UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

PROYECTO FIN DE CARRERA

SEGMENTACIÓN ESPACIAL DE SECUENCIAS MPEG EN EL DOMINIO COMPRIMIDO

Marcos Escudero Viñolo

Febrero 2008

SEGMENTACIÓN ESPACIAL DE SECUENCIAS MPEG EN EL DOMINIO COMPRIMIDO

AUTOR: Marcos Escudero Viñolo
TUTOR: Fabrizio Tiburzi Paramio

Grupo de Tratamiento de Imágenes GTI-UAM
Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Febrero de 2008

PROYECTO FIN DE CARRERA

Título: *Segmentación espacial de secuencias MPEG en el dominio comprimido*

Autor: D. Marcos Escudero Viñolo
Tutor: D. Fabrizio Tiburzi Paramio

Tribunal:

Presidente: José M. Martínez Sánchez

Vocal: Kostadin Koroutchev

Vocal secretario: Jesús Bescós Cano

Fecha de lectura: 22 de Febrero de 2008

Calificación:

Palabras clave:

Vectores de movimiento, segmentación de objetos, dominio comprimido, macrobloque, MPEG, Background multi-modal, intra-codificación, cuadro, movimiento de cámara, video-seguridad.

Resumen:

Este proyecto parte de una aproximación eficiente para la segmentación de objetos en tiempo real basada en la información de movimiento disponible en el dominio comprimido MPEG, aproximación que aborda el caso general en que la cámara no está fija. Se estudian diferentes técnicas para mejorar la citada aproximación o *algoritmo base* y ampliar su área de funcionamiento a la mayoría de situaciones posibles. Estas técnicas están encaminadas a mejorar la gestión de macrobloques intra-codificados, explotar la coherencia temporal de los objetos en movimiento utilizar la información de color disponible en el dominio comprimido para refinar las máscaras de segmentación a nivel de macrobloque. Asimismo se propone un mecanismo que permite la gestión de *backgrounds* multi-modales, en los que pueden aparecer peturbaciones debido, por ejemplo, a la influencia del viento o a la presencia de elementos como agua o fuego. Para la evaluación de los resultados obtenidos tanto cuantitativa como cualitativamente, se ha diseñado un *ground-truth* de secuencias representativas. Estas secuencias se han grabado en un estudio de croma con el objetivo de poder obtener las máscaras de objetos de manera casi automática.

Abstract:

This project starts from a state-of-the-art efficient approach to real-time video object segmentation in the MPEG domain. It then describes several techniques to extend this approach’s behavior to a more generic set of situations. These are focused on the management of intra-coded macroblocks, the exploitation of objects motion coherence, the integrated use of color information and an approach to discriminate objects from multimodal background (e.g., water, flames), always under camera motion conditions. Results evaluation is presented over a ground truth specifically generated with the help of chroma studio in order to reproduce almost real sequences in a controlled way.

Agradecimientos

Gracias a mi tutor, Fabricio, por su paciencia y porque este proyecto también es suyo.

Gracias a Jesús y a Chema por darme la posibilidad de hacerlo, y confiar en mi ímpetu y perseverancia. Gracias a Jesús, de nuevo, por las oportunidades que me ha dado.

Gracias a cada uno de los componentes del GTI que siempre han estado dispuestos a echarme una mano, ofrecerme una sonrisa y apoyarme con un comentario alentador.

Gracias a mis padres por todo lo que han hecho por mí, su mejor legado es la educación que me han dado, mi mejor aliado, su apoyo y cariño.

Gracias a mis abuelos y al resto de mi familia por creer en mí, motivarme y no dejar que me rinda nunca.

Gracias a mis amigos porque sin ellos no existiría emoción en la vida, por saberlos siempre disponibles, por saberlos eternos, porque no existen mejores personas que ellos.

Gracias a mi hermana, por enseñarme que la responsabilidad y la constancia pueden coexistir con la alegría, por no dejar que nunca me salga del camino.

Gracias a Elena por ser la luz que me guía cuando todo es oscuro, por volar a mi mente siempre que la necesito, por estar a mi lado siempre que añoro su calor. Por ser.

INDICE DE CONTENIDOS

1 Introducción 4
1.1 Motivación 4
1.2 Objetivos 4
1.3 Organización de la memoria 4
2 Estado del arte 6
2.1 Video MPEG 6
2.1.1 Introducción 6
2.1.2 Nomenclatura 6
2.1.3 La DCT 8
2.1.4 Compensación del movimiento 9
2.2 Información disponible en el dominio comprimido 10
2.2.1 Introducción 10
2.2.2 Información de color. 11
2.2.3 Información de movimiento. 11
2.3 Aplicaciones de la información de movimiento 12
2.3.1 Indexación de video 12
2.3.2 Movimiento de cámara 12
2.3.3 Segmentación basada en movimiento 13
2.3.3.1 Segmentación de objetos en situación de cámara fija 14
2.3.3.2 Segmentación de objetos en situaciones con movimiento de cámara 14
3 Descripción del algoritmo base 16
3.1 Arquitectura del sistema 16
3.2 Módulo para la extracción de la información disponible en el dominio comprimido 17
3.3 Módulo de rechazo iterativo 17
3.3.1 Fundamento 17
3.3.2 Descripción 17
3.4 Módulo de tracking a nivel de macrobloque 18
3.4.1 Fundamento 18
3.4.2 Descripción 19
3.5 Módulo de formación de objetos finales 19
3.5.1 Fundamento 19
3.5.2 Descripción 20
4 Descripción de las mejoras introducidas 23
4.1 Motivación 23
4.2 Arquitectura del sistema 23
4.3 Módulo para tratamiento de macrobloques intra-codificados 24
4.3.1 Fundamento 24
4.3.2 Descripción 25
4.4 Módulo para evitar la desaparición momentánea de objetos 26
4.4.1 Fundamento 26
4.4.2 Descripción 27
4.5 Módulo de apoyo por color a la segmentación 28
4.5.1 Fundamento 28
4.5.2 Descripción 29
4.6 Módulo para la gestión de backgrounds multi-modales 31
4.6.1 Fundamento 31

4.6.2 Descripción 32
5 Integración, pruebas y resultados 35
5.1 Integración 35
5.2 Pruebas 35
5.3 Resultados 35
5.3.1 Secuencias de prueba 36
5.3.2 Resultados cuantitativos 37
5.3.3 Resultados cualitativos 38
5.4 Resultados en proceso de publicación 42
6 Conclusiones y trabajo futuro 43
6.1 Conclusiones 43
6.2 Trabajo futuro 44
Referencias 46
Glosario 48
Anexos I
A Motivación, técnica y montaje de las escenas en croma I
A.1 Introducción I
A.2 Consideraciones para el diseño del Ground-Truth II
A.3 Factores críticos en los algoritmos de segmentación de objetos basados en movimiento III
A.4 Creación de las secuencias VI
A.5 Referencias VI
B Guiones VII
7 Presupuesto I

INDICE DE FIGURAS

FIGURA 1: ARQUITECTURA DEL ALGORITMO BASE 16

FIGURA 2: ARQUITECTURA DEL ALGORITMO PROPUESTO 23

FIGURA 3: ACIERTO EN OBJETOS FRENTE A ACIERTO EN RUIDOS PARA LAS POSIBLES UMBRALIZACIONES DEL LDA 33

FIGURA 4: EJEMPLOS DE CUADROS PERTENECIENTES A LA *SECUENCIA 1* PARA LA REALIZACIÓN DE PRUEBAS. 36

FIGURA 5: EJEMPLOS DE CUADROS PERTENECIENTES A LA *SECUENCIA 2* PARA LA REALIZACIÓN DE PRUEBAS. 37

FIGURA 6: EJEMPLOS DE CUADROS PERTENECIENTES A LA *SECUENCIA 3* PARA LA REALIZACIÓN DE PRUEBAS. 37

FIGURA 7: EJEMPLOS DE LOS RESULTADOS CUALITATIVOS OBTENIDOS PARA LA *SECUENCIA 1* 39

FIGURA 8: EJEMPLOS DE LOS RESULTADOS CUALITATIVOS OBTENIDOS PARA LA *SECUENCIA 2* 40

FIGURA 9: EJEMPLOS DE LOS RESULTADOS CUALITATIVOS OBTENIDOS PARA LA *SECUENCIA 3* 41

INDICE DE TABLAS

TABLA 1. COMBINACIONES DE ACIERTO EN RUIDO Y ACIERTO EN OBJETO CONSIDERADAS. (EXTRAÍDAS DE LA CURVA DE LA FIGURA 3). 33

TABLA 2. RESULTADOS CUANTITATIVOS DE CADA UNO DE LOS ALGORITMOS (BASE Y PROPUESTO) PARA LA *SECUENCIA 1*..... 38

TABLA 3. RESULTADOS CUANTITATIVOS DE CADA UNO DE LOS ALGORITMOS (BASE Y PROPUESTO) PARA LA *SECUENCIA 2*..... 38

1 Introducción

1.1 Motivación

La motivación de este Proyecto Fin de Carrera (PFC) es por un lado, el diseño, implementación e integración modular de un sistema capaz de segmentar objetos en movimiento en tiempo real y sin supervisión humana, sobre el conjunto más amplio de situaciones posible, tanto en casos de cámara fija como de cámara móvil. Para ello se partirá de un algoritmo representativo del estado del arte en esta cuestión, que hemos considerado ventajoso en términos de eficiencia y calidad de los resultados..Nos referiremos a éste como *algoritmo base*.

En segundo lugar, intentaremos mejorar el algoritmo base, proponiendo soluciones a problemas o carencias observados tanto en su diseño, como en los resultados obtenidos tras su implementación. Estas soluciones han estado inspiradas en el estudio exhaustivo del estado del arte y en el diseño de nuevas técnicas.

1.2 Objetivos

Nuestro objetivo es el de centrar las bases para el diseño de un sistema de ámbito global, que pueda ser usado en las múltiples aplicaciones posibles que, a nuestro entender, puede cubrir el área de segmentación no supervisada de objetos en tiempo real.

La memoria se redactará intentando no presuponer más que un conocimiento mínimo del área, para facilitar su comprensión a todo aquel interesado.

Los puntos a abordar en el proyecto pueden enumerarse:

i. Estudio del estado del arte actual.
ii. Implementación del algoritmo base.
iii. Estudio y gestión de los macrobloques intra-codificados
iv. Estudio y gestión de la coherencia temporal de objetos en movimiento
v. Estudio y gestión de la aplicación de información cromática para el refinamiento de las máscaras de objetos.
vi. Estudio y gestión de la aplicación del algoritmo a entornos de background multi-modal.

1.3 Organización de la memoria

La memoria consta de los siguientes capítulos:

- **Capítulo 1:** Introducción, objetivos y motivación del proyecto.

- **Capítulo 2:** Breve descripción de las características utilizadas en el proyecto del estándar MPEG, así como de la información disponible en el dominio comprimido y de las técnicas de segmentación basadas en movimiento existentes.

- **Capítulo 3:** Descripción modular detallada del algoritmo base, de su diseño y arquitectura, dle fundamento y funcionamiento de cada una de sus etapas. Análisis crítico de las situaciones no consideradas por el algoritmo base.

- **Capítulo 4:** Descripción modular detallada del sistema propuesto. Diseño y arquitectura. Motivación. fundamento y funcionamiento de cada una de las etapas de mejora introducidas.

- **Capítulo 5:** Elaboración de pruebas, generación de secuencias representativas. Presentación comparativa de los resultados obtenidos por el algoritmo base frente a los obtenidos por el propuesto.

- **Capítulo 6:** Conclusiones obtenidas tras el análisis de resultados. Problemas pendientes y trabajo futuro tanto para solventarlos, como para mejorar el sistema final propuesto.

- **Anexo A:** Artículo de investigación sobre la motivación, justificación y funcionamiento de la técnica del croma. Análisis de situaciones consideradas para la generación de las secuencias prueba.

- **Anexo B:** Guiones para la grabación de las secuencias prueba.

2 Estado del arte

2.1 Video MPEG

2.1.1 Introducción

Una explicación detallada del estándar de compresión MPEG (*Moving Pictures Expert Group*), se aleja de los objetivos de este PFC. Sin embargo, consideramos imprescindible un breve resumen de aquellas características de las que haremos uso en el proyecto.

MPEG es el estándar más utilizado en la compresión de video y audio para imágenes en movimiento. Analizaremos brevemente los estándares MPEG-1 y MPEG-2, haciendo especial hincapié en la parte dedicada a la predicción de movimiento en videos, base fundamental para la correcta comprensión del trabajo realizado y presentado en esta memoria.

ISO/IEC 11172 es el estándar donde se describe MPEG-1, nos centraremos en la segunda parte de éste, la destinada a vídeo. MPEG Video está diseñado explícitamente para la compresión de secuencias de video, es decir, para la compresión de una serie de imágenes separadas por un breve intervalo de tiempo.

MPEG video es un sistema híbrido de codificación, basado tanto en el uso de esquemas predictivos (basados en la estimación y compensación de movimiento) para sacar beneficio de la redundancia, como en la utilización de transformadas (la *Discrete Cosine Transform*, DCT) como compactadores de la energía de los errores de predicción.

La compensación de movimiento hace uso de la similitud entre una imagen y la siguiente. Estas técnicas de compresión basadas en información obtenida a partir de otras imágenes en la secuencia se denominan *interframe techniques.*

Cuando no disponemos de otra información aparte de la contenida en la propia imagen las técnicas de compresión deben hacer uso de las similitudes entre una parte de la imagen y sus partes contiguas. Este tipo de técnicas se denominan *intraframe techniques*. En el caso de MPEG-1/2 esta técnica se basa en la Transformada Discreta del Coseno (del inglés, DCT) calculada por bloques de 8 x 8.

Describiremos brevemente estas técnicas de codificación en las secciones 2.1.3 y 2.1.4, pero antes es necesario exponer aquella nomenclatura incluida en el estándar MPEG de la que haremos uso a lo largo del documento.

2.1.2 Nomenclatura

MPEG Video está diseñado como un sistema estratificado de capas. Cada secuencia de video en MPEG se divide en uno o más GOP (*Group of pictures* o grupo de imágenes). A su vez, cada grupo de imágenes contiene una o más imágenes (*picture*) de cuatro tipos posibles: *I, P, B y D*.

Las *I-pictures* se caracterizan por estar codificadas mediante *intraframe techniques* de manera independiente al resto de las imágenes del GOP. Aunque ocupan un mayor tamaño benefician la sincronización y son necesarias para garantizar el acceso aleatorio.

Tanto para las *P-pictures* como para las *B-pictures* se hace uso de las similitudes entre éstas y otras imágenes de referencia para su codificación. Las *P-pictures* obtienen predicciones de imágenes I ó P anteriores, mientras que las *B-pictures* toman como referencia las imágenes I o P más cercanas tanto anteriores como posteriores. Existe la posibilidad de codificar este tipo de imágenes total o parcialmente sin usar ninguna predicción, es decir, con *intraframe techniques*.

Existe otro tipo de imagen definida en el estándar, *D-pictures*, pero su uso no es común, se trata de una imagen en baja resolución, que no puede ser usada en combinación con los otros tipos de imagen.

MPEG-2 añade la posibilidad de de trabajar con secuencias entrelazadas para alcanzar tasas de transmisión más altas. En este contexto cabe destacar la diferencia entre el concepto imagen o cuadro (*frame*) y el concepto de campo, puesto que en MPEG-2 un cuadro puede contener campos I, P y/o B en pares de combinaciones restringidas por el tipo de cuadro [1].

Cada imagen MPEG está compuesta de *slices*. Una *slice* es una secuencia contigua de macrobloques. Esta estructura proporciona una gran flexibilidad en la señalización de cambios sobre algunos parámetros de codificación, así como también permite la optimización y el control de la tasa de bits requerida.

Un macrobloque consiste en una matriz de 16 x 16 muestras para luminancia (escala de grises) junto con dos matrices, generalmente de 8 x 8 muestras, una para cada una de las componentes de crominancia (color) asociadas al macrobloque. MPEG usa un sistema de representación de color YCbCr, con el cual pueden alcanzarse niveles más altos de compresión al estar más cercano a la percepción humana de los colores que RGB. En particular, debido a que el ojo humano no resuelve cambios espaciales rápidos en crominancia (Cb, Cr) tan fácilmente como cambios en luminancia (Y), las componentes de crominancia se pueden muestrear con una menor resolución espacial que la de luminancia.

En MPEG cada una de las matrices de crominancia se muestrea con una resolución dos veces menor que el componente de luminancia, por lo que cada macrobloque está compuesto de cuatro bloques 8 x 8 de luminancia y dos bloques 8 x 8 de crominancia. Este muestreo se define en MPEG-1 como *4:2:0.*

Cada píxel está definido a la mayor resolución espacial de muestreo, es decir, a la de luminancia; por lo tanto, la posición de las componentes de crominancia respecto de la luminancia ha de estar definida, ya que esta posición influirá en la representación del valor del píxel. MPEG-1 sólo permite un formato para este muestreo *4:2:0*, mientras que MPEG-2 alinea de manera diferente las muestras para esta resolución, y además define dos nuevas resoluciones: *4:2:2* y *4:4:4* con el objetivo de ofrecer la posibilidad de obtener mayor calidad digital

2.1.3 La DCT

La DCT permite descomponer un bloque de datos de tamaño 8x8 en una suma ponderada de frecuencias espaciales. Cada una de las frecuencias espaciales está asociada a un coeficiente que representa la contribución de ese patrón de frecuencia al bloque de datos analizados. Así, cada patrón de frecuencia se multiplica por su coeficiente asociado, y las 64 matrices 8 x 8 resultantes son sumadas píxel a píxel para reconstruir el bloque original.

Dependiendo de los coeficientes asociados a cada una de las frecuencias podemos intuir datos acerca del bloque del que provienen. Si sólo los coeficientes correspondientes a las bajas frecuencias son distintos de cero, entonces existirá poca variación entre los píxeles del bloque. Si por el contrario, las altas frecuencias están presentes y son no nulas, la intensidad del bloque cambia rápidamente píxel a píxel.

La componente DC, *direct current* (la frecuencia más baja), de cada bloque se corresponde con su valor medio e incluye la información más importante de la imagen desde un punto de vista perceptual. Las altas frecuencias espaciales AC, *alternative current*, contienen los detalles del bloque, por lo que, nuevamente atendiendo a criterios perceptuales, podríamos codificarlas con menor precisión que la componente DC. Esto se controla mediante el proceso de cuantificación.

Cada uno de los coeficientes se divide por un valor entero no nulo denominado valor de cuantificación, y redondeando este cociente se obtiene finalmente el coeficiente DCT cuantificado. La tabla de cuantificación es una matriz con 64 valores, uno por cada coeficiente DCT. Existen dos tablas de cuantificación posibles para MPEG-1, una para *intra-techniques* que está en concordancia con la respuesta en frecuencia del ojo humano, y otra para *inter-techniques* con un valor fijo de 16 en todos sus componentes.

Los valores de cuantificación *intra* se fijan de forma que se asocien a las altas frecuencias valores muy altos de cuantificación. Así, la precisión de estos coeficientes DCT cuantificados será menor, lo que permite al codificador descartar selectivamente valores de alta frecuencia espacial (que el ojo humano no puede apreciar fácilmente). Con una cuantificación inteligente, podemos conseguir que casi todas las altas frecuencias espaciales tengan un valor asociado muy bajo que, con el redondeo, será nulo, lo cual será de gran ayuda en la obtención de una codificación eficiente.

La DCT ha demostrado poseer muchas ventajas desde el punto de vista de la compresión de datos para MPEG. La principal es el uso de los coeficientes DCT para *intra coding*. Estos coeficientes están casi completamente decorrelados, es decir, son independientes unos de otros, lo cual permite crear un algoritmo relativamente simple para codificarlos. La decorrelación es de gran interés teórico y práctico para la construcción de un modelo de codificación simple.

Los coeficientes DCT cuantificados han de ser codificados con las menores pérdidas posibles para la tasa de bit requerida, de este modo, el decodificador podrá reconstruir valores más próximos a los originales.

MPEG codifica entrópicamente mediante la codificación Huffman, técnicas predictivas como DPCM (*Diferential Pulse Code Modulation*) para los coeficientes DC, y el uso de

símbolos especiales (EOB, *run-level*) para los AC. Una descripción detallada de las técnicas de codificación utilizadas por el estándar puede encontrarse en [1].

2.1.4 Compensación del movimiento

La propiedad de la decorrelación de la DCT sólo es útil en la codificación *intra*, ya que en codificación *inter-frame*, lo que realmente se codifica son los errores de predicción relativos entre imágenes. Consecuentemente, en este proceso de predicción, la decorrelación viene intrínseca.

La compensación de movimiento basada en bloques reduce significativamente la redundancia temporal. Para cada bloque existente en la trama se busca el bloque más similar de la imagen de referencia; de esta manera sólo la diferencia entre el bloque actual y el coincidente ha de ser codificada.

En un plano más general, podemos abstraer que los píxeles de una región en una imagen son comparados con los píxeles de una región en otra u otras imágenes de referencia. Las diferencias entre una y otra región son codificadas con la mayor precisión posible teniendo en cuenta la tasa de bit requerida. La región más similar en la o las imágenes de referencia se relaciona con la región original mediante un vector de compensación de movimiento (Motion Compensation Vector, MV). De esta manera, para cada imagen, el movimiento aparente de sus regiones puede ser descrito por una matriz bidimensional de vectores de movimiento que indiquen desplazamientos relativos a una imagen de referencia

Las tres cuestiones principales en el uso de compensación del movimiento son:

- La precisión de los vectores de movimiento.
- El tamaño de las regiones a las que se les asigna un vector de movimiento.
- El criterio de selección utilizado para elegir el vector de movimiento.

El decodificador sólo interpreta los datos que recibe; por lo tanto, es misión del codificador definir estos parámetros para calcular los vectores que mejor describan el movimiento de una región dada.

El tamaño de la región a la que se asigna el vector de movimiento determina el número de vectores de movimiento necesarios para describir el movimiento total de una imagen. Ha de existir un compromiso entre la precisión a la hora de describir movimientos complejos y el coste de calcular y transmitir los vectores de movimiento.

En MPEG-1/2 la región es un macrobloque, el cual, como hemos dicho previamente está compuesto de seis bloques, cuatro de luminancia y dos de crominancia. Como un mismo vector de desplazamiento esta asociado a los seis bloques, el desplazamiento debe de ser escalado para reflejar las diferentes resoluciones entre crominancia y luminancia.

La precisión de los vectores ha de ser intuitivamente de al menos un píxel, pero a veces se acude a aún mayor precisión. Ha de existir un compromiso entre el coste de transmitir una mayor precisión y la exactitud de la descripción. MPEG-2 siempre usa precisión de medio píxel en sus vectores de movimiento.

Existen varias técnicas para la búsqueda del macrobloque más similar en la imagen de referencia. Una de ellas, la más costosa, es la búsqueda exhaustiva, en la que se compara cada uno de los macrobloques de la imagen actual con todos los de la de referencia incluidos en su ventana de búsqueda, este tipo de búsqueda devuelve resultados muy precisos a pesar de consumir más tiempo que otras técnicas.

Hemos de tener en cuenta que la búsqueda de similitudes entre macrobloques es el proceso que más tiempo consume durante la fase de codificación; por ello, si estamos sujetos a restricciones temporales, existen otros algoritmos de búsqueda más rápidos. Estos algoritmos, asumiendo distintos condicionantes respecto a la distorsión y utilizando métricas eficientes para medirla, consiguen esquemas de búsqueda rápidos y de aceptables resultados. Un buen resumen y discusión de las principales técnicas existentes puede encontrarse en [2].

2.2 Información disponible en el dominio comprimido

2.2.1 Introducción

La información disponible en el dominio comprimido, es decir sin decodificar, nos permite realizar numerosas tareas de análisis en tiempo real pues la carga de operaciones involucrada en el procesamiento de estos datos es mucho menor que la requerida para decodificar y analizar la información descomprimida. Sin embargo, los datos que extraemos del dominio comprimido constituyen en muchos casos tan sólo una aproximación a la información real de análisis en la que estamos interesados, riesgo que deberemos asumir cuando los utilicemos.

Dividiremos el tipo de información que puede extraerse del dominio comprimido, en varias categorías a saber:

- Información visual-espacial.
- Información sobre aspectos de la codificación
- Información sobre el movimiento.

En lo que respecta a información visual espacial podemos extraer datos relativos al color, textura o formas de objetos presentes en un videoLa información de color disponible en el dominio comprimido será clave en el desarrollo de nuestro proyecto, como veremos en la sección 4.1.3.

En la categoría de codificación es muy frecuente estudiar los modos de codificación de cada macrobloque en relación a la tasa binaria disponible (*forward-predicted, backward-predicted, bidirectionally-predicted, intra-coded y skiped*), pues ciertas estadísticas sobre estos bloques pueden suministrar valiosa información para indexación y análisis de cuadros P y B.

Finalmente la información disponible de movimiento, a través de los MV, permite abordar áreas de estudio entre las que se incluyen la indexación de video, el movimiento de cámara o la segmentación de objetos.

A continuación pasaremos a analizar cómo esta información de bajo nivel puede ser gestionada eficientemente. Adicionalmente también se discutirán otras aplicaciones alternativas que se apoyan en ella y que, de alguna forma, se han considerado de interés.

El algoritmo base del que se parte hace uso casi exclusivo de la información de movimiento.

2.2.2 Información de color.

Las imágenes DC son imágenes cuyos píxeles corresponden al coeficiente DC de la DCT de cada bloque. Son, por lo tanto, imágenes de dimensión 8x8 veces menor que los cuadros del vídeo. Una secuencia de imágenes DC es una representación icónica del video original. A pesar de estar a menor resolución, contiene el contenido clave del video (a efectos perceptuales) y resulta muy especialmente útil para extraer información visual como el color. Esta información de color ha demostrado ser efectiva a la hora de indexar, recuperar y analizar video.

Los valores DC de la luminancia y la crominancia pueden usarse para construir vectores de color que permitan medir la similitud entre dos áreas de un cuadro determinado, así como para detectar cambios de plano o extracción de cuadros clave [3][4][5][6].

2.2.3 Información de movimiento.

La información de movimiento disponible en el dominio comprimido no representa el movimiento real de la escena sino que indican adónde se ha desplazado la región más parecida (que es lo realmente útil a efectos de eliminar redundancias). En [7] se recogen varias desventajas, o cuestiones a tener en cuenta, al utilizar los vectores de compensación de movimiento como estimadores del flujo óptico o movimiento aparente de la secuencia. En resumen:

- Los MV no representan el movimiento real, y dependen mucho de la estrategia del codificador empleado a la hora de obtener la estimación del movimiento.
- Codificadores de baja complejidad pueden utilizar algoritmos de búsqueda con rangos restringidos, por lo que no habrá vectores de movimiento grandes.
- El rendimiento del algoritmo de estimación depende mucho del GOP empleado en la codificación y de la tasa de cuadro utilizada.
- Los MV (especialmente los pequeños) se ven muy influidos por el ruido, y pueden no ser fiables en lo que a movimiento respecta.
- Existen áreas donde la probabilidad de encontrar vectores de movimiento erróneos se ve fuertemente incrementada; entre ellas están los bloques en los bordes del cuadro y las zonas homogéneas como el cielo.
- Los cuadros I no tienen MV, por lo que los vídeos codificados sólo con cuadros I no podrán ser analizados con algoritmos basados en la información de movimiento. De la misma forma, tampoco se pueden utilizar estos algoritmos en otros formatos de codificación como MJPEG, basado también en DCT pero sin compensación de movimiento.

A pesar de todo ello, la información de movimiento disponible en el dominio comprimido es pieza clave y principal en nuestro PFC; por ello, ahondaremos en ella con una mayor extensión. Comentaremos algunas técnicas relacionadas con la indexación de video o el

análisis del movimiento de cámara, pero haremos especial hincapié en la descripción de los trabajos más importantes llevados a cabo en el área de segmentación de objetos.

2.3 Aplicaciones de la información de movimiento

2.3.1 Indexación de video

La información de movimiento puede usarse para predecir de manera aproximada el contenido de uno o varios cuadros o incluso de toda la secuencia.

Los MV nos permiten buscar regiones determinadas en un video observando a lo largo del tiempo coincidencia en las trayectorias de movimiento [8]. Por otro lado, en [9] se propone un método para obtener, a partir de la información de movimiento, una estimación del tamaño, forma y número de objetos en movimiento presentes en el cuadro (distribución de la actividad del movimiento).

En [10] describen la posibilidad de extraer los cuadros claves que contengan la información principal de la secuencia, mediante la utilización de una curva que represente la cantidad de movimiento en función de de cada cuadro.

2.3.2 Movimiento de cámara

Entenderemos por movimiento de la cámara que graba, el conjunto de rotaciones, traslaciones y cambios en la distancia focal sobre o respecto a los tres ejes de la cámara. Existe una nomenclatura normalmente aceptada en casi todos los trabajos sobre movimiento de cámara, así como en los principales estándares, entre los que se incluye MPEG[11]. Respetaremos esta nomenclatura que denomina a las translaciones sobre los ejes vertical, horizontal y óptico, *track, boom* y *dolly* respectivamente, mientras que las rotaciones sobre dichos ejes se denominan *pan, till* y *roll*, y a las variaciones en la distancia focal *zoom*. Normalmente, los vectores de movimiento no nos permiten diferenciar las rotaciones de las translaciones sobre los ejes horizontal y vertical, ni el zoom del desplazamiento a lo largo del eje óptico; de esta manera en la práctica el número de patrones estándar suele quedar reducido a tres.

Es posible distinguir dos maneras básicas de abordar el análisis del movimiento de cámara en el dominio comprimido. La primera consiste en realizar un estudio directo de los vectores de movimiento, mientras que la segunda estudia el movimiento a partir de un modelo de cámara ajustado previamente.

La primera técnica, aunque pueda resultar más intuitiva, está en desuso. Algunos de los trabajos publicados pueden consultarse en [12], [13], [14], [15].

Los estudios que utilizan la técnica de ajuste previo del modelo de cámara suelen ser más formales. El procedimiento básico es el de ajustar el flujo óptico estimado en cada cuadro a una función paramétrica, para después estudiar e interpretar estos parámetros.

Entre los modelos utilizados están los modelos geométrico, afín, bilineal, o proyectivo, pero existen modelos más complejos que tienen en cuenta deformaciones debidas a la perspectiva como el definido en [16]. Los de uso más frecuente son el modelo afín y el bilineal de 8 parámetros.

El modelo afín se basa en la estimación de 6 parámetros que describen el movimiento de cámara en la escena según:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_5 \\ a_6 \end{bmatrix} \quad (1)$$

, donde $(x, y)^T$ y $(x', y')^T$ son las posiciones antes y después de la transformación. El modelo bilineal de ocho parámetros añade dos más, así:

$$\begin{aligned} x' &= a_1 x + a_2 y + a_3 xy + a_7 \\ y' &= a_4 x + a_5 y + a_6 xy + a_8 \end{aligned} \quad (2)$$

Entre las técnicas de ajuste a un modelo podríamos diferenciar las que realizan un filtrado previo a un ajuste no robusto por mínimos cuadrados, y las que utilizan técnicas de ajuste robustas.

Cuando los datos están contaminados de forma no gaussiana, el ajuste no robusto por mínimos cuadrados se ve muy afectado por el ruido; así, [17] propone un algoritmo basado en el uso del estimador de Tukey, mucho más robusto frente a errores que el de mínimos cuadrados, para obtener un campo de vectores de flujo óptico a partir de los vectores de movimiento analizados. Otra de las técnicas que realizan un ajuste robusto es la propuesta por [18], en la que se usa un modelo bilineal de 8 parámetros como el descrito en la ecuación (2), para estimar, mediante un enfoque iterativo basado en el ajuste robusto por mínimos cuadrados, el movimiento de cámara en cada cuadro. Analizaremos ese algoritmo detalladamente en la sección 3.3.

Entre las que combinan un ajuste no robusto con un filtrado previo podemos destacar el algoritmo propuesto por [19], que realiza un descarte secuencial de MV filtrando primero por textura, a continuación usando un filtro de mediana, para después aplicar uno de suavizado (media). Finalmente se estima el modelo afín (ecuación (1)) de la cámara a partir de los vectores restantes. Otra técnica híbrida, como la sugerida en [7], realiza un filtrado previo en función de un parámetro denominado “relevancia de bloque”, calculado directamente en el dominio comprimido a partir de los coeficientes DC y AC además de con los vectores de movimiento. A partir de los vectores asociados a estos bloques “relevantes”, se estima un modelo proyectivo del movimiento de la cámara.

2.3.3 Segmentación basada en movimiento

Los objetos que aparecen en la mayoría de las secuencias de video suelen diferir bastante en cuanto a forma, color y textura, por lo que criterios basados en estos parámetros no son útiles para la segmentación de objetos en la mayoría de las situaciones. Así, una aplicación que busque la extracción de objetos independiente del video analizado utilizará preferiblemente el movimiento como información discriminante entre éstos y el fondo o *background*.

La segmentación de objetos basada en movimiento sobre secuencias de vídeo comprimidas suele utilizar la información de los vectores de movimiento de los cuadros P y B como una estimación del movimiento asociado a cada macrobloque. En el caso más sencillo, en el cual un objeto grande se mueve sobre un background en situación de cámara fija, se puede extraer el objeto de manera aproximada simplemente buscando vectores de movimiento no nulos en cada cuadro. En la mayoría de los casos, la situación dista mucho de ser tan sencilla.
Llegados a este punto es imprescindible realizar una distinción entre los algoritmos de segmentación de objetos en situación de cámara fija, y aquellos que se extienden a situaciones donde la cámara realiza algún tipo de movimiento.

2.3.3.1 Segmentación de objetos en situación de cámara fija

Existen varios algoritmos relacionados con la segmentación de objetos sobre videos grabados con cámara fija y, en general, devuelven resultados más que aceptables en entornos como el de video seguridad, pero carecen de aplicación práctica cuando la cámara se mueve. La estrategia más usada por estos algoritmos es el diseño de un modelo de fondo basado en la intensidad de los píxeles que no cambian significativamente entre los cuadros. Diferencias respecto a este modelo a lo largo del video indicarán la presencia de objetos.
Ejemplos de segmentación de objetos en cámara fija pueden encontrarse en [20] y [21], que usan un modelo de fondo basado en mezcla de gaussianas, que se actualiza con el tiempo para evitar la detección de objetos una vez que se han detenido. El éxito de este tipo de algoritmos radica en el buen modelado que realiza la mezcla de gaussianas de la intensidad de píxel, asumiendo una distribución normal no correlada en la adquisición de ruido, y cambios de luz lentos y graduales.

2.3.3.2 Segmentación de objetos en situaciones con movimiento de cámara

Existen muchas menos publicaciones sobre algoritmos que extienden la segmentación de objetos a situaciones con movimiento de cámara. Los trabajos que utilizan vectores de movimiento aprovechan que los objetos se suelen mover de forma uniforme y formando regiones de movimiento constante en el conjunto o campo de MV. En estos métodos de segmentación por MV, se asume por tanto que uno o más objetos en movimiento se pueden caracterizar como zonas de movimiento homogéneo.

La resolución en cuanto a cantidad de vectores de movimiento por unidad de superficie que se obtiene trabajando en el dominio comprimido es sólo de macrobloque y, aunque ésta puede ser suficiente para algunas aplicaciones, en el caso de que se requiera mayor resolución es necesario decodificar los macrobloques que se encuentran en los bordes de los objetos.

Citaremos algunas de las técnicas existentes en el estado del arte actual, exponiendo los motivos por los cuales nos hemos decantado por una de ellas, y no por las otras, como base para nuestro algoritmo.

En [22] el algoritmo propuesto se basa en la información de los vectores de movimiento de los cuadros P inmediatamente anteriores y posteriores a cada cuadro I, derivando a partir de ellos un conjunto de regiones de interés (ROI). Posteriormente estas ROIs se proyectan sobre los cuadros I y se segmentan por color utilizando la información de los coeficientes DC. No partiremos de este algoritmo por considerar que sólo es útil para segmentar

cuadros intracodificados (cuadros I), lo que en muchos casos supone disponer de la segmentación con una resolución temporal hasta 15 veces inferior a la de la secuencia.

En [23] se presenta un algoritmo por fases. Inicialmente realiza una sobre-segmentación apoyándose en características de color, para lo cual requiere la selección de *n* zonas homogéneas como marcadores. Después utiliza la información de movimiento de los cuadros P para combinar las regiones obtenidas previamente, agrupando todas aquellas conexas y con movimiento similar. No partiremos este método, ya que hay veces en las que, bajo la influencia de determinados movimientos de cámara, no puede encontrar similitudes entre las regiones.

En [24] se propone acumular los vectores de movimiento a lo largo de varios cuadros tanto anteriores como posteriores al cuadro bajo análisis (con lo que inevitablemente se introduce un retardo), para obtener un campo de vectores más denso. Sobre este campo se aplica un filtro de mediana y se interpola entre los vectores restantes, con el objetivo de eliminar los que no corresponden al movimiento real. El proceso de interpolación asigna un vector a cada píxel, sobre los que se aplica un filtro gaussiano. Posteriormente, se estudian las regiones que comparten modelos de movimiento, y se ajusta un modelo afín (ver ecuación (1)) a cada región, para, iterativamente, refinar esta parametrización y construir los objetos resultantes. Finalmente, la pertenencia de cada píxel a un objeto es analizada mediante la utilización de una ventana de búsqueda adaptativa. Existe la posibilidad de realizar un refinado de los bordes de los objetos (para obtenerlos con resolución de píxel), para lo que sería imprescindible decodificar los macrobloques situados en esos bordes. Los autores de este método aseguran que el proceso puede realizarse a medida que se recibe el video, pero la carga de operaciones involucrada en el procesado de la información y en el de interpolación, sugieren que su realización en tiempo real no es plausible.

Finalmente, [25] propone una solución en tiempo real y sin supervisión humana, en la que utiliza un algoritmo iterativo similar al propuesto por [18] para construir las máscaras de objetos iniciales de cada cuadro P e I, después exige consistencia temporal de los objetos mediante un seguimiento o *tracking* recursivo, para finalmente analizar su consistencia espacial y construir las máscaras de objetos resultantes. Además, segmenta objetos del *background* y refina las máscaras finales a resolución de píxel. Dada su simplicidad y eficiencia, a la par que unos resultados aceptables, se decidió que este fuera el algoritmo base sobre el que trabajar para mejorar sus resultados y extenderlos al mayor número de situaciones posibles. Se analizará detalladamente el sistema en la sección 3.

Conviene resaltar que los algoritmos presentados se han probado en la mayoría de los casos sobre secuencias concretas y poco representativas, debido en parte al problema de la inexistencia de un corpus o ground-truth de secuencias con que trabajar. Es por ello que muchas de las situaciones habituales no fueron contempladas en el diseño de estos algoritmos; de ahí que nuestro objetivo sea seleccionar el más adecuad y extenderlo a situaciones más genéricas.

3 Descripción del algoritmo base

3.1 Arquitectura del sistema

Este capítulo intentará describir el algoritmo base sobre el que diseñaremos e implementaremos las mejoras descritas en el capítulo 4 . El sistema descrito en el artículo donde se presenta este algoritmo [25] es más extenso, y no sólo se reduce a la segmentación de objetos en dominio comprimido y tiempo real. Sin embargo, consideramos que es ésta la parte clave de dicho trabajo, y que mejoras sobre esta parte resultarán con total seguridad en mejoras en los resultados finales del sistema.

Presentamos la arquitectura global del sistema, para después centrarnos en una descripción modular de su parte dedicada a la segmentación de los objetos o *foreground*, presuponiendo la existencia de un algoritmo previo de detección de tomas (*shots*), y de un algoritmo posterior de refinamiento de objetos a nivel de píxel, algoritmos ambos que no serán considerados dentro de este proyecto.

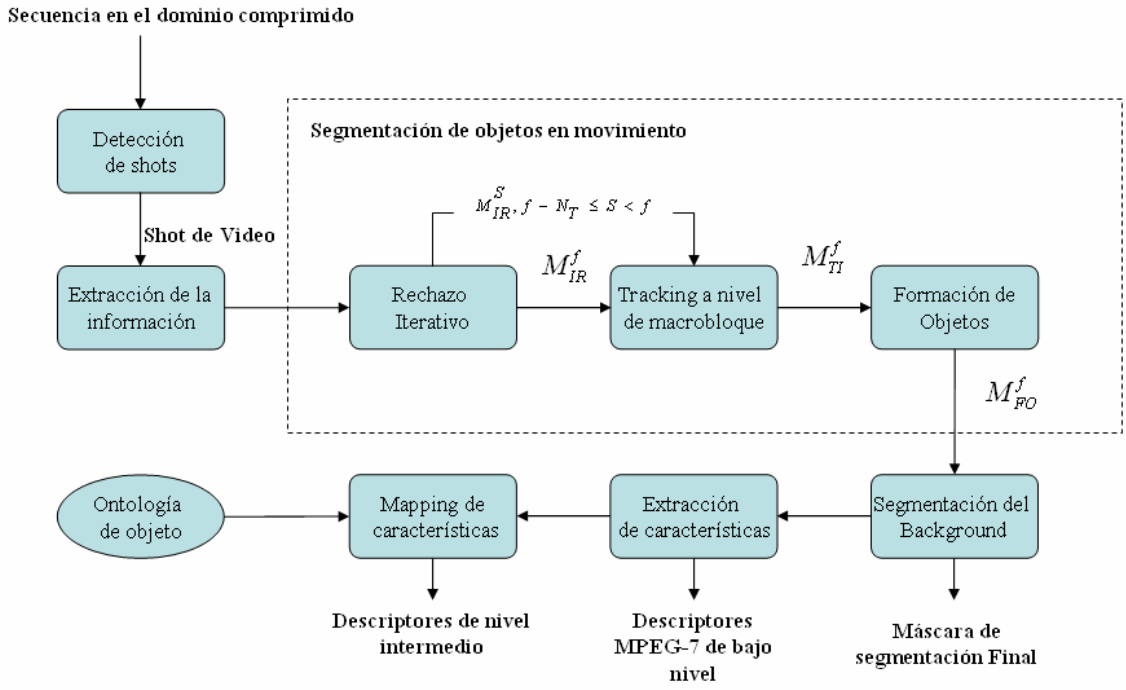


Figura 1: Arquitectura del algoritmo base

Es preciso destacar que el algoritmo de segmentación sólo usa los vectores de movimiento de los cuadros P, según indica su autor, por lo que para los cuadros I la información de movimiento se interpola entre la de los cuadros P anterior y posterior. Los cuadros B no son relevantes para este algoritmo, y la solución obtenida para cuadros P e I se extiende a los cuadros B. Por consiguiente en lo sucesivo y por razones de claridad, expresiones como "el cuadro anterior" o "el cuadro siguiente" se estarán refiriendo en realidad al cuadro anterior o sucesivo *analizado* (es decir, I ó P).

Nuestro trabajo se centrará en la mejora de la parte comprendida por las líneas discontinuas de la Figura 1. Describiremos cada uno de los módulos que la integran: módulo de rechazo iterativo, módulo de *tracking* a nivel de macrobloque y módulo de formación de objetos finales, así como también comentaremos brevemente el módulo previo para la extracción de información.

3.2 Módulo para la extracción de la información disponible en el dominio comprimido

La información del dominio comprimido puede obtenerse directamente de los ficheros MPEG tras una decodificación entrópica correspondiente a la primera etapa de un decodificador. Dado que no realizamos ni transformadas inversas ni las operaciones de compensación de movimiento necesarias para descomprimir propiamente una secuencia, esto supone un ahorro computacional importante.

La información relevante para la parte del algoritmo base a analizar son los vectores de movimiento, la tasa y el tipo de cuadro, el tipo de codificación de cada macrobloque y la resolución en macrobloques del cuadro. Nuestro algoritmo necesita además la información visual de los coeficientes DC de la DCT, como veremos en la sección 4.5.

Los coeficientes de la DCT, entre los que se incluye el valor DC, están disponibles únicamente en los cuadros I, pero también pueden estimarse en los cuadros P mediante aproximaciones con un buen compromiso entre coste y precisión.

3.3 Módulo de rechazo iterativo

3.3.1 Fundamento

El objetivo de este módulo es el de detectar, mediante el ajuste iterativo de los vectores de movimiento de cada cuadro P a un modelo paramétrico, aquellos macrobloques cuyos MV tengan un error, respecto a dicho modelo, mayor que la media de los errores computados para todos los macrobloques del cuadro. Dicho de otro modo, el objetivo es detectar los macrobloques que *no encajan* en el movimiento global de la escena. Los macrobloques así detectados serán *activados* como posible *foreground*, y pasarán a formar las máscaras iniciales que introduciremos en el siguiente módulo. Además, todo macrobloque intra-codificado en el cuadro se activará también en estas máscaras, con el fin de prevenir la aparición de huecos en los objetos. Como detallaremos más adelante, en la sección 4.3, esta solución no nos parece la más adecuada.

3.3.2 Descripción

En primer lugar se descartan los macrobloques en los bordes del cuadro, con el objetivo de prevenir la activación en las máscaras de macrobloques que, debido al descubrimiento de nuevo *background* a consecuencia del movimiento de cámara, o bien son intra-codificados o bien tienen asociados vectores de movimiento poco fiables. Pensamos que ésta es una solución necesaria, pero no suficiente, para prevenir la inclusión en la segmentación de macrobloques intra-codificados. Analizaremos este problema en la sección 4.3.1.

En cada iteración se compara con una máscara inicial, donde sólo los macrobloques intra-codificados y situados fuera del borde están activados. Sobre esta máscara se marcarán los macrobloques que hayan sido activados como posible *foreground* al final de cada iteración.

Para la estimación del modelo, se utilizan aquellos macrobloques que no hayan sido marcados como *foreground* en la iteración anterior. De esta manera, tanto la influencia de de vectores que no se correspondan con el movimiento real (*fake vectors*), como la influencia de los vectores de objetos ya activados, será nula en el cómputo de los parámetros y en el posterior cálculo del umbral de rechazo.

Con la información de movimiento de los citados macrobloques, se estima por mínimos cuadrados el movimiento predominante en la escena, usando para ello un modelo bilineal de 8 parámetros (ver ecuación (2)). Se activa como posible *foreground* todo macrobloque cuyo MV difiera del modelo un valor mayor que el umbral de rechazo; este umbral se calcula como el error medio respecto al modelo de todos los MV utilizados para su estimación.

Cabe destacar que este movimiento predominante puede ser el de la cámara, pero también el de un objeto en primer plano que ocupe la mayor parte de la imagen. El autor no solventó este problema, y nosotros lo plantearemos como trabajo futuro.

El proceso se repite durante un número determinado de iteraciones o hasta la convergencia del modelo, es decir, hasta llegar a una iteración en la que no se activen nuevos macrobloques. Según indicaciones del propio autor del algoritmo base, un valor de diez iteraciones sería una solución de compromiso entre el tiempo de ejecución y la capacidad de convergencia del mecanismo de ajuste.

Las máscaras procedentes del módulo de rechazo iterativo así obtenidas se denotarán M_{IR}.

3.4 Módulo de tracking a nivel de macrobloque

3.4.1 Fundamento

El objetivo de este módulo es el de comprobar y garantizar la consistencia temporal de las máscaras M_{IR}, y así evitar la segmentación final de objetos espúreos, frutos de la posible falsa activación de macrobloques durante la fase de rechazo iterativo. Estas activaciones espúreas pueden deberse a diversos factores: la presencia de *fake vectors*, la incapacidad del modelo para describir el movimiento de cámara, o la influencia de ruido sobre los vectores de movimiento.

Con este objetivo, se analizan las máscaras M_{IR} y sus correspondientes vectores de movimiento, los disponibles en cada cuadro P, y se va exigiendo a las regiones conexas activadas en las M_{IR}, continuidad entre cuadros consecutivos. La continuidad se comprueba verificando que existe solape entre las regiones *foreground* de un cuadro y la predicción o proyección temporal que se hace de ellas mediante *tracking* desde el cuadro anterior.

3.4.2 Descripción

Partiendo de las máscaras M_{IR} obtenidas en el módulo anterior, el objetivo es generar nuevas máscaras *mejoradas*, M_{TI}, formadas por regiones conexas (por simplicidad regiones) persistentes en el tiempo.

Dado un cuadro f, un macrobloque se activa en M_{TI}^{f} si y solo sí está activado en M_{IR}^{f}, y además solapa con los macrobloques *trackeados* de la máscara M_{TI}^{f}. Es decir, si su posición puede de alguna forma predecirse a partir de la máscara *mejorada* anterior. Para permitir la aparición de nuevas regiones en movimiento en cualquier cuadro además de en el primero de la secuencia este proceso se lleva a cabo en ventanas de tamaño $N_T = 4$ cuadros P/I, en las que la primera M_{TI} es inicializada con su M_{IR} correspondiente.

El *tracking* o proyección de movimiento de una máscara puede implementarse eficientemente utilizando los vectores de movimiento disponibles. Si denotamos con T el operador *tracking* utilizado (definido originalmente en [26] y mediante el cual se activan en la máscara *trackeada* M_{TI}^{f} todos los macrobloques que solapan, en al menos un 50 %, con los macrobloques proyectados de M_{TI}^{f}), el proceso recursivo que tiene lugar en cada ventana puede describirse como:

$$\begin{aligned} M_{TI}^{0} &= M_{IR}^{0} \\ M_{TI}^{f} &= M_{IR}^{f} \cap T(M_{TI}^{f-1}) \\ f &= 1...N_T \end{aligned} \qquad (3)$$

Con esta sencilla aproximación se produce sólo la segmentación de regiones que solapen en la mayoría de sus macrobloques, es decir, significativamente, con regiones segmentadas en cuadros anteriores. Nótese que en este esquema un error en la M_{IR}^{f} no se propaga indefinidamente, sino sólo en las N_T M_{TI} posteriores. Además, como se explica en [26], el *tracking* utilizado siempre resulta en máscaras mayores o iguales que la inicial, de ahí que los posibles errores introducidos den siempre lugar a áreas de objetos por exceso.

El esquema es eficiente y práctico, pero sólo resuelve el problema de las activaciones erróneas en las máscaras M_{TI}, no las faltas de activación. Trataremos este problema en la sección 4.4 de esta memoria.

3.5 Módulo de formación de objetos finales

3.5.1 Fundamento

El objetivo de éste módulo es el de garantizar la correcta separación de los objetos potenciales incluidos en las máscaras M_{TI}, así como el de relacionar estos objetos potenciales con los objetos segmentados anteriormente. Para ello, se asigna un identificador a cada región conexa existente en las M_{TI}, para después clasificarla bajo una

de las tres categorías predefinidas por el autor y, según sea esta categoría, gestionar posibles oclusiones entre objetos segmentados en cuadros anteriores.

En este módulo también se pueden incluir restricciones espacio-temporales para impedir la segmentación de objetos de tamaños o duraciones inferiores a unos valores previamente establecidos

3.5.2 Descripción

Los macrobloques activados en M_{TI} se agrupan en objetos potenciales utilizando conectividad 8. Aunque el autor sugiere conectividad 4, hemos observado que la conectividad 8 resulta mucho más versátil para segmentar formas complejas pero muy habituales como por ejemplo las de personas.

Los objetos así construidos se agrupan en las máscaras de objetos potenciales M_{OP}. Estos objetos no habrán de asociarse necesariamente con un único objeto, excepto para el caso particular del primer cuadro de un video ($f=0$), donde las máscaras de objetos finales, M_{OF} coincidirán con las potenciales, salvo por la aplicación de las restricciones temporales que veremos más adelante en esta sección.

Cada uno de los objetos potenciales ($\{S_k\}$, $k=1...N_{OP}^{f}$) de un cuadro f, (identificados como $M_{OP}^{f,k}$) puede asignarse a una de tres categorías: a un objeto preexistente, a varios objetos segmentados anteriormente, o un objeto de nueva aparición. Para ello se propone un sistema de clasificación basado en el solapamiento de macrobloques entre las máscaras $M_{OP}^{f,k}$ y las máscaras $T(M_{OF}^{f-1,k'})$, donde T es el operador de *tracking* definido en [26], que ahora se utiliza sobre las máscaras de objetos finales del cuadro anterior, y $k'=1...N_{OF}^{f-1}$ referencia cada uno de los objetos finales segmentados en el cuadro anterior.

Se definen así tres posibles categorías asignables a cada $M_{OP}^{f,k}$:

- Categoría 1

  Un *número significativo* de macrobloques pertenecientes a $M_{OP}^{f,k}$ solapa con un objeto *trackeado*: $S_m = T(M_{OF}^{f-1,m})$ y ninguno de los macrobloques de $M_{OP}^{f,k}$ solapa con otro objeto *trackeado* $T(M_{OF}^{f-1,k'})$, $k' \neq m$

- Categoría 2

  Un *número significativo* de macrobloques pertenecientes a $M_{OP}^{f,k}$ solapa con un objeto *trackeado* $S_m = T(M_{OF}^{f-1,m})$ y uno o varios de los macrobloques de $M_{OP}^{f,k}$ solapan con otro u otros objetos *trackeados* $\left\{T(M_{OF}^{f-1,r}), r \neq m\right\}$.

- Categoría 3

No existe ningún objeto *trackeado* $S_m = T(M_{OF}^{f-1,m})$ que solape un número significativo de macrobloques con $M_{OP}^{f,k}$, $k = 1...N_{OP}^{f}$.

En cuanto al término *número significativo* de macrobloques (*NSM*),se define para cada par de objetos $\left[T(M_{OF}^{f-1,k'}), M_{OP}^{f,k}\right]$ en función del tamaño en macrobloques del objeto final $S_{k'}$ y del potencial S_k:

$$NSM = \frac{S_k + S_{k'}}{4} \qquad (4)$$

Terminada la clasificación, la asignación de cada objeto $M_{OP}^{f,k}$ a uno o varios objetos finales depende de la categoría en la que haya sido clasificado. Así, los objetos $M_{OP}^{f,k}$ incluidos en la categoría 3 no pueden corresponderse con ningún objeto preexistente, por lo que dan lugar a nuevos objetos. De la misma manera, los objetos $M_{OP}^{f,k}$ pertenecientes a la categoría 1 pueden asignarse a un solo objeto preexistente, aunque cabe la posibilidad de que existan otros objetos $\left\{M_{OP}^{f,r}, r \neq k\right\}$ de la misma máscara asignables al mismo objeto *trackeado* S_m. En este último caso, sólo el objeto potencial de mayor tamaño de los $M_{OP}^{f,k}$ es asignado a este objeto S_m, es decir mantiene su identificador, mientras que el resto son asignadas al *background*.

Finalmente, los macrobloques de los objetos $M_{OP}^{f,k}$ clasificados en la categoría 2 pueden ser asignados a varios objetos previos. Para seleccionar el objeto al cual será asignado cada macrobloque del objeto potencial, se ajusta el movimiento de cada uno de los objetos candidatos a un modelo bilineal de 8 parámetros (ecuación (2)) y se evalúa el conjunto de modelos resultante para clasificar cada macrobloque de $M_{OP}^{f,k}$ por máxima verosimilitud: se asume que el modelo más probable, y por tanto el objeto al que se asigna el macrobloque, es aquel que arroja menor error entre el valor esperado o predicho por el modelo y el valor real del vector de movimiento asociado al macrobloque.

De esta manera se solventa el problema de las oclusiones entre objetos. Manteniendo el identificador de los objetos segmentados podemos diferenciar objetos adyacentes en el plano de la imagen.

Como elementos auxiliares a este módulo, se añaden dos sub-módulos cuya activación es dependiente de la aplicación. Tienen como objetivo restringir la segmentación de objetos en función de su tamaño o duración.

El módulo que restringe la segmentación de objetos atendiendo a criterios de tamaño ha de ser configurado con un número mínimo de macrobloques al comienzo del análisis, y su cometido es el de eliminar de las máscaras M_{OF} todo aquel objeto de dimensiones inferiores a este mínimo.

El modulo que restringe la segmentación de objetos de acuerdo a su duración temporal es algo más complejo, pues a partir de una cierta duración mínima expresada en segundos, el correspondiente número de cuadros debe derivarse en tiempo de ejecución teniendo en

cuenta la tasa de cuadros por segundo y la estructura del GOP. Este módulo obliga además a introducir un. retardo en la salida de resultados, puesto que el objeto no sólo ha de ser eliminado en la M_{OF} del cuadro en el que se detecte la violación de la restricción sino en las máscaras anteriores en las que hubiera sido segmentado.

4 Descripción de las mejoras introducidas

4.1 Motivación

Consideraremos el esquema o algoritmo descrito en el capítulo 3 como base para nuestros desarrollos. Según se ha comentado con anterioridad, esta decisión radica en su especial eficiencia y en la calidad de los resultados que devuelve. El objetivo es construir sobre esta base sólida un sistema con un rango de aplicación más amplio, como un pequeño paso más hacia un sistema de segmentación eficiente de aplicación global.

Nuestro objetivo principal no ha sido meramente identificar puntos flacos del algoritmo, sino sugerir nuevas ideas de cara a mejorarlo. Estas ideas han surgido de un estudio detallado del algoritmo base y de otras aproximaciones existentes en el área de segmentación basada en movimiento con las restricciones de operación en dominio comprimido y en tiempo real. Todas las mejoras descritas a continuación tienen su origen en problemas no identificados o, a nuestro entender, sólo parcialmente abordados por el autor del algoritmo base, en parte debido a la ausencia de un corpus representativo de la variedad de situaciones que presenta el problema de la segmentación.

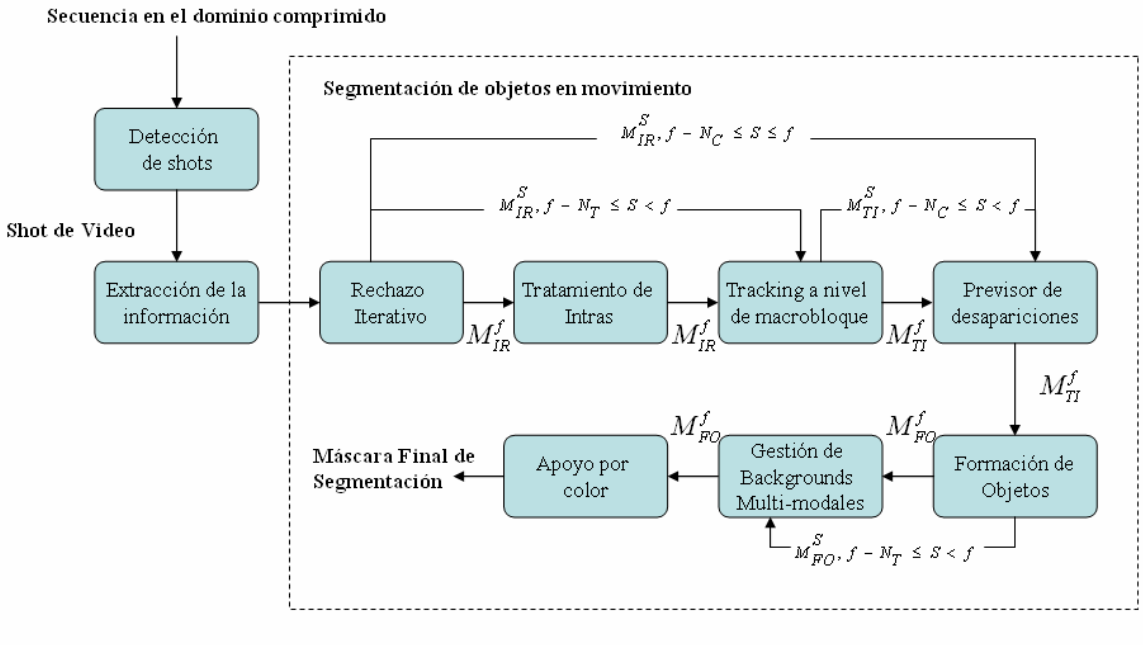


Figura 2: **Arquitectura del algoritmo propuesto**

4.2 Arquitectura del sistema

La introducción de nuevos módulos en el sistema base lleva implícita la necesidad de indicar y explicar su emplazamiento. Su situación influirá decisivamente en los resultados finales, pero también deberá garantizar que se conservan las principales ventajas del

algoritmo base: la obtención de resultados en tiempo real y sin supervisión humana. Así, el emplazamiento de los módulos ha de elegirse considerando su campo de acción y sus objetivos, pero sin incrementar en exceso la complejidad del sistema.

En base a estos criterios, el sistema propuesto es el ilustrado en la Figura 2. En las siguientes secciones detallaremos el fundamento y funcionamiento de cada uno de los módulos añadidos, siguiendo un esquema similar al utilizado en la descripción del algoritmo base.

4.3 Módulo para tratamiento de macrobloques intra-codificados

4.3.1 Fundamento

Como vimos en la sección 3.3.1, dedicada al módulo de rechazo iterativo, en el algoritmo base todo macrobloque intra-codificado es activado como posible *foreground* en las máscaras M_{IR}. Dado que no todos los macrobloques intra que aparecen en un cuadro se corresponden con objetos, esta decisión puede degradar de forma importante la calidad de la segmentación final.

Aunque no existe un criterio general para determinar de antemano los bloques que un codificador va a codificar en modo intra en un cuadro P, si pueden describirse un número de situaciones en la que esto es altamente probable:

- La entrada en el video de nuevos objetos o la aparición de cambios notables en los objetos existentes suele derivar en la codificación de alguno de sus macrobloques como intra. Efectivamente, el codificador no suele encontrar, con independencia de su ventana de búsqueda, regiones en cuadros inmediatamente anteriores en el video que sean similares a una región de nueva aparición. En segmentación, es crítico segmentar correctamente nuevos objetos que irrumpen en la imagen, como también lo es prevenir la aparición de huecos (es decir, la no segmentación) dentro de un objeto. El algoritmo base justifica su criterio activación sistemática de macrobloques intra en base a la relevancia de estas situaciones.

- Uno de los motivos más habituales para que se genere la intra-codificación de macrobloques es el *background* que se descubre por efecto del movimiento. Éste puede deberse al movimiento de cámara, o al movimiento de objetos. Siguiendo el esquema sugerido en la sección 3.3.1, esta situación puede inducir a errores como un crecimiento incorrecto de las M_{OF}, o incluso la segmentación de objetos inexistentes en el video.

- Por último, también suelen aparecer macrobloques intra-codificados debido al ruido introducido por los sensores y óptica de las cámaras, al ruido de cuantificación o a los cambios en la iluminación.

El objetivo de este módulo es el de prevenir la activación de macrobloques intra-codificados en cualquiera de los dos últimos casos, respetando su activación en el resto de situaciones.

4.3.2 Descripción

El esquema propuesto consiste en la aplicación de un filtro sobre las máscaras M_{IR} a la salida del módulo de rechazo iterativo. Este filtro ha de impedir la activación de macrobloques intra en situaciones no deseadas:

1. Ruido intra-codificado.
2. Descubrimiento de *background* debido al movimiento de cámara.
3. Descubrimiento de *background* debido al movimiento de objeto.

Pero a su vez ha de garantizar la activación, como posible *foreground*, de este tipo de macrobloques cuando:

4. Irrumpan nuevos objetos en la escena.
5. Los macrobloques pertenezcan a un objeto existente.

El caso 2 de esta lista fue considerado parcialmente por el algoritmo base; de ahí su decisión de no analizar los bordes o extremos de un cuadro. Sin embargo, esta consideración presupone un movimiento de cámara uniforme y moderado, lo que no siempre es cierto. Un *pan* (o cualquier movimiento relativo a los ejes *x* o *y*) largo y rápido, por ejemplo, descubrirá nuevo background en los extremos del cuadro, y éste podría ocupar más de una fila de macrobloques. Tal y como está diseñado, el posterior módulo de *tracking* no sería capaz de eliminar regiones así activadas debido a su persistencia temporal. Adicionalmente, no es solución desactivar macrobloques intra en anchuras desde los bordes de dos, tres e incluso cuatro filas o columnas de macrobloques, ya que estaríamos perjudicando a la segmentación de nuevos objetos que entrasen en la imagen (caso 4).

Observemos, además, que es complicado discriminar entre los casos 3 y 5, puesto que un objeto al moverse puede descubrir partes del background, pero también puede descubrir partes del mismo objeto que estaban ocultas. Claro ejemplo de esta situación serían los brazos de una persona moviéndose desde detrás de su espalda, o sus piernas saliendo de un agujero en el suelo.

Priorizando las necesidades de activación con el propósito de no perder nunca información relativa al objeto, marcaremos como posible *foreground* en M_{IR} un macrobloque intra-codificado en el cuadro f si y solo si cumple alguno de los siguientes requisitos:

- Pertenece a alguna de las máscaras de objetos *trackeadas* desde el cuadro anterior analizado: $T(M_{OF}^{f-1,k'})$, $k'=1...N_{OF}^{f-1}$.
- Es adyacente a alguna de las máscaras de objetos *trackeadas* desde el cuadro anterior analizado: $T(M_{OF}^{f-1,k'})$, $k'=1...N_{OF}^{f-1}$.
- No fue intra-codificado en el cuadro anterior analizado, $f-1$.

La condición de adyacencia debe garantizar la activación tanto de los intras adyacentes al objeto, como de los posibles intras adyacentes a los anteriores (recordar el ejemplo de los brazos). Para ello hemos analizado el solapamiento entre regiones conexas de

macrobloques intra-codificados y objetos *trackeados*. En caso de existir solapamiento, la región completa de macrobloques es considerada parte del *foreground*.

Mediante el cumplimiento de alguna de las dos primeras condiciones, activamos aquellos macrobloques intra-codificados que pertenecen a un objeto, mientras que con la tercera aseguramos la segmentación de objetos que entran en la escena.

Por otro lado, esta aproximación resolverá los problemas derivados de macrobloques intra-codificados por efecto del ruido o por el descubrimiento de *background* por movimiento de cámara, ya que estas regiones intra-codificadas no serán activadas en las M_{IR}.

Queda pendiente la no activación de aquellos macrobloques intra-codificados por descubrimiento de *background* debido al objeto (caso 3), situaciones éstas imposibles de diferenciar del descubrimiento de *foreground* (caso 5) usando exclusivamente criterios de movimiento. Por ello, trataremos de solucionar este problema en módulos posteriores.

Finalmente, es importante señalar que existe una situación de intra-codificación de macrobloques en cuadros P que ni el algoritmo base ni nuestra mejora pueden solventar. Supongamos una ventana de búsqueda muy pequeña en el proceso de codificación, y objetos muy rápidos durante todo el video. El codificador no conseguiría encontrar similitudes para los macrobloques del objeto en la ventana, y sistemáticamente los intra-codificaría a lo largo de todo el video. El algoritmo base, siguiendo su política de activación de todos los macrobloques intra, detectaría el objeto inicialmente, igual que el nuestro; pero, como la situación se repite constantemente, al no disponer de vectores para *trackearlo*, ambos algoritmos lo perderían ya que no sería activado a la salida del modulo de *tracking* a nivel de macrobloque. La consecuencia final es la pérdida o no segmentación del objeto, tanto para el algoritmo base como para nuestra mejora. Volveremos sobre este problema en el apartado de trabajo futuro.

4.4 Módulo para evitar la desaparición momentánea de objetos

4.4.1 Fundamento

Las máscaras M_{IR} procedentes del módulo de rechazo iterativo son muy propensas a errores de activación. El origen de estos errores puede agruparse en dos clases:

1. Macrobloques pertenecientes al *background*, incorrectamente activados.

2. Macrobloques pertenecientes a objetos en movimiento, incorrectamente no activados.

Los errores pertenecientes a la clase 1 pueden deberse a factores como la inexactitud de los vectores de movimiento, el ruido introducido por la cámara, los cambios de luz presentes en la escena o la generación excesiva de macrobloques intra-codificados por parte del codificador.

Una vez resuelta la sobre-activación de macrobloques intra-codificados (ver sección 4.3), a excepción de los casos en los que los objetos descubren el *background* que estudiaremos más adelante, el resto de errores de la primera clase son, asumiendo una reducida

persistencia temporal, gestionados eficazmente por el módulo de *tracking* descrito en la sección 3.4.

Se producirán errores de clase 2 cuando un objeto en movimiento o alguna de sus partes se detengan momentáneamente (bien de forma absoluta o relativa; a saber, cuando su velocidad sea significativamente más lenta que la de la cámara o la de otros objetos en el mismo cuadro). Un ejemplo de estas detenciones son las partes inmóviles de una persona mientras camina, es decir, alternativamente, cada uno de sus pies. Estudiaremos la influencia de objetos rápidos coexistentes en el cuadro en el apartado de trabajo futuro, por lo que no se considerarán en esta sección.

Aplicaciones como el seguimiento de objetos o la indexación basada en movimiento, requieren que un objeto no pierda su identificador si se queda estático momentáneamente. Así, detenciones puntuales de objetos o partes de ellos no deberían producir su desaparición o división en las máscaras finales de segmentación M_{OF}.

El algoritmo base no tiene en cuenta los errores de la clase 2, que además no sólo afectan al cuadro inicial en que el objeto queda sin segmentar. Efectivamente, debido a la presencia del módulo de *tracking* (sección 3.4), la desaparición (o, mas propiamente, la no segmentació) del objeto persiste en, al menos, los N_T cuadros siguientes.

El objetivo de este módulo es el de aprovechar la coherencia temporal de los objetos para evitar su desaparición de las máscaras finales de objetos M_{OF} por los motivos expuestos.

4.4.2 Descripción

El módulo de *tracking* requiere la activación de un macrobloque durante al menos N_T máscaras M_{IR} consecutivas para ser activado en la M_{TI}. El algoritmo propuesto en este módulo funciona de manera similar, requiriendo para la eliminación de un macrobloque activado en una M_{TI}^{f}, su desactivación durante N_C máscaras M_{IR} consecutivas.

De esta manera un macrobloque activado tras el módulo de *tracking* en el cuadro f, es decir, asignado al *foreground* en M_{TI}^{f}, permanecerá activado en las M_{TI}^{k}, $f < k < f + N_C$ siguientes independientemente de su estado en las M_{IR}^{k}, $f < k < f + N_C$. Asimismo, si el macrobloque permanece desactivado en las M_{IR}^{k}, $f < k \leq f + N_C$ será desactivado finalmente en la $M_{TI}^{N_C}$.

Este proceso se controla mediante una matriz contador, que almacena para cada macrobloque el número de cuadros analizados desde su última activación. Cada coordenada del contador representa el estado de un macrobloque.

Al comienzo del análisis, el contador será inicializado al valor N_C en todas sus posiciones. Cada activación de un macrobloque en la M_{TI} actualizará el valor en la posición correspondiente del contador a cero. Un macrobloque no activado en el cuadro f tras el módulo de *tracking* y con un valor $v < N_C$ asociado en la amtriz contador, provocará un

incremento de este valor a $v = v + 1$, y será activado a posteriori en la M_{TI}^{f} si y solo si el valor incrementado sigue siendo $v < N_C$.

El valor de N_C ha de ser cuidadosamente elegido, puesto que el esquema propuesto solventa los errores de clase 2 descritos en la sección anterior, pero puede crear un ensanchamiento o "estela" del objeto si el *background* descubierto se interpreta como *foreground* que permanece momentáneamente estático. Esta estela puede tener un tamaño máximo de $d \times (N_C - 1)$ macrobloques, siendo d la longitud máxima de los vectores de movimiento del vídeo, medidos en unidades de macrobloque. Este valor depende de la ventana de búsqueda en el proceso de codificación.

El efecto de esta estela sobre las máscaras finales de segmentación será solventado por un módulo posterior, pero para que sea abordable, N_C se ha estimado en la práctica a un valor de 2, con el cual sólo pueden crearse estelas estrechas de fácil y rápida desactivación posterior siguiendo el método propuesto en la sección 4.5.

En conclusión, el módulo presentado permite que objetos o partes de los mismos que se detienen de manera momentánea (bien de forma real o aparente), sean correctamente segmentados. Por el contrario en el algoritmo base, estos objeto desaparecen o son erróneamente fragmentados en al menos N_T máscaras M_{OF}.

4.5 Módulo de apoyo por color a la segmentación

4.5.1 Fundamento

El algoritmo base no hace uso de la información de color para la segmentación de objetos. Como vimos en la sección 2.2.2 del estado del arte, muchos de los algoritmos existentes en el área de segmentación de objetos hacen uso de ella para complementar los resultados obtenidos considerando únicamente los vectores de movimiento. Creemos por lo tanto, que la información de color disponible en el dominio comprimido puede ser muy útil para refinar las máscaras finales de objetos M_{OF}, y que puede suponer un apoyo a la segmentación por movimiento, sin incrementar excesivamente el tiempo de procesado.

Nuestra propuesta en este sentido consiste en aplicar el refinamiento por color sobre las máscaras M_{OF} en dos etapas, cada una de ellas relacionada con uno de los dos tipos de fuentes de error identificadas en la sección 4.4.1 para los que aún no se ha ofrecido una solución satisfactoria:

1. Análisis de macrobloques activados, pero sospechosos de pertenecer al *background*.

2. Análisis de macrobloques no activados como objetos pero susceptibles de serlo, situación relacionada con el interior de objetos en movimiento.

Mediante una primera etapa, ampliamos la capacidad del sistema para ajustarse a formas de objetos complejos, activando aquellos macrobloques que pertenezcan al objeto, pero a los que, por pertenecer parcialmente al *background*, se les ha asignado un vector de

movimiento similar a los del *background*. Esta medida está encaminada a resolver la primera fuente de error.

Por otra parte, una segunda etapa nos permite solventar dos problemas pendientes, relacionados con la segunda fuente de error: el descubrimiento de *background* por parte de los objetos (caso 3 de la sección 4.3.2), y la formación de estelas derivada de la estrategia del módulo para evitar desapariciones de macrobloques de *foreground* (sección 4.4.2).

En conclusión, la primera etapa activará macrobloques erróneamente asignados al *background*, mientras que la segunda desactivará macrobloques falsamente asociados a objetos o *foreground*.

4.5.2 Descripción

En este módulo se implementan las dos etapas de refinamiento resumidas en el apartado anterior; la técnica consiste en reclasificar cada macrobloque sospechoso de estar mal clasificado atendiendo a la información que proporcionan sus macrobloques vecinos. Definimos vecinos de un macrobloque bajo estudio a aquellos situados a una determinada distancia del examinado. A efectos de eficiencia, las posiciones de estos vecinos pueden estar precalculadas para cada macrobloque siguiendo el esquema sugerido en [27].

Para el diseño y la programación de este módulo es indispensable disponer en tiempo real de la información de pertenencia a *background*, *foreground* o estela de cada macrobloque. Esta información no necesita ser almacenada, puesto que ya está disponible en el contador descrito en la sección 4.4.2, por lo que este módulo no introduce mayor complejidad en este aspecto. Así, tanto para la elección de los macrobloques a reclasificar como para la elección del conjunto de vecinos a utilizar en cada reclasificación, utilizaremos esta información disponible, v, en el contador; a saber:

$$\begin{aligned} &v = 0 \rightarrow \textit{foreground} \\ &v = N_C \rightarrow \textit{background} \\ &0 < v < N_C \rightarrow \textit{estela} \end{aligned} \qquad (5)$$

La elección de qué macrobloques se va a reclasificar es distinta para cada etapa. En la primera, se aborda la posible reclasificación de aquellos macrobloques pertenecientes al *background* que estén situados a una distancia igual o menor a 2 de los macrobloques activados (ya sea como *foreground* o como estela). En cuanto a la segunda etapa, los macrobloques susceptibles de ser reclasificados son los activados como estela y los intra-codificados activados, de acuerdo con los problemas descritos en la sección de fundamento.

Una vez seleccionados los macrobloques susceptibles de ser reclasificados, ambas etapas operan sobre cada uno de ellos de una manera similar:

- Paso 1: Se determina el conjunto de macrobloques vecinos a uno dado que se pretende reclasificar. El criterio inicial para seleccionarlos es únicamente la distancia.

- Paso 2: Se particiona el conjunto anterior de vecinos del macrobloque dado en dos clases: la clase de macrobloques pertenecientes al *foreground* y la de los pertenecientes al *background*., de acuerdo con la información de pertenencia antes citada.

- Paso 3: Se analiza estadísticamente la similitud entre el macrobloque dado y las dos clases que lo rodean.

- Paso 4: Se asigna el macrobloque dado a la clase con la que mayor similitud presente.

El paso primero es ligeramente alterado en la primera etapa: se excluye del vecindario los macrobloques activados como estela y los intra-codificados activados, ya que este estado de activación no será definitivo hasta la finalización de la segunda etapa.

Una vez particionada la vecindad (paso 2), para el estudio de la similitud entre macrobloques (paso 3) definimos el *coeficiente de color* de un macrobloque como el vector de tres componentes resultante de la media de cada uno de los coeficientes DC (Y, Cb y Cr) asignados a cada uno de los cuatro bloques DCT $b = 1...4$ que forman un macrobloque (ver secciones 2.1.2 y 2.2.2). En conclusión, el coeficiente de color de un macrobloque mb será:

$$[Y_{mb}, Cb_{mb}, Cr_{mb}] = \frac{\sum_{b=1}^{4} [Y_b, Cb_b, Cr_b]}{4} \qquad (6)$$

Este coeficiente se calcula para todos los macrobloques involucrados en el proceso de reclasificación. Su valor será clave en la activación o desactivación de un macrobloque en las máscaras refinadas M_{OF}.

Para reasignar un macrobloque dado a una clase u otra debemos tener en cuenta su similitud respecto a ambas (paso 4). Para estimarla utilizamos el método de clasificación *K-Nearest-Neighbors*, que calcula los *K* vecinos más próximos de cada clase al macrobloque a reclasificar, y asigna este macrobloque a la clase que esté a menor distancia. En nuestro caso, el atributo utilizado para medir la proximidad a cada clase será el coeficiente de color definido en la ecuación (6). En cuanto al procedimiento utilizado para calcular las similitudes, se ha acudido a la distancia euclidea. Experimentalmente, se estableció un valor *K*=3.

Como resultado de este proceso obtenemos una clase para cada macrobloque a reclasificar (paso 5), y en el caso de pertenencia a uno o varios objetos, el identificador del objeto cromáticamente más similar.

La elección del orden de aplicación de estas etapas supone un riesgo inherente: si parte de la estela activada en la primera etapa es eliminada en la segunda, podrían existir macrobloques activados en la primera etapa que no fuesen adyacentes a ningún objeto. Consecuentemente, la adyacencia a los objetos de los macrobloques activados en la

primera etapa ha de ser comprobada finalizadas ambas. La razón por la que se ejecutan las etapas en este orden es la de poseer una información más valiosa y precisa de color para el *foreground* y el *background* en la segunda, mucho más crítica y decisiva que la primera.

4.6 Módulo para la gestión de backgrounds multi-modales

4.6.1 Fundamento

El objetivo de este módulo es el de extender la segmentación de objetos a entornos de *background* multi-modal. Queremos poder discriminar objetos en movimiento como personas, animales u objetos inanimados bajo la influencia de una fuerza física (objetos de *foreground*), de elementos en movimiento pero típicos del background, como el fuego, el agua agitada, la influencia del viento sobre plantas, o el humo.

El objetivo de este módulo no es sólo la eliminación de estos elementos típicos del background, sino también del *ruido* debido a la cámara, a la las variaciones de iluminación, al efecto sobre la estimación de movimiento de grandes áreas homogéneas o al propio proceso de codificación. Adicionalmente, con frecuencia se trata de ruido que persiste en el tiempo y que por tanto no puede ser eliminado por el módulo de *tracking* (sección 3.4). Para simplificar la redacción, denominaremos a los elementos en movimiento típicos del *background* y a este ruido como "objetos-ruido", asumiendo que la caracterización como tal de alguno de ellos es dependiente de la aplicación a la que destinemos el algoritmo (si el objetivo de nuestro análisis es detectar fuego, es imprescindible revisar el funcionamiento de este módulo).

La influencia en los resultados de la segmentación de estos objetos-ruido es crítica. Imaginemos un entorno de video seguridad en el cual se activase una alarma de actividad por el mero hecho del movimiento de un arbusto por efecto del viento, o el caso de un robot ignífugo que se adentrase en un incendio para rescatar humanos o animales atrapados en él y que no pudiese ser capaz de distinguir objetos en movimiento de los patrones de actividad del fuego.

Este módulo presenta una primera aproximación a la resolución del problema, que aspira a poder clasificar en tiempo real las máscaras finales de segmentación M_{OF} en objetos de *foreground* y objetos-ruido. La mejor manera de clasificar en tiempo real estos objetos es el uso de alguna técnica que permita extraer un modelo a partir de un conjunto de entrenamiento, pudiendo prescindir de ese conjunto en la etapa de segmentación. La precisión de este modelo dependerá de la representatividad de la muestra de cada clase con las que se entrene el clasificador.

Cualquier objeto que sea clasificado como objeto-ruido se eliminará de las máscaras M_{TI}, con lo que su influencia será progresivamente menor, y se segmentará sólo ocasionalmente.

Existe un punto clave a tener en cuenta: la extracción correcta de los objetos del *foreground* debe ser prioritaria sobre la eliminación de objetos-ruido. Siguiendo esta premisa, hemos desarrollado un modelo de clasificación que pasamos a describir a continuación. Asimismo, citamos otros métodos de clasificación considerados pero que,

debido a las restricciones temporales del PFC, no se ha llegado a documentar y probar con la suficiente profundidad, por lo que se han descartado temporalmente.

4.6.2 Descripción

El planteamiento inicial consistía en encontrar una manera de discriminar entre objetos-ruido y objetos válidos. Tras un análisis preliminar, observamos que en el conjunto de vectores asignados a los macrobloques de un objeto-ruido, prácticamente, cada vector tiene un módulo, dirección y sentido distintos, mientras que en el conjunto de vectores de movimiento de los macrobloques de objetos del foreground, aunque existen regiones con movimientos distintos, pueden identificarse patrones relativamente marcados. Esto nos indujo a estudiar en mayor detalle el movimiento de los objetos del *foreground* y de los objetos-ruido mediante su ajuste a modelos paramétricos, y a partir del análisis de la evolución temporal de estos modelos, a intentar encontrar características discriminantes entre ambos tipos de objetos.

La hipótesis de partida era que debido al movimiento caótico de los objetos-ruido, sus errores de ajuste a los modelos paramétricos serían mayores que los resultantes de ajustar objetos del *foreground*..Además, los valores de los parámetros del modelo asociado al movimiento de los objetos-ruido deberían variar más de un cuadro a otro..

En base a estos supuestos realizamos un estudio de la evolución de los modelos paramétricos a lo largo del tiempo, para lo cual consideramos la ventana temporal de duración N_T cuadros definida y utilizada en la sección 3.4. El objetivo es discriminar para cada objeto final del proceso, $M_{OF}^{f,m}$, su naturaleza: objeto válido u objeto-ruido.

Para cada objeto final $M_{OF}^{f,m}$ perteneciente a máscaras dentro de la ventana $f = 1...N_T$, calculamos los 8 parámetros del modelo bilineal (ver ecuación (2)), de modo que a cada uno se le asocia un conjunto de parámetros $a_i^{f,m}$ $(i = 1..8)$. A continuación obtenemos la mediana $e^{f,m}$ y la desviación típica $s^{f,m}$ del error de ajuste del modelo para cada vector de movimiento del objeto $M_{OF}^{f,m}$. Así definimos un vector de características (*features)* para cada objeto $M_{OF}^{f,m}$ y ventana temporal $f = 1...N_T$:

$$V^O = [\mu(e^{f,m}), \sigma(e^{f,m}), \mu(s^{f,m}), \sigma(s^{f,m}), \sigma(a_1^{f,m})...\sigma(a_8^{f,m})] \quad (7)$$

Estas *features* cumplen además el requisito de estar disponibles en tiempo real, sin añadir complejidad al algoritmo, para poder clasificar un objeto $M_{OF}^{f,m}$ antes de iniciar el análisis del siguiente cuadro.

Estudiamos varios métodos de clasificación: *Support Vector Machines* (SVM), redes neuronales (ANN) y *Lineal Discriminant Analysis* de Fisher (LDA). La difícil configuración tanto de la SVM como de las redes neuronales, en lo que respecta a la elección del *kernel* en el primer caso y a la elección del número de capas ocultas en el segundo, nos impidió alcanzar resultados robustos en el periodo de estudio dedicado, así que finalmente nos decantamos por el LDA-*Lineal Discriminant Analysis* de Fisher. El LDA proyecta nuestro espacio de dimensión 12 a un espacio unidimensional donde las dos clases que se pretende discriminar están más de-correladas entre sí y más correladas

internamente. Para ello realiza una transformación lineal de nuestro espacio de *features* v^o (ecuación (7)), devolviendo un espacio combinación de ellas.

Para entrenar el clasificador extrajimos un conjunto de muestras representativas para cada clase, y construimos el vector de *features* v^o (ecuación (7)) para cada muestra. Dichas muestras se extrajeron de una colección de noticiarios utilizados en el proyecto europeo MESH[28]. Entre los objetos-ruido presentes en las muestras incluimos: agua con diferentes tipos de movimiento, humo, fuego y efectos del viento en arbustos y árboles. Entre los objetos, incluimos personas, animales y objetos mecanizados. Analizamos un total de 53 objetos y 97 tipos de ruido en ventanas solapadas de N_T cuadros analizados (es decir, P o I), generando un total de 1370 muestras. Era importante que estas muestras no contuviesen objetos mezcla de objetos válidos y objetos-ruido, como por ejemplo, gente nadando. Creemos que este conjunto, aunque representativo, es claramente mejorable, y su mejora reportará un modelo más completo y fiable.

Una buena manera de evaluar nuestro modelo, dado el tamaño relativamente pequeño del conjunto de entrenamiento del que disponemos, es mediante validación cruzada, en nuestro caso *leave-one-out*, en la cual sucesivamente una de las muestras es extraída, y se entrena el modelo con el resto, después intentamos clasificar la extraída mediante ese modelo y observamos el éxito o el fracaso de esta clasificación. Analizando los porcentajes de acierto en la clasificación de objetos válidos y objetos-ruido para las distintas umbralizaciones posibles sobre el clasificador es posible llegar a la curva mostrada en la Figura 3:

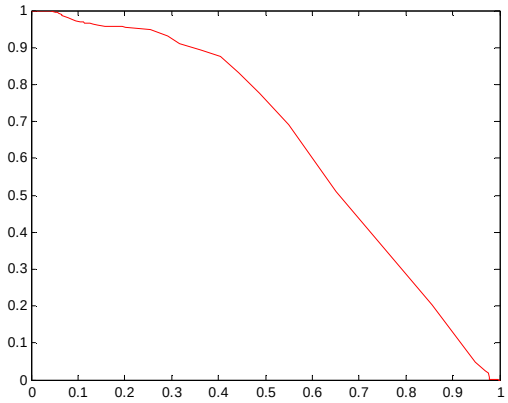


Figura 3: **Acierto en objetos frente a acierto en ruidos para las posibles umbralizaciones del LDA**

Detección de objeto	Detección de Ruido
95.10 %	24.48%
86.99 %	39.84 %

Tabla 1. **Combinaciones de acierto en ruido y acierto en objeto consideradas. (Extraídas de la curva de la Figura 3).**

A partir de esta curva es posible definir el umbral de separación entre ambas clases en función de lo tolerantes que queramos ser con los objetos-ruido, y de la trascendencia de la segmentación de objetos válidos para nuestro sistema (ver tabla 1).

Con el umbral precalculado, aplicamos el clasificador al vector de muestras para cada objeto detectado, y así conseguimos eliminar parte de los objetos-ruido presentes en *background* multi-modales.

5 Integración, pruebas y resultados

5.1 Integración

El sistema descrito en esta memoria está implementado como un prototipo, desarrollado en MatLab. Está previsto integrar una versión en C++ de esta implementación dentro de uno de los módulos de un sistema más amplio en el proyecto europeo MESH [28]. Concretamente, este módulo (IVOnLA) está destinado al análisis en tiempo real de noticiarios.

5.2 Pruebas

El objetivo de estas pruebas no es otro que el de realizar una comparativa tanto cuantitativa como cualitativa de los resultados obtenidos por el algoritmo base descrito en el capítulo 3, frente a los alcanzados tras la implementación e integración de las mejoras descritas en el capítulo 4.

Para la realización de las pruebas consideramos imprescindible disponer de un conjunto de secuencias representativas de los problemas que plantea la segmentación de objetos en movimiento, así como del *ground-truth* asociado. Ante la ausencia de un corpus de este tipo decidimos abordar la tarea de crearlo, basándonos en la generación de secuencias pseudo-artificiales mediante la técnica del *chroma-key*. Esto nos ha permitido obtener las máscaras de segmentación M_{GT} de gran calidad de forma semiautomática. En la grabación de estas secuencias intentamos considerar un buen número de situaciones típicas, tanto en el movimiento de los objetos como en los posibles movimientos de cámara, así como los diferentes tipos de *background* existentes. Un amplio resumen de la técnica, el montaje y la motivación del croma, así como de la justificación de los casos considerados en la grabación de estas secuencias puede consultarse en el Anexo A.A. Asimismo, los guiones y las situaciones clave que aparecen involucradas en cada una de estas secuencias se describen en el anexo A.B.

5.3 Resultados

No incluiremos los resultados para todas las secuencias que se podrían construir con los conjuntos de objetos y *backgrounds* de los que disponemos, ya que el tiempo de analizar todas las posibles combinaciones de los videos excedería los plazos inicialmente planificados para la realización de este proyecto.

Los resultados que a continuación se presentan intentan ofrecer una visión representativa del funcionamiento tanto del algoritmo base como del algoritmo propuesto. Para ello aplicamos ambos sobre cada una de las secuencias a analizar, obteniendo así para cada una dos conjuntos de máscaras de objetos finales, que comparamos con las M_{GT} o *ground-truth*. Esta comparación se ha hecho mediante parámetros cuantitativos como la precisión (*accuracy*) o el nivel de acierto (*recall*) de ambos algoritmos. Adicionalmente también presentamos una comparación cualitativa de las mejoras introducidas en las máscaras devueltas por cada algoritmo.

La precisión (P) y el nivel de acierto (R) vienen dados por las expresiones:

$$P = \frac{Tp}{Tp + Fp} \tag{8}$$

$$R = \frac{Tp}{Tp + Fn} \tag{9}$$

, donde Tp es el número de aciertos o verdaderos positivos, Tn el de verdaderos negativos, Fp el de falsos positivos y Fn el de falsos negativos.

Podríamos definir la precisión P, en nuestro caso, como la relación entre el número de macrobloques correctamente activados y el número total de aciertos y errores cometidos en la activación de los macrobloques. Los aciertos y errores se calcularán mediante comparación con las máscaras de *ground-truth*, M_{GT}. Análogamente, el acierto R cuantifica la relación que existe entre los macrobloques correctamente activados (comparando las M_{OF} con las M_{GT}) sobre el total de activados en las M_{OF}.

5.3.1 Secuencias de prueba

En esta sección se describen las características de las secuencias que hemos utilizado en nuestro proceso de evaluación. Ambas se han codificado con un GOP estándar IBBPBBP… de 13 cuadros[1].

En la primera secuencia (ver Figura 4) se incluye una situación de interacción de objetos rígidos y no rígidos en un escenario de complejidad media. Adicionalmente, se incluye un objeto de muy alta velocidad que, como se describió en la sección 4.3.2, tiene una alta probabilidad de ser intra-codificado. El *background* es unimodal y altamente texturado, lo cual garantiza una estimación más fiable del movimiento dominante a partir de los MV. El movimiento de cámara está formado por una sucesión de patrones *pan* y *tilt*.

Figura 4: Ejemplos de cuadros pertenecientes a la *Secuencia 1* para la realización de pruebas.

[1] El resto de los parámetros de codificación se corresponden a los establecidos por defecto en el codificador *ffmpeg* para un bitrate de 6Mbits/s.

La segunda secuencia (ver Figura 5) también incluye una situación de objetos rígidos y no rígidos que interactúan entre sí; sin embargo, la complejidad de la escena es en este caso mayor. Por un lado, si bien el *background* continúa siendo unimodal, el importante número de áreas cuasi-uniformes induce a una estimación del movimiento dominante menos precisa, ya que en estas zonas es probable que los MV no reflejen el movimiento de la escena. Por otro lado el número de patrones de movimiento de cámara también es mayor: se incluyen *pans*, *tilts* y *zooms*.

Figura 5: Ejemplos de cuadros pertenecientes a la *Secuencia 2* para la realización de pruebas.

La tercera secuencia (ver Figura 6) incluye un objeto rígido y uno no rígido que interaccionan entre sí. La cámara describe, como en la secuencia anterior, movimientos de *pan*, *tilt* y *zoom*. La complejidad de esta secuencia es, sin embargo, muy alta, debido a la presencia de un *background* multimodal formado por arbustos que se mueven con el viento. Se incluyen además regiones homogéneas como el cielo.

Figura 6: Ejemplos de cuadros pertenecientes a la *Secuencia 3* para la realización de pruebas.

5.3.2 Resultados cuantitativos

Esta sección presenta un análisis comparativo a nivel cuantitativo sobre las dos primeras secuencias descritas en la sección anterior. En esta comparativa no se han tenido en cuenta los *background* multimodales, que aparecen en la tercera secuencia de pruebas. Para ello se ha desactivado el módulo que los gestiona. El motivo es poder realizar una comparación más justa con el algoritmo base, que no considera la eliminación de elementos en movimiento del *background*.

La Tabla 2 muestra los resultados obtenidos para la aplicación de cada uno de los algoritmos sobre la primera secuencia, analizando los parámetros cuantitativos descritos en

la sección 5.2, P y R, así como su producto $R \times P$, que ofrece un indicador global del funcionamiento de cada algoritmo. Asimismo, también se incluyen las fracciones de acierto Tp y falsos positivos Fp sobre el total de macrobloques activados (i.e. marcados como objetos o *foreground*). La Tabla 3 muestra estos mismos resultados para la segunda secuencia presentada, de una mayor complejidad..

Secuencia 1 (45 cuadros)	Fracción de Tp (respecto del total de activados)	Fracción de Fp (respecto del total de activados)	P	R	$R \times P$
Algoritmo base	0.577171	0.696123	0.303877	0.577171	0.175389
Algoritmo propuesto	0.535002	0.311272	0.688728	0.535002	0.368471

Tabla 2. Resultados cuantitativos de cada uno de los algoritmos (base y propuesto) para la *Secuencia 1*

Secuencia 2 (cuadros)	Fracción de Tp (respecto del total de activados)	Fracción de Fp (respecto del total de activados)	P	R	$R \times P$
Algoritmo base	0.659965	0.465171	0.534829	0.659965	0.352968
Algoritmo propuesto	0.686189	0.471301	0.528699	0.686189	0.362787

Tabla 3. Resultados cuantitativos de cada uno de los algoritmos (base y propuesto) para la *Secuencia 2*

Comparando los resultados obtenidos para la primera secuencia y recogidos en la Tabla 2, observamos que la calidad de los resultados del algoritmo propuesto para esta secuencia es superior, en términos generales, a la de los obtenidos con el algoritmo base, pues aunque puede observarse un ligero descenso en el nivel de acierto, la precisión aumenta significativamente (por consiguiente mejorando el producto $R \times P$).
Por otro lado, la segunda secuencia se ha incluido en estos resultados más bien como muestra del trabajo pendiente de realizar. En ella pueden observarse algunas situaciones en las que el algoritmo propuesto funciona peor que el algoritmo base como resultado de la presencia de zonas homogéneas en el *background*. En efecto, al ser estas zonas homogéneas propensas a marcarse como objetos, son habitualmente extendidas por el modulo propuesto de refinamiento por color. En consecuencia, los resultados en cuanto a precisión son peores, pues existen muchos más macrobloques falsamente activados en las máscaras finales del algoritmo propuesto. Sin embargo el funcionamiento en general se mantiene ligeramente superior para el algoritmo propuesto.

5.3.3 Resultados cualitativos

En esta sección mostramos gráficamente algunos de los resultados obtenidos con el algoritmo propuesto frente a los resultantes del algoritmo base, tanto para las secuencias

con backgrounds *unimodales* discutidas en la sección anterior, como para la tercera secuencia con background *multimodal*.

Secuencias con backgrounds unimodales

Para mostrar el funcionamiento obtenido en la primera secuencia descrita en la sección 5.3.1 hemos seleccionado a modo de ejemplo tres cuadros (ver Figura 7) en los que los resultados de la segmentación obtenidos con el algoritmo base no son especialmente buenos, identificando en cada caso las causas que podrían estar implicadas en este problema. Presentamos, adicionalmente tanto las máscaras obtenidas por el algoritmo base como las máscaras obtenidas con el algoritmo propuesto, dando una breve descripción de los módulos que han contribuido significativamente a mejorar los resultados originales.

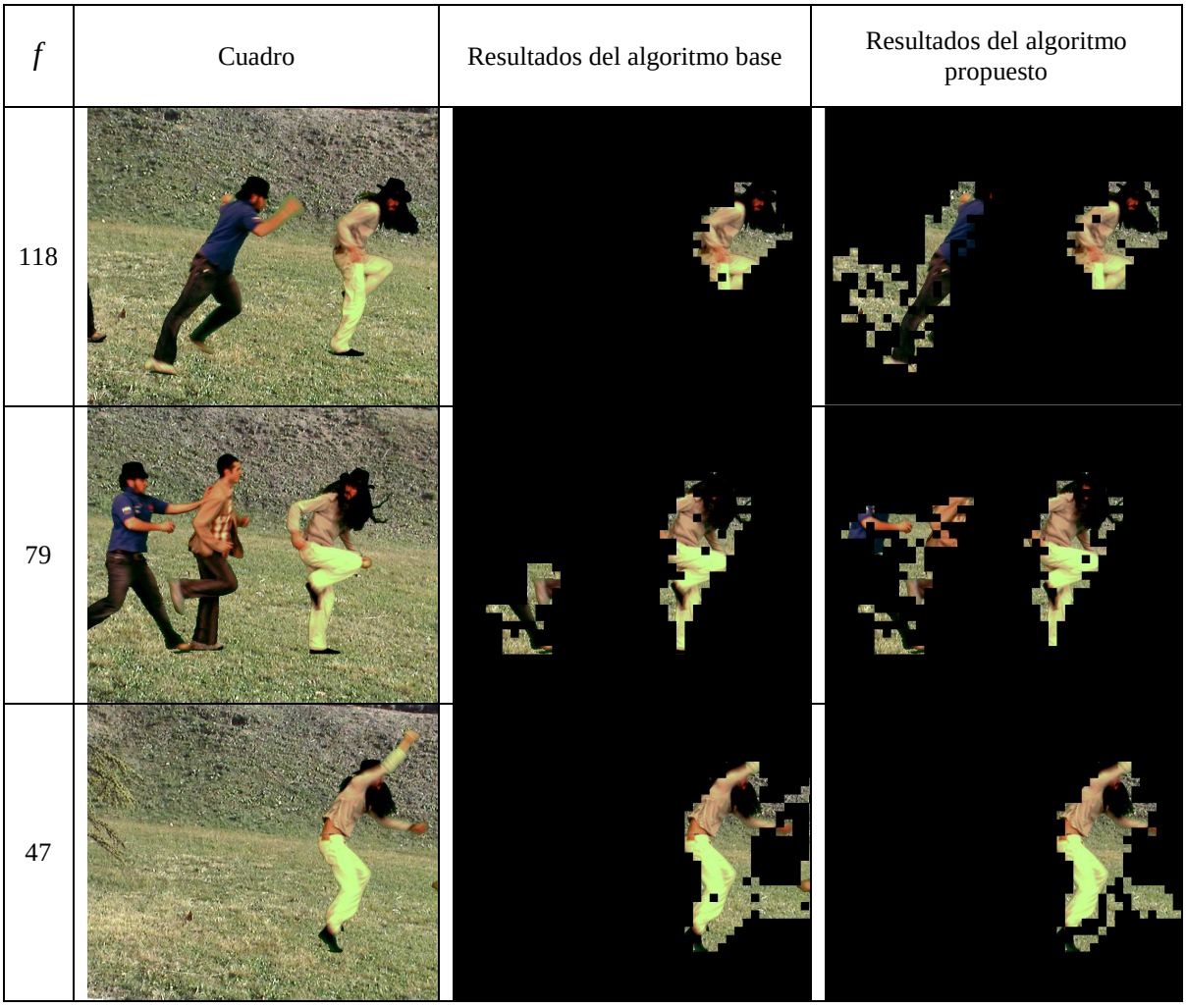


Figura 7: Ejemplos de los resultados cualitativos obtenidos para la *Secuencia 1*

En el primer cuadro seleccionado (cuadro 118 de la secuencia), la segmentación se ha visto degradada por la detención aparente del segundo corredor que aparece en la escena (debido a que su velocidad coincide, por un instante, con la de la cámara). Aunque el objeto

desaparece de las máscaras de segmentación del algoritmo base, el modulo previsor de desapariciones del algoritmo propuesto (ver sección 4.4), infiere que esta detección puede ser meramente circunstancial y no lo elimina de las máscaras correspondientes.

En el segundo cuadro seleccionado (cuadro 79 de la secuencia) el segundo corredor y gran parte del tercero no se segmentan en las máscaras del algoritmo base debido a que están formados por macrobloques intra-codificados. Dado, sin embargo, que existe una parte del tercer corredor que sí parece incluirse en las máscaras de objeto, el nuevo modulo de gestión de intra-codificados (sección 4.3) y el refinamiento por color (que activa los macrobloques cromáticamente similares, como se describió en sección 4.5) mejoran, como puede apreciarse, los resultados finales de manera substancial.

En el tercer cuadro de la tabla, (cuadro 147 de la secuencia), puede observarse una máscara de segmentación notablemente imprecisa. El módulo de refinamiento por color es el responsable de la ligera mejora en los resultados del algoritmo propuesto.

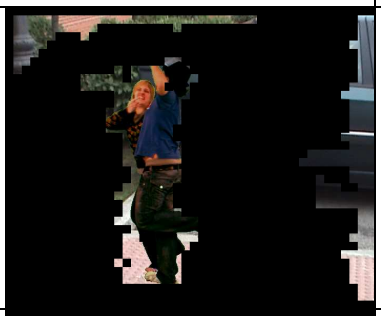
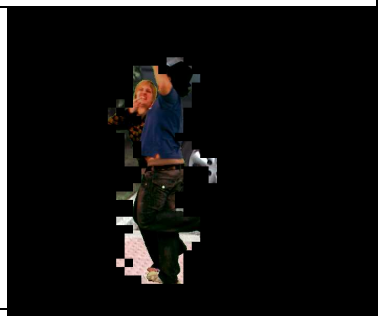
f	Cuadro	Resultados del algoritmo base	Resultados del algoritmo propuesto
218			
227			
103			

Figura 8: Ejemplos de los resultados cualitativos obtenidos para la *Secuencia 2*

Los resultados obtenidos para la segunda secuencia, calificada como de alta complejidad, son menos concluyentes (ver Figura 8). Durante una buena parte de esta secuencia el

algoritmo propuesto arroja mejores resultados que los obtenidos por el algoritmo base. Sin embargo, la aparición de áreas homogéneas incide directamente en la calidad de los vectores de movimiento y a partir de cierto momento, los resultados finales se degradan de forma considerable. En la citada Figura mostramos tres cuadros representativos de esta secuencia, junto a las máscaras obtenidas por el algoritmo base y aquellas resultantes de nuestra aproximación.

En el primer cuadro seleccionado (cuadro 218 de la secuencia), la cámara realiza un *zoom out* relativamente rápido, dando lugar a un buen número de macrobloques intra-codificados en los bordes del cuadro. Como puede observarse, la mejora propuesta en la gestión de macrobloques intra-codificados (ver sección 4.3), elimina los objetos inexistentes que se forman en las máscaras del algoritmo base.

En el segundo cuadro de la tabla (cuadro 227 de la secuencia), el personaje de la derecha, que está situado aproximadamente en el foco de un *zoom-out*, eventualmente se mantiene estático respecto a la cámara, por lo que desaparece en las máscaras finales en el algoritmo base. Como puede apreciarse, nuestro módulo previsor de desapariciones (ver sección 4.4) evita que este objeto se elimine de las máscaras. Asimismo, como ocurría en el cuadro anterior, la calidad de los resultados en los bordes del cuadro también se mejora.

Finalmente, el último ejemplo (cuadro 103 de la secuencia) constituye una muestra del peor funcionamiento ocasional de nuestro algoritmo. En particular el modulo de gestión de color extiende incorrectamente la máscaras correspondientes al personaje del *foreground* y a un área del *background* que no se corresponde con ningún objeto.

Secuencia con background multimodal

Se incluyen a modo de ejemplo (ver Figura 9) resultados para una secuencia con un *background* multimodal en la que podemos observar como el algoritmo base detecta los objetos-ruido y cómo estos se eliminan de las máscaras finales de objetos del algoritmo propuesto.

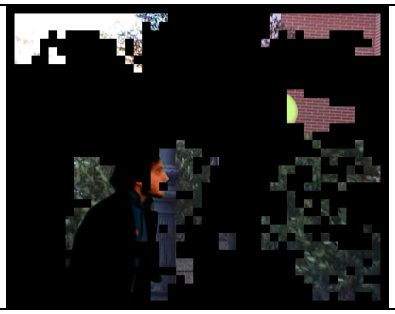
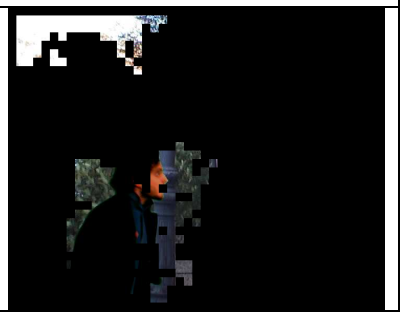
Original	Resultados del algoritmo base	Resultados del algoritmo propuesto
		

Figura 9: Ejemplos de los resultados cualitativos obtenidos para la *Secuencia 3*

Debe tenerse en cuenta que el funcionamiento de éste módulo es muy dependiente de la situación. Recordemos que sólo podemos eliminar parte del ruido, alrededor de un 30% (ver sección 4.3.2). Claro ejemplo de este funcionamiento es el cuadro incluido, en el que el módulo es capaz de eliminar tanto los arbustos en movimiento por el viento, como zonas

homogéneas de la pared, pero no es capaz de gestionar correctamente la homogeneidad del cielo.

5.4 Resultados en proceso de publicación

Tras valorar la calidad de los resultados obtenidos, y considerando el interés de la comunidad científica por lo innovador de alguna de las aproximaciones propuestas, se decidió redactar un resumen del trabajo realizado en un artículo de investigación para su envío al *International Conference on Imagen Processing, ICIP 2008*., la conferencia más prestigiosa en el ámbito en que se desarrolla este trabajo. Dado que la fecha límite de envío fue enero de 2008, al cierre de estas líneas no se tiene información sobre la eventual aceptación del trabajo propuesto.

Asimismo, tras constatar el interés de varios grupos de investigación europeos por el proceso de generación del *ground-truth* (ver Apéndice A.A), se ha redactado un segundo artículo que describe el análisis previo, el diseño y la realización de este corpus, así como la regulación de su cesión a terceros para fines de investigación. Este artículo, del cual se incluye una transcripción parcial en el citado anexo, ha sido aceptado para su publicación en la citada conferencia.

6 Conclusiones y trabajo futuro

6.1 Conclusiones

El trabajo que presenta esta memoria presenta un conjunto de mejoras introducidas sobre un algoritmo base de segmentación de objetos en movimiento. La principal conclusión es la consecución del objetivo de diseñar un sistema de aplicación más amplia que el sistema base; para ello, hemos solventado parcialmente varios de los problemas detectados, y hemos propuesto soluciones a nuevas situaciones no contempladas en el algoritmo inicial.

Los resultados obtenidos apoyan el desarrollo de una línea de investigación en este campo. Quedan aún muchos problemas por resolver para la construcción de un sistema de aplicación global, principalmente la eliminación de la segmentación final de todos los objetos-ruido descritos en la sección 4.6.1, así como algunas otras situaciones que describiremos en la sección sobre trabajo futuro.

En conclusión, consideramos que los objetivos del proyecto han sido cumplidos en su mayoría, a pesar de ser bastante ambiciosos:

i. Estudio del estado del arte actual: el estudio ha sido exhaustivo, y ha intentado cubrir los principales algoritmos existentes en el área de segmentación de objetos basada en la información de movimiento.
ii. Implementación del algoritmo base: en vista a los resultados incluidos por el autor en su descripción del algoritmo, y a los obtenidos por nuestra implementación para las mismas secuencias, el algoritmo ha sido implementado correctamente. En este sentido es necesario agradecer al propio autor la colaboración prestada para comprender algunos detalles de implementación.
iii. Estudio y gestión de los macrobloques intra-codificados: la técnica propuesta para la gestión de los macrobloques intra-codificados ha demostrado su éxito y funcionalidad en los resultados incluidos en esta memoria (ver sección 5.3).
iv. Estudio y gestión de la coherencia temporal de objetos en movimiento: el módulo que evita las desapariciones realiza correctamente su función (ver sección 5.3).
v. Estudio y gestión de la aplicación de información cromática para el refinamiento de las máscaras de objetos: este es quizás el módulo que más dudas genera; es cierto que refina levemente las máscaras de segmentación de objetos, pero por otro lado, aumenta las máscaras correspondientes a objetos-ruido. Abordaremos esta cuestión en el apartado de trabajo futuro.
vi. Estudio y gestión de la aplicación del algoritmo a entornos de *background* multi-modal. Se han estudiado los diferentes métodos de clasificación, y se ha propuesto tan sólo una primera aproximación a la gestión de este tipo de *background*. Es, por tanto, el módulo que más trabajo futuro requiere, dado que además su correcta implementación influye decisivamente en el funcionamiento de los otros módulos y, por ende, en el funcionamiento global del sistema.

6.2 Trabajo futuro

Esta sección aborda, siguiendo el orden de aparición en el documento, los aspectos donde se considera interesante realizar algún tipo de trabajo futuro. Las propuestas que siguen no son más que ideas con las que empezar a trabajar y a fecha de hoy están exclusivamente basadas en observaciones teóricas fruto de la experiencia adquirida durante el desarrollo del proyecto:

- La primera cuestión no resuelta era la incapacidad de ambos sistemas para diferenciar un movimiento predominante en la escena, producido por el movimiento de la cámara, de uno que describa el movimiento de un objeto grande en primer plano (sección 3.3). El funcionamiento de todo el sistema será erróneo si consideramos videos en los que un objeto ocupa la mayor parte del cuadro. Como posible solución plantemos el uso de algún tipo de estudio previo por color, como se propone en [7].

- El módulo de refinamiento por color ha de reconsiderarse. Es necesario trabajar sobre tareas como la utilización de métodos de clasificación más robustos que el *K-Nearest-Neighbors*, el estudio de un nuevo coeficiente de color para realizar la clasificación (ver sección 4.5), y acaso sobre la conveniencia de un nuevo emplazamiento del módulo (podría situarse antes del módulo de rechazo iterativo, como hemos conjeturado en el punto anterior)

- También nos percatamos en la sección 4.3.2 de la incapacidad de ambos sistemas para segmentar objetos que se mueven demasiado rápido y se codifican con ventanas de búsqueda pequeñas, lo que resulta en su intra-codificación a lo largo de todo el video y su desaparición en las máscaras M_{OF}. Una posible aproximación a la solución de este problema es la asignación a los intra que se activen tras el módulo 4.3, de vectores de movimiento similares a los que los rodean, interpolando ponderadamente estos según similitudes entre los diferentes macrobloques no intra-codificados adyacentes al intra. Esta idea necesita un estudio previo más riguroso, puesto que su implementación podría no ser tan trivial.

- Uno de los puntos claves de este trabajo futuro será la gestión de situaciones en las que coexistan objetos muy rápidos con objetos muy lentos. En estas situaciones la influencia del objeto rápido en el cálculo del umbral adaptativo (sección 3.3.2) podría resultar en la pérdida del objeto lento, puesto que su movimiento se diferenciaría del de la cámara mucho menos que el del objeto rápido. Tras un análisis preliminar es posible aventurar dos posibles maneras de abordar este problema:

    i. Estudiando la sustitución del algoritmo usado en el módulo de rechazo iterativo (sección 3.3), por otro algoritmo del estado del arte o propio, y observar si su funcionamiento solventa este problema.
    ii. Intentando mejorar el modelo de clasificación descrito en la sección 4.6.2. Si pudiésemos discriminar siempre entre objeto-ruido y objeto válido, podríamos

permitir la activación de todo movimiento, sabiendo que posteriormente eliminaríamos el ruido del resultado final.

- Para solventar el problema descrito en el punto anterior y a su vez mejorar la clasificación de los objetos-ruido como tal, se propone:

    i. Ampliar el número y la variedad de las muestras del conjunto de entrenamiento, incluyendo casos no considerados como la oclusión entre objetos.
    ii. Utilizar otros métodos de clasificación, en teoría más complejos, como las redes neuronales, aunque éstos no tienen necesariamente que ofrecer mejores resultados que los obtenidos.

- Finalmente consideramos que debemos extender el análisis de pruebas considerado a la variación de resultados según los diferentes parámetros utilizados en el proceso de codificación, como la ventana de búsqueda ó el GOP.

Referencias

[1] J.L. Michell, W.B. Pennebaker [et al.], "MPEG video Compression Standard", Digital Multimedia Standards Series, Kluwer Academic Publishers

[2] Hans Georg Musmann, Peter Pirsch, and Hans-Joachim Grallert. Advances in Picture Coding. Procc. IEEE, 73(4): 523-48, Apr 1985.

[3] Chen,J.-Y., Taskiran, C., Delp, E.J., Bouman, C.A., 1998. ViBE: a new paradigm for video database browsing and search. In: Proc. IEEE Workshop on Content-Based Access of Image and Video Databases.

[4] Kobla, V., Doermann, D., Faloutsos, C., 1997. VideoTrails: representing and visualizing structure in video sequences. In: Proc. ACM Multimedia 97, pp. 335–346

[5] Yeo, B.-L. and Liu, B., 1995. Rapid scene analysis on compressed videos. *IEEE Trans. Circuits Systems Video Technol.fs* 5 6, pp. 533–544.

[6] Patel, N.V. and Sethi, I.K., 1997. Video shot detection and characterization for video databases. *Patter Recognition* 30 4, pp. 583–592

[7] P. Kuhn, "Camera motion estimation using feature points in MPEG compressed domain". In Proceedings 2000 International Conference on Image Processing, 2000, vol. 3, pp. 596-9.

[8] Dimitrova, N., Golshani, F., 1995. Motion recovery for video content analysis. ACM Trans. Information Systems vol. 13, No. 4, October, pp. 408-439

[9] Divakaran, A., Sun, H., 2000. A descriptor for spatial distribution of motion activity for compressed video. In: Proc. SPIE Conf. on Storage and Retrieval for Media Databases 2000, San Jose, CA, January, pp. 392–398.

[10] Wolf, W., 1996. Key frame selection by motion analysis. In: Proc. ICASSP 96, vol. II, pp. 1228–1231

[11] ISO/IEC 15938-3: 2002 Information technology. Multimedia content description interface- MPEG-7 Part 3: Visual

[12] Sangkeun Lee; Hayes, .M. "Real-time camera motion classification for content-based Indexing and retrieval using templates", In 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings vol.4, pp.IV3664-7.

[13] Kobla, V.; Doermann, D.; Ning-Ip Lin, "Archiving, indexing, and retrieval of video in the compressed domain". In Proceedings of the SPIE – The International Society for Optical Engineering, 1996, vol.2916, pp. 78-89.

[14] Wei Xiong; Lee, J.C.-M., "Efficient scene change detection and camera motion annotation for video classification" In Computer Vision and Image Understanding Aug. 1998, vol 71, no 2, pp.166-181

[15] Xinqquan Zhu; Elmagarmid, A.K.; Xiangyang Xue; Lide Wu; Catlin, A.C., "Insight Video; toward hierarchical video content organization for efficient browsing, summarization and retrieval"In IEEE Transactions on Multimedia, Aug. 2005, vol 7, no 4, pp 648-666.

[16] S. Mann, R. W. Picard, "Video orbits of the projective group: A simple approach to featureless estimation of parameters", IEEE Trans. on Image Processing., Sept. 1997

[17] M. Durik and J. Benois-Pineau, "Robust motion characterisation for video indexing based on mpeg2 optical flow". In Proc. CBMI'2001, pages 57-64, September 2001

[18] T. Yu, Y. Zhang, “Retrieval of Video Clips using global motion information”, Electronic Letters, July 2001, vol. 37, n. 14
[19] M. Pilu, “On Using Raw MPEG Motion Vectors To Determine Global Camera Motion”, Internal Report, Hewlet-Packard, 1997
[20] L. Carminati, J. Benois-Pineau, “Gaussian mixture classification for moving object detection in video surveillance environment”, Proc. ICIP’2005, vol. 3, pp 113-16, Sep. 2005.
[21] C. Stauffer, W.E.L. Grimson, "Adaptive background mixture models for real-time tracking", Proc. CVPR’1999, Jun. 1999.
[22] H. Eng, K. Ma, “Spatio-temporal Segmentation of Moving Video Objects over MPEG Compressed Domain”, IEEE 2000
[23] M. L. Jamrozik, M. H. Hayes, “A Compressed Domain Video Object Segmentation System”, IEEE ICIP, 2002
[24] R. Venkatesh, K. Ramakrishnan, S. Srinivasan , “Video Object Segmentation: A Compressed Domain Approach”, IEEE Trans. Circuits System Video Technology., April 2004
[25] V. Mezaris, I. Kompatsiaris, N. Boulgouris, M. Strintzis, “Real-Time Compressed-Domain Spatiotemporal Segmentation and Ontologies for Video Indexing and Retrieval”, IEEE Trans. Circuits System Video Technology, May 2004
[26] L. Favalli, A. Mecocci, and F. Moschetti, “Object tracking for retrieval applications in MPEG-2,” *IEEE Trans. Circuits System Video Technology*. vol. 10, pp. 427–432, Apr. 2000.
[27] Maria Petrou, Pedro García Sevilla. “Dealing with Texture”,Ed Wiley 2006 pp 277-280.
[28] Multimedia semantic syndication for enhanced new services. www.mesh-ip.eu

Glosario

MV	Motion Vectors
MPEG	Moving Pictures Expert Group
DCT	Discrete Cosine Transform
AC	Alternative Current
DC	Direct Current
LS	Least Square
GOP	Group of Pictures
SVM	Super Vector Machine
LDA	Linear Discriminant Analysis
ICIP	International Conference on Image Processing
ISO	International Organization for Standardization
IEC	International Electrotechnical Commission
MJPEG	Motion JPEG
JPEG	Joint Photographic Experts Group
ROI	Region of Interest
IEEE	Institute of Electrical and Electronics Engineers

Anexos

A Motivación, técnica y montaje de las escenas en croma

En este anexo se presenta un resumen del artículo "A representative Ground-Trouth for the relative evaluation of motion based segmentation algorithms" pendiente de envío por invitación al congreso ICIP 2008.

A.1 Introducción

En los últimos años la segmentación espacial de objetos ha despertado un creciente interés en el campo del análisis de vídeo. De esta forma, si bien puede considerarse un tema aún en vías de investigación, numerosas aplicaciones en entornos de video vigilancia, biométrica o hipermedia adaptativa hacen ya un uso intensivo de las técnicas existentes. De forma paralela, también el estudio de mecanismos fiables para la validación de la calidad de los resultados obtenidos por estas técnicas está experimentando un fuerte desarrollo. La extensa documentación publicada al respecto no es sino un reflejo de la importancia y la complejidad de esta tarea. En los párrafos siguientes describiremos algunas de las aproximaciones más comunes para abordarla.

Un primer conjunto de aproximaciones, conocidas como técnicas subjetivas [1], se basan en la valoración de las máscaras finales de segmentación por parte de un número significativo de observadores. En este sentido, debe tenerse en cuenta que para que las puntuaciones asignadas a estas máscaras sean relevantes es necesario establecer previamente una serie de criterios y una metodología formal de evaluación. Ambos aspectos estarán, en gran medida, influenciados por la aplicación final de la segmentación. El principal inconveniente de estas aproximaciones es el importante coste que involucran, dado que para que los resultados sean estadísticamente significativos, se necesita un número relativamente elevado de personas durante el proceso de evaluación. Es por ello que suelen utilizarse solamente cuando el número de evaluaciones necesarias es bajo.

Un segundo conjunto de aproximaciones o métodos *stand-alone* [2] se basan en la comparación de una serie de propiedades relativas al tamaño, la textura, la forma o la distribución espacial de los objetos segmentados, con unos modelos preestablecidos que intentan reflejar los valores característicos en una segmentación "óptima". Sin embargo, la importante variabilidad (en cuanto a las propiedades mencionadas se refiere) en los objetos que pueden encontrarse en una situación general o las posibles interacciones entre estos objetos dificultan enormemente la construcción de dichos modelos. Por ello, su utilización queda restringida a entornos muy específicos.

Finalmente, hay un tercer conjunto de métodos conocidos como técnicas evaluación relativa, en los que la evaluación de los resultados se realiza por comparación respecto a una segmentación ideal de referencia, o *ground- truth*.. La precisión de la segmentación en estos casos se puede medir, bien en función de los porcentajes de error en la clasificación de los píxeles, o bien mediante métricas adicionales [3] en las que se considera la similitud entre máscaras desde un punto de vista semántico, atendiendo a propiedades de forma, el color o textura. Pueden destacarse además las métricas basadas en características

perceptuales [4] que resultan especialmente interesantes cuando el usuario final de la segmentación va a ser una persona.

Cuando la información de *ground- truth* de la que dependen las técnicas de evaluación relativa no está disponible, su extracción involucra un proceso muy costoso en términos de tiempo y esfuerzo. En este sentido debe tenerse en cuenta además, que el concepto de "objeto" depende enormemente de la aplicación, así por ejemplo, será diferente para un algoritmo de detección de caras que para un sistema de control de tráfico en tiempo real. Por lo tanto, no es posible hablar de un *ground- truth* "genérico" que pueda utilizarse para evaluar cualquier algoritmo de segmentación.

En este estudio, nos hemos centrado en la construcción de un *ground- truth* destinado a evaluar algoritmos que basándose en la información del dominio comprimido, discriminen los objetos del foreground respecto de los irrelevantes del background en base a criterios de movimiento.

A.2 Consideraciones para el diseño del Ground-Truth

La relevancia de los resultados obtenidos por medio de cualquier mecanismo de evaluación relativa depende, en gran medida, de la representatividad de las secuencias utilizadas durante esta evaluación. Parece natural, a su vez, que esta representatividad esté relacionada con la existencia de situaciones de baja, media y alta complejidad (entendiendo dicha complejidad como una estimación de lo difícil que puede resultar para un algoritmo segmentar dicha secuencia), lo que garantizaría que las conclusiones extraídas tras la evaluación abarcan el mayor número de casos posible.

Durante el estudio realizado en el área de segmentación de objetos basada en la información de movimiento, hemos detectado que la complejidad global de una secuencia depende de una serie de propiedades de los objetos, el fondo y el movimiento de cámara, involucrados en dicha secuencia. Estas propiedades se denominarán "factores críticos" y se describirán más adelante con un mayor detalle.

Los distintos valores de un factor crítico pueden incrementar (siendo entonces referidos como inicializaciones de alta complejidad) o reducir (dando lugar a inicializaciones de baja complejidad) significativamente la capacidad del algoritmo para obtener resultados precisos. De la misma forma también pueden existir inicializaciones complejidad media cuando el factor crítico puede tomar más de dos valores. En el diseño del *ground- truth* propuesto, hemos considerado que deben tenerse en cuenta inicializaciones tanto de complejidad alta, media y baja para cada uno de los factores críticos. Así, además de producir un conjunto de secuencias representativo para la evaluación, este *ground- truth* podrá usarse para identificar los puntos débiles de un algoritmo determinado. No obstante, si bien todas las inicializaciones comentadas intervienen en algún momento, por simplicidad y dado el elevado número de factores críticos, no se han tenido en cuenta todas las posibles combinaciones de estas inicializaciones. En particular, el grupo de combinaciones abordado se escogido en base a las que se han considerado más representativas de situaciones habituales en la vida real.

A.3 Factores críticos en los algoritmos de segmentación de objetos basados en movimiento

En este apartado enumeraremos los factores críticos de objetos, fondo y cámara, así como analizaremos su influencia en la complejidad de la secuencia. Asimismo, se presentarán posibles inicializaciones para estos factores que resulten en secuencias de alta y baja complejidad.

Factores críticos de objetos

Las propiedades de los objetos en movimiento que tienen una influencia directa y significativa en la segmentación pueden dividirse en propiedades de objetos individuales y propiedades relativas a grupos de objetos. Entre las propiedades de objetos individuales pueden citarse:

- Complejidad en textura: Considerando que los algoritmos de segmentación por movimiento deben estimar la diferencia entre un área determinada de un cuadro, y esa misma área en el cuadro anterior (en base a diferencias simples, mediante la evaluación de un modelo, por medio de cálculos a partir de aproximaciones del flujo óptico...), la estimación de esta diferencia será cuanto más fiable conforme la información espacial sea más representativa. Así inicializaciones de baja complejidad para este factor se corresponderán con objetos muy texturados. Alternativamente, la presencia de objetos homogéneos (la cual no permite la detección de cambios entre áreas), definirá inicializaciones de alta complejidad.

- Velocidad aparente: Es la suma de las velocidades de objetos y la cámara, y representa la velocidad de los objetos percibida en la secuencia. Por simplicidad la denominaremos simplemente "velocidad". Por un lado, debido al ruido y las inestabilidades típicas del fondo, es difícil extraer objetos que cambian muy lentamente a lo largo del video. Por otro lado, cualquier aproximación basada en una estimación previa del flujo óptico debe tener en cuenta que los objetos que se muevan muy rápido a lo largo del video requieren de ventanas de búsqueda muy grandes, lo que degrada la eficiencia e incluso influye en la calidad de los resultados de esta búsqueda (puesto que ventanas más grandes implican también mayor probabilidad de escoger elegir referencias inadecuadas). Además, cuando se trabaja en el dominio comprimido (como por ejemplo con vectores de movimiento MPEG), el tamaño de la ventana de búsqueda no puede ser alterado cuando se realiza la segmentación. En conclusión, objetos muy lentos o muy rápidos serán introducidos en inicializaciones de alta complejidad, mientras que objetos con velocidades comparables a la de la cámara aparecerán en las de baja complejidad.

- Estructura del objeto: Dependiendo de si en el objeto podemos encontrar un único patrón de movimiento o varios, podemos clasificarlo como rígido o no rígido respectivamente. Si tratamos de segmentar objetos rígidos, podemos establecer restricciones basadas en la coherencia espacial de su movimiento para facilitar su extracción. Para algunos objetos no rígidos, podemos hacer uso de estas restricciones si dividimos el objeto en áreas no rígidas, pero es complicado reconstruir el objeto a partir de ellas, puesto que pueden existir áreas del objeto inmóviles mientras el resto del objeto se encuentra en movimiento, como por

ejemplo, alternativamente, cada uno de los pies estáticos de una persona mientras camina. Por otra parte existen objetos no rígidos cuyo movimiento es totalmente caótico, en los que no puede realizarse ninguna aproximación de rigidez para ninguna de sus partes. Inicializaciones de alta complejidad serán aquellas que incluyan estos últimos, mientras que el resto de objetos no rígidos pueden incluirse en inicializaciones de complejidad media e incluso, junto con los rígidos, en aquellas de baja complejidad.

- Descubrimiento de objetos: Cuando un objeto o partes de él anteriormente ocultas se muestran como consecuencia de su movimiento, la ventana de búsqueda en el proceso de codificación no puede encontrar correspondencias para esas nuevas áreas en el video, y estos objetos suelen intra-codificarse. Esto repercute en la correcta estimación del flujo óptico, así como en la incapacidad de usar mecanismos de proyección basados en los vectores de movimiento (como el *tracking*) muy útiles en el seguimiento y control de la evolución temporal de los objetos. Por ello, inicializaciones de baja complejidad deben de evitar incluir este tipo de descubrimiento que, es debe considerarse por el contrario, en inicializaciones de media y alta complejidad.

- Tamaño del objeto: Este factor puede ser normalmente ignorado en los algoritmos de segmentación en casos de cámara fija. Por el contrario para algoritmos capaces de segmentar en situaciones de movimiento de cámara, el tamaño del background ha de ser mucho mayor que el de los objetos presentes en él, garantizando así, que el movimiento de cámara puede ser estimado como el dominante. Respecto al tamaño mínimo de los objetos no existen restricciones particulares salvo las exigidas por la semántica del propio algoritmo (para evitar segmentar regiones demasiado pequeñas). Así objetos de tamaño cercano al del background se corresponderán a inicializaciones de alta complejidad, mientras que el resto de tamaños (superiores a las restricciones específicas de cada algoritmo) serán incluidos en inicializaciones de baja complejidad.

En cuanto a los factores críticos que pueden asociarse a grupos de objetos han sido analizados los siguientes:

- Diferencia entre velocidades de objetos en un cuadro: Si en un mismo cuadro coexisten objetos cuyas velocidades son muy diferentes, la influencia del objeto rápido sobre el umbral con el que se discrimina el movimiento de los objetos del de cámara (muy utilizado en sistemas de segmentación adaptativos), puede suponer que el objeto lento no sea detectado. Así, inicializaciones de alta complejidad contendrán objetos de velocidades muy diferentes, mientras que en las de media y baja complejidad, éstos tendrán velocidades similares.

- Interacciones complejas entre objetos: Existen varios factores críticos que afectan negativamente tanto a la correcta estimación del flujo óptico, como al seguimiento y estudio de la evolución temporal de objetos:

    - Intersección de trayectorias. Cuando las trayectorias de dos o más objetos se cruzan, se produce el fenómeno denominado oclusión entre objetos. La correcta detección de la forma de estos objetos en las máscaras de segmentación vendrá determinada por la capacidad del algoritmo para

detectar y gestionar estas oclusiones. Por lo tanto, sólo deben tenerse en cuenta en inicializaciones de alta complejidad.

- Fusión y división de objetos. La separación no supervisada de objetos que se fusionan y se mantienen unidos durante un periodo de tiempo, es una tarea complicada. Labores como el tracking se ven perjudicadas por este tipo de situaciones. Por otro lado, si se pierden las referencias a los objetos originales, y estos objetos fusionados se dividen, se crearán nuevos objetos, degradando así la capacidad de segmentar o seguir un objeto a lo largo del tiempo. Por lo tanto, este tipo de interacciones sólo deben aparecer en inicializaciones de complejidad alta.

Factores críticos del Background

- Textura: Como indicamos anteriormente, las estimaciones de movimiento en áreas texturadas son mucho más fiables que aquellas correspondientes a áreas homogéneas, dado que en estas últimas pueden existir cientos de correspondencias válidas para cada uno punto. Por ello, en el caso de técnicas basadas en estimaciones del flujo óptico, las áreas homogéneas tienen a menudo vectores que no se corresponden con el movimiento dominante y suelen ser consideradas como objetos. Por tanto, backgrounds con numerosas regiones homogéneas estarán presentes en las inicializaciones de alta complejidad, mientras que backgrounds altamente texturados darán lugar a secuencias de baja complejidad.

- Multimodalidad: Es el factor que explica las pequeñas variaciones de algunos backgrounds debido a elementos como agua en movimiento, fuego, o plantas mecidas por el viento, que normalmente son consideradas irrelevantes desde un punto de vista semántico, pero que tienen una importante influencia negativa en los resultados de los algoritmos de segmentación. Por tanto, la presencia de un background multimodal determinará una inicialización de alta complejidad.

Factores críticos del movimiento de cámara

- Esquema de movimiento de cámara: El movimiento de la cámara influye sobre todas las características relacionadas con el movimiento en una secuencia, influyendo por lo tanto también en los resultados de la segmentación. El caso más simple es el de una cámara estática, que definirá la inicialización de complejidad más baja. La uniformidad en el movimiento de cámara puede estimarse robustamente y aprovecharse durante la segmentación, así, un movimiento de cámara uniforme se asociará con inicializaciones de complejidad media. Finalmente, las inicializaciones de complejidad alta deberán considerar movimientos de cámara rápidos y “a saltos”. En este caso, la escasa duración temporal de cada patrón de movimiento impide su estimación robusta, con los consecuentes resultados en la segmentación de objetos.

A.4 Creación de las secuencias

Considerando todos estos factores, redactamos una serie de guiones (ver Anexo B) que consideran diferentes combinaciones relevantes de factores críticos con inicializaciones de diferentes complejidades. Grabamos el foreground, es decir, los objetos, en un estudio de croma, y el background, por separado, en escenarios interiores y exteriores. Para esta grabación utilizamos cámaras de alta resolución con las que conseguimos video progresivo YUV descomprimido a una resolución de 720 x 576 a 25 cuadros por segundo.

A.5 Referencias

[1] K. McKoen, R. Navarro-Prieto, B. Duc, E. Durucan, F. Ziliani, and T. Ebrahimi, "Evaluation of video segmentation methods for surveillance applications," in *Proc. Eur. Signal and Image Processing Conf.*, vol. II, Tampere, Finland, pp. 1045–1048, Sept. 2000.

[2] M. Borsotti, P. Campdelli, and P. Schettini, "Quantitative evaluation of color image segmentation results," *Pattern Recognit. Lett.*, vol. 19, pp. 741–747, 1998

[3] P. Correia and F. Pereira, "Objective evaluation of relative segmentation quality," in *Proc. Int. Conference on Image Processing*, vol. 2, Vancouver, Canada, pp. 308–311, September 2000.

[4] M. Caramma, R. Lancini, and M. Marconi, "A perceptual PSNR based on the utilization of a linear model of HVS, motion vectors and DFT-3D," in *Proc. Eur. Signal and Image Processing Conf.*, vol. IV, Tampere, Finland, pp. 2185–2188, Sept. 2000,

B Guiones

Escenas de Foreground

Requeriremos de una serie de elementos para la grabación de las secuencias, a saber:

Objetos	Para dar soporte a
a. Peluches/Juguetes a cuerda Nota: La elección del peluche conlleva la necesaria utilización de guantes del color del croma, por ello sería preferible la selección de o el uso de algún tipo de juguete 'a cuerda' que ande 'solo' como puede ser un robot tipo Bender. b. Coche teledirigido, muy útil en la escena 2, podría sustituir al Bender en la escena 1 c. Periódicos. Deberían ser periódicos o revistas con muchas fotos o publicidad en color, para poder usarlo tanto como objeto poco o muy texturado. d. Caja de cartón, ha de ser lo suficientemente grande como para poder contener al coche teledirigido e. Raquetas de tenis playa	Objetos rígidos texturados
f. Personas El mínimo de personas necesarias es tres, pero para la escena 10 la posibilidad de contar con más personas incrementa la validez de la secuencias para testar el algoritmo. Dado que ahora no tenemos restricciones de espacio, o tenemos menos, cuantas más personas interactúen mayor será la exigencia a la que sometamos el algoritmo	Personas
g. Frutas monocromas-pera h. Pelota mediana monocroma. i. Pelota de tenis playa. j. Bolígrafos, lápices…material de oficina monocromo.	Objetos texturados 'no rígidos'. Sin y con alteraciones.

Describiremos los guiones utilizados para la grabación de las secuencias correspondientes al foreground:

Escenas del foreground		Para dar soporte a
1	'El paseo del peluche' El peluche atraviesa la escena realizando diferentes movimientos, saltos, cambio de trayectoria, sin giros ni rotaciones, tipo 2D.-Podemos utilizar el Bender, o el coche teledirigido	Situación básica: Objeto rígido texturado que se mueve a una velocidad determinada y sin más objetos interfiriendo

2	‘Mejor quedarse en casa’ La caja de cartón aparece gradualmente en la escena, a escasos centímetros de ser posada en el suelo, el coche teledirigido sale de la misma-la caja ha de estar en situación horizontal-mientras se mueve, un periódico intenta golpearlo.	Objetos texturados con movimiento rígido y no rígido que interaccionan. Objetos que se moverán a distintas velocidades, se dividen y fusionan.
3	‘El hombre caluroso’ La escena comienza con el hombre en un extremo del plano, vestido con cazadora. El hombre se mueve y realiza gestos de cansancio, sofoco y agobio, se quita la cazadora poco a poco y termina con gestos de alivio en el otro extremo del plano. Para intentar tener objetos con distintos movimientos, la cazadora debería ser soltada con el mayor ángulo posible, permitiendo así la descripción de una parábola lo suficientemente grande como para coexistir con el hombre que continúa caminando.	Objeto texturado con movimiento no rígido que interacciona con objeto texturado de movimiento más o menos rígido. Objetos que se dividen: cazadora-hombre, y que se mueven a velocidades diferentes.
4	‘Ingravidez’ La pelota monocroma cuelga del truss mediante un hilo transparente-o cuerda si lo otro no es viable, la persona intenta golpearla con la cabeza, produciendo movimientos no uniformes. La longitud de la secuencia sería de alrededor de 30 segundos. Es una secuencia ideal para el uso del zoom, si esta función de encuentra disponible en las cámaras utilizadas para la grabación, podríamos empezar con un plano sobre la pelota sola, debería oscilar para ser segmentada, después abrir el plano hasta permitir que la persona golpeando la pelota aparezca, y que los movimientos pendulares de la misma no se salgan del plano, al finalizar, un zoom in, hacia la pelota en movimiento completaría el ciclo. Al tratarse de una situación con escaso movimiento espacial-en realidad limitado por la longitud del hilo-.el cámara podría rodear la escena, y así conseguir cambios de plano, que generarían un efecto curioso posteriormente al montaje con el fondo.	Objeto texturado y objeto no texturado que interaccionan. Objetos que se cruzan, se fusionan y se dividen. Movimientos rígidos y no rígidos. Distintas velocidades cuando están separados.
5	‘Amor fetichista’ Una chica acaricia al peluche y lo mueve, lo abraza y lo acaricia. Al finalizar lo lanza al suelo desolada. La chica podría simular una conversación con el peluche, y reaccionar ante una negativa del mismo arrojándolo y saliendo corriendo del plano.	Objetos texturados que interaccionan. Múltiples movimientos no rígidos. Descubrimiento de macrobloques que serán intracodificados. Al finalizar división de objetos
6	‘El baile’ Dos personas bailan cruzándose y separándose y una tercera se suma posteriormente a la danza.	Objetos texturados con movimiento no rígido que interaccionan. Diferentes velocidades entre los objetos. Situaciones de fusión y división constantes. Situaciones de cruce y oclusión constantes. Situaciones de objetos texturados que no se solapan. Aparición de nuevos objetos en el plano.

7	‘Maldito domingo’ Un hombre -que se encuentra desde el principio de la escena sentado sobre una caja, una silla, o el suelo- lee el periódico tranquilamente, hasta que se azora y lo agita enervado. Una segunda persona aparece por el otro extremo del plano, le tira el periódico al suelo y sigue andando hacia el otro extremo del plano. Para permitir la situación de coexistencia de objetos que se mueven a velocidades diferentes, el hombre que lee debería moverse levemente, asombrándose perplejo encogiendo hombros y moviendo la cabeza observando fijamente al segundo sujeto, cuando éste último esté saliendo del plano, levantarse súbitamente y correr hacia el otro lado del mismo injuriando y haciendo aspavientos.	Objetos texturados con movimiento rígido y no rígido. Interacción entre objetos, división-fusión. Cruce entre objetos con diferentes velocidades al moverse. Aparición de nuevos objetos en el plano. Necesaria detección de objetos que se mueven a distintas velocidades y comparten plano.
8	‘La carrera desigual’ Un hombre visiblemente cansado aparece en el plano por la derecha, muerde la fruta monocroma y la lanza al suelo con el mayor ángulo posible sin salirse del plano. En el instante del impacto, y aproximadamente cuando la persona cansada atraviesa la mitad del plano, aparece una segunda persona que lo adelanta más relajado y agitando un bolígrafo-o algún objeto poco texturado- mientras lo recrimina girando la cabeza para mirarle desde adelante y finalmente fijando la vista al frente levantando los brazos como observando la meta. Esta escena debería ser grabada con un pan largo, permitiendo así mayor capacidad de movimiento a las personas-objetos-y un mayor realismo en la secuencia.	Objetos texturados y no texturados interaccionan. División de objetos: texturado-no texturado Oclusión entre objetos a distintas velocidades. Necesaria detección de objetos que se mueven a distintas velocidades y comparten plano. Objetos texturados con movimientos no rígidos interactúan entre sí y con objetos poco texturados de movimientos rígidos. Fomenta aparición de intras y oclusiones.
9	‘Agua falsa’ Una persona intenta beber del envase, y lo levanta por encima de la cabeza hasta que le cae sobre la cara un periódico arrugado, que se encontraba en el interior del envase, con su cara más texturada hacia la cámara. El hombre se agacha, recoge el papel arrugado y lo extiende, examinando su interior, después vuelve a arrugarlo y lo estruja sobre su cabeza/boca, esperando conseguir algunas gotas. Tras su fracaso, vuelve a lanzarlo al suelo y o bien cae con él, o bien se arrastra/deambula hasta salir del plano.	Interacción de objetos texturados de tamaños muy distintos que se mueven, se cruzan, se fusionan momentáneamente y se dividen. Cabe la posibilidad de controlar el envase por separado y observar varios objetos texturados que se mueven a distintas velocidades.

10	‘No serás capaz de alcanzarme’ Persecución, aspavientos y movimientos, organización similar a la de la carrera pero con más personas. La idea es la realización de un pan lo más extenso posible ateniéndonos a las limitaciones del plató, en el cual los objetos entren en diferentes momentos al plano, se alcancen, se adelanten y se ocluyan, se lancen objetos-fruta, peluches, bolígrafos- y éstos los impacten, pero para mayor versatilidad, permitir que haya objetos que se muevan más rápido y no sean alcanzados. Supongamos una situación con 4 personas. El sujeto A entra el primero en la escena y corre hacia delante Tras 2 segundos entra el sujeto B y antes de que el sujeto A salga del plano entran a escena los sujetos C y D al mismo tiempo. La cámara intenta seguir a todos los objetos sin que ninguno se salga del plano. El objeto B lanza objetos al objeto A, mientras el C y el D forcejean por adelantarse, hasta que uno de los dos, digamos el objeto C logra imponerse. El objeto D lanza objetos al C. Si alguno de los dos-A o C-es impactado se gira, se para y corre hacia el agresor, que intenta adelantarlo o sucumbe al contraataque, la escena termina cuando todos hayan llegado al extremo del plató y desaparezcan del plano.	Situación muy completa con todo tipo de objetos y todo tipo de interacciones, velocidades diferentes, ancho margen de movilidad y la prueba más exigente para el algoritmo.
11	‘La justa del tenis playa o los domingueros’ Dos personas aparecen jugando al tenis playa, se permiten interacciones entre hombre-pelota recogida de la misma del suelo, raqueta-pelota, y personas escenario-entrada y salida del plano en cámara fija.´	Situación en la cual objetos no-rígidos texturados interactúan con objetos rígidos más o menos texturados, depende del tipo de pelota utilizada. Pueden interactuar en diferentes planos, casi con seguridad se moverán a diferentes velocidades, y pueden darse situaciones en las que un objeto impacte sobre el otro.
12	‘Solitario’ Una persona rebota una pelota repidamente sobre una raqueta de tenis	Situación en la cual un objeto no-rígido texturado interactúa con objetos rígidos más o menos texturados. Situación de complejidad menor que la anterior.

13	'Bueno días caballeros-Buenos días señoritas' Escena muy simple en la cual se realiza un serie de encuentros entre personas sobre el eje z de la cámara –es decir perpendicular al fondo del croma. La gente aparece desde el croma y desde la cámara.	Situación crítica en la cual los objetos en movimiento no varían mucho en los mb que ocupan de una frame a otra, interesa observar el funcionamiento de tracking en estos casos.

Escenas de Background

En cuanto a las escenas del background, los guiones que las describen y justifican son:

Escenas del Background		Para dar soporte a
1	'El Laboratorio' PTZ y cámara fija sobre el laboratorio evitando la aparición de gente trabajando susceptible de ser segmentado como objetos en movimiento	Situación básica. Background unimodal con elementos texturados y poco texturados.
2	'Esperando al ascensor' PTZ y cámara fija sobre el descansillo de la cuarta planta. Intentar evitar la aparición de personas en el plano.	Background unimodal con elementos poco texturados.
3	'Verde' PTZ y cámara fija sobre un prado	Background unimodal con elementos texturados.
4	'El aparcamiento de la segunda planta' PTZ y cámara fija sobre el aparcamiento	Background unimodal con elementos texturados y poco texturados.
5	'Vendaval en el aparcamiento de la segunda planta' PTZ y cámara fija sobre el aparcamiento, incluyendo el descampado próximo con algunas de sus plantas y arbustos en movimiento por el viento y algunos coches estacionados.	Background multimodal con elementos con cambios poco significativos (plantas).
6	'El impredecible cielo' PTZ y cámara fija sobre el cielo, incluyendo si las condiciones lo permiten nubes.	Background susceptible de ser unimodal texturado, o multimodal según luz, viento, iluminación... Situación muy común en muchos videos

7	‘La fuerza del agua’ PTZ y cámara fija sobre agua en movimiento. Pendiente de elección del escenario; Estanques, ríos, fuentes, mar…	Background multimodal con mucho ruido, movimiento no susceptible de ser parametrizado mediante algún modelo, de cambios muy significativos.
8	‘La prisa mata’ PTZ y cámara fija sobre un semáforo de peatones en verde, cuando esté parpadeando y cuando cambie a rojo. Intentar evitar a los peatones.	Background multimodal con cambios muy significativos.
9	‘Te llaman’ Crear un escenario repleto de objetos inmóviles, entre ellos un móvil en situación stand by, llamar al móvil y grabar en cámara fija y PTZ su iluminación.	Background unimodal con cambios significativos.
10	‘Duelo al sol’ Escenario: un descampado a ser posible repleto de plantas secas y cardos. PTZ sobre un ángulo amplio, intentando incluir movimientos de la vegetación.	Background multimodal con cambios poco significativos. ¡Ojo!, pueden aparecer cambios de iluminación en toda la escena debido a la luz natural, situación intrínseca a todos los videos, si es gradual-lo más común- es resoluble.
11	‘La cruz de guía’ PTZ y cámara fija sobre la cruz verde indicadora de farmacia, mientras los LEDS que la forman se apagan y encienden	Background multimodal con cambios muy significativos
12	‘Caleidoscopio’ Usar algún fondo de pantalla o video artificial con formas similares a las del caleidoscopio.	Background multimodal con cambios muy significativos
13	‘El techo’ Grabación PTZ del techo y sus fluorescentes.	Background unimodal de elementos texturados con cambios posiblemente poco significativos. ¡Ojo con el parpadeo del fluorescente y la frecuencia de grabación!
14	‘Hoguera’ PTZ y cámara fija sobre fuego.	Background multimodal con cambios muy significativos

7 Presupuesto

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... ..... 2.000 €
- Material de oficina ..... 150 €
- Total de ejecución material ..... 2.150 €

2) Gastos generales

- 16 % sobre Ejecución Material ..... 344 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material ..... 129 €

4) Honorarios Proyecto

- 800 horas a 15 € / hora ..... 12.000 €

5) Material fungible

- Gastos de impresión ..... 60 €
- Encuadernación ..... 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto ..... 14.410 €

7) I.V.A. aplicable

- 16% Subtotal Presupuesto ..... 2.305,6 €

8) Total presupuesto

- Total Presupuesto ..... 16.715,6 €

Madrid, Febrero de 2008

El Ingeniero Jefe de Proyecto

Fdo.: Marcos Escudero Viñolo
Ingeniero de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un sistema de segmentación de objetos espacio-temporales en movimiento. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es

obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad “Presupuesto de Ejecución de Contrata” y anteriormente llamado ”Presupuesto de Ejecución Material” que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.