

UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



Detección de objetos estáticos de primer plano en escenarios altamente concurridos de video-seguridad

-PROYECTO FIN DE CARRERA-

Diego Ortego Hernández

Septiembre 2013

Detección de objetos estáticos de primer plano en escenarios altamente concurridos de video-seguridad

Autor: Diego Ortego Hernández

Tutor: Juan Carlos San Miguel Avedillo

Ponente: José María Martínez Sánchez

email: {diego.ortego@estudiante.uam.es, Juancarlos.Sanmiguel@uam.es, JoseM.Martínez@uam.es}



Video Processing and Understanding Lab

Departamento de Tecnología Electrónica y de las Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Septiembre 2013

Trabajo parcialmente financiado por el gobierno español bajo el proyecto TEC2011-25995 (EventVideo)



Resumen

En este trabajo, se propone un algoritmo de detección de regiones estacionarias de primer plano en entornos densamente poblados de video-vigilancia basado en la acumulación espacio-temporal de información de frente, movimiento y estructura.

En primer lugar, se realiza un estudio del estado del arte para conocer los problemas actuales en la materia. Después el trabajo se centra en combinar distintas informaciones que permitan filtrar el mayor número de falsos positivos posible. Para ello, las regiones de interés se obtienen mediante una sustracción de fondo. La información de movimiento es estimada aplicando filtros de mediana sobre ventanas temporales, permitiendo así filtrar regiones con objetos en constante movimiento. Adicionalmente se emplea información de estructura para determinar qué zonas solo están activas en el frente debido a un cambio de la iluminación y removerlas del resultado final.

Finalmente, se evalúa el nuevo algoritmo sobre numerosas secuencias, verificando así la notable mejora respecto a las aproximaciones de la literatura.

Palabras clave

Frame, imagen, sustracción de fondo, frente, fondo, blob, movimiento, estructura, acumulación, oclusión, región estática.

Abstract

In this work, we propose an approach for stationary foreground detection in video-surveillance based on spatio-temporal variation of foreground, motion and structure data.

First, an study of related work has been done. Then the work is focused on the combination of different features to reduce false positives to the minimum. To achieve this target, the regions of interest are obtained by background subtraction. Motion information allows to filter out the moving regions and it is estimated using median filters over sliding windows. Furthermore, structure information is used to compute which areas are activated in the foreground just for being a shadow or illumination change, with the purpose of remove them from the final result.

Finally, the results over challenging video-surveillance sequences show a notable improvement of the proposed approach against the related work.

Keywords

Frame, image, background subtraction, foreground, background, blob, motion, structure, accumulation, occlusion, static region.

Agradecimientos

En primer lugar, quiero agradecer a mi tutor, Juan Carlos San Miguel, la gran ayuda y atención que ha tenido conmigo durante la realización del PFC a lo largo del último año, no podría haber tenido un tutor mejor.

Quiero dar las gracias a mis padres y mi hermana por su apoyo durante toda la carrera y por haberme guiado siempre por el buen camino. Un recuerdo también para mis abuelos (los que están y los que ya no) que están muy orgullosos de tener un Ingeniero más en la familia. También quiero acordarme de mis tíos, Pili y Alfonso, y mis primos Carlos y Pilar con los que he pasado muy buenos momentos.

Quiero hacer especial mención a mi primo y amigo Carlos, Teleco ya desde hace unos años y un ejemplo a seguir, que me ayudó muchísimo en los comienzos de la carrera.

Especial mención a mis amigos Sergio, Joaquín, Víctor, Dani, Elena, Marta, Pablo, Mario, Willy, Víctor (suizo), Dani (suizo), que me hacen pasar siempre muy buenos ratos.

También agradecer a mis compañeros y amigos de clase Luis, Gonzalo, Cecilia, Guille y Juanpa por los grandes momentos que hemos compartido a lo largo de estos años en la EPS.

Agradecer también la ayuda que siempre me han prestado los compañeros del VPU siempre que la he necesitado.

Gracias a todos.

Diego Ortego Hernández

Septiembre 2013

Índice general

1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	3
1.3. Organización de la memoria	4
2. Estado del arte	5
2.1. Introducción	5
2.2. Sustracción de fondo	6
2.3. Clasificación de métodos de detección de regiones estáticas	7
2.4. Métodos de detección de regiones estáticas seleccionados	9
2.4.1. Acumulación de máscaras de frente	9
2.4.2. Submuestreo de máscaras de frente	11
2.4.3. Submuestreo de máscaras de frente y movimiento	13
2.4.4. Dos modelos de fondo actualizados a diferentes velocidades	16
2.4.5. Dos modelos de fondo actualizados a diferentes velocidades (avanzado)	20
2.4.6. Análisis a nivel de región para validar regiones de frente	23
2.4.7. Propiedades del modelo de fondo e interacción con una etapa de <i>tracking</i>	25
2.5. Limitaciones de las métodos existentes	27
3. Algoritmo propuesto	29
3.1. Estructura general	29
3.2. Análisis realizados	30
3.2.1. Análisis de frente	30
3.2.2. Análisis de movimiento	31
3.2.3. Análisis de estructura	37
3.3. Combinación de análisis	42
3.4. Manejo de oclusiones	43
3.5. Configuración del algoritmo	45

4. Resultados experimentales	47
4.1. <i>Datasets</i> disponibles	47
4.2. Métrica	49
4.3. Resultados	50
5. Conclusiones y trabajo futuro	61
5.1. Conclusiones	61
5.2. Trabajo futuro	63
Bibliografía	65
Appendix.	68
A. Apéndice 1	69
A.0.1. Otras características descartadas	69
B. Publicaciones	75
C. Presupuesto	83
D. Pliego de condiciones	85

Índice de figuras

1.1. Ejemplos de objeto abandonado, robado y un vehículo estacionado ilegalmente.	1
1.2. Ejemplos de máscara de frente y modelo de fondo para detección de objetos abandonados y vehículos.	2
1.3. Ejemplos erróneos de detección de eventos utilizando sustracción de fondo.	3
2.1. Diagrama habitual de un sistema de vídeo-seguridad.	6
2.2. Ejemplos de sustracción de fondo en secuencias de vídeo-seguridad.	7
2.3. Ejemplo de detección de regiones estáticas basado en acumulación.	10
2.4. Ejemplos de errores del algoritmo basado en acumulación de máscaras.	11
2.5. Esquema de muestreo y combinación para obtener regiones estacionarias.	12
2.6. Ejemplos de error de la aproximación basada en acumulación de máscaras.	13
2.7. Esquema de funcionamiento del algoritmo basado en muestreo de máscaras de frente y movimiento.	14
2.8. Errores del algoritmo basado en muestreo de máscaras de frente y movimiento.	15
2.9. Hipótesis establecidas en función de las dos máscaras de frente a largo y corto plazo.	17
2.10. Errores de la aproximación basada en combinar dos modelos de fondo (1).	18
2.11. Errores de la aproximación basada en combinar dos modelos de fondo (2).	19
2.12. Errores de la aproximación basada en combinar dos modelos de fondo (3).	20
2.13. Máquina de estados que modela el histórico de clasificaciones sobre los píxeles.	21
2.14. Ejemplos de funcionamiento para la máquina de estados (FSM) que modela los píxeles en función de su histórico de hipótesis.	22
2.15. Esquema de la aproximación basada en análisis de regiones.	23
2.16. Ejemplos de funcionamiento del análisis de regiones.	24
2.17. Esquema del algoritmo basado en propiedades del modelo.	25
2.18. Ejemplos de funcionamiento del algoritmo basado en propiedades del modelo.	27
3.1. Esquema del algoritmo propuesto.	29
3.2. Ejemplos de la acumulación de frente propuesta.	31

3.3.	Ejemplos de errores de la acumulación de frente propuesta.	32
3.4.	Esquema propuesto para extracción de movimiento inventanada.	33
3.5.	Comparativa de movimiento obtenido con diferentes técnicas de cálculo del umbral.	34
3.6.	Ejemplo de $MHI_t(\mathbf{x})$ empleando la característica de movimiento propuesta (PRO) y el <i>frame-difference</i> básico (FD).	35
3.7.	Ejemplo de extracción de movimiento en condiciones de oclusiones.	36
3.8.	Esquema para el cálculo de la medida de similitud SSIM.	38
3.9.	Ejemplos de la medida de similitud SSIM.	39
3.10.	Esquema de cálculo de la modificación de <i>SSIM</i> , el mapa <i>RSSIM</i>	40
3.11.	Efecto del tamaño para calcular la característica SSIM.	41
3.12.	Resultados del análisis de estructura.	42
3.13.	Ejemplos del algoritmo final propuesto.	44
3.14.	Ejemplos del manejo de oclusiones propuesto.	45
4.1.	Ejemplo de las 4 perspectivas disponibles en PETS2006.	48
4.2.	Ejemplo de las 4 perspectivas disponibles en PETS2007.	48
4.3.	Ejemplo de los dos escenarios disponibles en AVSS2007.	48
4.4.	Ejemplo de las 3 secuencias grabadas.	49
4.5.	Comparativa de características para mostrar la aportación individual de cada una.	51
4.6.	Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (1).	55
4.7.	Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (2).	56
4.8.	Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (3).	58
4.9.	Ejemplo de errores del algoritmo propuesto.	59
A.1.	Característica de estructura SSIM como post-procesado (1).	70
A.2.	Característica de estructura SSIM como post-procesado (2).	72
A.3.	Característica de color (romaticidad) como post-procesado.	72
A.4.	Ejemplo de característica de color como una característica más a acumular para una secuencia creada artificialmente con movimiento continuado.	73
A.5.	Algoritmos robo/abandono para eliminar detecciones fantasma	74

Índice de tablas

2.1. Comparativa de los métodos del estado del arte más relevantes.	28
4.1. Descripción de las secuencias empleadas en la evaluación.	49
4.2. Comparativa de las diferentes características.	52
4.3. Comparativa de resultados del algoritmo propuesto frente a técnicas del estado del arte.	54

Acrónimos

BLOB	<i>Binary Large Object</i>
FD	<i>Frame-difference</i>
MoG	<i>Mixture of Gaussians</i>
SSIM	<i>Structural Similarity</i>
FSM	Finite State Machine
SFG	Stationary Foreground
I	<i>Image</i>
F	<i>Foreground</i>
B	Background
FHI	<i>Foreground History Image</i>
MHI	<i>Motion History Image</i>
RHI	<i>Region History Image</i>
SHI	<i>Stationary History Image</i>
GT	<i>Ground-Truth</i>
TP	<i>True Positive</i>
FP	<i>True Negative</i>
FN	<i>False Negative</i>
HSV	<i>Hue Saturation Value</i>

Capítulo 1

Introducción

1.1. Motivación

En la actualidad, el análisis automático de secuencias de vídeo-vigilancia es un área importante de investigación como consecuencia de la necesidad de seguridad en entornos públicos como aeropuertos, estaciones de tren o metro y eventos masivos. En este contexto, cobra importancia la detección de regiones estáticas [1][2][3], es decir, el reconocimiento de aquellos objetos/personas/vehículos que permanezcan inmóviles en la secuencia de vídeo durante un tiempo determinado. Estos algoritmos se emplean en diversas aplicaciones, como puede ser la detección de vehículos estacionados en monitorización de tráfico [4] o la detección de interacciones persona-objeto en una escena [5], tales como robo o abandono [6] (ver Figura 1.1).

Comúnmente se requiere que las aplicaciones desarrolladas operen correctamente en entornos complejos [1][7][8][9], proporcionando una solución a los problemas derivados de la alta concurrencia de objetos o de la alta variabilidad presente en las secuencias de vídeo. Adicionalmente el análisis en tiempo real es deseable en aplicaciones de apoyo al personal de seguridad/monitorización.



Figura 1.1: Ejemplos de objeto abandonado (izquierda), robado (centro) y un vehículo estacionado ilegalmente (derecha), indicados mediante un rectángulo rojo.

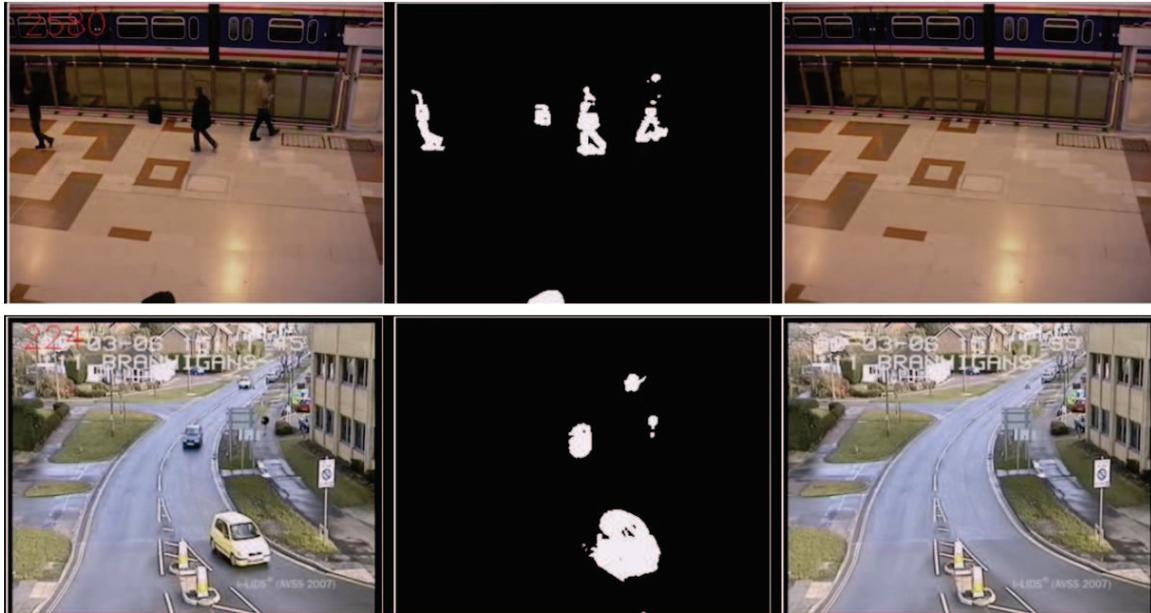


Figura 1.2: Ejemplos de máscara de frente (centro) y modelo de fondo (derecha) del *frame* bajo análisis (izquierda) para detección de objetos abandonados (primera fila) y vehículos (segunda fila).

Es habitual que la detección de regiones estáticas se base en una etapa previa de detección de objetos utilizando sustracción de fondo¹ [10][11], que como se muestra en la Figura 1.2, permite obtener una máscara binaria con las regiones pertenecientes al frente de la imagen² a partir de un modelo de fondo establecido. Esta etapa es crítica, pues para obtener el frente de la imagen se necesita comparar el *frame* actual con un modelo de fondo, cuya inicialización es especialmente compleja en situaciones de alta densidad de objetos en la escena. Por tanto, la información obtenida en esta etapa inicial es crucial y va a marcar el funcionamiento de técnicas posteriores, pues deberán dar solución a los errores surgidos.

Tras disponer de una máscara con los objetos de interés, se detectan las regiones estacionarias mediante el uso de un algoritmo [1][7][9][12][13]. No obstante, el análisis se complica en escenarios densamente poblados, donde se producen numerosos cambios de iluminación, oclusiones continuadas y gran concurrencia de objetos (y en consecuencia de sus sombras) que merman el rendimiento de los algoritmos.

Finalmente, la información obtenida es utilizada en etapas de alto nivel, donde se emplean métodos que persiguen identificar las interacciones y actividades que tienen lugar en la escena

¹A lo largo de esta memoria se empleará el término sustracción de fondo para designar indistintamente lo que en otros trabajos se designa como algoritmo de *background subtraction*, de segmentación fondo-figura o de segmentación frente-fondo.

²A lo largo de la memoria se empleará el término frente para designar el primer plano o *foreground* de la imagen obtenido en la sustracción de fondo.

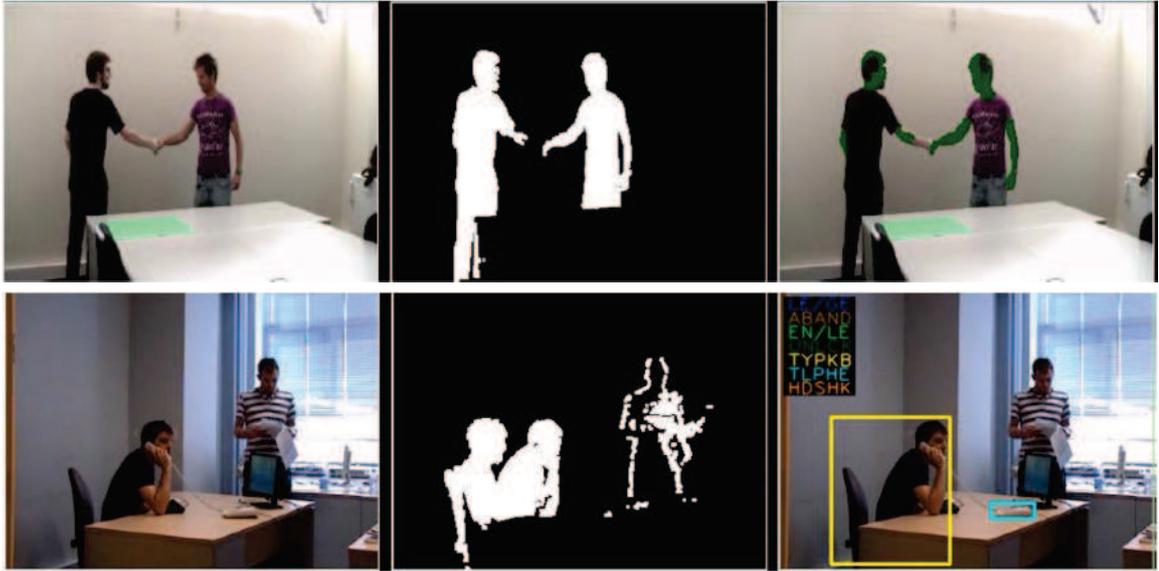


Figura 1.3: Detección de eventos. Primera fila: detección de eventos fallida (derecha) debido al fallo en la sustracción de fondo (centro) al detectar el brazo de la persona de la izquierda en el *frame* bajo análisis (izquierda). Segunda fila: El rectángulo amarillo (derecha) marca un evento de tecleo cuando en realidad se está hablando por teléfono.

[5] y que actualmente tampoco consiguen un rendimiento aceptable en escenarios altamente concurridos, debido a su dependencia con etapas previas (ver primera fila de Figura 1.3) y a la alta variabilidad y especificidad de situaciones que pueden producirse (ver segunda fila de Figura 1.3).

Por tanto, la motivación de este proyecto es contribuir al área de detección de regiones estáticas donde los estudios recientes muestran una baja eficiencia en escenarios complejos [14]. Además, esta etapa es de especial interés para mejorar los resultados de los sistemas actuales de vídeo-vigilancia que hacen uso de técnicas de este tipo para llevar a cabo análisis a más alto nivel.

1.2. Objetivos

El objetivo de este PFC es el desarrollo de un algoritmo de detección de regiones estáticas en secuencias de vídeo-seguridad que mejore el actual estado del arte en entornos altamente concurridos y sea capaz de operar en tiempo real. El objetivo inicial se divide en los siguientes sub-objetivos:

1. Estudio del estado del arte actual: En esta etapa se van a estudiar las diversas propuestas existentes para la detección objetos de interés. Para ello se va a partir de los algoritmos disponibles en el Grupo de Tratamiento e Interpretación de Vídeo de la Universidad Autó-

noma de Madrid (VPU Lab) para analizar a fondo, tanto en escenarios sencillos como en entornos densamente poblados, las limitaciones que presentan. El objetivo de este estudio es conocer las soluciones actuales al problema propuesto.

2. Desarrollo de un nuevo algoritmo: Esta parte engloba la elaboración de un nuevo algoritmo. Para ello se van a explorar distintas características para mejorar la robustez frente a ruido en la imagen, zonas de paso y personas cuasi inmóviles en la imagen. Además se va a utilizar información temporal, mediante la acumulación de valores de píxeles en el tiempo para incorporar robustez frente a oclusiones y otros efectos. El objetivo es desarrollar una técnica que solucionelos problemas existentes en la literatura.
3. Evaluación del algoritmo desarrollado: Se va a analizar el método desarrollado, comparándolo con técnicas relevantes del estado del arte, en secuencias de vídeo de distintas complejidades. En esta etapa se van a grabar secuencias que contengan problemas encontrados en la literatura, con el objetivo de ampliar la evaluación en escenarios altamente concurridos más allá de los *datasets* públicos existentes.

1.3. Organización de la memoria

La memoria consta de los siguientes capítulos:

- Capítulo 1: Introducción, motivación del proyecto y objetivos.
- Capítulo 2: Estudio del estado del arte en la detección de regiones estacionarias de frente en sistemas de vídeo-vigilancia.
- Capítulo 3: Algoritmo de detección de regiones estáticas propuesto.
- Capítulo 4: Resultados experimentales.
- Capítulo 5: Conclusiones y trabajo futuro.

Capítulo 2

Estado del arte

En este capítulo se estudia el estado del arte relacionado con la detección de regiones estacionarias de frente en secuencias de vídeo-vigilancia.

Este estudio se encuentra dividido en las siguientes secciones: introducción a la detección de regiones estáticas (sección 2.1), clasificación de los métodos existentes (sección 2.3), análisis de los métodos más relevantes (sección 2.4) y por último una recopilación de aspectos críticos en la detección de regiones estáticas y un análisis comparativo de la literatura existente (sección 2.5).

2.1. Introducción

Los métodos propuestos en la literatura para el análisis de secuencias de vídeo-vigilancia emplean típicamente un esquema como el mostrado en la Figura 2.1. La primera etapa consiste en la localización de aquellas regiones donde sucede algo relevante. Para ello, la mayoría de las propuestas se basan en métodos de sustracción de fondo [10][11][15], que proporcionan una máscara binaria de frente con las regiones deseadas.

A continuación se aplica una técnica para determinar qué regiones de interés permanecen estáticas. Existe una gran variedad de métodos que tratan de distinta manera las máscaras de frente para realizar las detecciones de regiones estáticas tales como acumulaciones [2][9][12] o submuestreo [7][16]. Otras emplean varios modelos de fondo y en consecuencia varias máscaras de frente [17][18]. También hay métodos que aprovechan propiedades del modelo de sustracción de fondo [8][19]. Algunos métodos recientes como [3] emplean puntos característicos para detectar regiones estáticas, evitando los problemas de la etapa de sustracción de fondo. En la sección 2.3 se lleva a cabo una clasificación más detallada.

La información obtenida de la detección de regiones estacionarias se emplea en aplicaciones de alto nivel donde se busca, entre otros, clasificaciones persona-objeto o reconocimiento de las actividades de la escena (p. ej., robo o abandono), valiéndose en ocasiones de etapas de seguimiento de objetos [6][20]. Este último aspecto no va a ser tratado en este trabajo.

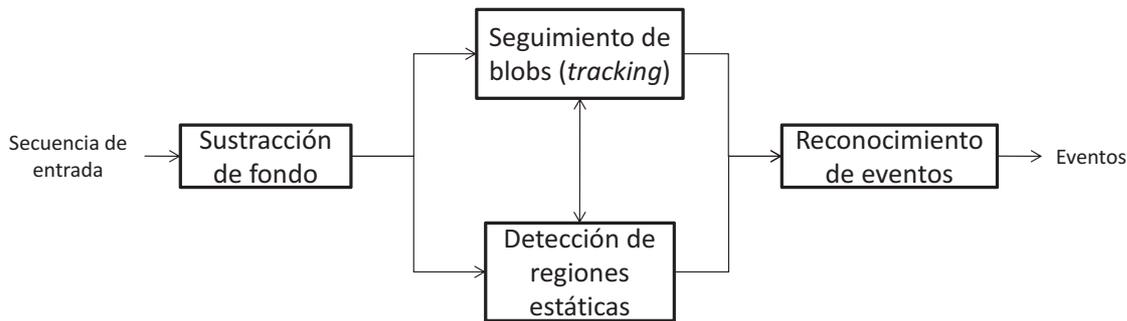


Figura 2.1: Diagrama habitual de un sistema de vídeo-seguridad.

2.2. Sustracción de fondo

Como se ha explicado, la mayoría de técnicas emplean la sustracción de fondo como medio para obtener las regiones de interés. Este proceso consiste en obtener una máscara de regiones de frente mediante algún tipo de comparación entre la imagen bajo análisis (I_t) y un modelo de fondo (B_t):

$$FG_t = f(I_t, B_t), \quad (2.1)$$

Tradicionalmente [8][17][21] se emplean modelos de caracterización de los píxeles de fondo basados en el método de Mezcla de Gaussianas (MoG) [11], pues a la hora de modelar el fondo, este tiene en cuenta posibles cambios de iluminación en la escena, objetos moviéndose lentamente y el ruido introducido por la cámara.

No obstante, ésta y otros métodos más recientes basados en una adaptación del modelo de fondo [15][22], condicionan su buen comportamiento frente a factores fotométricos (cambios de iluminación, camuflajes, sombras y reflejos) a la absorción por parte del fondo tanto de regiones no deseadas como de regiones de interés donde se encuentran objetos estacionarios (ver Figura 2.2). Además una adaptación correcta no siempre se consigue en entornos densamente poblados donde concurren numerosas sombras y oclusiones (ver Figura 2.2).

En definitiva, esta etapa es crítica pues la mayor parte de los métodos del estado del arte dependen de ella. Esto provoca que no se produzca una rápida adaptación a las condiciones de la escena manteniendo los objetos de interés y en consecuencia se condiciona enormemente el rendimiento del posterior análisis de regiones estáticas.



Figura 2.2: Ejemplos de sustracción de fondo del algoritmo [15]. La primera columna muestra un primer instante de una secuencia, la segunda columna muestra su máscara de frente y la tercera columna es una máscara de frente en un instante posterior (se detalla cada instante para cada fila a continuación). Primera fila: Al ir actualizando el fondo, se solucionan problemas ocasionados por cambios de iluminación en la máscara de frente de la tercera columna (*frame* 563), que sí aparecen en la máscara extraída en *frames* previos (segunda columna, *frame* 393). Segunda fila: Muestra como un objeto abandonado (columna primera, *frame* 2459) es absorbido con el paso del tiempo (columna tercera, *frame* 4421). Tercera fila: La segunda columna representa la máscara extraída con numerosos errores del *frame* de la primera columna (6398). La tercera columna (*frame* 7368) muestra como el algoritmo no es capaz de adaptarse a los cambios rápidos que tienen lugar en entornos altamente concurridos debidos a cambios de iluminación rápidos por las numerosas sombras originadas por las personas.

2.3. Clasificación de métodos de detección de regiones estáticas

En esta sección se describe una clasificación de métodos de detección de regiones estáticas más recientes y relevantes. A diferencia de la clasificación empleada en [14], que divide los métodos en

función de si emplean uno o varios modelos de fondo y en cada categoría distingue entre métodos con submuestreo o con análisis *frame a frame*, se ha decidido replantear esta clasificación para reflejar el número de mecanismos, adicionales a la sustracción de fondo, empleados. En esta clasificación, en general, son los métodos más recientes las que combinan mayor número de técnicas y en consecuencia obtienen un mayor rendimiento. Por tanto, se tienen dos categorías en función de si utilizan en el análisis una o varias características (la gran mayoría basados en una etapa previa de sustracción de fondo):

- Mono-característica: Esta categoría incluye técnicas en las que, tras determinar las zonas de frente, se aplica un método de detección de regiones estáticas sencillo. Algunos de los siguientes métodos son:
 - Métodos basados en técnicas sencillas de sustracción de fondo, seguidos una etapa de seguimiento de objetos. Estos métodos se conocen como métodos clásicos [13][23][24].
 - Métodos que emplean una acumulación temporal de regiones de frente a partir de un modelo de sustracción de fondo [2][12], para después umbralizar la acumulación y obtener las regiones estacionarias.
 - Métodos que emplean submuestreo de máscaras de frente [16], para después combinar las máscaras calculadas.
 - Métodos basados en propiedades del modelo de sustracción de fondo [19], donde se modelan el frente, fondo y regiones estáticas con las diferentes gaussianas empleadas.
 - Métodos que emplean dos modelos de fondo actualizados a distintas velocidades [17][18], de manera que a partir de las dos máscaras de frente se establecen diversas hipótesis que, tras algún tipo de modelado llevan a la detección estática buscada.
- Multi-característica: esta categoría engloba métodos que combinan diferentes análisis para detectar regiones estacionarias solucionando problemas de la sustracción de fondo. Se pueden encontrar distintos métodos, varias de ellas surgidas de la evolución de métodos mono-característica:
 - Métodos que emplean acumulación temporal en la que añaden información a nivel de región o de bordes [9][25], para evitar falsos positivos en la sustracción de fondo por cambios de iluminación y detecciones fantasma. La acumulación se umbraliza para detectar regiones estáticas. Estas técnicas son una evolución de la técnica mono-característica propuesta en [12].
 - Métodos que emplean submuestreo y añaden información de movimiento [7] para mejorar el rendimiento en entornos densamente poblados. Esta técnica es una evolución de la técnica mono-característica propuesta en [16].

- Métodos que emplean dos modelos de fondo y añaden al análisis información de movimiento u otras técnicas para verificar la detección [6][26]. Estas técnicas son una evolución del método mono-característica propuesto en [17].
- Métodos basados en propiedades del modelo de sustracción de fondo, utilizando las diferentes Gaussianas de un modelo MoG para modelar fondo, regiones estáticas y regiones en movimiento, que añaden información para hacer frente a cambios de iluminación y detecciones fantasma [8][21][27]. Estas técnicas son una evolución de la técnica mono-característica propuesta en [19].
- Métodos que emplean filtrados secuenciales a nivel de *blob* [1], considerando información de movimiento, temporal, de apariencia y de bordes para establecer las detecciones estacionarias a partir de una máscara de frente.
- Métodos que emplean puntos característicos para la detección de vehículos aparcados ilegalmente [3], en los que se prescinde de una sustracción de fondo precisa y se establecen puntos característicos de fondo y frente (estáticos y dinámicos) para su posterior acumulación espacio-temporal para generar las regiones estacionarias.

2.4. Métodos de detección de regiones estáticas seleccionados

En este apartado se describen los métodos del estado del arte más recientes, llevando a cabo un análisis de sus virtudes y defectos. Este análisis permitió orientar las líneas a seguir hasta sintetizar un nuevo algoritmo, propuesto en el capítulo 3 de la memoria.

2.4.1. Acumulación de máscaras de frente

Este método es propuesto en [12] donde se busca detectar regiones estáticas mediante la acumulación de máscaras de frente obtenidas de manera consecutiva. Para ello, el algoritmo construye una imagen intermedia $S(x, y)$, donde el valor de cada píxel (entre 0 y 255) determina si dicho píxel es estacionario, o no. Un valor de 0 indica que el píxel pertenece al fondo de la imagen, y un valor de 255 indica que nos encontramos ante un píxel de una posible región estática. Al comenzar el análisis del vídeo, todos los píxeles de la imagen intermedia S se inician a 0. Para cada nueva imagen, cada píxel de la imagen S se actualiza basándose en una máscara de frente, obtenida de una comparación de la imagen bajo análisis $I(x, y)$ con el modelo de fondo $B(x, y)$ (algoritmo de sustracción de fondo). Dos parámetros controlan la actualización de la imagen $S(x, y)$: $C(x, y)$ y $D(x, y)$. $C(x, y)$ toma valor 1 cuando el píxel pertenece al frente de la imagen y 0 cuando pertenece al fondo. $D(x, y)$ tiene un comportamiento inverso a $C(x, y)$. Las ecuaciones de actualización son las siguientes:

$$S(x, y) = S(x, y) + C(x, y) \left(\frac{255}{t \times \text{framerate}} \right), \quad (2.2)$$

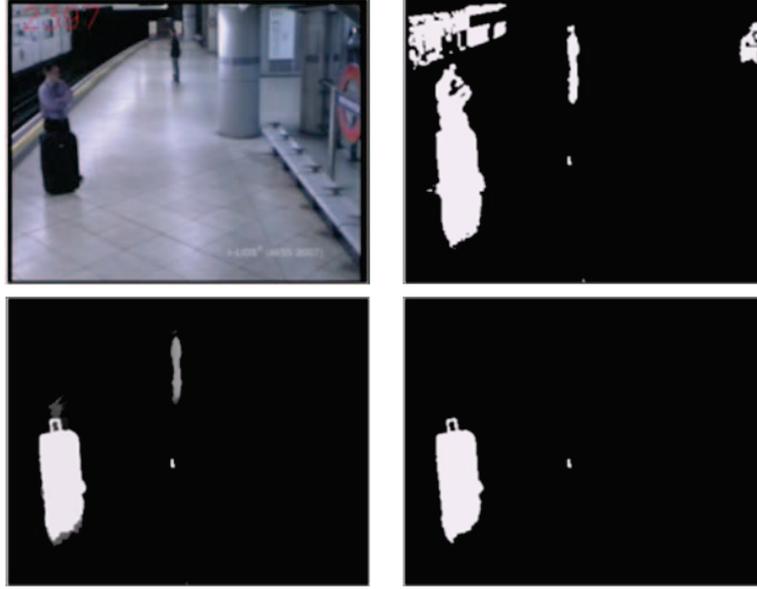


Figura 2.3: Ejemplo del algoritmo [12] basado en acumulación de máscaras. Primera fila: *frame* bajo análisis (izquierda) y máscara de frente (derecha). Segunda fila: Acumulación (izquierda) y $SFG_{Acc}(x, y)$ (derecha).

$$S(x, y) = S(x, y) - rD(x, y)\left(\frac{255}{t \times framerate}\right), \quad (2.3)$$

donde t es el tiempo necesario para declarar un objeto estacionario, r es un número positivo para ponderar el efecto de la disminución, consiguiendo así que cuando un píxel candidato a ser estacionario deja de pertenecer al frente (y pasa a pertenecer al fondo), el valor de $S(x, y)$ en dicho píxel disminuya rápidamente. Los píxeles que pertenezcan a un objeto estacionario harán que el valor de $S(x, y)$ aumente imagen a imagen hasta llegar al máximo valor posible de 255 en un tiempo t . Por último, se realiza una umbralización de la imagen en tonalidad de grises, $S(x, y)$, para decidir qué píxeles pertenecen a regiones estáticas, obteniendo así una imagen binaria $SFG_{Acc}(x, y)$ con las regiones estáticas a 1 (ver Figura 2.3):

$$SFG_{Acc}(x, y) = \begin{cases} 1 & \text{si } S(x, y) > \tau \\ 0 & \text{resto} \end{cases}, \quad (2.4)$$

donde τ es el umbral que controla la aparición de regiones estáticas, cuyo máximo y valor ideal para respetar el tiempo de comienzo de las alarmas es 255.

Es importante hacer varias observaciones de este método:

- Es robusto a oclusiones de objetos que sufren camuflajes siempre que el peso de la disminución r no sea excesivamente grande, pues llevaría a perder aquellos píxeles que mo-



Figura 2.4: Ejemplos de errores del algoritmo [12]. Primera fila: *Frame* bajo análisis (izquierda), máscara de frente (centro) y $SFG_{Acc}(x, y)$ (derecha) con errores en la sustracción de fondo causados por sombras que cambian la iluminación en diferentes zonas (bordillo del andén o señal del metro) y que son acumulados, llevando así a superar el umbral de detección τ . Segunda fila: *frames* 3542 (izquierda) y 3708 (centro) que muestran una zona de paso y máscara estática del *frame* 3708 marcando con un círculo azul los *blobs* que son falsas detecciones por movimiento.

mentáneamente sufren el error mencionado. Es necesaria también una ligera rebaja de τ respecto al máximo de 255, para poder tolerar esos descensos.

- Supone que el modelo de fondo es correcto y no lleva a cabo ningún mecanismo para tratar errores en el mismo (cambios de iluminación, fantasmas, sombras, etc) (ver Figura 2.4).
- No es robusto al movimiento continuado (ver Figura 2.4), pues pueden ocurrir detecciones continuadas de frente debido al movimiento continuado (zona de paso).

2.4.2. Submuestreo de máscaras de frente

Este método [16] emplea una técnica basada en el submuestreo de máscaras binarias de frente para localizar las regiones estáticas dentro de las mismas (ver Figura 2.5).

En [16] se indica 6 como el número de máscaras binarias muestreadas durante el tiempo fijado como estático, es decir, que se obtienen 6 máscaras binarias de frente $\{M_1, \dots, M_6\}$, cuyos píxeles con valor 1 indican que pertenecen al frente de la imagen, y los píxeles con valor 0 indican que pertenecen al fondo. Después, se realiza una multiplicación lógica píxel a píxel de

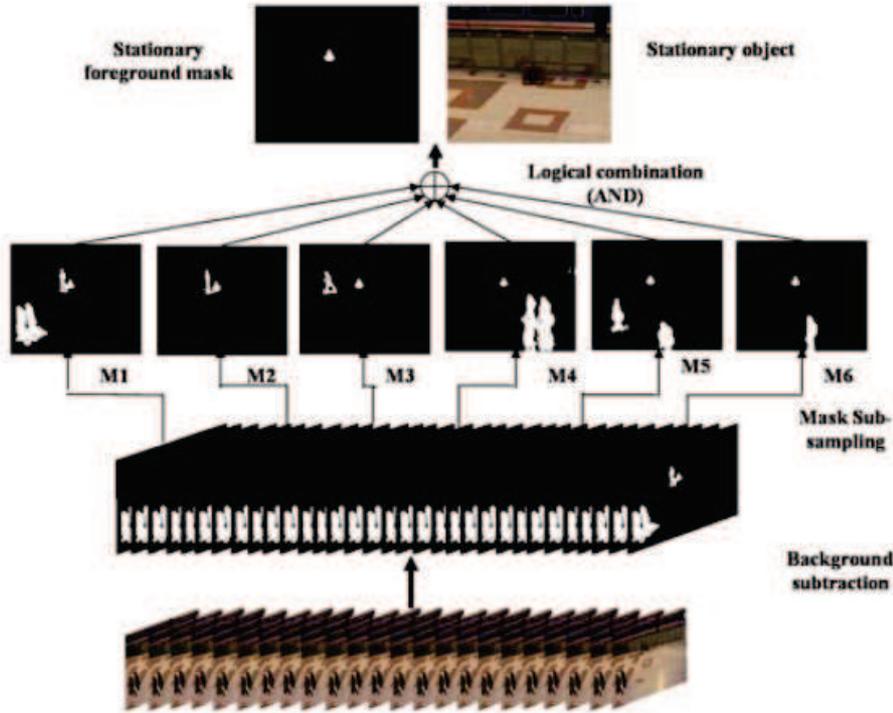


Figura 2.5: Esquema de muestreo y combinación para obtener regiones estacionarias propuesto por [16] (extraído de [14]).

cada una de las 6 máscaras binarias de frente obtenidas, para calcular así la imagen final de objetos estacionarios $SFG_{Sub}(x, y)$, que se obtiene del siguiente modo:

$$SFG_{Sub} = M_1 \times M_2 \times M_3 \times M_4 \times M_5 \times M_6, \quad (2.5)$$

A continuación, se aplica una etapa posterior de filtrado a la máscara SFG_{Sub} para eliminar posible ruido impulsivo incluido en dicha máscara. Los valores restantes en la máscara que tengan valor 1 pertenecen a regiones estáticas.

En la Figura 2.5 se observa la secuencia de imágenes a analizar, sus máscaras binarias, el submuestreo de 6 de ellas y la máscara $SFG_{Sub}(x, y)$ que surge de su combinación y que incluye las regiones estacionarias (en este caso un objeto).

Una vez explicado su funcionamiento, es importante tener en cuenta varios aspectos:

- Es robusto a oclusiones, pero esta característica depende de que el instante de muestreo de alguna de las 6 máscaras no incluya errores de camuflaje.
- Supone que el modelo de fondo es correcto y no lleva a cabo ningún mecanismo para tratar errores en el mismo (cambios de iluminación, fantasmas, sombras, etc).

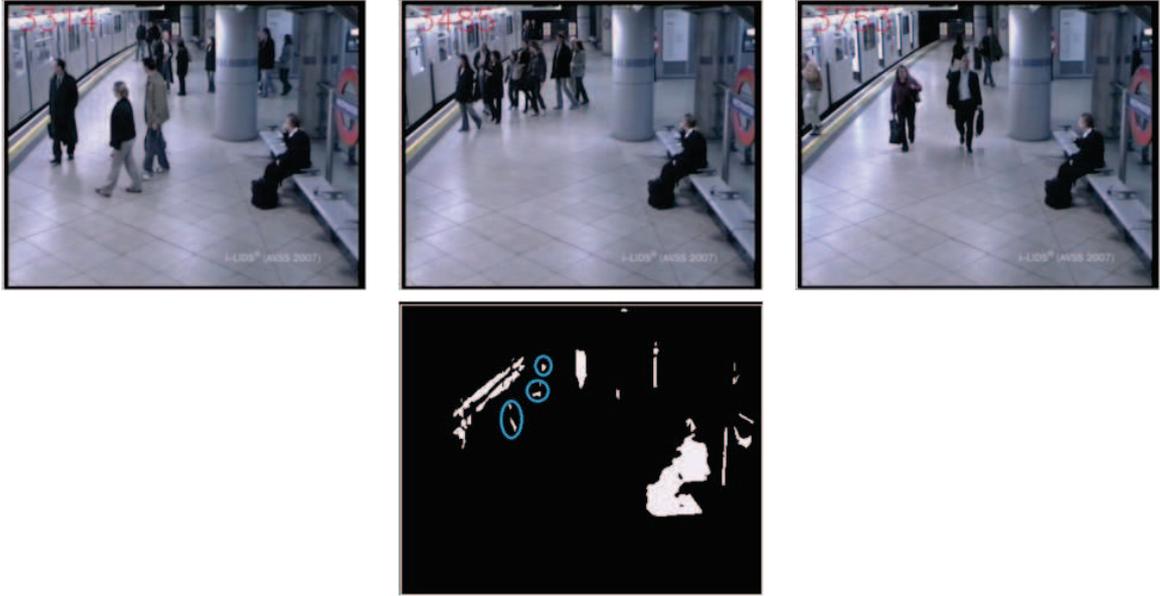


Figura 2.6: Primera fila: *Frames* 3314, 3485 y 3753 que muestran una zona de paso. Segunda fila: SFG_{Sub} del *frame* 3753 marcando con un círculo azul los *blobs* que son falsas detecciones originadas por muestrear en una zona de paso donde el frente está activo continuamente.

- La aleatoriedad del instante de muestreo es una desventaja importante, pues provoca un descenso importante del rendimiento en zonas de paso (habituales en entornos densamente poblados), donde el tránsito de personas provoca numerosos falsos positivos al combinar las máscaras (ver Figura 2.6).

2.4.3. Submuestreo de máscaras de frente y movimiento

En este método [7] se propone analizar características de la señal de vídeo en diferentes instantes de tiempo (sub-muestreo) pero añadiendo información de movimiento para afrontar los problemas del método descrito en 2.4.2 con las zonas de paso (ver Figura 2.7).

Primeramente se analiza la persistencia de una región de frente en distintos instantes temporales, obteniendo $SFG_{Sub}(x, y)$ al igual que en la sección 2.4.2. Posteriormente, se extrae el movimiento de dichas regiones en los instantes seleccionados en el muestreo anterior. Para analizar el movimiento de las regiones estáticas se utiliza la técnica de *frame-difference* (ΔI), que consiste en restar dos imágenes entre sí, dando como resultado un estimador de movimiento en dicha imagen, siendo ΔI_t la imagen resultante de la diferencia en el instante t , I_t la imagen en el instante t , y I_{t-1} la imagen en el instante $t-1$. Destacar que para realizar esta diferencia, primero se ha de pasar de una imagen perteneciente al espacio de color RGB a una imagen en grises (que toma valores de 0 a 255). Tras realizar esta resta, dicha imagen se umbraliza píxel a píxel para

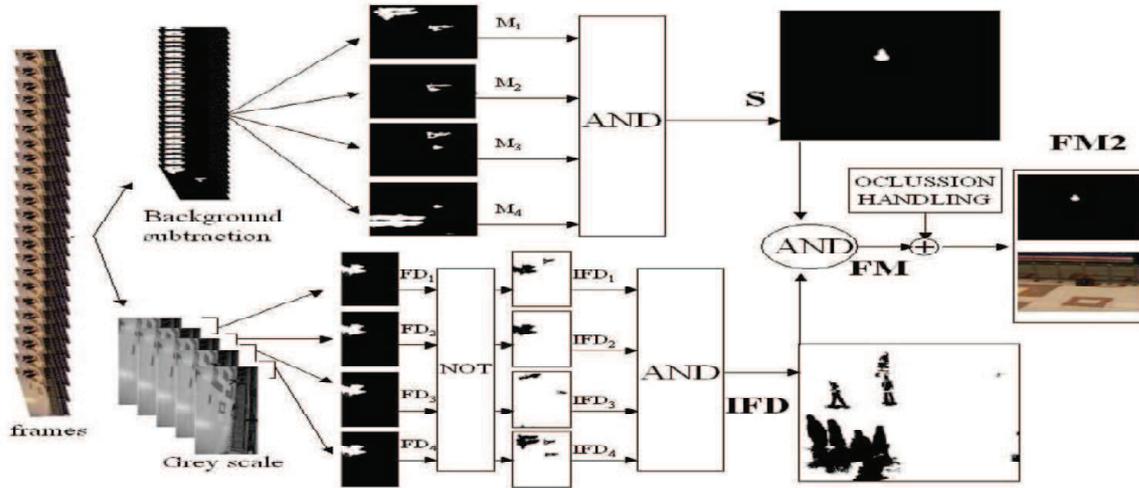


Figura 2.7: Esquema de funcionamiento del algoritmo basado en muestreo de máscaras de frente y movimiento, extraído de [7].

obtener una máscara binaria:

$$\Delta I_t = I_t - I_{t-1}, \quad (2.6)$$

$$\Delta I_t(x, y) = 1 \quad \text{si} \quad \Delta I_t(x, y) > \tau, \quad (2.7)$$

$$\Delta I_t(x, y) = 0 \quad \text{si} \quad \Delta I_t(x, y) < \tau, \quad (2.8)$$

Esta máscara devuelve con valor 1 las regiones en movimiento, y con valor 0 las regiones que no han sufrido movimiento, tras establecer un umbral τ ([7] propone $\tau = 20$). Por lo tanto, para destacar las regiones estáticas donde no hay movimiento, se realiza una inversión de la máscara (símbolo \sim) (ver Figura 2.7):

$$IFD_t(x, y) = \sim (\Delta I_t(x, y)), \quad (2.9)$$

Por tanto, el resultado de ambos análisis son dos máscaras binarias indicando las propiedades deseadas. El último paso consiste en realizar la multiplicación a nivel de píxel de $SFG_{Sub}(x, y)$ por la máscara $IFD(x, y)$ obtenida al realizar el método de *frame-difference* inverso. Así se obtiene la máscara final de regiones estáticas $SFG_{IFD}(x, y)$ (FM en la Figura 2.7):

$$SFG_{IFD}(x, y) = IFD(x, y) \times SF_{Sub}(x, y), \quad (2.10)$$

El objetivo de añadir un análisis de movimiento es la eliminación de regiones donde existe frente



Figura 2.8: Ejemplos del algoritmo [7]. De izquierda a derecha, *frame* bajo análisis, SFG_{Sub} , IFD y SFG_{IFD} . Primera fila: La detección por movimiento que aparece en SFG_{Sub} , no aparece en SFG_{IFD} gracias a la información de movimiento de IFD . Segunda fila: El movimiento capturado en IFD , que ha tenido lugar en *frame* previos delante de la maleta, provoca su eliminación de la máscara SFG_{IFD} al no tener blobs que recuperar en el manejo de oclusiones. Tercera fila: Las detecciones por movimiento que aparecen en SFG_{Sub} , son removidas en el resultado final (SFG_{IFD}) gracias al filtrado de movimiento que proporciona IFD .

activo pero a su vez se está produciendo movimiento (ver primera y tercera fila de la Figura 2.8). Una vez se tiene la máscara estática, es necesario incorporar al esquema un manejo de oclusiones para hacer frente a situaciones en las que haya gente que se cruza con el objeto de interés en el instante de muestreo, originando así que la máscara IFD marque la zona como movimiento y en consecuencia, provocando así la desaparición del objeto estático de la máscara SFG_{IFD} al llevar a cabo la combinación lógica. Para solucionar este aspecto se lleva a cabo un análisis a nivel de *blob* en el que se recupera una región estática cuando en un instante se produce la situación descrita y en el instante anterior SFG_{IFD} tenía el *blob* marcado como estático.

Una vez expuesto su funcionamiento es conveniente hacer varias observaciones sobre este método:

- Aunque añadir la máscara IFD permite eliminar falsos positivos producidos por SFG_{IFD} y por tanto tratar situaciones con mucho movimiento (frecuentes en entornos densamente poblados), añade un aspecto negativo y es que si un objeto que aún no ha sido detectado sufre oclusiones continuas (IFD indica movimiento en la zona de interés por la aleatoriedad

del muestreo sin poder detectar no-movimiento tras la oclusión), cuando se cumpla el tiempo estático el objeto no será detectado pues la recuperación de oclusiones solo se aplica tras una detección inicial (ver primera fila de Figura 2.8).

- Al igual que los métodos anteriores, supone un modelo de fondo correcto y no realiza ningún mecanismo para tratar errores en el mismo (cambios de iluminación, fantasmas, sombras, etc) (ver zona de la derecha, señal de metro, en la segunda fila de la Figura 2.8).

2.4.4. Dos modelos de fondo actualizados a diferentes velocidades

En este método, basado en el estudio realizado en [17], se propone el uso de varios modelos de sustracción de fondo, en lugar de uno sólo como se ha visto en los anteriores métodos. Concretamente se propone la utilización de dos modelos MoG que difieren en la velocidad de actualización del fondo de escena. Para detectar regiones estáticas, lo normal es modelar la imagen de fondo con un solo modelo matemático, pero esta técnica decide utilizar dos modelos matemáticos: un modelo a largo plazo, que se actualiza de forma lenta, es decir, que no incorpora rápidamente a su modelo de fondo (B_L) los objetos que permanecen parados y un modelo a corto plazo que se actualiza más rápidamente, es decir, que incorpora más rápidamente a su modelo de fondo (B_C) los objetos que permanecen estáticos.

Para cada imagen, se estima el modelo a largo plazo y el modelo a corto plazo comparando la imagen actual I con ambos modelos de fondo, B_L y B_C . Al realizar esa comparación, se obtienen dos máscaras binarias de frente F_L y F_C , donde $F(x, y) = 1$ indica que el píxel perteneciente a la posición (x,y) no se corresponde con el fondo de la imagen. En este método, dependiendo del valor de F_L y F_C , pueden ocurrir 4 hipótesis, que se detallan a continuación (ver Figura 2.9):

- $F_L = 1$ y $F_C = 1$, se considera que el píxel (x,y) corresponde a un objeto en movimiento debido a que $I(x, y)$ no se asemeja a ninguno de los dos modelos de fondo.
- $F_L = 1$ y $F_C = 0$, se considera que el píxel (x,y) corresponde a una región estacionaria, pues $I(x, y)$ se asemeja a $B_C(x, y)$ (que tiene el objeto absorbido a corto plazo), pero no se asemeja a $B_L(x, y)$ que aún no ha absorbido el objeto estático.
- $F_L = 0$ y $F_C = 1$, se considera que el píxel (x,y) es un píxel del fondo de la escena que ha sido ocluido anteriormente. Es decir, un objeto ha ocluido el fondo provocando la absorción por parte de $B_C(x, y)$, pero no ha permanecido el tiempo suficiente como para ser considerado estático y ser absorbido por $B_L(x, y)$, provocando, al ser removido de la escena, la situación descrita.
- $F_L = 0$ y $F_C = 0$, se considera que el píxel (x,y) pertenece al fondo de la escena, debido a que se asemeja a ambos modelos de fondo.

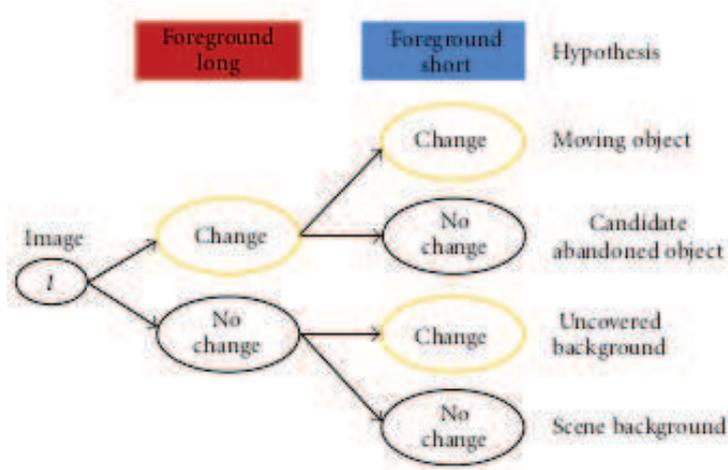


Figura 2.9: Hipótesis establecidas en función de las dos máscaras de frente (F_L y F_C) (Extraído de [17]).

La decisión de píxel estático se lleva a cabo sintetizando una imagen acumulación $E(x, y)$ a partir de las hipótesis anteriores:

$$E(x, y) = \begin{cases} E(x, y) + 1, & F_L(x, y) = 1 \wedge F_C(x, y) = 0 \\ E(x, y) - k, & F_L(x, y) \neq 1 \vee F_C(x, y) \neq 0 \\ \max_e, & E(x, y) > \max_e \\ 0, & E(x, y) < 0 \end{cases}, \quad (2.11)$$

La finalidad de esta imagen es controlar el tiempo necesario para considerar un píxel como estático y eliminar ruido del resultado final. Un píxel será estático cuando:

$$SFG_{DB}(x, y) = \begin{cases} 1 & \text{si } E(x, y) = \max_e \\ 0 & \text{resto} \end{cases}, \quad (2.12)$$

donde \max_e , es el valor máximo de $E(x, y)$ y se obtiene al restar al tiempo mínimo establecido para considerar un píxel estacionario (en *frames*), el tiempo (en *frames*) que necesita un píxel permanecer parado antes de ser absorbido por el modelo B_C , pues mientras no ha sido absorbido la situación es $F_L = 1$ y $F_C = 1$ y no se incrementa $E(x, y)$.

Esta técnica tiene por tanto un modelo de fondo a corto plazo (B_C) que se adapta rápido a los cambios existentes en la escena y un modelo a largo plazo (B_L) que se adapta más lentamente (ver Figura 2.10), en función de las tasas que se decidan ([17] utiliza una tasa a corto plazo 30 veces superior a la tasa a largo plazo).



Figura 2.10: En la primera fila se tienen el *frame* bajo análisis, F_C y B_C , y en la segunda fila F_L , B_L y la máscara de regiones estáticas. Se puede ver como la maleta que permanece inmóvil ha sido absorbida por B_C pero no por B_L , razón por la cual se incluye como región estática, tal y como se establece en la Figura 2.9.

Si bien esta técnica es capaz de adaptarse a los cambios que se producen en el fondo una vez que B_L ha absorbido la situación correcta, es importante tener en cuenta algunos aspectos:

- Aunque, tras un tiempo, es capaz de adaptarse a la situación actual y recuperarse de falsos positivos (al absorber B_L la situación), esto acarrea la absorción de los objetos estáticos por parte de B_L y en consecuencia, la desaparición (no deseada) de sus alarmas correspondientes (ver Figura 2.11).
- Además son habituales cambios de iluminación temporales en la escena, en los que no transcurre el tiempo necesario para que B_L absorba la situación, originando así la aparición de falsos positivos (ver Figura 2.12). Si el cambio de iluminación perdura será absorbido por B_L , pero mientras tanto falsos positivos aparecerán en SFG_{DB} .
- Bajo ciertas condiciones de configuración del algoritmo MoG (considerando solo la primera Gaussiana como modelo de fondo), tal y como emplean otros autores [18], el algoritmo no es capaz de tratar objetos estáticos temporales. Esto es debido a que cuando una región estática es absorbida por ambos modelos de fondo, es decir, que se da la situación $F_L = 0$



Figura 2.11: Misma ordenación que la Figura 2.10. El objeto que se observa en el *frame* bajo análisis (abandonado en torno a 9500 *frames* antes) ya no se detecta pues los modelo a corto y largo plazo lo han absorbido (ver B_C y B_L , en este último el color es tenue, debido a que el objeto no es el único modelo de fondo).

y $F_C = 0$ y después esa región es removida, acabará llegándose a la situación $F_L = 1$ y $F_C = 0$ (cuando B_C absorba el fondo correcto), produciéndose así una detección fantasma, pues el modelo a largo plazo conserva aún en esa posición el objeto estático que ha sido removido. Si se tienen en cuenta varias gaussianas, se tienen en cuenta varios modelos de fondo que evitan la detección fantasma.

- Arrastra los mismos problemas que [12] en cuanto a robustez a oclusiones, pues lleva a acabo una acumulación similar. Por tanto, para manejar este problema, se deberá permitir detecciones ligeramente por debajo de max_e y no fijar k con un valor muy elevado (análogamente a como ocurría en el método del apartado 2.4.1 con 255 y r).
- El algoritmo no hace frente a aquellas situaciones en las que una posición espacial tiene una detección cuasi-continua de frente ocasionada por movimiento y que por tanto puede llegar a originar falsos positivos si el movimiento es lo suficientemente intenso.

En resumen, la técnica es capaz de adaptarse a las condiciones de iluminación tras un tiempo (con los aspectos positivos que conlleva en cuanto a detecciones fantasma o cambios de iluminación) a costa de dejar de detectar objetos que pueden ser de interés. Además, sí B_L tiene errores, en el resultado final contendrá falsas detecciones mientras B_C tiene el fondo correcto y B_L no.

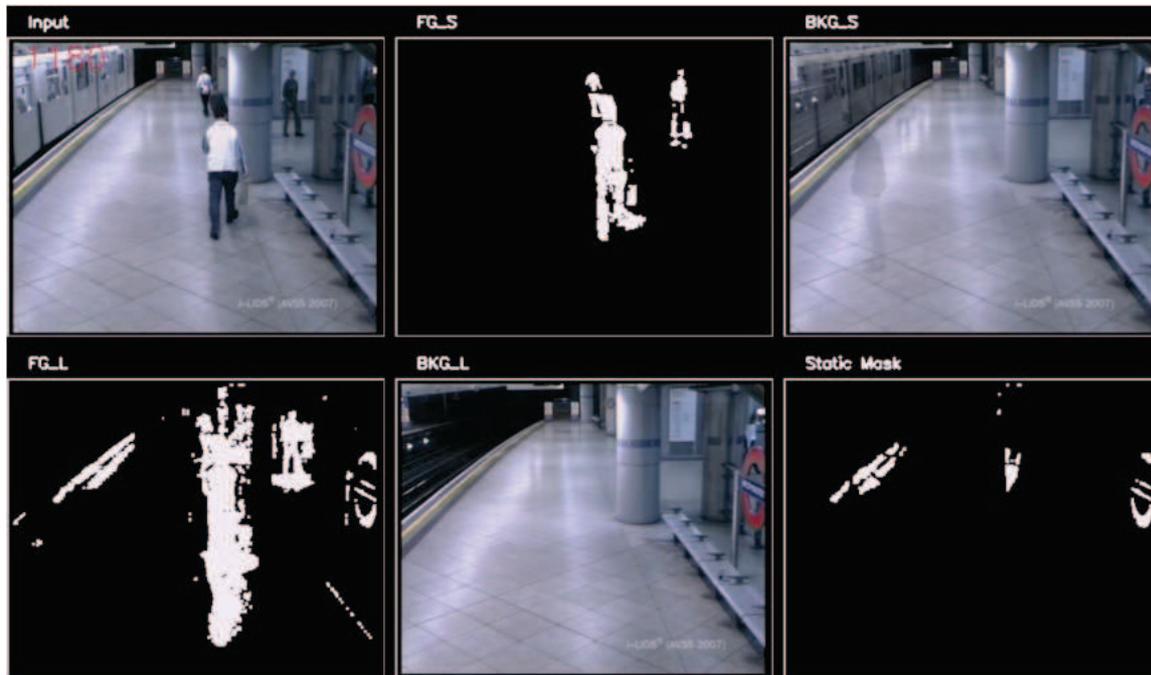


Figura 2.12: Misma ordenación que la Figura 2.10. Se puede apreciar como B_C ha absorbido los cambios de iluminación del borde del andén, pero B_L no, con lo que al permanecer el cambio de iluminación durante el tiempo estático establecido, se origina una falsa detección en SFG_{DB} .

2.4.5. Dos modelos de fondo actualizados a diferentes velocidades combinado con una máquina de estados

Este método, presentado en [18], extiende al anterior pero cambia la manera de tratar absorciones de regiones estáticas en el modelo de fondo. Al igual que en el modelo anterior se obtienen dos máscaras binarias de frente F_L y F_C (correspondientes a los modelos a largo y corto plazo) donde $F(x, y) = 1$ indica que el píxel perteneciente a la posición (x,y) no se corresponde con el fondo de la imagen. En este método, dependiendo del valor de F_L y F_C , en lugar de llevar a cabo una acumulación, se guarda para cada píxel el último valor de fondo conocido y se establecen una serie de transiciones en una FSM (ver Figura 2.13) para evitar que la absorción de una región estática por ambos modelos suponga la pérdida de regiones estáticas y una posterior detección fantasma cuando el objeto sea removido de la escena (como puede ocurrir en [17]):

Los estados BG (*Background*), MP (*Moving pixel*), PAP (*Partial absorbed pixel*), UBG (*Uncovered background region*) y AP (*Absorbed pixel*) modelan las situaciones ya manejadas por las hipótesis de 2.4.4 (ver Figura 2.9). Una vez alcanzado el estado AP (píxel absorbido por ambos modelos) es donde tiene lugar la gran diferencia con el método anterior, pues mientras que en 2.4.4 al absorber los dos modelos el objeto desaparece, continúe o no en la escena, aquí

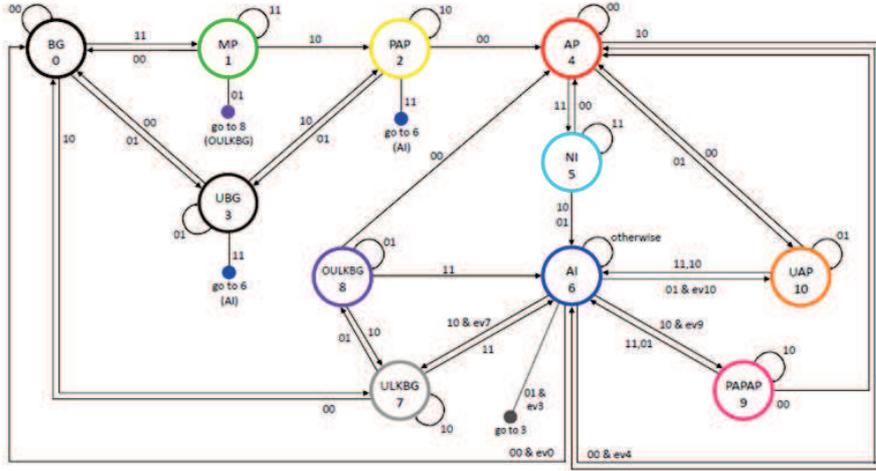


Figura 2.13: Máquina de estados (FSM) que modela los píxeles en función de su histórico de hipótesis (extraído de [18]), donde p. ej., 10 define $F_L = 1$ y $F_C = 0$.

se modela la situación con una serie de transiciones que permiten continuar con la detección y evitar una detección fantasma en caso de ser removido (ver Figura 2.14). Si estando en AP tiene lugar una situación 10^1 ó 01 , se pasa a NI (*New indetermination*), pueden estar ocurriendo varias situaciones:

1. Que se tenga un objeto ocluyendo y que después desaparezca, volviendo así a AP.
2. Que se tenga un objeto ocluyendo y que permanezca, siendo así absorbido por B_C y en consecuencia pasando a AI (*Absorbed indetermination*).
3. Que se haya removido el objeto absorbido (quedando el fondo libre de oclusión), pasándose así a AI cuando B_C absorba la nueva situación.

Por tanto, en AI tendremos que determinar si B_C describe o no el fondo correctamente, es decir, si está ocurriendo 3 o en su defecto 1 ó 2 (los eventos que aparecen en la Figura 2.14 se refieren a esta decisión: fondo/frente). Para tomar esta decisión es necesario consultar el último valor conocido de fondo, de forma que si coincide se modelará el píxel a través de los estados ULKBG (*Uncovered last known background*) y OULKBG (*Occluded uncovered last known background*) y en caso contrario mediante PAPAP (*Partially absorbed pixel over absorbed pixel*) o por UAP (*Uncovered absorbed pixel*).

Es importante resaltar que las transiciones salientes del estado AI dependen de una comparación con un modelo de fondo guardado (que se supone es correcto) y que sufre actualización siempre

¹Para abreviar, se va a hacer referencia al valor de F_L y F_C , que origina la transición al estado descrito, directamente con sus valores. En este caso 10 hace referencia a que $F_L = 1$ y $F_C = 0$ provocan la transición al estado *New Indetermination*.



Figura 2.14: Las 5 imágenes muestran, de izquierda a derecha, el paso del tiempo para un objeto estático. Inicialmente el objeto es abandonado (color verde-estado MP), después es absorbido a corto plazo (color amarillo-estado PAP) y luego se absorbe también a largo plazo (color rojo-estado AP). Cuando el objeto es removido (color azul-estado NI) y tras absorber B_C la situación actual (visionado del fondo) y establecerse en AI que los píxeles donde estaba el objeto se corresponden con el fondo, entonces se pasa al estado ULKBG (color gris) que en ningún caso será considerado un estado estático (evitando así una detección fantasma).

que se realiza una transición desde el estado BG. La decisión estática es:

$$SFG_{DB_FSM}(x, y) = \begin{cases} 1 & \text{si } (x, y) \in \{PAP, AP, AI\} \\ 0 & \text{resto} \end{cases}, \quad (2.13)$$

No obstante los autores mencionan los estados estáticos como ajustables a diferentes sensibilidades, siendo la mínima agrupación de estados estáticos $\{AP\}$ y la máxima $\{PAP, AP, NI, AI, OULBKG, PAPAP, UAP\}$.

Una vez expuesto su funcionamiento, es importante hacer varias observaciones:

- Si, al igual que hace esta técnica, se toma como premisa en una técnica con un solo modelo de fondo que su inicialización es correcta y no se actualiza dicho fondo, se consigue también evitar cualquier detección fantasma producida por objetos estáticos temporales. Sin embargo, lo que sí aporta esta técnica (al emplear dos modelos de fondo y manejar la absorción de objetos por parte de ambos), es la recuperación a errores en el modelo de fondo sin perder las detecciones estáticas correctas. No obstante, los errores que tengan los algoritmos de sustracción de fondo (MoG) en F_L y F_C por cambios de iluminación, se van a arrastrar con independencia de que el modelo de fondo guardado sea el correcto (con una iluminación diferente), pues al realizar la comparación en AI se va a establecer que el *frame* actual difiere del fondo.
- La comparativa con el fondo guardado es crucial para que se haga efectiva la adaptación. En otras palabras, si hay un cambio de iluminación el algoritmo debe ser capaz de determinar que *frame* y fondo son iguales (esta comparación no es especificada).
- Al igual que la aproximación presentada en 2.4.4, no se tratan situaciones de intenso movimiento.

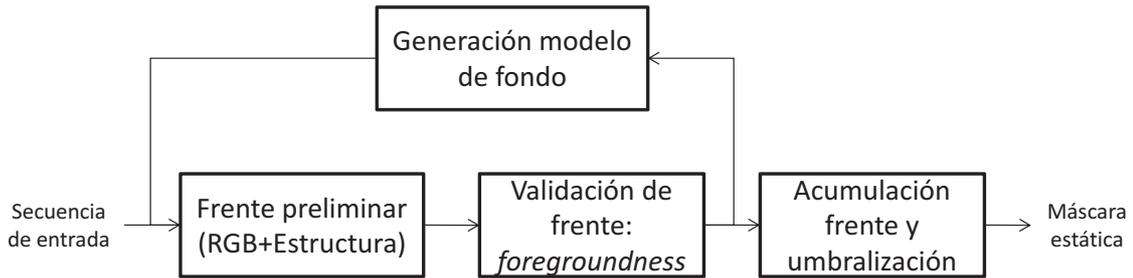


Figura 2.15: Esquema empleado por [9] para la detección de regiones estacionarias.

2.4.6. Análisis a nivel de región para validar regiones de frente

En el método propuesto en [9], se propone utilizar información a nivel de región para incrementar la robustez frente a detecciones fantasma y cambios de iluminación, empleando valores anteriores para determinar si los píxeles están sufriendo cambios (es decir, si hay movimiento).

El método puede dividirse en 3 partes (ver Figura 2.15): obtención del frente, filtrado de *blobs* de frente y actualización de fondo mediante una puntuación (*foregroundness*) y acumulación y umbralización para detección de las regiones estáticas.

Primero se calculan qué píxeles pertenecen al frente de la escena, para ello se combina (en cada píxel) información a nivel de píxel y de región:

- Se calcula la diferencia de color D_c , que consiste en restar los tres canales del espacio de color RGB del *frame* actual y el fondo de la imagen.
- Se calcula la diferencia estructural D_s , que incluye información de una región alrededor del píxel (R), para así proporcionar robustez a cambios locales de iluminación. Esta diferencia estructural se calcula como $D_s = 1 - S$, donde S es la estructura de semejanza entre el *frame* actual y el fondo y se calcula como $S = \max\{S^{fb}, S^{bf}\}$. S^{fb} es la puntuación de semejanza del *frame* en el fondo y S^{bf} la puntuación de semejanza del fondo en el *frame*. Se calculan ambas puntuaciones para asegurar un mejor rendimiento.

Una vez que se tienen la diferencia de color y la diferencia estructural, se calcula la diferencia híbrida como $D = D_c \times D_s$, que adquirirá valores pequeños cuando cualquiera de las dos diferencias sea pequeña. A continuación, se realiza una umbralización de la imagen D para establecer una primera proposición de máscara de frente (aún con numerosos falsos positivos por detecciones fantasma o cambios de iluminación).

A continuación el algoritmo propone dos análisis para reducir los falsos positivos presentes en la máscara:

- Eliminación de detecciones fantasma: Se realiza un análisis del gradiente a lo largo del contorno del *blob*, tanto en el *frame* actual (C^f) como en el fondo (C^b), concluyendo que si

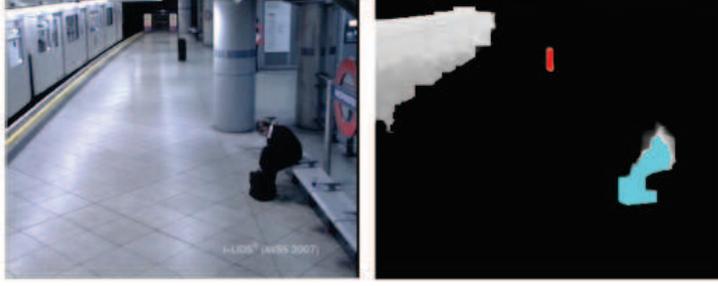


Figura 2.16: *Frame* bajo análisis (izquierda), máscara de regiones estáticas (regiones en rojo y azul).

el *blob* consigue un valor mayor en el fondo (más cambios de intensidad al tener el modelo de fondo un objeto que ya no está en la escena) que en la imagen actual ($C^b > C^f$) entonces se trata de una detección fantasma. De este análisis se extrae una puntuación F_c (mayor cuanto menor sea C^b).

- Eliminación de detecciones causadas por cambios de iluminación: Emplea información estructural, realizando el mismo análisis que D_s pero considerando R como el *blob* bajo análisis en lugar de un vecindario de píxel. De este análisis se obtiene una puntuación de frente F_b (mayor cuanto menor sea la semejanza estructural con el fondo).

La puntuación final de *foregroundness* se calcula como $F = \min\{F_c, F_b\}$ y 0 para todo píxel no perteneciente a un *blob*. Esta puntuación se emplea además para modelar la tasa de actualización del fondo, que será menor para aquellas regiones con una F mayor.

A continuación se lleva a una acumulación de F , para lo cual se define una puntuación a nivel de píxel $a(x, y)$ como:

$$a(x, y) = F(x, y) \times \delta(x, y), \quad (2.14)$$

$$\delta(x, y) = \begin{cases} 1 & \text{si } \Delta f(x, y) < t \\ 0 & \text{resto} \end{cases}, \quad (2.15)$$

donde $\Delta f(x, y) = I_t(x, y) - I_{t-1}(x, y)$ y por tanto si no hay movimiento ($\Delta f(x, y) < t$) la acumulación es positiva y en caso contrario se fija a 0. Por último tiene lugar una umbralización para determinar las regiones estáticas (ver Figura 2.16):

$$SFG_{DS}(x, y) = \begin{cases} 1 & \text{si } a(x, y) > \tau \\ 0 & \text{resto} \end{cases}, \quad (2.16)$$

Una vez explicado el funcionamiento es importante hacer varias observaciones:

- Arrastra los mismos problemas que el método propuesto en sección 2.4.1 en cuanto a robustez a oclusiones se refiere, pues lleva a cabo una operativa similar.



Figura 2.17: Esquema del algoritmo propuesto en [8].

- Hay que tener cuidado con la actualización del fondo que se haga, pues podrían absorberse objetos de interés.
- Se tratan cambios de iluminación y detecciones fantasma.
- El tratamiento que se hace del movimiento (diferencia entre *frames* adyacentes), no es eficaz en zonas de paso para detectar no-movimiento ocluido continuamente (como se va a explicar en el siguiente capítulo), provocando la pérdida de regiones estacionarias si no se realiza una correcta umbralización como la que se va a proponer en el capítulo 3.

2.4.7. Propiedades del modelo de fondo e interacción con una etapa de *tracking*

En este trabajo, propuesto en [8], sigue el esquema descrito en la Figura 2.17.

Para la sustracción de fondo se emplea el modelo MoG[11] que es adecuado para fondos donde hay objetos no estáticos (hojas de árboles en movimiento, olas, etc) cuyos píxeles varían de valor entorno a un conjunto finito de valores donde un único valor (media y varianza) no es adecuado para modelar la situación. Por tanto, con el modelo MoG se propone dar solución a este problema tratando la intensidad de los píxeles con una mezcla de k distribuciones Gaussianas (donde k es un número pequeño, frecuentemente de 3 a 5) definidas cada una por una media, una varianza y un peso (cuyo valor determina la cantidad de distribución k -ésima que modela dicho píxel como fondo en cada instante). En este caso se utilizan tres Gaussianas, donde la primera modela el fondo, la segunda se emplea para modelar objetos estáticos y la tercera modela píxeles con cambios rápidos u objetos de frente.

Junto con el modelo MoG se integra la técnica de tratamiento de cambios rápidos de iluminación *Phong shading* [28]. Esta técnica (también usada en [27]) se basa en la idea de que en una imagen la intensidad de un píxel de un objeto $I(x, y)$ se obtiene con el producto de la iluminación de la fuente de luz $I_l(x, y)$ por la reflectancia de la superficie del objeto $I_o(x, y)$. Por tanto, modelando la imagen actual como $I = I_l \times I_o$ y el fondo como $B = B_l \times B_o$, se establece la

igualdad $I_o = B_o$ (pues solo la componente de reflectancia contiene información de los objetos) a partir de la cual van a eliminarse falsos positivos.

A continuación se emplea la segunda Gaussiana para realizar la detección estática, determinando que un píxel es estacionario cuando la segunda Gaussiana en ese píxel ($w_2(x, y)$) supera cierto peso (τ):

$$SFG_{MoG}(x, y) = \begin{cases} 1 & \text{si } w_2(x, y) > \tau \\ 0 & \text{resto} \end{cases}, \quad (2.17)$$

Este método trata la absorción progresiva de objetos estáticos, llevando a cabo una absorción total por parte del modelo de fondo (primera Gaussiana) cuando detecta que un objeto comienza a disminuir de tamaño. Es en este momento cuando se lleva a cabo una clasificación robo/abandono (mediante *Edge energy* [29] y *Region growing* [30]) que lleva a no perder las detecciones.

El método puede operar con lo descrito anteriormente, pero se propone una interacción entre un módulo de *tracking* y la sustracción de fondo para tratar la absorción de objetos estáticos y objetos moviéndose lentamente. Los autores, basándose en las detecciones del módulo de seguimiento, proponen:

1. Suprimir la absorción de objetos de frente que se mueven lentamente.
2. Absorber los objetos de frente que permanecen parados durante un tiempo, guardando el modelo de fondo previo a la absorción. Además el objeto puede guardarse para ser detectado cuando éste reanude el movimiento.
3. Recuperar el fondo original guardado (evitando así una detección fantasma) cuando se vuelve a detectar frente en la zona, ya sea porque el objeto seguido vuelve a moverse o porque se tiene un nuevo objeto.

Las detecciones se mantienen mientras no se detecte el fondo previo a la absorción (etapa 2 del *tracking*). Una detección final de ejemplo se muestra en la Figura 2.18:

Una vez explicado el método, es importante hacer varias observaciones:

- Para adaptarse a las condiciones del fondo se lleva a cabo una actualización del fondo que lleva a absorber regiones estáticas (pues acaban detectándose como píxeles persistentes). Estas absorciones de regiones estacionarias sufren una clasificación de objeto robado o abandonado.
- Los cambios rápidos de iluminación son tratados por el método *Phong shading* que, aunque trata bien los cambios de iluminación, no es suficiente para eliminar todo falso positivo. Por tanto, en caso de permanecer estáticos serán absorbidos y tratados como eventos indeterminados en la clasificación robo/abandono.



Figura 2.18: Detecciones realizadas tras absorber los objetos al modelo de fondo y clasificar como abandono.

- La idea de interacción *tracking*-sustracción de fondo funciona bien en entornos sencillos (tal y como se demuestra en el trabajo), pero no en entornos altamente concurridos donde hay numerosas regiones de frente a seguir que además sufren numerosas oclusiones y que van a provocar un descenso del rendimiento del algoritmo debido a la baja eficacia del *tracking*.

2.5. Limitaciones de las métodos existentes

Atendiendo a las métodos existentes en la literatura, se han observado varios aspectos críticos a la hora de realizar detecciones de regiones estáticas:

- Inicio del modelo de fondo: Es un punto crítico, sobre todo en escenarios altamente concurridos, provocando falsos positivos por fantasmas o cambios de iluminación.
- Actualización del modelo de fondo: Supone una limitación muy importante pues, aquellos modelos que llevan a cabo una adaptación del modelo de fondo a la escena, reduciendo así falsos positivos derivados de una mala inicialización, tienen problemas para no absorber también los objetos estáticos. Además si el objeto estático lo es solo de forma temporal, entonces va a producir una falsa detección que debe ser tratada.
- Movimiento: En escenarios densamente poblados es habitual encontrar zonas de paso con elevado movimiento de personas que puede provocar numerosas falsas detecciones, al tener numerosas regiones de frente (no estáticas) activas continuamente. Es un problema importante el filtrado de estas situaciones manteniendo un equilibrio con detecciones que sí deben realizarse pero que sufren oclusiones de manera continua.
- Cambios de iluminación, sombras y reflejos: Son circunstancias que ocurren habitualmente, sobre todo en entornos densamente poblados, que provocan numerosos falsos positivos cuando los cambios en la escena son rápidos si solo se confía en el algoritmo de sustracción de fondo (como ocurre en muchas técnicas).

Métodos	Robustez frente a				
	Inicio del fondo	Oclusiones con intenso movimiento	Movimiento continuado	Cambios de iluminación	Objetos estáticos temporales
[12]	No	Sí	No	No	Sí
[16]	No	Sí	No	No	Sí
[17]	Sí	Sí	No	No	No
[18]	No	Sí	No	No	Sí
[7]	No	No	Sí	No	Sí
[9]	Sí	Sí	Sí	Sí	No
[25]	Sí	No	No	Sí	No
[8]	Sí	Sí	No	Sí	No
[26]	Sí	Sí	Sí	No	No
[27]	Sí	Sí	No	Sí	No
[1]	Sí	Sí	Sí	No	No
[3]	Sí	Sí	No	Sí	Sí
Propuesta	No	Sí	Sí	Sí	Sí

Tabla 2.1: Comparativa de las métodos del estado del arte más relevantes.

El comportamiento de las métodos más relevantes, considerando entornos altamente concurridos, frente a estos y otros aspectos se refleja en la tabla 2.1:

Un aspecto ajeno a la calidad de la detección pero muy importante a la hora de evaluar la calidad de los métodos, es el tiempo utilizado en la activación de alarmas de objetos estáticos. Muchas técnicas llevan a cabo una evaluación con tiempos elevados (relativos a la duración de las secuencias, p. ej, 1 minuto para secuencias de 2 o 3 minutos). Este aspecto no permite juzgar la robustez frente a cambios rápidos de iluminación, que se dan en entornos altamente concurridos por la alta presencia de personas y sus sombras, en muchos de los *datasets* públicos en los que se prueban los métodos.

Capítulo 3

Algoritmo propuesto

En este capítulo se propone un algoritmo para la detección de regiones estáticas capaz de operar en entornos altamente concurridos gracias a su robustez a oclusiones, intenso movimiento, sombras y cambios de iluminación. El capítulo se divide en cinco secciones, la primera proporciona una visión general del algoritmo (sección 3.1), la segunda explica los tipos de análisis y características utilizadas (sección 3.2), la tercera detalla la combinación de todas ellas (sección 3.3), la cuarta el manejo de oclusiones (sección 3.4) y la última (sección 3.5) explica la configuración necesaria para un correcto funcionamiento del método.

3.1. Estructura general

Se propone realizar un análisis espacio-temporal de tres características: frente, movimiento y estructura (ver Figura 3.1). Cada análisis consta de dos etapas: extracción de característica y su acumulación *frame a frame*. De la acumulación espacio-temporal de las características,

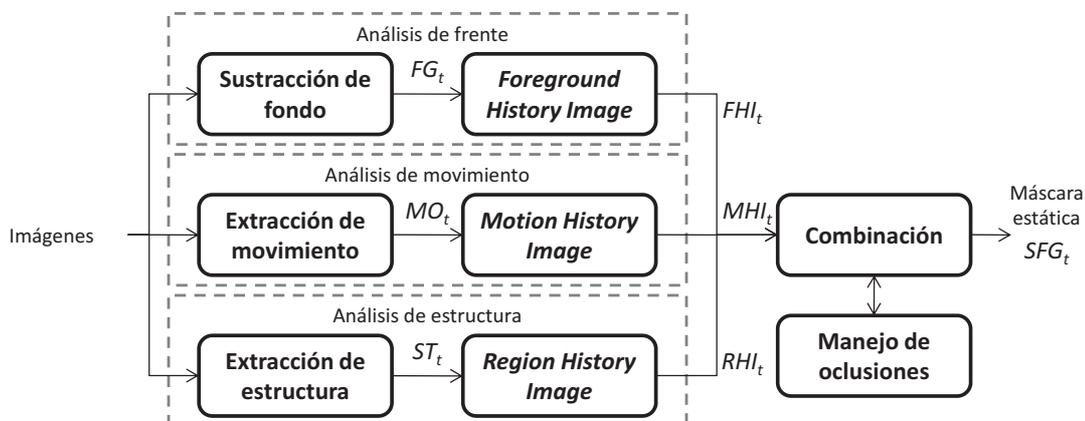


Figura 3.1: Esquema del algoritmo propuesto.

se obtienen tres imágenes (*History Images*), que se combinan para modelar la variación de frente-movimiento-estructura a lo largo del tiempo. Con la imagen resultado se lleva a cabo una umbralización para obtener la máscara de regiones estáticas.. Por último, se incluye en esta decisión un manejo de oclusiones para recuperar píxeles estacionarios perdidos por las frecuentes oclusiones en escenarios altamente concurridos. En el resto de este capítulo se describe cada análisis realizado, su combinación y del manejo de oclusiones para detectar las regiones estáticas.

3.2. Análisis realizados

3.2.1. Análisis de frente

La primera característica empleada es la máscara de regiones de frente, para extraerla se utiliza el algoritmo de sustracción de fondo Gamma, propuesto en [10], por su bajo coste computacional y su robustez al ruido. En este algoritmo, el modelo de fondo se representa, para cada píxel, con un valor medio y una varianza común para todos los píxeles que se corresponde con la varianza del ruido de la escena. La detección de píxeles de fondo y objeto se realiza píxel a píxel mediante el cálculo de la diferencia generada por una ventana cuadrada de tamaño Q alrededor del píxel a analizar, tanto de la imagen I como del modelo de fondo existente B . Cuando esa distancia elevada al cuadrado supera un cierto umbral β , dicho píxel se considera frente:

$$FG_t(\mathbf{x}) = 1 \iff \sum_{\mathbf{d} \in \mathcal{N}(\mathbf{x})} (I_t(\mathbf{d}) - B_t(\mathbf{d}))^2 > \beta, \quad (3.1)$$

donde \mathbf{x} y \mathbf{d} son ubicaciones $\{x, y\}$ de los píxeles; $\mathcal{N}(\mathbf{x})$ es la ventana $Q \times Q$ centrada en \mathbf{x} e I_t y B_t son la imagen y el fondo actual en el instante t . $FG_t(\mathbf{x}) = 1(0)$ indica frente (fondo) para cada píxel en la posición \mathbf{x} .

Una vez que se ha obtenido la máscara de frente, se acumula su variación temporal para obtener lo que llamamos *Foreground History Image* $FHI_t(\mathbf{x})$ (ver Figura 3.2). Se realiza una operación de actualización diferente, en función de si el píxel pertenece a una detección de frente o fondo:

$$FHI_t(\mathbf{x}) = FHI_{t-1}(\mathbf{x}) + w_{pos}^f \cdot FG_t(\mathbf{x}), \quad (3.2)$$

$$FHI_t(\mathbf{x}) = FHI_{t-1}(\mathbf{x}) - w_{neg}^f \cdot (\sim FG_t(\mathbf{x})), \quad (3.3)$$

donde \sim es la operación de inversión lógica NOT y w_{pos}^f y w_{neg}^f son dos pesos para controlar la contribución de las detecciones de frente ($FG_t(\mathbf{x}) = 1$) y fondo ($\sim FG_t(\mathbf{x}) = 1$).

Para proporcionar sentido temporal a esta acumulación, los incrementos y decrementos utilizados deben ser siempre los mismos, concretamente FHI_t debe incrementarse de uno en uno



Figura 3.2: De izquierda a derecha: *frame* actual, frente detectado y acumulación temporal de la característica ($FHI_t(\mathbf{x})$). Se puede apreciar la errónea acumulación de un cambio de iluminación provocado por el vagón.

($w_{pos}^f = 1$) cuando los píxeles pertenecen al frente y resetearse a 0 cuando los píxeles pertenecen al fondo (así se mantiene la acumulación como un contador del número de *frames* que permanece cada píxel de frente activo de forma consecutiva). Sin embargo, errores ocasionales en la máscara de frente pueden causar la pérdida de las detecciones si se lleva a cabo el reseteo descrito. Esta situación es habitual en entornos altamente concurridos donde las regiones estáticas sufren oclusiones continuadas de objetos en rápido movimiento que pueden tener camuflaje. Por tanto, el peso de penalización w_{neg}^f debe disminuir FHI_t en mayor medida que w_{pos}^f lo incrementa, pero sin llevar a cabo un reseteo, así se consigue cierta robustez contra errores de la máscara de frente (p. ej., $w_{neg}^f = 15$), como se muestra en la Figura 3.3.

Por tanto, el resultado de este análisis es $FHI_t(\mathbf{x})$, una puntuación que incrementa cuando el píxel pertenece al frente y disminuye cuando el píxel pertenece al fondo otorgando cierta robustez a oclusiones de regiones que contienen errores en la detección de frente.

3.2.2. Análisis de movimiento

Trabajos recientes demuestran el uso de información de movimiento para filtrar falsos positivos ocasionados por un intenso tránsito de personas en un área, ayudando así a mejorar el rendimiento de la detección de regiones estáticas [7][9]. El empleo actual de esta información se centra en umbralizar diferencias entre *frames* (mediante *frame-difference*) e incorporarlas a un análisis de acumulación temporal [9] o aplicar la aproximación de submuestreo sobre esas diferencias [7]. No obstante, las regiones estáticas sufren oclusiones constantes en entornos densamente poblados, provocando que sea difícil observar el no-movimiento, que proporciona el objeto estático constantemente ocluido, más allá de unos pocos *frames*. Por tanto, para obtener un análisis correcto del movimiento sin perder aquellas regiones que estén ocluidas en un instante concreto, es necesario elegir correctamente la frecuencia de muestreo [7] (aspecto que no se puede garantizar especialmente en zonas de paso). En [9] se depende del análisis *frame a frame*,

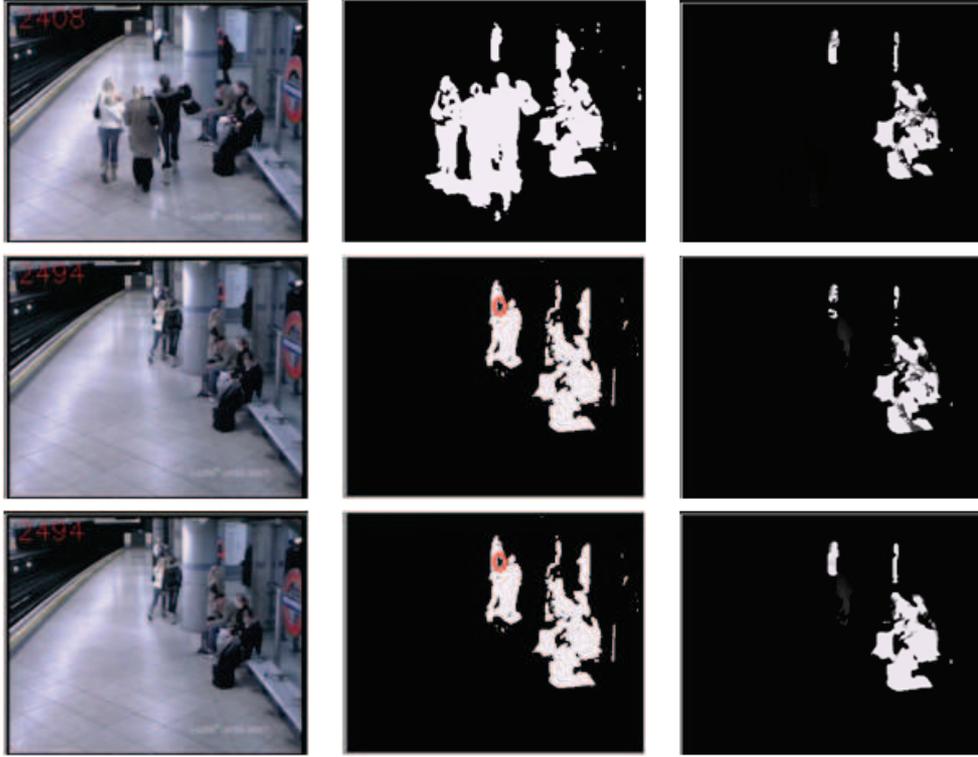


Figura 3.3: Ejemplo de oclusión con error de frente. De izquierda a derecha: *frame* actual, frente detectado y acumulación temporal de la característica ($FHI_t(\mathbf{x})$). Primera fila: *frame* 2408 con una maleta libre de oclusiones y gente aproximándose. Segunda fila: *frame* 2494, la puntuación FHI_t está calculada con $w_{neg}^f = FHI_{t-1}$ (reseteo), lo cual provoca que al tener una persona con camuflaje ocluyendo la maleta (círculo rojo, segunda columna), se pierdan varias partes de la maleta en FHI_t . Tercera fila: Se muestra el mismo caso que en la segunda pero con un decremento controlado ($w_{neg}^f = 15$), lo cual lleva a preservar mejor la maleta en FHI_t .

que trata por igual zonas donde hay movimiento con y sin objetos detrás, lo cual lleva a perder detecciones o a no eliminar falsos positivos correctamente según la umbralización realizada.

En este trabajo se propone un esquema para resolver estas limitaciones, extendiendo el análisis de movimiento a una ventana de duración T posterior y anterior al *frame* bajo análisis (para poder detectar los objetos visibles en breves intervalos) y realizando el análisis *frame* a *frame* (para evitar la elección de un instante de muestreo) (ver Figura 3.4). Aunque las regiones estáticas se ven afectadas por múltiples oclusiones en escenarios altamente concurridos, habitualmente éstas pueden observarse durante algunos breves periodos, de manera que la región más predominante en breves intervalos temporales (ventana T) corresponde al objeto estático.

Para extraer la característica de movimiento, aplicamos un filtro de mediana en dos ventanas temporales antes y después del *frame* bajo análisis en cada instante temporal:

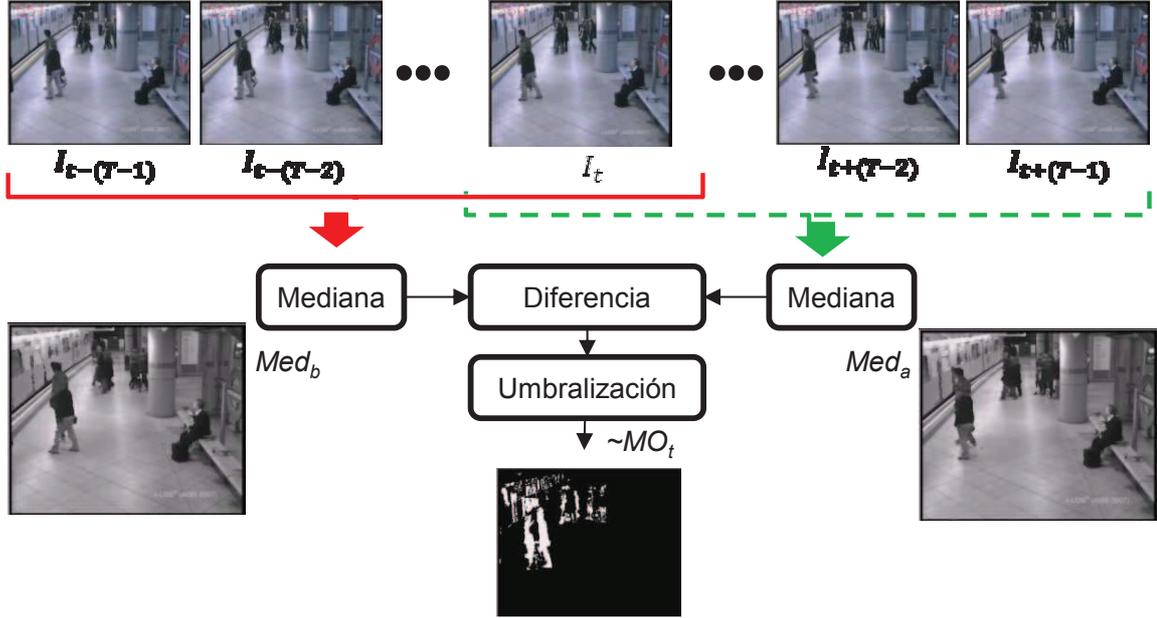


Figura 3.4: Esquema de extracción de movimiento utilizando un filtro mediana en una ventana temporal previa y posterior al *frame* bajo análisis.

$$Med_b = Mediana\{I_{t-T+1}, \dots, I_t\} \quad (3.4)$$

$$Med_a = Mediana\{I_t, \dots, I_{t+T-1}\} \quad (3.5)$$

donde Med_a y Med_b son las imágenes mediana de las ventanas temporales de longitud T consideradas antes y después de I_t (todas las imágenes en escala de grises). En consecuencia se está introduciendo un retardo en cada instante t para poder incluir en el análisis los siguientes $T - 1$ *frames*. La elección de T depende de la velocidad de los objetos y la duración de las oclusiones, teniendo que emplear valores tanto mayores como mayor sea la duración de la oclusión. Pruebas empíricas con secuencias reales muestran un buen rendimiento para valores de T comprendidos entre 10 y 20. A continuación se obtiene la imagen final de no-movimiento como:

$$MO_t(\mathbf{x}) = \begin{cases} 1 & \text{si } |Med_b - Med_a| < \tau \\ 0 & \text{resto} \end{cases}, \quad (3.6)$$

donde $MO_t(\mathbf{x}) = 1$ indica ausencia de movimiento y τ es el umbral que determina cuándo no hay movimiento. Hemos obtenido τ aplicando el método de Kapur [31] sobre la imagen $|Med_b - Med_a|$. Este método calcula el umbral óptimo dividiendo los píxeles en dos grupos, objeto y fondo y buscando el τ que maximiza la suma de entropías de esos dos conjuntos:

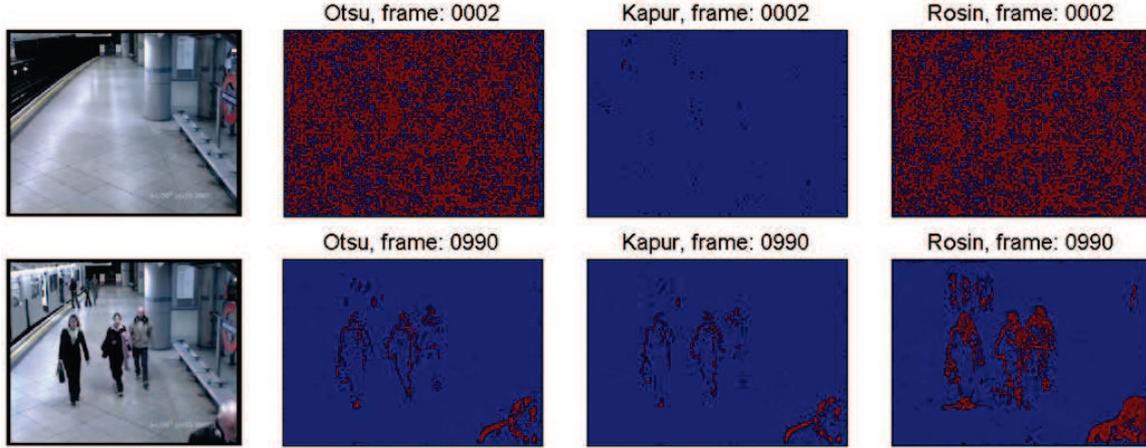


Figura 3.5: Comparativa de movimiento obtenido con diferentes técnicas de cálculo del umbral. En la primera fila puede apreciarse como el algoritmo de Kapur es el único de los tres que es robusto a situaciones sin movimiento. Por otro lado, en la segunda fila, se puede ver como el método utilizado detecta menos movimiento que los otros dos.

$$\tau = \max\{H(\tau) = H_f(\tau) + H_b(\tau)\}, \quad (3.7)$$

$$H_f(\tau) = - \sum_{g=0}^{\tau} \frac{p(g)}{P(\tau)} \times \log\left(\frac{p(g)}{P(\tau)}\right), \quad (3.8)$$

$$H_b(\tau) = - \sum_{g=\tau+1}^{255} \frac{p(g)}{1 - P(\tau)} \times \log\left(\frac{p(g)}{1 - P(\tau)}\right), \quad (3.9)$$

donde $p(g)$ es la probabilidad del valor en escala de grises g . Se ha elegido este método por su robustez en situaciones en las que no hay movimiento, en las que otros métodos típicos como [32] o [33] provocan numerosos errores (ver Figura 3.5).

Finalmente, se acumula temporalmente la variación de $MO_t(\mathbf{x})$, obteniendo lo que llamamos *Motion History Image* $MHI_t(\mathbf{x})$, siendo similar a la acumulación de frente $FHI_t(\mathbf{x})$:

$$MHI_t(\mathbf{x}) = MHI_{t-1}(\mathbf{x}) + w_{pos}^m \cdot MO_t(\mathbf{x}), \quad (3.10)$$

$$MHI_t(\mathbf{x}) = MHI_{t-1}(\mathbf{x}) - w_{neg}^m \cdot (\sim MO_t(\mathbf{x})), \quad (3.11)$$

donde w_{pos}^m y w_{neg}^m son dos pesos para controlar la contribución de los píxeles sin movimiento ($MO_t(\mathbf{x}) = 1$) y con movimiento ($\sim MO_t(\mathbf{x}) = 1$). Al igual que ocurre en $FHI_t(\mathbf{x})$, se debe incrementar $MHI_t(\mathbf{x})$ de uno en uno ($w_{pos}^m = 1$) cuando los píxeles no sufren movimiento (para mantener así una coherencia temporal entre acumulaciones) y resetear los valores de $MHI_t(\mathbf{x})$ a 0 cuando los píxeles sufren movimiento, para mantener así una coherencia temporal en la

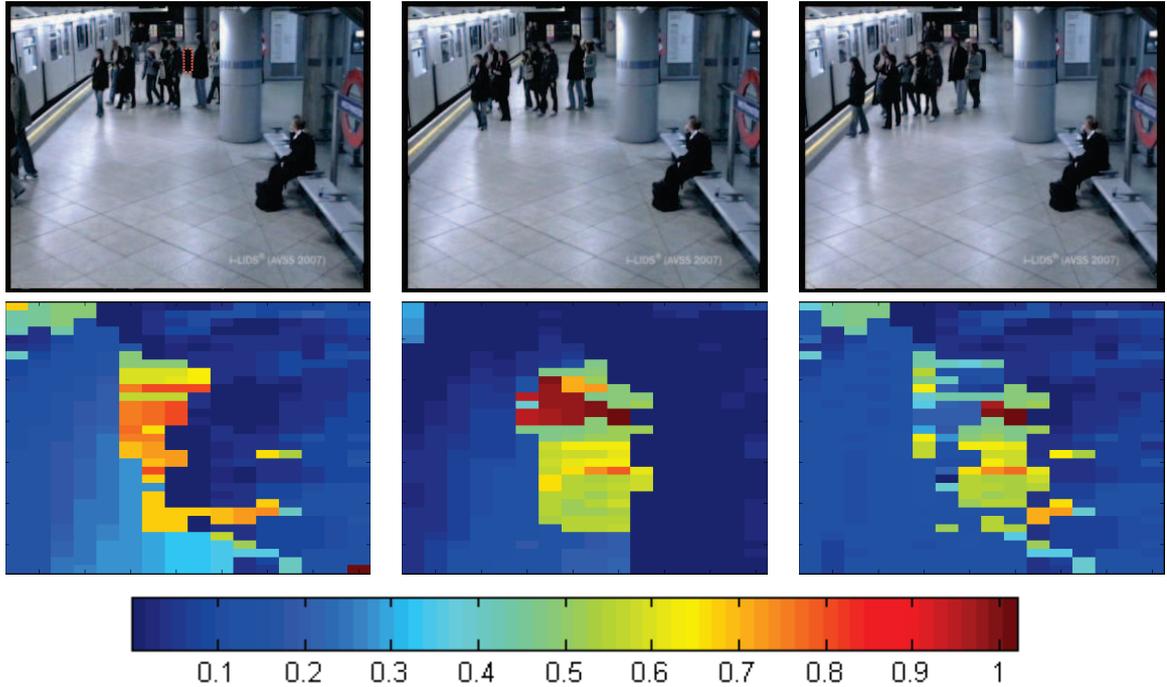


Figura 3.6: Ejemplo de $MHI_t(\mathbf{x})$ empleando la característica de movimiento propuesta (PRO) y el *frame-difference* básico (FD) [7]. Primera fila: frames 875, 917 y 940. Segunda fila: $MHI_{917}(\mathbf{x})$ resultados del *frame* 917 usando FD (izquierda), PRO (centro) y la diferencia absoluta de ambos (derecha) para el rectángulo rojo de línea discontinua en el *frame* 875 (maleta). PRO estima mejor el no-movimiento de la región estática ocluida teniendo valores mayores para $MHI_{917}(\mathbf{x})$.

puntuación obtenida. A diferencia de $FHI_t(\mathbf{x})$ cuyo objetivo es detectar regiones de frente, el propósito de $MHI_t(\mathbf{x})$ es compensar los valores altos de $FHI_t(\mathbf{x})$ causados por intenso movimiento, manteniendo solo aquellos valores de $FHI_t(\mathbf{x})$ libres de movimiento, es decir, filtrar regiones de frente. Por tanto, se fija $w_{neg}^m = MHI_{t-1}(\mathbf{x})$ para resetear $MHI_t(\mathbf{x})$ a 0 cuando se detecta movimiento. La Figura 3.6 muestra un ejemplo de extremo movimiento con una región estática ocluida continuamente solo es detectada utilizando $MHI_t(\mathbf{x})$.

En resumen, el resultado del análisis de movimiento es $MHI_t(\mathbf{x})$, una puntuación que se incrementa cuando en el píxel no hay movimiento y se resetea en caso contrario. Gracias al esquema de extracción de la característica de no-movimiento, es capaz de detectar y mantener píxeles estáticos que sufren oclusiones en condiciones de extremo movimiento (ver Figura 3.7).

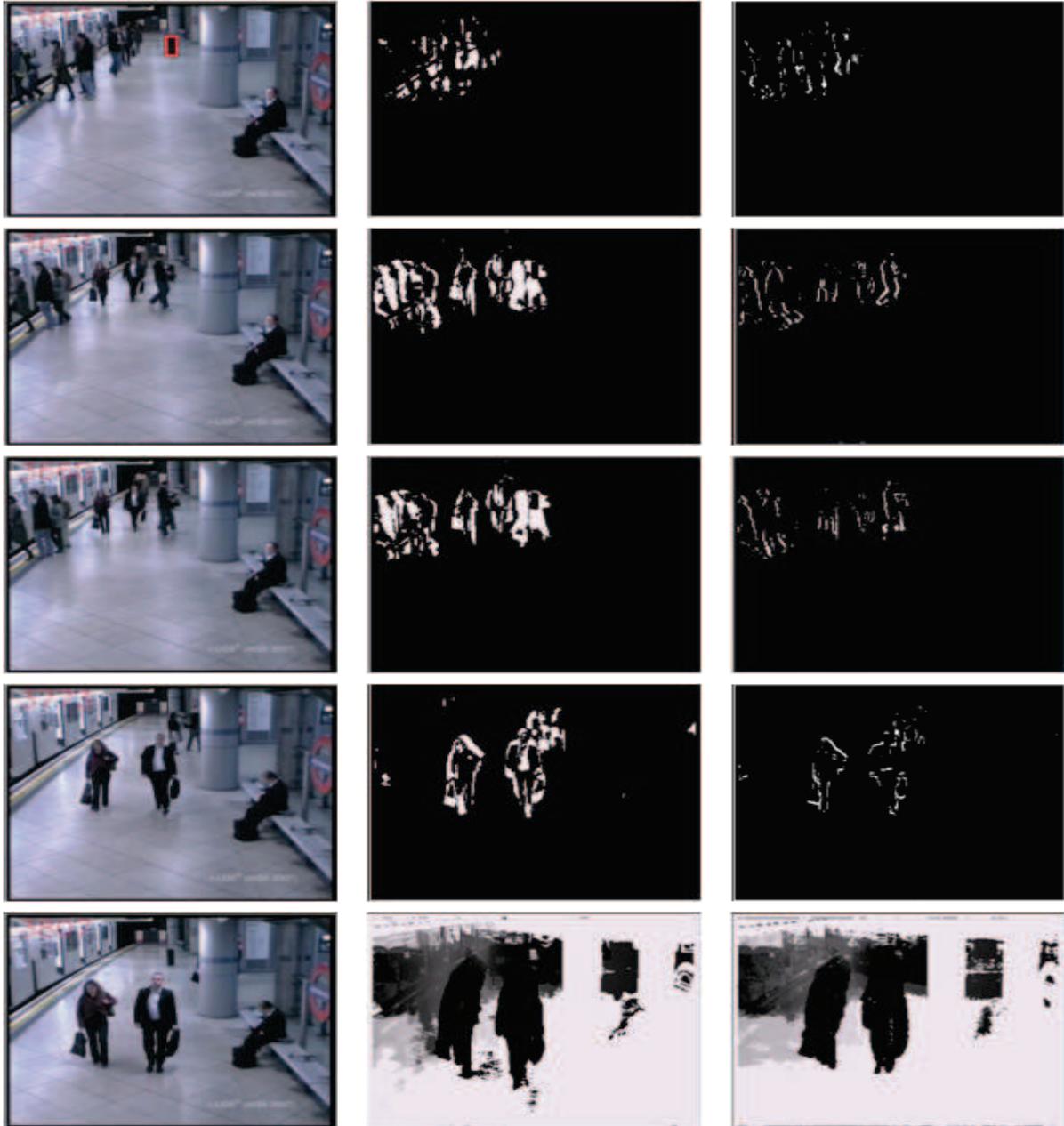


Figura 3.7: Ejemplo de extracción de movimiento en condiciones de oclusiones. De izquierda a derecha, cada fila (salvo la última) muestra *frame* bajo análisis, imagen diferencia empleando el filtro de mediana e imagen diferencia entre *frames* adyacentes. En cada fila se muestran los *frames* 3617, 3657, 3660 y 3723, que representan sucesivos instantes temporales en algunos de los cuales se ocluye una maleta (marcada en el *frame* bajo análisis de la primera fila en rojo). Se puede apreciar como el esquema de extracción de movimiento propuesto (segunda columna) es capaz de detectar en condición de oclusiones que no existe movimiento en la zona de la maleta (fila dos, tres y cuatro), mientras que el *frame-difference* convencional (tercera columna) sí que detecta movimiento en las personas que pasan por delante. La última fila representa un instante posterior a las oclusiones (3807) en el que se muestra como la característica $MHI_t(\mathbf{x})$ para el modelo propuesto mantiene mejor la puntuación para los píxeles de la maleta.

3.2.3. Análisis de estructura

Trabajos recientes [9][27] muestran el uso de información a nivel de región como vía para corregir las limitaciones de la sustracción de fondo relacionadas con los cambios de iluminación. Como se presentó en el estado del arte, estos trabajos emplean diversas técnicas en las que se comparan regiones del *frame* bajo análisis y del modelo de fondo para establecer su similitud.

En este trabajo se propone la utilización de la técnica descrita en [34], conocida como *Structural Similarity* (SSIM), para extraer una característica de estructura que nos permita eliminar falsos positivos causados por sombras o cambios de iluminación.

SSIM es un método empleado para determinar la calidad percibida en una imagen, que se basa en la idea de que el sistema visual humano está altamente adaptado para la extracción de información estructural de una escena y que por tanto, mediante una medida de su degradación, puede obtenerse una buena aproximación de la calidad percibida. Es un algoritmo *full-reference*, es decir, que asume la existencia de una imagen libre de distorsión (en nuestro caso es el modelo de fondo), con la que llevar a cabo la comparación de calidad. El aspecto que nos interesa de SSIM es su independencia de luminancia y contraste de la escena, pues la información estructural de una imagen es independiente de la iluminación que haya en la misma. Concretamente van a considerarse luminancia y contraste locales, pues ambos pueden variar según la zona de la escena.

El sistema separa la medida de similitud teniendo en cuenta 3 aspectos: luminancia, contraste y estructura. Siendo a y b las 2 señales (imágenes) a comparar, se tiene como medida de similitud el mapa de valores $SSIM(a, b)$:

$$SSIM(a, b) = l(a, b) \times c(a, b) \times s(a, b), \quad (3.12)$$

donde $l(a, b)$, $c(a, b)$ y $s(a, b)$ son respectivamente las medidas de comparación de luminancia, contraste y estructura (ver Figura 3.8).

Como medida comparativa de la luminancia se utiliza:

$$l(a, b) = \frac{2(1 + R)}{1 + (1 + R)^2 + \frac{C_1}{\mu_a^2}}, \quad (3.13)$$

$$\mu_a = \frac{1}{K} \sum_{i=1}^K a_i, \quad (3.14)$$

donde a_i son los píxeles pertenecientes a la imagen I_t , K define el número de píxeles de la ventana local empleada para el cálculo, μ_a define la luminancia media de la imagen de referencia, R mide el cambio de luminancia relativo a μ_a y C_1 es una constante de valor casi despreciable. Se define entonces la luminancia media de la imagen a comparar como $\mu_b = (1 + R)\mu_a$, que es proporcional a la luminancia de referencia. Si $\mu_a \simeq \mu_b$ entonces el valor de R es pequeño y en consecuencia

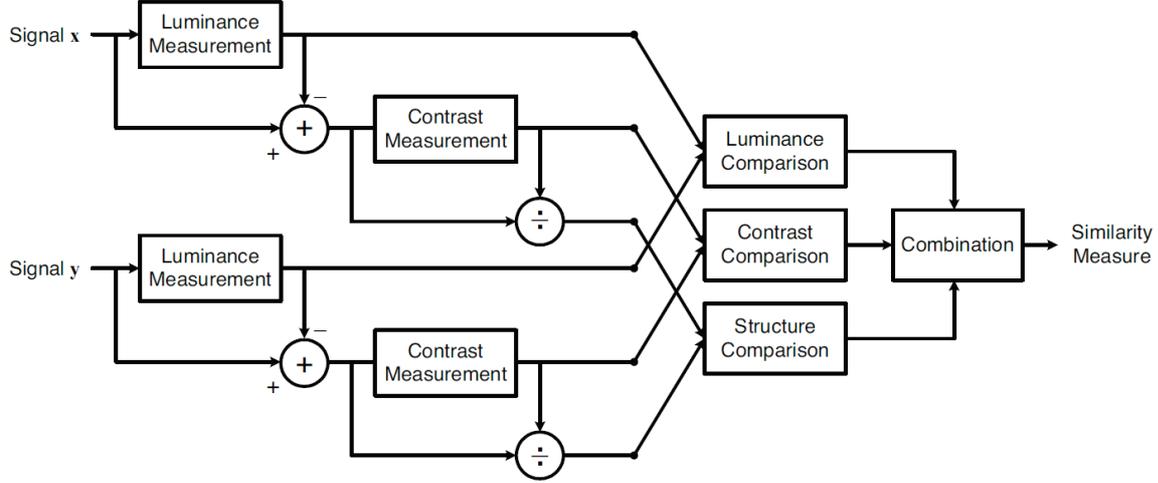


Figura 3.8: Esquema para el cálculo de la medida de similitud SSIM (extraído de [34]).

$l(a, b)$ tiene un valor elevado (alta similitud en la luminancia).

Para la estimación de las señales de contraste se emplean las desviaciones típicas (σ_a y σ_b), empleando la siguiente función de comparación:

$$c(a, b) = \frac{2\sigma_a\sigma_b + C_2}{\sigma_a^2 + \sigma_b^2 + C_2}, \quad (3.15)$$

$$\sigma_a = \left(\frac{1}{K-1} \sum_{i=1}^K (a_i - \mu_a)^2 \right)^{\frac{1}{2}}, \quad (3.16)$$

donde σ_b se define análogamente a σ_a y C_2 es una constante de valor casi despreciable. Como se puede apreciar, si $\sigma_a \simeq \sigma_b$ entonces $c(a, b)$ es grande (alta calidad).

Por último, para comparar la estructura se utiliza el coeficiente de correlación de ambas imágenes:

$$s(a, b) = \frac{\sigma_{ab} + C_3}{\sigma_a\sigma_b + C_3}, \quad (3.17)$$

$$\sigma_{ab} = \frac{1}{K-1} \sum_{i=1}^K (a_i - \mu_a)(b_i - \mu_b), \quad (3.18)$$

donde σ_{ab} es la covarianza de σ_a y σ_b y C_3 es una constante de valor casi despreciable definida como $C_3 = \frac{C_2}{2}$.

$SSIM(a, b)$ cumple las condiciones de simetría ($SSIM(a, b) = SSIM(b, a)$), acotación ($SSIM(a, b) \leq 1$) y máximo único ($SSIM(a, b) = 1$ solo si $a = b$) y es un mapa de valores de similitud para cada píxel. La razón de incluir las constantes C_1 , C_2 y C_3 es evitar resultados inestables cuando $(\mu_a^2 + \mu_b^2)$ o $(\sigma_a^2 + \sigma_b^2)$ tienen valores próximos a cero. Además es importante comentar que la ventana local empleada para el análisis de similitud en cada píxel es una función

Región de fondo	Región de frente	Matriz de SSIM (por regiones)				Tamaño
		0.9546	0.9373	0.9164	0.8698	65x110
		0.8897	0.9186	0.9270	0.9258	
		0.8139	0.9133	0.9294	0.9364	
		0.8379	0.9251	0.9455	0.9446	
		0.9399	0.9550	0.9774	0.9825	157x41
		0.9658	0.9685	0.9756	0.9800	
		0.9642	0.9540	0.9518	0.9527	
		0.9547	0.9461	0.9556	0.9745	
		0.9271	0.9568	0.9434	0.9397	32x96
		0.8538	0.8946	0.9015	0.9236	
		0.8213	0.8664	0.8658	0.8949	
		0.9341	0.8994	0.9054	0.9118	
		0.3675	0.0835	0.0973	0.3866	60x84
		0.3731	0.1291	0.0494	0.0802	
		0.1440	0.1526	0.3875	0.6427	
		0.1332	0.0988	0.2208	0.5760	
		0.1355	0.2364	0.1977	0.1329	40x115
		0.1354	0.1835	0.1675	0.1250	
		0.2301	0.3964	0.4327	0.4412	
		0.6218	0.7917	0.8504	0.8406	

Figure 3.9: Primera columna: Región del modelo de fondo. Segunda columna: Misma región que la primera columna sufriendo cambios de iluminación (filas 1, 2 y 3) o oclusión por parte de un objeto (filas 4 y 5). Tercera columna: Mapa de $SSIM$ calculado dividiendo la imagen en 16 bloques. Se aprecia en las tres primeras filas un $SSIM$ elevado pues primera y segunda columna son iguales. Sin embargo cuando un objeto ocluye el fondo se obtiene una similitud baja tal y como muestran las filas 4 y 5. Adicionalmente la fila 6 muestra como para zonas donde se tiene una sombra, se obtiene una $SSIM$ elevada.

de suavizado Gaussiana simétrica circular de tamaño 11×11 .

Por último, como el método se emplea para medir calidad percibida en una imagen, el resultado buscado es un único valor calculado como la media del mapa $SSIM(a, b)$ obtenido para cada píxel:

$$MSSIM(a, b) = \frac{1}{M} \sum_{j=1}^M SSIM_j(a, b), \quad (3.19)$$

donde M es el número de valores del mapa de $SSIM(a, b)$ calculado.

En la Figura 3.9 se muestran ejemplos de las capacidades de $SSIM$ mediante comparaciones de regiones de la imagen con cambios de iluminación, sombras u objetos con el fondo original, obteniendo valores de $SSIM$ bajos cuando se compara un objeto con el fondo y altos cuando la comparación es entre una zona afectada por una sombra o cambio de iluminación y el fondo.

Para extraer la característica de estructura se han realizado varias modificaciones a partir de $SSIM$ (ver Figura 3.10).

Primero se calcula el mapa de $SSIM$ entre un modelo de fondo (se ha aprovechado el modelo

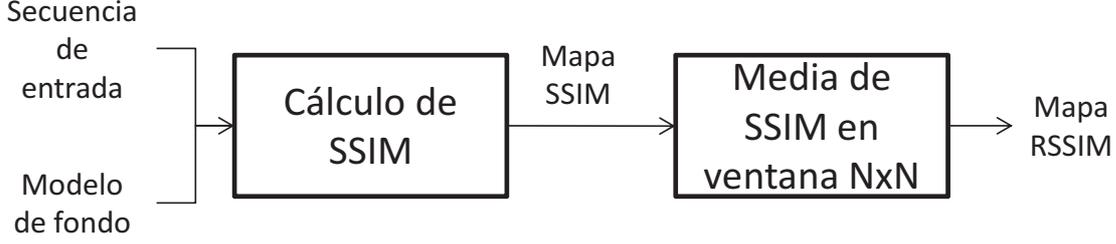


Figura 3.10: Esquema de cálculo de la modificación de $SSIM$, el mapa $RSSIM$.

calculado en la característica de frente) y el *frame* bajo análisis. A continuación, se ha modificado cada valor del mapa aplicando a cada píxel la media de valores de $SSIM$ de un vecindario de $N \times N$ centrado en cada píxel. La razón de incluir el valor de $SSIM$ de píxeles cercanos en el análisis, es incorporar al esquema cierta robustez a zonas donde tiene lugar saturación tras el cambio de iluminación y en las que una transformación inversa de luminancia no es capaz de recuperar la información original. La nueva característica de estructura se designa como un mapa de valores de similitud de píxel calculados como sigue:

$$RSSIM_i(\mathbf{x}) = \sum_{i \in R} \frac{SSIM_i(a, b)}{N \times N}, \quad (3.20)$$

donde R es el vecindario $N \times N$ centrado en el píxel \mathbf{x} . N debe ser mayor para incrementar la robustez a zonas saturadas por iluminación, no obstante valores muy elevados llevan a perder detecciones no muy grandes al ser la mayor parte del vecindario regiones de fondo (valores de $RSSIM_t(\mathbf{x})$ elevados). Resultados experimentales muestran un buen rendimiento ante zonas no muy grandes que sufren saturación con $N = 12$ (ver Figura 3.11).

Finalmente se lleva a cabo la acumulación de la variación temporal de $RSSIM_t(\mathbf{x})$, obteniendo así la *Region History Image* $RHI_t(\mathbf{x})$, que sigue unas reglas similares a $FHI_t(\mathbf{x})$ y $MHI_t(\mathbf{x})$:

$$RHI_t(\mathbf{x}) = RHI_{t-1}(\mathbf{x}) + w_{pos}^r \cdot ST_t(\mathbf{x}), \quad (3.21)$$

$$RHI_t(\mathbf{x}) = RHI_{t-1}(\mathbf{x}) - w_{neg}^r \cdot (\sim ST_t(\mathbf{x})), \quad (3.22)$$

donde w_{pos}^r y w_{neg}^r son dos pesos para controlar la contribución de los píxeles con diferente estructura ($ST_t(\mathbf{x}) = 1$) y con igual estructura ($\sim ST_t(\mathbf{x}) = 1$) (ver Figura 3.11). $ST_t(\mathbf{x})$ se define como:

$$ST_t(\mathbf{x}) = \begin{cases} 1 & \text{si } RSSIM_t(\mathbf{x}) \leq 1 - RSSIM_t(\mathbf{x}) \\ 0 & \text{resto} \end{cases}, \quad (3.23)$$

Al igual que ocurre en las acumulaciones de frente y movimiento, se debe incrementar

$RHI_t(\mathbf{x})$ de uno en uno ($w_{pos}^r = 1$) cuando los píxeles tienen diferente estructura y resetear los valores de $RHI_t(\mathbf{x})$ a 0 cuando los píxeles tienen misma estructura, para mantener así una coherencia temporal en la puntuación obtenida. Gracias a la utilización de información a nivel de región, esta característica es robusta a camuflajes, hecho que permite llevar a cabo un reseteo de $RHI_t(\mathbf{x})$ cuando ($ST_t(\mathbf{x}) = 1$), es decir, se fija $w_{neg}^r = RHI_{t-1}(\mathbf{x})$.

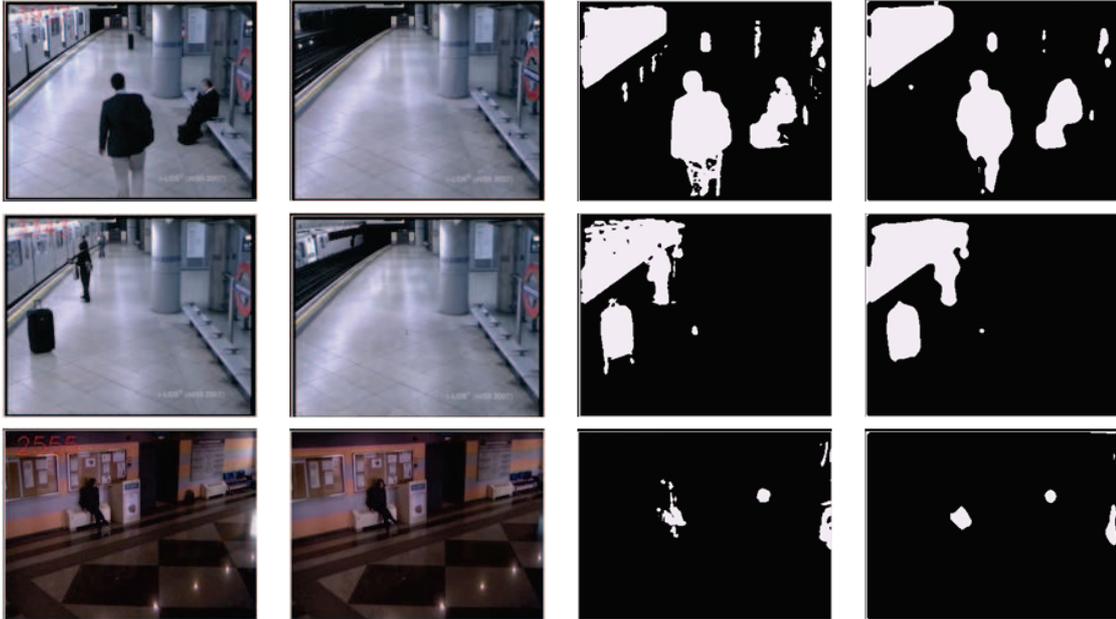


Figura 3.11: De izquierda a derecha: *frame* bajo análisis, fondo, $ST_t(\mathbf{x})$ con $N = 1$ y $ST_t(\mathbf{x})$ con $N = 12$. Las tres filas muestran como, al aumentar el tamaño considerado para determinar la característica de estructura, se reducen/eliminan falsos positivos por sombras y cambios de iluminación de zonas que sufren saturación en su luminancia. Este hecho añadido a la variabilidad temporal de sombras y cambios de iluminación, provocan que al acumular la característica en estas regiones se obtengan buenos comportamientos frente a estos factores. No obstante, tal y como se puede observar en la tercera columna, emplear un valor elevado de N lleva a disminuir la precisión de la característica en cuanto a regiones con diferente estructura se refiere.

En resumen, el resultado del análisis de estructura es $RHI_t(\mathbf{x})$ (ver Figura 3.12), una puntuación que se incrementa cuando en el *frame* actual se tiene diferente estructura que en el fondo y que se resetea en caso contrario, evitando así incrementar píxeles pertenecientes a sombras o cambios de iluminación.

Tras observar la Figura 3.12, se puede ver que $RHI_t(\mathbf{x})$ aporta una acumulación muy similar a $FHI_t(\mathbf{x})$, sin embargo esta nueva puntuación no acumula regiones donde existen sombras o cambios de iluminación y proporciona una robustez a camuflajes mayor que $FHI_t(\mathbf{x})$. No obstante, como se verá en la evaluación, incluir $FHI_t(\mathbf{x})$ en el esquema proporciona mejores

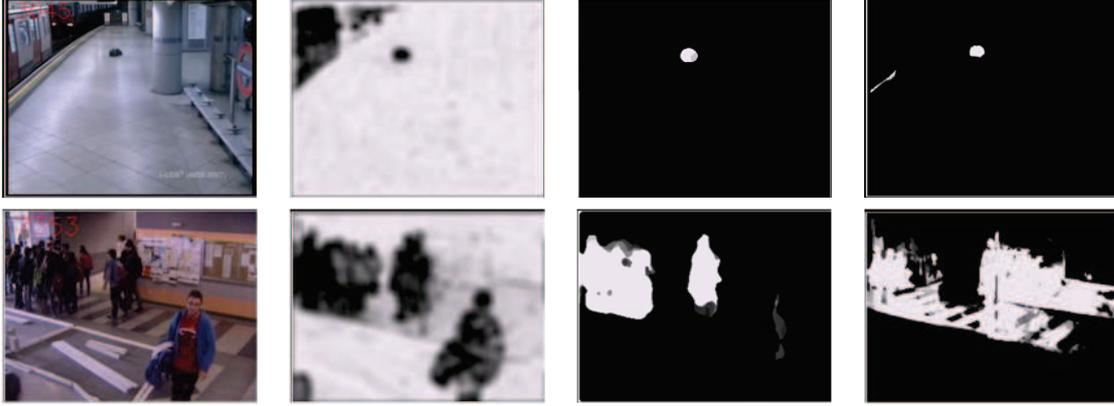


Figura 3.12: De izquierda a derecha: $frame$, $RSSIM_t(\mathbf{x})$, $RHI_t(\mathbf{x})$ y $FHI_t(\mathbf{x})$. En la primera fila vemos como el mismo cambio de iluminación que acumula $FHI_t(\mathbf{x})$, $RHI_t(\mathbf{x})$ es capaz de evitarlo. Además en la segunda fila se muestra un claro ejemplo en el que la acumulación de $RSSIM_t(\mathbf{x})$ consigue evitar los falsos positivos originados por sombras que tienen lugar en $FHI_t(\mathbf{x})$.

resultados que no hacerlo. Esto se debe a que hay una mayor precisión en la característica de frente que en la de estructura, pues al incluir información a nivel de región en la segunda máscara es algo menos precisa, hecho que repercute a la hora de remover detecciones que dejan de ser estáticas.

3.3. Combinación de análisis

Después de obtener $FHI_t(\mathbf{x})$, $MHI_t(\mathbf{x})$ y $RHI_t(\mathbf{x})$, se lleva a cabo su combinación con el objetivo de obtener una puntuación final. Primero se normalizan al rango $[0, 1]$ las 3 imágenes, considerando el *frame rate* del vídeo (fps) y el tiempo de detección de región estática (t_{static}):

$$\overline{FHI}_t(\mathbf{x}) = \min\{1, FHI_t(\mathbf{x})/(fps \cdot t_{static})\}, \quad (3.24)$$

$$\overline{MHI}_t(\mathbf{x}) = \min\{1, MHI_t(\mathbf{x})/(fps \cdot t_{static})\}. \quad (3.25)$$

$$\overline{RHI}_t(\mathbf{x}) = \min\{1, RHI_t(\mathbf{x})/(fps \cdot t_{static})\}. \quad (3.26)$$

Una vez normalizadas se procede a combinarlas calculando su media y sintetizando lo que llamamos *Stationary History Image* $SHI_t(\mathbf{x})$, que representa la variación temporal de frente-movimiento-estructura. Por último, se tiene que calcular la máscara de regiones estacionarias, que idealmente se debe hacer mediante una umbralización como sigue:

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{si } SHI_t(\mathbf{x}) \geq \eta \\ 0 & \text{resto} \end{cases}, \quad (3.27)$$

donde $\eta \in (0, 1]$ es el umbral para la detección estática. Su valor debe ser alto ($\eta = 1$), si se quiere

respetar el tiempo de detección estática). $\overline{FHI}_t(\mathbf{x})$, $\overline{MHI}_t(\mathbf{x})$ y $\overline{RHI}_t(\mathbf{x})$ deben marcar como región estática el objeto a detectar, pues las tres contribuyen por igual a SHI_t , no siendo posible detectar una región como estacionaria teniendo en cuenta solo una o dos de las puntuaciones.

No obstante, las regiones estáticas continuamente ocluidas no eran detectadas en muchos de los casos debido a que $\overline{MHI}_t(\mathbf{x})$ no es capaz de capturar el no-movimiento ocluido, lo suficiente como para realizar su aportación completa a SHI_t . Por tanto, se ha modificado la detección estática añadiendo una condición adicional que reduce η en píxeles donde $\overline{FHI}_t(\mathbf{x})$ y $\overline{RHI}_t(\mathbf{x})$ alcanzan un valor elevado para poder detectar incrementos de $\overline{MHI}_t(\mathbf{x})$ que no llegan a ser muy elevados por sufrir numerosas oclusiones:

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{si } \overline{FHI}_t(\mathbf{x}) \geq \eta \& \overline{RHI}_t(\mathbf{x}) \geq \eta \& \\ & SHI_t(\mathbf{x}) \geq \eta \cdot factorTh \\ 0 & \text{otherwise} \end{cases}, \quad (3.28)$$

donde $factorTh \in (0, 1)$ pondera el umbral η . Debe tener un valor alto (bajo) para secuencias que presentan un bajo (alto) movimiento. Eq. 3.28 permite rebajar el umbral de detección estática para píxeles marcados con un alto valor de *Foreground History Image* ($\overline{FHI}_t(\mathbf{x}) \geq \eta$) y *Region History Image* ($\overline{RHI}_t(\mathbf{x}) \geq \eta$), es decir, para píxeles candidatos a ser estáticos sin considerar el movimiento. De esta manera se consigue detectar regiones estáticas que sufren continuas oclusiones, pues para aquellas zonas donde haya una región estática con movimiento delante, $\overline{MHI}_t(\mathbf{x})$ aportará mayor puntuación que si hay movimiento sin objeto gracias a la extracción del movimiento realizada. En cualquier caso η debe ser superior a $\frac{2}{3}$ para no permitir una detección basada solo en dos de las tres puntuaciones. Además, con las condiciones impuestas, sigue siendo η quien se encarga del tiempo de aparición de las regiones estacionarias, pues no permite la activación de alarma sin que $\overline{FHI}_t(\mathbf{x})$ y $\overline{RHI}_t(\mathbf{x})$ alcancen su valor (sin reducción). En resumen, se emplea un esquema con umbral reducido (Eq. 3.28) para poder detectar regiones estáticas ocluidas constantemente, obteniendo así una máscara de regiones estáticas como la mostrada en la Figura 3.13.

3.4. Manejo de oclusiones

Tras detectar las regiones estáticas, puede tener lugar una reducción de los valores de $\overline{MHI}_t(\mathbf{x})$ como consecuencia de oclusiones parciales o totales, lo que lleva a una reducción de los valores de $SHI_t(\mathbf{x})$ y en consecuencia a no satisfacer las condiciones de la ecuación 3.28. Por tanto, se ha añadido un método de manejo de oclusiones para recuperar detecciones perdidas debido a la situación expuesta. Para cada píxel se comprueban las siguientes condiciones y se propagan las detecciones como sigue:



Figura 3.13: Primera fila, de izquierda a derecha, $frame$, $SHI_t(\mathbf{x})$ y $SFG_t(\mathbf{x})$. Segunda fila: $FHI_t(\mathbf{x})$, $MHI_t(\mathbf{x})$ y $RHI_t(\mathbf{x})$. Se puede ver como al umbralizar la acumulación temporal de frente-movimiento-estructura detecta las regiones estáticas con robustez a falsas detecciones producidas por sombras o cambios de iluminación (borde del andén, señal de metro en lado derecho o sombra de la mujer sentada en el banco).

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{si } SFG_{t-1}(\mathbf{x}) = 1 \& \\ & \overline{FHI}_t(\mathbf{x}) \geq \eta \cdot factorOc \& \overline{RHI}_t(\mathbf{x}) \geq \eta , \\ 0 & \text{resto} \end{cases} \quad (3.29)$$

donde $factorOc \in (0, 1)$ es una tolerancia a errores de la característica de frente (p.ej., camuflajes) que reducen $FHI_t(\mathbf{x})$ y causan la pérdida de detecciones. Esta recuperación se emplea cuando los píxeles tienen unas puntuaciones elevadas de frente y estructura ($\overline{FHI}_t(\mathbf{x}) \geq \eta \cdot factorOc$ y $\overline{RHI}_t(\mathbf{x}) \geq \eta$) y en el instante anterior pertenecían a regiones estáticas ($SFG_{t-1}(\mathbf{x}) = 1$), compensando así las reducciones de $\overline{MHI}_t(\mathbf{x})$ descritas (ver Figura 3.14). Cuanto más pequeño sea $factorOc$, mayor robustez a errores de frente pero menor precisión en la desaparición de detecciones estáticas que ya no se encuentran en la escena. A diferencia de otros trabajos,[1][7], que llevan a cabo un análisis a nivel de *blob*, aquí se trata la situación a nivel de píxel, pues se obtienen resultados satisfactorios tal y como muestra la Figura 3.14.



Figura 3.14: Detección estática con y sin manejo de oclusiones. En la primera fila se muestran *frames* bajo análisis y en la segunda sus máscaras estáticas correspondientes. Por columnas (de izquierda a derecha): instante 2429, instante 2498 sin recuperación de oclusiones e instante 2498 con recuperación de oclusiones. La máscara de la segunda columna muestra una situación de movimiento donde la característica de movimiento no es suficiente para mantener el objeto ocluido y en consecuencia se pierde al descender la acumulación $MHI_t(\mathbf{x})$. No obstante aplicando la recuperación a oclusiones (tercera columna), se consigue mantener la detección pues los píxeles del objeto eran estáticos en instantes anteriores y $FHI_t(\mathbf{x})$ y $RHI_t(\mathbf{x})$ siguen siendo elevados.

3.5. Configuración del algoritmo

Para llevar a cabo la configuración de la aproximación, se han utilizado los parámetros habituales de otras aproximaciones de estado del arte: *frame rate* (25 fps) y tiempo de detección estática ($t_{static} = 20$ segundos). Además un funcionamiento correcto del algoritmo obliga a utilizar: $w_{pos}^f = w_{pos}^m = w_{pos}^r = 1$, $w_{neg}^m = MHI_{t-1}(\mathbf{x})$, $w_{neg}^s = RHI_{t-1}(\mathbf{x})$ y $\eta = 1$ para garantizar que ninguna detección aparece antes de t_{static} . Idealmente se querría tener $w_{neg}^f = FHI_{t-1}(\mathbf{x})$, pero errores en la máscara de frente (p. ej., camuflajes) impiden utilizar esta configuración. En su lugar, para tolerar estos errores, se ha fijado empíricamente un valor de $w_{neg}^f = 15$ y un nuevo parámetro $factorOc = 0,8$ (una mejora en la sustracción de fondo podría evitar este parámetro y fijar $w_{neg}^f = 15$ a su valor ideal). Por último, tras realizar pruebas en entornos altamente concurridos, observamos una caída de entre 50-25% de la aportación total de $MHI_t(\mathbf{x})$ para $t_{static} = 10-20$ seg, lo cual llevó a establecer un doble umbral con el parámetro $factorTh = \frac{\eta}{3} + \frac{\eta}{3} + (\frac{\eta}{3} \times 0,27)$ (siendo las dos $\frac{\eta}{3}$ las contribuciones de FHI_t y RHI_t y la ponderación del 27% la contribución de MHI_t) para poder realizar detecciones en situaciones de intenso movimiento.

En consecuencia se tiene un valor de $factorTh = 0,756$. El valor de la ventana temporal se fijó empíricamente como $T = 10$, para mantener una buena captura de no-movimiento tras las oclusiones y no introducir demasiado retardo. Por último, como vecindario para el cálculo de *RSSIM* se emplea una ventana de 12×12 ($N = 12$), pues tras realizar pruebas se eliminaban las mayoría de las falsas detecciones causadas por zonas saturadas de un tamaño no muy elevado. En cuanto a la actualización del modelo de fondo, se ha decidido emplear una política de no actualización para evitar problemas con objetos estáticos temporales. De esta manera, las únicas detecciones fantasma que aparecen son las debidas a una mala captura del fondo (al removerse objetos que están en él).

En resumen, obviando los parámetros del algoritmo de sustracción de fondo, los parámetros variables en este algoritmo para conseguir un buen funcionamiento son: w_{neg}^f , $factorOc$, $factorTh$, T , N . Los valores comentados son los utilizados en todos los experimentos del capítulo 4.

Capítulo 4

Resultados experimentales

En este capítulo se muestran los resultados experimentales obtenidos para el algoritmo presentado en el capítulo 3. Las pruebas se han llevado a cabo tanto en entornos sencillos como complejos (alta concurrencia de objetos y personas). Para la implementación de todos los algoritmos, se ha empleado la librería pública de tratamiento de imágenes OpenCV¹. Los test realizados se han ejecutado en un Pentium(R) D que opera a 2.8 GHz y 2 GB de RAM.

El capítulo está dividido en 3 apartados: la sección 4.1 presenta los *datasets* disponibles para la evaluación del algoritmo, la sección 4.2 explica los criterios empleados en la evaluación y la sección 4.3 muestra los resultados de la evaluación del algoritmo propuesto frente a varias aproximaciones del estado del arte.

4.1. *Datasets* disponibles

Este apartado describe los *datasets* públicos utilizados para evaluación:

- PETS2006 (<http://www.cvg.rdg.ac.uk/PETS2006/data.html>): las secuencias de vídeo-seguridad de este escenario contienen escenas de abandono de objetos en situaciones sencillas cuya dificultad aumenta de manera progresiva. El *dataset* está formado por siete secuencias capturadas desde 4 cámaras diferentes (ver Figura 4.1).
- PETS2007 (<http://pets2007.net>): este *dataset* está formado por 9 escenas de robo y abandono visionadas desde 4 perspectivas distintas (ver Figura 4.2). Las secuencias tienen un nivel de complejidad que aumenta progresivamente.
- AVSS2007 (<http://www.eecs.qmul.ac.uk/~andrea/avss2007.html>): este *dataset* cuenta con tres secuencias de complejidad creciente para cada escenario: abandono de objetos en

¹<http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.4.6/OpenCV-2.4.6.0.exe/download>



Figura 4.1: Ejemplo de las 4 perspectivas disponibles en PETS2006.



Figura 4.2: Ejemplo de las 4 perspectivas disponibles en PETS2007.



Figura 4.3: Ejemplo de los dos escenarios disponibles en AVSS2007: abandono (izquierda) y vehículos aparcados ilegalmente (derecha).

estaciones de metro y vehículos estacionados ilegalmente (ver Figura 4.3). Además existen dos vídeos adicionales de larga duración y elevada complejidad para cada escenario.

Para evaluar los algoritmos se han utilizado 15 secuencias de los *datasets* PETS2006, PETS2007 y AVSS07, por contener escenas altamente concurridas aptas para la evaluación deseada. No obstante, los *datasets* anteriores no cuentan con un gran número de situaciones con elevada concurrencia de sombras que deterioran enormemente el rendimiento de las aproximaciones del estado del arte. En consecuencia, se ha añadido a la evaluación 3 secuencias más, extraídas de 3 horas y media de vídeo grabado en el hall de la universidad (ver Figura 4.4). Por tanto, se han evaluado los algoritmos en 18 secuencias de vídeo (147146 frames, 127 objetos anotados), donde se producen una gran variedad de situaciones (ver Tabla 4.1). Para poder llevar cabo el test, se ha elaborado un *ground-truth* (GT) en el que se han anotado los instantes de parada del objeto y fin de alarma, el tipo de detección (persona u objeto) y una breve descripción del evento.



Figura 4.4: Ejemplo de las 3 secuencias grabadas.

Criterios	Escenarios sencillos (<i>Non-crowded</i>)						
	AVSS07		PETS06			PETS07	TOTAL
	Easy	S7_C3	S4_C3	S4_C4	S5_C3		
Inicialización del fondo	L	L	L	L	L		
Cambios de iluminación	L	-	-	-	M		
Nivel de movimiento	L	L	L	L	L		
Complejidad total	L	L	L	L	L		
Número de frames	4291	3401	3051	3051	2900	16694	
Fondo correcto al inicio	No	No	No	No	No		
Regiones anotadas	2	1	3	4	2	12	

Criterios	Escenarios complejos (<i>Crowded</i>)															TOTAL
	AVSS07				PETS07		PETS06						HALL			
	Med	Hard	AB	PV	S5_C1	S5_C2	S7_C1	S7_C4	S1_C1	S1_C4	S4_C1	S4_C2	H_S1	H_S2	H_S3	
Inicialización del fondo	L	L	H	L	H	H	H	M	H	M	H	H	H	M	H	-
Cambios de iluminación	L	L	L	H	M	M	-	-	-	-	-	-	L	L	L	-
Nivel de movimiento	M	H	H	H	H	H	H	M	H	L	H	H	H	M	H	-
Complejidad total	H	H	H	H	H	H	H	M	H	M	H	H	H	M	H	-
Número de frames	4834	5311	32875	26750	2900	2900	3401	3401	3021	3021	3051	3051	10000	10834	15102	130452
Fondo correcto al inicio	Sí	Sí	Sí	No	No	No	No	No	No	No	No	No	No	Sí	Sí	-
Regiones anotadas	14	13	39	10	3	3	1	1	2	3	6	3	3	2	12	115

Tabla 4.1: Descripción de las secuencias empleadas en la evaluación y regiones estáticas anotadas. (Key. L:Low. M:Medium. H:High).

4.2. Métrica

Para evaluar el rendimiento del algoritmo propuesto, se ha decidido emplear las medidas estándar de *Precision* (P), *Recall* (R) y *F-score* (F):

$$P = TP/(TP + FP), \quad (4.1)$$

$$R = TP/(TP + FN), \quad (4.2)$$

$$F = 2 \cdot P \cdot R/(P + R), \quad (4.3)$$

donde TP, FP y FN indican detecciones correctas, incorrectas y perdidas respectivamente. Estas detecciones se han contabilizado visualmente, considerando los siguientes criterios:

- Como retraso de aparición y desaparición de regiones estáticas se ha empleado un margen de 150 *frames* (6 segundos a 25fps). Si se sobrepasa este margen en la desaparición de objetos removidos entonces se contabiliza un falso positivo (FP). Mientras que si el objeto sobrepasa el margen de inicio se considera que la detección no se realiza (FN).
- Si la detección no es constante durante el tiempo de inicio y fin de alarma (más allá de las tolerancias mostradas), entonces se considera que la detección no se realiza (FN).
- Para considerar un *blob* con detecciones falsas y correctas conexas como detección válida, se exige que se cumpla de manera aproximada la condición de *spatial-overlap* (SO) superior a 0.5 entre la detección del GT y la detección realizada (DT), siendo:

$$SO = \frac{2(GT \cap DT)}{|GT| + |DT|} \quad (4.4)$$

En caso contrario, se contabiliza un FN.

- Se ha tenido cierta flexibilidad en la detección de personas, pues éstas no permanecen completamente estáticas. Esta variación provoca cambios del *blob* a lo largo del tiempo, no siendo siempre igual a la máscara anotada en el GT.

4.3. Resultados

La Tabla 4.2 muestra una comparativa del rendimiento obtenido por la combinación de las distintas características empleadas en el algoritmo propuesto. Como se puede observar la gran mejora viene producida por la característica de estructura pues permite remover falsos positivos originados por sombras y cambios de iluminación. Además, incorporar información de movimiento también supone un incremento, aunque inferior, pues la buena configuración de la acumulación de frente y de estructura (fuerte decremento de FHI_t y reseteo de RHI_t) consigue reducir enormemente los problemas. En la Figura 4.5, se muestran 3 ejemplos de la aportación de cada característica.

En la Tabla 4.3 se muestra la comparativa de la aproximación propuesta con varios algoritmos del estado del arte. Se puede observar como se consigue mejorar en las 5 secuencias sencillas

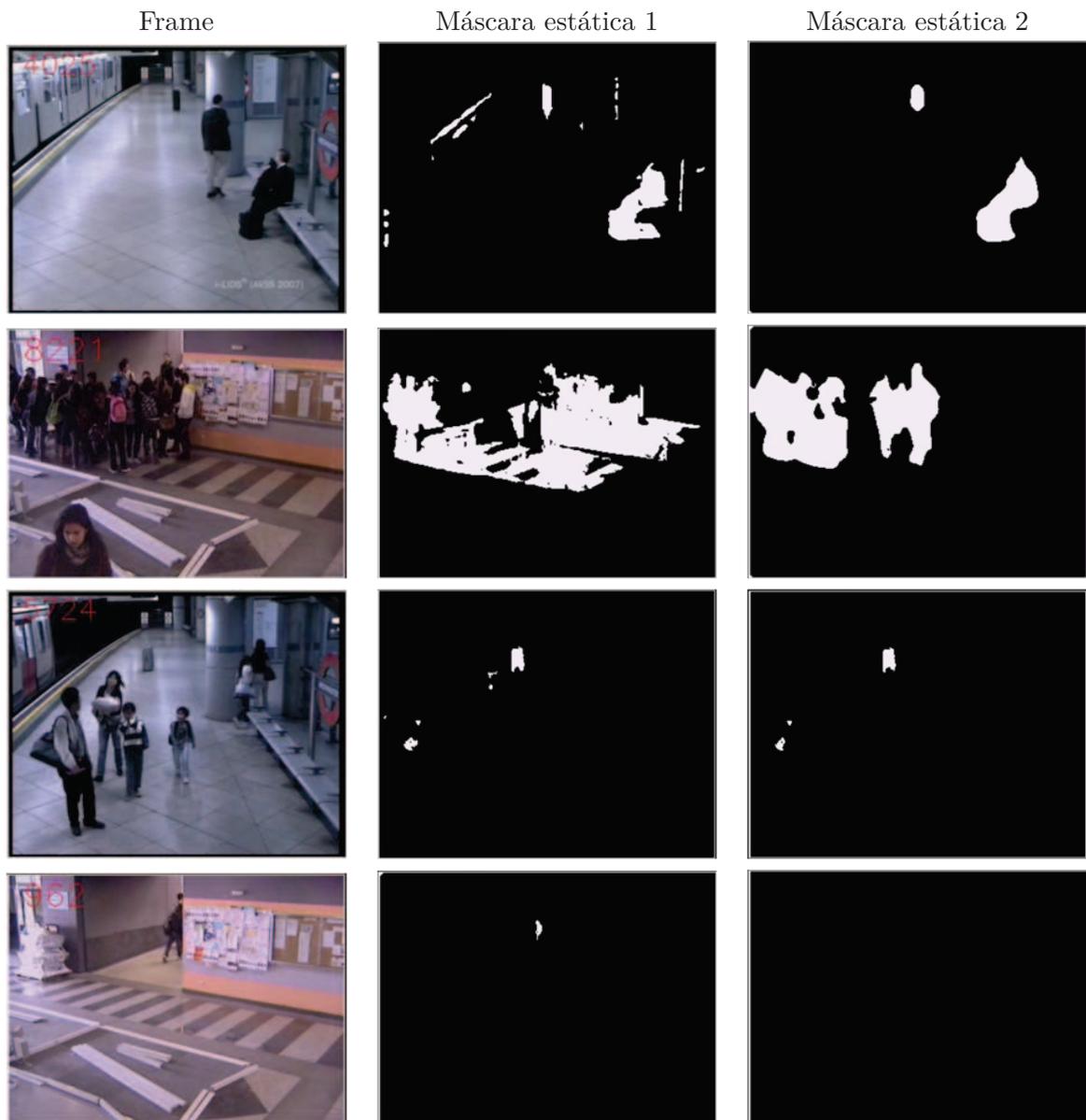


Figura 4.5: Comparativa de características para mostrar la aportación individual de cada una. Primera y segunda fila (de izquierda a derecha): *frame* bajo análisis, máscara estática obtenida con la característica de frente y máscara estática obtenida mediante la característica de estructura. En ambas filas se muestra como la característica de estructura aporta un resultado similar a la característica de frente pero eliminando los *blobs* causados por sombras o cambios de iluminación. Tercera fila, de izquierda a derecha, *frame* bajo análisis, máscara estática para la característica de frente y máscara estática para la combinación de frente y movimiento. En la máscara de la segunda columna aparecen dos *blobs* a la izquierda de la maleta detectada originados por un intenso movimiento en esa zona en frames anteriores, que en la tercera columna son removidos gracias al añadido de movimiento (los *blobs* que perduran además de la detección, se deben a cambios de iluminación). Por último en la cuarta fila se muestra, de izquierda a derecha, *frame* bajo análisis, máscara estática para la combinación de estructura y movimiento y máscara estática de la combinación de las tres características. En este ejemplo se aprecia como incluir la característica de frente aporta eliminar un *blob*⁵¹ que mantiene la característica de estructura por ser menos precisa.

Característica/as	Escenarios sencillos (<i>Non-crowded</i>)					
	AVSS07	PETS06			PETS07	Mean
	Easy	S7_C3	S4_C3	S4_C4	S5_C3	
Frente	P .33	1	1	.80	.50	.73
	R 1	1	1	1	1	1
	F .50	1	1	.89	.66	.81
Estructura	P .40	1	.75	1	.50	.73
	R .1	1	1	1	1	1
	F .57	1	.86	1	.66	.82
Frente y movimiento	P .33	1	1	.80	.50	.73
	R 1	1	1	1	1	1
	F .50	1	1	.89	.66	.81
Estructura y movimiento	P 0.67	1	1	1	.50	.83
	R 1	1	1	1	1	1
	F .80	1	1	1	.66	.89
Frente, movimiento y estructura	P .67	1	1	1	1	.93
	R 1	1	1	1	1	1
	F .80	1	1	1	1	.96

Característica/as	Escenarios complejos (<i>Crowded</i>)															Media
	AVSS07				PETS07		PETS06						HALL			
	Med	Hard	AB	PV	S5_C1	S5_C2	S7_C1	S7_C4	S1_C1	S1_C4	S4_C1	S4_C2	H_S1	H_S2	H_S3	
Frente	P .74	.72	.57	.12	.17	.30	.06	.14	.12	.50	.27	.06	.40	.67	.18	.33
	R 1	1	1	.60	1	1	1	1	1	1	.67	.33	.67	1	.33	.84
	F .85	.84	.73	.20	.29	.46	.12	.25	.22	.67	.38	.10	.50	.80	.23	.44
Estructura	P .87	.92	.67	.24	.19	.33	.17	.25	.33	.50	.57	.25	.60	.67	.44	.47
	R 1	.92	1	1	1	1	1	1	1	.67	.67	.33	1	1	1	.91
	F .93	.92	.80	.38	.32	.50	.29	.40	.50	.57	.61	.29	.75	.80	.61	.58
Frente y movimiento	P .74	.72	.72	.12	.17	.27	.1	.16	.13	.50	.41	.07	.40	1	.31	.39
	R 1	1	.97	.60	1	.66	1	1	1	1	.83	.33	.67	1	.42	.83
	F .85	.84	.83	.20	.29	.39	.18	.28	.23	.67	.55	.10	.50	1	.36	.48
Estructura y movimiento	P .87	.92	.74	.24	.19	.33	.20	.25	.33	.50	.57	.25	.60	1	.54	.50
	R 1	.92	1	1	1	1	1	1	1	.67	.67	.33	1	1	1	.91
	F .93	.92	.85	.38	.32	.50	.33	.40	.50	.57	.61	.29	.75	1	.71	.60
Frente, movimiento y estructura	P .93	.92	.84	.29	.20	.33	.25	.33	.33	.50	.57	.25	.50	1	.73	.53
	R 1	.92	.95	1	1	1	1	1	1	.67	.67	.33	.67	1	.92	.87
	F .96	.92	.89	.44	.33	.50	.40	.50	.50	.57	.61	.29	.57	1	.82	.62

Tabla 4.2: Comparativa de las diferentes características utilizando *Precision* (P), *Recall* (R) y *F-score* (F). Resultados en negrita indican el mejor rendimiento.

con una incremento medio del *F-score* de 17%. Para las 15 secuencias en entornos complejos, se consigue mejorar en 13 de las 15, consiguiendo una mejora global del *F-score* de en torno al 44%. Las 2 secuencias en las que desciende el rendimiento es debido a un descenso de la puntuación

R (por perder alguna detección del GT). Si se observan las puntuaciones R y P, en ocasiones descendiendo R por perder alguna detección pequeña, no obstante el gran incremento conseguido en P, al tratar los cambios de iluminación e intenso movimiento que acontece en entornos muy concurridos, consigue una mejora sustancial del *F-score*. La mayoría de problemas que continua habiendo y que merman el rendimiento se deben a una mala inicialización del fondo, pues si se atiende a los resultados obtenidos para las secuencias complejas en las que se tiene el fondo correcto, se puede observar un rendimiento muy elevado: 96 % en Med, 92 % en Hard, 89 % en AB, 100 % en H_S2 y 82 % en H_S3.

En la Figura 4.6 se observan ejemplos del buen funcionamiento de la aproximación propuesta frente al estado del arte. En la primera columna se muestra como el algoritmo propuesto es capaz de eliminar las falsas detecciones por movimiento (las que aparecen a la izquierda de la maleta en todas las aproximaciones salvo en [7] y la propuesta) y , a diferencia de [7], mantener la detección de la maleta que ha sido ocluida en instantes anteriores de forma continuada. Además se filtran las falsas detecciones por cambios rápidos de iluminación que tienen lugar en la escena y que el resto de métodos no trata. En contraposición, la detección (muy fina) de la persona que está tras la columna se pierde debido a la información de región considerada. En la segunda columna se muestra otro ejemplo de una situación en la que falsos positivos por movimiento y cambios de iluminación, presentes en las aproximaciones del estado del arte, son filtrados satisfactoriamente con el método propuesto. En la tercera columna se muestra un ejemplo de cómo se elimina una sombra (presente en todas las aproximaciones salvo en la propuesta) y de cómo errores en la inicialización de fondo son reducidos enormemente (sin haberse buscado este efecto) como consecuencia de la información de movimiento y de estructura empleada.

En la Figura 4.7 se muestran más ejemplos del buen funcionamiento del algoritmo. En la primera columna se observa como el algoritmo propuesto elimina las falsas detecciones por cambios de iluminación (zona izquierda de la imagen) que aparecen en el resto de algoritmos del estado del arte evaluados. En la segunda columna se muestran casos en los que nuestra aproximación no suprime completamente *blobs* ocasionados por cambios de iluminación (zona derecha de la imagen) debido a la saturación de luminancia producida. La detección de la mochila no aparece en [17] porque ha sido absorbida por el modelo de fondo y en consecuencia se considera que se ha perdido. En la tercera columna se muestra una escena altamente concurrida donde la aproximación propuesta es la única capaz de detectar de forma adecuada las regiones estáticas evitando los numerosos falsos positivos del estado del arte, que no son robustos a los rápidos cambios de iluminación en la escena por la elevada concurrencia de personas (y sus sombras).

En la Figura 4.8 se muestran más ejemplos del buen funcionamiento del algoritmo. En la

Aproximación	Escenarios sencillos (<i>Non-crowded</i>)						
	AVSS07	PETS06			PETS07	Mean	
	Easy	S7_C3	S4_C3	S4_C4	S5_C3		
[12]	P	.29	1	1	.80	.50	.72
	R	1	1	1	1	1	1
	F	.44	1	1	.89	.66	.80
[16]	P	.29	1	.75	.80	.50	.67
	R	1	1	1	1	1	1
	F	.44	1	.85	.89	.66	.77
[7]	P	.40	1	1	.80	.50	.74
	R	1	1	1	1	1	1
	F	.57	1	1	.89	.66	.82
[17]	P	.33	.33	.60	.50	.33	.42
	R	1	1	1	1	1	1
	F	.50	.50	.75	.67	.50	.58
Proposed	P	.67	1	1	1	1	.93
	R	1	1	1	1	1	1
	F	.80	1	1	1	1	.96
%Δ best	P	67.5	0	0	25	100	25.7
	R	0	0	0	0	0	0
	F	40.3	0	0	12.3	51.5	17.1

Aproximación	Escenarios complejos (<i>Crowded</i>)																
	AVSS07				PETS07		PETS06						HALL			Media	
	Med	Hard	AB	PV	S5_C1	S5_C2	S7_C1	S7_C4	S1_C1	S1_C4	S4_C1	S4_C2	H_S1	H_S2	H_S3		
[12]	P	.61	.48	.51	.12	.16	.23	.05	.12	.12	.42	.29	.05	.43	.67	.15	.29
	R	1	1	1	.60	1	1	1	1	1	.83	.33	1	1	.33	.87	
	F	.76	.65	.68	.20	.28	.37	.10	.22	.22	.60	.43	.10	.60	.80	.23	.42
[16]	P	.56	.52	.42	.12	.17	.25	.05	.12	.12	.42	.29	.07	.25	.25	.17	.25
	R	1	1	1	.60	1	1	1	1	1	.83	.33	1	1	.33	.87	
	F	.72	.68	.59	.20	.29	.40	.11	.22	.22	.60	.43	.10	.40	.40	.22	.37
[7]	P	.60	.58	.78	.12	0	.27	.14	.16	.10	.50	.38	.09	.40	.67	.31	.34
	R	.85	.76	.82	.60	0	.66	1	1	.50	1	.83	.33	0.67	1	.42	.70
	F	.70	.66	.80	.20	0	.39	.25	.28	.16	.67	.52	.14	.50	.80	.36	.43
[17]	P	.36	.60	.47	.20	.14	.21	.07	.08	.10	.43	.22	.10	.17	.33	.09	.24
	R	1	.92	.61	.90	.67	1	1	1	1	.67	.67	.33	.50	.08	.76	
	F	.53	.73	.53	.32	.23	.35	.12	.14	.19	.60	.33	.18	.22	.40	.09	.33
Proposed	P	.93	.92	.84	.29	.20	.33	.25	.33	.33	.50	.57	.25	.50	1	.73	.53
	R	1	.92	.95	1	1	1	1	1	.67	.67	.33	.67	1	.92	.87	
	F	.96	.92	.89	.44	.33	.50	.40	.50	.50	.57	.61	.29	.57	1	.82	.62
%Δ best	P	52.5	53.3	7.7	45	17.6	22.2	78.6	106.2	175	0	50	150	16.3	49.2	135.5	55.9
	R	0	-8.7	-5.3	11.1	0	0	0	0	-49.2	-23.9	-100	-49.2	0	119	0	
	F	26.3	26	11.2	37.5	13.8	25	60	78.6	127.3	-17.5	17.3	61.1	-5.2	25	127.8	44.2

Tabla 4.3: Comparativa de resultados del algoritmo propuesto frente a técnicas del estado del arte, utilizando *Precision* (P), *Recall* (R) y *F-score* (F). Los resultados en negrita indican el rendimiento mayor.

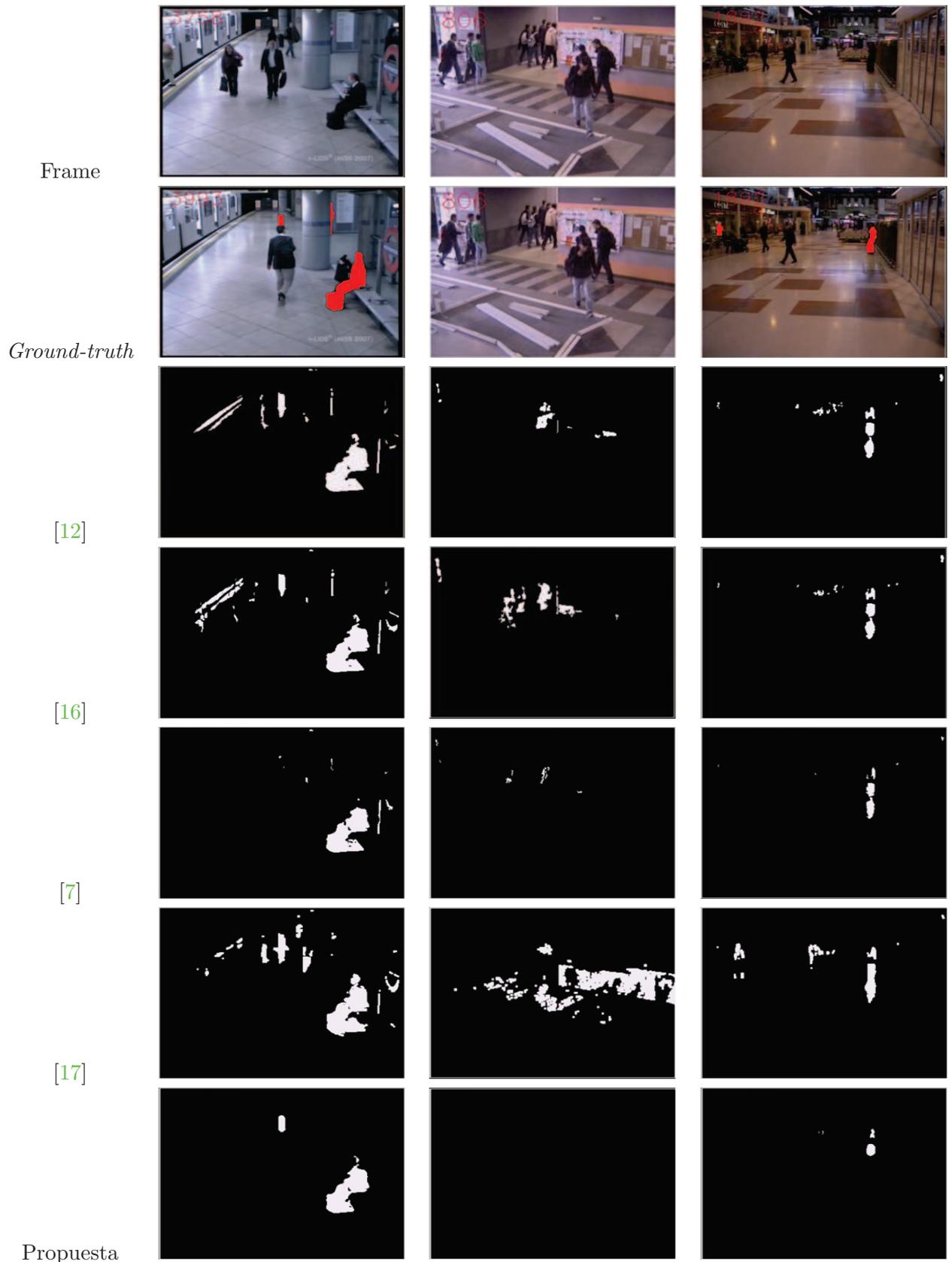


Figura 4.6: Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (1).

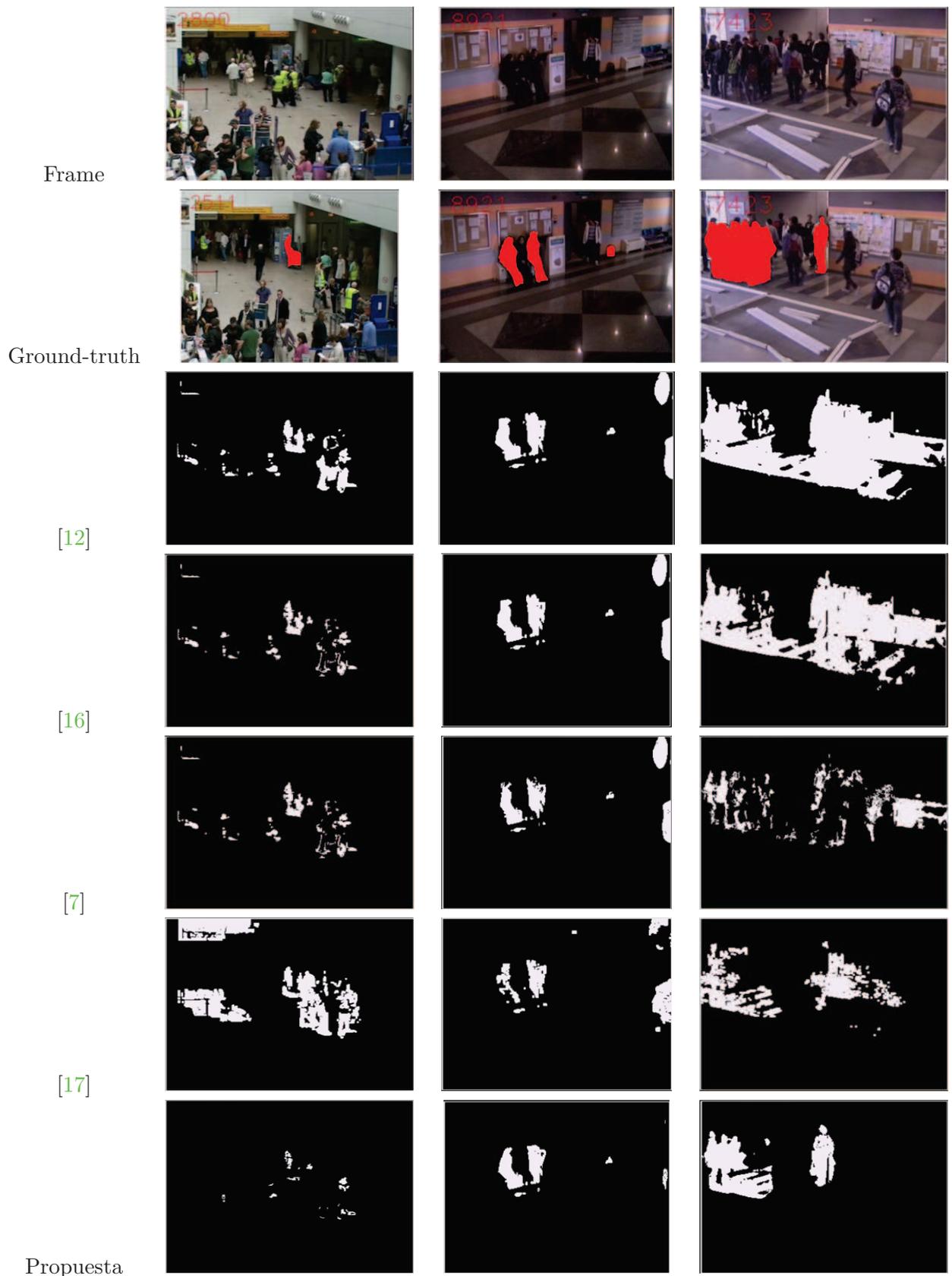


Figura 4.7: Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (2).

primera columna se observa como el algoritmo propuesto elimina las falsas detecciones por cambios de iluminación (zona izquierda de la imagen) que aparecen en el resto de algoritmos del estado del arte evaluados (salvo en [7] donde también se eliminan pero la detección de la maleta no se realiza). En la segunda columna se muestra un ejemplo donde se está teniendo un movimiento continuado en una región de manera que la información de movimiento de la propuesta (y también de [7]) son capaces de eliminar las falsas detecciones pero no así el resto de aproximaciones (las detecciones no aparecen en [17] porque han sido absorbidas por el modelo de fondo). En la tercera columna se muestra una escena donde el *frame* actual ha sufrido un cambio de iluminación con respecto al fondo que se capturó de manera que aparecen numerosos falsos positivos que solo la aproximación propuesta es capaz de evitar. En la última columna, las falsas detecciones que aparecen el método propuesto se deben a errores en la captura del fondo (compartidos con el resto de aproximaciones).

Si bien se ha conseguido mejorar en varios aspectos, la aproximación desarrollada continua teniendo situaciones que no es capaz de tratar de forma correcta:

- El algoritmo falla en áreas donde, una vez se tiene FG o SSIM al máximo por sufrir intenso movimiento (pues siempre hay objetos en una posición en la que no se visiona el fondo en ningún momento), se queda algo estático lo suficiente como para que MHI_t sobrepase el 27% exigido por el $factorTh$ para la detección estática. Es decir, si en una zona con intenso movimiento donde se está empleando esta información para filtrar, se queda algo estático la detección llegará antes de tiempo, o en caso de que un objeto se quede parado menos tiempo del considerado como estático se realizará una detección incorrecta al tener FHI_t y RHI_t con un valor máximo (ver primera fila de la Figura 4.9).
- Aunque la robustez a cambios de iluminación es muy buena, algunas situaciones no se resuelven completamente como pueden ser grandes cambios de iluminación que se dan habitualmente en entornos al aire libre (ver segunda fila de la Figura 4.9).
- Los errores del modelo inicial de fondo no se solucionan (ver tercera fila de la Figura 4.9).
- No hay una robustez total a oclusiones de objetos que sufren camuflaje. Esto se debe, en general, a los errores de FHI_t (ver cuarta fila de la Figura 4.9).

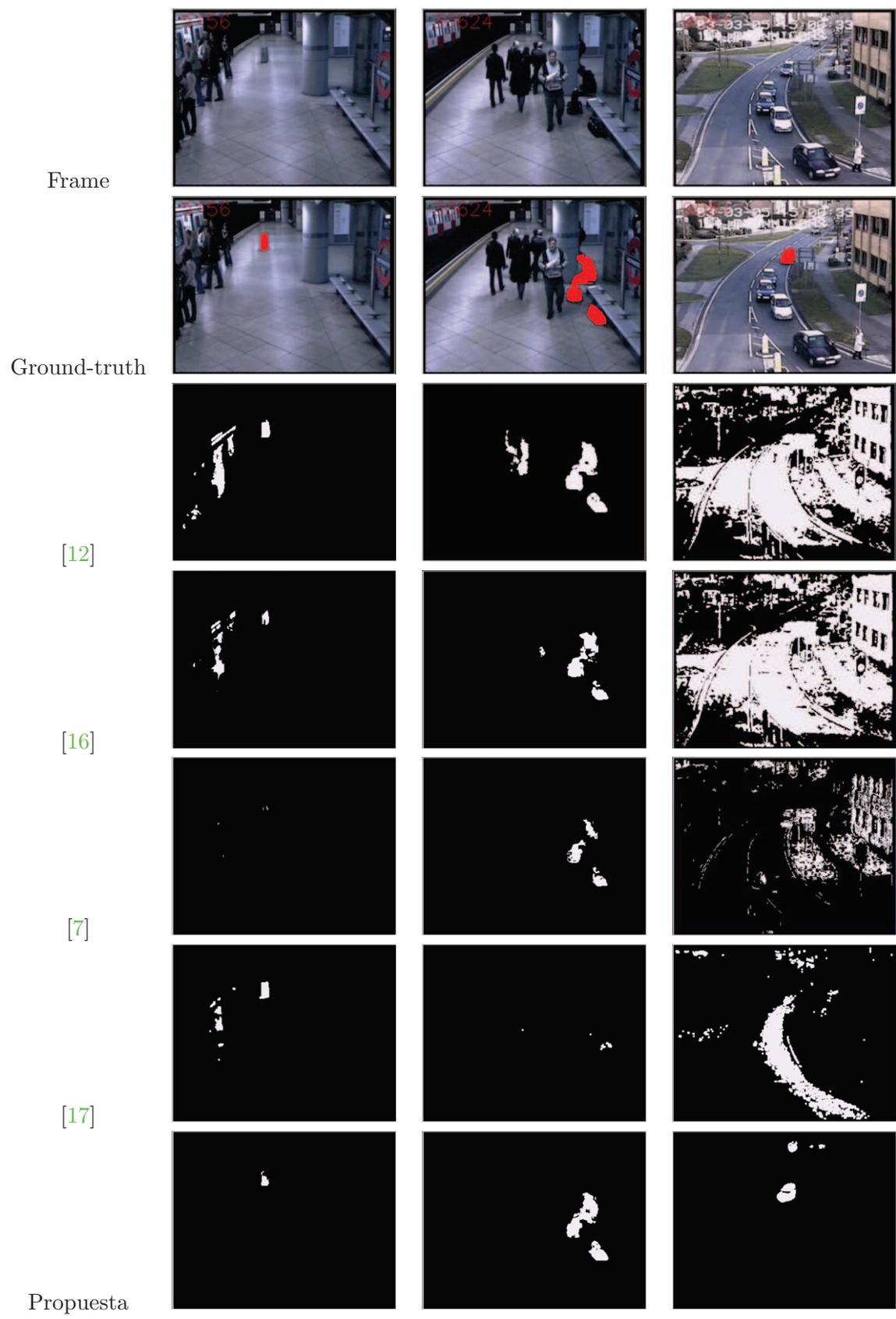


Figura 4.8: Máscaras estáticas de los algoritmos del estado del arte y el algoritmo propuesto (3).



Figura 4.9: Ejemplo de errores del algoritmo propuesto. De izquierda a derecha (salvo segunda fila): *frame* bajo análisis, modelo de fondo y máscara estática. La segunda fila la forman: *frame* bajo análisis, *frame* con objeto a detectar, máscara de frente y máscara estática.

Capítulo 5

Conclusiones y trabajo futuro

5.1. Conclusiones

En este PFC se ha presentado un algoritmo de detección de regiones estáticas capaz de operar en entornos altamente poblados por su robustez a oclusiones, cambios de iluminación, sombras y situaciones de elevado movimiento.

En primer lugar, se realizó un estudio detallado del estado del arte de los sistemas de vídeo-seguridad orientados a la detección de regiones estáticas. De este estudio se extrajeron las aproximaciones más importantes (la mayoría basadas en una etapa de sustracción de fondo), sus problemas y las estrategias empleadas en el estado del arte para resolverlos. Se observó que las aproximaciones más recientes combinan múltiples análisis de la secuencia para detectar regiones estacionarias y que, en su mayoría, son evolución de otras técnicas anteriores y más sencillas que emplean un único análisis de la escena.

Tras examinar algunas de las aproximaciones más recientes para la sustracción de fondo¹, se observó que se trata de un campo con un gran margen de mejora debido a la extendida incapacidad para adaptarse a los cambios de iluminación del fondo (sobre todo a los cambios rápidos) y mantener detecciones estáticas en el proceso. Este aspecto motivó la utilización de un modelo sencillo, con bajo coste computacional y robusto al ruido, el segmentador Gamma.

Otro problema encontrado es la inicialización del modelo de fondo, que se realizó en caliente, es decir, estableciendo como fondo los valores de los píxeles que se mantuvieran invariables durante un cierto tiempo (en algunas secuencias se añadió el fondo correcto al principio). Muchos trabajos obvian este punto, considerando que es algo de lo cual se dispone. Esto es aceptable hasta cierto punto pues lo ideal sería poder disponer de un fondo que el *software* actualice para poder reflejar los cambios, tanto físicos como de iluminación, que se den en la escena. Se decidió no actualizar el fondo inicial obtenido, para evitar cualquier pérdida de detecciones o

¹<http://changedetection.net/>

detección fantasma surgida por la absorción de un objeto por parte del fondo y su posterior desaparición. Sin embargo, la inicialización llevada a cabo no es aceptable en entornos y en entornos densamente poblados supone disponer de numerosos errores que se traducen en falsas detecciones en la máscara estática.

Una vez obtenidas las regiones de frente se pasó a realizar las detecciones estáticas. Del estudio del estado del arte se observó que los métodos que realizan una acumulación del frente en *frames* sucesivos son muy robustos a oclusiones si se configuran adecuadamente. No obstante, había un problema con el movimiento continuo en zonas de paso, al que se dio solución mediante un análisis del mismo en intervalos temporales adyacentes al *frame* bajo análisis y una reducción del umbral estático bajo determinadas condiciones. Con este esquema de extracción de movimiento se ha conseguido, a diferencia de lo que ocurría en el estado del arte mediante el empleo del *frame-difference*, mantener detecciones que sufren oclusiones de manera frecuente (hasta cierto límite). Además este esquema combinado con la reducción del umbral estático que se realiza (necesaria para la detección de objetos ocluidos continuamente), permite la eliminación de falsos positivos y el mantenimiento de tiempos de aparición y detecciones en zonas con movimiento continuado (aspectos que no se mencionan en el estado del arte cuando se emplea información de movimiento).

En este punto, una limitación muy importante que se seguía teniendo eran los cambios de iluminación rápidos y lentos. Para solucionar este aspecto se decidió llevar a cabo una estrategia parecida a otros trabajos, es decir, ayudarse de información de estructura para determinar si el *frame* bajo análisis y el fondo son iguales y remover en consecuencia aquellos píxeles donde se confirme este aspecto. La gran mejora conseguida viene de la mano de esta información, que emplea una medida de correlación de dos señales (independientes de luminancia y contraste) para calcular la información de estructura y ha demostrado ser muy eficaz. No obstante, se está utilizando una técnica ya desarrollada que tiene ciertas limitaciones. Por ejemplo, aunque la información de estructura es independiente de luminancia y contraste, estas son consideradas en la puntuación de similitud utilizada (pues la técnica está pensada como medida de calidad de imágenes), pudiendo así afectar en la puntuación obtenida.

Para llevar a cabo la evaluación del algoritmo implementado se utilizaron inicialmente *data-sets* públicos destinados a la detección de robo/abandono de objetos en escenarios densamente poblados. Sin embargo, tras observar las secuencias disponibles se vio la necesidad de emplear nuevos vídeos que añadiesen situaciones más complejas de gran concurrencia de personas, por lo que se realizaron grabaciones en el hall de la universidad. Tras evaluar el algoritmo propuesto en comparación con otras propuestas, se confirmó la gran mejora conseguida gracias al triple análisis de la escena realizado. Es importante mencionar que muchas aproximaciones del estado del arte emplean tiempos de evaluación muy elevados que, en algunos casos, suponen evitar importantes problemas (p. ej. se evitan falsos positivos que se deben a cambios rápidos

de iluminación, evitando así considerar la merma que supondrían en el rendimiento). Nuestra decisión en este sentido fue emplear un tiempo estático de 20 segundos para considerar los problemas de cambios rápidos que ocurren en los *datasets* públicos, sin tener un número excesivo de anotaciones que supondrían tiempos aún menores.

Por último, las detecciones fantasma que debidas a una mala inicialización del modelo de fondo, aparecen durante toda la secuencia pues, aunque se han explorado algunas técnicas basadas en análisis de bordes (ver Apéndice A) parecidas a las comentadas en el estado del arte, no se utilizaron al proporcionar malos resultados.

5.2. Trabajo futuro

La detección de regiones estacionarias de primer plano en escenarios densamente poblados continua siendo un reto para futuras líneas de investigación.

En lo que respecta al PFC, se puede incluir más información para mejorar el filtrado de píxeles realizado. Por ejemplo, se pueden analizar los *blobs* para eliminar detecciones fantasma derivadas de la mala inicialización de fondo o realizar un análisis de la apariencia de las detecciones realizadas para saber si hay o no un cambio de objeto. También puede modificarse el método de obtención de la información empleada, por ejemplo analizando las diversas técnicas para extraer información de estructura. Otra posible línea de mejora es trabajar en la inicialización automática del modelo de fondo para generar un modelo correcto en entornos densamente poblados desde un primer momento, evitando así cualquier problema de detecciones fantasma.

Por otro lado, si se replantea la forma de hacer frente a las limitaciones encontradas, sería muy útil trabajar en la elaboración de un algoritmo de sustracción de fondo que proporcione una máscara de frente lo más limpia (libre de errores) y precisa (evitando camuflajes) posible desde un primer momento y no en etapas posteriores (como se está haciendo con la información de estructura). Además, en lo que respecta al tratamiento del modelo de fondo, muchos problemas vienen por disponer de algoritmos que llevan a cabo una actualización del fondo tanto de objetos de interés (que en caso de ser removidos en el futuro, provocarán una detección fantasma) como de *blobs* falsamente detectados (como pueden ser cambios de iluminación). Por tanto, para condicionar lo menos posible el análisis de regiones estáticas, se debe trabajar en el desarrollo de un algoritmo de sustracción de fondo que sea capaz de actualizarse sin eliminar detecciones estáticas (evitando así detecciones fantasma).

Algunos trabajos recientes, muestran el empleo de puntos característicos para llevar a cabo la detección de regiones de frente, valiéndose de su invarianza a cambios de iluminación y evitando las limitaciones de la sustracción de fondo. Por tanto, su exploración es una alternativa a la sustracción de fondo, que es uno de los mayores problemas actuales.

Bibliografía

- [1] J. Kim, B. Kang, H. Wang, and D. Kim. Abnormal object detection using feedforward model and sequential filters. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 70–75, Beijing (China), Sept. 2012. [1](#), [2](#), [9](#), [28](#), [44](#)
- [2] L. Maddalena and A. Petrosino. Stopped object detection by learning foreground model in videos. *Neural Networks and Learning Systems, IEEE Transactions on*, 24(5):723–735, 2013. [1](#), [5](#), [8](#)
- [3] A. Albiol, Laura Sanchis, A. Albiol, and J.M. Mossi. Detection of parked vehicles using spatiotemporal maps. *Intelligent Transportation Systems, IEEE Transactions on*, 12(4):1277–1291, 2011. [1](#), [5](#), [9](#), [28](#)
- [4] J.T. Lee, M. S. Ryoo, M. Riley, and J.K. Aggarwal. Real-time illegal parking detection in outdoor environments using 1-d transformation. *IEEE Trans. Circuits Syst. Video Technol.*, 19(7):1014–1024, July 2009. [1](#)
- [5] Juan C. SanMiguel and Josiçœ M. Martiçœnez. A semantic-based probabilistic approach for real-time video event recognition. *Computer Vision and Image Understanding*, 116(9):937–952, 2012. [1](#), [3](#)
- [6] J. Ferryman, D. Hogg, J. Sochman, A. Behera, J. Rodriguez-Serrano, S. Worgan, L-Li, V. Leung, M. Evans, P. Cornic, S. Herbin, S. Schlenger, and M. Dose. Robust abandoned object detection integrating wide area visual surveillance and social context. *Pattern Recogn. Lett.*, in press, 2013. [1](#), [5](#), [9](#)
- [7] A. Bayona, J.C. SanMiguel, and J.M. Martiçœnez. Stationary foreground detection using background subtraction and temporal difference in video surveillance. In *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, pages 4657–4660, Sept. 2010. [1](#), [2](#), [5](#), [8](#), [13](#), [14](#), [15](#), [28](#), [31](#), [35](#), [44](#), [53](#), [54](#), [55](#), [56](#), [57](#), [58](#)
- [8] Y. Tian, A. Senior, and M. Lu. Robust and efficient foreground analysis in complex surveillance videos. *Mach. Vision Appl.*, 23(5):967–983, 2012. [1](#), [5](#), [6](#), [9](#), [25](#), [28](#), [71](#)
- [9] Jiyang Pan, Quanfu Fan, and S. Pankanti. Robust abandoned object detection using region-level analysis. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 3597–3600, 2011. [1](#), [2](#), [5](#), [8](#), [23](#), [28](#), [31](#), [37](#), [71](#)
- [10] A. Cavallaro, Steiger O., and Ebrahimi T. Semantic video analysis for adaptive content delivery and automatic description. *IEEE Trans. Circuits Syst. Video Technol.*, 15(10):1200–1209, Oct. 2005. [2](#), [5](#), [30](#)

- [11] Chris Stauffer and W. E L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages –252 Vol. 2, 1999. [2](#), [5](#), [6](#), [25](#)
- [12] S. Guler and J. A. Silverstein. Stationary objects in multiple object tracking. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 248–253, Sept. 2007. [2](#), [5](#), [8](#), [9](#), [10](#), [11](#), [19](#), [28](#), [54](#), [55](#), [56](#), [58](#)
- [13] J.C. San Miguel and J.M. Martínez. Robust unattended and stolen object detection by fusing simple algorithms. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 18–25, Sept. 2008. [2](#), [8](#)
- [14] A. Bayona, J. C. SanMiguel, and J. M. Martínez. Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 25–30, Genova (Italy), Sep. 2009. [3](#), [7](#), [12](#)
- [15] M. Hofmann, P. Tiefenbacher, and G. Rigoll. Background segmentation with feedback: The pixel-based adaptive segmenter. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 38–43, 2012. [5](#), [6](#), [7](#)
- [16] C. Jing-Ying, L. Huei-Hung, and C. Liang-Gee. Localized detection of abandoned luggage. *EURASIP J. Adv. Signal Process.*, Article ID 675784, 2010. [5](#), [8](#), [11](#), [12](#), [28](#), [54](#), [55](#), [56](#), [58](#)
- [17] F. Porikli, Y. Ivanov, and T. Haga. Robust abandoned object detection using dual foregrounds. *EURASIP J. Adv. Signal Process.*, Article ID 197875, 2008. [5](#), [6](#), [8](#), [9](#), [16](#), [17](#), [20](#), [28](#), [53](#), [54](#), [55](#), [56](#), [57](#), [58](#)
- [18] R. Evangelio and T. Sikora. Static object detection based on a dual background model and a finite-state machine. *EURASIP J Image Video Process.*, Article ID 858502, 2011. [5](#), [8](#), [18](#), [20](#), [21](#), [28](#)
- [19] R. Mathew, Zhenghua Yu, and Jian Zhang. Detecting new stable objects in surveillance video. In *Multimedia Signal Processing, 2005 IEEE 7th Workshop on*, pages 1–4, 2005. [5](#), [8](#), [9](#)
- [20] Thi Thi Zin, P. Tin, T. Toriu, and H. Hama. A series of stochastic models for human behavior analysis. In *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pages 3251–3256, 2012. [5](#)
- [21] Q. Fan and S. Pankanti. Modeling of temporarily static objects for robust abandoned object detection in urban surveillance. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 36–41, Sep. 2011. [6](#), [9](#)
- [22] O. Barnich and M. Van Droogenbroeck. Vibe: A universal background subtraction algorithm for video sequences. *Image Processing, IEEE Transactions on*, 20(6):1709–1724, 2011. [6](#)
- [23] Chris Stauffer and W. E L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages –252 Vol. 2, 1999. [8](#)

- [24] M.D. Beynon, D.J. Van Hook, M. Seibert, A. Peacock, and D. Dudgeon. Detecting abandoned packages in a multi-camera video surveillance system. In *Advanced Video and Signal Based Surveillance, 2003. Proceedings. IEEE Conference on*, pages 221–228, 2003. 8
- [25] W. Hassan, P. Birch, B. Mitra, N. Bangalore, R. Young, and C. Chatwin. Illumination invariant stationary object detection. *Computer Vision, IET*, 7(1):1–8, 2013. 8, 28
- [26] Liu Xiya, Wang Jingling, and Zhang Qin. An abandoned object detection system based on dual background and motion analysis. In *Computer Science Service System (CSSS), 2012 International Conference on*, pages 2293–2296, 2012. 9, 28
- [27] Q. Fan and S. Pankanti. Robust foreground and abandonment analysis for large-scale abandoned object detection in complex surveillance videos. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 58–63, Beijing (China), Sept. 2012. 9, 25, 28, 37, 71
- [28] Kurt Skifstad and Ramesh Jain. Illumination independent change detection for real world image sequences. *Computer Vision, Graphics, and Image Processing*, 46(3):387 – 399, 1989. 25
- [29] J. Connell, A.W. Senior, A. Hampapur, Y.-L. Tian, L. Brown, and S. Pankanti. Detection and tracking in the ibm peoplevision system. In *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, volume 2, pages 1403–1406 Vol.2, 2004. 26
- [30] Ying-Li Tian, Rogerio Feris, and Arun Hampapur. Real-Time Detection of Abandoned and Removed Objects in Complex Environments. In *The Eighth International Workshop on Visual Surveillance - VS2008*, Marseille, France, 2008. Graeme Jones and Tieniu Tan and Steve Maybank and Dimitrios Makris. 26
- [31] C. Su and A. Amer. A real-time adaptive thresholding for video change detection. In *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, pages 157–160, 2006. 33
- [32] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, January 1979. 34
- [33] Paul L. Rosin. Unimodal thresholding. *Pattern Recognition*, 34(11):2083–2096, November 2001. 34
- [34] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, 2004. 37, 38
- [35] R. Cucchiara, C. Grana, G. Neri, M. Piccardi, and A. Prati. The Sakbot system for moving object detection and tracking. In *Video-based Surveillance Systems: Computer Vision and Distributed Processing (Part II - Detection and Tracking)*, pages 145–158. Kluwer Academic Publishers, 2001. 69
- [36] L. Caro Campos, J.C. SanMiguel, and J.M. Martinez. Discrimination of abandoned and stolen object based on active contours. In *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, pages 101–106, 2011. 71
- [37] J.C. SanMiguel, L. Caro, and J.M. Martinez. Pixel-based colour contrast for abandoned and stolen object discrimination in video surveillance. *Electronics Letters*, 48(2):86–87, 2012. 71, 74

Apéndice A

Apéndice 1

A.0.1. Otras características descartadas

- Característica de estructura como post-procesado

Se ha explorado incorporar la característica de *SSIM* a modo de post-procesado de la combinación de las características de frente y movimiento. La idea es analizar los *blobs* del resultado y extraer la media de *SSIM* de sus píxeles. Si el valor obtenido es superior a un cierto umbral, entonces el *blob* es removido por considerarse igual al fondo de la imagen. Algunos *blobs* ocasionados por cambios de iluminación son eliminados como se puede ver en la figura A.1. Sin embargo, a primera vista, se tiene un importante problema, no se pueden tratar *blobs* que tienen detecciones correctas e incorrectas (ver Figura A.1) .

En consecuencia se ha pasado a realizar un análisis a nivel *sub-blob*, concretamente por bloques, con lo que se consigue tratar el *blob* internamente (ver Figura A.2). No obstante se tiene otro problema: no hay robustez a oclusiones pues al comparar el objeto que ocluye y el fondo la puntuación desciende y no se elimina el *blob* (ver tercera fila de la Figura A.2), por lo que se decidió acumular una variación temporal de estructura ($RSSIM_t(\mathbf{x})$) para conseguir la robustez buscada que se tiene en el algoritmo propuesto.

- Característica de color

Se exploró emplear la técnica descrita en [35] para tratar sombras y cambios de iluminación. Esta técnica trabaja en el espacio de color HSV, por corresponderse con la percepción humana del color, para llevar a cabo la eliminación de sombras y reflejos basándose en la idea de que una sombra sobre el fondo no provoca una variación significativa de la tonalidad.



Figura A.1: De izquierda a derecha: *frame* bajo análisis, máscara obtenida de la combinación de $FHI_t(\mathbf{x})$ y $MHI_t(\mathbf{x})$ y su post-procesado empleando $SSIM$. En la primera fila se puede ver como se elimina el *blob* causado por cambio de iluminación. Sin embargo, en la segunda y tercera fila se puede apreciar como no se puede hacer frente a falsas detecciones incluidas en un mismo *blob* en el que hay objetos estáticos. Así en la segunda fila no se puede eliminar la sombra de la maleta y en la tercera fila se remueve todo el *blob* sin atender a que existen regiones estáticas.

La idea era extraer una característica de color mediante una imagen con valor 0 para valores sin cambios de iluminación y sin sombras, y valor 1 para el resto de píxeles. Al igual que se explica en A.0.1, emplear esta característica a modo post-procesado lleva a no soportar oclusiones en las regiones de interés (con sombras o cambios de iluminación), por lo que esta configuración se desechó. En su lugar se decidió probar a acumular la variación temporal de la característica para soportar oclusiones. Esta configuración introducía varios problemas:

1. En primer lugar, los cambios de iluminación tan solo se reducen, no se eliminan por completo (ver Figura A.3), por lo que no es una alternativa válida a la característica de estructura.

2. Visto el primer problema, podía plantearse emplear esta característica como un añadido a las características empleadas en el algoritmo propuesto, sin embargo no es posible ya que esta característica detecta sombra en los contornos de los objetos, lo cual lleva a decrementar enormemente la puntuación final de regiones estáticas en continua oclusión (pues las sombras del contorno de las personas llevan a reducir la puntuación). Este problema puede pensarse que es abordable si se lleva a cabo un decremento controlado (similar al realizado en $FHI_t(\mathbf{x})$), no obstante para mantener detecciones en situaciones de oclusiones continuadas se necesitaba reducir enormemente la contribución de esta puntuación al $factorTh$ (hasta entre un 10% y 20% de la aportación máxima), provocando una reducción del umbral sustancial y en consecuencia causando la posibilidad de realizar detecciones sin requerir contribución de todas las puntuaciones (ver Figura A.4).

En resumen, la característica ni permite eliminar sombras y cambios de iluminación con un buen rendimiento, ni sirve como complemento a las características de frente, movimiento y estructura por variar los fundamentos de funcionamiento del umbral reducido al incorporarla al sistema.

■ Robo/abandono

Se exploró el empleo de varias técnicas de robo/abandono basadas en contornos activos [36] y en el contraste del color [37] a modo post-procesado de *blobs* para intentar eliminar detecciones fantasma (estableciendo un robo como una detección fantasma y en consecuencia como un *blob* a remover). Algunos algoritmos del estado del arte como [9][8][27] emplean análisis de bordes (cálculo de energías a lo largo de contornos de *blobs* en *frame* bajo análisis y modelo de fondo) característicos de la discriminación robo/abandono. No obstante, ninguno conseguía resultados satisfactorios.



Figura A.2: De izquierda a derecha: *frame* bajo análisis, máscara obtenida de la combinación de $FHI_t(\mathbf{x})$ y $MHI_t(\mathbf{x})$ y su post-procesado empleando $SSIM$. En la primera y segunda fila se puede ver como los *blobs* ya sí pueden distinguirse distintas partes dentro de un *blob* (a diferencia de los mostrado en la figura A.1), para así eliminar zonas que corresponden a falsos positivos. No obstante la tercera fila muestra que no hay robustez a oclusiones, pues cuando la mujer pasa por delante de la zona de interés no se consigue eliminar el blob debido a que al comparar el frame actual con el fondo no se optiene un valor de $SSIM$ elevado.

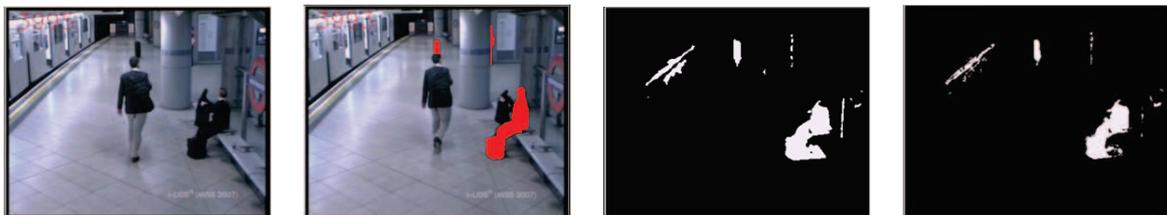


Figura A.3: Ejemplo de post-procesado con la característica de color. De izquierda a derecha: *Frame* bajo análisis, *ground-truth*, máscara estática basada en las características de frente y movimiento y post-procesado de esa máscara. Se puede observar como el post-procesado no elimina las falsas detecciones por sombras o iluminación.

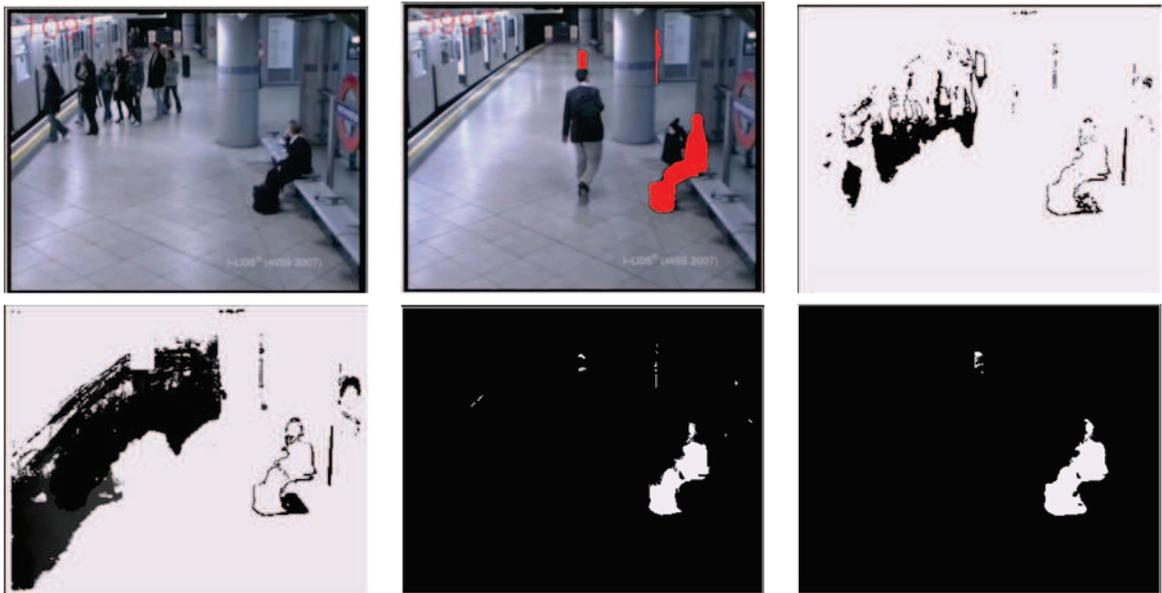


Figura A.4: Ejemplo de característica de color como una característica más a acumular para una secuencia creada artificialmente con movimiento continuado. De izquierda a derecha, primera fila: *frame* bajo análisis, *ground-truth* y característica de color; segunda fila: acumulación de la característica de color, máscara estática obtenida al combinar la característica de color con características de frente y movimiento y por último la máscara obtenida por el algoritmo propuesto. La máscara obtenida con la característica de color no consigue mantener una detección aceptable de la maleta ni disminuyendo la aportación de la puntuación a un 15%, hecho que además provoca que puedan realizarse detecciones basándose solo en 2 de las 3 puntuaciones. Sin embargo con la característica propuesta de estructura no se tiene ese problema como puede verse en la tercera columna de la segunda fila.

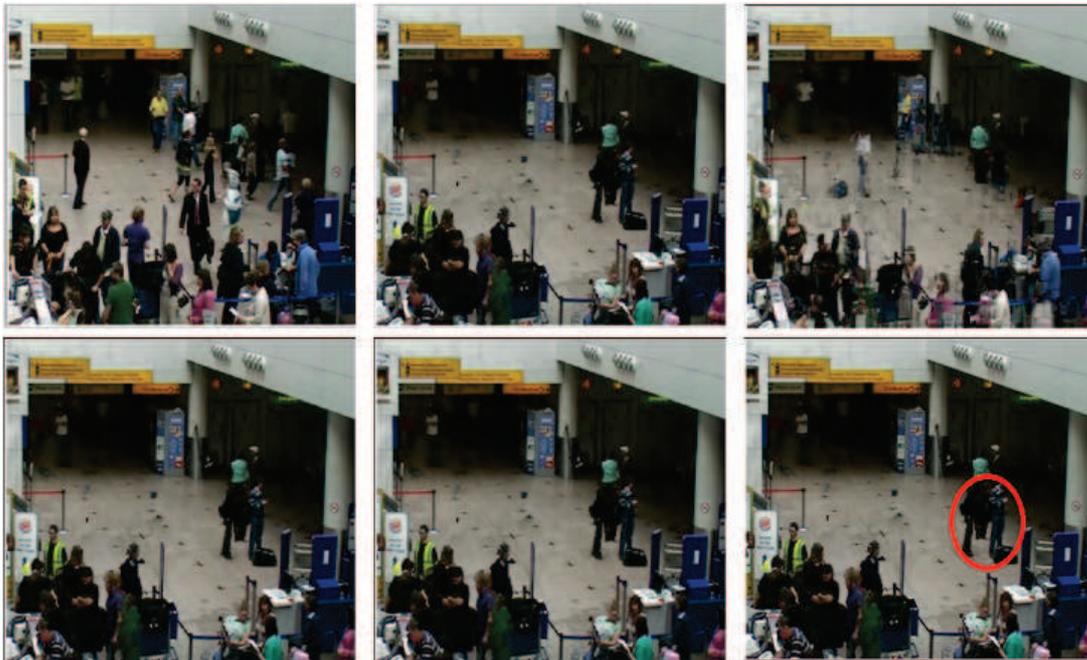


Figura A.5: Algoritmos robo/abandono para eliminar detecciones fantasma (*frame* 2300). Primera fila (de izquierda a derecha): *Frame* y fondos tras emplear algoritmos robo/abandono basados en contraste del color [37]. Segunda fila: fondos tras emplear algoritmos robo/abandono basados en contornos activos y fondo utilizado para la evaluación. En rojo aparecen marcadas unas personas que provocan detecciones fantasma en la máscara estática y que no son removidas con la detección robo de los algoritmos. Solo en la tercera columna de la primera fila se consiguen remover, pero a cambio se han incorporado al fondo numerosos errores.

Apéndice B

Publicaciones

Parte de este trabajo ha sido incluido en la siguiente publicación:

- Diego Ortego, Juan C. SanMiguel: "Stationary foreground detection for video-surveillance based on foreground and motion history images", en 10th IEEE International Conference on Advanced Video and Signal based Surveillance, Proceedings, Krakow (Poland), 2013.

Stationary foreground detection for video-surveillance based on foreground and motion history images

Diego Ortego, Juan C. SanMiguel
Video Processing and Understanding Lab

Escuela Politécnica Superior, Universidad Autónoma de Madrid, SPAIN

Email: Diego.Ortego@estudiante.uam.es, Juancarlos.Sanmiguel@uam.es

Abstract

Stationary foreground detection is a common stage in many video-surveillance applications. In this paper, we propose an approach for stationary foreground detection in video based on the spatio-temporal variation of foreground and motion data. Foreground data are obtained by Background Subtraction to detect regions of interest. Motion data allows to filter out the moving regions and it is estimated using median filters over sliding windows. Spatio-temporal patterns of both data are computed through history images and the final detection is obtained using a two-threshold scheme that considers motion activity. Partial visibility of stationary foreground for short-time intervals is handled to increase robustness. The results over challenging video-surveillance sequences show an improvement of the proposed approach against the related work.

1. Introduction

Detecting stationary foreground regions in video has recently become an active area of research in many video-surveillance areas such as the detection of abandoned objects [1] and illegally parked vehicles [2]. This task remains unsolved for complex sequences such as crowded scenarios as it faces many challenges related with illumination changes, low resolution images, object occlusions, high density of moving objects (increasing the number of cast shadows) and initialization of the detection algorithms.

Common stationary foreground detectors are based on the background subtraction approach [3], which provides binary foreground maps. Some proposals focus on tracking foreground regions to detect the stationary ones [4][5]. They are limited as current tracking performance is only acceptable in situations with few moving objects [6]. Avoiding tracking, many pixel-wise approaches are proposed based on dual-backgrounds [6][7][8], accumulators [9][10], sub-sampling [11], specific object classifiers [12] or proper-

ties of background models [13][14]. However, background subtraction presents many false positives in crowds that decrease stationary detection performance. Recently, combinations between foreground and motion analysis have been investigated to address these limitations in crowds [15].

In this paper, we propose an approach for detecting stationary foreground regions in video that combines foreground and motion data. Building on the concept of History Images [16], we develop energy maps (images) that account for spatio-temporal patterns of foreground and motion. Foreground data are extracted using a standard background subtraction approach [17]. Motion data is obtained by computing frame differences in the nearby frames (before and after the analysis instant) using a median filter. Finally, both energy maps are combined through a two-threshold technique that considers the spatial location of motion activity. Occlusion handling is included at pixel level to tolerate partial visibility of stationary regions. The proposed approach is evaluated and compared on video-surveillance datasets in presence of detection challenges such as occlusions, illumination changes and clutter.

The structure of this paper is as follows. Section 2 discusses the related work. Section 3 describes the proposed approach. Experimental results are presented in Section 4. Finally, Section 5 summarizes the main conclusions.

2. Related work

Many approaches have been proposed for stationary region detection in video [3]. They can be classified into based on tracking [4][5] or background subtraction [10]. As tracking accuracy is significantly degraded in complex sequences, such as crowded videos, this section focuses on the second category that does not use tracking and can be applied to a wide variety of video-surveillance scenarios.

Two major error sources affect the performance of detection approaches based on background subtraction. The first corresponds to photometric factors (illumination changes, camouflages, shadows and reflections) whereas the second

derives from sequences with high density of moving objects (multiple occlusions and algorithm initialization). Adaptive background subtraction (ABS) has been proposed to handle photometric errors by continuously updating the background model [13]. Combinations of fast and slow adaptation rates can be used for stationary detection [6]. However, such adaptation might decrease detection performance as static objects can be incorporated into the background before they become static [12]. Thus, slow rates are preferred that reduce the robustness to photometric errors. Moreover, background initialization is complex in crowded sequences that, if incorrect, may lead to many false positives (of foreground), which decrease stationary detection performance.

In this context, several approaches have been developed based on temporal accumulation [9][10] and sub-sampling [11] of foreground masks. Moreover, modeling properties can be used such as the transitions of Gaussian Mixture Modeling (GMM) approach [13]. They can be extended by defining the states of foreground pixels through finite-state-machines such as for GMMs [14] and dual-backgrounds [8]. Recent results show that sampling approaches present best results with high spatial accuracy (less noise in the final mask) and low temporal accuracy (detection delay) [14]. However, selection of the sampling instants remains unsolved, which is critical for efficient analysis. All previous approaches are limited for crowded scenes as many false stationary detections are produced due to the high amount of detected foreground. Specific object classifiers can be used for solving this limitation in crowds [12]. However, it requires to know the objects of interest, which is often not available. Foreground sampling can be combined with inter-frame motion [15], demonstrating that motion could be used to remove false detections in crowds. However, it shares the drawbacks of sampling and the spatial accuracy dependency with camouflage errors.

In summary, no approach is able to perfectly perform in crowded scenes considering low false positive detection and spatio-temporal robustness of static mask. We propose an approach that combines the most relevant features of existing approaches avoiding the sub-sampling drawbacks.

3. Stationary region detection

In this section, we describe the proposed approach for stationary foreground detection. It comprises two analysis, both at pixel level on frame-by-frame basis, for foreground and motion data (see Figure 1). Each analysis has two stages to model spatio-temporal patterns: feature extraction and history image computation. The two resulting history images are combined to get an image representing the foreground-motion variation over time, which is thresholded to get the stationary foreground mask. Finally, occlusion handling is performed to recover lost pixels due to frequent object occlusions in crowds.

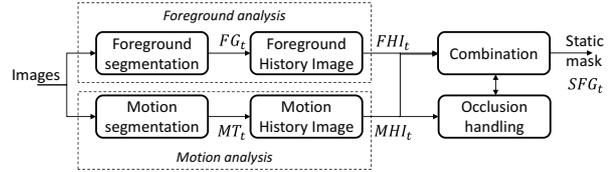


Figure 1. Overview of the proposed approach.

3.1. Foreground analysis

First, background subtraction is applied to detect foreground. We have used the method proposed in [17] due to its low computational cost and robustness to noise. Detection considers pixel neighborhood and it is summarized as:

$$FG_t(\mathbf{x}) \iff \sum_{\mathbf{d} \in \mathcal{N}(\mathbf{x})} (I_t(\mathbf{d}) - B_t(\mathbf{d}))^2 > \beta, \quad (1)$$

where \mathbf{x} and \mathbf{d} are pixel locations $\{x, y\}$; $\mathcal{N}(\mathbf{x})$ is an $N \times N$ patch centered at \mathbf{x} ; I_t and B_t are current and background frames and β is a decision threshold. $FG_t(\mathbf{x}) = 1(0)$ indicates foreground (background) for the pixel located at \mathbf{x} .

Then, we measure the foreground temporal variation to get a Foreground History Image $FHI_t(\mathbf{x})$, considering foreground and background detections as follows:

$$FHI_t(\mathbf{x}) = FHI_{t-1}(\mathbf{x}) + w_{pos}^f \cdot FG_t(\mathbf{x}), \quad (2)$$

$$FHI_t(\mathbf{x}) = FHI_{t-1}(\mathbf{x}) - w_{neg}^f \cdot (\sim FG_t(\mathbf{x})), \quad (3)$$

where \sim is the logical NOT operation; w_{pos}^f and w_{neg}^f are two weights to manage the contribution of the foreground ($FG_t(\mathbf{x}) = 1$) and background ($\sim FG_t(\mathbf{x}) = 1$) detections. For giving a temporal sense to stationary detection, we should increase FHI_t values one-by-one ($w_{pos}^f = 1$) when they belong to foreground and reset FHI_t values to 0 when they are background. Nevertheless, temporally sparse errors in foreground detection may cause losing correct stationary detections if reset to 0. This frequently happens in crowds where a static region is occluded by fast moving objects which cause camouflage errors. Hence, penalization weight w_{neg}^f should decrease FHI_t at a higher rate than positive one w_{pos}^f without resetting to 0 for increasing robustness against foreground errors (e.g., $w_{neg}^f = 15$).

Finally, the result of this analysis is $FHI_t(\mathbf{x})$, a foreground score that increases when the pixel is foreground and decreases when it belongs to the background model.

3.2. Motion analysis

Recent works show the use of motion information for filtering false positives caused by high densities of moving objects, thus helping to detect stationary regions [5][15]. Current use focuses on thresholding inter-frame differences and

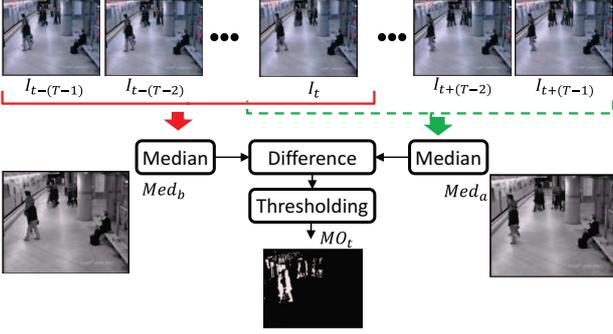


Figure 2. Motion extraction scheme using median filtering over temporal windows before and after the frame under analysis.

then, applying the sub-sampling approach over the temporal sequence of differences [15]. However, stationary regions are frequently occluded in presence of many moving objects and, therefore, the no-motion state (or static) is difficult to observe for such regions during all frames of a determined time interval. Hence, successful performance requires selecting the correct number and frequency of the samples taken, which can not be guaranteed for all situations.

We propose to solve these limitations by extending the motion analysis over temporal windows of length T (see Figure 2). Although multiple occlusions affect stationary regions in crowds, they usually last for few frames and the most predominant region in short-time intervals corresponds to the stationary one. Moreover, History Images [16] could be employed instead of sampling approach to avoid deciding when samples have to be taken.

For extracting motion using temporal windows, we apply a median filter before and after the frame under analysis:

$$Med_b = Median\{I_{t-T+1}, \dots, I_t\} \quad (4)$$

$$Med_a = Median\{I_t, \dots, I_{t+T-1}\} \quad (5)$$

where Med_a and Med_b are the median images of temporal windows of length T taken after and before I_t (all images at gray level). A delay is introduced for each instant t to get the next $T - 1$ frames. The choice of T depends on the speed of objects and duration of occlusions, requiring high values for slow occlusions. Empirical testing over real sequences obtained good performance for T values ranging from 10 to 20. Then, final motion image is obtained as:

$$MO_t(\mathbf{x}) = \begin{cases} 1 & \text{if } |Med_b - Med_a| < \tau \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where $MO_t(\mathbf{x}) = 1$ is for absence of motion and τ is a threshold to set the no-motion case. We automatically get τ by applying the Kapur method [18] on $|Med_b - Med_a|$.

Finally, temporal variation of the no-motion mask $MO_t(\mathbf{x})$ is computed via the Motion History Image $MHI_t(\mathbf{x})$, which is similar to the foreground case:

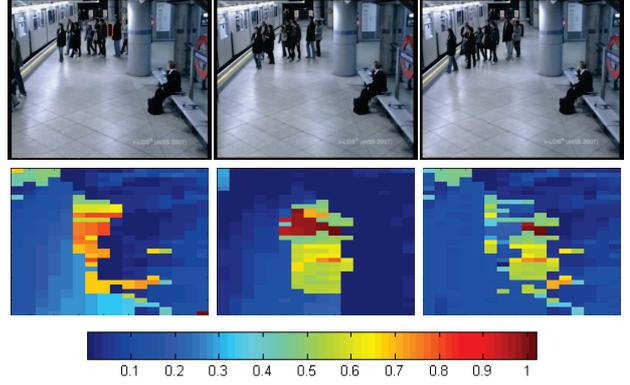


Figure 3. Example of $MHI_t(\mathbf{x})$ using the median-based proposed approach (PRO) and the standard frame difference (FD) [15]. First row: frames 875, 917 and 940 of sequence AVSS07 Med. Second row: $MHI_{917}(\mathbf{x})$ results of frame 917 using FD (left), PRO (center) and their absolute difference (right) for the red-dashed rectangle in frame 875 (suitcase). PRO estimates better the no-motion state of the stationary region with higher values in $MHI_{917}(\mathbf{x})$.

$$MHI_t(\mathbf{x}) = MHI_{t-1}(\mathbf{x}) + w_{pos}^m \cdot MO_t(\mathbf{x}), \quad (7)$$

$$MHI_t(\mathbf{x}) = MHI_{t-1}(\mathbf{x}) - w_{neg}^m \cdot (\sim MO_t(\mathbf{x})), \quad (8)$$

where w_{pos}^m and w_{neg}^m are two weights for controlling the contribution of the no-motion ($MO_t(\mathbf{x}) = 1$) and motion ($\sim MO_t(\mathbf{x}) = 1$) cases. Similarly to stationary detection using $FHI_t(\mathbf{x})$, we should increase $MHI_t(\mathbf{x})$ values one-by-one ($w_{pos}^m = 1$) when they belong to the no-motion state and reset $MHI_t(\mathbf{x})$ values to 0 when they belong to the motion state. We use this scheme as $MHI_t(\mathbf{x})$ is included to compensate high values of $FHI_t(\mathbf{x})$ caused by continuous motion of moving objects, only keeping $FHI_t(\mathbf{x})$ values of non-moving pixels. Hence, we set $w_{neg}^m = MHI_{t-1}(\mathbf{x})$ to reset $MHI_t(\mathbf{x})$ to 0 when motion is detected. Figure 3 depicts an example where the no-motion of the entire stationary region is only detected with the proposed $MHI_t(\mathbf{x})$.

Finally, the result of the motion analysis is $MHI_t(\mathbf{x})$, a no-motion score that increases when the pixel does not suffer motion or decreases when it suffers motion.

3.3. Combination

After obtaining $FHI_t(\mathbf{x})$ and $MHI_t(\mathbf{x})$, we normalize them to the range $[0, 1]$ considering the video framerate (fps) and the stationary detection time (t_{static}):

$$\overline{FHI}_t(\mathbf{x}) = \min\{1, FHI_t(\mathbf{x})/(fps \cdot t_{static})\}, \quad (9)$$

$$\overline{MHI}_t(\mathbf{x}) = \min\{1, MHI_t(\mathbf{x})/(fps \cdot t_{static})\}. \quad (10)$$

Then, we compute the mean of both normalized images to get a stationary history image $SHI_t(\mathbf{x})$ representing foreground-motion variation over time. Finally, stationary detection mask is obtained by thresholding as:

Criteria	Non-crowded					Crowded													Total
	AVSS07		PETS06			PETS07		PETS06						HALL					
	Easy	S7_C3	S4_C3	S4_C4	S5_C3	Med	Hard	S5_C1	S5_C2	S7_C1	S7_C4	S1_C1	S1_C4	S4_C1	S4_C2	H_S1	H_S2	H_S3	
Background Initialization	L	L	L	L	L	L	L	H	H	H	M	H	M	H	H	H	M	H	-
Illumination changes	L	-	-	-	M	L	L	M	M	-	-	-	-	-	-	L	L	L	-
Motion level	L	L	L	L	L	M	H	H	H	H	M	H	L	H	H	H	M	H	-
Overall complexity	L	L	L	L	L	H	H	H	H	H	M	H	M	H	H	H	M	H	-
Number of frames	4291	3401	3051	3051	2900	4834	5311	2900	2900	3401	3401	3021	3021	3051	3051	10000	10834	15102	87521
Annotated stationary regions	2	1	3	4	2	14	13	3	3	1	1	2	2	6	3	3	1	11	75

Table 1. Description of the sequences of the evaluation set. (Key. L:Low. M:Medium. H:High).

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{if } SHI_t(\mathbf{x}) \geq \eta \\ 0 & \text{otherwise} \end{cases}, \quad (11)$$

where $\eta \in (0, 1]$ is the threshold for stationary detection. Its value should be high ($\eta = 1$, if no foreground or motion errors). $FHI_t(\mathbf{x})$ and $MHI_t(\mathbf{x})$ must indicate the stationary regions to detect as they equally contribute to SHI_t being not possible such detection relying only on one of them.

However, stationary regions constantly occluded remain undetected in most of the situations as $\overline{MHI}_t(\mathbf{x})$ is not able to capture the required consecutive no-motion to allow the increase of its values. We include an additional condition that reduces η in pixels where $\overline{FHI}_t(\mathbf{x})$ has reached a high value with previous or current motion:

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{if } \overline{FHI}_t(\mathbf{x}) \geq \eta \& \overline{MHI}_t(\mathbf{x}) < \eta \& \\ & SHI_t(\mathbf{x}) \geq \eta \cdot factorTh \\ 0 & \text{otherwise} \end{cases}, \quad (12)$$

where $factorTh \in (0, 1)$ weights the threshold η . It should have high (low) values for sequences presenting low (high) motion activity. Eq. 12 allows to apply a lower threshold to pixels with previous or current motion ($\overline{MHI}_t(\mathbf{x}) < \eta$) and high foreground history image values ($\overline{FHI}_t(\mathbf{x}) \geq \eta$), obtaining detections in situations with motion. In summary, a two-thresholding scheme is proposed that applies conditions to pixels with no-motion (Eq. 11) and motion in previous time instants (Eq. 12).

3.4. Occlusion handling

After detecting stationary regions, a reduction of $\overline{MHI}_t(\mathbf{x})$ values might occur due to total or partial occlusions and, therefore, reducing the values of $SHI_t(\mathbf{x})$ to satisfy any of the conditions in Eqs. 11 and 12. We add an occlusion handling method to recover initial detections where they are lost. Unlike previous works [5][15], we focus on pixels instead at blob level for such handling as it more robust to foreground errors. For each pixel we check some conditions and propagate previous detections as follows:

$$SFG_t(\mathbf{x}) = \begin{cases} 1 & \text{if } SFG_{t-1}(\mathbf{x}) = 1 \& \overline{MHI}_t(\mathbf{x}) < \eta \& \\ & \overline{FHI}_t(\mathbf{x}) \geq \eta \cdot factorOc \\ 0 & \text{otherwise} \end{cases}, \quad (13)$$

where $factorOc \in (0, 1)$ is the tolerance to temporally sparse foreground errors (e.g., camouflages) that reduce $FHI_t(\mathbf{x})$ and cause loss of static detections. This recover is applied when pixels have experienced motion ($\overline{MHI}_t(\mathbf{x}) < \eta$) and high accumulated foreground ($\overline{FHI}_t(\mathbf{x}) \geq \eta \cdot factorOc$). The lower values of $factorOc$, the higher robustness against errors. However too low values delay the disappearance of static detections which no longer exist.

4. Experimental results

In this section, we present and compare the experimental results of the proposed approach.

4.1. Setup

Experiments are performed on selected sequences from AVSS2007¹, PETS2006² and PETS2007³ datasets. They provide a diverse set of public video-surveillance scenarios (see Table 1). We also use a larger dataset for crowded situations recorded at a faculty hall (HALL). We manually annotated all stationary regions as ground truth.

To evaluate detection performance, we use standard Precision (P), Recall (R) and F-score (F) measures:

$$P = TP / (TP + FP), \quad (14)$$

$$R = TP / (TP + FN), \quad (15)$$

$$F = 2 \cdot P \cdot R / (P + R), \quad (16)$$

where TP, FP and FN are, respectively, correct, false and missed detections (as compared to ground-truth ones).

To set up the proposed approach, we use the common values for framerate (25 fps) and stationary detection time ($t_{static} = 20$ secs). We use the following values to guarantee that no static region appears before t_{static} : $w_{pos}^f = 1$, $w_{neg}^m = \overline{MHI}_{t-1}(\mathbf{x})$ and $\eta = 1$. Temporally sparse foreground errors (i.e., camouflages) are tolerated by empirically setting $w_{neg}^f = 15$ and $factorOc = 0.8$. Finally, after testing on crowded videos, we observed a decrease around 50-25% of $\overline{MHI}_t(\mathbf{x})$ values with $t_{static} = 10-20$ secs, thus, we set $factorTh = 0.625$ to

¹<http://www.avss2007.org/>

²<http://www.cvg.rdg.ac.uk/PETS2006/>

³<http://www.cvg.rdg.ac.uk/PETS2007/>

Approach	Non-crowded							Crowded														
	AVSS07			PETS06			PETS07	Mean	AVSS07			PETS07			PETS06				HALL			Mean
	Easy	S7_C3	S4_C3	S4_C4	S5_C3			Med	Hard	S5_C1	S5_C2	S7_C1	S7_C4	S1_C1	S1_C4	S4_C1	S4_C2	H_S1	H_S2	H_S3		
[9]	P	.33	1	1	.80	.50	.72	.58	.48	.16	.20	.05	.12	.12	.33	.27	.05	.50	1	.34	0.32	
	R	1	1	1	1	1	1	1	1	1	1	1	1	1	.83	.33	1	1	1	1	0.93	
	F	.50	1	1	.88	.66	.81	.73	.65	.28	.33	.10	.22	.22	.50	.41	.10	.67	1	.51	.44	
[11]	P	.33	1	.75	.80	.50	.67	.51	.52	.17	.21	.05	.12	.12	.33	.27	.07	.30	.14	.37	0.24	
	R	1	1	1	1	1	1	1	1	1	1	1	1	1	.83	.33	1	1	1	1	0.93	
	F	.50	1	.85	.88	.66	.78	.68	.68	.29	.35	.11	.22	.22	.50	.41	.10	.46	.25	.54	.37	
[15]	P	.40	1	1	.80	.50	.74	.60	.58	0	.16	.14	.16	.10	.40	.38	.09	.60	.50	.55	0.33	
	R	1	1	1	1	1	1	.85	.76	0	.66	1	1	.50	1	.83	.33	1	1	.91	0.53	
	F	.57	1	1	.88	.66	.82	.70	.66	0	.26	.25	.28	.16	.57	.52	.14	.75	.67	.69	.43	
Proposed	P	.33	1	1	.80	.50	.72	.66	.68	.17	.23	.1	.16	.13	.40	.41	.07	.60	1	.58	0.39	
	R	1	1	1	1	1	1	1	1	1	1	1	1	1	.83	.33	1	1	1	1	0.93	
	F	.50	1	1	.88	.66	.81	.80	.81	.29	.37	.18	.28	.23	.57	.55	.10	.75	1	.73	.51	

Table 2. Comparative results of the proposed approach using Precision (P), Recall (R) and F-score (F). Bold indicates best results.

consider the contributions of $FHI_t(\mathbf{x})$ and $MHI_t(\mathbf{x})$ (respectively, 100% and 25%) to $SHI_t(\mathbf{x})$ in crowds. The same parameters are used for all the experiments.

4.2. Results

Table 2 compares the proposed approach with the most popular ones based on foreground accumulation [9], subsampling [11] and foreground-motion sampling [15]. In non crowded sequences, results are very similar getting all high performance. The best results are obtained by [15] because there are not many occlusions, so it is able to avoid false detections through motion analysis without losing correct detections. For crowded sequences, many occlusions and high motion take place. Previous works [9][11] are not good enough in these situations, getting in general very high Recall values but low Precision ones (high false positive rate). [15] is capable to improve Precision in most of the sequences, because it eliminates many false detections. However, this filtering removes stationary detections in many cases, so Recall is also decreased counteracting the previous improvement. The proposed approach is able to maintain the stationary region detection rate (Recall) and still removing the false detections caused by high motion. Globally, the proposed approach has an improvement around 18% and 16% for, respectively, Precision and F-score as compared to the best results of the selected approaches.

Figure 4 shows some visual examples of the compared approaches. Examples of rows 1, 2, 4 and 5 show the performance improvement of the proposed approach removing false detections caused by high motion. Furthermore, unlike [15], examples 1, 2, 3 demonstrate how the proposed method is able to keep detections in the stationary mask, although a motion analysis is included for dealing with high density situations in [15]. All examples exhibit false detections caused by non-correct background models due to the high complexity for their initialization and other photometric factors (shadows and illuminations).

5. Conclusions

This paper has presented an approach for stationary foreground region detection. It computes spatio-temporal variations of foreground and motion data extracted from the video sequence. A two-threshold scheme is applied to combine the previous analysis and detect stationary regions. The results over heterogeneous datasets show that the proposed approach is effectively applied to crowded sequences outperforming related work and demonstrating the use of motion to remove false positive detections.

As future work, we will explore the use of complex models for foreground detection and background initialization, automatic tuning of algorithm parameters and the use of region-level information.

Acknowledgments

This work has been partially supported by the Spanish Government (TEC2011-25995 EventVideo).

References

- [1] J. Ferryman, D. Hogg, J. Sochman, A. Behera, J. Rodriguez-Serrano, S. Worgan, L-Li, V. Leung, M. Evans, P. Cornic, S. Herbin, S. Schlenger, and M. Dose. Robust abandoned object detection integrating wide area visual surveillance and social context. *Pattern Recogn. Lett.*, in press, 2013. 1
- [2] J.T. Lee, M. S. Ryoo, M. Riley, and J.K. Aggarwal. Real-time illegal parking detection in outdoor environments using 1-d transformation. *IEEE Trans. Circuits Syst. Video Technol.*, 19(7):1014–1024, July 2009. 1
- [3] A. Bayona, J. C. SanMiguel, and J. M. Martínez. Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 25–30, Genova (Italy), Sep. 2009. 1, 2
- [4] J.C. San Miguel and J.M. Martínez. Robust unattended and stolen object detection by fusing simple algorithms. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 18–25, Sept. 2008. 1, 2



Figure 4. Sample results for stationary foreground detection for (from top to bottom row) *Hard*, *S5_C1*, *S5_C1*, *S4_C1* and *H_S2* sequences.

- [5] J. Kim, B. Kang, H. Wang, and D. Kim. Abnormal object detection using feedforward model and sequential filters. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 70–75, Beijing (China), Sept. 2012. 1, 2, 3.2, 3.4
- [6] F. Porikli, Y. Ivanov, and T. Haga. Robust abandoned object detection using dual foregrounds. *EURASIP J. Adv. Signal Process.*, Article ID 197875, 2008. 1, 2
- [7] A. Singh, S. Sawan, M. Hanmandlu, V. K. Madasu, and B. C. Lovell. An abandoned object detection system based on dual background segmentation. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 352–357, Sep. 2009. 1
- [8] R. Evangelio and T. Sikora. Static object detection based on a dual background model and a finite-state machine. *EURASIP J Image Video Process.*, Article ID 858502, 2011. 1, 2
- [9] S. Guler and J. A. Silverstein. Stationary objects in multiple object tracking. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 248–253, Sept. 2007. 1, 2, 4, 4.2
- [10] L. Maddalena and A. Petrosino. Stopped object detection by learning foreground model in videos. *IEEE Trans. Neural Netw. Learn. Sys.*, (in press), 2013. 1, 2
- [11] C. Jing-Ying, L. Huei-Hung, and C. Liang-Gee. Localized detection of abandoned luggage. *EURASIP J. Adv. Signal Process.*, Article ID 675784, 2010. 1, 2, 4, 4.2
- [12] M. Bhargava, C. Chen, M.S. Ryoo, and J.K. Aggarwal. Detection of object abandonment using temporal logic. *Mach. Vision Appl.*, 20:271–281, 2009. 1, 2
- [13] Y. Tian, A. Senior, and M. Lu. Robust and efficient foreground analysis in complex surveillance videos. *Mach. Vision Appl.*, 23(5):967–983, 2012. 1, 2
- [14] Q. Fan and S. Pankanti. Modeling of temporarily static objects for robust abandoned object detection in urban surveillance. In *Proc. IEEE Conf. Adv. Video Signal Based Surveill. (AVSS)*, pages 36–41, Sep. 2011. 1, 2
- [15] A. Bayona, J.C. SanMiguel, and J.M. Martínez. Stationary foreground detection using background subtraction and temporal difference in video surveillance. In *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, pages 4657–4660, Sept. 2010. 1, 2, 3.2, 3, 3.4, 4, 4.2
- [16] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3):257–267, 2001. 1, 3.2
- [17] A. Cavallaro, Steiger O., and Ebrahimi T. Semantic video analysis for adaptive content delivery and automatic description. *IEEE Trans. Circuits Syst. Video Technol.*, 15(10):1200–1209, Oct. 2005. 1, 3.1
- [18] C. Su and A. Amer. A real-time adaptive thresholding for video change detection. In *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, pages 157–160, 2006. 3.2

Apéndice C

Presupuesto

1. Ejecución Material

- Compra de ordenador personal (Software incluido) 2.000 €
- Alquiler de impresora láser durante 6 meses 260 €
- Material de oficina 150 €
- Total de ejecución material 2.400 €

2. Gastos generales

- 16 % sobre Ejecución Material 352 €

3. Beneficio Industrial

- 6 % sobre Ejecución Material 132 €

4. Honorarios Proyecto

- 1800 horas a 15 €/ hora 27.000 €

5. Material fungible

- Gastos de impresión 280 €
- Encuadernación 200 €

6. Subtotal del presupuesto

- Subtotal Presupuesto 32.774 €

7. I.V.A. aplicable

- 21 % Subtotal Presupuesto 6.882,5 €

8. Total presupuesto

- Total Presupuesto 39.656,5 €

Madrid, FECHA

El Ingeniero Jefe de Proyecto

Fdo.: Diego Ortego Hernández

Ingeniero Superior de Telecomunicación

Apéndice D

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un sistema basado en detectar regiones estáticas en secuencias de vídeo-seguridad. 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 partidaalzada 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.