

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**DETECCIÓN DE OBJETOS
ABANDONADOS/ROBADOS EN
SECUENCIAS DE VÍDEO-SEGURIDAD**

Álvaro Bayona Gómez

Septiembre 2009

Detección de objetos abandonados/robados en secuencias de vídeo-seguridad

AUTOR: Álvaro Bayona Gómez
TUTOR: Juan Carlos San Miguel Avedillo
PONENTE: José María Martínez Sánchez



Vídeo Processing and Understanding Lab
Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Septiembre de 2009

Palabras clave

Segmentación, Background (BG), Foreground (FG), vídeo-seguridad, robo-abandono, regiones estáticas, secuencia de vídeo, *blob*, detector, segmentador Gamma, precision, recall, mezcla de Gaussianas (MoG), frame, imagen, submuestreo

Resumen

El principal objetivo de este PFC es conseguir detectar objetos abandonados o robados a partir de secuencias de vídeo-seguridad. Este objetivo involucra la detección de regiones estáticas en la imagen que no pertenezcan al fondo, la clasificación de esas regiones entre persona y objeto, y hacer una discriminación final entre robo y abandono.

Con este fin, se han estudiado, evaluado e implementado los algoritmos más relevantes existentes en la literatura, así como nuevas técnicas, para su posterior análisis de prestaciones en secuencias de vídeo-seguridad con alta densidad de personas.

Abstract

The main objective of this Master Thesis (PFC) is to detect abandoned or stolen objects in vídeo surveillance sequences. It involves the detection of stationary foreground objects, their classification into human or objects, and their final discrimination into unattended or stolen objects.

For this purpose, we have studied, implemented and evaluated the most relevant techniques from the state of art, and propose some new techniques. Then, a comparative evaluation is performed for the vídeo-surveillance sequences with high people density.

Agradecimientos

En primer lugar, quiero agradecer a mi tutor, Juan Carlos San Miguel, el apoyo prestado durante todos estos meses para la realización de dicho proyecto, así como la cantidad de cosas que me ha enseñado relacionadas con el mundo de la investigación.

Quiero dar las gracias a José María Martínez y a Jesús Bescós por la ayuda prestada durante todos estos años que he tardado en acabar la carrera, por darme la oportunidad de entrar en el VPU-Lab para realizar este proyecto final de carrera, y por haberme dado la oportunidad de formarme como profesional mediante una publicación internacional y su posterior defensa.

También quiero agradecer a mis padres todo el apoyo y el cariño prestado estos años, en los buenos y en los malos ratos, así como la educación recibida de su parte.

Un recuerdo también para mis abuelas, las cuales seguro que se encuentran orgullosas de tener un nieto ingeniero.

Agradecer a todos los profesores de la universidad, del colegio, amigos, primos y tíos también su cariño mostrado.

Sin todos ellos, esto nunca habría sido posible.

Gracias a todos.

Álvaro Bayona Gómez

INDICE DE CONTENIDOS

1 INTRODUCCIÓN.....	1
1.1 MOTIVACIÓN	1
1.2 OBJETIVOS.....	3
1.3 ORGANIZACIÓN DE LA MEMORIA	4
2 ESTADO DEL ARTE	7
2.1 DETECCIÓN DE REGIONES ESTÁTICAS	7
2.1.1 <i>Introducción</i>	7
2.1.2 <i>Características de la etapa de segmentación fondo-frente</i>	9
2.1.3 <i>Clasificación de métodos basados en segmentación fondo-frente para detectar regiones estáticas</i>	11
2.2 CLASIFICACIÓN PERSONA-OBJETO	13
2.3 DISCRIMINACIÓN ABANDONO-ROBO	14
2.4 DATASETS UTILIZADOS	16
3 SISTEMA BASE EN DETECCIÓN DE OBJETOS ABANDONADOS/ROBADOS.....	19
3.1 ESTRUCTURA GENERAL	19
3.2 SEGMENTACIÓN DE FONDO/FIGURA	20
3.3 SEGUIMIENTO DE BLOBS	26
3.4 DETECCIÓN DE OBJETOS ESTÁTICOS	27
3.5 DETECCIÓN DE OBJETOS ABANDONADOS/ROBADOS	29
4 DETECCIÓN DE REGIONES ESTÁTICAS	33
4.1 MÉTODOS DE DETECCIÓN DE REGIONES ESTÁTICAS SELECCIONADOS.....	33
4.1.1 <i>Método basado en una aproximación clásica</i>	33
4.1.2 <i>Método basado en la acumulación de máscaras de foreground</i>	33
4.1.3 <i>Método basado en las propiedades del modelo de mezcla de Gaussianas</i>	35
4.1.4 <i>Método basado en el submuestreo de máscaras binarias.</i>	38
4.1.5 <i>Método basado en diferentes modelos de fondo actualizados a diferentes velocidades</i>	42
4.2 ALGORITMO DE DETECCIÓN DE REGIONES ESTÁTICAS PROPUESTO	44
4.2.1 <i>Descripción</i>	44
4.2.2 <i>Análisis del algoritmo</i>	47
4.2.3 <i>Solución al problema de las oclusiones</i>	49
5 CLASIFICACIÓN ROBO/ABANDONO	55
5.1 INTRODUCCIÓN	55
5.2 SISTEMA BASE	56
5.2.1 <i>Detector basado en color</i>	57
5.2.2 <i>Detectores basados en el gradiente alto y gradiente bajo</i>	59
5.2.3 <i>Combinación de detectores</i>	62
5.3 DETECTORES PROPUESTOS	64
5.3.1 <i>Detector basado en gradiente global</i>	64
5.3.2 <i>Detectores basados en la forma del contorno</i>	64
5.3.2.1 <i>Detector basado en los momentos de Hu</i>	64
5.3.2.2 <i>Detector basado en la comparación de histogramas de contorno</i>	66
5.3.2.3 <i>Detector basado en comparación bi-direccional de contorno</i>	66
5.4 ESQUEMAS DE FUSIÓN.....	66
5.4.1 <i>Esquema de fusión basado en la regla de la suma</i>	67
5.4.2 <i>Esquema de fusión basado en la regla del producto</i>	67
5.4.3 <i>Esquema de fusión basado en la regla del máximo</i>	68
5.4.4 <i>Esquema de fusión basado en la regla del mínimo</i>	68

5.4.5 Esquema de fusión basado en la media geométrica.....	69
5.4.6 Esquema de fusión basado en la estimación Bayesiana.....	69
5.4.7 Método de Dempster-Shafer.....	71
6 RESULTADOS EXPERIMENTALES	75
6.1 RESULTADOS DE LOS MÉTODOS DE DETECCIÓN DE REGIONES ESTÁTICAS IMPLEMENTADOS	75
6.1.1 Datasets utilizados.....	76
6.1.2 Medidas objetivas.....	78
6.1.3 Medidas subjetivas	79
6.1.4 Comparativa.....	80
6.2 RESULTADOS DE LOS DETECTORES DE ROBO-ABANDONO IMPLEMENTADOS	84
6.2.1 Datasets utilizados.....	84
6.2.2 Fase de entrenamiento	85
6.2.3 Fase de test.....	89
7 CONCLUSIONES Y TRABAJO FUTURO	92
7.1 CONCLUSIONES	92
7.2 TRABAJO FUTURO.....	94
REFERENCIAS	95
ANEXOS	97
ANEXO A: PUBLICACIONES DERIVADAS DE ESTE PROYECTO	97

INDICE DE FIGURAS

<i>FIGURA 1-1. EJEMPLO DE OBJETO ABANDONADO Y DE OBJETO ROBADO</i>	1
<i>FIGURA 1-2. EJEMPLO DE DETECCIÓN DE OBJETO DE INTERÉS BASADA EN SUSTRACCIÓN DEL FONDO</i> .	2
<i>FIGURA 2-1. CLASIFICACIÓN DE MÉTODOS BASADOS EN DETECTAR REGIONES ESTÁTICAS</i>	13
<i>FIGURA 3-1. DIAGRAMA DE FLUJO DEL SISTEMA BASE</i>	20
<i>FIGURA 3-2. DIAGRAMA DE FLUJO DE LA SEGMENTACIÓN FONDO/FIGURA</i>	21
<i>FIGURA 3-3. MÉTODO DE RECONSTRUCCIÓN DE LA EROSIÓN</i>	22
<i>FIGURA 3-4. EJEMPLO DE ELIMINACIÓN DE RUIDO EN UNA MÁSCARA BINARIA</i>	22
<i>FIGURA 3-5. EJEMPLO DE ELIMINACIÓN DE SOMBRAS EN UNA MÁSCARA BINARIA</i>	24
<i>FIGURA 3-6. EJEMPLO DE RECTÁNGULOS ENGLOBANTES</i>	25
<i>FIGURA 3-6. EJEMPLO DE EXTRACCIÓN DE BLOBS</i>	26
<i>FIGURA 3-8 EJEMPLOS DE CLASIFICACIÓN PERSONA/OBJETO.</i>	28
<i>FIGURA 3-9 EJEMPLO DE CONTORNOS ACTIVOS (SNAKES)</i>	30
<i>FIGURA 3-10 EJEMPLO DEL AJUSTE DEL CONTORNO PARA UN OBJETO ABANDONADO Y OTRO ROBADO</i>	31
<i>FIGURA 4-1 EJEMPLO DEL MÉTODO BASADO EN LA ACUMULACIÓN DE MÁSCARAS DE FG</i>	35
<i>FIGURA 4-2 DIAGRAMA DE TRANSICIONES PARA EL MODELO MOG</i>	36
<i>FIGURA 4-3 EJEMPLO DEL MÉTODO BASADO EN LAS PROPIEDADES DEL MODELO DE FONDO</i>	38
<i>FIGURA 4-4 ESQUEMA DEL MÉTODO BASADO EN EL SUBMUESTREO DE MÁSCARAS DE FG</i>	40
<i>FIGURA 4-5 EJEMPLO DEL MÉTODO BASADO EN EL SUBMUESTREO DE MÁSCARAS DE FG</i>	41
<i>FIGURA 4-6 CLASIFICACIÓN PARA EL MÉTODO QUE USA DOS MODELOS DE FONDO</i>	43
<i>FIGURA 4-7 EJEMPLO DEL MÉTODO BASADO EN MÁS DE UN MODELO DE FONDO</i>	43
<i>FIGURA 4-8 ESQUEMA SEGUIDO PARA OBTENER LA MÁSCARA FDI</i>	46
<i>FIGURA 4-9 EJEMPLO DEL ALGORITMO IMPLEMENTADO</i>	49
<i>FIGURA 4-10 EJEMPLO DE LA SOLUCIÓN AL PROBLEMA DE LAS OCLUSIONES (I)</i>	51
<i>FIGURA 4-11 EJEMPLO DE LA SOLUCIÓN AL PROBLEMA DE LAS OCLUSIONES (II)</i>	51

<i>FIGURA 4-12 EJEMPLO DE LA SOLUCIÓN AL PROBLEMA DE LAS OCLUSIONES (III)</i>	52
<i>FIGURA 4-13 EJEMPLO DE LA SOLUCIÓN AL PROBLEMA DE LAS OCLUSIONES (IV)</i>	52
<i>FIGURA 5-1 EJEMPLO DE FUNCIONAMIENTO DE UN DETECTOR DE ROBO-ABANDONO</i>	56
<i>FIGURA 5-2 EJEMPLO DE DETECTOR BASADO EN COLOR</i>	58
<i>FIGURA 5-3 EJEMPLO DE LOS DETECTORES DE GRADIENTE PARA UN OBJETO ABANDONADO</i>	61
<i>FIGURA 5-4 EJEMPLO DE LOS DETECTORES DE GRADIENTE PARA UN OBJETO ROBADO</i>	62
<i>FIGURA 5-5 ESQUEMA DE FUSIÓN IMPLEMENTADO PARA EL SISTEMA BASE</i>	63
<i>FIGURA 5-6 MOMENTOS DE HU</i>	65
<i>FIGURA 6-1 EJEMPLOS DE IMÁGENES DEL ESCENARIO SENCILLO</i>	76
<i>FIGURA 6-2 EJEMPLOS DE IMÁGENES DEL ESCENARIO COMPLEJO.</i>	77
<i>FIGURA 6-3 DIFERENTES TIPOS DE FONDOS UTILIZADOS (FÁCIL-MEDIO-COMPLEJO)</i>	84
<i>FIGURA 6-4 RESULTADOS OBTENIDOS PARA EL DETECTOR HISTOGRAMA DE COLOR</i>	88
<i>FIGURA 6-5 RESULTADOS OBTENIDOS PARA EL DETECTOR BASADO EN COMPARACIÓN DE HISTOGRAMAS</i>	89

INDICE DE TABLAS

<i>TABLA 5-1. EJEMPLO DEL MODELO DE FUSIÓN DE DEMPSTER-SHAFER.....</i>	<i>73</i>
<i>TABLA 6-1. EVALUACIÓN OBJETIVA DE LOS MÉTODOS IMPLEMENTADOS EN EL ESCENARIO SIMPLE</i>	<i>80</i>
<i>TABLA 6-2. EVALUACIÓN OBJETIVA DE LOS MÉTODOS IMPLEMENTADOS EN EL ESCENARIO COMPLEJO</i>	<i>80</i>
<i>TABLA 6-3. EVALUACIÓN OBJETIVA (PRECISION Y RECALL) DE LOS MÉTODOS IMPLEMENTADOS.....</i>	<i>82</i>
<i>TABLA 6-4. EVALUACIÓN SUBJETIVA DE LOS MÉTODOS IMPLEMENTADOS</i>	<i>83</i>
<i>TABLA 6-5. VALORES DE μ Y σ PARA LOS DIFERENTES TIPOS DE SECUENCIAS CROMA</i>	<i>86</i>
<i>TABLA 6-6. VALORES DE μ Y σ PARA SECUENCIAS CROMA Y REALES.....</i>	<i>87</i>
<i>TABLA 6-7. RESULTADOS DE LOS DETECTORES Y ESQUEMAS DE FUSIÓN IMPLEMENTADOS.....</i>	<i>90</i>

1 Introducción

1.1 Motivación

En la actualidad, el análisis automático de secuencias de vídeo-seguridad se ha convertido en un área de investigación muy activa en respuesta a la creciente necesidad en temas de seguridad [1][2]. Estos sistemas proporcionan herramientas de análisis automáticas con el objetivo de ayudar al personal de seguridad a centrar su atención cuando se encuentran ante una situación compleja.

Problemas como la detección de vehículos estacionados o, la detección de objetos abandonados/robados en lugares públicos (estaciones de metro, aeropuertos, eventos deportivos), cobran cada vez mayor importancia en los sistemas actuales de vídeo-seguridad debido a la gran cantidad de gente que se concentra en esos lugares y los potenciales riesgos de seguridad que ello implica.

La figura 1.1 muestra un ejemplo de un objeto abandonado y un objeto robado



Figura 1-1. Ejemplo de objeto abandonado y de objeto robado

En este tipo de entornos previamente definido, la detección de los objetos de interés es una tarea crucial para identificar las posibles amenazas en dichos entornos. Concretamente, la detección de personas y sus regiones estáticas asociadas (e.g., maleta) en secuencias de vídeo con alta densidad de objetos es un reto para los sistemas actuales, debido a diversos

problemas relacionados con la extracción/mantenimiento del fondo de la escena, variaciones de apariencia del objeto, movimiento de objetos con respecto a la cámara, cambios de iluminación, velocidad de los objetos y oclusiones entre objetos en movimiento y estáticos. Recientemente, la detección de objetos basada en sustracción de fondo¹ se ha vuelto una de las técnicas más utilizadas [3][4][5][10]. Esta técnica trata de detectar los objetos de la escena (llamado *foreground*²) mediante una diferencia entre la imagen actual y el fondo de la escena (*background*). La figura 1.2 muestra un ejemplo de detección de objeto de interés basada en sustracción del fondo:

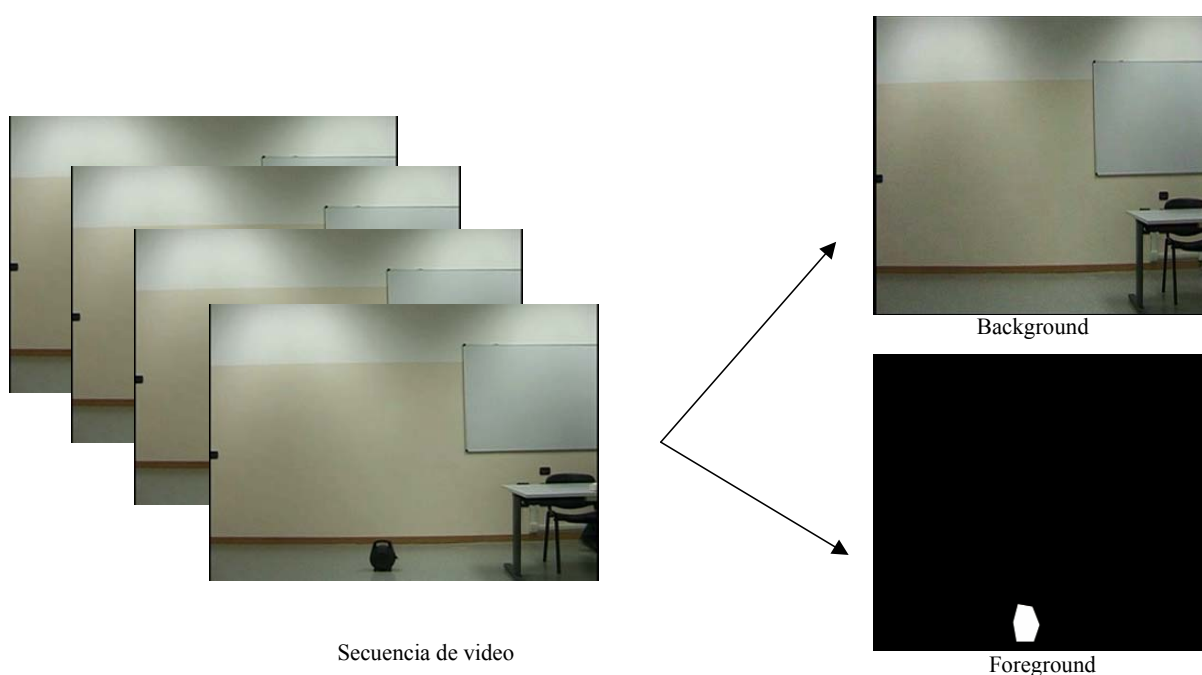


Figura 1-2. Ejemplo de detección de objeto de interés basada en sustracción del fondo

Como se ve en la figura 1.2, el proceso de segmentación devuelve dos imágenes, una de *background*, que se corresponde con el modelo de fondo, y otra de *foreground*, donde se detectan las partes de la imagen que no pertenecen al fondo.

Posteriormente, técnicas de seguimiento de objetos [13] son aplicadas para extraer las trayectorias de los objetos de interés de la escena. Estas trayectorias son utilizadas en etapas de análisis de alto nivel (e.g., detección de eventos) para detectar actividades

¹ A lo largo de esta memoria se utilizarán indistintamente los términos segmentación de fondo, sustracción de fondo y segmentación fondo-figura para referirse a la técnica sustracción de fondo (background subtraction).

² A lo largo de esta memoria se utilizarán indistintamente los términos frente, primer plano y foreground, para referirse a la máscara binaria de objetos en movimiento obtenida tras la etapa de detección de objeto.

potencialmente peligrosas (e.g., abandono objetos). Similarmente a la etapa previa (detección de objeto), la detección en entornos con alta densidad de objetos presenta múltiples problemas principalmente relacionados con oclusiones, velocidad de objetos y variaciones de apariencia.

Finalmente, las etapas de análisis de alto nivel utilizan los resultados de etapas previas (principalmente detección y seguimiento de objetos) para detectar las actividades observadas en la escena [2]. En esta etapa de análisis, las principales dificultades se deben a la definición de las actividades de interés (e.g., accesos a áreas prohibida) en términos de características de alto, medio y bajo nivel.

1.2 Objetivos

El principal objetivo de este PFC es el estudio, evaluación y mejora de un sistema de vídeo-seguridad capaz de detectar objetos abandonados/robados en secuencias complejas con alta densidad de personas. Para llevar a cabo el objetivo propuesto anteriormente, se proponen los siguientes sub-objetivos:

1. Estudio del estado del arte actual

Se realizará un estudio del arte de las diferentes partes de un sistema de vídeo-seguridad centrado en detectar objetos abandonados o robados. Específicamente se analizarán las áreas de detección de objetos de interés, detección de personas y clasificación robo/abandono.

2. Estudio del sistema base para detección de robo/abandono.

Se realizará un estudio en profundidad de las distintas etapas que componen el sistema base [20] disponible en el Grupo de Tratamiento e Interpretación de Vídeo (VPU) analizando las ventajas e inconvenientes que presenta dicho sistema para analizar entornos complejos (con alta densidad de personas).

3. Definición de entornos complejos y adquisición de secuencias de prueba

En esta etapa se estudiarán las secuencias de vídeo existentes para posteriormente clasificarlas dependiendo de su complejidad y utilizarlas como conjunto de secuencias de prueba. La complejidad está definida por la dificultad de extraer los objetos de interés en la escena y la dificultad de modelar el fondo de la escena.

4. Detección de objetos de interés estáticos en entornos complejos

En esta etapa se implementarán varios algoritmos seleccionados del estado del arte basados en la técnica de sustracción del fondo. Posteriormente se evaluarán estudiando sus ventajas e inconvenientes. Finalmente, se diseñará e implementará un algoritmo que detecte dichas regiones de manera robusta y fiable en entornos complejos previamente definidos.

5. Clasificación de objetos de interés entre robo o abandono

En esta etapa se analizarán los detectores existentes en el sistema base y se propondrán nuevos detectores. El objetivo de estos detectores es diferenciar entre robo y abandono. Adicionalmente, se estudiarán nuevos esquemas de fusión con los detectores disponibles. Finalmente se diseñará un proceso de entrenamiento y evaluación estándar para los detectores existentes.

6. Análisis de resultados y conclusiones en entornos complejos

Se analizarán las mejoras propuestas comparándolas frente al sistema base en entornos complejos con alta densidad de personas. Principalmente se evaluará la detección de regiones estáticas y los detectores de robo/abandono propuestos (nuevos detectores y esquemas de fusión), comparándolos frente al sistema base y frente a los resultados de otros investigadores.

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 sistemas de vídeo-seguridad, centrándose en la detección de regiones estáticas así como en la clasificación entre persona-objeto y robo-abandono. Análisis de conjuntos de datos (*datasets*) disponibles.

Capítulo 3: Análisis de un sistema básico de vídeo-seguridad orientado a la detección de objetos abandonados-robados.

Capítulo 4: Detección de regiones estáticas. Implementación de algoritmos significativos. Propuesta de algoritmo de detección de regiones estáticas.

Capítulo 5: Clasificación robo-abandono. Estudio de detectores, y esquemas de fusión.

Capítulo 6: Resultados experimentales

Capítulo 7: Conclusiones y trabajo futuro

Anexo A: Publicaciones derivadas de este proyecto

2 Estado del arte

En este capítulo se describe el estado del arte relacionado con los sistemas de vídeo-seguridad cuyo objetivo es la detección de objetos abandonados o robados. Tras un estudio previo, se decidió que en este tipo de sistemas, las partes más críticas eran: detección de regiones estáticas, clasificación persona-objeto, y por último, discriminación entre objeto robado y abandonado.

Este estudio se encuentra dividido en las siguientes secciones: estudio de las técnicas más relevantes en la detección de regiones estáticas existentes (sección 2.1), técnicas de clasificación entre persona o objeto (sección 2.2), métodos que se usan para decidir si un objeto ha sido robado o abandonado (sección 2.3), y la última sección habla acerca de los *datasets* utilizados en este PFC (sección 2.4).

2.1 Detección de regiones estáticas

2.1.1 Introducción

Analizando las aproximaciones existentes en la literatura, el primer problema con el que nos encontramos cuando se trata de detectar objetos abandonados o robados en una secuencia de vídeo-seguridad es la localización de la región donde sucede algo relevante.

Para la detección de dichas regiones, existen muchas aproximaciones basadas en la técnica de segmentación frente-fondo (*background subtraction*) con el fin de detectar los objetos que pertenecen al primer plano de la imagen. Estas aproximaciones se diferencian entre sí en el tipo de modelo de fondo que utilizan, y en las técnicas basadas en dicho modelo de fondo para encontrar las regiones estáticas.

Por ejemplo, en algunos trabajos se presentan esquemas basados en la distorsión de la cromaticidad para detectar regiones estáticas. Este sistema es bastante robusto, pero también detecta sombras y cambios en la iluminación. En [7], el autor presenta un método basado en medidas espectrales, espaciales y temporales, que se incorporan a un espacio de trabajo Bayesiano cuya misión es caracterizar el modelo de fondo para cada píxel, con el fin de detectar regiones estáticas.

Actualmente, modelos de caracterización de los píxeles del fondo basados en el método de mezcla de Gaussianas (MoG) se utilizan en bastantes aproximaciones [3][4][5][10] debido

a que tiene en cuenta a la hora de modelar el fondo posibles cambios de iluminación en la imagen, secuencias multimodales, objetos moviéndose lentamente, y el ruido introducido por la cámara. Un ejemplo de métodos basados en MoG se encuentra en [19].

A su vez, la detección de regiones estáticas es muy importante por dos razones: debe mejorar la calidad de los datos inicialmente detectados en la segmentación frente-fondo, y, una vez seleccionado el objeto perteneciente al frente de la imagen, nos va a permitir realizar una clasificación entre persona/objeto y una discriminación entre robo o abandonado.

En los últimos tiempos, gran parte de la investigación en aplicaciones de vídeo-vigilancia (p.e., detección robo/abandono) se ha centrado en detectar regiones estáticas, y tratar de realizar una clasificación correcta entre personas y objetos. Por ejemplo, en algunos ejemplos se usa un sistema que acumula los resultados de regiones estáticas del *foreground* de modo temporal, y con ello es capaz de estimar si dichas regiones estáticas son personas, efectos luminosos u objetos.

Algunas aproximaciones usan un modelo de máquina de estados para modelar un posible abandono de objetos, cuya finalidad es poder detectar regiones estáticas. En otras, se usa una lógica matemática capaz de diferenciar entre objetos y personas estáticas dentro de la máscara final de *foreground*.

En [14], se presenta una aproximación basada en detectar regiones estáticas mediante el uso de un contador que se actualiza en una etapa de *tracking*. En [7], la secuencia de vídeo de entrada se procesa a diferentes velocidades, produciendo dos modelos de fondo, uno que se actualiza a largo plazo y otro que se actualiza a corto plazo. Con esos modelos se definen unas reglas para detectar regiones estáticas.

Estos ejemplos descritos anteriormente no distinguen si los cambios producidos en las regiones estáticas que no son clasificadas como personas se deben a un objeto abandonado o a un objeto robado. Este proceso se explicará más adelante en este capítulo del PFC.

En esta sección se describe la técnica mas utilizada para separar las partes de la escena que no pertenecen al fondo de la misma, que es la técnica de segmentación fondo-frente. A continuación, se describe su uso para la detección de regiones estáticas, y por último se detalla una clasificación realizada con las técnicas más relevantes encontradas en la literatura.

El apartado está dividido en dos sub-apartados: El primer apartado aportará algunas características necesarias para la etapa de segmentación fondo-frente, y resumirá alguno de los métodos más utilizados para ello (apartado 2.1.1), y el segundo apartado establece una clasificación de métodos basados en segmentación fondo-frente para detectar regiones estáticas (2.1.2).

2.1.2 Características de la etapa de segmentación fondo-frente

La segmentación de objetos habitualmente tiene como objetivo discriminar los objetos en movimiento del primer plano (*foreground*) de una imagen, del resto de los objetos o fondo (*background*).

Existen muchas técnicas distintas de segmentación de secuencias de vídeo dependiendo del tipo de secuencias a analizar y de los resultados que se esperan obtener. La segmentación que emplearemos en este proyecto se basa en el modelado de fondo, que consiste en una serie de algoritmos que mantienen un modelo matemático de apariencia del fondo a través del cual, para cada imagen, se pueden extraer mediante el uso de ciertos algoritmos (p.e. comparación) los objetos en movimiento (primer plano) y el fondo.

Debido a que las secuencias de vídeo no son ideales (es decir, en la escena a analizar no nos encontramos sólo con regiones estáticas que se diferencien del fondo), la etapa de segmentación presenta múltiples complicaciones, siendo una de las tareas más complejas y delicadas dentro del procesamiento de la señal de vídeo en un sistema de vídeo-seguridad. Un buen algoritmo de segmentación debe ser capaz de detectar objetos en movimiento, y debe lograr mediante el uso de un bajo coste computacional y en tiempo real, solventar los siguientes problemas:

Ruido: Un buen algoritmo debe de ser capaz de eliminar el ruido procedente de la captación de imágenes por las cámaras de vídeo, ya que puede producir errores en la segmentación y generar nuevas regiones de frente.

Sombras y reflejos: Las fuentes luminosas y los diferentes objetos de la escena pueden interferir entre sí, y generar efectos que suelen ser categorizados como objetos en movimiento (sombras y reflejos). Como es de esperar, las sombras y los reflejos no forman parte del frente de la imagen (*foreground*), pero tampoco forman parte del fondo (*background*), por lo tanto deben ser eliminadas. Sin embargo, las sombras inherentes al fondo deben analizarse como parte de él. Esta eliminación de sombras y reflejos es

complicada de obtener en un modelo paramétrico de fondo, así que para resolver dicho inconveniente, una vez realizada la segmentación y quedarnos con la máscara de *foreground*, aplicaremos un post-procesado a dicha máscara para eliminar los efectos producidos por las sombras y los reflejos, que se puede basar en técnicas de filtrado y en operaciones morfológicas (erosión y dilatación).

Cambios de iluminación: Son variaciones de iluminación que se pueden producir a lo largo del día si la escena ha sido capturada al aire libre, o tratándose de una secuencia de interior, por las distintas fuentes de iluminación de nuestra secuencia. Estos cambios pueden ser malinterpretados por nuestro modelo de segmentación, reconociendo dichos cambios como objetos en movimiento. Para evitar este fenómeno, hemos de tener un modelo que actualice el valor de nuestro modelo de fondo correctamente imagen a imagen (*frame a frame*).

Actualización del fondo de la escena: Esta es una característica muy importante de todo modelo de segmentación para la detección de objetos abandonados/robados en secuencias de vídeo-seguridad, por varias razones:

Inicialización del fondo de la secuencia: Es muy complicado que, cuando empezamos a grabar una secuencia, el modelo de fondo del primer *frame* coincida con el modelo de fondo buscado. Este hecho puede ser debido a que, en ese momento, existan objetos en movimiento en la escena que no puedan ser discriminados como fondo al no haber un modelo anterior (por ejemplo, una persona moviéndose en la escena). El modelo de segmentación tiene que ser capaz de, con el paso del tiempo (*frames*), actualizar dicho modelo y eliminar a la persona en movimiento del fondo de la escena.

Detección de cambios importantes en la secuencia: En los sistemas orientados a la detección de objetos abandonados/robados, es importante que el modelo de segmentación actualice dicho modelo de fondo cuando un objeto ha sido abandonado en la escena, con el fin de que dicho objeto pertenezca al fondo de la imagen para un procesado futuro. Lo mismo ocurre cuando se trata de un objeto robado.

Determinar los parámetros de funcionamiento de los algoritmos: Una de las tareas más complejas e importantes de este tipo de sistemas automáticos de análisis es el ajuste de los parámetros de los algoritmos que determinan el fondo y los objetos en las secuencias. En cada algoritmo, podemos encontrar diferentes parámetros (umbrales, pesos,..) con distintas

sensibilidades. Los parámetros más significativos serán los que alteren en mayor medida los resultados del sistema y por ello, hemos de ser capaces de identificarlos y estudiarlos a fondo en función de las características de las secuencias y de los resultados que esperamos obtener.

Fondos multimodales: Son el tipo de fondo que podemos encontrar en secuencias con objetos en movimiento lento y/o periódico (movimiento de las hojas de los árboles, movimiento ondulatorio del agua,...). Nuestro algoritmo ha de ser capaz de reconocer estos fondos multimodales como fondo de la imagen, y nunca como frente

Los principales métodos utilizados para la segmentación fondo frente se pueden clasificar en modelos paramétricos, entre los que destacan el método de la Gaussiana simple [23], el modelo de mezcla de Gaussianas (MoG) [19] y el segmentador Gamma [24], y modelos no paramétricos, donde destaca el método basado en estimar la densidad del núcleo (KDE) [25], o métodos basados en modelos ocultos de Markov (HMM) [26].

Con el fin de extraer correctamente los objetos pertenecientes al *foreground* para su posterior análisis, y evitar todos estos contratiempos, los modelos más usados por los investigadores, y los que se usarán en este PFC son el modelo de mezcla de Gaussianas y el segmentador Gamma.

2.1.3 Clasificación de métodos basados en segmentación fondo-frente para detectar regiones estáticas

En esta sección se describe una clasificación de métodos de detección de regiones estáticas con el fin de establecer diferentes categorías (ver figura 2.1). Una característica a destacar de esta clasificación es que no se va a tener en cuenta el seguimiento de objetos (*tracking*) para los distintos métodos implementados.

Para empezar, hemos decidido dividir las aproximaciones existentes en dos categorías, dependiendo si usan un modelo de fondo o más de un modelo de fondo.

A su vez, dependiendo del uso que dichas aproximaciones hagan con las máscaras de *foreground* obtenidas, dichas aproximaciones pueden clasificarse en:

- **Análisis imagen a imagen.** Esta categoría describe los métodos que emplean modelos de segmentación frente-fondo bastante comunes, seguidos de otro tipo de análisis. En función de dicho tipo de análisis, pueden aparecer diferentes categorías, que son las siguientes:

- Basados en el uso de técnicas sencillas de segmentación frente-fondo, un post-procesado de la máscara de *foreground* seguido a su vez de alguna otra etapa de análisis (por ejemplo, *tracking*). Esto se conoce como aproximaciones clásicas [3][10][18][19][20].
- Basados en la acumulación de máscaras de *foreground*. Dicha acumulación se realiza *frame a frame* y con ella se puede modelar una máscara final de *foreground*, de donde se extraen las regiones estáticas [9][10][14].
- Basados en algunas propiedades del modelo de fondo utilizado, como por ejemplo considerando las transiciones entre los diferentes estados de un modelo de mezcla de Gaussianas, o observando el valor de algunos parámetros como, por ejemplo, el peso de las Gaussianas [11][12][13][15].
- **Análisis de máscaras de foreground muestreadas.** Estas aproximaciones intentan detectar regiones estáticas analizando la secuencia de vídeo a diferentes velocidades, aprovechándose de las ventajas espacio-temporales que ello conlleva [8][17].

A su vez, existen aproximaciones que combinan más de un modelo de fondo para cada píxel. Este tipo de aproximaciones han sido menos utilizadas por los investigadores para tratar de detectar regiones estáticas. Sin embargo, en función de la tasa binaria de procesamiento del vídeo, o del número de modelos de fondo utilizados para detectar regiones estáticas, podemos hacer esta clasificación:

- **Aproximaciones basadas en el análisis imagen a imagen.** En esta categoría tenemos métodos que combinan las propiedades de los diferentes modelos de fondo que utilizan. [21].
- **Aproximaciones basadas en el sub-muestreo.** Estas aproximaciones detectan regiones estacionarias analizando la secuencia de vídeo a través de los diferentes modelos de fondo debido a que cada modelo de fondo se muestrea con una tasa binaria diferente [7].

La figura 2.1 muestra un esquema de dicha clasificación.

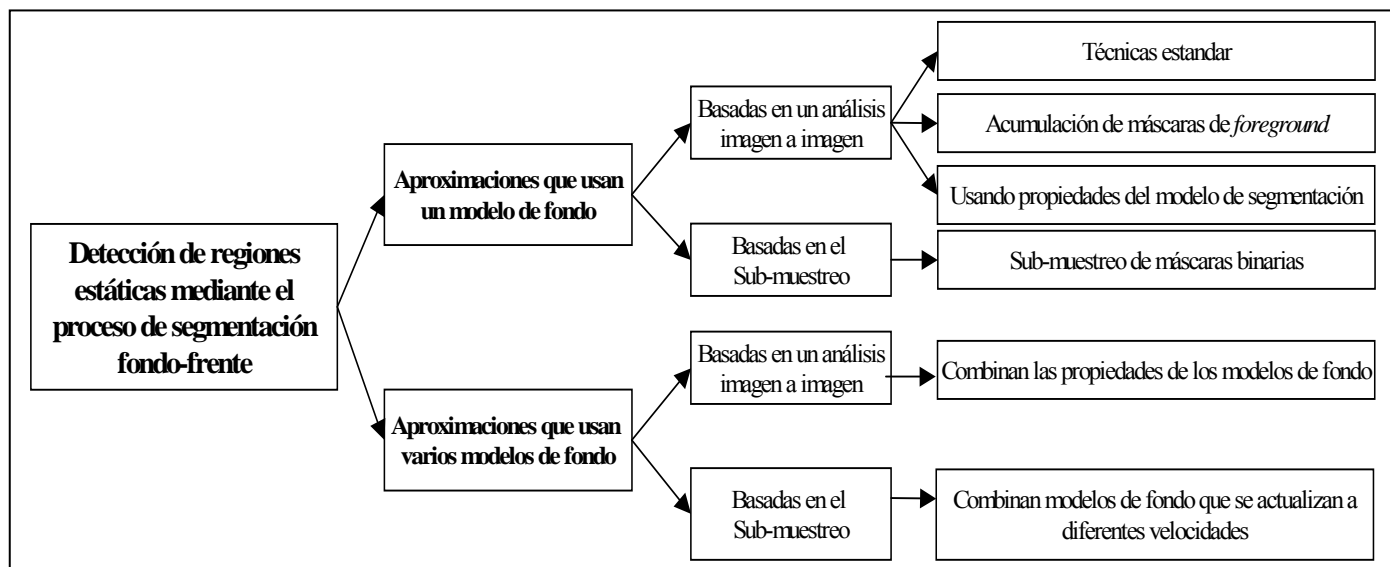


Figura 2-1. Clasificación de métodos basados en detectar regiones estáticas

2.2 Clasificación persona-objeto

La clasificación automática de las regiones estáticas entre persona y objetos que se realiza en secuencias de vídeo es un problema complejo a medida que incrementamos el número de tareas a realizar. Pero este problema no está sólo limitado a la vigilancia automatizada de vídeo y la supervisión. La complejidad de este problema es, principalmente, debido a dos motivos.

En primer lugar, no hay ningún modo fácil de definir un modelo que caracteriza como una persona aparecerá en una imagen, ya que la gente puede adoptar una enorme variedad de las posturas que rápidamente pueden cambiar, por ejemplo, andando, sentándose en un banco, levantándose, etc. Este reconocimiento a su vez se complica debido a la ropa, accesorios personales (paraguas, mochilas, etc) o incluso el peinado.

En segundo lugar, las secuencias de vídeo deben ser analizadas en tiempo real en una mayoría de usos, lo que dificulta mucho los algoritmos a implementar en cuanto a coste computacional.

El algoritmo de clasificación destinado a determinar si un objeto es persona o no se denomina detector de personas. Estos detectores pueden ser clasificados en dos amplias familias dependiendo si analizan contornos o regiones.

La mayoría de detectores de personas pertenecen a los basados en análisis de contorno. En muchos casos, estos detectores aplican una primera fase de entrenamiento para determinar una amplia gama de posturas de la gente típicas (andando, alguien parado en algún sitio, alguien sentado, etc.).

Algunas técnicas descritas en [22] determinan la postura y la posición de diferentes partes del cuerpo de la silueta de una persona. Para ello, se calculan las partes cóncavas y convexas del contorno dado. Los vértices de los polígonos formados y las direcciones principales de la silueta son usados para predecir la posición de la cabeza y, a partir de entonces, torso, pies y manos. Otro método descrito en [22] calcula de modo iterativo la elipse más grande contenida en cada región estática obtenida después de la etapa de segmentación. La relación de aspecto de las elipses y las iteraciones requeridas para encontrarlos permite al sistema para determinar si la silueta corresponde a una persona o no.

Estas dos técnicas son de las más implementadas en la actualidad debido a sus buenos resultados, y a su simplicidad y eficacia para integrarse en sistemas que trabajan en tiempo real.

2.3 Discriminación abandono-robo

En esta sección, se describen las aproximaciones existentes para discriminar los objetos estáticos entre robados o abandonados. Como se avanzó en el apartado 2.1 de este PFC, muchos artículos de investigación centran el esfuerzo en la detección de regiones de interés, pero pocos se centran en distinguir objetos abandonados de objetos robados, debido a que normalmente se suelen imponer condiciones contextuales a las secuencias a analizar (p.e., que solo sea posible el abandono de objetos).

Por lo tanto, detectar un objeto abandonado o un objeto robado se convierte, en muchos casos, en un problema de seguimiento de la persona que se acerca a la región estática

algunas imágenes antes de que se detecte la región estática. (Esto es debido a, que, tanto para abandonar un objeto, como para robarlo, es necesario que antes una persona haya pasado por la misma región).

En [15], se pretende detectar objetos abandonados o robados proporcionando una mayor descripción semántica de la detección mediante la extracción de características de dichas regiones estáticas. Posteriormente, estas características se utilizan para distinguir entre un objeto abandonado y un objeto robado.

En [12], se propone un método basado en el estudio de las características del contorno de la región estática a analizar. Dichos estudios se centran en estudiar el cambio de la energía del contorno de la región estática perteneciente a la imagen actual frente a la energía que posee la imagen de fondo original en las mismas coordenadas donde se sitúa la región estática en la imagen actual.

Análogamente, en [16], usan un modelo estadístico de la imagen de fondo con el objetivo de detectar regiones estáticas, para después post-procesar dichas regiones eliminando posibles sombras y ruido. Más tarde, las regiones estáticas se clasifican como objetos abandonados o robados comparando sus bordes con los bordes de la máscara de *foreground* obtenida.

En [11], los objetos extraídos por el sistema de vídeo-vigilancia se clasifican como persona en movimiento, persona detenida, objeto en movimiento o objeto detenido. En la siguiente etapa, sólo se estudian los objetos detenidos para detectar si corresponde a un objeto abandonado, o a un objeto robado. Esto se consigue analizando el histograma de color de dicha región estática a la que pertenece el objeto en la imagen actual y en la imagen de fondo.

En [20], tras discriminar primero las regiones estáticas que corresponden a personas humanas, se propone un método basado en la fusión de algunos detectores sencillos para mejorar la discriminación entre robo y abandono.

En definitiva, las técnicas más relevantes de la literatura centradas en diferenciar objetos abandonados de objetos robados usan información o bien del contorno del objeto a analizar, o bien información del color de la región a analizar.

A su vez, muchos estudios han debatido acerca de las ventajas y las desventajas de usar esquemas de fusión para combinar todas estas propuestas existentes en la literatura. En este

PFC, se han utilizado esquemas de fusión ya que, como se verá a lo largo de esta memoria, obtienen mejores resultados.

2.4 Datasets utilizados

En este apartado se hace referencia a las secuencias de test más significativas utilizadas en este PFC. Estas secuencias son bastante conocidas en el mundo de la vídeo-seguridad, y se utilizan para realizar diferentes pruebas.

Dichas secuencias se muestran a continuación:

- PETS2006 (<http://pets2006.net>).

Las secuencias de vídeo-seguridad que pertenecen a este escenario contienen escenas de abandono de objetos en escenarios sencillos cuya dificultad aumenta de manera progresiva. El *dataset* está formado por siete secuencias que se observan desde 4 cámaras diferentes.

- PETS2007 (<http://pets2007.net>).

Este *dataset* está formado por tres escenarios: un escenario sencillo de seguimiento de objetos, otro escenario de robo de objetos y el último escenario perteneciente al abandono de objetos. Estos escenarios tienen diferentes secuencias con un nivel de complejidad que aumenta de manera progresiva.

- AVSS2007 (<http://www.elec.qmul.ac.uk/staffinfo/andrea/avss2007.html>)

Este *dataset* cuenta con tres secuencias de entrenamiento basadas en detectar objetos abandonados en estaciones de metro densamente pobladas con niveles de dificultad que varían de unas secuencias a otras.

- CAVIAR (<http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>)

En este *dataset* se analizan escenas sencillas basadas en el abandono de objetos. Destacar que en este *dataset* la cámara graba desde otro punto de vista diferente al de los anteriores *datasets*.

- VISOR (http://www.openvisor.org/video_videosInCategory.asp?idcategory=14)

Dataset formado por 10 secuencias simples con casos de abandono de dificultad sencilla.

- CANDELA (<http://www.multitel.be/~va/candela/>)

Dataset de un nivel de dificultad más complejo que el anterior, formado por 16 secuencias de abandono de objetos en entornos algo poblados.

- WCAM (http://wcam.epfl.ch/seq_video_surv/seq_video_surv.html)

Dataset formado por 7 secuencias a analizar, tanto de robo como de abandono. Las secuencias son de una dificultad bastante sencilla, al no haber apenas movimiento en la escena.

3 Sistema base en detección de objetos abandonados/robados

3.1 Estructura General

En este proyecto se parte de un sistema completo desarrollado para evaluar la detección de objetos abandonados o robados en secuencias de vídeo-seguridad, descrito en la figura 3.1. Este sistema está diseñado para poder trabajar en tiempo real como parte de un sistema de vídeo-vigilancia. Esto implica que el coste computacional de los algoritmos de procesamiento de imágenes no ha de ser elevada. Los algoritmos de análisis seleccionados tienen que tener una carga computacional y un tiempo de ejecución acorde a la información que vayan a generar.

Primeramente se realiza una etapa de adquisición del vídeo, para, a continuación, aplicar una etapa de segmentación fondo-figura basada en sustracción del fondo, con el objetivo de detectar los píxeles que no pertenecen al fondo de la imagen (estos píxeles pertenecerán al *foreground* de la imagen). A su vez, el módulo de extracción de *blobs* analiza las regiones compactas de la máscara binaria de *foreground* obtenida para detectar *blobs*.

Después, el módulo de seguimiento de *blobs* (*tracking*), genera la trayectoria de los *blobs* entre imágenes (*frames*) consecutivos usando información de color y de posición.

A continuación, el sistema analiza el movimiento de los *blobs* para obtener aquellos que son estáticos, que se corresponderán con las regiones estáticas que no pertenecen al fondo de la imagen, para clasificarlas en persona u objeto.

En el módulo de detección de objetos abandonados o robados, los *blobs* clasificados como objetos estáticos son analizados para determinar si son objetos abandonados o robados.

Este capítulo se describen brevemente los modelos y métodos utilizados en las diferentes etapas de análisis: Segmentación de fondo/figura (sección 3.2), seguimiento de *blobs* (sección 3.3), detección de regiones estáticas (sección 3.4) y detección de objetos abandonados/robados (sección 3.5).

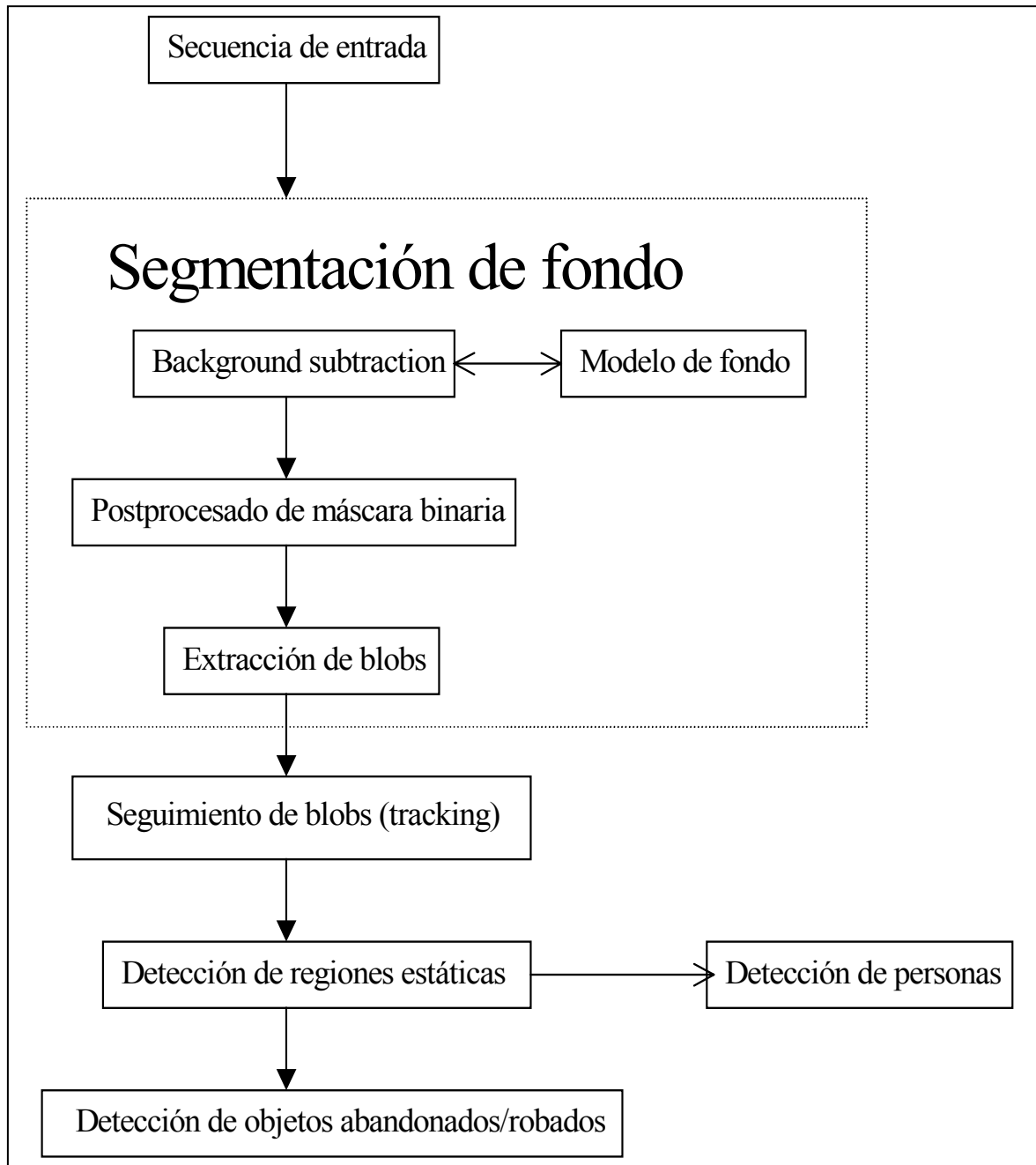


Figura 3-1. Diagrama de flujo del sistema base

3.2 Segmentación de fondo/figura

El diagrama de flujo de la segmentación de fondo/figura se muestra en la figura 3.2, y está basado en la técnica de segmentación fondo-figura y un post-procesado del resultado obtenido. Este método ha sido seleccionado debido a su bajo coste computacional y al uso de cámaras estáticas que permiten mantener fácilmente el mismo modelo de fondo.

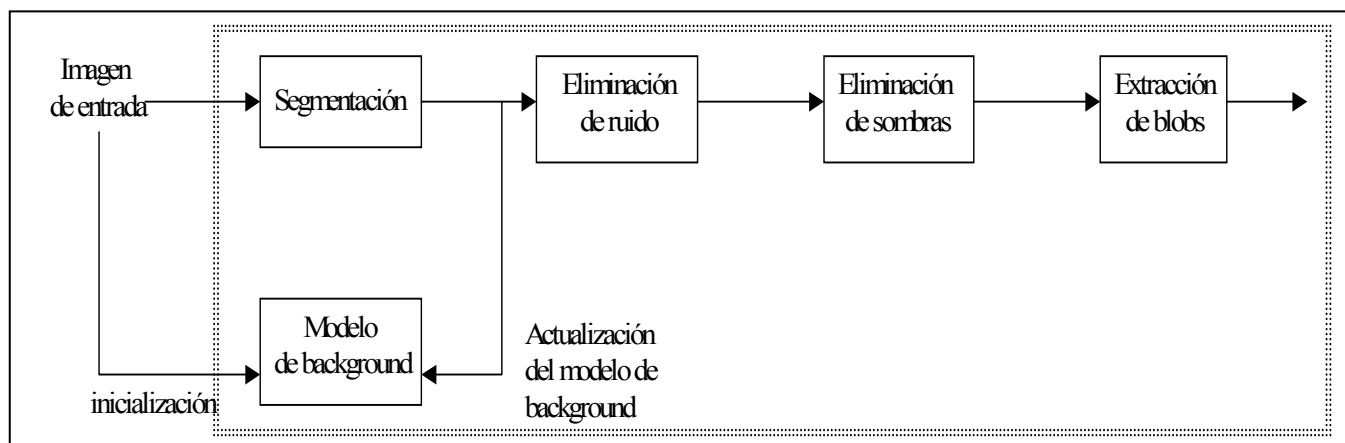


Figura 3-2. Diagrama de flujo de la segmentación fondo/figura

El módulo está compuesto de 4 etapas, con el objetivo de extraer los píxeles pertenecientes al *foreground* de la imagen. La primera etapa es la segmentación fondo/figura. En esta etapa se decide qué píxeles pertenecen al fondo de la escena y cuales al primer plano o figura.

La segmentación se realiza mediante el segmentador Gamma. En el segmentador Gamma el modelo de fondo se representa, para cada píxel, por 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 los píxeles de fondo y objeto se realiza píxel a píxel mediante el cálculo de la diferencia generada por una ventana cuadrada de lado $2W$ alrededor del píxel a analizar tanto de la imagen actual I como del modelo de fondo existente B . Cuando esa distancia elevada al cuadrado supera un cierto umbral β , dicho píxel se considera un píxel perteneciente al *foreground*. La ecuación 3.1 muestra la fórmula correspondiente:

$$foreground(I[x, y]) \Leftrightarrow \sum_{i=-W}^W \sum_{j=-W}^W (I[x+i, y+j] - B[x+i, y+j])^2 > \beta \quad (3.1)$$

Posteriormente, se realiza una eliminación de ruido en la máscara binaria obtenida. Esta eliminación se lleva a cabo usando operadores morfológicos. Específicamente, se aplica una combinación sencilla de las operaciones morfológicas de erosión y dilatación conocida como reconstrucción de la erosión (ver figura 3.3). La principal ventaja de esta operación radica en que conserva la forma de los objetos pertenecientes a la máscara de *foreground*

eliminando pequeñas impurezas que se puedan observar en la máscara binaria. El tamaño de las impurezas eliminadas (que se suele corresponder a ruido aleatorio generado por el dispositivo que captura la escena) depende del tamaño del elemento estructurante usado en las operaciones de erosión y de dilatación.

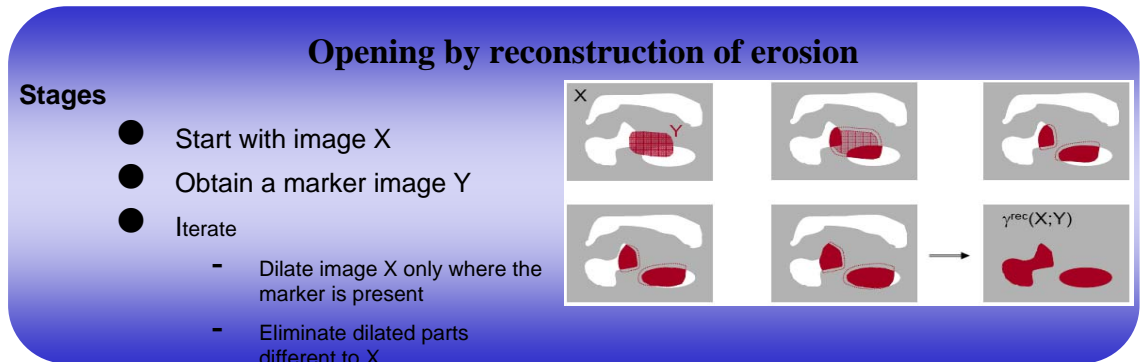


Figura 3-3. Método de reconstrucción de la erosión

A continuación, se muestra un ejemplo de reducción de ruido en una máscara binaria con ruido, sin procesar (a), y el resultado de aplicar a dicha máscara operadores morfológicos (b) (figura 3.4):

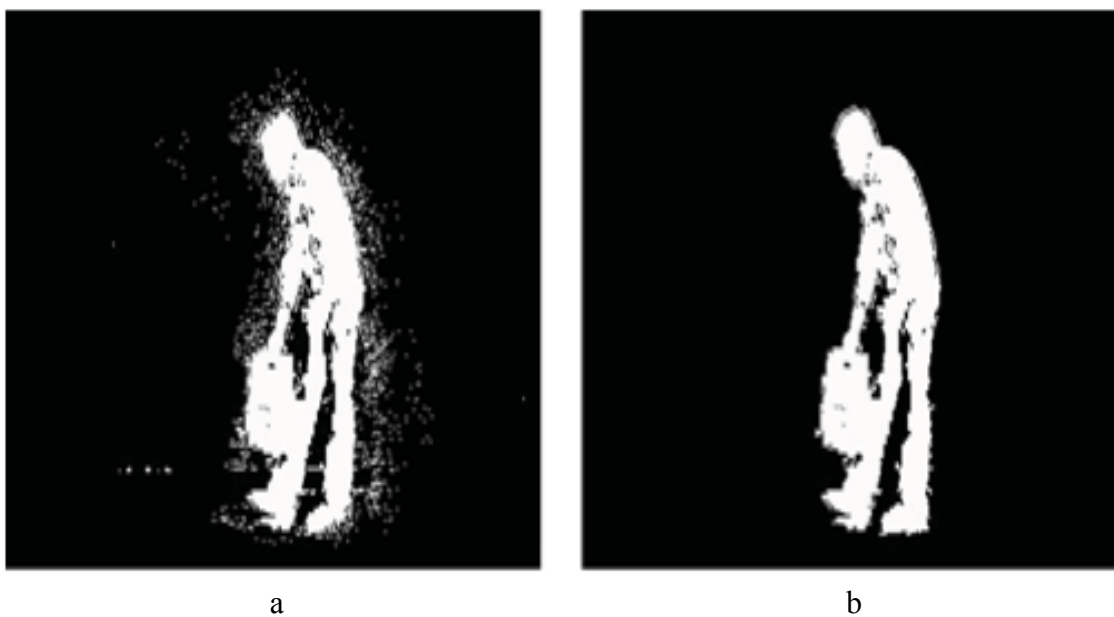


Figura 3-4. Ejemplo de eliminación de ruido en una máscara binaria

Tras la etapa de eliminación de ruido, se pasa a la etapa de eliminación de sombras. En la etapa de segmentación de objetos con respecto del fondo de la imagen, casi siempre suelen aparecer sombras pertenecientes a los objetos o a las personas que se clasifican como parte del frente de la imagen. Para ello se ha decidido utilizar el espacio de color HSV en lugar del espacio de color RGB utilizado en las imágenes del sistema base. El motivo principal es la separación entre intensidad y color que se produce en el espacio HSV permitiendo hacer un análisis exhaustivo de cada uno.

Tras la conversión de las imágenes de fondo y actual del espacio de colores RGB al espacio HSV, se calculan 3 mapas de colores sobre las tres componentes (H, S y V) con el objetivo de detectar los píxeles pertenecientes a sombras en la máscara de *foreground* obtenida a la salida del detector de ruido. Primero, el Valor (V) entre la imagen de fondo y la imagen actual se usa para filtrar los posibles píxeles pertenecientes a sombras. A continuación, el algoritmo se basa en que el cambio en la cromaticidad (calculado a partir de la Tonalidad (H) y la Saturación (S)) entre la imagen actual y la imagen de background no es muy grande. La decisión de si un píxel pertenece a la sombra o no, se basa en la siguiente ecuación:

$$SP_t(x, y) = \begin{cases} 1 & \text{if } \alpha \leq \frac{I_t^V}{B_t^V} \geq \beta \\ & \wedge \|I_t^S(x, y) - B_t^S(x, y)\| \leq \tau_S \\ & \wedge \|I_t^H(x, y) - B_t^H(x, y)\| \leq \tau_H \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

donde α y β son constantes que marcan que la relación entre los valores de V entre la imagen actual (I_t^V) y la imagen del fondo (B_t^V) no superan ni rebajan un cierto umbral, y la diferencia entre los valores tanto de S (I_t^S y B_t^S) como de H (I_t^H y B_t^H) tampoco superan un cierto umbral. Los cálculos se hacen píxel a píxel (en el caso de dicho ejemplo, píxel para la posición (x,y)).

Para concluir, los píxeles clasificados como fondo se eliminan de la máscara anterior para obtener una nueva máscara binaria ya sin sombras ni ruido. A continuación se muestra la eliminación de sombras en la figura 3.5:

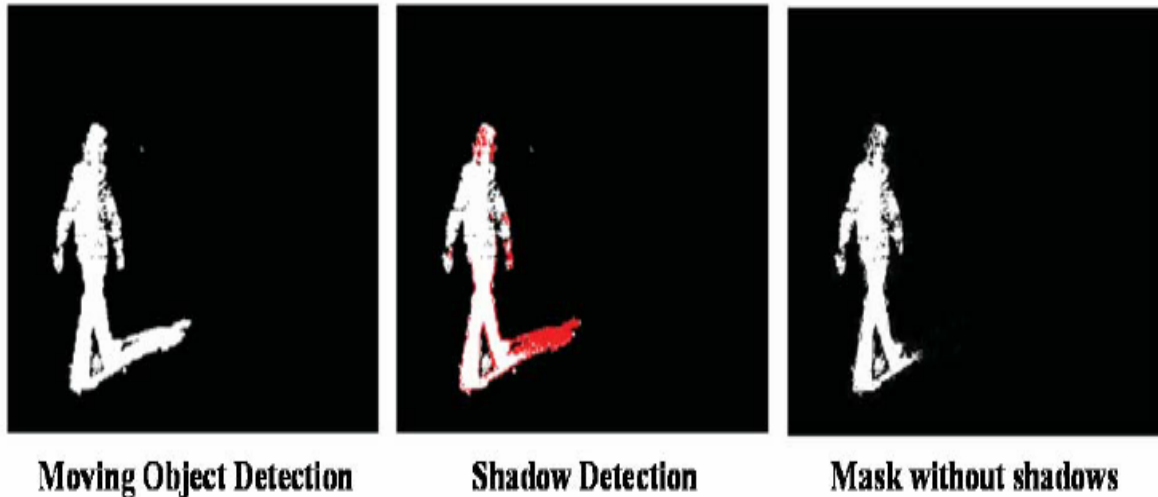


Figura 3-5. Ejemplo de eliminación de sombras en una máscara binaria

Tras la detección y post-procesado de la máscara de píxeles de primer plano, el módulo de extracción de *blobs* extrae información acerca de componentes conexas en la máscara binaria obtenida anteriormente. Esta operación se realiza en dos etapas: la primera etapa se basa en la detección de *blobs*, y la segunda es una etapa de post-procesado.

El objetivo de la etapa de detección de *blobs* es la extracción de componentes conexas de la máscara binaria. El algoritmo empleado es “*Connected Component Algorithm using 8 neighborhood*”. Este algoritmo está basado en un análisis para cada píxel y una comparación con sus píxeles vecinos agrupando aquellos que tienen valores similares obteniendo regiones conexas. A cada región conexa se le asigna una etiqueta diferente. Además, se calculan los rectángulos englobantes (o *Bounding Box*) de cada uno de los *blobs* seleccionados (ver ejemplos en la figura 3.6).



Figura 3-6. Ejemplo de rectángulos englobantes

Tras este proceso, se lleva a cabo una etapa de post-procesado de los *blobs* obtenidos anteriormente. Debido a restricciones de la aplicación o a consideraciones sobre el ruido introducido en la máscara de *foreground*, algunos de ellos pueden ser demasiado pequeños para ser considerados como *blobs*. Este problema se puede solucionar eliminando las regiones cuya área es más pequeña que un umbral determinado (el área se mide en píxeles). Para cada imagen, ese umbral se calcula como un porcentaje del tamaño medio del rectángulo envolvente de un *blob*, y los rectángulos envolventes obtenidos cuyo tamaño sea menor al de dicho umbral, serán eliminados de la lista final de regiones consideradas como parte de la máscara de objetos. Se puede ver un ejemplo de este proceso en la figura 3.7.

La figura 3.7 muestra una imagen de una secuencia de vídeo cualquiera (a), la imagen de fondo (b), la máscara binaria de objetos extraída (c), los *blobs* etiquetados cada uno con un color diferente una vez la máscara obtenida anteriormente ha sido post-procesada (d), los rectángulos envolventes para cada uno de los *blobs* obtenidos (e), para, finalmente, quedarse con aquellos cuyo rectángulo envolvente está por encima de cierto umbral (f).

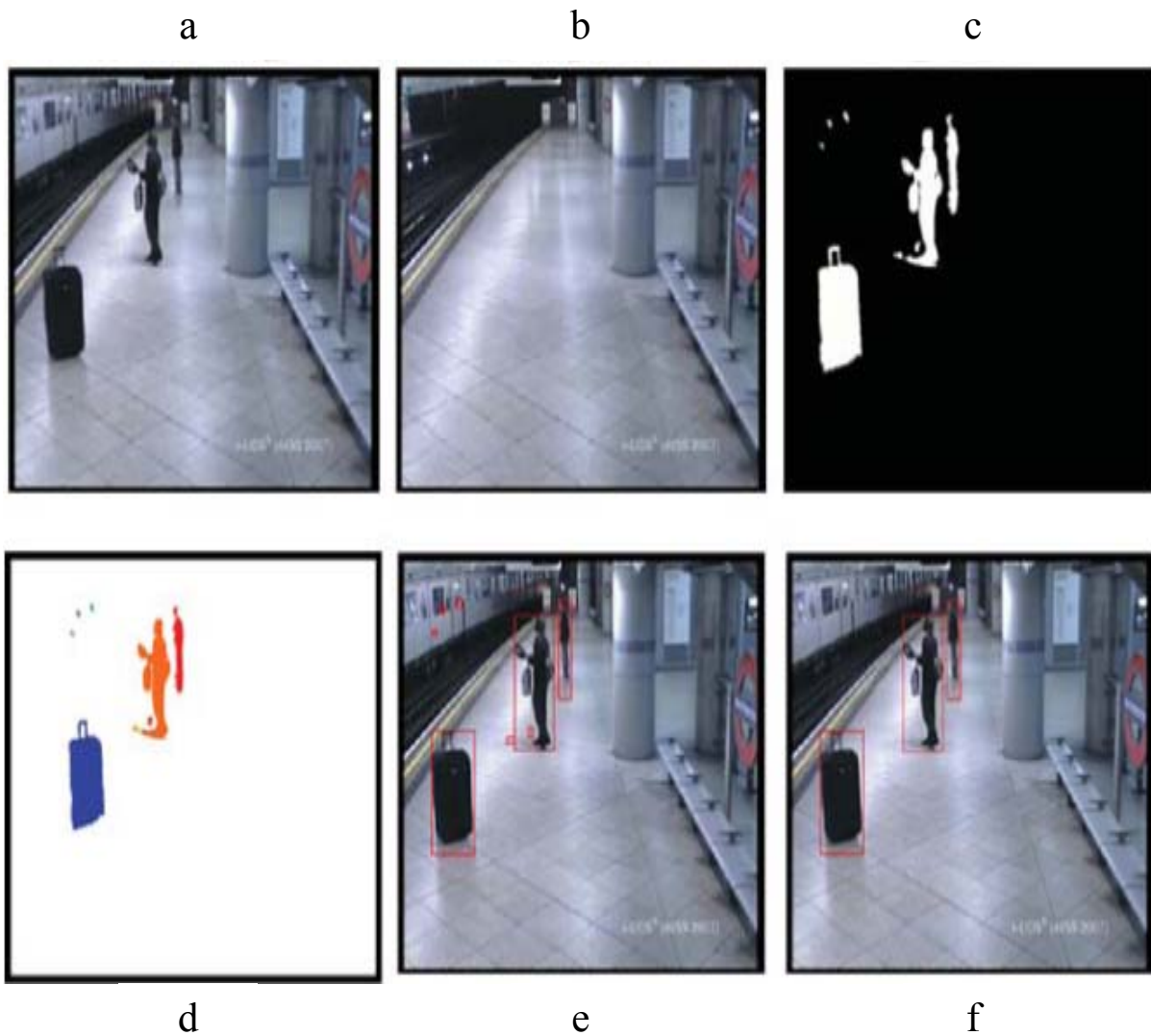


Figura 3-7. Ejemplo de extracción de blobs

3.3 Seguimiento de blobs

Este módulo realiza un seguimiento sobre los *blobs* relevantes de cada imagen identificados en el módulo anterior. Primeramente, se identifican los diferentes *blobs* en la imagen actual y posteriormente se trata de encontrar semejanzas entre dichos *blobs* y los *blobs* determinados en la imagen anterior. Así pues el problema de seguimiento de *blobs* se traduce en un problema de correspondencia entre imágenes consecutivas. Específicamente, este módulo calcula una matriz de *matching* MM para determinar la correspondencia. Esta matriz esta basada en:

- Distancia euclídea entre posiciones de centroides (X,Y)

- Distancia euclídea en el espacio de color RGB

La construcción de la matriz MM para cada posición se realiza de la siguiente manera:

$$MM_{j,i} = \sqrt{(\Delta Y / Y \text{ dim})^2 + (\Delta X / X \text{ dim})^2 + \sqrt{(\Delta R / 255)^2 + (\Delta G / 255)^2 + (\Delta B / 255)^2}} \quad (3.3)$$

donde los subíndices j, i indican un píxel de la imagen, ΔY y ΔX son las distancias euclídeas entre posiciones de centroides, normalizados frente a su valor máximo posible, $Y \text{ dim}$ y $X \text{ dim}$ respectivamente, que son las dimensiones de la imagen a analizar. ΔR , ΔG y ΔB son las distancias euclídeas, también normalizadas frente a su posible valor máximo, que en este caso es 255.

3.4 Detección de objetos estáticos

A continuación, se pasa a la etapa de detección de objetos estáticos, que se basa en dos premisas principales: la detección de *blobs*/regiones estacionarios y su clasificación entre persona y objeto.

La detección de regiones estacionarias se consigue estudiando la velocidad de cada *blob* extraído. Si un *blob* permanece estático (es decir, si su velocidad es aproximadamente 0), será considerado como estático durante esa imagen. En este sistema, se decidió considerar una región del *foreground* como estática si no cambiaba durante 50 imágenes consecutivas.

A continuación, se clasifica cada *blob* estacionario como persona o como objeto. Este proceso es muy importante en la detección de eventos en sistemas de vídeo-seguridad. Para la clasificación de los *blobs*, se usa la combinación de dos algoritmos sencillos:

- El primer algoritmo se basa en una sencilla medida, el ratio de cada *blob*. Dicho ratio se calcula como la relación entre el ancho (w) y la altura (h) de un *blob* de la forma $\text{ratio} = w/h$, y basándose en las pruebas realizadas, se puede aproximar dicho ratio a una distribución Gaussiana de $\mu = 0.3$ y $\sigma = 0.2$.

- El segundo algoritmo se basa en el porcentaje de píxeles con valor 1 (en blanco) dentro del rectángulo envolvente del *blob*. Si dicho porcentaje es menor que un cierto umbral, el *blob* es clasificado como persona. Si está por encima de dicho umbral, el *blob* es clasificado como objeto. Dicho umbral p se ha determinado empíricamente y su valor es en torno al 70% y el 75%.

Estos dos algoritmos se combinan según un esquema convencional de Bayes (ver sección 5.4.6).

La figura 3.8 muestra un ejemplo de este esquema detector de personas:

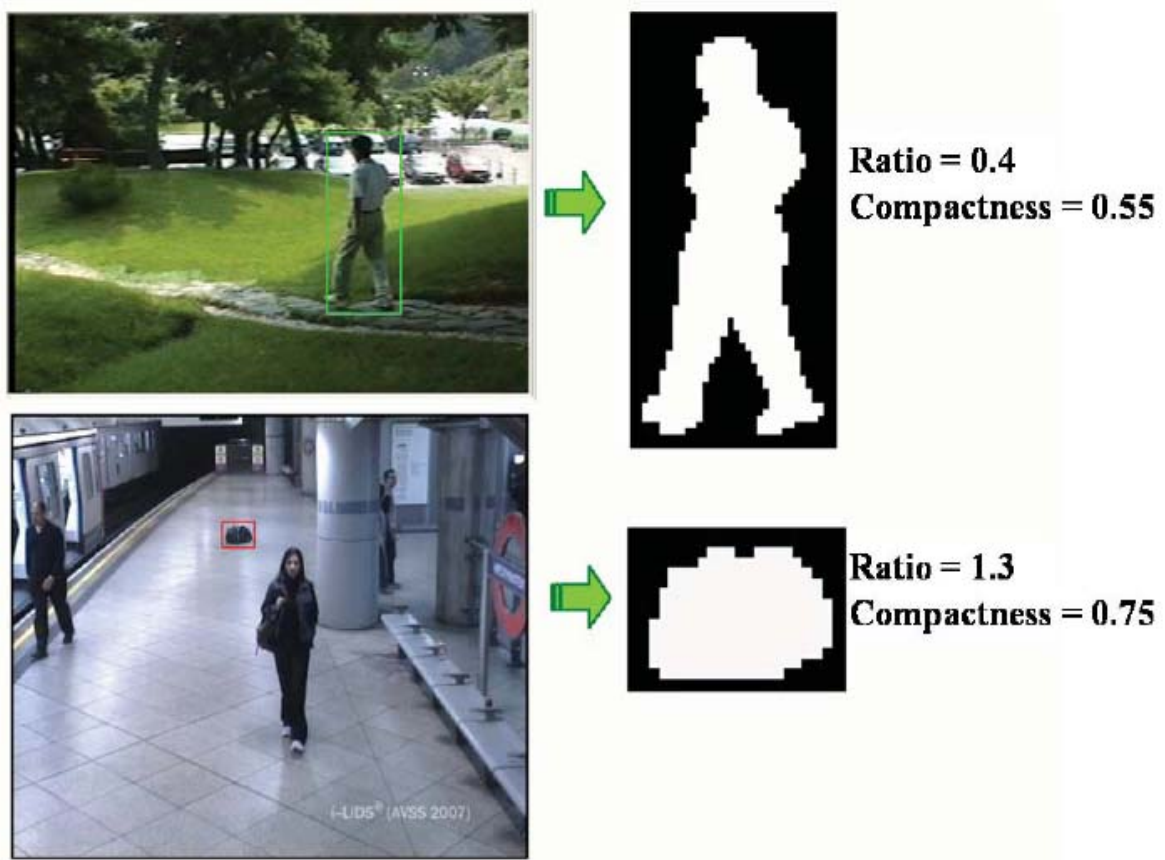


Figura 3-8 Ejemplos de clasificación persona/objeto.

En la figura 3.8 se aprecian dos imágenes, de la imagen superior se ha seleccionado un *blob* que corresponde a una persona, y de la imagen inferior se ha seleccionado un *blob* correspondiente a un objeto. Se observa como los *blobs* pertenecientes a personas suelen tener un ratio de aspecto menor que un objeto y un porcentaje menor de activación de píxeles dentro del rectángulo envolvente.

3.5 Detección de objetos abandonados/robados

Finalmente, la última etapa de análisis es la detección de objetos abandonados o robados. Esta etapa se basa en extraer un patrón de la máscara de *foreground* correspondiente al objeto estático, corregirlo mediante contornos activos, posteriormente se intenta extraer el mismo patrón en la imagen actual y el fondo de la escena y finalmente se calculan distancias entre patrones (máscara *foreground*-imagen actual y máscara *foreground*-fondo escena) para determinar si el objeto estaba en la imagen actual o en el fondo de la escena.

Contornos activos (o *snakes*) es una técnica que se usa para ajustar el contorno inicialmente definido en base a un mapa de probabilidad calculado. Esta técnica se suele utilizar en segmentación de objetos y seguimiento basado en contornos.

Un contorno activo se puede definir como una curva minimizadora de energía, guiada por fuerzas restrictivas externas e influenciada por fuerzas de la imagen, que tiende a localizarse en características de ésta como líneas y bordes.

A continuación vemos en la figura 3.9 la evolución de un *snake* hasta dar con el contorno de un hueso buscado en una radiografía.

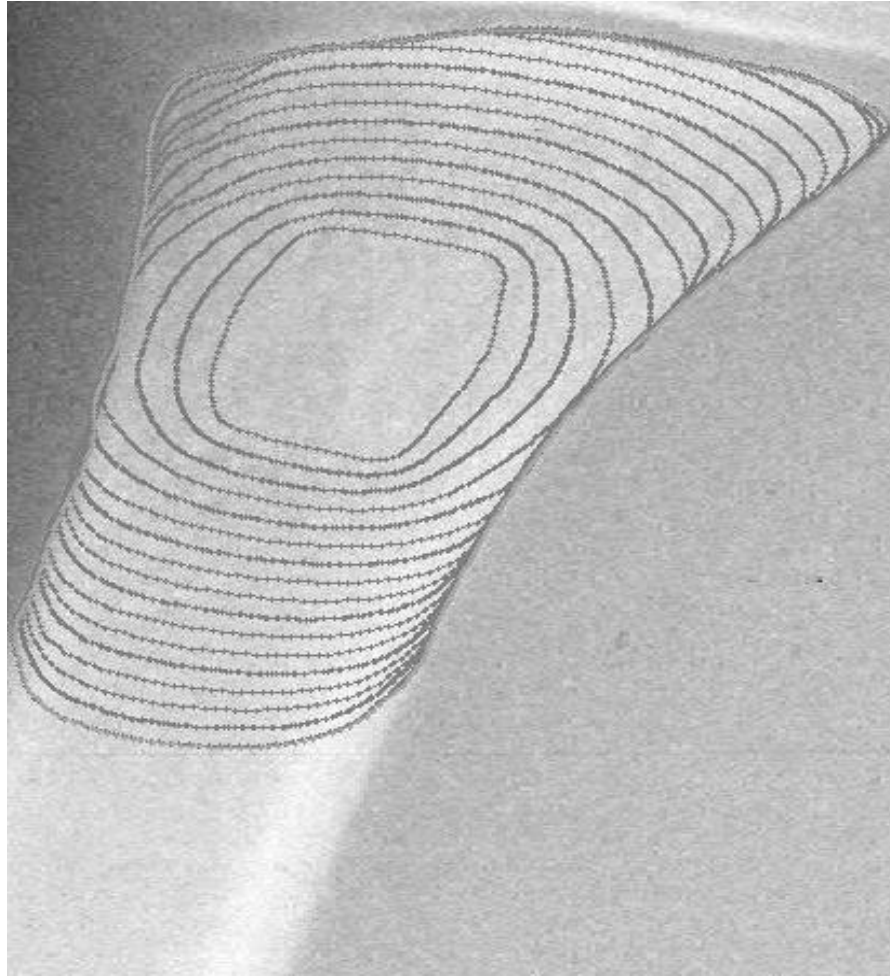


Figura 3-9 Ejemplo de contornos activos (snakes)

El contorno activo tiene dos tipos de energía: Interna y Externa. La Energía Interna se define por la forma del contorno activo y está compuesta de Energía Continua y Energía de Curvatura. La Energía Continua define cómo están de separados unos de otros los puntos del contorno. La Energía de Curvatura define cuánto de liso es el contorno. La Energía Externa es la energía de la imagen. Esta se define a través de los datos de la imagen de la cual el contorno es seleccionado. Como usamos la imagen del gradiente como la energía de imagen, los datos son atraídos a los puntos, líneas o curvas cuyo valor del gradiente es bastante alto. Esta atracción a los altos valores del gradiente explica porqué los contornos activos son tan convenientes para el ajuste del contorno de objeto. La energía en cualquier punto del contorno activo se define como muestra la siguiente ecuación:

$$E = \alpha E_{continuity} + \beta E_{curvature} + \lambda E_{image} \quad (3.4)$$

donde α , β y λ son los parámetros que se usan para controlar los contornos activos.

El ajuste de contornos de objetos mediante *snakes* ha sido probado con diferentes valores, y basándonos en los resultados obtenidos, se ha decidido usar los siguientes valores: $\alpha = 1.0$, $\beta = 1.0$ y $\lambda = 2.0$. Se ha dado un alto valor a λ para buscar una rápida convergencia a los puntos con un gradiente alto (bordes).

En el sistema base, contornos activos es aplicado para buscar el contorno detectado inicialmente (en la máscara de *foreground*) en la imagen actual o en la imagen de fondo. Después se aplican los correspondientes detectores de robo/abandono. En la figura 3.10 se muestran dos ejemplos del ajuste de la forma del objeto. Estos ejemplos muestran como la forma del objeto se ajusta en caso de un objeto abandonado (a), y se deforma cuando nos encontramos en un caso de objeto robado (b). Ambos ajustes mostrados se han realizado extrayendo el contorno de la máscara de *foreground* y ajustándola en la imagen actual.

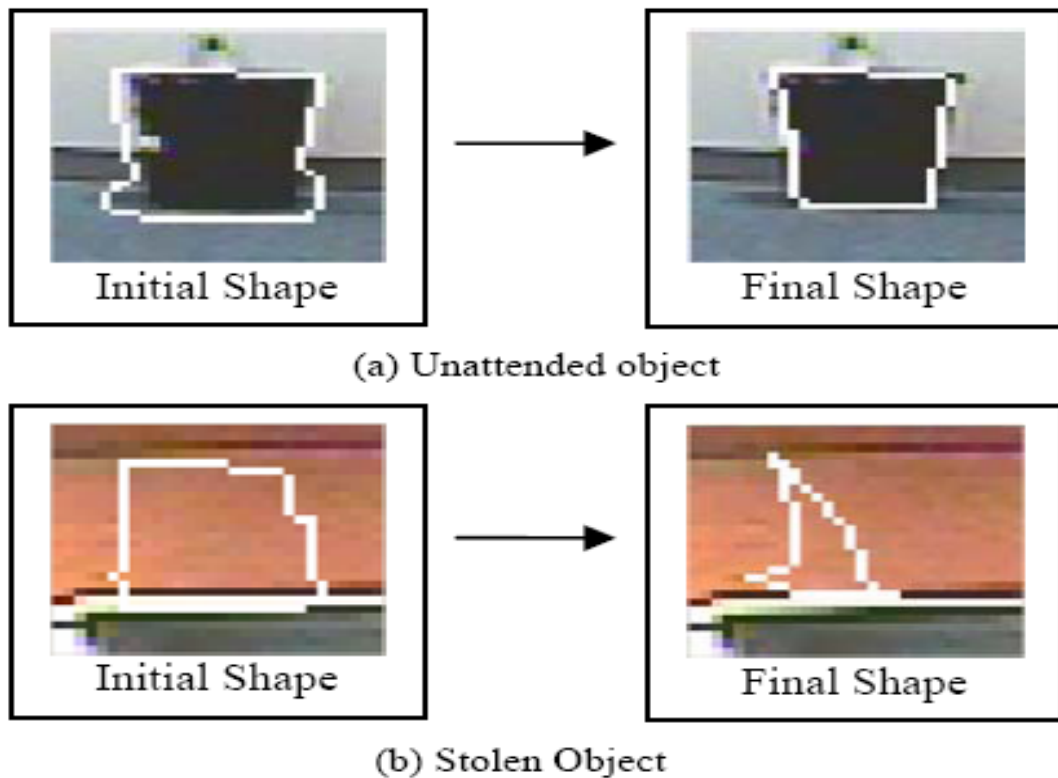


Figura 3-10 Ejemplo del ajuste del contorno para un objeto abandonado y otro robado

El patrón extraído mediante contornos activos se compara con la imagen actual y la imagen de fondo, y se estudian sus similitudes y diferencias. Para ello se utilizan una serie de detectores basados en color y el valor del contorno (o gradiente).

Cada detector genera una puntuación de robo y una puntuación de abandono proporcionada por el análisis realizado. Para generar una probabilidad (o evidencia) dada una puntuación x , se asume que las puntuaciones obtenidas tanto para robo como para abandono siguen una distribución Gaussiana con media μ y una desviación típica σ . Ambos parámetros se determinan experimentalmente para cada puntuación existente y para cada clase existente (robo o abandono) mediante el uso de un conjunto de datos de entrenamiento. Así para cada clase, la evidencia o probabilidad de un detector dada una puntuación obtenida es definida como un valor entre 0 y 1. El valor se calcula con la siguiente ecuación:

$$E_{\mu,\sigma}(x) = e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad (3.5)$$

donde μ y σ son la media y la desviación típica de la clase correspondiente (robo o abandono). Posteriormente, las evidencias obtenidas por cada uno de los detectores se combinan y se obtiene una puntuación (evidencia final), una para el caso de robo (E_S), y otra para el caso de abandono (E_U).

Finalmente se considera que un objeto es abandonado si $E_U > E_S$, o robado si $E_U < E_S$.

4 Detección de regiones estáticas

En este capítulo se describe una de las partes más crítica de la detección de robo o abandono de objetos, que es en la detección de regiones estáticas, debido a que los objetos robados o abandonados son regiones estáticas.

Para realizar una detección fiable de regiones estacionarias, es necesario saber las características del vídeo a analizar tales como imágenes por segundo (*framerate*) o distorsión espacial introducida por la cámara.

Este capítulo está dividido en 2 apartados, el primero describirá los métodos implementados para detectar regiones estáticas. Se han seleccionado estos métodos del estado del arte debido a su bajo coste computacional y los buenos resultados que cosechan. En el segundo apartado se propone un nuevo método destinado a la detección de regiones estáticas, que demuestra ser bastante robusto y fiable en entornos con alta densidad de personas, y, por consiguiente, también funciona a la perfección en secuencias sencillas.

4.1 Métodos de detección de regiones estáticas seleccionados

En este apartado se describen con detalle los métodos seleccionados del estado del arte para la tarea de detección de regiones estáticas. Posteriormente, se han implementado dichos métodos con la misma técnica de sustracción de fondo para evaluar el efecto de cada aproximación. Finalmente, se ha propuesto un nuevo método basado en el análisis de las ventajas/desventajas de cada método.

4.1.1 Método basado en una aproximación clásica

Este método es el mismo que el que se ha descrito en el sistema base, (apartado 3.4 de esta memoria). Este algoritmo ha sido proporcionado por el grupo de investigación VPU-Lab, y se basa en analizar los datos proporcionados por la etapa de seguimiento de *blobs* (*tracking*) que son extraídos de la imagen actual. Para ello se intenta buscar *blobs* semejantes de la imagen anterior, y aquellos *blobs* cuya velocidad sea menor que un umbral (cercano a 0) durante 50 imágenes consecutivas, se considerarán regiones estáticas.

4.1.2 Método basado en la acumulación de máscaras de *foreground*

Este método se basa en [14], y busca detectar regiones estáticas mediante la acumulación de máscaras de *foreground* obtenidas en instantes consecutivos. Para ello, el algoritmo

produce una imagen intermedia $S(x,y)$, donde el valor de cada píxel (entre 0 y 255) determina si dicho píxel es estacionario, o no es estacionario. Un valor de 0 indica que el píxel pertenece al background de la imagen, y un valor de 255 indicará 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 inicializan a 0. A continuación, se utiliza $I(x,y)$ para representar el píxel en la posición (x,y) de la imagen actual y $B(x,y)$ el píxel correspondiente en la imagen de fondo. Para cada nueva imagen, cada píxel de la imagen S se actualizará basándose en una comparación entre el píxel de I y de B .

Dos contadores intermedios se utilizan para realizar dicha actualización. Un contador positivo, $C(x,y)$, y un contador negativo $D(x,y)$. C indica, píxel a píxel, durante cuántas imágenes consecutivas el píxel (x,y) pertenece al *foreground*. Este contador se resetea a 0 en el momento que dicho píxel pertenezca de nuevo al background de la imagen. El contador negativo D indica durante cuantas imágenes seguidas el píxel $I(x,y)$ pertenece al background en vez de al *foreground*. Ambos contadores se inician con el valor 0. Las ecuaciones de actualización son las siguientes:

$$\begin{aligned} S(x, y) &= S(x, y) + C(x, y)(255/t * FrameRate) \\ S(x, y) &= S(x, y) - rD(x, y)(255/t * FrameRate) \end{aligned} \quad (4.1)$$

donde t es el tiempo necesario para declarar un objeto estacionario, en nuestro caso 30 segundos, r es un número positivo con el que aumentar el efecto de D para que, cuando un posible píxel candidato a ser estacionario deja de pertenecer al *foreground* y pasa a pertenecer al *foreground*, el valor de S en dicho píxel se decremente rápidamente. Destacar que dicha imagen intermedia S solo toma valores entre 0 y 255, y que las oclusiones no afectan en absoluto a este método.

Los píxeles que pertenecen a un objeto estacionario harán que el valor de S aumente imagen a imagen hasta llegar al máximo valor posible de 255. El tiempo que tarda en llegar depende del valor asignado al parámetro t .

Cabe destacar que la imagen obtenida, S , es una imagen en tonalidad de grises, por lo tanto, el último paso de dicho algoritmo consiste en realizar una umbralización de S respecto a un valor predeterminado. Estrictamente, el valor 255 correspondería con el tiempo t definido para detectar una región estática pero tras varias pruebas, se ha decidido que ese umbral tome el valor de 240. Así, finalmente, obtenemos una imagen binaria donde los píxeles a '1' determinan posibles regiones estáticas.

A continuación se muestra un ejemplo de este proceso en la figura 4.1:

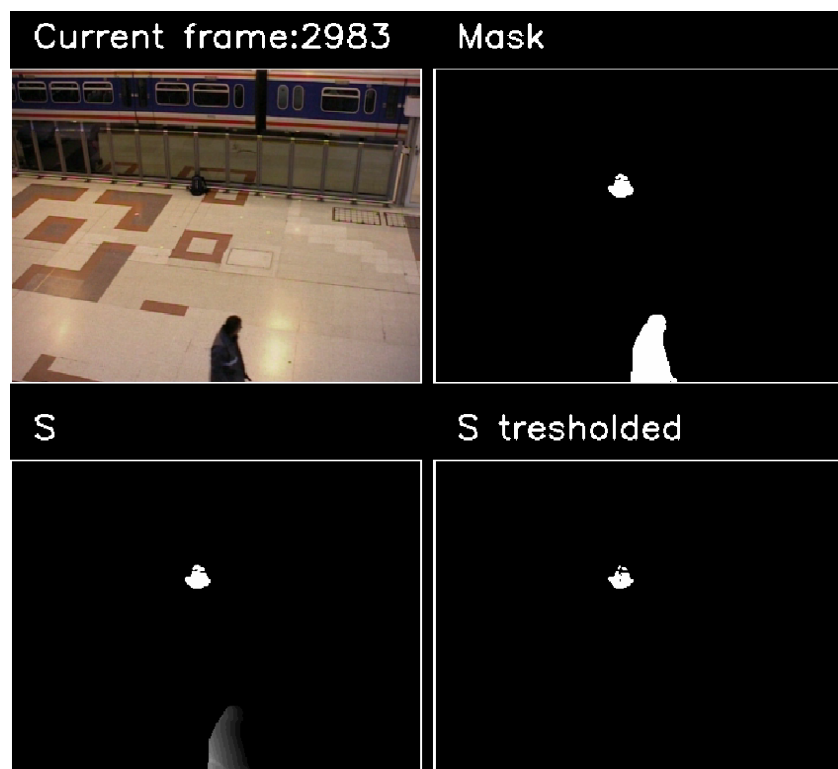


Figura 4-1 Ejemplo del método basado en la acumulación de máscaras de FG

En la figura 4.1 se aprecia, un objeto estacionario, y una persona en movimiento. Se muestra la máscara binaria y la imagen intermedia S, en escala de grises. La última imagen muestra ese proceso de umbralización de S, quedándose solo las regiones estacionarias.

4.1.3 Método basado en las propiedades del modelo de mezcla de Gaussianas

Este método se basa en el estudio de las propiedades del modelo de segmentación fondo-frente utilizado. Para este método, basado en [12], en vez de utilizar el segmentador gamma, se va a utilizar el modelo de mezcla de Gaussianas (MoG) para separar el frente del fondo de una imagen.

El modelo MoG se basa en usar K Gaussianas por píxel, que se modelan a través de una media (μ), una desviación típica (σ) y un factor de peso (ω). La media indica el valor mas probable de dicho píxel para cada píxel, la desviación típica lo que se puede alejar dicho

valor medio por cada uno de los lados, y el peso indica que Gaussiana es la de mayor importancia. Destacar que la suma de los pesos de una Gaussiana para cada píxel es igual a 1.

En el trabajo descrito en [12], el modelo usa una mezcla de 3 Gaussianas por píxel para determinar si un píxel pertenece al background o al *foreground*.

Este método se basa en observar los cambios de estados desde que una región aparece en la escena hasta cuando entra a formar parte del fondo (Gaussiana que modela el fondo de la imagen). El diagrama de estados con sus correspondientes transiciones se muestra en la figura 4.2 para un píxel determinado.

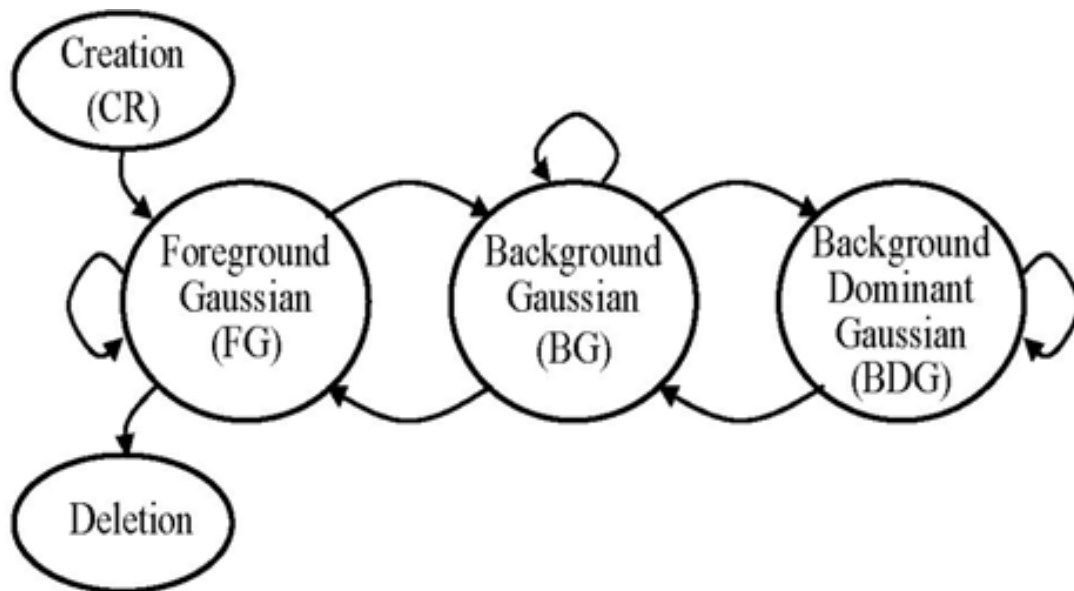


Figura 4-2 Diagrama de transiciones para el modelo MoG

A continuación se describe el proceso realizado por el método para cada píxel de una región detectada como *foreground*.

Cuando una nueva región entra en la escena, la Gaussiana de los píxeles pertenecientes a la región que poseen el menor peso se elimina y se cambia por otra Gaussiana centrada en el valor de dicho píxel. Si dicho región no es una región estacionaria, la nueva Gaussiana creada permanecerá en el estado de Gaussiana de *Foreground* (FG). En cambio, si la región resulta ser estacionaria, y permanece durante un largo periodo de tiempo, la Gaussiana que pertenecía al estado de *Foreground* avanzará hasta el estado de Gaussiana

de fondo (BG). Y si permanece durante más tiempo, dicha Gaussiana pasará al estado de fondo dominante (BDG).

Esto implica que, observando el cambio de estado de la Gaussiana de cada píxel perteneciente a la nueva región, específicamente cuando una Gaussiana pasa del estado de FG al estado de BG y, por último, al estado de BGD, nos encontraremos ante un píxel que puede formar parte de un objeto estacionario.

Sin embargo, este método no es muy efectivo, ya que dicha condición no es suficiente para clasificar un objeto robado debido a que los cambios en el estado de BDG pueden ser causados por regiones inestables como sombras o cambios de iluminación.

Para la implementación de dicho algoritmo, se ha tenido en cuenta las transiciones de las Gaussianas. Para cada imagen, se comprueba cual es la Gaussiana activa para todos los píxeles. Si la Gaussiana activa cambia y pasa de ser la Gaussiana perteneciente al BGD a ser la Gaussiana de FG, se estudia el comportamiento de dicho píxel para futuras imágenes. Cuando este píxel pasa de FG a BG, y luego de BG a BGD, se pinta como 1 en una máscara binaria creada para detectar las regiones estacionarias. Esto hace que en cada iteración haya que hacer cálculos y comparaciones píxel a píxel para toda la imagen, lo que hace que el análisis tenga un coste computacional mayor que los otros métodos, lo cual no interesa cuando nos encontramos ante aplicaciones en tiempo real como es el caso de vídeo-seguridad.

Destacar que este método no tolera oclusiones, ya que al producirse una oclusión delante de un objeto abandonado, se crearía una nueva Gaussiana de FG para dicho objeto, perdiendo el seguimiento realizado anteriormente.

A continuación, se muestra en la figura 4.3 un ejemplo sobre el funcionamiento de dicho método, donde se puede ver todo lo explicado anteriormente.

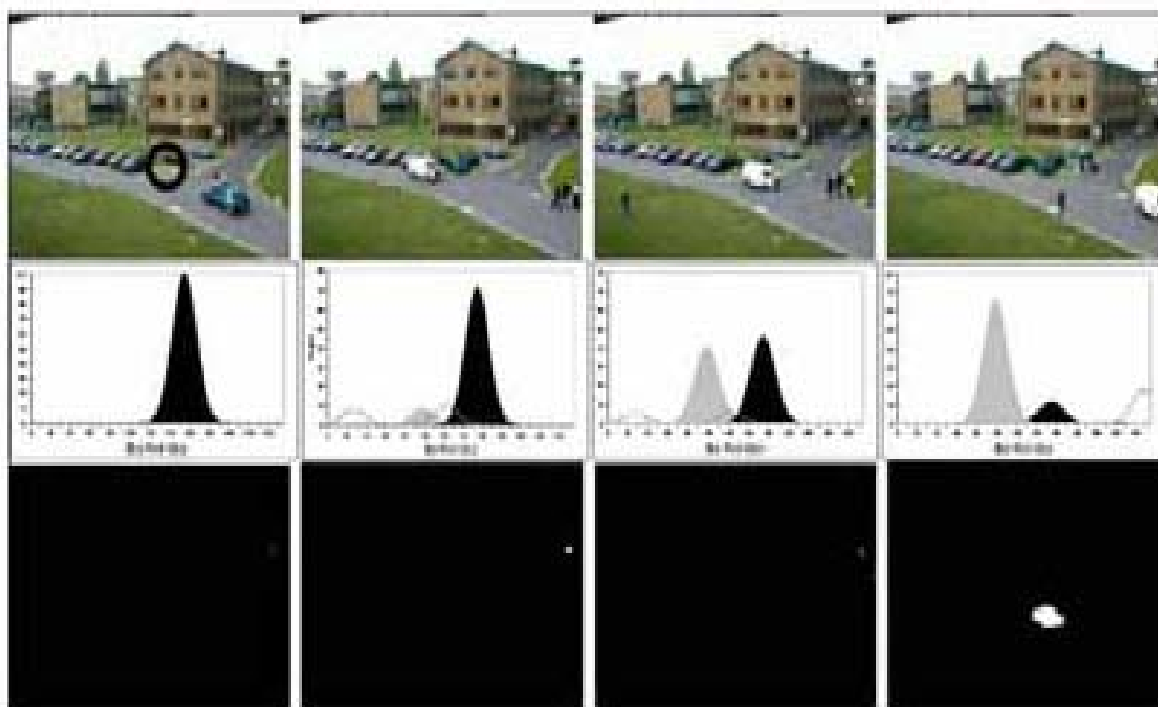


Figura 4-3 Ejemplo del método basado en las propiedades del modelo de fondo

En la figura 4.3, en la primera fila podemos observar las imágenes del vídeo, el coche aparcado y señalado con un círculo negro acaba de entrar en la escena. En la segunda fila, se ve la distribución de Gaussianas de un píxel perteneciente al vehículo. Al principio, hay una Gaussiana dominante (la de fondo negro), y se ve como se crea una Gaussiana de color gris para modelar dicho cambio. Se observa que al avanzar en el tiempo, dicha Gaussiana que pertenecía al FG pasa a pertenecer al BDG, ya que es la Gaussiana de mayor peso. En la tercera fila se observan los píxeles correspondientes a regiones estáticas en la escena. Destacar que el coche sólo se marca al final de la secuencia, cuando ya se ha producido el cambio de Gaussiana dominante.

4.1.4 Método basado en el submuestreo de máscaras binarias.

Este método se basa en el artículo [8]. Emplea una nueva técnica basada en el submuestreo de máscaras binarias de *foreground* para localizar las regiones estáticas dentro de las mismas.

Considerando F_i cada una de las imágenes a procesar, B la imagen de fondo y σ la desviación típica de la imagen, consideraremos que el píxel correspondiente a la posición (x,y) se considerará que pertenece al fondo si :

$$|F_k(x, y) - B_k(x, y)| > w(x, y) * \sigma(x, y) \quad (4.2)$$

donde $w(x,y)$ es una función que mide los cambios de iluminación de cada uno de los píxeles de la imagen, que puede ser causada por un ángulo incorrecto de la cámara, o, simplemente, por una mala configuración de la cámara.

El número de máscaras binarias muestreadas para este caso, como se indica en [8] ha sido el de 6.

Tras esto, obtenemos 6 máscaras binarias de *foreground* $\{M_1, \dots, M_6\}$, cuyos píxeles con valor 1 indican que pertenecen al *foreground*, y los píxeles con valor 0 indican que pertenecen al *background*. Después, se realiza una multiplicación lógica píxel a píxel de cada una de las 6 máscaras binarias de *foreground* obtenidas, para obtener una imagen final de objetos estacionarios S, que se obtiene del siguiente modo:

$$S = M_1 * M_2 * M_3 * M_4 * M_5 * M_6 \quad (4.3)$$

Después, se aplica una etapa posterior de filtrado a la máscara S para eliminar posible ruido impulsivo incluido en dicha máscara.

Tras esta operación de reducción de ruido, se puede afirmar que un valor para un objeto en dicha máscara de '1' significará que esa región ha permanecido estática durante los 30 últimos segundos, pudiendo considerarse ya como objeto sospechoso, y pasaría a las siguientes etapas de análisis.

A continuación se muestra un esquema general de este método en la figura 4.4:

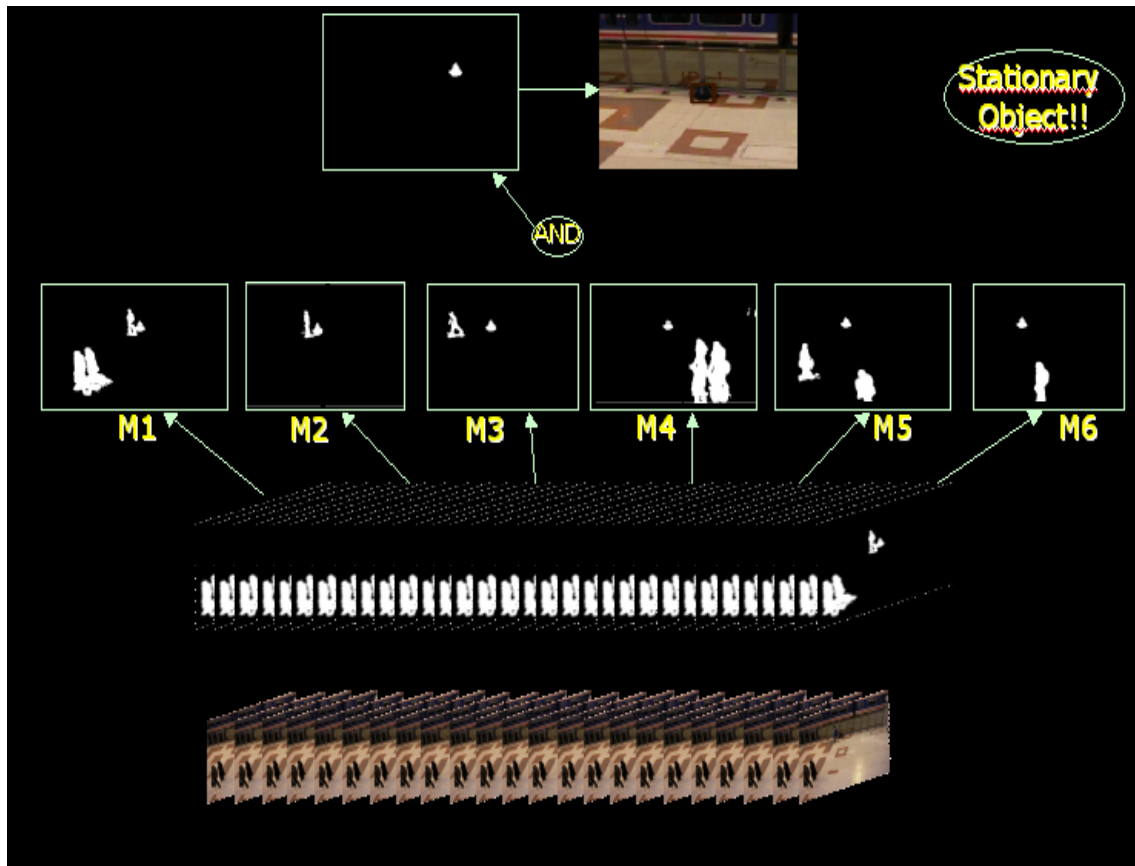


Figura 4-4 Esquema del método basado en el submuestreo de máscaras de FG

En la figura 4.4 se ve la secuencia de imágenes a analizar, sus máscaras binarias, y como se submuestran 6 de ellas para después calcular la máscara final, y detectar la región estacionaria (en este caso un objeto).

En la figura 4.5, se muestra otro ejemplo de cómo este método tolera oclusiones y es muy valido para escenas densamente pobladas:

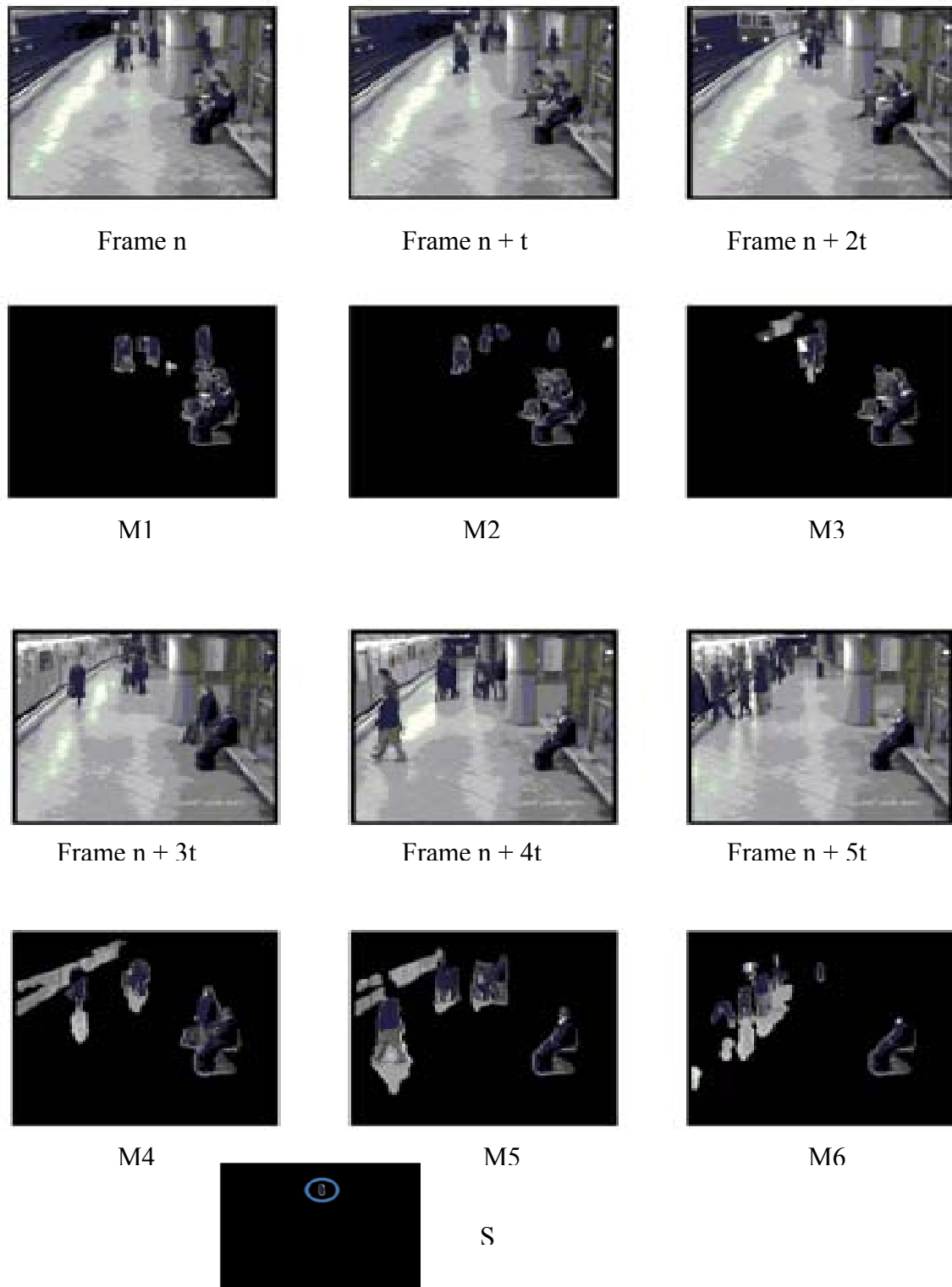


Figura 4-5 Ejemplo del método basado en el submuestreo de máscaras de FG

En la figura 4.5 se observa que la máscara S tolera oclusiones tanto parciales como totales. Destacar que dicha imagen S se ha extraído a través de 6 máscaras binarias, pero fijándose sólo en el objeto de interés de la escena. Cabe resaltar que, si una persona se ubica delante

de un objeto durante bastante tiempo, no tendría sentido decir que hay algo abandonado o robado, debido a que en la imagen no se ve, y puede que dicho objeto ya no se encuentre en la misma posición.

4.1.5 Método basado en diferentes modelos de fondo actualizados a diferentes velocidades

En este método, basado en el estudio realizado en [7], se propone el uso de varios modelos de segmentación de fondo, siendo esta la principal diferencia con respecto a los anteriores. En este método se utilizarán dos modelos diferentes, y la novedad reside en que se actualizan a diferentes velocidades.

Para detectar regiones estáticas, lo normal es modelar la imagen de fondo con un solo modelo matemático, pero este método decide utilizar dos modelos matemáticos, un modelo a largo plazo (B1), que se actualizará con unos parámetros diferentes, de manera más lenta, y otro modelo a corto plazo (B2), que se actualizará a mayor velocidad. .

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, B1 y B2. Al realizar esa comparación, se obtienen dos máscaras binarias $F1$ y $F2$, 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 $F1$ y de $F2$, pueden ocurrir 4 hipótesis, que se detallan a continuación:

- $F1 = 1$ y $F2 = 1$, 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 fondos propuestos.
- $F1 = 1$ y $F2 = 0$, se considera que el píxel (x,y) corresponde a una región estacionaria.
- $F1 = 0$ y $F2 = 1$, se considera que el píxel (x,y) es un píxel del fondo de la escena que ha sido ocluido anteriormente.
- $F1 = 0$ y $F2 = 0$, se considera que el píxel (x,y) pertenece al fondo de la escena, debido a que se asemeja con ambos modelos de fondo.

Esta clasificación se puede ver en la figura 4.6

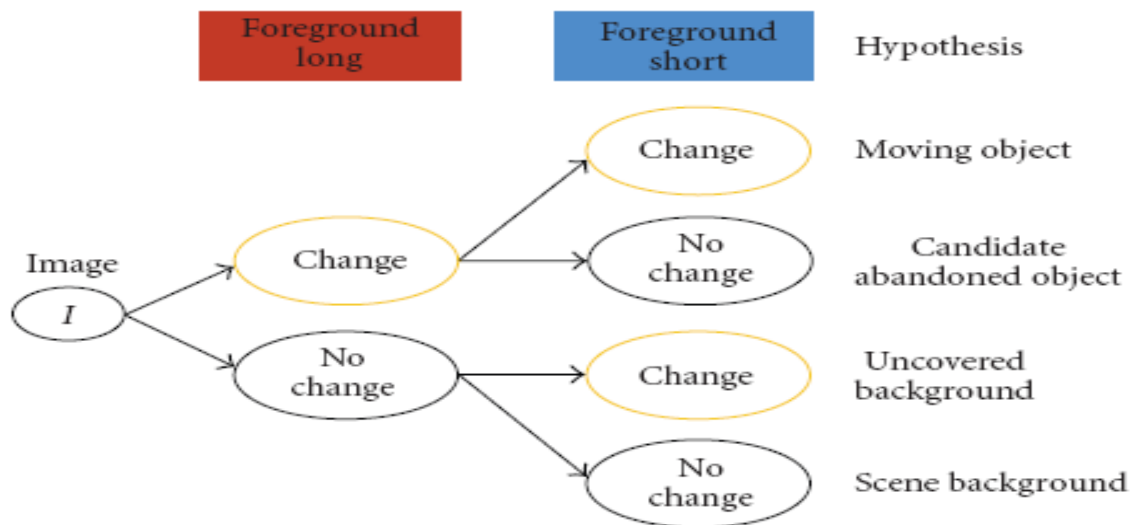


Figura 4-6 Clasificación para el método que usa dos modelos de fondo

Cabe destacar que el modelo de fondo a corto plazo se adapta más rápido a los cambios existentes en la escena. En contra, el modelo a largo plazo tarda bastante más en adaptarse a dichos cambios.

A continuación se muestra un ejemplo de este método en la figura 4.7:

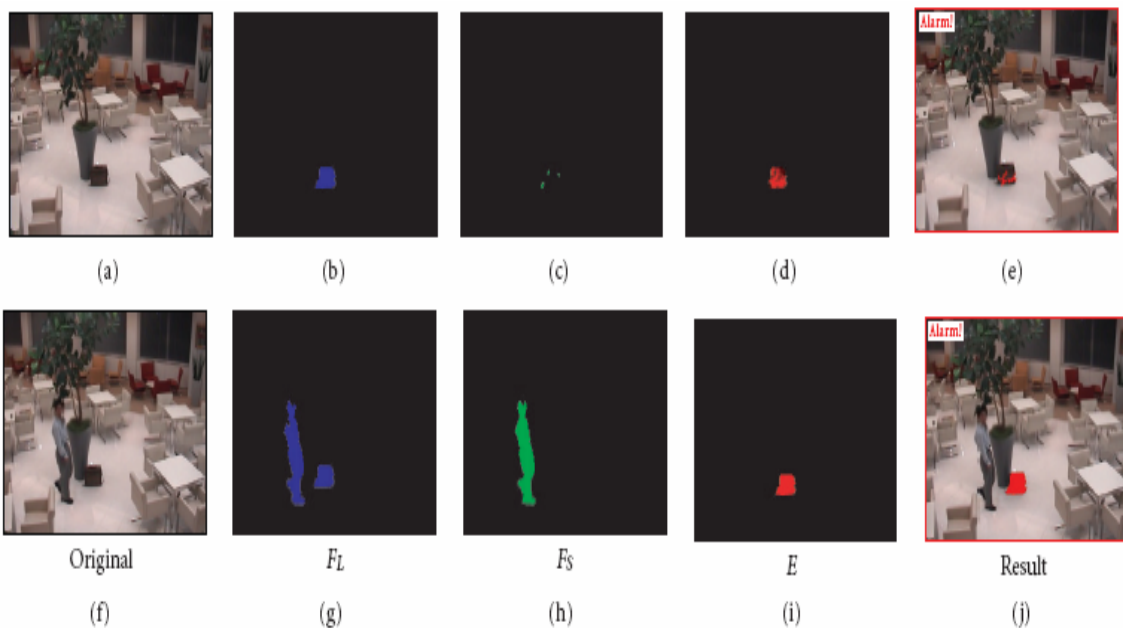


Figura 4-7 Ejemplo del método basado en más de un modelo de fondo

La primera fila pertenece a la imagen 350, y la segunda a la imagen 630. La máscara de *foreground* a largo plazo (F1) se ve como recoge los objetos en movimiento, y las regiones estáticas. La máscara de *foreground* a corto plazo F2 sólo captura los objetos en movimiento. La máscara final, denotada como E, se comporta de modo más estable a medida que el objeto lleva más tiempo en la escena.

Este método ha demostrado ser robusto y fiable para detectar regiones estáticas. No requiere información previa a la hora de su ejecución, y muestra la máscara de las regiones estáticas aun cuando ellas están ocluidas.

4.2 Algoritmo de detección de regiones estáticas propuesto

En este apartado se propone un algoritmo robusto, fiable y sencillo capaz de detectar las regiones estáticas dentro de secuencias de vídeo-seguridad complejas, como aeropuertos o estaciones de metro. Esta sección se divide en tres sub-apartados, el primero donde se describe el algoritmo, el segundo donde se detalla su implementación, y el tercer apartado donde se describe una posible solución al problema de las oclusiones entre regiones.

4.2.1 Descripción

Para la realización de este algoritmo, se realizó un estudio de las técnicas más recientes pertenecientes al estado del arte (sección 2.1), así como a las características de los algoritmos implementados en la sección 4.1.

Del estudio se concluyó que un algoritmo robusto a la hora de detectar regiones estáticas ha de ser capaz de solventar los problemas derivados del ruido introducido por las cámaras, las sombras y los reflejos de las personas u objetos de la imagen, posibles cambios de iluminación (como una puesta de sol), tiene que tener un modelo de fondo que se actualice de modo consistente y coherente, dicho modelo de fondo tiene que ser parametrizable para poder obtener buenos resultados en secuencias de diferentes características, tiene que soportar oclusiones parciales y totales, y lo mas importante, tiene que ser capaz de extraer las regiones estáticas de la escena del modo más preciso y fiable en entornos con alta densidad de personas (donde el número de oclusiones es bastante alto).

Para la detección de dichas regiones estáticas, se propone analizar características de la señal de vídeo en diferentes instantes de tiempo (sub-muestreo). Primeramente se analiza

la persistencia de una región de *foreground* en distintos instantes temporales. Posteriormente, se analiza el movimiento de dichas regiones en los instantes seleccionados en el muestreo anterior. El objetivo de este análisis es la eliminación de regiones de *foreground* siempre activas donde claramente se observa que las regiones son distintas (e.g., zonas de paso donde siempre existe *foreground* activo pero a su vez esta en movimiento). El resultado de ambos análisis son dos máscaras binarias indicando las propiedades deseadas.

Para analizar la persistencia de las regiones de *foreground* se utiliza el método descrito en la sección 4.1.4 (sub-muestro de máscaras binarias de *foreground*). El resultado del sub-muestreo de máscaras de *foreground* y su posterior combinación lógica (utilizando una operación *AND*) nos permite obtener una máscara binaria *S* formada por regiones estáticas a partir de un único modelo de fondo. Para analizar el movimiento de las regiones estáticas se utiliza la técnica de “*frame difference* (FD)”, que consiste en restar dos imágenes entre sí, dando como resultado un estimador de movimiento dentro de dicha imagen, siendo FD_t la imagen resultante del *frame difference* en el instante *t*, I_t la imagen en el instante *t*, y I_{t-1} la imagen resultante 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 obtener una máscara binaria (ver ecuación 4.4)

$$FD_t = I_t - I_{t-1}$$

$$FD_t(x, y) = 1 \text{ si } FD_t(x, y) > \tau \quad (4.4)$$

$$FD_t(x, y) = 0 \text{ si } FD_t(x, y) < \tau$$

Tras diversas pruebas, se ha fijado experimentalmente el valor del umbral τ en 20.

Esta máscara final nos devuelve en blanco (valor 1) las regiones en movimiento, y en negro (valor 0) las regiones que no han sufrido movimiento. Por lo tanto, como lo que queremos es destacar las regiones estáticas, realizaremos una inversión de la imagen final obtenida de la siguiente manera:

$$FDI_i(x, y) = not(FD_i(x, y)) \quad (4.5)$$

Para realizar esta resta, no se realiza la resta entre *frames* consecutivos, sino que se aplica el mismo intervalo de muestreo que el utilizado para extraer la otra máscara binaria (por ejemplo, 170 *frames*), para ver realmente el recorrido de cambio de los objetos o personas en movimiento dentro de la escena. Se realizará el mismo número de etapas para obtener esta máscara que para obtener la máscara anterior.

Con esta máscara podremos detectar regiones de movimiento dentro de la escena, por lo tanto supondremos que, en dichas regiones de movimiento no puede haber un objeto estacionario.

Finalmente, cada máscara obtenida se computa lógicamente mediante una multiplicación binaria para dar lugar a la máscara final que se utiliza para detectar regiones estáticas. La siguiente figura muestra el esquema utilizado:

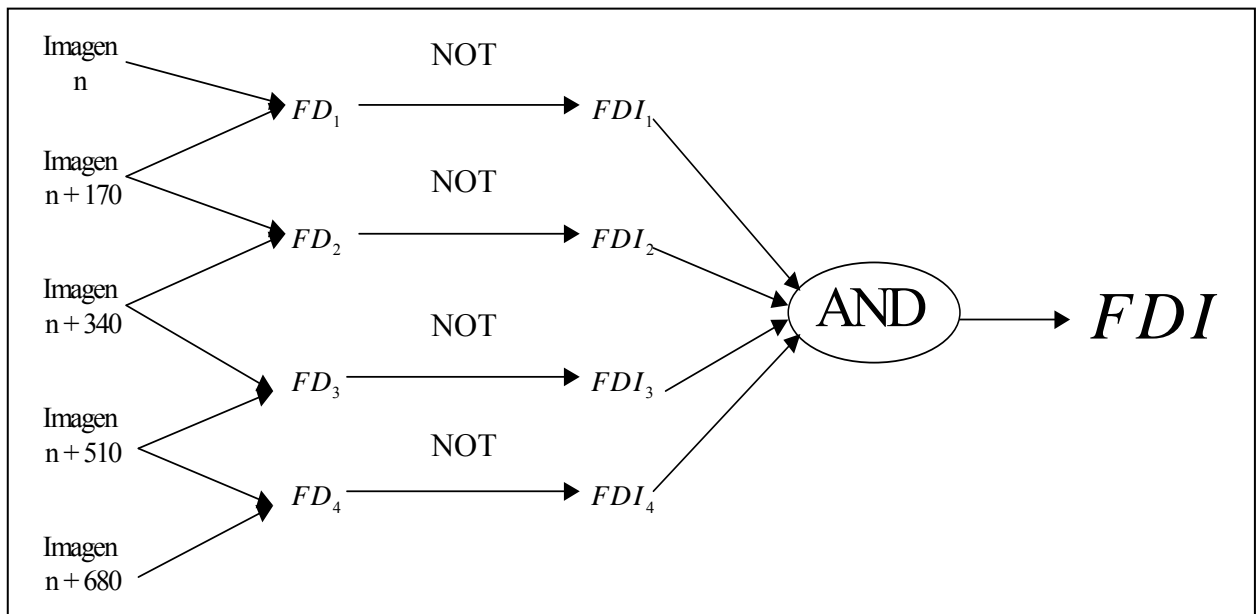


Figura 4-8 Esquema seguido para obtener la máscara FDI

En la imagen se observa como se obtienen las máscaras pertenecientes al “*frame difference*”, para luego invertirlas y obtener la máscara final FDI.

Ahora, el último paso consiste en realizar la multiplicación píxel a píxel de la máscara S correspondiente a detectar regiones estáticas basándose en el submuestreo de máscaras

binarias, por la máscara FDI obtenida al realizar el método de “*frame difference* inverso”. Así obtendremos la máscara final de regiones estáticas FM. La ecuación es la siguiente:

$$FM(x, y) = FDI(x, y) * S(x, y) \quad (4.6)$$

4.2.2 Análisis del algoritmo

A la hora de realizar la implementación, se evaluaron varias secuencias de *datasets* conocidos (ver apartado 2.4 de este PFC) que poseían diferentes características. Se llegó a la conclusión de que en escenarios sencillos, los resultados eran bastante fiables, pero en escenarios complejos, los resultados variaban bastante a la hora de ajustar los parámetros de prueba.

Se seleccionaron 3 parámetros críticos a la hora de la evaluación de dicho sistema. Estos parámetros son:

- Frecuencia de muestreo (medida en imágenes).
- Número de etapas de muestreo.
- Parámetros críticos del segmentador gamma.

Cabe destacar, que del segmentador gamma, se seleccionó la variable “*varnoise*”. Un valor bajo de dicha variable hacía que el segmentador separara con mucho detalle el frente del fondo de la imagen, a costa de introducir mucho ruido aleatorio en la máscara de *foreground*. En cambio, si utilizábamos un valor alto, el segmentador eliminaba todo el ruido, pero la calidad de los objetos extraídos era muy baja. Finalmente, tras calibrar y probar dicha variable entre los valores de 8 y 15, en intervalos de 1 en 1, se decidió que el valor óptimo para este tipo de aplicaciones de detección de objetos robados era de 11.

Los otros dos parámetros a ajustar son la frecuencia de muestreo y el número de etapas de muestreo. Se observó que estos parámetros guardaban relación con el tiempo que se considera una región como estacionaria, que a lo largo de este proyecto ha sido de 30 segundos. De media, los resultados han demostrado que este algoritmo trabaja a una velocidad de 25 imágenes por segundo. Haciendo pruebas, se puede pasar, de tener 2

etapas de muestreo cada 15 segundos (por lo tanto se seleccionaría una imagen de cada 375), o bien, 30 etapas de muestreo cada segundo (una imagen de cada 25).

Evidentemente, ninguno de los dos casos es el óptimo, debido a que si se seleccionan pocas etapas de muestreo, el número de falsos positivos perteneciente a regiones estáticas aumenta de modo considerable, ya sea en torno a ruido, o en torno a gente pasando por la imagen, ya que a la que coincidan dos personas diferentes en esas imágenes muestreadas, el resultado final incluiría dicha silueta humana. En cambio, si se seleccionan muchas etapas de muestreo, la probabilidad de que ocurra una oclusión es muy elevada, lo que haría que el objeto nunca se llegara a detectar por el algoritmo (ya que para solventar el problema de las oclusiones, es necesario que se haya detectado la región estática al menos una vez, como veremos en el siguiente sub-apartado).

A su vez, como dijimos con anterioridad, si el sistema trabaja a una velocidad aproximada de 25 *frames* por segundo, y para detectar un objeto estacionario tienen que pasar 30 segundos (o lo que es lo mismo, $25 \cdot 30 = 750$ *frames*), se llegó a la conclusión de que el número óptimo de etapas era de 4, por lo tanto la frecuencia de muestreo necesaria entre imagen e imagen es de 175 *frames* ($750/4$).

La siguiente figura muestra algunos resultados obtenidos por dicho algoritmo

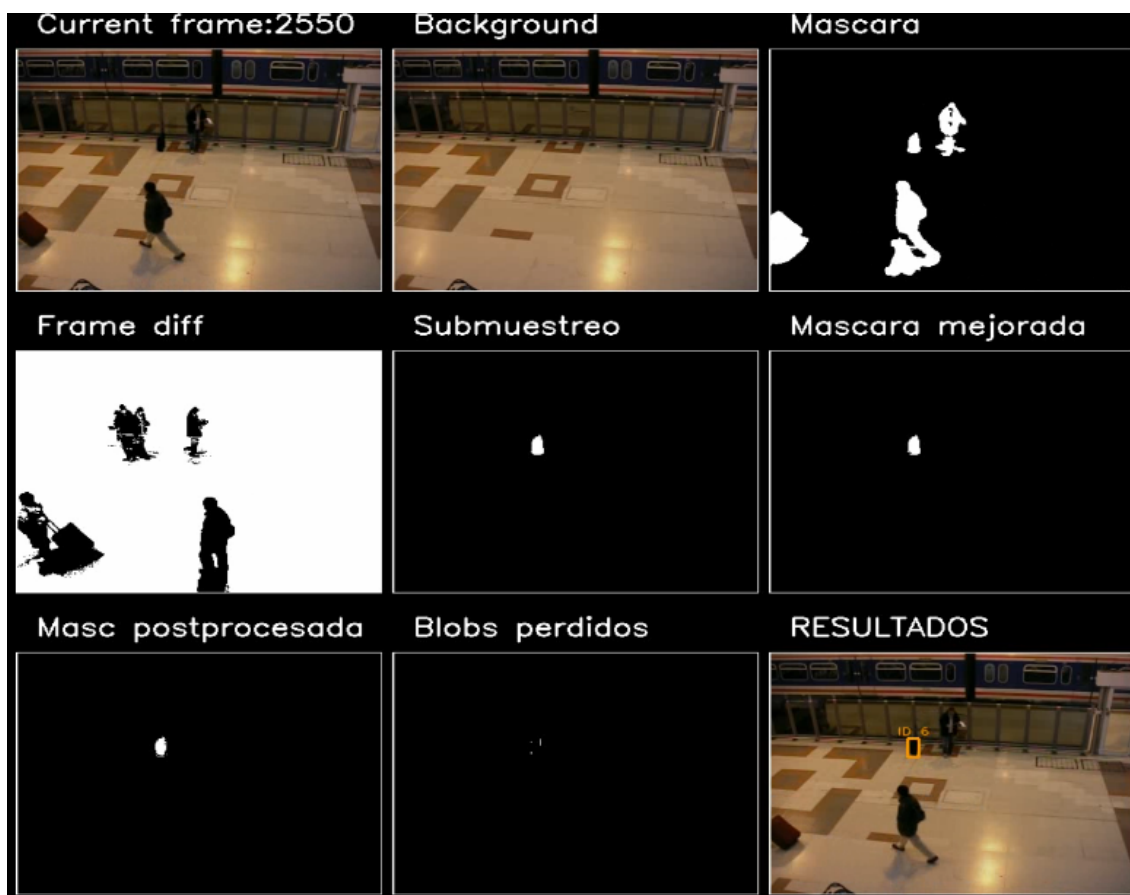


Figura 4-9 Ejemplo del algoritmo implementado

En la figura 4.9 se puede ver como, partiendo de la imagen de original, de un modelo de fondo, y de una máscara binaria para cada *frame*, calculamos la máscara FDI, que nos marca en blanco las regiones que no han sufrido variaciones, y en negro las regiones en movimiento (como son personas en movimiento). La etapa de submuestreo muestra el resultado obtenido por la máscara S, la máscara mejorada (FM) se corresponde con una mejora de la máscara original donde solo se detectan regiones estáticas, el objetivo de dicho algoritmo. Luego dicha máscara se post-procesa para eliminar sombras y ruido, y en la imagen de resultados vemos cómo se señala el objeto abandonado dentro de la escena.

4.2.3 Solución al problema de las oclusiones

Sin embargo, al realizar varias pruebas, se llegó a la conclusión que este método no soportaba oclusiones parciales temporalmente (e.g., una persona que se cruza con el objeto de interés en el instante de muestreo) porque la máscara FDI, la cual estima el movimiento dentro de la escena, marcaba la zona de la oclusión como '0', y al hacer la multiplicación binaria el resultado final daba '0'.

Para solucionar este problema incrementando la robustez del algoritmo en secuencias complejas (donde este tipo de oclusiones es muy frecuente), se ha diseñado otro algoritmo interno que opera con las máscaras S y FDI obtenidas anteriormente.

Para ello, primeramente se guarda una lista con los *blobs* obtenidos en el análisis de regiones estáticas anterior. Posteriormente se compara dicha lista con la lista de *blobs* extraídos en la etapa actual y si falta algún *blob* se estudia si es debido a que el objeto estático ha desaparecido o si se ha producido alguna oclusión.

Para ello, se observó el comportamiento de las máscaras S y FDI, y se obtuvieron las siguientes conclusiones:

- $FDI = 1$ y $S = 1$, imposible que eso ocurra, entonces el *blob* seguiría en la lista de *blobs* actual.
- $FDI = 1$ y $S = 0$, imposible que suceda ya que si la máscara S está a 0, quiere decir que ha habido un cambio, y ese cambió a su vez también se debería de recoger en el valor de FDI.
- $FDI = 0$ y $S = 1$, caso claro de oclusión, ha habido cambio en la imagen de submuestreo, pero ha habido también cambio a corto plazo, que lo marca la máscara de FDI.
- $FDI = 0$ y $S = 0$, la región estática ha desaparecido, y no se marca ninguna de las dos máscaras a 1.

Para determinar si el objeto está ocluido o no, se selecciona el rectángulo envolvente de ambas máscaras, FDI y S, y se calcula el porcentaje de píxeles a 1 del objeto en la máscara de S. Si se ha producido una oclusión, ese porcentaje debe de ser muy alto, por lo tanto, se considerará oclusión si el porcentaje de píxeles supera el 70%. Si eso ocurre, se copia el *blob* almacenado en memoria de l

a etapa anterior, a la etapa actual, y se incluye en la lista de *blobs* actuales.

A continuación se muestran 4 figuras donde se observa el efecto de las oclusiones:

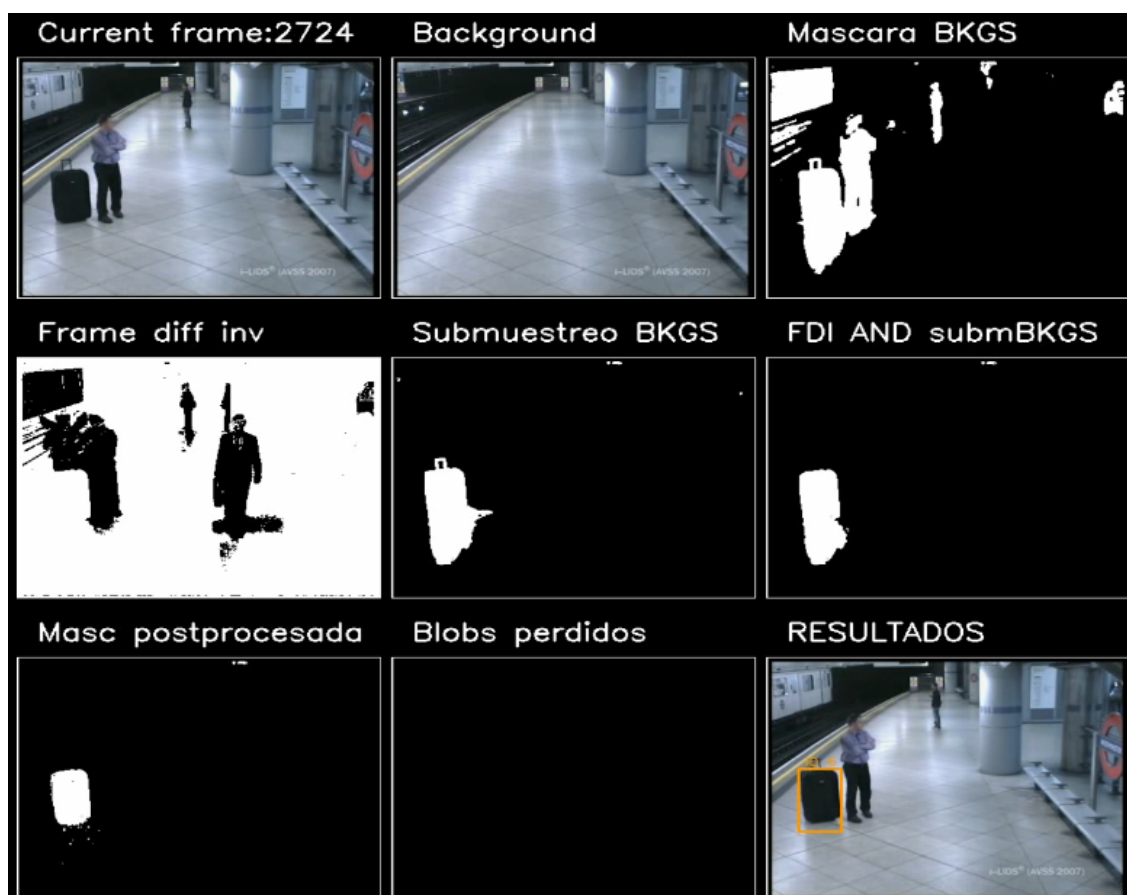


Figura 4-10 Ejemplo de la solución al problema de las oclusiones (I)

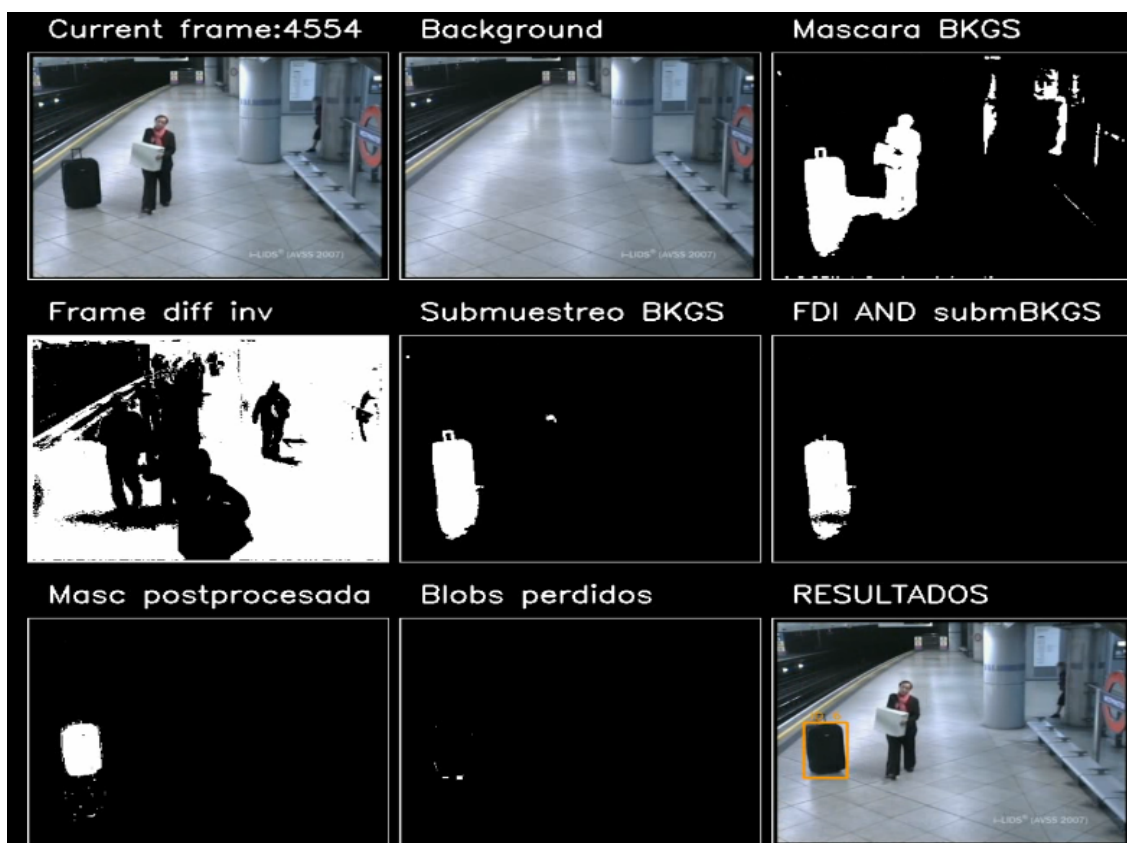


Figura 4-11 Ejemplo de la solución al problema de las oclusiones (II)

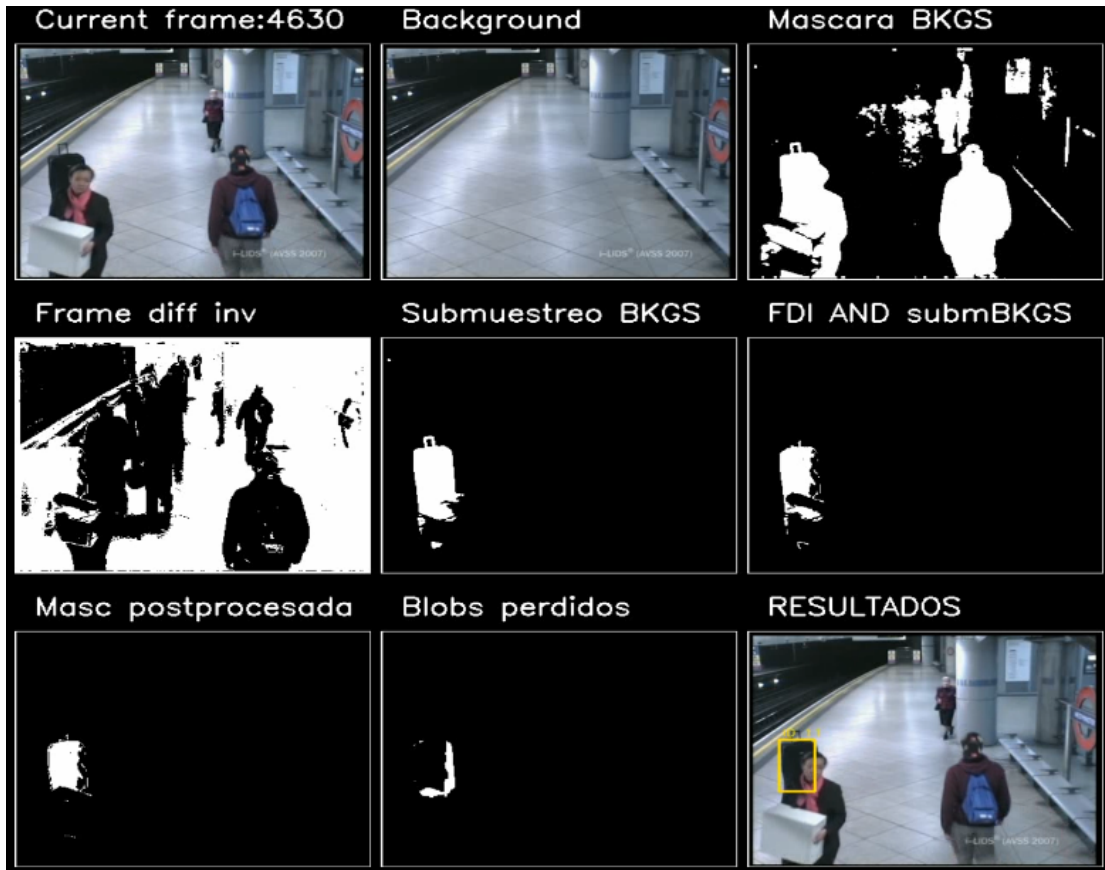


Figura 4-12 Ejemplo de la solución al problema de las oclusiones (III)

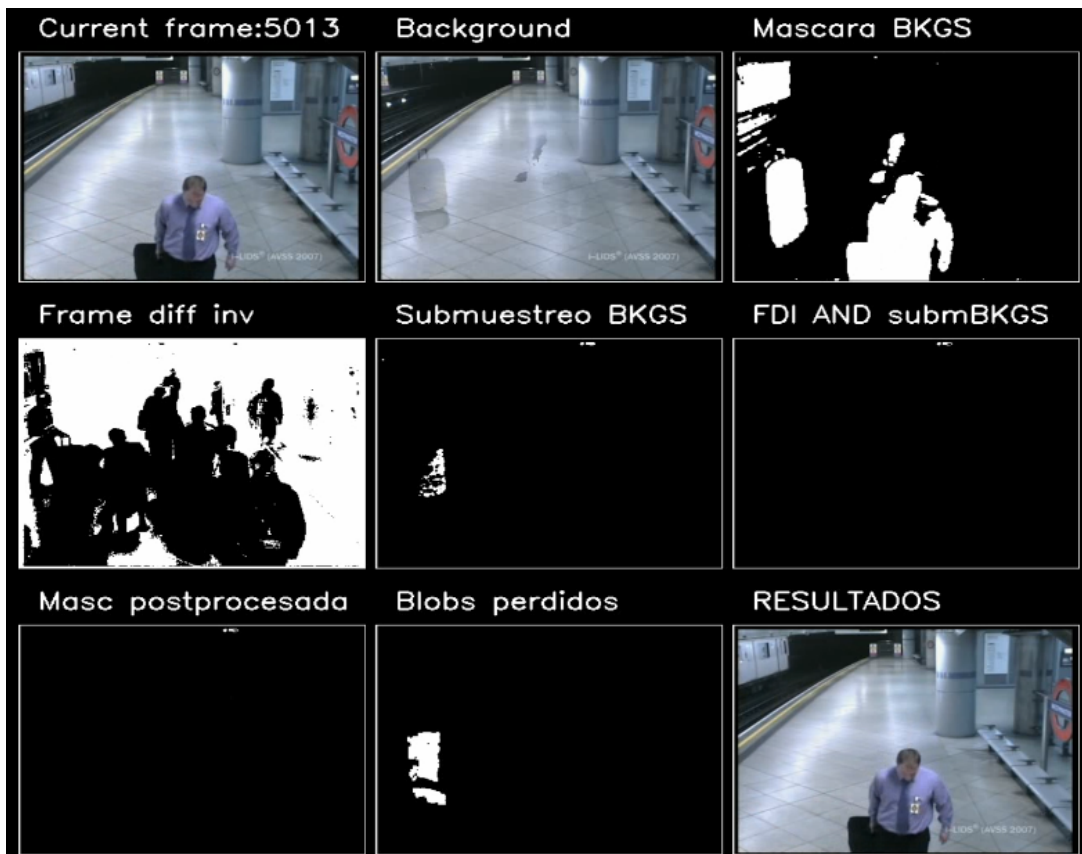


Figura 4-13 Ejemplo de la solución al problema de las oclusiones (IV)

En la figura 4.10 se aprecia un objeto abandonado en la imagen. En la figura 4.11 se ve venir a una persona que va a ocluir el objeto, veremos que ocurre. En la figura 4.12 se produce la oclusión parcial. Se puede observar como la máscara FDI (en la parte del objeto abandonado) tiene una zona en negro, y la máscara de submuestreo sigue en blanco, por lo tanto al hacer la multiplicación y obtener FM, la máscara se obtiene de modo incorrecto. Entonces, al post-procesar dicha máscara se comprueba que hay una oclusión (se aprecia también al ver que se ha perdido parte de un *blob* en la máscara *blobs* perdidos), y en la imagen final de resultados, se sigue marcando el objeto aunque pase la persona por delante.

En la figura 4.12, se ve que se retira el objeto, y el algoritmo no señala ninguna oclusión debido a que la máscara de submuestreo presenta la mayoría de valores a 0.

5 Clasificación robo/abandono

5.1 Introducción

A lo largo de este capítulo se describen los detectores de robo o abandono implementados en este proyecto que nos dirán, acorde con un análisis de diferentes características extraídas, si un objeto ha sido abandonado o robado de la escena.

Todos los detectores de este capítulo aplican una etapa de pre-procesado basada en contornos activos con el fin de seleccionar el contorno del objeto a analizar. Esta etapa es idéntica a la aplicada en el sistema base y esta descrita en la sección 3.5.

Cada detector genera una puntuación de robo y una puntuación de abandono proporcionada por el análisis realizado. Para generar una probabilidad (o evidencia) dada una puntuación x , se asume que las puntuaciones obtenidas tanto para robo como para abandono siguen una distribución Gaussiana con media μ y una desviación típica σ . Ambos parámetros se determinan experimentalmente para cada puntuación existente y para cada clase existente (robo, E_S o abandono, E_U) mediante el uso de un conjunto de datos de entrenamiento. Así para cada clase, la evidencia o probabilidad de un detector dada una puntuación obtenida es definida como un valor entre 0 y 1

Combinando dichas probabilidades de robo y abandono, se han implementado diversos esquemas de fusión seleccionando previamente los mejores detectores de los 7 existentes. Dichos esquemas de fusión tendrán como finalidad proporcionar resultados más estables, y por lo tanto ofrecer mayor fiabilidad a la decisión final tomada.

Este capítulo está dividido en 2 sub-apartados. En el primero se estudian los detectores implementados en el sistema base, junto al esquema de fusión seleccionado para ellos. En el segundo apartado, se detallan los nuevos detectores y nuevos esquemas de fusión propuestos que combinan las evidencias proporcionadas por cada detector (tanto de robo como de abandono) con el objetivo de mejorar los resultados previos y abrir nuevas líneas de trabajo futuro.

Destacar que un objeto se considerará abandonado si su evidencia final de abandono es mayor que su evidencia de robo, y robado si sucede lo contrario, si su evidencia de robo es

mayor que su evidencia de abandono, tanto para los detectores sencillos como para los esquemas de fusión implementados.

En la siguiente figura se muestra un ejemplo de funcionamiento de un detector de robo-abandono:

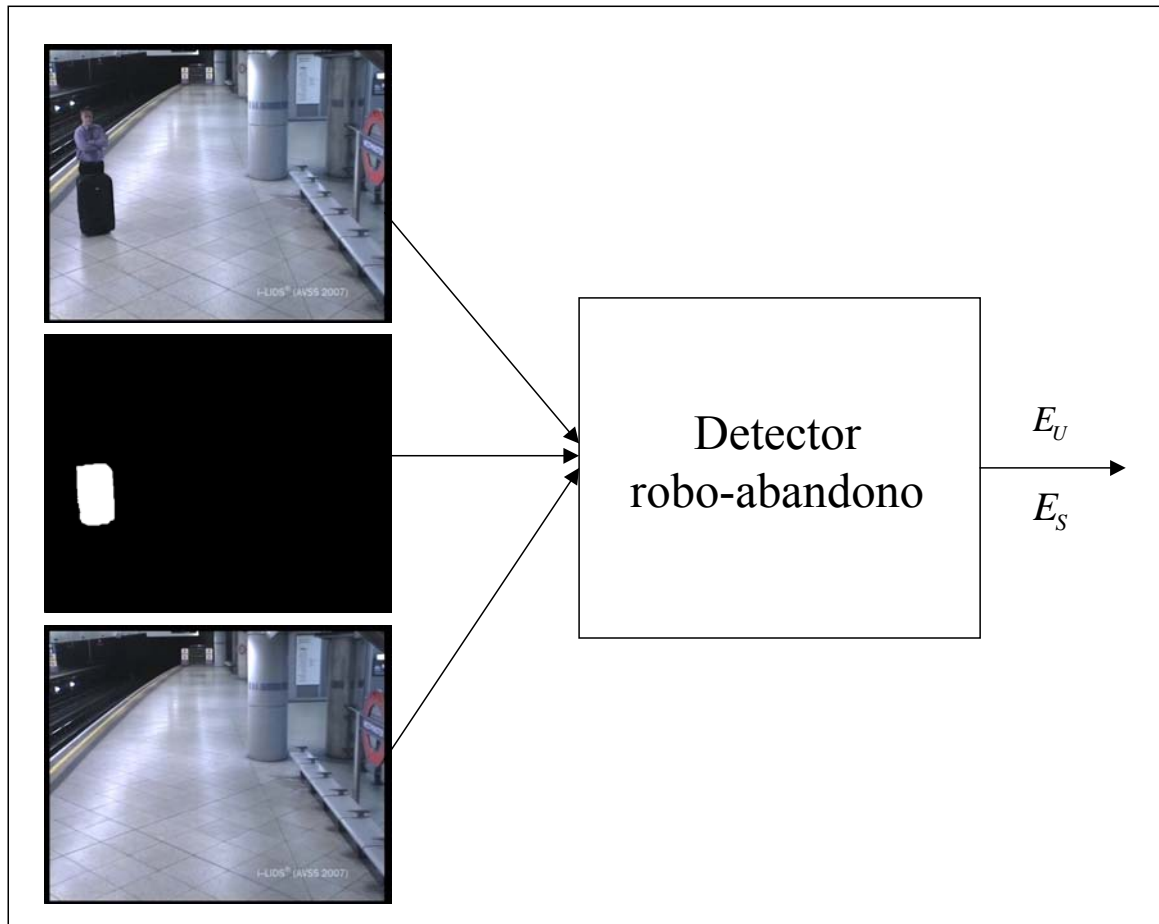


Figura 5-1 Ejemplo de funcionamiento de un detector de robo-abandono

En la figura 5.1 se muestran los elementos necesarios que necesita un detector: una imagen actual, una máscara de la región estática a clasificar, y una imagen del fondo. El detector devolverá una evidencia de robo (E_S), y una evidencia de abandono (E_U).

5.2 Sistema base

En este apartado se explican detalladamente los detectores implementados en el sistema base [20], así como el esquema final existente.

5.2.1 Detector basado en color

Este detector se basa en la semejanza de color entre las regiones delimitadas por el contorno de objeto. Esta aproximación se basa en comprobar si el color del *blob*/objeto bajo análisis es similar al color del “supuesto fondo” de la imagen (alrededores del *blob* delimitados por el rectángulo englobante y el contorno del objeto).

En el detector descrito, se ha mejorado la medida de similitud de color previa utilizando siempre el mismo fondo (calculado en la segmentación inicial), para las diferencias de color. La suposición de que el “supuesto fondo” es similar en la imagen actual y la imagen de fondo original no es siempre verdadera. Por ejemplo, si tenemos dos objetos robados a la vez y en regiones cercanas, el algoritmo propuesto podría reducir la eficiencia debido a la información de color que hay detrás de dicho objeto robado.

Usando el mismo fondo supuesto, la medida propuesta es robusta frente estos posibles eventos. Tras la extracción de la región de interés, se calcula el histograma de color de 3 imágenes usando el canal H (tonalidad) del modelo de color HSV.

Cada imagen la clasificamos como R1, para las partes de la imagen que no se corresponden con el objeto estático, y R2 para las partes de la imagen que se corresponden con el objeto estático (ejemplo en la figura 5.2, tercera imagen de la primera fila).

Estos tres histogramas son los siguientes, y se muestran en la figura 5.2:

- H1, que es el histograma perteneciente a R1 de la imagen de fondo).
- H2, que es el histograma perteneciente a R2 de la imagen actual).
- H3, que es el histograma perteneciente a R3 de la imagen de fondo.







Background Image	Current Image	Object mask
		
Color histogram of R1 in background image (H1)	Color histogram of R2 in current image (H2)	Color histogram of R2 in background image (H3)
		

Figura 5-2 Ejemplo de detector basado en color

Tras esto, se calcula la distancia de Battacharya (5.0) para obtener similitudes entre los histogramas de color H1 y H3, y H1 y H2.

$$dBat(H_i, H_j) = -Ln\left(\sum_{x \in X} \sqrt{H_i(x)H_j(x)}\right) \quad (5.0)$$

Este algoritmo determina el objeto como abandonado si el histograma H1 es muy parecido al histograma de H3. Por el contrario, si el histograma H1 es mas parecido a H2, el objeto es considerado como robado. La figura 5.2 muestra un ejemplo para el caso de un objeto abandonado.

Tras estos cálculos, se calcula el valor del *score* del detector, que se denota por M_{CH} (calculado como la diferencia entre las distancias de Battacharya entre los histogramas H1-H3 y H1-H2),y se usa para calcular las evidencias de robo y de abandono. Para cada clase (robo o abandono), la evidencia se define según las siguientes fórmulas:

$$M_{CH} = dBat(H1, H3) - dBat(H1, H2) \quad (5.1)$$

$$E_{CH(U, S)} = E_{\mu_{CH(U, S)}, \sigma_{CH(U, S)}} (M_{CH})$$

Donde μ y σ son, respectivamente, la media y la desviación típica de cada una de las clases (robo y abandono), calculada a partir de las secuencias de entrenamiento, y $E_{CH(U, S)}$ se corresponde con la evidencia de este modelo para la clase de robo o abandono.

5.2.2 Detectores basados en el gradiente alto y gradiente bajo

Estos dos detectores se basan en el análisis de los valores del gradiente a lo largo del contorno obtenido después de su ajuste mediante Contornos Activos en la imagen actual y el fondo de la escena (eliminando la información redundante). Así pues, en este tipo de detectores se analiza la información numérica a lo largo del contorno utilizando la técnica del gradiente.

Esta aproximación es similar a la mostrada en [20]. Las diferencias principales radican en el proceso de extracción del borde de objeto, y la eliminación de la información redundante del borde. El algoritmo que se utiliza para extraer los bordes es el de SUSAN, el cual solo utiliza información binaria para ello, y, computacionalmente es muy eficiente. Pero algunos recientes estudios muestran que el algoritmo de SUSAN no es la mejor opción para extraer la información de los bordes. Los métodos del gradiente alto y del gradiente bajo extraen la información de los bordes mediante el uso del gradiente, que proporciona más información que el algoritmo de SUSAN (por ejemplo, un rango de 0-255 en vez de un rango binario).

Después de la extracción y ajuste del contorno, se procede a calcular el gradiente en la imagen actual eliminando la información redundante. Esta información redundante son aquellos valores de gradiente idénticos en la imagen actual y fondo, es decir, existe un contorno (o no) en ambas imágenes (fondo e imagen actual) con lo cual no aporta información válida para el análisis a realizar. La diferencia entre el gradiente de la imagen actual y fondo garantiza la eliminación de dicha información. Después del proceso de cálculo del gradiente (detección de bordes), cada detector calcula las evidencias de ser un objeto abandonado o robado analizando el valor de los gradientes calculados.

El primer detector, el del gradiente alto, se basa en encontrar los puntos alrededor de los objetos ajustados mediante contornos activos de la imagen actual cuyo gradiente es máximo.

El segundo detector, el del gradiente bajo, se basa en encontrar los puntos alrededor de los objetos ajustados mediante contornos activos de la imagen actual cuyo gradiente es menor. Debido a la posibilidad de que la información extraída del gradiente no se analice exactamente a nivel de píxel, esta búsqueda se generaliza usando una pequeña ventana alrededor del punto examinado.

Las ecuaciones para calcular los *scores* correspondientes son las siguientes:

Gradiente alto:

$$M_{HG}(x, y) = \begin{cases} 1 & any(W(x, y)) > th_HG \\ 0 & othercase \end{cases} \quad (5.2)$$

Gradiente bajo:

$$M_{LG}(x, y) = \begin{cases} 1 & \frac{1}{N^2} \sum^W W(x, y) < th_LG \\ 0 & othercase \end{cases} \quad (5.3)$$

donde, $W(x,y)$ es la ventana de tamaño $N \times N$ centrada en un punto del contorno (x,y) y $\{th_HG, th_LG\}$ son los umbrales de selección para el gradiente alto y el gradiente bajo, respectivamente. En este proyecto, hemos decidido seleccionar los valores $th_HG = 220$ y $th_LG = 30$, tras realizar varias pruebas.

Finalmente, todos los valores de M_{HG} y de M_{LG} se usan para calcular la probabilidad de parecido del contorno. Para cada clase, U (robado) o S (abandonado), dicha probabilidad (E) se define, respectivamente, de la siguiente manera:

$$E_{HG(U,S)} = E_{\mu_{HG(U,S)}, \sigma_{HG(U,S)}} (M_{HG}) \quad (5.4)$$

$$E_{LG(U,S)} = E_{\mu_{LG(U,S)}, \sigma_{LG(U,S)}} (M_{LG}) \quad (5.5)$$

donde, $\mu_{HG(U,S)}$, $\sigma_{HG(U,S)}$, $\mu_{LG(U,S)}$ y $\sigma_{LG(U,S)}$ son los valores de las medias y las desviaciones estándar de las medidas del gradiente alto y del gradiente bajo, entrenadas por varias secuencias de test.

Como se puede ver en las figuras 5.3 y 5.4, el detector basado en el gradiente alto da una medida próxima a 1 cuando un objeto es abandonado y 0 cuando un objeto es robado (figura 5.3). En cambio, el detector basado en el gradiente bajo da un valor próximo a 1 cuando un objeto ha sido robado, y próximo a 0 cuando un objeto es abandonado (figura 5.4).

Background image	Current image	Object mask
Gradient of background image for analyzed object	Gradient of current image for analyzed object	Gradient Difference between current and background image
$M_{HGU} = 0.70$		$M_{HGS} = 0.30$

Figura 5-3 Ejemplo de los detectores de gradiente para un objeto abandonado





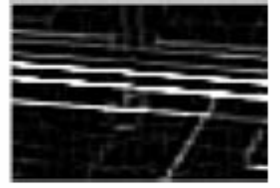

Background image	Current image	Object mask
		
Gradient of background image for analyzed object	Gradient of current image for analyzed object	Gradient Difference between current and background image
		
$M_{LGU} = 0.32$		$M_{LGS} = 0.68$

Figura 5-4 Ejemplo de los detectores de gradiente para un objeto robado

5.2.3 Combinación de detectores

En este esquema de fusión se combinan los detectores implementados anteriormente, que son el detector basado en el histograma de color, el basado en el gradiente alto, y el basado en el gradiente bajo, con el objetivo de encontrar unas evidencias más fiables a la hora de calificar un objeto como abandonado o robado (ver figura 5.5).

Estas evidencias finales, E_U y E_S , se obtienen combinando las evidencias obtenidas por cada uno de los tres detectores.

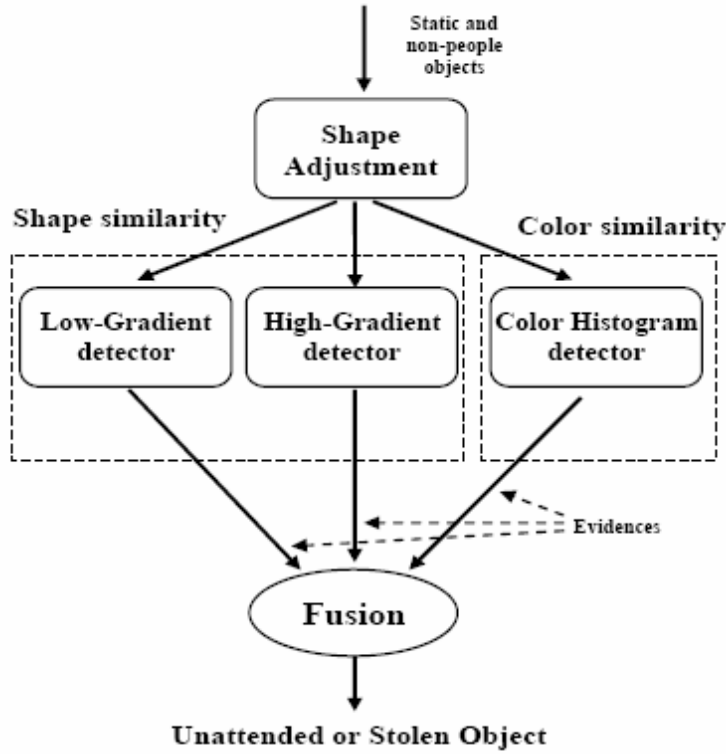


Figura 5-5 Esquema de fusión implementado para el sistema base

Las fórmulas para calcular E_U y E_S son las siguientes:

$$E_U = \frac{H(E_{HGU} - \rho)E_{HGU} + H(E_{LGU} - \rho)E_{LGU} + H(E_{CHU} - \rho)E_{CHU}}{H(E_{HGU} - \rho) + H(E_{LGU} - \rho) + H(E_{CHU} - \rho)} \quad (5.6)$$

$$E_S = \frac{H(E_{HGS} - \rho)E_{HGS} + H(E_{LGS} - \rho)E_{LGS} + H(E_{CHS} - \rho)E_{CHS}}{H(E_{HGS} - \rho) + H(E_{LGS} - \rho) + H(E_{CHS} - \rho)} \quad (5.7)$$

Los valores para las evidencias solo son tenidos en cuenta si superan un cierto umbral de relevancia ρ (En este proyecto, se ha determinado un valor de $\rho = 0.7$), ya que la función $H(x)$ es la función escalón unitario, que toma el valor 1 para cualquier valor de x mayor o igual que 0, y toma el valor 0 para los valores negativos de x .

En el caso de que los tres detectores tengan una evidencia menor que ρ , el resultado final sería la indeterminación 0/0, y la evidencia final se calcularía como la media de las tres evidencias obtenidas anteriormente.

Finalmente, el objeto se considera abandonado si $E_U > E_S$ y robado si $E_U < E_S$.

5.3 Detectores propuestos

En esta sección se definen los nuevos detectores propuestos para detectar objetos robados o abandonados, así como nuevos esquemas de fusión.

Estos esquemas de fusión se implementarán solamente para los dos mejores detectores individuales, que serán aquellos que mejores resultados obtengan tras una fase de entrenamiento.

5.3.1 Detector basado en gradiente global

El método del gradiente global consiste en integrar los métodos del gradiente alto y del gradiente bajo en un solo método. La forma de extraer las evidencias correspondientes al gradiente alto y al gradiente bajo es la misma que la de la sección 5.2.2, pero las evidencias finales varían debido a que dichas evidencias se suman para conseguir una evidencia más fiable.

$$E_{GRD(U, S)} = E_{HG(U, S)} + E_{LG(U, S)} \quad (5.8)$$

Como se ve en la ecuación 5.7, se obtendrán dos evidencias finales, una para el caso de robo y otra para el caso de abandono.

5.3.2 Detectores basados en la forma del contorno

En esta sección se implementarán tres detectores que se basan en la comparación de la información obtenida a través del contorno del objeto a analizar en la imagen actual, máscara binaria y fondo de la escena.

5.3.2.1 Detector basado en los momentos de Hu

Este detector aplica operaciones de matching de contornos basadas en los momentos de Hu. Para ello, se extraen dos contornos (uno de la máscara binaria y otro de la imagen

actual), y se calcula una similitud de contornos. Esta similitud se calcula utilizando los momentos de Hu.

Los momentos de Hu (siete en total) que se utilizarán son los siguientes:

$$\begin{aligned}
 \phi_1 &= \eta_{20} + \eta_{02} \\
 \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
 \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
 \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
 \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\
 &\quad (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
 \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + \\
 &\quad 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
 \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - \\
 &\quad (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
 \end{aligned}$$

Figura 5-6 Momentos de Hu

donde

$$\eta_{rs} = \frac{\mu_{rs}}{\mu_{00}^t} \quad t = \frac{r + s}{2} + 1 \quad (5.9)$$

y a su vez, μ_{rs} indica el momento central de orden $r + s$ normalizado respecto a otro momento fijo, μ_{00}^t

Estos momentos de Hu son invariantes a la traslación, rotación y escalamiento. Esto significa, que dos regiones que tengan la misma forma pero sean de distinto tamaño, y a su vez estén ubicadas en posiciones diferentes y orientadas de distinto modo, los momentos de Hu de ambas regiones serán iguales.

Tras esto, se calcula una distancia euclídea respecto al segundo momento de Hu (únicamente), y se decide si el objeto es abandonado o robado.

5.3.2.2 *Detector basado en la comparación de histogramas de contorno*

Al igual que el método anterior, este detector aplica operaciones de *matching* de contornos. Primero extrae los contornos del objeto pertenecientes a la máscara binaria y a la imagen actual, y calcula una similitud de contornos basado en el cálculo de un histograma de variación espacial para cada uno de los contornos.

Dichos histogramas se calculan considerando cada par de puntos de los bordes del contorno. El ángulo entre los bordes, y la distancia entre dichos puntos de los bordes se calcula para cada par de puntos. El ángulo entre cada uno de los puntos pertenecientes a los bordes define qué fila del histograma cambia su valor en relación a la distancia mínima y máxima calculada previamente.

Finalmente, calcula la distancia de Battacharya entre dichos histogramas, y esa distancia se usa para hacer un *matching* de contorno.

5.3.2.3 *Detector basado en comparación bi-direccional de contorno*

Este detector aplicar una nueva filosofía de comparación bi-direccional similar al detector del gradiente combinado. La idea es integrar la similitud del contorno extraído de la máscara binaria frente a la imagen actual y al fondo.

Este detector realiza la misma operación de *matching* de contorno que el detector basado en los momentos de Hu pero para dos pares de imagen, el par "contorno original"- "imagen actual" y "contorno original"- "imagen del fondo". El contorno extraído de la máscara es considerado como "contorno original" y los otros contornos se corresponden con el ajuste con Contornos activos en las imágenes actual y de fondo respectivamente

Esto devuelve dos marcadores, y el marcador final de robo o de abandono se obtiene sumando los dos marcadores anteriores.

5.4 *Esquemas de fusión*

El concepto de "esquemas de fusión" consiste en la combinación de dos o más detectores simples mediante algoritmos matemáticos con el objetivo de obtener resultados más fiables de cara a la clasificación de un objeto entre abandonado o robado.

La notación utilizada para toda esta sección es:

- $P(U)$: Probabilidad final de considerar un objeto abandonado.
- $P(S)$: Probabilidad final de considerar un objeto robado
- $P_{D_i}(U)$: Probabilidad del detector i ($i=1,2$) de considerar un objeto abandonado.
- $P_{D_i}(S)$: Probabilidad del detector i ($i=1,2$) de considerar un objeto robado

Cabe destacar que, para la mayoría de los esquemas de fusión, se ha calculado en la fórmula la probabilidad de robo $P(S)$, pero la fórmula sería análoga para el caso de calcular la probabilidad de abandono $P(U)$.

5.4.1 Esquema de fusión basado en la regla de la suma

Un esquema de fusión simple es el basado en la regla de la suma, que consiste en realizar una media aritmética tanto de las probabilidades de robo seleccionadas, como de las probabilidades de abandono. El esquema de fusión sigue la siguiente regla (en el ejemplo, para 2 detectores):

$$P(S | D_1, D_2) = \frac{P_1(S | D_1) + P_2(S | D_2)}{2} \quad (5.10)$$

donde $P(S | D_1, D_2)$ indica la probabilidad de un evento de ser considerado como robado (S) o abandonado (U), conociendo previamente las probabilidades de dos detectores seleccionados previamente, que son $P_1(S | D_1)$ y $P_2(S | D_2)$.

5.4.2 Esquema de fusión basado en la regla del producto

El esquema de fusión basado en la regla del producto es otro método bien conocido, que se detalla en la siguiente ecuación:

$$P(S | D_1, D_2) = \frac{P_1(S | D_1) * P_2(S | D_2)}{P_1(S | D_1) * P_2(S | D_2) + (1 - P_1(S | D_1)) * (1 - P_2(S | D_2))} \quad (5.11)$$

donde $P(S | D_1, D_2)$ indica la probabilidad de un evento de ser considerado como robado (S) o abandonado (U), conociendo previamente las probabilidades de dos detectores seleccionados previamente, que son $P_1(S | D_1)$ y $P_2(S | D_2)$.

Este método va a obtener unas probabilidades mas bajas, debido que al multiplicar dos números entre 0 y 1, el resultado final se va a acercar siempre más a 0 que a 1.

5.4.3 Esquema de fusión basado en la regla del máximo

El detector basado en la regla del máximo viene dado por una ecuación muy similar al detector basado en la regla del producto.

$$P(S | D_1, D_2) = \frac{\max(P_1(S | D_1), P_2(S | D_2))}{\max(P_1(S | D_1), P_2(S | D_2)) + \max(1 - P_1(S | D_1), 1 - P_2(S | D_2))} \quad (5.12)$$

En esta ecuación, lo que cambia respecto a la ecuación de la regla del producto es la selección del máximo de la probabilidad de ambos detectores, en vez de realizar la multiplicación, lo que conllevará a un numerador mayor, y un resultado más próximo a 1. La notación es la misma que para los detectores anteriores.

5.4.4 Esquema de fusión basado en la regla del mínimo

Otro método de fusión muy similar a los dos anteriores, aquí se sustituye la selección del máximo por la selección del mínimo, como se puede observar en la siguiente ecuación, que es la que describe al esquema de fusión:

$$P(S | D_1, D_2) = \frac{\min(P_1(S | D_1), P_2(S | D_2))}{\min(P_1(S | D_1), P_2(S | D_2)) + \min(1 - P_1(S | D_1), 1 - P_2(S | D_2))} \quad (5.13)$$

Aquí nos encontramos en el mismo caso que el del esquema de fusión basado en la regla del producto, el resultado obtenido será muy pequeño. La notación no varía.

5.4.5 Esquema de fusión basado en la media geométrica

Este esquema de fusión calcula la media geométrica de dos detectores simples de robo o abandono, mediante la siguiente ecuación:

$$P(S | D_1, D_2) = \sqrt{(P_1(S | D_1) * P_2(S | D_2))} \quad (5.14)$$

Este esquema de fusión es mas robusto que el de la media aritmética, ya que al realizar la media geométrica, no se nota tanto el error producido por uno de los dos detectores como en el esquema de fusión basado en la media aritmética.

5.4.6 Esquema de fusión basado en la estimación Bayesiana

La estimación Bayesiana ofrece una aplicación mucho más robusta a la hora de combinar detectores de robo/abandono debido al uso de las reglas de la teoría de la probabilidad (regla de Bayes).

La evidencia del detector se representa mediante una probabilidad condicionada $P(Y | X)$, donde $P(Y) = P(Y | X)$ si X permanece constante. Esta probabilidad $P(Y | X)$ toma un valor entre 0 y 1, donde el valor 1 representa una probabilidad del 100% de que ocurra el suceso Y (en nuestro caso, robo o abandono), a partir de cierta información del suceso X. El valor de 0 representa una probabilidad del 0%.

Este estimador Bayesiano se basa en el teorema de la probabilidad conocido como la Regla de Bayes, que dice:

$$P(Y | X) = \frac{P(X | Y)P(Y)}{P(X)} \quad (5.15)$$

donde $P(Y | X)$ representa la confianza acerca de la hipótesis del suceso Y en base de la información existente en el suceso X, que se calcula multiplicando la probabilidad del suceso Y, $P(Y)$, por el factor $P(X | Y)$, que es la probabilidad de obtener X sabiendo que Y es cierto. El denominador $P(X)$ es un factor de normalización.

Cuando nos encontramos ante varios detectores, podemos definir una función de confianza de la siguiente manera, para el caso de un objeto robado:

$$L(D_i | S) = \frac{P(D_i | S)}{P(D_i | -S)} \quad (5.16)$$

donde $P(D_i | S)$ es la probabilidad del detector i de considerar el objeto robado), $P(D_i | -S)$ es la probabilidad complementaria del caso anterior ($-S = 1-S$), y $L(D_i | S)$ es el valor de confianza para ese suceso.

Si definimos la probabilidad principal (que indica la probabilidad del evento S de ser robo) del siguiente modo:

$$O(S) = \frac{P(S)}{P(-S)} \quad (5.17)$$

y asumiendo que el resultado obtenido para cada uno de los detectores es independiente entre ellos, el valor de confianza final viene dado por la siguiente expresión (en nuestro caso solo utilizaremos dos detectores):

$$O(S | D_1, D_2) = O(S) \prod_{i=1}^2 L(D_i | S). \quad (5.18)$$

Para después, calcular la probabilidad final de robo o abandono de la siguiente manera:

$$P(S | D_1, D_2) = \frac{O(S | D_1, D_2)}{1 + O(S | D_1, D_2)} \quad (5.19)$$

Esta fórmula es la que usaremos para calcular la probabilidad de robo o abandono para este nuevo esquema de fusión.

A continuación se mostrará un ejemplo para entender mejor toda esta notación matemática. Dados dos detectores $S1$ y $S2$, llamaremos $P(U)$ a la probabilidad de objeto abandonado, y $P(S)$ a la probabilidad de objeto robado, tenemos los siguientes datos:

$$P(S) = P(U) = 0.5, \quad P(D_1 | S) = 0.82 \text{ y } P(D_2 | S) = 0.71.$$

Aplicamos la ecuación 5.18 obtenemos los siguientes resultados:

$$O(S | D_1, D_2) = \frac{P(S)}{P(-S)} \cdot \frac{P(D_1 | S)}{P(D_1 | -S)} \cdot \frac{P(D_2 | S)}{P(D_2 | -S)} = \frac{0.5}{0.5} \cdot \frac{0.82}{0.18} \cdot \frac{0.71}{0.29} = 11.15.$$

A continuación, aplicamos la ecuación 5.19 para obtener la probabilidad final:

$$P(S | D_1, D_2) = \frac{O(S | D_1, D_2)}{1 + O(S | D_1, D_2)} \cdot \frac{11.15}{12.15} = 0.92$$

Esto nos dice que la probabilidad del suceso S (robo) para los detectores S1 y S2 es de un 92%, mejorando la probabilidad previa que teníamos en cada uno de los detectores.

Cabe destacar que la probabilidad de abandono para el mismo ejemplo no sería del 8%, ya que todos los detectores con los que se trabaja tienen un grado de incertidumbre.

5.4.7 Método de Dempster-Shafer

El método de Dempster-Shafer se basa en la mezcla de información procedente de varios detectores, combinando dicha información para mejorar los resultados obtenidos de modo independiente por cada uno de los detectores.

Este método se usa para fusionar la información de dos detectores, D_1 y D_2 . De cada detector, se tiene una probabilidad de robo $P_{D_i}(S)$, otra probabilidad de abandono $P_{D_i}(U)$ y una probabilidad de incertidumbre $P_{D_i}(I)$. La suma de estas 3 probabilidades tiene que ser 1.

Dicho método difiere con la estimación Bayesiana en que las probabilidades que se calculan en él dependen de las condiciones de robo, abandono y de incertidumbre, y no solo de la condición de robo o de abandono como en el caso anterior.

Combinando dichas probabilidades, se obtienen 9 probabilidades, 3 pertenecientes al abandono, 3 pertenecientes al robo, y otras 3 probabilidades que no utilizaremos debido a que se basan en el cruce de eventos incompatibles (robo-abandono, $P_{D_1}(S)$ y $P_{D_2}(U)$, y abandono-robo, $P_{D_1}(U)$ y $P_{D_2}(S)$), y al cruce de ambas incertidumbres ($P_{D_1}(I)$ y $P_{D_2}(I)$).

La probabilidad final de robo y de abandono, se calcula de la siguiente manera:

$$P(U) = P_1(U) + P_2(U) + P_3(U) \quad (5.20)$$

$$P(S) = P_1(S) + P_2(S) + P_3(S) \quad (5.21)$$

donde, las probabilidades parciales son:

$$\begin{aligned}
 P_1(U) &= \frac{P_{D_1}(U) * P_{D_2}(U)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))} \\
 P_2(U) &= \frac{P_{D_1}(U) * P_{D_2}(I)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))} \\
 P_3(U) &= \frac{P_{D_1}(I) * P_{D_2}(U)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))} \\
 P_1(S) &= \frac{P_{D_1}(S) * P_{D_2}(S)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))} \\
 P_2(S) &= \frac{P_{D_1}(S) * P_{D_2}(I)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))} \\
 P_3(S) &= \frac{P_{D_1}(I) * P_{D_2}(S)}{1 - (P_{D_1}(U) * P_{D_2}(S) + P_{D_2}(U) * P_{D_1}(S))}
 \end{aligned} \quad (5.22)$$

A continuación, se muestra un ejemplo de dicho algoritmo en la tabla 5.1:

	$P_{D_2}(S) = 0.71$	$P_{D_2}(U) = 0.12$	$P_{D_2}(I) = 0.17$
$P_{D_1}(S) = 0.82$	$P_1(S) = 0.72775$	$k_1 = 0.1148$	$P_2(S) = 0.15375$
$P_{D_1}(U) = 0.12$	$k_2 = 0.0852$	$P_1(U) = 0.021$	$P_2(U) = 0.0225$
$P_{D_1}(I) = 0.06$	$P_3(S) = 0.05325$	$P_3(U) = 0.0105$	$k_3 = 0.01125$

Tabla 5-1. Ejemplo del modelo de fusión de Dempster-Shafer

La probabilidad final de robo, $P(S) = P_1(S) + P_2(S) + P_3(S)$, será de 0.93745.

La probabilidad final de abandono, $P(U) = P_1(U) + P_2(U) + P_3(U)$, será de 0.054.

La probabilidad final de incertidumbre, de la combinación de los dos detectores, que corresponde a k_3 , es de 0.01125.

La suma de estas tres posibilidades, es 1, por lo tanto se corrobora que estamos ante un método que corrobora las reglas de la teoría de la probabilidad, descartándose el valor de k_1 y k_2 .

6 Resultados experimentales

En este capítulo se muestran los resultados experimentales de los dos módulos en los que se ha trabajado para mejorar los algoritmos existentes del sistema base: el módulo de detección de regiones estáticas y el módulo de clasificar un objeto entre objeto robado y objeto abandonado.

Para la implementación de todos los algoritmos, se ha utilizado la librería pública de tratamiento de imagen OpenCV (disponible en <http://sourceforge.net/projects/opencv>). Los tests realizados se han ejecutado en un Pentium IV que opera a 2.8GHz y 1GB de RAM.

El capítulo está dividido en dos apartados: El primer apartado se centrará en la evaluación de los algoritmos de detección de regiones estáticas implementados (apartado 6.1), y el segundo evaluará los diferentes detectores implementados en este PFC para distinguir el evento objeto robado del evento objeto abandonado (apartado 6.2).

6.1 Resultados de los métodos de detección de regiones estáticas implementados

En este apartado se va a proceder a la evaluación de los algoritmos de detección de regiones estáticas. Para proceder a la evaluación de estos algoritmos, primero se va a hablar de los *datasets* utilizados para proceder a dicha evaluación, después se hablará de la métrica que se ha utilizado para realizar la evaluación, y finalmente se hará una comparativa objetiva y subjetiva de los algoritmos implementados. Esta parte se distingue entre objetiva y subjetiva debido a que la clasificación objetiva se hace en base a un “*ground truth*” generado por dichos *datasets* utilizados, en cambio, no poseemos de dicha información perteneciente al “*ground truth*” para comparar los datos pertenecientes a la evaluación subjetiva.

Este apartado se divide en 4 sub-apartados, el primero explica las secuencias de vídeo utilizadas (apartado 6.1.1), el segundo y el tercero cuentan los parámetros utilizados para la realización de dichas pruebas clasificando dichos parámetros en objetivos (apartado 6.1.2), y subjetivos (apartado 6.1.3), y el cuarto sub-apartado realizará una comparativa entre las medidas objetivas y las medidas subjetivas (apartado 6.1.4).

6.1.1 Datasets utilizados

Para realizar la evaluación de los algoritmos de detección de regiones estáticas implementados en este PFC, se van a utilizar dos datasets bastante utilizados en el análisis de secuencias de vídeo-seguridad: PETS2006 y AVSS2007 (explicados en el apartado 2.4 de este PFC). Estas secuencias de vídeo-seguridad se pueden clasificar en dos categorías, simple (secuencias correspondientes al *dataset* PETS2006) y complejo (secuencias correspondientes al *dataset* AVSS2007).

El escenario simple presenta las siguientes características:

- Dificultad medio-baja para extraer regiones estáticas pertenecientes al *foreground*.
- Densidad media de objetos en la imagen.
- Presenta cambios muy suaves de iluminación.
- Velocidad de los objetos en movimiento bastante baja
- No presenta oclusiones en las regiones estáticas pertenecientes al *foreground*

A continuación, se muestran algunas imágenes pertenecientes al escenario sencillo (figura 6.1):



Figura 6-1 Ejemplos de imágenes del escenario sencillo

A su vez, el escenario complejo presenta las siguientes características:

- Alta dificultad para extraer regiones estáticas pertenecientes al *foreground*
- Alta densidad de objetos o personas en la imagen.
- Presenta cambios ligeros de iluminación
- Velocidad media-alta de los objetos en movimiento de la escena
- Presenta muchas oclusiones en las regiones estáticas pertenecientes al *foreground*.

A continuación, se muestran algunas imágenes pertenecientes al escenario complejo (figura 6.2):

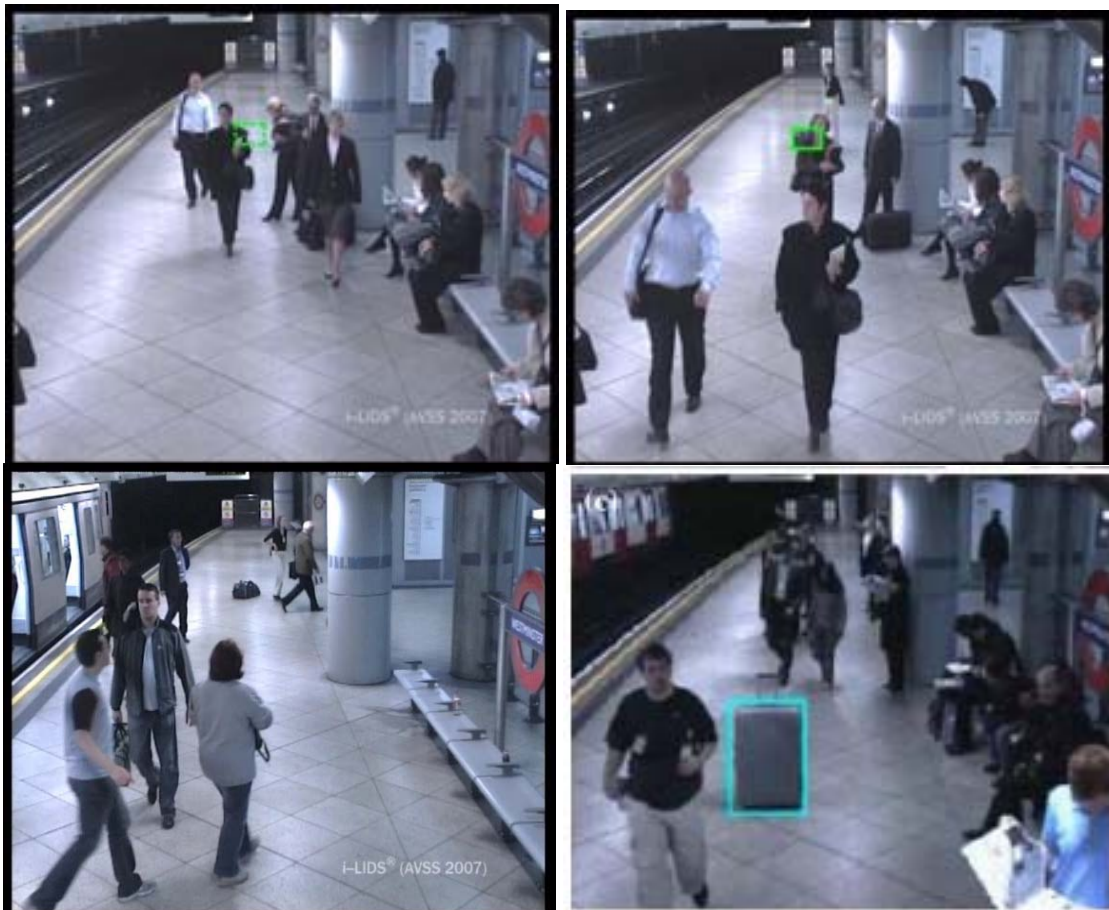


Figura 6-2 Ejemplos de imágenes del escenario complejo.

6.1.2 Medidas objetivas

En este apartado se comparan los resultados obtenidos de modo objetivo y subjetivo para cada uno de los algoritmos implementados en cada uno de los escenarios explicados en el apartado anterior.

Como medidas objetivas, hemos seleccionado la precisión (en segundos) de la detección de la región estática, la duración de la señal de alarma, y la precisión y el recall. Pasaremos a dar una definición detallada de ellos:

Detección de la región estática: Esta medida se define apuntando el segundo en el que el algoritmo detecta una región estática en la escena. Esta medida se compara frente a los datos del “*ground-truth*”

Duración de la señal de alarma: Mide el tiempo que el objeto permanece estático en la escena. Este contador de tiempo se parará o bien, cuando la región estática desaparezca, o bien, cuando acabe el vídeo a procesar. Esta medida se compara frente a los datos del “*ground-truth*”

Precisión: Se define como el número total de eventos de robo o abandono detectados correctamente con respecto al total de regiones estáticas detectadas. Se expresa en la siguiente ecuación:

$$PR = \frac{TP}{TP + FP}$$

donde PR es la precisión, TP indica el número de eventos detectados correctamente y FP el número de falsos positivos detectados por el algoritmo. El valor de la precisión siempre va a oscilar entre 0 y 1, siendo 0 cuando no se detecta ningún evento correcto, y 1 cuando sólo se detectan los eventos correctos.

Recall: Se define como el número de eventos de robo o abandono detectados correctamente con respecto al total de eventos de robo o abandono (que marca el ‘ground truth’) existentes en la secuencia a analizar. La ecuación que describe el recall es la siguiente:

$$R E = \frac{T P}{T P + T N}$$

donde, RE es el recall, TP indica el número de eventos detectados correctamente y TN el número de eventos existentes en la secuencia a analizar que no han sido detectados. El valor del recall siempre estará entre 0 y 1, y vale 0 cuando no se detecta ningún evento correcto y 1 cuando se detectan todos los eventos correctos.

Pero la precisión sirve para orientarse mucho mejor acerca de aquellos sistemas donde se detectan muchos falsos positivos, y el recall para aquellos otros sistemas que introducen muy pocos falsos positivos a costa de perder eventos detectados.

6.1.3 Medidas subjetivas

Se han seleccionado 5 medidas subjetivas para poder llevar a cabo un análisis de los algoritmos de detección de regiones estáticas implementados. Los parámetros seleccionados son los siguientes:

- **Exactitud a la hora de extraer la máscara final de regiones estáticas (MS1):** mide la precisión con la que el método a estudiar extrae la máscara final de regiones estáticas.
- **Tolerancia a oclusiones (MS2):** mide si el método estudiado tolera oclusiones tanto parciales, como totales.
- **Ruido introducido (MS3):** mide el ruido introducido en la máscara final de regiones estáticas.
- **Carga computacional (MS4):** mide el tiempo de ejecución de cada uno de los algoritmos implementados.
- **Dificultad a la hora de ajustar los parámetros (MS5):** mide el grado de dificultad de cada método para ajustar los parámetros con el fin de detectar regiones estáticas.

6.1.4 Comparativa

Para realizar la comparativa correspondiente entre todos los métodos implementados, y en función de las medidas a observar, se van a mostrar 4 tablas que muestran el funcionamiento de dichos algoritmos.

La primera tabla (tabla 6.1) muestra los resultados de los métodos implementados en el escenario sencillo, la segunda tabla (tabla 6.2) muestra los resultados de los métodos implementados en el escenario complejo, la tercera tabla (tabla 6.3) muestra los valores de la precisión y el recall para todos los métodos, y la cuarta tabla (tabla 6.4) muestra una comparación subjetiva en función de las 5 medidas subjetivas explicadas en el apartado anterior.

Cabe destacar, que la notación método 4.1.1 se refiere al método implementado en el apartado 4.1.1 de este PFC, y así sucesivamente.

Escenario sencillo	Ground truth		Método 4.1.1		Método 4.1.2		Método 4.1.3		Método 4.1.4		Método 4.1.5		Método propuesto	
	S1_T1_C3	2:05	0:36	2:03	0:37	2:06	0:37	2:08	0:38	2:05	0:36	2:05	0:36	2:05
S4_T1_C3	1:54	0:37	1:56	0:34	1:55	0:41	1:56	0:38	1:55	0:36	1:55	0:37	1:54	0:37
S5_T1_C3	2:11	0:56	2:12	0:54	2:14	0:59	2:14	0:57	2:11	0:56	2:11	0:56	2:11	0:56

Tabla 6-1. Evaluación objetiva de los métodos implementados en el escenario simple

Escenario complejo	Ground truth		Método 4.1.1		Método 4.1.2		Método 4.1.3		Método 4.1.4		Método 4.1.5		Método propuesto	
	AB_easy	1:52	0:08	1:52	0:08	1:52	0:08	1:53	0:08	1:52	0:08	1:52	0:08	1:52
AB_medium	1:43	0:19	1:44	0:18	1:43	0:19	1:45	0:18	1:43	0:19	1:43	0:19	1:43	0:19
AB_hard	1:26	0:26	1:26	0:26	1:26	0:26	1:27	0:25	1:26	0:26	1:26	0:26	1:26	0:26

Tabla 6-2. Evaluación objetiva de los métodos implementados en el escenario complejo

En estas dos tablas se puede observar como los resultados empeoran del escenario sencillo al escenario complicado. En el escenario sencillo (tabla 6.1), los resultados son mejores, ya que los resultados se parecen a los resultados aportados por el “*ground-truth*”. Esto es debido a que en dicho escenario, los objetos estáticos se localizan en regiones muy poco pobladas donde el número de oclusiones, cambios de iluminación y objetos en movimiento es muy bajo. Además, la gran diferencia de color existente entre el objeto a detectar y el fondo de la imagen permite proceder a una identificación casi perfecta mediante el uso de técnicas de segmentación frente-fondo.

En cambio, en la Tabla 6.2, perteneciente al escenario complicado, se observa mayor heterogeneidad en los resultados obtenidos. En esta secuencia, los objetos abandonados se localizan en lugares donde hay continuamente gente cruzándose, por lo tanto es bastante fácil que se produzcan oclusiones.

De este análisis, podemos observar que el método implementado en este PFC para detectar regiones estáticas presenta los mejores resultados, debido a la combinación de técnicas de submuestreo junto a técnicas de *frame difference*.

Si se analizan ambos escenarios a la vez, se observa que, si se incrementa el número de objetos móviles en la escena y el número de oclusiones, los resultados son mucho menos precisos, pero debido a que los algoritmos se han ajustado para acercarse lo más posible a los datos aportados por el *ground-truth*, el tiempo de detección solo varía en unos pocos segundos.

Las aproximaciones que peores resultados presentan son las que se basan en un modelo tradicional, debido a que las oclusiones constantes hacen que se tarde en reconocer las regiones estáticas.

Aproximación	ESCENARIO		ESCENARIO	
	SENCILLO (PETS2006)		COMPLEJO (AVSS2007)	
	Precision	Recall	Precision	Recall
4.1.1	0.05	1	0.01	1
4.1.2	0.6	1	0.1	1
4.1.3	0.5	1	0.03	1
4.1.4	0.75	1	0.33	1
4.1.5	0.37	1	0.05	1
Método propuesto	1	1	0.47	1

Tabla 6-3. Evaluación objetiva (precision y recall) de los métodos implementados

En la tabla 6.3 se observa que el recall para todos los casos es 1. Esto se debe a que todos los algoritmos implementados reconocen las regiones estáticas pertenecientes a objetos abandonados o robados dentro de la escena. Sin embargo, se puede ver como el valor de la precisión disminuye considerablemente cuando se pasa del escenario sencillo al escenario complejo. El algoritmo propuesto, que tenía un valor de precisión del 100% para el escenario sencillo, cae hasta el 47% cuando cambiamos de escenario.

Cabe destacar que obtiene los mejores resultados dicho algoritmo, debido a que el número de falsos negativos es mucho menor que en el resto de los casos.

Aproximación	MS1	MS2	MS3	MS4	MS5
4.1.1	Muy bajo	Muy bajo	Muy alto	Medio	Muy bajo
4.1.2	Alto	Medio	Bajo	Medio	Alto
4.1.3	Bajo	Alto	Alto	Alto	Muy alto
4.1.4	Alto	Muy alto	Muy bajo	Medio	Medio
4.1.5	Medio	Alto	Medio	Bajo	Bajo
Metodo propuesto	Muy alto	Muy alto	Muy bajo	Bajo	Bajo

Tabla 6-4. Evaluación subjetiva de los métodos implementados

Los resultados de la evaluación subjetiva se recogen en la tabla 6.4. Se puede observar que las oclusiones no influyen para nada en las aproximaciones basadas en el sub-muestreo de máscaras binarias, así como en el modelo de acumulación de máscaras de *foreground* o en el algoritmo implementado.

Respecto al ruido introducido, las aproximaciones basadas en técnicas de sub-muestreo obtienen mejores resultados porque el ruido, que se supone independiente, se submuestra reduciendo su efecto. En las aproximaciones clásicas, suele introducirse bastante ruido en la máscara final de regiones estáticas que debe de ser eliminado en posteriores etapas de procesado, utilizando por ejemplo operadores morfológicos o filtrado de ruido.

El coste computacional es muy alto en las aproximaciones que se basan en las propiedades del modelo de fondo, debido a que hay que realizar bastantes iteraciones en cada imagen y en cada píxel. En el método implementado, el coste computacional es bajo, debido a que sólo se usa una etapa de fondo, y no se hacen todas las comprobaciones del algoritmo a cada imagen, sino cada 175 imágenes (que es el tiempo de muestreo).

El ajuste de parámetros es bastante complicado en las aproximaciones basadas en las propiedades del modelo matemático, debido a que es necesario hacer un ajuste muy

detallado para no obtener resultados incorrectos, y son varios los parámetros a ajustar (actualización, cambio de estado en las Gaussianas, etc...). Sin embargo, las aproximaciones clásicas no poseen una dificultad elevada para ajustar los parámetros, debido a que ligeros errores a la hora de encontrar los parámetros óptimos pueden ser solventados por las posteriores etapas de análisis. Los métodos basados en el submuestreo presentan una dificultad media, debido a que la tasa de muestreo es un parámetro crítico y difícil de ajustar.

6.2 Resultados de los detectores de robo-abandono implementados

En este apartado se ha analizado el funcionamiento de los detectores de robo-abandono del sistema base y los propuestos.

6.2.1 Datasets utilizados

Este análisis se ha realizado sobre una clasificación basada en dos *datasets* utilizados: secuencias reales y secuencias cromas (grabadas por el grupo de investigación VPULab) [27].

A su vez, se ha realizado otra clasificación referente a las secuencias cromas, en función de la complejidad del modelo de fondo a analizar. Se han establecido tres categorías: fondo sencillo, fondo medio y fondo complejo. El fondo sencillo es un fondo uniforme, el fondo intermedio es un fondo un poco más saturado con más variaciones y el fondo complejo es el que posee una mayor dificultad para analizar. Estos fondos se muestran en la figura 6.3.

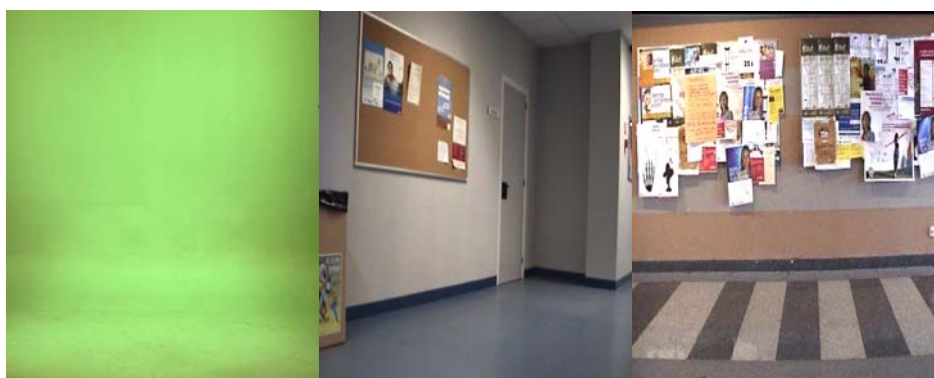


Figura 6-3 Diferentes tipos de fondos utilizados (fácil-medio-complejo)

En resumen, las anotaciones sobre secuencias cromas se van a utilizar para entrenar los detectores disponibles y las anotaciones sobre secuencias reales para la fase de test.

6.2.2 Fase de entrenamiento

Para proceder al análisis de los detectores, primero vamos a evaluar las secuencias cromas. Esta evaluación servirá para realizar el entrenamiento de los detectores existentes e implementados. En concreto, se determinará la media y varianza de los modelos Gaussianos de evidencia de cada detector. Adicionalmente, también se muestran las medias/varianzas del conjunto de test (secuencias reales) para observar la desviación existente.

Para ello, obtendremos primero una distribución Gaussiana a través de los *scores* obtenidos para dichos detectores.

A continuación se muestran las tablas de medias y varianzas obtenidas para cada distribución Gaussiana diferenciando entre las obtenidas para el evento de robo y el evento de abandono para cada detector implementado.

Para estas tablas, utilizaremos la siguiente notación para referirnos a los detectores:

Histograma de color (CH), gradiente alto (GH), gradiente bajo(GL), gradiente combinado (GRD), momentos de Hu (CON), comparación de histogramas de contorno (SH), comparación en dos direcciones de contorno (CONB).

	CROMA_facil		CROMA_medio		CROMA_complejo	
	μ	σ	μ	σ	μ	σ
CH	ROBO: 0.37 AB: -0.71	ROBO: 0.2 AB: 0.19	ROBO:0.62 AB:-0.81	ROBO:0.19 AB:0.16	ROBO:0.47 AB:-0.5	ROBO:0.25 AB:0.29
GH	ROBO:0.00086 AB:0.17	ROBO: 0.0064 AB: 0.29	ROBO:0.0021 AB:0.54	ROBO:0.015 AB:0.45	ROBO:0.035 AB:0.54	ROBO:0.12 AB:0.43
GL	ROBO:0.97 AB:0.29	ROBO: 0.14 AB: 0.42	ROBO:0.99 AB:0.23	ROBO:0.068 AB:0.39	ROBO:-0.95 AB:0.28	ROBO:0.15 AB:0.42
GRAD	ROBO:-1 AB:-0.048	ROBO: 0.035 AB: 0.56	ROBO:-0.99 AB:0.31	ROBO:0.05 AB:0.74	ROBO:0.9 AB:0.27	ROBO:0.26 AB:0.78
CON	ROBO: 0.85 AB: 0.69	ROBO: 0.15 AB: 0.37	ROBO:0.86 AB:0.69	ROBO:0.13 AB:0.38	ROBO:0.88 AB:0.69	ROBO: 0.12 AB: 0.38
CONB	ROBO: 0.18 AB:-1.5	ROBO: 0.41 AB:3.1	ROBO:0.18 AB:0.01	ROBO: 0.4 AB:2.4	ROBO:0.22 AB:0.72	ROBO:0.46 AB:3.4
SH	ROBO: 1 AB: 0.44	ROBO: 2.2 AB: 1.8	ROBO:0.88 AB:0.27	ROBO: 1.6 AB:1.5	ROBO: 1 AB: 0.39	ROBO: 2 AB:2.9

Tabla 6-5. Valores de μ y σ para los diferentes tipos de secuencias Croma

Destacar que estos datos se han obtenido con 187 muestras para la categoría CROMA_facil, 748 muestras para la categoría CROMA_medio y 561 muestras para la categoría CROMA_complejo.

Como se ve en la tabla, hay detectores que van a obtener muy buenas evidencias, debido a que los valores de μ y σ son bastante diferentes entre si, y no se van a solapar unos con otros. En cambio, otros detectores como el caso CON, obtiene valores muy malos para la media y la varianza, y esto causará errores de detección (el valor de la media para el robo y

para el abandono se puede solapar a poco que se separen un poco sus scores de la media obtenida, ya que estará mas cerca de la media del otro evento que del evento que le corresponde).

En la siguiente tabla se muestran los valores obtenidos tanto de la media como de la varianza para las Gaussianas de para las secuencias cromas (1496 muestras en total) frente a las secuencias reales (287 muestras en total):

	CROMA		REALES	
	μ	σ	μ	σ
CH	ROBO: 0.53 AB: -0.68	ROBO: 0.23 AB: 0.26	ROBO:0.15 AB:-0.34	ROBO:0.19 AB:0.16
GH	ROBO:0.014 AB:0.5	ROBO: 0.075 AB: 0.45	ROBO:0.25 AB:0.4	ROBO:0.19 AB:0.23
GL	ROBO:0.97 AB:0.26	ROBO: 0.11 AB: 0.4	ROBO:0.71 AB:0.4	ROBO:0.2 AB:0.25
GRAD	ROBO:-0.96 AB:0.25	ROBO: 0.17 AB: 0.74	ROBO:-0.35 AB:0.07	ROBO:0.41 AB:0.45
CON	ROBO: 0.86 AB: 0.69	ROBO: 0.13 AB: 0.38	ROBO:0.81 AB:0.59	ROBO:0.07 AB:0.15
CONB	ROBO: 0.19 AB:-0.46	ROBO: 0.43 AB:3	ROBO:0.5 AB:0.024	ROBO: 1.1 AB:1.2
SH	ROBO: 0.98 AB: 0.33	ROBO: 1.9 AB: 2.1	ROBO:0.68 AB:0.13	ROBO: 0.64 AB:0.15

Tabla 6-6. Valores de μ y σ para secuencias Croma y reales

De esta tabla se puede ver como los resultados varían ligeramente al pasar de las secuencias CROMA a las secuencias reales. Posibles máscaras mal obtenidas para determinar los objetos pueden desencadenar estos resultados.

Destacar lo mismo que en la tabla 6.5, hay algunos detectores que se van a solapar. Lo ideal es que los detectores quedaran todos de la siguiente manera, como se muestra en la figura 6.4, y no del modo que quedan algunos, como el que se muestra en la figura 6.5:

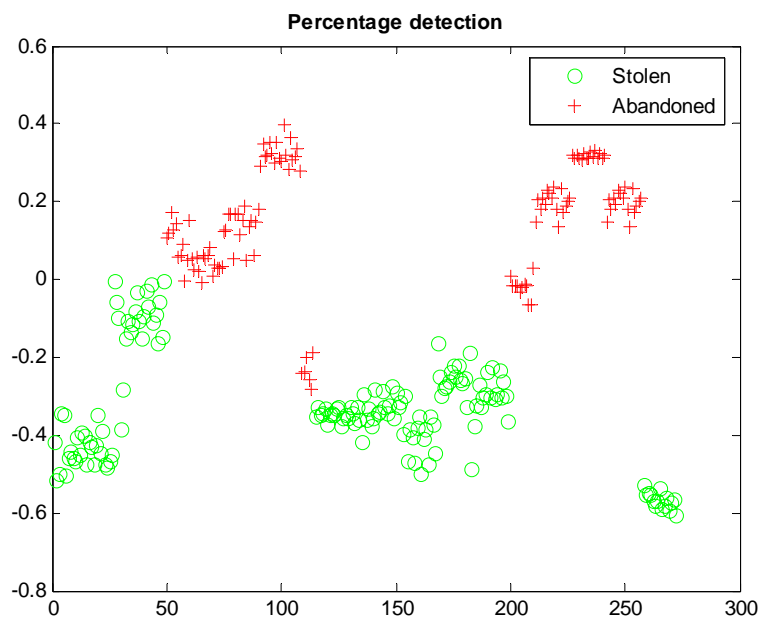


Figura 6-4 Resultados obtenidos para el detector histograma de color

Se puede ver en la figura 6.4 como las zonas tanto de robo como de abandono están bien definidas, y salvo un par de eventos de abandono que caen por la zona de robo, se pueden obtener resultados con un alto porcentaje de acierto.

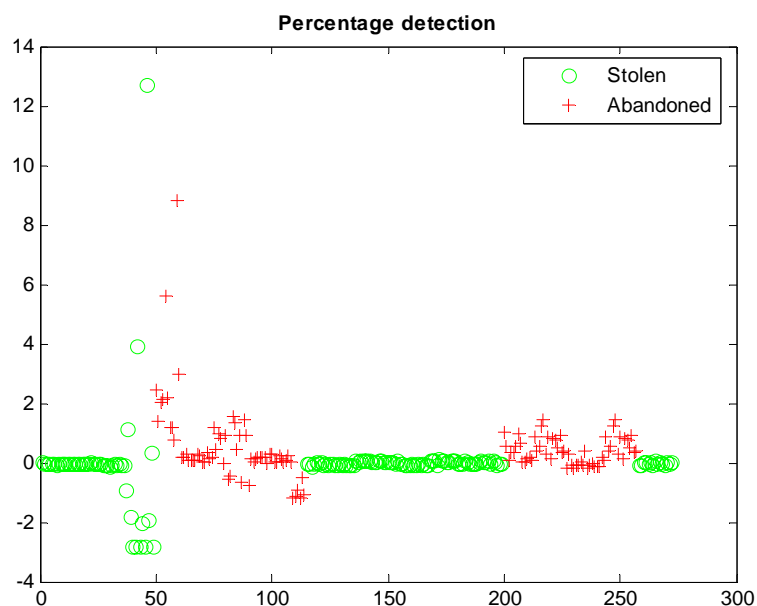


Figura 6-5 Resultados obtenidos para el detector basado en comparación de histogramas

En cambio, en el detector que se muestra en la figura 6.5, es prácticamente imposible poder establecer una distribución para determinar si estamos ante un evento de robo o de abandono.

6.2.3 Fase de test

A continuación, se muestran los resultados obtenidos en % de acierto para los detectores individuales, y los esquemas de fusión implementados. Decir que dichos esquemas de fusión se han implementado seleccionando los dos mejores detectores en base a sus porcentajes de acierto.

Los resultados obtenidos para ambas secuencias se muestran a continuación:

	CROMA (1496 muestras)		REALES (287 muestras)	
	Errores	%	Errores	%
CH	16	1.07	15	5.23
GH	348	23.3	92	32.1
GL	197	13.2	39	13.6
GRAD	196	13.1	63	22
CON	718	48	123	42.9
CONB	630	42.1	105	36.6
SH	465	31.1	119	41.5
Sistema base	283	18.9	51	17.8
Media Arit	144	9.63	12	4.18
Multiplicación	17	1.14	6	2.09
Máximo	144	9.63	14	4.88
Mínimo	14	0.936	6	2.09
Media Geom.	792	52.9	164	57.1
Bayes	124	8.29	6	2.09
Dempster-Shafer	144	9.63	22	7.67

Tabla 6-7. Resultados de los detectores y esquemas de fusión implementados

Analizando la tabla 6.7, se observa que los dos mejores detectores individuales son el detector basado en color y el basado en el gradiente bajo. Estos han sido los detectores utilizados para implementar los esquemas de fusión de este PFC (salvo el del sistema base).

Destacar que los errores para los detectores individuales son menores en porcentaje para las secuencias CROMA, pero los esquemas de fusión presentan mejores resultados en cuanto a porcentaje para las secuencias REALES que para las secuencias CROMA. Esto se puede deber a que las secuencias croma presentan resultados bastante más óptimos, tanto en el caso de robo como en el caso de abandono (al ser secuencias mas sencillas de analizar).

Los resultados obtenidos por los detectores basados en la comparación de contornos son bastante imprecisos (como se puede ver en la figura 6.5), esto se cree que puede ser debido a un problema de diseño de sus algoritmos en el caso de que en la región a ajustar no contenga un objeto.

A su vez, analizando los esquemas de fusión, se observa como el esquema basado en la media geométrica obtiene resultados bastante pobres, en cambio el resto obtienen resultados mas o menos uniformes, destacando los esquemas de fusión basados en la multiplicación, y en la regla del mínimo.

Destacar que el esquema de fusión de Dempster-Shafer tiene un inconveniente, y es que en algunos casos, la suma de la evidencia de robo y de la de abandono es mayor que 1, por lo tanto nos queda la incertidumbre negativa. Esto se puede solucionar normalizando dicha suma a 1.

Pero llama la atención los buenos resultados que se obtienen mediante el histograma de color, ya que en muchos casos los esquema de fusión empeoran dichos resultados. Esto se debe en gran medida a la diferencia de color que suele haber entre un objeto estático y los colores del fondo de las imágenes.

Como consecuencia, para la implementación de un sistema de vídeo-seguridad, se escogería el modelo Bayesiano para detectar objetos abandonados o robados.

7 Conclusiones y trabajo futuro

7.1 Conclusiones

En este PFC se ha desarrollado un estudio de los sistemas de vídeo-seguridad orientados a la detección de objetos abandonados o robados en entornos densamente poblados.

Para ello, se procedió a realizar un exhaustivo estudio del estado del arte, donde se identificaron las diferentes partes de un sistema de vídeo-seguridad, y se distinguieron tres partes claramente diferenciadas: Detección de regiones estáticas, clasificación objeto-persona y discriminación robo-abandono. En este PFC se decidió centrarse en las partes de detección de regiones estáticas y en los discriminadores de robo-abandono.

Para la detección de regiones estáticas, se realizó una clasificación del estudio del arte, y se seleccionaron las técnicas más destacadas de la literatura referentes a la detección de regiones estáticas basándose en la técnica de segmentación frente-fondo.

Se observó que la mayor diferencia se encontraba en las técnicas utilizadas para post-procesar las máscaras de *foreground* obtenidas, así como en el número de modelos de fondo implementados.

Tras esto, se procedió a implementar los métodos más relevantes, y se realizó una comparativa entre ellos.

A su vez, se ha diseñado un algoritmo nuevo para detectar regiones estáticas, bastante robusto ante oclusiones, cambios de iluminación y escenas muy pobladas de gente en movimiento.

En la parte orientada a discriminar entre robo y abandono, se han implementado siete detectores, que devuelven una evidencia de robo y una evidencia de abandono. Aprovechando la diversidad de resultados de dichos detectores, se decidió implementar varios esquemas de fusión con el fin de mejorar dichas evidencias de robo y de abandono obtenidas por los detectores individuales.

Para ello, se escogieron los dos mejores detectores, y se propusieron ocho esquemas de fusión.

Tras esto, se procedió a la evaluación de la detección de regiones estáticas, donde se han realizado dos clasificaciones: una clasificación objetiva (basada en un *ground truth*) y otra subjetiva basada en la selección de algunas propiedades de dichos detectores.

Para llevar a cabo la evaluación, se utilizaron dos *datasets* bastante conocidos, el del PETS2006 y AVSS2007. Uno corresponde a un escenario sencillo (PETS2006), y el otro *dataset* corresponde a un escenario complejo (AVSS2007), con mucha gente en movimiento y bastantes oclusiones.

Se puede ver como los resultados para el escenario sencillo son bastante buenos para casi todas las aproximaciones, pero estos resultados empeoran cuando pasamos a escenas con mucha gente andando y bastantes oclusiones.

Se observa que el algoritmo implementado obtiene mejores resultados experimentales que los algoritmos existentes en la literatura, en términos de precisión a la hora de detectar regiones estacionarias.

Para el análisis de los detectores de robo y abandono, se utilizaron dos tipos de secuencias: unas secuencias de entrenamiento grabadas por el grupo de investigación para diversos tipos de fondo (CROMA), y otras secuencias pertenecientes a imágenes reales. Tras combinar los mejores detectores en un esquema de fusión, se ha decidido que el modelo Bayesiano es el más fiable a la hora de detectar objetos abandonados o robados, fusionando en dicho modelo las evidencias que se obtienen de los detectores basados en color y en el gradiente bajo (con porcentajes de, aproximadamente, un 95% de acierto).

Sin embargo, algunas de estas operaciones añaden un coste computacional algo elevado al sistema, y no debemos olvidar que esta herramienta ha de trabajar en tiempo real, por lo tanto también nos van a interesar algoritmos con bajo coste computacional. El algoritmo creado para detectar regiones estáticas cumple dicha condición, y el esquema de fusión Bayesiano también.

7.2 Trabajo futuro

La detección de objetos en entornos muy poblados sigue siendo un reto de cara a futuras líneas de investigación.

En lo que respecta a este PFC, se cree que se puede seguir mejorando el algoritmo de detección de regiones estáticas, mejorando las oclusiones, y las zonas de paso a analizar.

La etapa de detección de persona-objeto es algo que tiene mucho trabajo por desarrollar, y será de gran ayuda para reducir la cantidad de falsos positivos introducidos en el sistema.

A su vez, el estudio de más esquemas de fusión puede aportar nuevas ideas para mejorar los esquemas ya existentes en este documento. Adicionalmente, se propone el desarrollo de un modelo de ajuste de contornos basado en la técnica de *snakes* descrita con el objetivo de buscar la deformación deseada cuando no existe un objeto en la imagen actual (robo) o el fondo de la escena (abandono).

Otra futura línea de investigación se puede basar en localizar a la persona que roba o abandona el objeto, mediante el uso de un tracking robusto.

REFERENCIAS

- [1] Plataniotis, K.N.; Regazzoni, C.S. “Visual-centric Surveillance Networks and Services”. *IEEE Signal Processing Magazine*, 22(2):12-15, 2005.
- [2] Ferrando, S.; Gera, G.; Regazzoni, C. “Classification of Unattended and Stolen Objects in Video-Surveillance System”. *Proc. of AVSS 2006*, pp. 21-27.
- [3] Beynon, M. “Detecting abandoned packages in a multi-camera video surveillance system”, *Proc. of AVSS 2003*, pp. 221–228.
- [4] Li, L. “Statistical Modeling of Complex Backgrounds for Foreground Object Detection”, *IEEE Trans. on Image Processing*, 13 (11):1459-1472, 2004.
- [5] Piccardi, M. “Background subtraction techniques: a review”, *IEEE Trans. on System, Man and Cybernetics.vol.4*, 3099-3104, 2004.
- [6] Spagnolo, P. et al. “An Abandoned/Removed Objects Detection Algorithm and Its Evaluation on PETS Datasets”. *Proc. of AVSS 2006*, pp. 17-21.
- [7] Porikli, F.; Ivanov, Y.; Haga, T. “Robust Abandoned Object Detection Using Dual Foregrounds”, *Journal on Advances in Signal Processing*, art. 30, 11 pp., 2008.
- [8] Liao,H-H.; Chang,J-Y.; Chen, L-G. “A localized Approach to abandoned luggage detection with Foreground –Mask sampling”, *Proc. of AVSS 2008*, pp. 132-139.
- [9] Huwer, S.; Niemann, H. “Adaptive Change detection for Real-Time Surveillance Applications”, *Proc. of Visual Surveillance 2000*, pp. 37-46.
- [10] Cheng, S.; Xingzhi Luo; Bhandarkar, S.M. “A Multiscale parametric background Model for Stationary Foreground Object Detection” , *Proc. of Motion and Video Computing 2007*, 8 pp.
- [11] Mieziako, R.; Pokrajac, D. “Detecting and Recognizing Abandoned Objects in Crowded Environments”, *Proc. of Computer Vision System 2008*, pp. 241-250.
- [12] Mathew, R.; Yu, Z.; Zhang, J. “Detecting new stable objects in surveillance video”, *Proc. of Multimedia Signal Processing 2005*, pp. 1-4.
- [13] Guler, S.; Silverstein, J.A.; Pushee, I.H. “Stationary objects in multiple object tracking”, *Proc. of AVSS 2007*, pp. 248-253.
- [14] Guler, S.; Farrow, K. “Abandoned Object detection in crowded places” , *Proc. of PETS 2006*, June 18-23.
- [15] Bhargava, M.; Chen, C-C. Ryoo, M. S.; Aggarwal, J. K. “Detection of abandoned objects in crowded environments”, *Proc. of AVSS 2007*, pp. 271-276.
- [16] Bird, N. “Real time, online detection of abandoned objects in public areas”, *Proc. of Robotics and Automation 2006*, pp. 3775-3780.
- [17] Porikli, F. “Detection of temporarily static regions by processing video at different frame rates”, *Proc. of AVSS 2007*, pp. 236-241.
- [18] Martínez, J.; Herrero, J.; Orrite, C. “Automatic Left luggage Detection and Tracking using a Multi-camera UKF”, *Proc. of PETS 2006*, pp 59-66.
- [19] Stauffer, C; Grimson, W. E. L. “Adaptive background mixture models for real-time tracking,” *Proc. of CVPR 1999*, vol. 2, pp. 2246-2252.
- [20] San Miguel J.C.; Martinez, J.M. “Robust unattended and stolen object detection by fusing simple algorithms”, *Proc. of AVSS 2008*, pp. 18-25.
- [21] Tian, Y. “Robust and efficient foreground analysis for real-time video surveillance”, *Proc. of CVPR 2005*, pp. 1182-1187.

- [22] Fernandez-Carbajales, V. et al. "Robust people detection by fusion of evidence from multiple methods", Proc of Int Workshop on image Analysis for Multimedia Interactive Services, pp55-58, 2008
- [23] Wren, Azarbayejani, Darrell, Pentland, 1997. Pfinder: Real-time tracking of the human body. PAMI.
- [24] Cavallaro, A., Steiger O., Ebrahimi T., "Semantic Video Analysis for Adaptive Content Delivery and Automatic Description", IEEE Transactions of Circuits and Systems for Video Technology, 15(10):1200-1209, Octubre 2005.
- [25] Elgammal A, Duraiswami R, Harwood D, Davis LS. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. Proc. IEEE Jul. 2002; 90(7): 1151-1163.
- [26] Cristani M., Bicego M., Murino V. Multi-level background initialization using Hidden Markov Models. In First ACM SIGMM Int. workshop on Video surveillance 2003; 11-20.
- [27] Tiburzi F., Escudero M., Bescós J., Martínez J.M., "A Ground-truth for Motion-based Video-object Segmentation", IEEE International Conference on Image Processing, ICIP'2008, Workshop on Multimedia Information Retrieval: New Trends and Challenges, pp.17-20, San Diego (USA), Oct. 2008.

ANEXOS

ANEXO A: Publicaciones derivadas de este proyecto

Publicaciones en congresos internacionales:

- Bayona, A.; San Miguel, J.C; Martínez, J.M: "Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques", Proc. of IEEE AVSS2009, pp. 25-30, (ISBN: 0-7695-3718-4)

Conferencia IEEE AVSS: Clasificación CORE A

Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques

Álvaro Bayona, Juan Carlos SanMiguel, José M. Martínez
Video Processing and Understanding Lab

Escuela Politécnica Superior, Universidad Autónoma de Madrid, SPAIN

E-mail: {Alvaro.Bayona, Juancarlos.SanMiguel, JoseM.Martinez}@uam.es

Abstract

In several video surveillance applications, such as the detection of abandoned/stolen objects or parked vehicles, the detection of stationary foreground objects is a critical task. In the literature, many algorithms have been proposed that deal with the detection of stationary foreground objects, the majority of them based on background subtraction techniques. In this paper we discuss various stationary object detection approaches comparing them in typical surveillance scenarios (extracted from standard datasets). Firstly, the existing approaches based on background-subtraction are organized into categories. Then, a representative technique of each category is selected and described. Finally, a comparative evaluation using objective and subjective criteria is performed on video surveillance sequences selected from the PETS 2006 and i-LIDS for AVSS 2007 datasets, analyzing the advantages and drawbacks of each selected approach.

1. Introduction

Currently the automatic analysis of video surveillance sequences has become an area of very active research in response to the increasing demand of security issues in public areas [1][2]. Video surveillance systems aim to provide automatic analysis tools that may help the supervisor personnel in order to focus his/her attention when a dangerous or strange event takes place.

In this context, the detection of stationary objects is receiving a special attention because it is a critical analysis stage in applications like the detection of abandoned objects or parked vehicles frequently used in the surveillance of public areas. Additionally, the recognition of stationary objects in crowded unconstrained contexts is a challenging task. Issues related to occlusions (by moving or stationary objects), appearance variations (e.g., color composition, shape) as people move relatively to the camera, lighting changes, speed of the objects and the density of moving objects in the scene should be taken into account.

In the detection of stationary foreground objects, background-subtraction based approaches have become the most popular choice due to the common use of fixed cameras and the assumption that the illumination changes in the scene are gradual [3][4][10]. However, some works exists [5] that don't use this approach for analyzing static images.

In this paper we present a comparative evaluation of the stationary foreground object detection approaches based on background-subtraction[5][6]. Firstly, these approaches are hierarchically organized into different categories. Then, representative approaches of these categories are selected and discussed. Finally, the selected approaches are tested and compared identifying their advantages and drawbacks in two typical scenarios for video surveillance: the detection of abandoned objects and parked vehicles. This comparison is provided through an objective and a subjective evaluation of the selected approaches.

The remainder of this paper is structured as follows: section 2 describes the classification for background-subtraction based approaches, section 3 describes the ones selected to be compared, section 4 shows experimental results and section 5 closes the paper with some conclusions.

2. Classification of background-subtraction based methods for stationary object detection

In this section we describe the proposed classification for categorizing the stationary foreground detection approaches based on background-subtraction techniques (see Fig. 1). As most of the existing approaches incorporate some kind of tracking analysis in their system, we have decided to exclude the use of tracking from the criteria used in the classification.

Firstly, we have divided the existing approaches in two categories depending on their use of one or more background subtraction models.

Depending on the use of the foreground maps computed in the background subtraction analysis, one-model based approaches can be classified in:

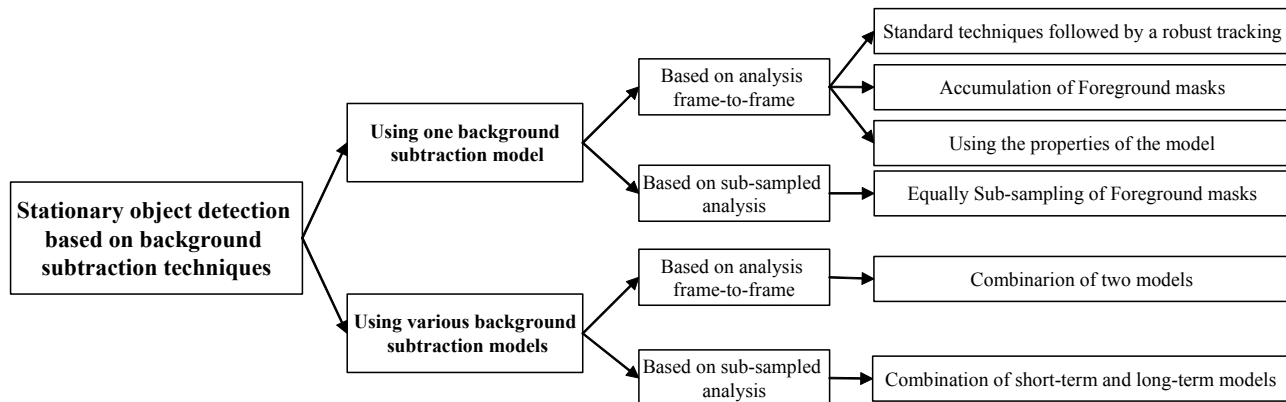


Figure 1: Classification of the background-subtraction based methods for stationary object detection

- *Based on frame-by-frame analysis.* This category describes the methods that employ typical background subtraction techniques followed by another type of analysis. Depending on this analysis stage we have the following approaches:
 - o Based on the use of standard background techniques followed by another analysis stage (e.g., tracking) [3][10][18][19][20].
 - o Based on the accumulation of foreground masks computed frame-by-frame [9][10][14].
 - o Based on the properties of the background subtraction model used [11][12][13][15].
- *Based on a sub-sampled analysis.* These approaches try to detect stationary objects by analyzing the video sequence at different framerates [8][17].

Existing approaches combining two or more background subtraction models have been less investigated. However, a classification based on the processing framerate can be done as follows:

- *Based on frame-by-frame analysis.* In this category, we have methods that combine the different properties using two or more background subtraction techniques [21].
- *Based on a sub-sampled analysis.* These approaches detect stationary objects by analyzing the video sequence with various background subtraction methods at different framerates [7].

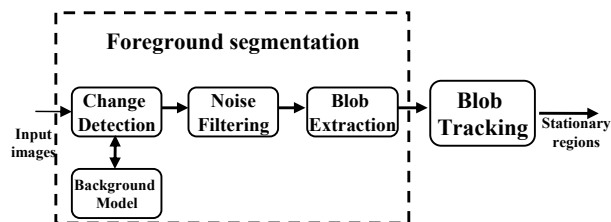


Figure 2: Stationary object detection procedure using [19]

3. Overview of selected approaches

In this section we describe the background-subtraction based approaches selected to be compared. We have chosen one representative approach for each previously described category attending to its implementation cost and detailed description in the papers studied.

3.1. One-model based

3.1.1 Based on frame-by-frame analysis

A. Based on the use of standard background techniques

As an example of this category, we have decided to implement the approach described in [19]. In this approach a typical background segmentation stage with a Gaussian Mixture Model (GMM) is proposed, followed by a blob tracking analysis stage. This tracking analysis is based on finding the correspondence between the blobs identified in two consecutive frames. Some rules, as colour, shape, distance or object size are used in this module to perform the tracking analysis. Fig. 2 depicts the processing scheme followed in the selected approach.

B. Based on the accumulation of foreground masks

As an example of this category, we have decided to implement the approach described in [14]. It is based on the accumulation of foreground masks to compute a confidence map to indicate the presence of stationary foreground objects.

In this algorithm, an intermediate image, $S(x,y)$, where each pixel indicates the confidence of the complexity image pixel belonging to a stationary object, is computed.

Initially, all the pixels of the confidence image are set to 0 being updated at every frame analysis. This update is based on the foreground masks obtained by previously applying a background subtraction stage. Two counter maps are calculated to update the confidence image: an increment counter $C(x,y)$, used when a pixel doesn't fit

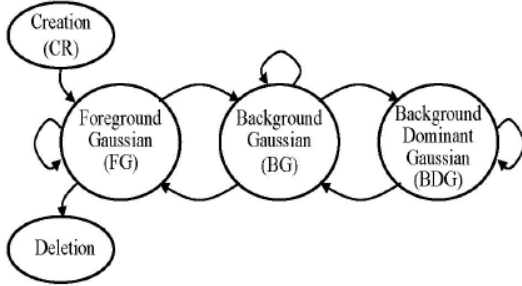


Figure 3: Transition states of distributions in GMM [12]

with the background (it is labelled as foreground), and a decrement counter $D(x,y)$, used when a pixel fits with the background. The confidence image S is updated every frame pixel by pixel using these counters and depending on the defined *Framerate* of the video sequence and stationary object detection time t . Finally, S is thresholded to obtain a binary mask indicating the presence of stationary foreground objects.

C. Based on properties of the model

As an example of this category, we have decided to implement the approach described in [12]. It is based on the use of the GMM for detecting foreground objects and inspecting the properties of that model to detect stationary objects.

The stationary object detection is based on the observation of the transition states between the new Gaussian distributions created (for the new foreground pixels detected) and their transition to the dominant background state. Three Gaussians distributions are used in the GMM model resulting in the transition state diagram shown in Fig. 3. This approach describes a set of necessary conditions and corresponding observations on the transition state diagram to detect stationary objects imposing time stability, spatial stability and enough distribution weight constraints.

3.1.2 Based on a sub-sampled analysis

As an example of this category, we have selected the approach described in [8]. It is based on sampling the foreground-mask computed (see Fig 4). Firstly, a background subtraction stage based on modelling each pixel with a Gaussian distribution is performed. Additionally, a weight term is added for each pixel to take into account the gradual intensity change in image resolution or image deformations. Then, a number of sample foreground masks are taken from the last frames analyzed. The authors used 6 samples to determine the foreground mask (S) multiplying 6 binary foreground masks. Each active pixel of S (value equal to 1) indicates that a pixel has been foreground in the last 30 seconds and it presents a high probability of being stationary.

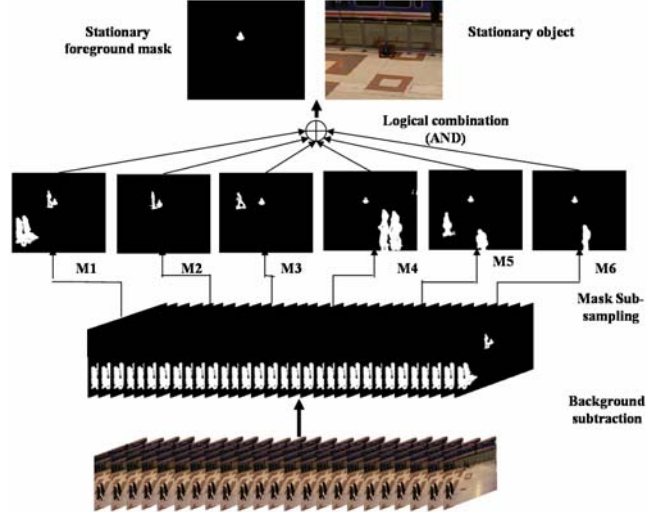


Figure 4: Foreground-mask subsampling procedure [8]

3.2. Two-model based

As an example of this category, we have decided to implement the approach described in [7]. In this method, a detection stage based on the application of two background subtraction methods at different framerate is proposed. The two models are based on the GMM employing one model for short-term detection (updating it every frame) and another for long-term detection (updating it every n frames). Short-term background is adapted faster and the scene changes are introduced more quickly on it. On the other hand, long-term background is adapted to the changes of the scene at a lower learning rate. Then, the foreground masks of the two models are computed at every frame and a combination of them is performed as shown in Fig. 5.

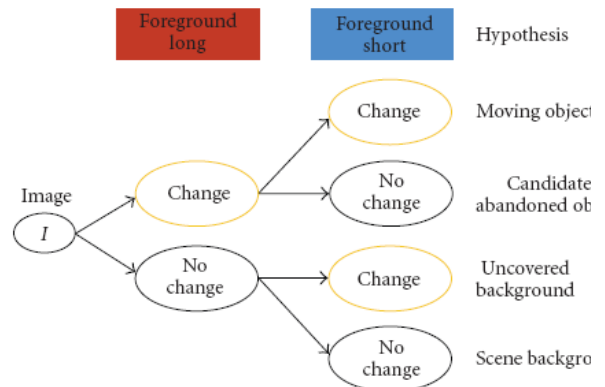


Figure 5: Combination of two background subtraction methods to detect stationary regions [7]

4. Performance evaluation comparison

In this section, experimental results of the selected approaches are presented and compared. The system has been implemented using the OpenCV image processing library (<http://sourceforge.net/projects/opencv/>). Tests were executed on a Pentium IV with a CPU frequency of 2.8 GHz and 1GB RAM. Additionally, we have decided to manually annotate the stationary regions corresponding to persons in the results obtained by the selected approaches in order to discard them. Then, stationary regions corresponding to objects have been evaluated with the ground-truth data provided in the datasets.

4.1. Experimental data

For the evaluation, we have selected two different types of sequences. Firstly, test sequences from the PETS 2006 dataset (available at <http://pets2006.net/>) have been selected as a simple scenario. This scenario presents lower stationary foreground extraction complexity, middle foreground object density and the speed of the scene objects is low. Secondly, test sequences from the i-LIDS dataset for AVSS2007 (available at <http://www.elec.qmul.ac.uk/staffinfo/andrea/avss2007.html>) have been selected as a complex scenario. This crowded scenario presents higher stationary foreground extraction complexity, higher foreground object density and the speed of the scene objects is variable ranging from low to high velocities.

4.2. Performance evaluation comparison metrics

The results of the selected approaches have been objectively and subjectively compared. Additionally, we have considered a foreground object as stationary if it

remains static during 30 seconds.

As objective measures, we have selected the accuracy (in time) of the foreground object detection, the duration of the alarm/detection and Precision/Recall measures for the stationary foreground detection task.

For the subjective evaluation, we have selected 5 subjective measures. As the difficulty of the detection in crowded scenarios is determined by the occlusions between objects, moving object density and velocity, large perspective distortion, or the similarity in appearance of different people, we have decided to select a set of characteristics relative to these aspects (see Table 3). To evaluate them, we have ranged the characteristics between very low and very high.

4.3. Performance evaluation comparison

4.3.1 Objective evaluation

The results obtained from the experiments are summarized in Table 1 and 2 for, respectively, the simple and complex scenarios. Additionally, the precision and recall results of the stationary foreground detection task are presented in Table 3.

As it can be observed in Table 1, the overviewed approaches obtained similar results in the PETS2006 test sequences (simple scenario). In this scenario, stationary foreground objects are located in a low-dense area where the number of occlusions, lighting changes and moving objects is low. Additionally, colour dissimilarity between the stationary objects and the background allows a perfect identification by the background subtraction procedure.

On the other hand, as it can be observed in Table 2, results are more heterogeneous in complex (crowded) sequences. On these scenes, the presence of occlusions,

Table 1: Objective evaluation of the selected approaches for the simple scenario

PETS2006 Sequences	Ground Truth		Approach 1[19]		Approach 2[14]		Approach 3[12]		Approach 4[8]		Approach 5[7]	
	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration
S1 T1 C3	1:52	0:08	1:52	0:08	1:52	0:08	1:55	0:06	1:52	0:08	1:52	0:08
S4 T1 C3	1:43	0:19	1:44	0:18	1:43	0:19	1:45	0:18	1:43	0:19	1:43	0:19
S5 T1 C3	1:26	0:26	1:26	0:26	1:26	0:26	1:22	0:30	1:26	0:26	1:26	0:26

Table 2: Objective evaluation of the selected approaches for the complex scenario

AVSS2007 Sequences	Ground Truth		Approach 1[19]		Approach 2[14]		Approach 3[12]		Approach 4[8]		Approach 5[7]	
	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration	Start Time	Alarm Duration
AB_Easy	2:05	0:36	2:03	0:37	2:06	0:37	2:14	0:38	2:05	0:36	2:06	0:37
AB_Medium	1:54	0:37	1:56	0:34	1:57	0:41	2:07	0:38	1:55	0:36	1:52	0:37
AB_Hard	2:11	0:56	2:12	0:54	2:17	0:59	2:27	0:57	2:11	0:56	2:15	0:50

Table 3: Precision (P) and Recall (R) results for the stationary object detection task

Approach	PETS2006 Sequences		AVSS2007 Sequences	
	P	R	P	R
Approach 1 [19]	0.05	1	0.01	1
Approach 2 [14]	0.6	1	0.1	1
Approach 3 [12]	0.5	1	0.03	1
Approach 4 [8]	0.75	1	0.33	1
Approach 5[7]	0.37	1	0.05	1

moving objects, and moving people is higher than in simple sequences. Approaches based on sub-sampling or accumulation (like [8][14]) obtain better results than methods based on simple background segmentation and tracking .

In Table 3, we can observe that all the annotated events are detected for both defined scenarios (Recall equals to 1). These results can be achieved by selecting the optimum parameters of each approach for each scenario. On the other hand, the Precision results show that the selected approaches perform the stationary foreground detection task with medium and very low accuracy for simple and complex scenarios respectively. This is due to the high amount of moving objects in the complex scenario. Approaches based on sub-sampling produce the best results due to the logical combination stage applied that eliminates most of the false positives.

Analyzing both scenarios, we can see that when increasing the number of mobile objects in the scene and the number of occlusions, results are less exact, but always with a minimal variance around ground truth results within a few seconds.

4.3.2 Subjective evaluation

The results of the subjective evaluation are reported in Table 4 for the simple and complex scenarios.

Table 4: Subjective evaluation of the selected approaches

Selected Approaches	Foreground Extraction accuracy	Tolerance to occlusions	Noise introduced	Computational Load	Parameter Adjustment Difficulty
Approach 1 [19]	Very Low	Very Low	Very High	Medium	Very Low
Approach 2 [14]	High	Medium	Low	Medium	High
Approach 3 [12]	Low	High	High	High	Very High
Approach 4 [8]	High	Very High	Very Low	Medium	Medium
Approach 5 [7]	Medium	High	Medium	Low	Low

In general, analyzing the results obtained, we can observe that occlusions are completely removed from the final mask on sub-sampling approaches [8] and partially on the others [7][14]. In the simple scenario, all approaches (except [19]) present good results, but in complex scenarios (like the ones in the AVSS 2007 sequences), only [20] and [8] present good detection results.

Regarding the noise introduced, sub-sampling methods obtain better results because the noise (supposed to be statistically independent) is sub-sampled reducing its effect. In classic approaches, there is considerable noise in the mask and it should be removed in the following stages with different procedures (like noise filtering).

Computational cost is directly related with the number of background subtraction stages applied and the base technique used (GMM, KDE ...). Approaches with light background-subtraction stages (like [8]) perform the detection faster than the others. The addition of additional analysis stages (like [19][14]) obviously increases the computational load. For example in [7], after the light stationary object detection, a light tracking stage is performed resulting in a low computational cost. Finally, approaches that apply various background subtraction stages add a low or high computational cost if, respectively, they analyze samples or the whole video sequence.

Parameter adjustment is difficult in approaches like [12] due to the need of fine tuning for inspecting the detection model. Nevertheless, more basic approaches (like [19]) do not present a high difficulty in the adjustment because slight errors in parameter settings can be corrected in the following analysis stages (using technologies like noise or shadow filtering). Sub-sampling approaches present medium difficulty because the sub-sampled time is the critical parameter of the scheme and it depends on the velocity of the objects and the framerate of the scene under analysis. Two model based approaches do not have so much parameterization problems except the time to update the models in sub-sampled schemes.

5. Conclusions

This paper has presented a comparative evaluation of representative approaches based on background subtraction techniques for detecting stationary foreground objects. Firstly, the existing approaches have been classified into different categories. Then, representative approaches have been selected and described. Finally, an objective and subjective comparison has been performed.

Main conclusions of the study are the following. The results of the objective evaluation show that the detection of stationary foreground objects in simple scenarios is achieved with high accuracy in all the tested approaches. On the other hand, detection results for the complex scenario are more heterogeneous. Approaches based on sub-sampling schemes or accumulation of foreground masks assure the best results. In these type of scenarios the sub-sampling rate is a critical parameter (depending on the velocity of the moving objects) to determine the stationary objects. Subjective evaluation shows that sub-sampling based approaches obtain the best results on accuracy in the stationary foreground mask presenting a high tolerance to occlusions (frequently in complex scenes) and intermediate difficulty in parameters adjustment. In the case of complex scenarios, approaches based on standard background-subtraction techniques present the worst performance in all subjective measures due to the difficulty of analyzing the foreground masks in the following analysis stages (e.g., tracking). On the other hand, the difficulty in the adjustment of the parameters in these approaches is very low. Approaches based on the properties of the background-subtraction model used present low accuracy in the extracted mask due to the difficulty of the parameters adjustment phase. Concluding, for general-purpose stationary object detection, sub-sampling based approaches obtain the best results adding a low computational cost in the overall system.

6. Acknowledgments

This work has partially supported by the Cátedra UAM-Infoglobal ("Nuevas tecnologías de vídeo aplicadas a sistemas de video-seguridad"), the Spanish Administration agency CDTI (CENIT-VISION 2007-1007), by the Spanish Government (TEC2007-65400 SemanticVideo), by the Comunidad de Madrid (S-050/TIC-0223-ProMultiDis), by the Consejería de Educación of the Comunidad de Madrid, and by The European Social Fund.

7. References

- [1] Plataniotis, K.N.; Regazzoni, C.S. "Visual-centric Surveillance Networks and Services". *IEEE Signal Processing Magazine*, 22(2):12-15, 2005.
- [2] Ferrando, S.; Gera, G.; Regazzoni, C. "Classification of Unattended and Stolen Objects in Video-Surveillance System". *Proc. of AVSS 2006*, pp. 21-27.
- [3] Beynon, M. "Detecting abandoned packages in a multi-camera video surveillance system", *Proc. of AVSS 2003*, pp. 221-228.
- [4] Li, L. "Statistical Modeling of Complex Backgrounds for Foreground Object Detection", *IEEE Trans. on Image Processing*, 13 (11):1459-1472, 2004.
- [5] Piccardi, M. "Background subtraction techniques: a review", *Proc. of SMC 2004*, vol. 4, pp. 3099-3104.
- [6] Spagnolo, P. et al. "An Abandoned/Removed Objects Detection Algorithm and Its Evaluation on PETS Datasets". *Proc. of AVSS 2006*, pp. 17-21.
- [7] Porikli, F.; Ivanov, Y.; Haga, T. "Robust Abandoned Object Detection Using Dual Foregrounds", *Journal on Advances in Signal Processing*, art. 30, 11 pp., 2008.
- [8] Liao, H.-H.; Chang, J.-Y.; Chen, L.-G. "A localized Approach to abandoned luggage detection with Foreground -Mask sampling", *Proc. of AVSS 2008*, pp. 132-139.
- [9] Huwer, S.; Niemann, H. "Adaptive Change detection for Real-Time Surveillance Applications", *Proc. of Visual Surveillance 2000*, pp. 37-46.
- [10] Cheng, S.; Xingzhi Luo; Bhandarkar, S.M. "A Multiscale parametric background Model for Stationary Foreground Object Detection", *Proc. of Motion and Video Computing 2007*, 8 pp.
- [11] Mieziako, R.; Pokrajac, D. "Detecting and Recognizing Abandoned Objects in Crowded Environments", *Proc. of Computer Vision System 2008*, pp. 241-250.
- [12] Mathew, R.; Yu, Z.; Zhang, J. "Detecting new stable objects in surveillance video", *Proc. of Multimedia Signal Processing 2005*, pp. 1-4.
- [13] Guler, S.; Silverstein, J.A.; Pushee, I.H. "Stationary objects in multiple object tracking", *Proc. of AVSS 2007*, pp. 248-253.
- [14] Guler, S.; Farrow, K. "Abandoned Object detection in crowded places", *Proc. of PETS 2006*, June 18-23.
- [15] Bhargava, M.; Chen, C.-C.; Ryoo, M. S.; Aggarwal, J. K. "Detection of abandoned objects in crowded environments", *Proc. of AVSS 2007*, pp. 271-276.
- [16] Bird, N. "Real time, online detection of abandoned objects in public areas", *Proc. of Robotics and Automation 2006*, pp. 3775-3780.
- [17] Porikli, F. "Detection of temporarily static regions by processing video at different frame rates", *Proc. of AVSS 2007*, pp. 236-241.
- [18] Martínez, J.; Herrero, J.; Orrite, C. "Automatic Left luggage Detection and Tracking using a Multi-camera UKF", *Proc. of PETS 2006*, pp 59-66.
- [19] Stauffer, C; Grimson, W. E. L. "Adaptive background mixture models for real-time tracking," *Proc. of CVPR 1999*, vol. 2, pp. 2246-2252.
- [20] San Miguel J.C.; Martínez, J.M. "Robust unattended and stolen object detection by fusing simple algorithms", *Proc. of AVSS 2008*, pp. 18-25.
- [21] Tian, Y. "Robust and efficient foreground analysis for real-time video surveillance", *Proc. of CVPR 2005*, pp. 1182-1187.

PRESUPUESTO

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... 2.000 €
- Alquiler de impresora láser durante 9 meses 50 €
- Material de oficina 150 €
- Total de ejecución material 2.200 €

2) Gastos generales

- 16 % sobre Ejecución Material 352 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material 132 €

4) Honorarios Proyecto

- 640 horas a 15 € / hora 9600 €

5) Material fungible

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

6) Subtotal del presupuesto

- Subtotal Presupuesto..... 12060 €

7) I.V.A. aplicable

- 16% Subtotal Presupuesto 1929.6 €

8) Total presupuesto

- Total Presupuesto..... 13989,6 €

Madrid, Septiembre de 2009

El Ingeniero Jefe de Proyecto

Fdo.: Álvaro Bayona Gómez
Ingeniero Superior de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un sistema basado en detectar objetos abandonados o robados 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.

PROYECTO FIN DE CARRERA

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

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

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

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

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

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

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

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

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

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

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

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la

PROYECTO FIN DE CARRERA

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.

PROYECTO FIN DE CARRERA

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.