp
opencv_ffmpeg300.dll	Res/	gear2.bmp
	Res/	line.bmp
	Res/	line2.bmp
	Res/	pause.bmp
	Res/	play.bmp
	Res/	prev.bmp
	Res/	regla.bmp
	Res/	regla2.bmp
	Res/	stop.bmp
	Res/	umbral.bmp
	Res/	view.bmp
	Res/	t_end.bmp
	Res/	t_ini.bmp
	Res/	timing.bmp

#### 13.1.2. REPRODUCIR ARCHIVO DE VIDEO

- Primer paso, ejecutar el archivo HAV.exe para abrir la aplicación.
- Segundo paso, Seleccionar la opción *Abrir video* en el menu *Archivo*.

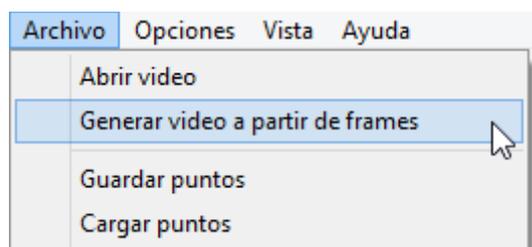


- Tercer paso, localizar el directorio en el que esté situado el archivo de video a reproducir, seleccionarlo y Aceptar.

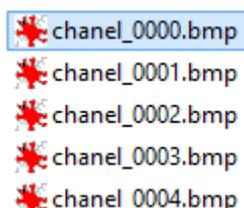
### 13.1.1.3. CONVERTIR SECUENCIA DE FRAMES EN VIDEO

Para convertir una secuencia de frames en video primero se ha de tener un conjunto de frames en formato mapa de bits (bmp) ordenados numéricamente de forma consecutiva.

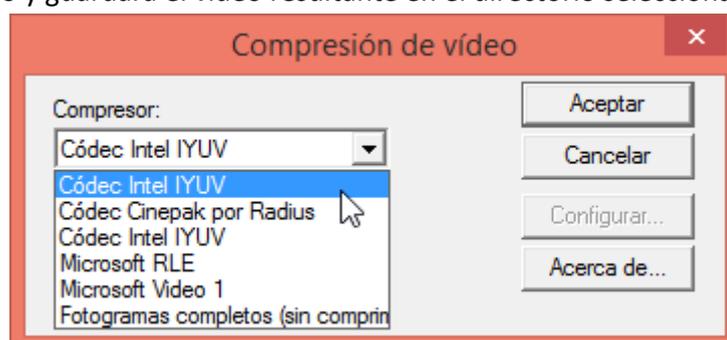
- Primer paso, ejecutar el archivo HAV.exe para abrir la aplicación.
- Segundo paso, Seleccionar la opción *Generar video a partir de frames* en el menu *Archivo*.



- Tercer paso, localizar el directorio en el que esté situado el conjunto de frames, seleccionar el primero y Aceptar.



- Cuarto paso, localizar el directorio en el que se desea guardar el archivo de video resultante, nombrarlo y Aceptar.
- Quinto paso, selecciona un compresor para crear el archivo de video y pulsa Aceptar. Tras lo cual la aplicación unificará el conjunto de frames usando el códec seleccionado y guardará el video resultante en el directorio seleccionado.



#### 13.1.4. CALCULAR ÁNGULOS DURANTE LA REPRODUCCIÓN

Como preparación para calcular ángulos durante la reproducción, antes debemos tener un archivo de video ya abierto.

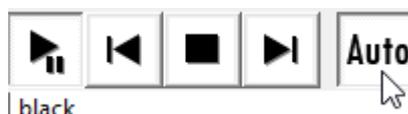
- Primer paso, pulsar el botón para calcular ángulos en la barra de herramientas:



- Segundo paso, a continuación la reproducción se pausará, lo cual nos permitirá marcar los tres puntos que conformarán el ángulo que se desea calcular. Tanto el primer como el tercer punto que se marque deberán ser las aristas, mientras que el segundo punto marcará el punto desde el que se calculará el ángulo. Una vez marcados los tres puntos se añadirán los cálculos a la tabla adecuada y se mostrará en el ángulo en la reproducción.



- Tercer paso, si no se busca el ángulo conjugado salta al siguiente paso. Si el ángulo que se busca es el conjugado, basta con pulsar el botón de intercambio de ángulos (  ) y pulsar de nuevo sobre el segundo punto.
- Cuarto paso, si se desea rastrear dichos puntos durante la reproducción basta con pulsar el botón AUTO seguido del botón PLAY/PAUSA para continuar con la reproducción.



- Quinto paso, cuando se desee detener el rastreo, pulsar el botón PLAY/PAUSA para detener la reproducción y de nuevo el botón AUTO.

#### 13.1.5. CALCULAR DISTANCIAS DURANTE LA REPRODUCCIÓN

Como preparación para calcular ángulos durante la reproducción, antes debemos tener un archivo de video ya abierto y la medida de distancias calibradas.

### Calibrado:

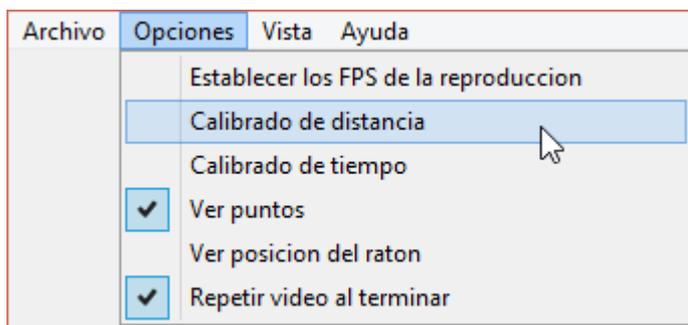
Para poder calibrar la distancia es necesario tener una medida de referencia en algún fotograma del video. De forma que se pueda calcular la distancia real en base a la distancia de la imagen.

Para ello empezamos reproduciendo el archivo de video con el que se desea trabajar.

- Primer paso, una vez tengamos el video en reproducción, nos situamos en el fotograma que tiene la medida de referencia (en el ejemplo se encuentra marcado con cinta aislante de color blanco).



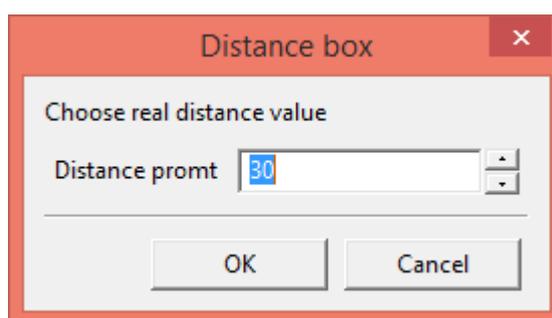
- Segundo paso, ir al menú Opciones y seleccionar el Calibrado de distancia.



- Tercer paso, mantener pulsado el botón izquierdo del ratón mientras se arrastra desde un lado a otro de la medida de referencia.



- Cuarto paso, una vez soltamos el botón, una nueva ventana de dialogo nos pedirá que indiquemos cual es la medida real que separa la distancia que se ha indicado en el video. Por ejemplo, si esa medida son 30 cm, escribiremos 30 en el cuadro y pulsaremos aceptar.



Una vez pulsamos OK ya tendremos calibradas las distancias y podemos hacer uso de las demás herramientas de medición.

#### Medida de distancia con la regla

- Primer paso, pulsar para botón de “Regla” 
- Segundo paso, pulsar y arrastra la distancia que se desea calcular. Alternativamente se puede pulsar en un punto y a continuación en otro para conocer la distancia que las separa.



- Tercer paso, a continuación se abrirá una ventana indicando la distancia que separa el punto inicial del punto final.

#### Medida de distancia por puntos

- Primer paso, pulsar para botón de “Calcula Zancada” 
- Segundo paso, marcar el punto inicial para calcular la medida. Este punto aparecerá en la tabla de medidas de Distancia con la distancia 0.
- Tercer paso, marcar el punto final de la medición. El punto y el resultado de la distancia marcada aparecerán a continuación en la tabla de Distancia. Este punto final puede tanto situarse en el mismo fotograma que el primero, como en un fotograma posterior.

	Frame	Distancia	wxPoint
1	8	0.00	(504,260)
2	20	10.99	(408,261)
3			

- De seguir marcando puntos de esta manera, la tabla seguirá almacenándolos alternativamente, de forma que la medida de los puntos pares corresponde siempre referente al anterior, mientras que los puntos impares serán siempre 0.

## 14. ANEXO III

## 14.1. LISTADO DE CASOS DE USO COMPLETOS

Actor Principal	Casos de Uso
Usuario	19. <a href="#">Abrir archivo de video</a>
	20. <a href="#">Abrir secuencia de imágenes</a>
	21. <a href="#">Guardar datos</a>
	22. <a href="#">Cargar datos</a>
	23. <a href="#">Agregar gráfica</a>
	24. <a href="#">Guardar gráfica como imagen</a>
	25. <a href="#">Exportar a Excel</a>
	26. <a href="#">Exportar a CSV</a>
	27. <a href="#">Salir del programa</a>
	28. <a href="#">Modificación de los FPS</a>
	29. <a href="#">Calibrado de la distancia</a>
	30. <a href="#">Calibrado de tiempo</a>
	31. <a href="#">Ver puntos</a>
	32. <a href="#">Ocultar puntos</a>
	33. <a href="#">Ver posición del ratón</a>
	34. <a href="#">Ocultar posición del ratón</a>
	35. <a href="#">Repetir video al terminar</a>
	36. <a href="#">No Repetir video al terminar</a>
	37. <a href="#">Ver Botones de Puntos</a>

	38. <a href="#">Ocultar Botones de Puntos</a>
	39. <a href="#">Ver Botones de Distancia</a>
	40. <a href="#">Ocultar Botones de Distancia</a>
	41. <a href="#">Ver Botones de Ángulos</a>
	42. <a href="#">Ocultar Botones de Ángulos</a>
	43. <a href="#">Ver Botones de Tiempo</a>
	44. <a href="#">Ocultar Botones de Tiempo</a>
	45. <a href="#">Ver Botones de Reproductor</a>
	46. <a href="#">Ocultar Botones de Reproductor</a>
	47. <a href="#">Ver Leyenda de la gráfica</a>
	48. <a href="#">Ocultar Leyenda de la gráfica</a>
	49. <a href="#">Ver posición en la gráfica</a>
	50. <a href="#">Ocultar posición en la gráfica</a>
	51. <a href="#">Ver información sobre la aplicación</a>
	52. <a href="#">Reproducir archivo de video</a>
	53. <a href="#">Pausar archivo de video</a>
	54. <a href="#">Modo búsqueda</a>
	55. <a href="#">Detener la reproducción</a>
	56. <a href="#">Avanzar frame</a>
	57. <a href="#">Retroceder frame</a>
	58. <a href="#">Cálculo de Flujo óptico para los puntos marcados</a>
	59. <a href="#">Marcar punto en el frame actual</a>
	60. <a href="#">Borrar punto en el frame actual</a>
	61. <a href="#">Marcar Frame inicial</a>

	62. <a href="#">Marcar Frame final</a>
	63. <a href="#">Calcular el tiempo transcurrido entre los marcadores de Frame inicial y final</a>
	64. <a href="#">Marcar punto en el frame actual para distancias</a>
	65. <a href="#">Establecer horizontal para el cálculo de zancadas automático</a>
	66. <a href="#">Cálculo de la distancia entre dos puntos de la imagen.</a>
	67. <a href="#">Cálculo del ángulo entre tres puntos de la imagen</a>
	68. <a href="#">Cálculo del ángulo conjugado</a>
	69. <a href="#">Ocultar ángulos</a>
	70. <a href="#">Cambiar el color de la gráfica</a>
	71. <a href="#">Seleccionar pestaña de ángulos</a>
	72. <a href="#">Seleccionar pestaña de distancias</a>
	73. <a href="#">Desplazar gráfica.</a>
	74. <a href="#">Acercar gráfica</a>
	75. <a href="#">Alejar gráfica</a>
	76. <a href="#">Crear nueva anotación</a>
	77. <a href="#">Sobrescribir anotación</a>
	78. <a href="#">Seleccionar anotación</a>

## CASOS DE USO

<b>Nombre</b>	Abrir archivo de video	<b>ID</b>	1.1
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p style="text-align: center;">Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p style="text-align: center;">Usuario</p>
<p><b>Descripción:</b></p> <p style="text-align: center;">El usuario pulsa sobre Archivo y busca la opción “Abrir video” de la aplicación. Navega por los directorios hasta el archivo que desea carga y lo selecciona pulsando “Aceptar”. La aplicación carga el archivo y comienza su reproducción.</p>
<p><b>Trigger:</b></p> <p style="text-align: center;">Pulsar sobre el botón Abrir en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se ha abierto el video desde la ruta especificada.</li> <li>2. Comienza a reproducirse el video.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Se ejecuta la aplicación.
2. Se pulsa sobre el menú Archivo.
3. Se pulsa la opción de Abrir video.
4. Se selecciona la ruta donde se encuentra el archivo que se desea abrir.
5. Se abre el archivo y comienza a reproducirse en la ventana.

**Flujo Alternativo:**

1. Aplicación en ejecución, con video abierto previamente.
2. Se pulsa sobre el menú Archivo.
3. Se pulsa el botón de Abrir en la aplicación.
4. Se cierra el archivo de video actualmente en ejecución.
5. Se devuelve al punto 4 del Flujo Normal de ejecución.

**Excepción:**

En caso de que el archivo sea de tipo erróneo, esté corrupto, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Abrir secuencia de imágenes	<b>ID</b>	1.2
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Generar video a partir de frames” de la aplicación. Navega por los directorios hasta una carpeta que contenga toda una secuencia de imágenes, ordenadas numéricamente, que desea cargar, y lo selecciona pulsando “Aceptar”. A continuación se elige el nombre con el que se desea guardar la nueva secuencia de video. Finalmente se selecciona uno de los códec disponibles.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón “Generar video a partir de frames” en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. En la ruta especificada se ha generado un archivo de video a partir de la secuencia de imágenes.</li> </ol>

**Flujo Normal:**

1. Se ejecuta la aplicación.
2. Se pulsa sobre el menú Archivo.
3. Se pulsa la opción de Generar video a partir de frames.
4. Se selecciona la ruta donde se encuentra la secuencia de imágenes.
5. Se selecciona la primera imagen de la secuencia.
6. Se selecciona la ruta donde se generará el archivo de video.
7. Se selecciona el códec a utilizar a partir de una lista de codecs disponibles.
8. Se devuelve al flujo normal de ejecución.

**Flujo Alternativo:****Excepción:**

En caso de que el archivo sea de tipo erróneo, esté corrupto, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Guardar Datos	<b>ID</b>	1.3
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Guardar Datos” de la aplicación. Navega por los directorios hasta la carpeta donde se desea guardar el archivo, se selecciona el nombre y se pulsa “Aceptar”. La aplicación guarda los datos en el archivo.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Guardar Datos en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> <li>2. Se ha abierto un video.</li> <li>3. Hay datos para guardar, ya sean anotaciones o marcadores en el video.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. En la ruta especificada se ha creado un archivo con los datos almacenados.</li> <li>2. La aplicación sigue en ejecución.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Se pulsa sobre el menú Archivo.
2. Se pulsa la opción de Guardar Datos.
3. Se selecciona la ruta donde se desea guardar el archivo.
4. Se escribe un nombre para el archivo.
5. Se pulsa Aceptar.
6. Se devuelve al flujo normal de ejecución.

**Flujo Alternativo:****Excepción:**

En caso de que no tenga permiso para escribir en la ruta especificada, o no hayan puntos marcados:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Cargar Datos	<b>ID</b>	1.4
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Cargar Datos” de la aplicación. Navega por los directorios hasta el archivo que desea carga y lo selecciona pulsando “Aceptar”. La aplicación carga el archivo.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Cargar Datos en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> <li>2. Un video se encuentra abierto.</li> <li>3. Se dispone de un archivo de Datos que se pueda abrir.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se han cargado los datos desde la ruta especificada.</li> <li>2. Los datos son visibles tanto en la gráfica, como en el reproductor.</li> </ol>

**Flujo Normal:**

1. Se ejecuta la aplicación.
2. Se pulsa sobre el menú Archivo.
3. Se pulsa la opción de Cargar Datos.
4. Se selecciona la ruta donde se encuentra el archivo que se desea abrir.
5. Se abre el archivo y continúa el flujo de ejecución.

**Flujo Alternativo:****Excepción:**

En caso de que el archivo sea de tipo erróneo, esté corrupto, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Agregar gráfica	<b>ID</b>	1.5
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Agregar gráfica” de la aplicación. Navega por los directorios hasta el archivo que desea carga y lo selecciona pulsando “Aceptar”. La aplicación carga el archivo y lo muestra en las gráficas.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Agregar gráfica en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> <li>2. Un video se encuentra abierto.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se ha abierto el video desde la ruta especificada.</li> <li>2. Comienza a reproducirse el video.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Se ejecuta la aplicación.</li> <li>2. Se pulsa sobre el menú Archivo.</li> </ol>

3. Se pulsa la opción de Agregar Gráfica.
4. Se selecciona la ruta donde se encuentra el archivo que se desea abrir.
5. Se abre el archivo, que se visualizará en el panel de gráficas.
6. Se devuelve al flujo normal de ejecución.

**Flujo Alternativo:****Excepción:**

En caso de que el archivo sea de tipo erróneo, esté corrupto, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Guardar gráfica como imagen	<b>ID</b>	1.6
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Guardar gráfica como imagen” de la aplicación. Navega por los directorios hasta la carpeta en la que se desea guardar la imagen, se selecciona un nombre y completamos la acción pulsando “Aceptar”.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Guardar gráfica como imagen en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se ha abierto el video desde la ruta especificada.</li> <li>2. Comienza a reproducirse el video.</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Se ejecuta la aplicación.</li><li>2. Se pulsa sobre el menú Archivo.</li><li>3. Se pulsa la opción de Guardar gráfica como imagen.</li><li>4. Se selecciona la ruta y el nombre para guardar el archivo.</li><li>5. Se guarda el archivo.</li><li>6. Se devuelve al flujo normal de ejecución.</li></ol>
<p><b>Flujo Alternativo:</b></p>
<p><b>Excepción:</b></p> <p>En caso de que el archivo sea de tipo erróneo, esté corrupto, o se cancele la operación:</p> <ol style="list-style-type: none"><li>1. Un mensaje avisará al usuario de la incidencia.</li><li>2. Se cancela la operación.</li><li>3. Se devuelve el flujo de operación a la aplicación abierta.</li></ol>
<p><b>Includes:</b></p>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Exportar a Excel	<b>ID</b>	1.7
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Exportar a Excel” de la aplicación. La aplicación ejecuta Microsoft Excel y exporta los datos de los marcadores a su formato.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Exportar a Excel en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> <li>2. Existen datos que puedan exportarse.</li> <li>3. Microsoft Excel se encuentra instalado en el sistema.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se ha abierto la aplicación de Microsoft Excel.</li> <li>2. Se han exportado los datos</li> <li>3. La aplicación prosigue su flujo de ejecución.</li> </ol>

**Flujo Normal:**

1. Se ejecuta la aplicación.
2. Se pulsa sobre el menú Archivo.
3. Se pulsa la opción de Exportar a Excel.
4. La aplicación abre Excel y exporta los datos.
5. Continúa el flujo de ejecución de la aplicación.

**Flujo Alternativo:****Excepción:**

En caso de que Excel no se encuentre instalado, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Exportar a CSV	<b>ID</b>	1.8
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	06-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario pulsa sobre Archivo y busca la opción “Exportar a CSV” de la aplicación. Navega por los directorios hasta la carpeta donde se desea guardar se pulsa “Aceptar”. La aplicación guardará los datos en un archivo de texto siguiendo el formato CSV.</p>
<p><b>Trigger:</b></p> <p>Pulsar sobre el botón Exportar a CSV en la aplicación.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. El usuario tiene la aplicación en ejecución.</li> <li>2. Hay datos que se puedan exportar.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Se ha creado un nuevo archivo de texto con los datos exportados.</li> </ol>

2. Se devuelve la aplicación a su flujo de ejecución.

**Flujo Normal:**

1. Se ejecuta la aplicación.
2. Se pulsa sobre el menú Archivo
3. Se pulsa la opción de Exportar a CSV.
4. Se selecciona la ruta y el nombre donde guardar el archivo de texto.
5. Se pulsa Aceptar.
6. Se devuelve la aplicación a su flujo de ejecución

**Flujo Alternativo:****Excepción:**

En caso de que no hayan datos que exportar, o se cancele la operación:

1. Un mensaje avisará al usuario de la incidencia.
2. Se cancela la operación.
3. Se devuelve el flujo de operación a la aplicación abierta.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Salir del programa	<b>ID</b>	1.9
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Quando se desee cerrar la aplicación se busca el botón de salir o el botón en el exterior de la ventana para cerrar la aplicación.</p>
<p><b>Trigger:</b></p> <p>Botón de "Salir" o "Cerrar ventana".</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación cerrada.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Se pulsa sobre el menú Archivo</li> <li>3. Pulsar el botón "Salir del programa".</li> </ol>

4. Cierre de la aplicación.

**Flujo Alternativo:**

1. Aplicación en ejecución.
2. Se pulsa sobre el botón para cerrar la ventana.
3. Cierre de la aplicación.

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Modificación de los FPS	<b>ID</b>	1.10
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca alterar la velocidad de reproducción de video, por lo que se dirige al menú Opciones, y pulsa el botón Modificación de los FPS. Una nueva ventana se abre indicando la velocidad actual de reproducción. Cambiando esta cifra y pulsando Aceptar para que los efectos tengan lugar.</p>
<p><b>Trigger:</b></p> <p>Botón de “Modificación de los FPS”</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video en reproducción.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Valor de FPS del video en reproducción modificado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Opciones.
3. Pulsar el botón "Modificación de los FPS".
4. Cambiar el número.
5. Pulsa Aceptar.
6. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Calibrado de la distancia	<b>ID</b>	1.11
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca establecer una medida de referencia para los cálculos de distancia en reproducción de video, por lo que se dirige al menú Opciones, y pulsa el botón Calibrado de la distancia. Una nueva ventana se abre indicando un valor de referencia predefinido. Cambiando esta cifra y pulsando Aceptar para que los efectos tengan lugar.</p>
<p><b>Trigger:</b></p> <p>Botón de “Calibrado de la distancia”</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video en reproducción.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Valor de referencia cambiado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Opciones
3. Pulsar el botón "Calibrado de distancia".
4. Cambiar la medida de referencia.
5. Pulsar Aceptar.
6. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Calibrado de tiempo	<b>ID</b>	1.12
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca establecer una medida de referencia para los cálculos de tiempo en reproducción de video, por lo que se dirige al menú Opciones, y pulsa el botón Calibrado de tiempo. Una nueva ventana se abre indicando un valor de referencia predefinido. Cambiando esta cifra y pulsando Aceptar para que los efectos tengan lugar.</p>
<p><b>Trigger:</b></p> <p>Botón de “Calibrado de tiempo”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video en reproducción.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Medida de tiempo modificada.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Opciones.
3. Pulsar el botón “Calibrado de tiempo”.
4. Cambiar la medida de referencia.
5. Pulsar Aceptar.
6. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver puntos	<b>ID</b>	1.13
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca mostrar durante la reproducción de video los puntos marcados en la pantalla de video, por lo que se dirige al menú Opciones, y pulsa el botón Ver Puntos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Puntos”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver Puntos” desmarcado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver Puntos” marcado como activado.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Se pulsa sobre el menú Opciones.
3. Pulsar el botón "Ver Puntos".
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar puntos	<b>ID</b>	1.14
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar durante la reproducción de video los puntos marcados en la pantalla de video, por lo que se dirige al menú Opciones, y pulsa el botón Ver Puntos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Puntos”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver Puntos” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver Puntos” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Se pulsa sobre el menú Opciones
3. Pulsar el botón “Ver Puntos”.
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver posición del ratón	<b>ID</b>	1.15
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca mostrar durante la reproducción de video la posición del ratón en la pantalla de video, por lo que se dirige al menú Opciones, y pulsa el botón Ver posición del ratón.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver posición del ratón”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver posición del ratón” desmarcado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver posición del ratón” marcado como activado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Opciones
3. Pulsar el botón “Ver posición del ratón”.
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar posición del ratón	<b>ID</b>	1.16
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar durante la reproducción de video la posición del ratón en la pantalla de video, por lo que se dirige al menú Opciones, y pulsa el botón Ver posición del ratón.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver posición del ratón”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver posición del ratón” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver posición del ratón” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Opciones
3. Pulsar el botón "Ver posición del ratón".
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Repetir video al terminar	<b>ID</b>	1.17
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca que la reproducción del video se repita una vez llega al final del video. Para ello va al menú Opciones y marca "Repetir video al terminar".</p>
<p><b>Trigger:</b></p> <p>Botón de "Repetir video al terminar".</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>3. Aplicación en ejecución.</li> <li>4. Botón "Repetir video al terminar" desmarcado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>2. Botón "Repetir video al terminar" marcado como activado.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>5. Aplicación en ejecución.</li> </ol>

6. Se pulsa sobre el menú Opciones
7. Pulsar el botón “Repetir video al terminar”.
8. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	No Repetir video al terminar	<b>ID</b>	1.18
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca que la reproducción del video se repita una vez llega al final del video. Para ello va al menú Opciones y desmarca "Repetir video al terminar".</p>
<p><b>Trigger:</b></p> <p>Botón de "Repetir video al terminar".</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón "Repetir video al terminar" marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón "Repetir video al terminar" desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Se pulsa sobre el menú Opciones
3. Pulsar el botón "Repetir video al terminar".
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Botones de Puntos	<b>ID</b>	1.19
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción para marcar y desmarcar puntos sobre la reproducción de video. Para ello se dirige al menú Vista y marca Ver Botones de Puntos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Puntos”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver Botones de Puntos” desmarcado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver Botones de Puntos” marcado como activado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Vista
3. Pulsar el botón "Ver Botones de Puntos".
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Botones de Puntos	<b>ID</b>	1.20
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción para marcar y desmarcar puntos sobre la reproducción de video. Para ello se dirige al menú Vista y desmarca Ver Botones de Puntos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Puntos”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Botón “Ver Botones de Puntos” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Botón “Ver Botones de Puntos” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Se pulsa sobre el menú Vista
3. Pulsar el botón “Ver Botones de Puntos”.
4. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Botones de Distancia	<b>ID</b>	1.21
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción para calcular la distancia de Zancada sobre la reproducción de video. Para ello se dirige al menú Vista y marca Ver Botones de Distancia.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Distancia”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>3. Aplicación en ejecución.</li> <li>4. Botón “Ver Botones de Distancia” desmarcado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>2. Botón “Ver Botones de Distancia” marcado como activado.</li> </ol>
<p><b>Flujo Normal:</b></p>

5. Aplicación en ejecución.
6. Se pulsa sobre el menú Vista
7. Pulsar el botón “Ver Botones de Distancia”.
8. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Botones de Distancia	<b>ID</b>	1.22
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar en la barra de herramienta la opción para calcular la distancia de Zancada sobre la reproducción de video. Para ello se dirige al menú Vista y desmarca Ver Botones de Distancia.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Distancia”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>3. Aplicación en ejecución.</li> <li>4. Botón “Ver Botones de Distancia” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>2. Botón “Ver Botones de Distancia” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

5. Aplicación en ejecución.
6. Se pulsa sobre el menú Vista
7. Pulsar el botón “Ver Botones de Distancia”.
8. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Botones de Ángulos	<b>ID</b>	1.23
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción para calcular diversos ángulos sobre la reproducción de video. Para ello se dirige al menú Vista y marca Ver Botones de Ángulos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Ángulos”.</p>
<p><b>Precondición:</b></p> <p>5. Aplicación en ejecución. 6. Botón “Ver Botones de Ángulos” desmarcado.</p>
<p><b>Postcondición:</b></p> <p>3. Botón “Ver Botones de Ángulos” marcado como activado.</p>
<p><b>Flujo Normal:</b></p>

9. Aplicación en ejecución.
10. Se pulsa sobre el menú Vista
11. Pulsar el botón “Ver Botones de Ángulos”.
12. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Botones de Ángulos	<b>ID</b>	1.24
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar en la barra de herramienta la opción para calcular diversos ángulos sobre la reproducción de video. Para ello se dirige al menú Vista y desmarca Ver Botones de Ángulos.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Ángulos”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>5. Aplicación en ejecución.</li> <li>6. Botón “Ver Botones de Ángulos” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>3. Botón “Ver Botones de Ángulos” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

9. Aplicación en ejecución.
10. Se pulsa sobre el menú Vista
11. Pulsar el botón “Ver Botones de Ángulos”.
12. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Botones de Tiempo	<b>ID</b>	1.25
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción para calcular el tiempo transcurrido entre dos frames sobre la reproducción de video. Para ello se dirige al menú Vista y marca Ver Botones de Tiempo.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Tiempo”.</p>
<p><b>Precondición:</b></p> <p>7. Aplicación en ejecución. 8. Botón “Ver Botones de Tiempo” desmarcado.</p>
<p><b>Postcondición:</b></p> <p>4. Botón “Ver Botones de Tiempo” marcado como activado.</p>
<p><b>Flujo Normal:</b></p>

13. Aplicación en ejecución.
14. Se pulsa sobre el menú Vista
15. Pulsar el botón “Ver Botones de Tiempo”.
16. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Botones de Tiempo	<b>ID</b>	1.26
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar en la barra de herramienta la opción para calcular el tiempo transcurrido entre dos frames sobre la reproducción de video. Para ello se dirige al menú Vista y desmarca Ver Botones de Tiempo.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Tiempo”.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>7. Aplicación en ejecución.</li> <li>8. Botón “Ver Botones de Tiempo” marcado como activado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>4. Botón “Ver Botones de Tiempo” desmarcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

13. Aplicación en ejecución.
14. Se pulsa sobre el menú Vista
15. Pulsar el botón “Ver Botones de Tiempo”.
16. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Botones de Reproductor	<b>ID</b>	1.27
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible en la barra de herramienta la opción de controlar la reproducción de video. Para ello se dirige al menú Vista y marca Ver Botones de Reproductor.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Reproductor”.</p>
<p><b>Precondición:</b></p> <p>9. Aplicación en ejecución. 10. Botón “Ver Botones de Reproductor” desmarcado.</p>
<p><b>Postcondición:</b></p> <p>5. Botón “Ver Botones de Reproductor” marcado como activado.</p>
<p><b>Flujo Normal:</b></p>

17. Aplicación en ejecución.
18. Se pulsa sobre el menú Vista
19. Pulsar el botón “Ver Botones de Reproductor”.
20. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Botones de Reproductor	<b>ID</b>	1.28
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar en la barra de herramienta la opción de controlar la reproducción de video. Para ello se dirige al menú Vista y desmarca Ver Botones de Reproductor.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Botones de Reproductor”.</p>
<p><b>Precondición:</b></p> <p>9. Aplicación en ejecución. 10. Botón “Ver Botones de Reproductor” marcado como activado.</p>
<p><b>Postcondición:</b></p> <p>5. Botón “Ver Botones de Reproductor” desmarcado.</p>
<p><b>Flujo Normal:</b></p>

17. Aplicación en ejecución.
18. Se pulsa sobre el menú Vista
19. Pulsar el botón “Ver Botones de Reproductor”.
20. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver Leyenda de la gráfica	<b>ID</b>	1.29
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible la leyenda de la gráfica. Para ello se dirige al menú Vista y marca Ver Leyenda de la gráfica.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Leyenda de la gráfica”.</p>
<p><b>Precondición:</b></p> <p>11. Aplicación en ejecución. 12. Botón “Ver Leyenda de la gráfica” desmarcado.</p>
<p><b>Postcondición:</b></p> <p>6. Botón “Ver Leyenda de la gráfica” marcado como activado.</p>
<p><b>Flujo Normal:</b></p> <p>21. Aplicación en ejecución.</p>

22. Se pulsa sobre el menú Vista
23. Pulsar el botón “Ver Leyenda de la gráfica”.
24. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar Leyenda de la gráfica	<b>ID</b>	1.30
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar la leyenda de la gráfica. Para ello se dirige al menú Vista y desmarca Ver Leyenda de la gráfica.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver Leyenda de la gráfica”.</p>
<p><b>Precondición:</b></p> <p>11. Aplicación en ejecución. 12. Botón “Ver Leyenda de la gráfica” marcado como activado.</p>
<p><b>Postcondición:</b></p> <p>6. Botón “Ver Leyenda de la gráfica” desmarcado.</p>
<p><b>Flujo Normal:</b></p> <p>21. Aplicación en ejecución.</p>

22. Se pulsa sobre el menú Vista
23. Pulsar el botón “Ver Leyenda de la gráfica”.
24. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver posición en la gráfica	<b>ID</b>	1.31
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca tener disponible la posición del ratón en la gráfica. Para ello se dirige al menú Vista y marca Ver posición en la gráfica.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver posición en la gráfica”.</p>
<p><b>Precondición:</b></p> <p>13. Aplicación en ejecución. 14. Botón “Ver posición en la gráfica” desmarcado.</p>
<p><b>Postcondición:</b></p> <p>7. Botón “Ver posición en la gráfica” marcado como activado.</p>
<p><b>Flujo Normal:</b></p> <p>25. Aplicación en ejecución.</p>

26. Se pulsa sobre el menú Vista
27. Pulsar el botón “Ver posición en la gráfica”.
28. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ocultar posición en la gráfica	<b>ID</b>	1.32
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca ocultar la posición del ratón en la gráfica. Para ello se dirige al menú Vista y desmarca Ver posición en la gráfica.</p>
<p><b>Trigger:</b></p> <p>Botón de “Ver posición en la gráfica”.</p>
<p><b>Precondición:</b></p> <p>13. Aplicación en ejecución. 14. Botón “Ver posición en la gráfica” marcado como activado.</p>
<p><b>Postcondición:</b></p> <p>7. Botón “Ver posición en la gráfica” desmarcado.</p>
<p><b>Flujo Normal:</b></p> <p>25. Aplicación en ejecución.</p>

26. Se pulsa sobre el menú Vista
27. Pulsar el botón “Ver posición en la gráfica”.
28. Vuelta al Flujo normal de ejecución.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Ver información sobre la aplicación	<b>ID</b>	1.33
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	16-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>El usuario busca saber más sobre la aplicación y su desarrollador. Para ello se dirige al menú Ayuda y Pulsa la opción Sobre...</p>
<p><b>Trigger:</b></p> <p>Botón de "Sobre..."</p>
<p><b>Precondición:</b></p> <p>15. Aplicación en ejecución.</p>
<p><b>Postcondición:</b></p> <p>8. Ventana de información abierta.</p>
<p><b>Flujo Normal:</b></p> <p>29. Aplicación en ejecución. 30. Se pulsa sobre el menú Ayuda</p>

31. Pulsar el botón "Sobre...".

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Reproducir archivo de video	<b>ID</b>	1.34
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Para la correcta reproducción del video es necesario pulsar el botón de reproducción cuando un video se encuentre abierto y en pausa.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón de reproducción.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto y en pausa.</li> </ol>
<p><b>Postcondición:</b></p> <ul style="list-style-type: none"> <li>• Video en reproducción</li> </ul>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Video abierto y en pausa.
3. Pulsar el botón de reproducción.
4. Video en reproducción.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Pausar archivo de video	<b>ID</b>	1.35
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Quando se desea pausar la reproducción del video es necesario pulsar el botón de pausa. Quando se vuelve a pulsar el botón de pausa se entiende que se desea volver a la reproducción del video.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón de pausa.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video en reproducción.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Video en pausa</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Video abierto y en reproducción.
3. Pulsar el botón de pausa.
4. Video en pausa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Modo búsqueda	<b>ID</b>	1.36
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>A veces se desea buscar un momento concreto en la reproducción del video, para ello se dispone de una barra de búsqueda por la que poder trasladarse en la reproducción del video. Basta con mover el icono o marcador de tiempo de la barra de búsqueda para hacer uso de esta.</p>
<p><b>Trigger:</b></p> <p>Mover el icono en la barra de búsqueda</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación abierta</li> <li>2. Video abierto</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. La marca de tiempo del video se traslada a la nueva ubicación</li> </ol>

**Flujo Normal:**

1. Aplicación abierta.
2. Video en pausa.
3. Se mueve el marcador de la barra de búsqueda.
4. La reproducción se ajusta a la nueva ubicación y continúa en pausa en el nuevo punto.

**Flujo Alternativo:**

1. Aplicación abierta.
2. Video en reproducción.
3. Se mueve el marcador de la barra de búsqueda.
4. Se ajusta a la nueva ubicación y su continúa reproducción.

**Excepción:**

\*Cuando se trata de mover el marcador durante la reproducción, es posible que el marcador no detecte correctamente la nueva ubicación y regrese a su ubicación original.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Detener la reproducción.	<b>ID</b>	1.37
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Con el propósito de detener completamente la reproducción, basta con pulsar el botón de Detener en la barra de herramientas.</p>
<p><b>Trigger:</b></p> <p>Botón de reproducción Detener.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Video detenido.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>

2. Video abierto.
3. Pulsar el botón de Detener.
4. Video detenido.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Avanzar frame.	<b>ID</b>	1.38
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Se dispone de un botón para pasar frame a frame el video, de forma que sea posible analizar cada imagen del video sin tener que estar pausando y deshaciendo la pausa a cada momento.</p>
<p><b>Trigger:</b></p> <p>Botón de reproducción Frame a Frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Video en pausa.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto y en reproducción.
3. Pulsar el botón de Avanzar Frame.
4. El video avanza un frame.
5. Video en pausa.

**Flujo Alternativo:**

1. Aplicación en ejecución.
2. Video abierto y en pausa.
3. Pulsar el botón de Avanzar Frame.
4. El video avanza un frame.
5. Video en pausa.

**Excepción:**

Quando se pulsa Avanzar Frame estando en la última posición, se notifica al usuario que ya se encuentra en el último frame. El video permanece en el último frame.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Retroceder frame.	<b>ID</b>	1.39
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	11-11-2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Se dispone de un botón para retroceder frame a frame el video, de forma que sea posible analizar cada imagen del video sin tener que estar pausando y deshaciendo la pausa a cada momento.</p>
<p><b>Trigger:</b></p> <p>Botón de Retroceder Frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Video en pausa.</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Aplicación en ejecución.</li><li>2. Video abierto y en reproducción.</li><li>3. Pulsar el botón de Retroceder Frame.</li><li>4. El video retrocede un frame.</li><li>5. Video en pausa.</li></ol>
<p><b>Flujo Alternativo:</b></p> <ol style="list-style-type: none"><li>1. Aplicación en ejecución.</li><li>2. Video abierto y en pausa.</li><li>3. Pulsar el botón de Retroceder Frame.</li><li>4. El video retrocede un frame.</li><li>5. Video en pausa.</li></ol>
<p><b>Excepción:</b></p> <p>Quando se pulsa Retroceder Frame estando en la primera posición, se notifica al usuario que ya se encuentra en el primer frame. El video permanece en el primer frame.</p>
<p><b>Includes:</b></p>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Cálculo de Flujo óptico para los puntos marcados	<b>ID</b>	1.40
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Si se desea que la aplicación siga los puntos marcados en la pantalla a lo largo de los frames, basta con pulsar el botón de Flujo óptico, indicado con la palabra "Auto" y a continuación darle a Reproducir (1.34).</p>
<p><b>Trigger:</b></p> <p>Botón de Flujo óptico.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Puntos marcados en el frame actual.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Punto marcado en los subsiguientes frames.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto.
3. Pulsar el botón de Flujo Óptico.
4. Pulsar el botón de Reproducir.

**Flujo Alternativo:****Excepción:****Includes:**

- Caso 1.34
- Caso 1.41
- Caso 1.46
- Caso 1.49

**Requisitos Especiales:****Notas:**

<b>Nombre</b>	Marcar punto en el frame actual	<b>ID</b>	1.41
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Se crea un marcador en la pantalla para un frame determinado. Para ello se pulsa sobre el botón de Marcar Frame y a continuación se pulsa sobre en la pantalla donde se desea establecer el marcador.</p>
<p><b>Trigger:</b></p> <p>Botón de Marcar punto.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Punto marcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto.
3. Situarse en el Frame deseado.
4. Pulsar Marcar puntos.
5. Pulsar sobre en la imagen donde se desea crear un marcador.
6. Vuelta al flujo normal del programa.

**Flujo Alternativo:****Excepción:**

Si ya existe un marcador en ese punto, se notifica al usuario y prosigue el flujo normal del programa.

**Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Borrar punto en el frame actual	<b>ID</b>	1.42
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Se borra un punto marcado en la pantalla, y todas sus relaciones. Para ello se pulsa sobre el botón de borrar y a continuación se pulsa sobre el punto que se desea borrar.</p>
<p><b>Trigger:</b></p> <p>Botón de borrado.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Puntos marcados en el frame actual.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Punto marcado eliminado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto.
3. Situarse en el Frame con marcadores que se desea borrar.
4. Pulsar Borrar puntos.
5. Pulsar sobre el punto que se desea borrar.
6. El punto, así como sus relaciones (ángulos) queda borrado.
7. Vuelta al flujo normal del programa.

**Flujo Alternativo:****Excepción:****Includes:**

- Caso 1.41
- Caso 1.46
- Caso 1.49

**Requisitos Especiales:****Notas:**

<b>Nombre</b>	Marcar Frame inicial	<b>ID</b>	1.43
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Marcar frame inicial para el cálculo de la diferencia de tiempo.</p>
<p><b>Trigger:</b></p> <p>Botón para marcar el Frame inicial.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Situarse sobre el primer Frame que se desea calcular.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Frame inicial marcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto.
3. Situarse sobre el primer Frame que se desea calcular.
4. Pulsar Marcar Frame inicial.
5. Frame inicial marcado
6. Vuelta al flujo normal del programa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Marcar Frame final	<b>ID</b>	1.44
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Marcar frame final para el cálculo de la diferencia de tiempo.</p>
<p><b>Trigger:</b></p> <p>Botón para marcar el Frame final.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Situarse sobre el último Frame que se desea calcular.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Frame final marcado.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Aplicación en ejecución.
2. Video abierto.
3. Situarse sobre el último Frame que se desea calcular.
4. Pulsar Marcar Frame final.
5. Frame final marcado
6. Vuelta al flujo normal del programa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Calcular el tiempo transcurrido entre los marcadores de Frame inicial y final	<b>ID</b>	1.45
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	18.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Existe la posibilidad de marcar dos frames del video para saber cuál es la diferencia de tiempo entre ambos. Para ello, una vez se han marcado los frames inicial y final, basta con pulsar el botón situado entre ambos.</p>
<p><b>Trigger:</b></p> <p>Botón para Calcular el tiempo entre 2 frames.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Frame inicial marcado.</li> <li>4. Frame final marcado.</li> <li>5. Medida de tiempo calibrada.</li> </ol>
<p><b>Postcondición:</b></p>

1. Muestra la diferencia de tiempo en pantalla.

**Flujo Normal:**

1. Aplicación en ejecución.
2. Video abierto.
3. Marcar los frames inicial y final haciendo uso de los casos de uso 1.43 y 1.44.
4. Una caja de texto nos indicará el frame de principio, final y su diferencia.
5. Aceptar retornará al flujo normal del programa.

**Flujo Alternativo:****Excepción:****Includes:**

- Caso 1.12
- Caso 1.43
- Caso 1.44

**Requisitos Especiales:****Notas:**

<b>Nombre</b>	Marcar punto en el frame actual para distancias	<b>ID</b>	1.46
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Existe la posibilidad de marcar dos (o más) puntos para calcular la distancia que los separa. Cada nuevo punto marcado se añade y calcula las distancias entre las zancadas del animal. Para ello es necesario primero haber marcado otros dos puntos que servirán de referencia y a los que se les asignará una distancia conocida.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Calcular distancias en un frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Medida de referencia calibrada previamente por el caso 1.11.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Muestra la distancia que los separa en la tabla y gráfica sobre distancias.</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Pulsar el botón “Marcar punto en el frame actual para distancias”.</li><li>2. Marca punto en la pantalla, tantas veces como sea necesario.</li><li>3. Se muestra en la tabla y gráfica los resultados del cálculo.</li></ol>
<p><b>Flujo Alternativo:</b></p>
<p><b>Excepción:</b></p>
<p><b>Includes:</b></p> <ul style="list-style-type: none"><li>• Caso 1.11</li></ul>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Establecer horizontal para el cálculo de zancadas automático	<b>ID</b>	1.47
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Para el cálculo automatizado de la distancia que los separa primero se ha de establecer la horizontal sobre la cual se considera que la zancada del animal empieza o acaba. Una vez hecho se marca el punto inicial de la zancada con el caso 1.46. Finalmente se deja que la operación de Flujo Óptico (caso 1.40) siga el rastro del marcador, en cada frame que pase de un lado a otro de la horizontal añadirá un nuevo punto marcado. Cada nuevo punto marcado se añade y calcula las distancias entre las zancadas del animal. Para ello es necesario primero haber marcado otros dos puntos que servirán de referencia y a los que se les asignará una distancia conocida.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Establecer horizontal para el cálculo de zancadas automático.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Video abierto.</li> <li>3. Medida de referencia calibrada previamente por el caso 1.11.</li> </ol>

<p><b>Postcondición:</b></p> <ol style="list-style-type: none"><li>1. Establece la horizontal de la zancada.</li></ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Pulsar el botón “Establecer horizontal para el cálculo de zancadas automático”.</li><li>2. Arrastrar el ratón por el frame hasta situarse en la horizontal deseada.</li></ol>
<p><b>Flujo Alternativo:</b></p>
<p><b>Excepción:</b></p>
<p><b>Includes:</b></p> <ul style="list-style-type: none"><li>• Caso 1.11</li><li>• Caso 1.40</li><li>• Caso 1.46</li></ul>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Calcular la distancia entre dos puntos de la imagen	<b>ID</b>	1.48
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Existe la posibilidad de marcar dos (o más) puntos en un mismo frame para calcular la distancia que los separa. Para ello es necesario primero haber marcado otros dos puntos que servirán de referencia y a los que se les asignará una distancia conocida.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Calcular distancias en un frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución</li> <li>2. Video en pausa.</li> <li>3. Medida de referencia calibrada previamente por el caso 1.11.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Muestra la distancia que los separa en un cuadro de texto.</li> <li>2. Video en pausa</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Aplicación en ejecución.</li><li>2. Video en pausa.</li><li>3. Pulsar el botón “Calcular distancias en un frame”.</li><li>4. Se marcan dos puntos en la imagen que es la distancia que se desea conocer.</li><li>5. Se muestra en un cuadro de texto el resultado.</li></ol>
<p><b>Flujo Alternativo:</b></p> <ol style="list-style-type: none"><li>1. Aplicación en ejecución.</li><li>2. Video en pausa.</li><li>3. Pulsar el botón “Calcular distancias en un frame”.</li><li>4. Se marca un punto y se arrastra hasta el segundo punto en la imagen que es la distancia que se desea conocer.</li><li>5. Se muestra en un cuadro de texto el resultado.</li></ol>
<p><b>Excepción:</b></p>
<p><b>Includes:</b></p> <ul style="list-style-type: none"><li>• Caso 1.11</li></ul>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Calcular el ángulo entre tres puntos de la imagen	<b>ID</b>	1.49
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Existe la posibilidad de marcar tres puntos en un mismo frame para calcular el ángulo de separación. A continuación es posible dejar que la operación de flujo óptico rastree la posición relativa de estos puntos en los frames consecutivos.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Calcular ángulos en un frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución</li> <li>2. Video en pausa.</li> </ol>
<p><b>Postcondición:</b></p> <ul style="list-style-type: none"> <li>• Muestra el ángulo de separación de 3 puntos.</li> </ul>
<p><b>Flujo Normal:</b></p>

1. Pulsar el botón "Calcular ángulos en un frame".
2. Se marcan en la imagen tres puntos.
3. Se calcula su ángulo de separación conforme al punto intermedio y se dibuja en el frame.

**Flujo Alternativo:**

1. Tras marcar los tres puntos se activa la opción AUTO para el flujo óptico.
2. Se pulsa el botón Reproducir en las herramientas.
3. La aplicación detectará la posición relativa de los puntos en los siguientes frames, marcándolos a su vez.

**Excepción:****Includes:**

- Caso 1.40

**Requisitos Especiales:****Notas:**

<b>Nombre</b>	Cálculo del ángulo conjugado	<b>ID</b>	1.50
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando el botón de Visor de detalles es posible cambiar el ángulo al ángulo conjugado. Para ello se pulsa sobre el botón de Visor de detalles y a continuación sobre el ángulo en concreto que se quiera visualizar.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Visor de detalles.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución</li> <li>2. Video en pausa.</li> <li>3. Ángulo calculado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Muestra el ángulo conjugado de separación de 3 puntos.</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Pulsar el botón “Visor de detalles”.</li><li>2. Se pulsa sobre el ángulo en cuestión.</li><li>3. Se calcula su ángulo conjugado y se dibuja en el frame.</li></ol>
<p><b>Flujo Alternativo:</b></p>
<p><b>Excepción:</b></p>
<p><b>Includes:</b></p>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Ocultar ángulos	<b>ID</b>	1.51
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando el botón de Visor de detalles es posible cambiar el ángulo conjugado y ocultarlo. Para ello se pulsa sobre el botón de Visor de detalles y a continuación sobre el ángulo en concreto que no se quiera visualizar.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Visor de detalles.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución</li> <li>2. Video en pausa.</li> <li>3. Ángulo conjugado.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Oculta el ángulo de separación de 3 puntos.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Pulsar el botón “Visor de detalles”.
2. Se pulsa sobre el ángulo conjugado en cuestión.
3. Se oculta el ángulo de la imagen.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Cambiar el color de la gráfica	<b>ID</b>	1.52
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Hay dos maneras de cambiar el color de las gráficas generadas en las pestañas correspondientes. El primer método consiste en pulsar el botón que hay junto al cuadro de texto, bajo la caja de herramientas. Este método abrirá la ventana predeterminada por el sistema para la selección de color personalizado. El segundo método consiste en situarse en el cuadro de texto bajo la barra de herramientas e introducir el nombre del color deseado en inglés y pulsar Enter. Para que ambos métodos tengan efecto sobre la gráfica debe cambiarse el color antes de marcar el punto deseado en la imagen.</p>
<p><b>Trigger:</b></p> <p>Pulsar el botón para Cambiar color.</p> <p>Ó</p> <p>Escribir el nombre del color deseado en el cuadro de texto.</p>
<p><b>Precondición:</b></p> <p>1. Aplicación en ejecución</p>

2. Video en pausa.
<b>Postcondición:</b>  1. El color del siguiente marcador en pantalla cambiará en la gráfica correspondiente.
<b>Flujo Normal:</b>  1. Pulsar el botón “Cambiar color”. 2. Seleccionar un color y pulsar Aceptar. 3. Regreso al flujo normal del programa.
<b>Flujo Alternativo:</b>  1. Escribir el nombre del color en la caja de texto. 2. Regreso al flujo normal del programa.
<b>Excepción:</b>
<b>Includes:</b>
<b>Requisitos Especiales:</b>
<b>Notas:</b>

<b>Nombre</b>	Seleccionar pestaña de ángulos	<b>ID</b>	1.53
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando sobre la pestaña llamada Ángulos se muestra la gráfica y tabla con la información relacionada con los ángulos calculados en el video.</p>
<p><b>Trigger:</b></p> <p>Pestaña de Ángulos</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Pestaña de distancia activa.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Pestaña de ángulos activa.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Pulsar la pestaña "Ángulos".</li> </ol>

2. Regreso al flujo normal del programa.
<b>Flujo Alternativo:</b>
<b>Excepción:</b>
<b>Includes:</b>
<b>Requisitos Especiales:</b>
<b>Notas:</b>

<b>Nombre</b>	Seleccionar pestaña de distancias	<b>ID</b>	1.54
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando sobre la pestaña llamada distancia se muestra la gráfica y tabla con la información relacionada con las distancias calculadas en el video.</p>
<p><b>Trigger:</b></p> <p>Pestaña de distancia</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Pestaña de ángulos activa.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Pestaña de distancia activa.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Pulsar la pestaña "Distancia".</li> </ol>

2. Regreso al flujo normal del programa.
<b>Flujo Alternativo:</b>
<b>Excepción:</b>
<b>Includes:</b>
<b>Requisitos Especiales:</b>
<b>Notas:</b>

<b>Nombre</b>	Desplazar gráfica	<b>ID</b>	1.55
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Manteniendo pulsado el botón derecho sobre la gráfica y moviendo el cursor logramos desplazarnos por la gráfica.</p>
<p><b>Trigger:</b></p> <p>Manteniendo pulsando el botón derecho del ratón sobre la gráfica.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Vista de la Gráfica trasladada.</li> </ol>
<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"> <li>1. Mantener pulsando el botón derecho del ratón sobre la gráfica.</li> <li>2. Mover el cursor a donde se desee.</li> </ol>

3. Regreso al flujo normal del programa.
<b>Flujo Alternativo:</b>
<b>Excepción:</b>
<b>Includes:</b>
<b>Requisitos Especiales:</b>
<b>Notas:</b>

<b>Nombre</b>	Acercar gráfica	<b>ID</b>	1.56
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando el Botón derecho del ratón y seleccionando Zoom in logramos cambiar la vista y acercarla. También es posible Manteniendo pulsado el botón izquierdo y arrastrándolo para formar un cuadro de la zona sobre la que se quiere visualizar en más detalle.</p>
<p><b>Trigger:</b></p> <p>Manteniendo pulsando el botón izquierdo del ratón sobre la gráfica.</p> <p>Ó</p> <p>Pulsando el botón derecho y seleccionando Zoom in.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Vista de la Gráfica cambiada.</li> </ol>

<p><b>Flujo Normal:</b></p> <ol style="list-style-type: none"><li>1. Mantener pulsando el botón izquierdo del ratón sobre la gráfica.</li><li>2. Mover el cursor para formar un cuadro de la zona que se desea acercar.</li><li>3. Vista de la gráfica cambiada.</li><li>4. Regreso al flujo normal del programa.</li></ol>
<p><b>Flujo Alternativo:</b></p> <ol style="list-style-type: none"><li>1. Pulsar el botón derecho del ratón sobre la gráfica.</li><li>2. Seleccionar Zoom in.</li><li>3. Vista de la gráfica cambiada.</li><li>4. Regreso al flujo normal del programa.</li></ol>
<p><b>Excepción:</b></p>
<p><b>Includes:</b></p>
<p><b>Requisitos Especiales:</b></p>
<p><b>Notas:</b></p>

<b>Nombre</b>	Alejar gráfica	<b>ID</b>	1.57
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Pulsando el Botón derecho del ratón y seleccionando Zoom out logramos cambiar la vista y alejarla. También es posible pulsando la opción Fit, que encuadra la gráfica completa en el cuadro.</p>
<p><b>Trigger:</b></p> <p>Pulsando el botón derecho y seleccionando Fit.</p> <p>Ó</p> <p>Pulsando el botón derecho y seleccionando Zoom out.</p>
<p><b>Precondición:</b></p> <p>1. Aplicación en ejecución.</p>
<p><b>Postcondición:</b></p> <p>1. Vista de la Gráfica cambiada.</p>
<p><b>Flujo Normal:</b></p>

1. Pulsar el botón derecho del ratón sobre la gráfica.
2. Seleccionar Fit.
3. Vista de la gráfica cambiada.
4. Regreso al flujo normal del programa.

**Flujo Alternativo:**

1. Pulsar el botón derecho del ratón sobre la gráfica.
2. Seleccionar Zoom out.
3. Vista de la gráfica cambiada.
4. Regreso al flujo normal del programa.

**Excepción:****Includes:****Requisitos Especiales:****Notas:**

<b>Nombre</b>	Crear nueva anotación	<b>ID</b>	1.58
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Con el video pausado, escribiendo sobre el cuadro de texto para las anotaciones, y a continuación pulsando Guardar Nota.</p>
<p><b>Trigger:</b></p> <p>Botón Guardar Nota.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. No hay notas en el frame actual.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Nota almacenada.</li> <li>2. Nuevo Frame marcado con anotación.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Pausar video.
2. Escribir anotación.
3. Pulsar Guardar Nota.
4. Regreso al flujo normal del programa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Sobreescribir anotación	<b>ID</b>	1.59
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Con el video pausado, escribiendo sobre el cuadro de texto para las anotaciones, ya sea modificando la nota existente o añadiendo más anotaciones al mismo. A continuación pulsando Guardar Nota.</p>
<p><b>Trigger:</b></p> <p>Botón Guardar Nota.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Hay una anotación en el frame actual.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Nota almacenada.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Pausar video.
2. Escribir anotación.
3. Pulsar Guardar Nota.
4. Regreso al flujo normal del programa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**

<b>Nombre</b>	Seleccionar anotación	<b>ID</b>	1.60
<b>Creado por</b>	Deriman Franco García	<b>Fecha</b>	23.11.2015
<b>Modif. por</b>		<b>Fecha Modif</b>	

<p><b>Actor principal:</b></p> <p>Usuario</p>
<p><b>Personal involucrado o intereses:</b></p> <p>Usuario</p>
<p><b>Descripción:</b></p> <p>Con la aplicación pausada, pulsando sobre el campo de selector de Frame la aplicación nos dirigirá al Frame marcado y abrirá la nota existente en el campo de Anotaciones.</p>
<p><b>Trigger:</b></p> <p>Selección de Frame.</p>
<p><b>Precondición:</b></p> <ol style="list-style-type: none"> <li>1. Aplicación en ejecución.</li> <li>2. Hay anotaciones almacenadas.</li> </ol>
<p><b>Postcondición:</b></p> <ol style="list-style-type: none"> <li>1. Reproducción trasladada al punto en el que la anotación tuvo lugar.</li> <li>2. Nota abierta en el campo Anotaciones.</li> </ol>
<p><b>Flujo Normal:</b></p>

1. Pausar video.
2. Seleccionar Frame en la lista de anotaciones.
3. Reproducción se traslada al punto seleccionado, anotación relacionada abierta.
4. Regreso al flujo normal del programa.

**Flujo Alternativo:**

**Excepción:**

**Includes:**

**Requisitos Especiales:**

**Notas:**