

UNIVERSIDAD AUTÓNOMA DE MADRID  
ESCUELA POLITÉCNICA SUPERIOR



# Integración y evaluación de sistemas de robo-abandono de objetos en vídeo-seguridad

Jorge Gómez Vicente  
Tutor: Juan Carlos San Miguel Avedillo

**-PROYECTO FIN DE CARRERA-**

Ingeniería de Telecomunicación

Departamento de Tecnología Electrónica y de las Comunicaciones  
Escuela Politécnica Superior  
Universidad Autónoma de Madrid  
Julio 2016



# Integración y evaluación de sistemas de robo-abandono de objetos en vídeo-seguridad

Jorge Gómez Vicente.

Tutor: Juan Carlos San Miguel Avedillo

Ponente: José María Martínez Sánchez



Video Processing and Understanding Lab  
Departamento de Tecnología Electrónica y de las Comunicaciones  
Escuela Politécnica Superior  
Universidad Autónoma de Madrid  
Julio 2016

Trabajo parcialmente financiado por el gobierno español bajo el proyecto  
TEC2014-53176-R (HA-Video)





# Resumen

En este trabajo se propone un sistema configurable de detección de robo-abandono de objetos en vídeo-seguridad que integra las técnicas más relevantes en cada una de sus etapas. Se presenta una formalización del problema, seguida de una descripción de las distintas etapas de análisis requeridas para la detección. En primer lugar, se realiza un estudio del estado del arte para conocer los problemas actuales en la materia. Después el trabajo se centra en integrar los algoritmos más recientes y relevantes de la literatura en cada una de las etapas del sistema y diseñar las interfaces necesarias para su ejecución de manera secuencial. Por último se evalúan y comparan distintas configuraciones del sistema a nivel de detección de regiones estáticas y clasificación robo-abandono sobre un conjunto de secuencias de vídeo heterogéneo. La detección de eventos de robo y abandono es muy dependiente de los parámetros que modulan la absorción de elementos del frente estático por parte del modelo de fondo. Puesto que la discriminación robo-abandono de objetos es la última etapa del sistema de análisis de vídeo, sus resultados se ven afectados por la acumulación de errores de las etapas anteriores.

## **Palabras clave:**

análisis de vídeo, detección de abandono de objetos, detección de robo de objetos, detección de objetos estáticos, video-seguridad, video-vigilancia



# Abstract

This work proposes a configurable abandoned-stolen object detection system in security-video that integrates the most relevant techniques in each one of its stages. A formalization of the problem is presented, followed by a description of the different analysis stages required for the detection. Firstly, this work analyses the state of the art to know the present date problems about the matter. Secondly, the work focuses on the integration of the most recent and relevant algorithms of the literature in every single phase of the system. It also designs the necessary interfaces for its execution in a sequential order. To conclude, the different configurations of the system regarding the detection of static regions as well are evaluated and compared, while it is classified as abandoned-stolen about a compound of heterogeneous videos sequences. Abandoned/stolen object event detection depends on the parameters that modulate the absorption of the blobs from the static foreground by the background model. The results of abandoned/stolen detection stage are affected by the propagation of errors in the earlier stages as it is the last stage of video analysis system.

## **Keywords:**

video analysis, detection of abandoned objects, detection of stolen objects, detection of stationary objects, , video-surveillance.



# Agradecimientos

*En primer lugar, quiero agradecer a mi tutor Juan Carlos San Miguel su inestimable ayuda, su predisposición y atención siempre que la he necesitado durante la realización de este proyecto así como los consejos y herramientas ajenas al proyecto que me ha enseñado y que me serán de gran utilidad en el futuro.*

*A mis profesores, José María Martínez, por haberme brindado la oportunidad de realizar este proyecto en VPU-Lab y a Jesús Bescós, mi tutor académico, por su orientación durante los primeros años de carrera.*

*A los doctorandos de VPU-Lab Fulgencio, Diego, Marcos y Rafael por su atención y ayuda prestada en el laboratorio siempre que he acudido a ellos con mis dudas.*

*Quiero dar las gracias a todos los compañeros y profesores con los que he tenido la suerte y el honor de compartir mi formación académica en la universidad ya que de una forma u otra todos han contribuido a mi formación ya no como ingeniero, sino como persona.*

*Quiero agradecer a mis amigos y familiares todo el apoyo y cariño recibido por su parte durante los años que he necesitado para terminar la carrera, ya que sin ellos no lo habría logrado.*

*Por último, me gustaría dedicar este trabajo a la memoria de mi padre, fallecido el 3 de junio de 2015. Él es el responsable de que yo haya cursado unos estudios superiores apoyandome anímicamente y económicamente hasta el último de sus días. Siempre fue un referente y un modelo a seguir como persona y su mayor orgullo en la vida era ver a su hijo terminar la carrera y haberle proporcionado las herramientas necesarias para labrarse su propio futuro.*

*Jorge Gómez Vicente*

*Julio 2016*



# Índice general

<b>Resumen</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Agradecimientos</b>	<b>ix</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	3
1.3. Estructura de la memoria . . . . .	4
<b>2. Estado del arte</b>	<b>5</b>
2.1. Introducción . . . . .	5
2.2. Segmentación de objetos de interés . . . . .	8
2.2.1. Descripción . . . . .	8
2.2.2. Principales técnicas . . . . .	9
2.3. Detección de regiones estáticas . . . . .	11
2.3.1. Descripción . . . . .	11
2.3.2. Principales técnicas . . . . .	14
2.4. Clasificación de regiones estáticas . . . . .	18
2.5. Discriminación entre objetos robados o abandonados . . . . .	19
2.5.1. Descripción . . . . .	20
2.5.2. Principales técnicas . . . . .	20
2.6. Datasets disponibles . . . . .	22
2.7. Métrica y evaluación . . . . .	25
2.7.1. Parámetros de rendimiento . . . . .	25
2.7.2. Resultados . . . . .	27
2.8. Herramientas utilizadas . . . . .	28
2.8.1. OpenCV . . . . .	28
2.8.2. VirtualDub . . . . .	28
2.8.3. ViPER . . . . .	29
<b>3. Sistema base</b>	<b>31</b>
3.1. Trabajo previo . . . . .	32
3.2. Arquitectura general y configuración del sistema diseñado . . . . .	33

3.2.1.	Arquitectura general del sistema . . . . .	33
3.2.2.	Configuración del sistema . . . . .	34
3.3.	Algoritmos integrados en las distintas etapas del sistema . . . . .	35
3.3.1.	Técnicas de sustracción de fondo . . . . .	37
3.3.2.	Técnicas de detección de regiones estáticas . . . . .	39
3.3.3.	Técnicas de detección de personas . . . . .	40
3.3.4.	Técnicas de discriminación robo-abandono . . . . .	43
3.4.	Implementación del sistema . . . . .	46
3.4.1.	Captura de la secuencia de vídeo-vigilancia y extracción de parámetros relevantes. . . . .	46
3.4.2.	Sustracción de fondo. MyBGSSelector y MyDualBGSSelector .	46
3.4.3.	Detección de regiones estáticas. MySFGDSelector . . . . .	48
3.4.4.	Extracción de blobs. BlobExtractor . . . . .	51
3.4.5.	Detección de personas. MyPEDESTRIANDETECTORSelector	52
3.4.6.	Personas estáticas y en movimiento. Funciones Solape y DrawS- taticPeople . . . . .	53
3.4.7.	Definición de una bloblist de objetos. Funciones CompareBlobs y DefineObjectBlobList . . . . .	55
3.4.8.	Clasificación robo-abandono. MyCLSSFRSelector . . . . .	56
3.4.9.	Anotación de resultados. EventController . . . . .	58
3.5.	Definición de ground-truths . . . . .	58
<b>4.</b>	<b>Resultados experimentales</b>	<b>63</b>
4.1.	Secuencias analizadas . . . . .	63
4.2.	Métricas de evaluación seleccionadas . . . . .	64
4.3.	Evaluación de las técnicas de detección de regiones estáticas . . . . .	65
4.3.1.	Evaluación de detección de regiones estáticas mediante sub- muestreo de máscaras de frente . . . . .	67
4.3.2.	Evaluación de detección de regiones estáticas mediante acumu- lación de máscaras de frente . . . . .	69
4.3.3.	Comparativa y conclusiones de los resultados obtenidos . . . . .	72
4.4.	Evaluación de las técnicas de clasificación robo-abandono . . . . .	73
4.4.1.	Evaluación de clasificación robo-abandono mediante contraste de color de píxel . . . . .	74
4.4.2.	Evaluación de clasificación robo-abandono mediante gradiente alto . . . . .	75
4.4.3.	Comparativa y conclusiones de los resultados obtenidos . . . . .	75
<b>5.</b>	<b>Conclusiones y trabajo futuro.</b>	<b>79</b>
5.1.	Conclusiones. . . . .	79
5.2.	Trabajo futuro. . . . .	80
	<b>Bibliografía</b>	<b>83</b>
	<b>A. Presupuesto</b>	<b>91</b>





# Índice de figuras

1.1. Ejemplo de detecciones de robo y abandono de objetos. . . . .	2
2.1. Diagrama de bloques de sistema de vídeo-seguridad. . . . .	7
2.2. Algoritmo de sustracción de fondo. . . . .	8
2.3. Algoritmo de detección de regiones estáticas. . . . .	13
2.4. Técnicas de detección de regiones estáticas. . . . .	14
2.5. Detección de personas. . . . .	19
2.6. Diagrama de discriminación robo-abandono . . . . .	20
2.7. Discriminación robo-abandono . . . . .	21
2.8. Frames de datasets públicos . . . . .	24
3.1. Diagrama de bloques del sistema propuesto. . . . .	32
3.2. Guía de configuración del sistema 1 . . . . .	36
3.3. Guía de configuración del sistema 3 . . . . .	36
3.4. Diagrama de algoritmo Subsampling . . . . .	40
3.5. Diagrama algoritmo History Images . . . . .	41
3.6. Diagrama algoritmo DBM . . . . .	41
3.7. Histograma de color . . . . .	43
3.8. Pixel Color Contrast . . . . .	44
3.9. Gradientes . . . . .	45
3.10. Resultados módulo BGS . . . . .	49
3.11. Resultados módulo SFGD . . . . .	51
3.12. Extracción de blobs . . . . .	52
3.13. Función Solape . . . . .	54
3.14. Función DrawStaticPeople . . . . .	55
3.15. Diagrama CompareBlobs . . . . .	56
3.16. Diagrama DefineObjectBlobList . . . . .	56
3.17. Resultados sistema completo . . . . .	58
3.18. Anotación GT . . . . .	61
3.19. Frames ground truth . . . . .	61
4.1. Errores del modelo de fondo. . . . .	65
4.2. Errores derivados de sombras. . . . .	66
4.3. Errores derivados de reflejos. . . . .	66
4.4. Errores de la máscara de frente generada. . . . .	66
4.5. Error IMBS ACC . . . . .	71

4.6. Clasificadores frente a oclusiones . . . . .	76
4.7. Clasificadores frente errores de fondo . . . . .	77

# Índice de tablas

2.1. Tabla de técnicas de segmentación de objetos de interés. . . . .	10
2.2. Tabla de aproximaciones de detección de regiones estáticas de interés. . . . .	15
2.3. Tabla de robusteces de aproximaciones de detección de regiones estáticas de interés. . . . .	16
2.4. Datasets más utilizados. . . . .	25
2.5. Comparativa de resultados detección de regiones estáticas . . . . .	27
2.6. Comparativa de resultados discriminación robo-abandono . . . . .	28
3.1. Algoritmos BGS integrados . . . . .	39
3.2. Algoritmos SFGD integrados . . . . .	41
3.3. Detectores de personas integrados . . . . .	42
3.4. Clasificadores robo/abandono integrados . . . . .	45
3.5. Módulo de BGS . . . . .	48
3.6. Modulo de SFGD . . . . .	51
3.7. Módulo detección de personas . . . . .	53
3.8. Módulo de discriminación robo/abandono . . . . .	58
4.1. Retos presentes en las secuencias evaluadas . . . . .	64
4.2. Anotaciones secuencias evaluadas . . . . .	64
4.3. Resultados Subsampling - FASOM . . . . .	67
4.4. Resultados Subsampling - LOBSTER . . . . .	68
4.5. Resultados Subsampling - IMBS . . . . .	69
4.6. Resultados ACC - FASOM . . . . .	70
4.7. Resultados ACC - LOBSTER . . . . .	70
4.8. Resultados ACC - IMBS . . . . .	71
4.9. Comparativa detección regiones estáticas . . . . .	73
4.10. Resultados PCC . . . . .	74
4.11. Resultados Gradiente Alto . . . . .	75
4.12. Comparativa detección discriminación . . . . .	76



# Capítulo 1

## Introducción

### 1.1. Motivación

Hoy en día, se experimenta un crecimiento en la demanda de sistemas de vídeo-seguridad como consecuencia del aumento de la preocupación por la seguridad global, especialmente para su aplicación en las zonas comunes, como aeropuertos, estaciones, trenes subterráneos, entrada a los edificios y eventos masivos debido a la gran cantidad de gente que se concentra en esos lugares y los potenciales riesgos de seguridad que ello implica [1][2]. Estos sistemas deben proporcionar herramientas de análisis automáticas con el objetivo de ayudar al personal de seguridad a centrar su atención cuando se encuentran ante una situación compleja.

Tradicionalmente, la tarea de supervisión se ha llevado a cabo por operadores humanos que tienen que analizar simultáneamente la información de diferentes cámaras. Por lo que cabe esperar una reducción de eficiencia a medida que los operadores tienen que procesar mayores cantidades de información visual generada por estas cámaras. Por esta razón, el análisis automático de secuencias de vídeo en tiempo real está emergiendo como una solución para ayudar a los operadores a centrar su atención en eventos específicos relacionados con la seguridad. En esta situación, la detección de objetos abandonados o robados en lugares públicos así como la detección de vehículos estacionados se han convertido en uno de los temas de investigación más prometedores. Esta detección tiene como objetivo proporcionar un control permanente de la información capturada por la cámara de manera que se puedan tomar las medidas oportunas y llevar a cabo las acciones apropiadas.

En general, la detección de objetos abandonados o robados se logra mediante el desarrollo de un sistema de análisis de vídeo que comprende las siguientes etapas: segmentación de fondo, detección de regiones estacionarias, discriminación



Figura 1.1: Ejemplo de aplicaciones de detecciones de robo y abandono de objetos. Izquierda: secuencia perteneciente al dataset CAVIAR. Derecha: secuencia perteneciente al dataset Outdoor Scene.

persona/objeto de las regiones estáticas detectadas y por último clasificación robado/abandonado de los objetos. En primer lugar, los objetos en movimiento (primer plano) se diferencian del fondo de la escena en la etapa de segmentación de fondo. Entonces, las regiones estáticas, que son aquellos elementos del primer plano que permanecen inmóviles en la secuencia de vídeo durante un tiempo determinado, se detectan mediante el análisis de objetos de primer plano en el tiempo. Cada región estática detectada se clasifica por tipo (persona, grupo de personas, equipaje, ...). Y adicionalmente, para aquellas regiones clasificadas como objetos estacionarios, se lleva a cabo un análisis con el fin de determinar si el objeto ha sido abandonado o robado de la escena. La Figura 1.1 muestra los resultados obtenidos de la aplicación de un sistema de análisis de vídeo para la detección de robo-abandono de objetos.

Comúnmente se requiere que las aplicaciones desarrolladas operen correctamente en entornos complejos tales como multitudes o situaciones que cambian con rapidez. Cada etapa en el sistema tiene diferentes retos que afectan a su rendimiento. Los cambios en las condiciones de iluminación y los fondos multimodales pueden dar lugar a un resultado erróneo de segmentación de fondo, lo que dificulta la detección de objetos de interés. Por otra parte, en entornos de alta densidad de objetos, las regiones estáticas pueden no ser detectadas correctamente debido a la velocidad de los objetos y el elevado número de oclusiones que tienen lugar. La clasificación de las regiones estáticas detectadas puede verse afectada por la variabilidad de la apariencia del objeto durante la secuencia de vídeo y puesto que los objetos potencialmente robados o abandonados pueden tener tanto una forma como un color completamente arbitrario, no pueden aplicarse métodos específicos de reconocimiento de objetos. En último lugar, cabe destacar que es deseable el análisis en tiempo real en aplicaciones de apoyo al personal de seguridad/monitorización por lo que los algoritmos utilizados en cada una de las etapas del sistema deberían ser de baja complejidad.

La motivación de este PFC surge a raíz de los avances realizados por el VPU Lab en las distintas etapas de sistemas de robo/abandono y la ausencia de un sistema

completo compuesto por ellas con el fin de contribuir a la mejora de los resultados de los sistemas actuales para llevar a cabo análisis automático de alto nivel. A su vez, como incentivo, se ha de tener en cuenta la creciente necesidad en temas de seguridad tal y como refleja el aumento de la demanda de sistemas de análisis automático de secuencias de vídeo-seguridad.

## 1.2. Objetivos

El objetivo de este PFC es el desarrollo de un sistema de detección de robo/abandono configurable y su posterior evaluación en entornos de operación de complejidad variable (multitudes, operación continuada,...).

El objetivo mencionado puede ser desglosado en varios objetivos generales:

1. Estudio del estado del arte

Antes de cualquier proyecto hay que realizar una revisión del estado del arte en el campo tratado. En este caso, habrá que centrar dicha revisión en el campo de reconocimiento automático de eventos de robo y abandono de objetos en secuencias de vídeo. En especial, se realizará un estudio teórico de las etapas de sustracción de fondo, detección de regiones estáticas, detección de personas y clasificación robo/abandono, las aproximaciones más relevantes de la literatura, los datasets disponibles y la librería de tratamiento de imágenes OpenCV.

2. Análisis de las técnicas disponibles en el VPU-Lab

Se realizará un análisis de las tecnologías existentes en el VPU-Lab (<http://www-vpu.eps.uam.es>) relacionadas con las distintas etapas del robo/abandono de objetos. (sustracción de fondo, detección de regiones estáticas, detección de personas y clasificación robo/abandono) así como las distintas tecnologías alternativas recientes.

3. Diseño e implementación del sistema

Se llevará a cabo la implementación de un sistema completo de detección de robo y abandono de objetos mediante la integración de las tecnologías existentes en el VPU-Lab así como la implementación de alternativas y algoritmos existentes. Para ello, se describirán las interfaces entre las distintas etapas del sistema y se definirán las reglas del sistema que permiten la detección de objetos de interés. En paralelo se detallarán los ground truths de los datasets disponibles, es decir, las detecciones correctas que se deben detectar en un caso ideal en las secuencias de vídeo analizadas, con el fin de obtener los resultados de interés en la posterior evaluación del sistema.

#### 4. Evaluación del sistema propuesto

Se elaborará una evaluación exhaustiva y comparativa del sistema implementado mediante la integración de las tecnologías existentes en el VPU-Lab frente a las tecnologías alternativas sometiendo el sistema a los dataset actuales representativos de condiciones reales (reflejos/sombras, multitudes, operación continuada, etc) de los que se dispone y se obtendrán las tasas de detecciones correctas y fallidas frente a los ground truths definidos. A continuación se confrontarán los resultados obtenidos con los referentes al estado del arte, de manera que se puedan extraer las conclusiones pertinentes.

### 1.3. Estructura de la memoria

La memoria del proyecto se divide en los siguientes capítulos:

- Capítulo 1. Introducción: introducción, motivación y objetivos del proyecto.
- Capítulo 2. Estado del arte: sistemas de detección de robo y abandono de objetos (segmentación de objetos de interés, detección de regiones estáticas y clasificación de las mismas) y evaluación ( métricas de evaluación existentes, datasets disponibles y resultados actuales de las técnicas pertenecientes al estado del arte. Opencv, Viper-GT, Viper-PE, VirtualDub y DiVA.
- Capítulo 3. Diseño e implementación del sistema propuesto: arquitectura del sistema, técnicas incluidas en el sistema e implementación del sistema.
- Capítulo 4. Evaluación del sistema propuesto: métricas seleccionadas, elección de secuencias sobre las que evaluar el sistema, experimentos realizados y resultados obtenidos.
- Capítulo 5. Conclusiones y trabajo futuro.
- Referencias y anexos.

## Capítulo 2

# Estado del arte

En este capítulo se estudia el estado del arte relacionado con los sistemas de detección de robo y abandono de objetos en secuencias de vídeo-vigilancia. Este estudio se encuentra dividido en las siguientes secciones: introducción los sistemas de vídeo-vigilancia en la sección 2.1, definición de la segmentación de objetos de interés y técnicas más relevantes de la literatura en la sección 2.2, descripción de la detección de regiones estáticas y las técnicas de mayor impacto en la sección 2.3, clasificación de las regiones estáticas detectadas entre persona y objeto en la sección 2.4, discriminación robo-abandono de los objetos de interés en la sección 2.5, datasets públicos disponibles para la evaluación de sistemas de vídeo-vigilancia en la sección 2.6, parámetros de rendimiento y resultados de los sistemas de vídeo-vigilancia disponibles en la literatura en la sección 2.7 y por último una recopilación y descripción de las herramientas utilizadas durante el desarrollo de este proyecto en la sección 2.8.

### 2.1. Introducción

Tal y como se describe en [3], originalmente, los sistemas de vídeo-vigilancia estaban formados por un circuito cerrado de televisión (CCTV). Es decir, un conjunto de cámaras que capturaban las secuencias de vídeo que eran transmitidas hasta los paneles de visualización. Tradicionalmente, aunque la señal de vídeo fuera inicialmente digital, la transmisión se realizaba mediante cable coaxial utilizando una señal analógica de vídeo compuesto que llegaba hasta la sala de visualización que normalmente estaba compuesta por una matriz de monitores donde podían verse las imágenes de todas las cámaras. Estos sistemas formarían la primera generación de sistemas de vídeo-vigilancia.

El salto a la *segunda generación* de sistemas de vídeo-vigilancia se produce al aprovechar que la imagen es capturada en formato digital para realizar un procesamiento sobre ella. Ahora la imagen no se convierte a formato analógico, sino que es recibida directamente por un ordenador que la procesa y ejecuta sobre las imágenes algún algoritmo de análisis. La mayor parte de los algoritmos utilizados, son algoritmos de detección de eventos en tiempo real que ayudan al supervisor del sistema a detectar eventos producidos en las escenas que graban las cámaras facilitando su labor. Estos sistemas aunque son más avanzados que los de la primera generación tienen el problema de que para gestionar un gran número de cámaras y procesar algoritmos sobre las señales de vídeo que proceden de las cámaras hacen falta potentes equipos informáticos conectados a las cámaras.

En la *tercera generación* se produce un avance en cuanto al diseño de los sistemas. Se comienza a plantear una arquitectura distribuida frente a la tradicional centralizada. También se remarca la importancia de que el sistema debe ser escalable al estar en constante aumento el número de cámaras presentes en los entornos vídeo-vigilados. Este aumento del número de fuentes de vídeo abre las puertas a algoritmos multicámara que utilicen simultáneamente imágenes de diferentes dispositivos de captura. La descentralización producida en los sistemas de esta generación permitirá distribuir las diferentes etapas de procesamiento entre diferentes elementos de la red, pudiendo de esta manera ejecutar algoritmos más complejos en tiempo real al ser procesada cada parte del algoritmo en una máquina diferente.

El análisis automático de eventos en secuencias de vídeo-seguridad es un área de investigación muy activa. En este contexto, la detección de objetos abandonados o robados en lugares públicos es uno de los eventos que más atención ha captado recientemente convirtiéndose en uno de los temas de investigación más prometedores. Esta detección requiere el desarrollo de sistemas complejos para detectar, seguir y analizar los objetos de interés. 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.

Los sistemas propuestos en la literatura para el análisis de secuencias de vídeo-vigilancia emplean típicamente un esquema como el mostrado en la Figura 2.1. La primera etapa consiste en la localización de aquellas regiones donde sucede algo rele-

vante. Para la detección de dichas regiones, existen muchas aproximaciones basadas en la técnica de segmentación frente-fondo (background subtraction) que proporcionan una máscara binaria de frente con las regiones deseadas 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. Actualmente, los modelos de caracterización de los píxeles del fondo basados en el método de mezcla de Gaussianas (MoG) son frecuentemente utilizadas debido a que modelan el fondo ante posibles cambios de iluminación en la imagen, secuencias multimodales, objetos moviéndose lentamente, y el ruido introducido por la cámara.

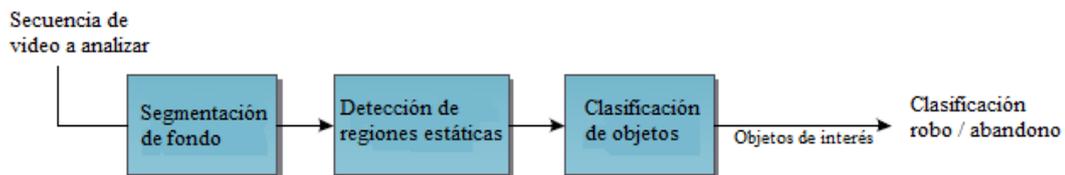


Figura 2.1: Diagrama de bloques habitual en un sistema de vídeo-seguridad para la detección de robo y abandono de objetos.

A continuación se aplica una técnica para determinar qué regiones de interés permanecen estáticas. 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 discriminación entre persona/objeto y una clasificación entre robo o abandonado. Existe una gran variedad de métodos que tratan de distinta manera las máscaras de frente para realizar las detecciones de regiones estáticas tales como acumulaciones o sub-muestreo. Otras emplean varios modelos de fondo y en consecuencia varias máscaras de frente y también hay métodos que aprovechan propiedades del modelo de sustracción de fondo. La información obtenida de la detección de regiones estáticas se emplea en aplicaciones de alto nivel donde se busca, entre otros, clasificaciones persona-objeto o reconocimiento de las actividades de la escena (p. ej., robo o abandono), valiéndose en ocasiones de etapas de seguimiento de objetos. A continuación se detallan las etapas mencionadas.



Figura 2.2: Resultados obtenidos tras aplicar un algoritmo de sustracción de fondo a la secuencia de vídeo AVSS 2007 abandoned baggage easy. De izquierda a derecha se presentan un imagen de la secuencia, el modelo de fondo calculado por el algoritmo y la máscara de frente obtenida por el mismo.

## 2.2. Segmentación de objetos de interés

### 2.2.1. Descripción

El primer paso en muchas aplicaciones de visión por ordenador consiste en la localización de objetos de interés dentro de una escena mediante la distinción entre los píxeles que pertenecen al primer plano o foreground (objetos estáticos o en movimiento) de los píxeles que componen el fondo o background de la escena. 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. Como se ha explicado, la mayoría de técnicas emplean la sustracción de fondo (background subtraction) como medio para obtener las regiones de interés. Este proceso consiste en obtener una máscara binaria de regiones de frente mediante algún tipo de comparación entre la imagen bajo análisis ( $I_t$ ) y un modelo de fondo ( $B_t$ ):

$$FG_t = f(I_t, B_t)$$

De entre los métodos expuestos en la literatura, los más utilizados son aquellos que operan a nivel de píxel. Algunos enfoques modelan el fondo mediante el promedio de los frames analizados o a partir del cálculo de la mediana de cada píxel en un buffer de los últimos  $N$  frames previos. Muchos autores modelan cada píxel con una función de densidad de probabilidad donde los píxeles con bajas probabilidades son más propensos a pertenecer a los objetos en movimiento de primer plano. La Figura 2.2 muestra el modelo de fondo y la máscara de frente generados por un algoritmo de sustracción de fondo.

Un buen algoritmo de segmentación debe tener una tasa de actualización, que es el tiempo tras el cual se modifica el modelo de fondo generado incorporando los

cambios producidos en el escenario, que le permita ser capaz de detectar objetos en movimiento, y debe realizarlo mediante el uso de un bajo coste computacional y en tiempo real. Esta etapa es crítica, ya que los resultados del resto de módulos de un sistema de detección de robo y abandono de objetos dependerán de los resultados de esta primera etapa de segmentación de fondo y en consecuencia se condiciona enormemente el rendimiento del posterior análisis de regiones estáticas. Debido a que las secuencias de vídeo no son ideales, 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 ya que los algoritmos de detección de regiones en movimiento desarrollados hasta la fecha son sensibles a los cambios repentinos de iluminación, las condiciones ambientales (noche, lluvia, niebla...), la actualización del fondo de la escena, sombras y reflejos, los efectos de camuflaje (similitud fotométrica de objeto y fondo), el ruido o los fondos multimodales [4][5]. Una segmentación de objetos de interés incorrecta tiene un impacto negativo en el análisis posterior. Los cambios de iluminación o reflejos suelen dar lugar a falsos positivos, que posteriormente se traducen en regiones estáticas detectadas como objetos abandonados o robados. Si las sombras proyectadas por los objetos que constituyen el frente se detectan como parte de dichos objetos, un discriminador basado en características de color puede producir una clasificación errónea, ya que la zona detectada compartirá rasgos característicos referentes a su color con el fondo. Si los objetos del primer plano permanecen estáticos, con el tiempo se convierten en parte del modelo de fondo. Por lo tanto, la tasa de actualización del modelo de fondo debe tenerse en cuenta a la hora de decidir el tiempo mínimo que una región debe permanecer estática para ser detectada como abandonada o robada.

### 2.2.2. Principales técnicas

En la Tabla 2.1 se resumen las técnicas de segmentación de objetos de interés más utilizadas en la literatura y se presentan sus características y robusteces frente a los retos más comunes que se suelen encontrar en las secuencias de vídeo-seguridad de las que se dispone en la literatura.

**GMM (Gaussian Mixture Model):** Modelo paramétrico basado en la caracterización de cada píxel de la imagen mediante una mezcla de gaussianas [6][7][8][9][10][11][12][13][14]. Cada píxel de la imagen está representado por la suma ponderada de K gaussianas (siendo K una variable a definir, usualmente entre 3 y 5). Cada Gaussiana se describe en términos de medias y desviaciones estándar de cada componente de color del píxel.

Técnica	Tipo	Robustez			Nivel de análisis	Característica
		Cambios de iluminación	Oclusiones	Sombras y reflejos		
GMM [6][7][8][9][10][11][12][13][14]	BGS	Si	Si	Si	Pixel	Color, Brillo y textura
HFSM + VSM [15][16]	*	Si	Si	No	Pixel y región	Intensidad, área, forma y velocidad
ESD [17]	BGS	Si	Si	No	Bordes de píxel	Bordes
3D-SOBS(SOM) [18]	BGS	Si	No	Si	Pixel	Brillo
KDE [19]	BGS	Si	Si	No	Pixel y región	profundidad
FD [20]	BGS	Si	No	Si	Pixel y región	Color, bordes y forma
Color + Struct. Diff. [21]	*	Si	No	Si	Pixel y región	Color y bordes

Tabla 2.1: Comparativa de las principales técnicas de segmentación de objetos de interés más relevantes de la literatura. Aquellas técnicas de segmentación de objetos que no basan en segmentación de fondo han sido señaladas mediante el valor \*.

### **HFSM + SVM (Hierarchical Finite StateMachine + Support Vector Machine):**

Modelo no paramétrico basado en la caracterización de cada píxel de la imagen mediante una máquina de estados jerárquica de tres capas [15][16]. La capa de píxel se compone de tres estados de píxel (fondo b, primer plano f y s estática); su estado inicial es b porque cada píxel es considerado como píxel de fondo durante el modelado de la imagen de fondo. La capa de píxel determina el estado de cada píxel mediante el uso de características de intensidad y de duración de tiempo.

**ESD (Edge-Segment Distribution):** Modelo no paramétrico basado en la caracterización de cada píxel de la imagen mediante la información extraída de sus bordes [17]. Para modelar el fondo se extraen los bordes de cada uno de los frames de entrenamiento y se genera el mapa de bordes. A continuación se crea la distribución de bordes acumulando los bordes extraídos y se genera un mapa estadístico. Se define un mapa de distribución estadística que permite generar el modelo de fondo óptimo.

**SOM (Self-Organized Model):** Modelo paramétrico basado en la definición de un mapa neuronal para cada píxel de la imagen mediante n vectores ponderados. Dada una imagen de N filas y P columnas, se obtendrá un mapa neuronal de 3 dimensiones  $N \times P \times n$  compuesto por los mapas neuronales asignados a cada mapa de la imagen [18][22][23]. El modelo de fondo es generado mediante la superposición de capas compuestas por los vectores con mayor ponderación de cada píxel durante un número de frames de entrenamiento determinado.

**KDE (Kernel Density Estimation):** Modelo no paramétrico basado en la caracterización de cada píxel de la imagen mediante la estimación de su función densidad de probabilidad realizando un análisis de K frames consecutivos de la secuencia de vídeo objetivo considerados como frames de entrenamiento [19]. El modelo de fondo generado sera más próximo al modelo de fondo óptimo a costa de aumentar el número de frames de entrenamiento.

**FD (Imagen Difference):** Técnica basada en el modelado de la máscara de frente mediante la diferencia de dos frames [20]. Se asume un imagen como background y se detectan los objetos en movimiento mediante la resta aritmética del background con los sucesivos frames en cada instante  $t$  dando como resultado la imagen diferencia que mostrará cierta intensidad para los píxeles que hayan variado de un imagen a otro.

**Color + Structure Difference:** Técnica basada en la caracterización de cada píxel de la imagen mediante la información extraída de su color y la estructura de los píxeles vecinos [21]. Con el fin de generar un mapa de foreground preliminar, es necesario eliminar los píxeles que podrían ser considerados como fondo con certeza. Para cada píxel, primero se calcula su diferencia de color entre el imagen y el modelo de fondo actual. La diferencia de color es simplemente el máximo absoluto de la diferencia del valor de píxel en cada uno de los tres canales de color. A continuación se calcula la diferencia estructural en un área alrededor del píxel comparando las estructuras del imagen y del background inicial. Tras calcular la diferencia de color y la diferencia estructural, se calcula la diferencia híbrida como producto de ambas diferencias y por último se umbraliza la imagen diferencia obtenida.

En la actualidad, el método de mezcla gaussianas (GMM), es el más utilizado debido a su capacidad para manejar variaciones de fondo, tales como cambios en la iluminación gradual, sombras y movimientos repetitivos tales como por ejemplo, árboles meciéndose.

## 2.3. Detección de regiones estáticas

### 2.3.1. Descripción

Tras el proceso de segmentación de objetos de interés, el siguiente paso es determinar que que objetos de la escena han permanecido estáticas. La detección de regiones estáticas es muy importante por dos razones: debe mejorar la calidad de los datos inicialmente detectados en la segmentación de objetos de interés, 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. Para la detección de dichas regiones, encontramos en la literatura muchas aproximaciones basadas en la técnica de segmentación frente-fondo (background subtraction). Hemos clasificado dichas aproximaciones en dos grupos en función de las características de la secuencia de vídeo bajo estudio:

- *Background subtraction & tracking*: La combinación de sustracción de fondo y seguimiento de objetos es aplicable en aquellas secuencias de vídeo que presentan baja densidad de objetos permitiendo así realizar el seguimiento de las regiones estáticas deseadas. El proceso de seguimiento puede implicar a cualquier objeto móvil presente en la escena, sin reconocer de qué objeto se trata (seguimiento precategórico) o el seguimiento de un objeto específico del cual se tiene un modelo predefinido (seguimiento categórico) [24][25][26].
- *Background subtraction & analysis*: Esta técnica es utilizada en aquellas secuencias que presentan alta densidad de objetos en la escena. Una vez realizada la segmentación de fondo se procede a la aplicación de un algoritmo que analiza la persistencia del primer plano a lo largo del tiempo. La información obtenida en esta etapa es crucial ya que marcará el funcionamiento de técnicas posteriores [27].

La Figura 2.3 muestra un ejemplo de técnica de detección de regiones estáticas

Las aproximaciones basadas en seguimiento de objetos presentan muchas dificultades al analizar escenas que contemplen escenarios altamente concurridos y por lo tanto las expectativas a cerca de su precisión se consideran bajas. Además, el coste computacional se verá incrementado a medida que aumente el número de objetos a seguir por el sistema. Por estas razones, en este proyecto nos centraremos únicamente en aquellas aproximaciones de la literatura basadas en el análisis de los resultados obtenidos en la etapa de segmentación de objetos de interés sin utilizar seguimiento. La Figura 2.3 muestra un ejemplo de los resultados obtenidos por una técnica de detección de regiones estáticas basada en el análisis de los resultados de la sustracción de fondo.

En [28] se propone una clasificación de estas técnicas de detección de regiones estáticas no basadas en el seguimiento de objetos, en función del tipo de análisis realizado y mediante el número de modelos de fondo utilizados tal y como muestra la Figura 2.4.

Atendiendo a esta clasificación tenemos que aquellas aproximaciones basadas en un único modelo de fondo se dividen 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



Figura 2.3: Resultados obtenidos tras aplicar un algoritmo de sustracción de fondo sobre la secuencia AVSS 2007 abandoned baggage easy y una técnica de detección de regiones estáticas durante más de 10 segundos sobre la máscara binaria de frente obtenida como resultado de la sustracción de fondo. De izquierda a derecha se presentan un imagen de la secuencia, la máscara de frente obtenida por el algoritmo de sustracción de fondo y la máscara de frente estática obtenida por el algoritmo de detección de regiones estáticas.

pos-procesado de la máscara de foreground seguido a su vez de alguna otra etapa de análisis. Esto se conoce como aproximaciones clásicas.

- Basados en la acumulación de máscaras de foreground. Dicha acumulación se realiza imagen a imagen y con ella se puede modelar una máscara final de foreground, de donde se extraen las regiones estáticas.
- 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.
- 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.

Mientras que aquellas aproximaciones para la detección de regiones estáticas que hacen uso de varios modelos de fondo se clasifican en:

- 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.
- Aproximaciones basadas en el sub-muestreo. Estas aproximaciones detectan regiones estáticas 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.

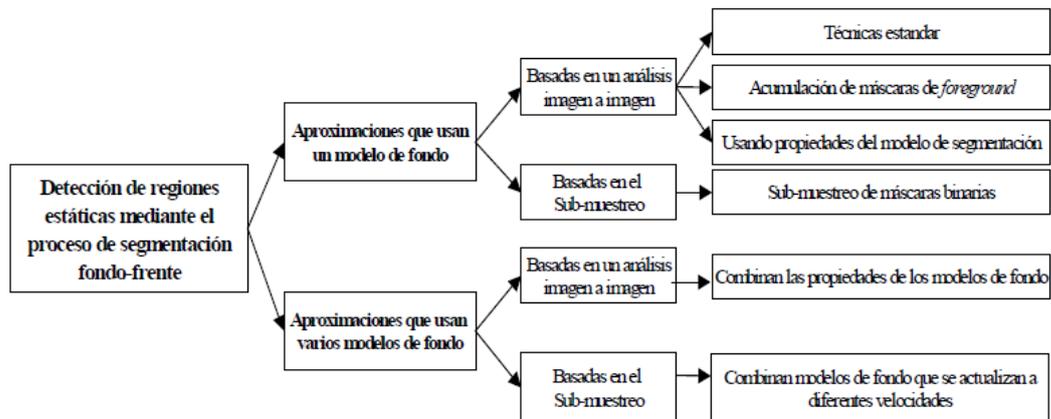


Figura 2.4: Clasificación de las técnicas de detección de regiones estáticas basadas en una etapa previa de sustracción de fondo combinada con un análisis de la máscara de frente obtenida. Imagen extraída de [28]

Como se ha mencionado con anterioridad, las secuencias de vídeo tienen cambios de iluminación o reflejos que suelen dar lugar a falsos positivos en la etapa de segmentación de objetos de interés, que posteriormente se traducen en regiones estáticas detectadas como objetos abandonados o robados. Si las sombras proyectadas por los objetos que constituyen el frente se detectan como parte de dichos objetos, un discriminador basado en características de color puede producir una clasificación errónea, ya que la zona detectada compartirá rasgos característicos referentes a su color con el fondo. Si los objetos del primer plano permanecen estáticos, con el tiempo se convierten en parte del modelo de fondo. Por lo tanto, la tasa de actualización del modelo de fondo debe tenerse en cuenta a la hora de decidir el tiempo mínimo que una región debe permanecer estática para ser detectada como abandonada o robada.

### 2.3.2. Principales técnicas

En las Tablas 2.2 y 2.3 se resumen las técnicas de detección de regiones estáticas de interés más utilizadas en la literatura y se presentan sus características y robustez frente a los retos más comunes que se suelen encontrar en las secuencias de vídeo-seguridad de las que se dispone en la literatura. Además tal y como se observa en la tabla, se pueden clasificar nuevamente las aproximaciones de la literatura en dos categorías en función de si utilizan en el análisis una o varias técnicas combinadas:

- Mono-característica: Esta categoría incluye técnicas en las que, tras determinar las zonas de frente, se aplica un método de detección de regiones estáticas sencillo.

Técnica	BGS utilizado	# modelos de fondo	característica analizada	Nivel de análisis
ACC + Motion Filtering [6]	GMM + ME	1	Máscara de frente	Píxel
ACC [7]	GMM	1	Color, Brillo	Píxel, Región
ACC [17]	ESD	1	Contornos	Contorno de píxel
ACC [15][16]	HFSM + VSM	1	Intensidad, Área, Forma, Movimiento	Píxel, Región, Evento
DBM + ACC [8]	GMM	2	Texturas	Píxel, Región
DBM + ACC [29]	-	-	Algoritmo BGS	Píxel
DBM + ACC [20]	FD	2	Máscara de frente	Píxel, Imagen
DBM + ACC [19]	KDE	2	Intensity, Profundidad, MHI	Píxel
NoobaVSS BGSLib	NoobaVSS	*	Filtrado de movimiento	Texto
GMM + ACC [10] [11]	GMM	1	MHI, SHI	Píxel
GMM + ACC [9]	GMM + FSM	1	Modelo de fondo	Píxel, Región
3D-SOBS + ACC [18]	SOM	1	Máscara de frente	Píxel
GSC [12]	GMM	3	Cambio de estado de Gaussiana	Píxel
GSC + SRH [13]	GMM	3	Cambio de estado de Gaussiana	Píxel
RA + FD [21]	Color+Struc diff	1	Diferencia de valores absolutos	Píxel, Región
Subsampling	-	1	Máscara de frente	Píxel

Tabla 2.2: Comparativa de las principales técnicas de detección de regiones estáticas de interés más relevantes de la literatura.

- Multi-característica: esta categoría engloba métodos que combinan diferentes análisis para detectar regiones estáticas solucionando problemas de la sustracción de fondo. Se pueden encontrar distintos métodos, varias de ellas surgidas de la evolución de métodos mono-característica.

**Motion Filtering + ACC (Acumulador de máscaras de frente):** Técnica multi-característica con un único modelo de fondo que realiza el análisis imagen a imagen. Combina un acumulador de máscaras de frente junto con un filtro de movimiento para añadir robustez frente a escenas que presentan alta densidad de objetos. El acumulador busca detectar regiones estáticas mediante la acumulación de máscaras de frente obtenidas de manera consecutiva. Para ello, el algoritmo construye una imagen intermedia  $S(x, y)$ , donde el valor de cada píxel (entre 0 y 255) determina si dicho píxel es estático, o no. Un valor de 0 indica que el píxel pertenece al fondo de la imagen, y un valor de 255 indica que nos encontramos ante un píxel de una posible región estática. Al comenzar el análisis del vídeo, todos los píxeles de la imagen intermedia  $S$  se inician a 0. Para cada nueva imagen, cada píxel de la imagen  $S$  se actualiza basándose en una máscara de frente, obtenida de una comparación de la imagen bajo análisis  $I(x, y)$  con el modelo de fondo  $B(x, y)$  (algoritmo de sustracción de fondo). Ésta técnica presenta robustez frente a oclusiones de objetos que sufren camuflajes [6].

**ACC:** Técnica mono-característica que hace uso de un único modelo de fondo y realiza un análisis imagen a imagen mediante la acumulación de máscaras de frente [7][15][16][17].

Técnica	Robustez			
	Cambios de iluminación	Fantasmas	Sombras y reflejos	Multitudes
ACC + Motion Filtering [6]	Sí	Sí	Sí	Sí
ACC [7]	Sí	Sí	Sí	No
ACC [17]	Sí	No	Sí	No
ACC [15][16]	Sí	No	No	Sí
DBM + ACC [8]	Sí	Sí	Sí	No
DBM + ACC [29]	Sí	Sí	Sí	No
DBM + ACC [20]	Sí	Sí	Sí	No
DBM + ACC [19]	Sí	Sí	Sí	No
NoobaVSS BGSLib	*	*	*	*
GMM + ACC [10] [11]	Sí	Sí	Sí	No
GMM + ACC [9]	Sí	Sí	Sí	No
3D-SOBS + ACC [18]	Sí	Sí	Sí	No
GSC [12]	Sí	Sí	Sí	No
GSC + SRH [13]	Sí	Sí	Sí	Sí
RA + FD [21]	Sí	Sí	Sí	No

Tabla 2.3: Comparativa de la robustez de las principales técnicas de detección de regiones estáticas de interés más relevantes de la literatura.

**DBM + ACC (Dual background Model):** Técnica multi-característica que combina un acumulador y un análisis basado en el submuestreo de dos modelos de fondo diferentes actualizados a diferentes velocidades. Éste método propone, para la detección de regiones estáticas, la utilización de dos modelos MoG que difieren en la velocidad de actualización del fondo de escena. Un modelo a largo plazo, que se actualiza de forma lenta, es decir, que no incorpora rápidamente a su modelo de fondo (BL) los objetos que permanecen parados y un modelo a corto plazo que se actualiza más rápidamente, es decir, que incorpora más rápidamente a su modelo de fondo (BC) los objetos que permanecen estáticos. Para cada imagen, se estima el modelo a largo plazo y el modelo a corto plazo comparando la imagen actual  $I$  con ambos modelos de fondo, BL y BC. Al realizar esa comparación, se obtienen dos máscaras binarias de frente FL y FC, donde  $F(x, y) = 1$  indica que el píxel perteneciente a la posición  $(x,y)$  no se corresponde con el fondo de la imagen [8][14][29][19][20]. En este método, dependiendo del valor de FL y FC, pueden ocurrir 4 hipótesis, que se detallan a continuación:

- FL = 1 y FC = 1, se considera que el píxel  $(x,y)$  corresponde a un objeto en movimiento debido a que dicho píxel de la imagen bajo análisis no se asemeja a ninguno de los dos modelos de fondo.
- FL = 1 y FC = 0, se considera que el píxel  $(x,y)$  corresponde a una región estática, pues su localización en la imagen actual bajo análisis se asemeja al píxel homólogo del modelo de fondo a corto plazo (que tiene el objeto

absorbido a corto plazo), pero no se asemeja al píxel homólogo del modelo de fondo a largo plazo que aún no ha absorbido el objeto estático.

- $FL = 0$  y  $FC = 1$ , se considera que el píxel  $(x,y)$  es un píxel del fondo de la escena que ha sido ocluido anteriormente. Es decir, un objeto ha ocluido el fondo provocando la absorción por parte del modelo de fondo a corto plazo, pero no ha permanecido el tiempo suficiente como para ser considerado estático y ser absorbido por el modelo de fondo a largo plazo, provocando, al ser removido de la escena, la situación descrita.
- $FL = 0$  y  $FC = 0$ , se considera que el píxel  $(x, y)$  pertenece al fondo de la escena, debido a que se asemeja a ambos modelos de fondo.

**SOM + ACC:** Técnica multi-característica con un único modelo de fondo que realiza un análisis imagen a imagen mediante la acumulación de las máscaras de frente generadas mediante la comparación de cada imagen con el modelo generado por el algoritmo SOM(3D-SOBS). [18]

**GSC (Gaussian State Change):** Técnica mono-característica que realiza un análisis imagen a imagen mediante el uso de varios modelos de fondo combinando propiedades de los mismos. Para este método se utiliza la técnica GMM en la etapa de sustracción de fondo y se modelan el frente, fondo y regiones estáticas con las diferentes gaussianas empleadas. El modelo MoG se basa en usar  $K$  Gaussianas por píxel, que se modelan a través de una media ( $\mu$ ), una desviación típica ( $\sigma$ ) y un factor de peso ( $\omega$ ). 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. 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).[13][12]

**GMM + ACC:** Técnica multi-característica que realiza un análisis imagen a imagen mediante el uso de varios modelos de fondo combinando propiedades de los mismos para posteriormente realizar una acumulación de las máscaras binarias generadas por el modelo MoG. [9][10][11]

**RA + FD (Running Average + Imagen Difference):** Técnica multi-característica que realiza un análisis imagen a imagen.[21]

**Subsampling:** Este método emplea una técnica basada en el submuestreo de máscaras binarias de frente para localizar las regiones estáticas dentro de las mismas

[30]. Después, se realiza una multiplicación lógica píxel a píxel de cada una de las 6 máscaras binarias de frente obtenidas, para calcular así la imagen final de objetos estáticos.

## 2.4. Clasificación de regiones estáticas

El objetivo de esta etapa consiste en clasificar las regiones estáticas detectadas por el módulo anterior entre personas y objetos. Aunque a primera vista, esta etapa pueda interpretarse como un problema de discriminación entre dos clases distintas (personas frente a objetos), una región estática detectada que no ha sido clasificada como persona puede considerarse simplemente como un objeto genérico. Esta etapa es importante, ya que nos permite excluir aquellas regiones estáticas clasificadas como personas de un análisis posterior más detallado. El algoritmo de clasificación asociado a este problema se denomina detector de personas. [31]

La clasificación automática de regiones estáticas entre persona y objetos es un problema complejo. La complejidad de este problema es, principalmente, debido a dos motivos:

- En primer lugar, la gran variabilidad con respecto a la apariencia en la escena debido a la ropa, accesorios personales (paraguas, mochilas, etc) o incluso el peinado que la persona puede presentar hace difícil de caracterizar con precisión toda la clase mediante el uso de una única función. Y además hay que tener en cuenta la enorme cantidad de posturas que una persona puede adoptar y modificar a gran velocidad, lo que da lugar a complejas siluetas difíciles de modelar, como por ejemplo, andar, sentarse en un banco, levantarse, etc.
- En segundo lugar, en la mayoría de casos en los que se requiere un análisis de secuencias de vídeo, se desea que dicho análisis pueda ser realizado a tiempo real, lo que dificulta mucho los algoritmos a implementar en cuanto a coste computacional.

De acuerdo con [32], dos enfoques son predominantes en las aproximaciones para la clasificación de regiones disponibles en la literatura dependiendo de si se basan en curvas de nivel (siluetas) o regiones. En la literatura, se puede encontrar una variedad de esquemas de modelado de personas: siluetas [33][34] articulaciones, modelos volumétricos [35]. En algunos casos, la detección de las partes del cuerpo [36], a diferencia de los modelos de cuerpo completo, puede ser suficiente para llevar a cabo la clasificación. 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

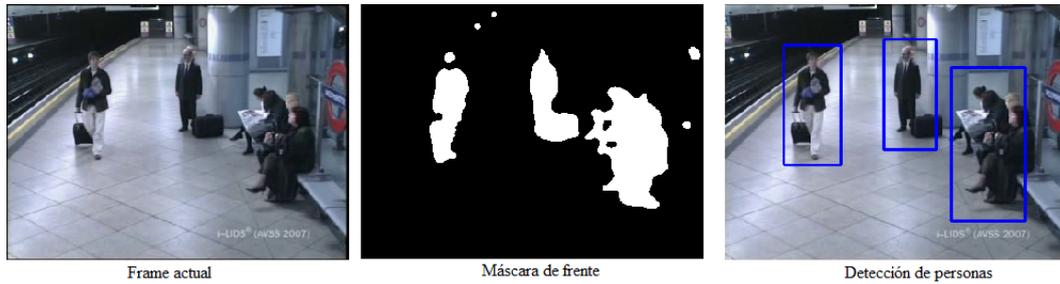


Figura 2.5: Resultados obtenidos tras aplicar un algoritmo de detección de personas sobre la máscara binaria de frente tras aplicar un algoritmo de sustracción de fondo a la secuencia AVSS 2007 abandoned baggage medium. De izquierda a derecha se presentan un imagen de la secuencia, la máscara de frente obtenida por el algoritmo de sustracción de fondo y los blobs de la máscara de frente que han sido clasificados como persona.

usados para predecir la posición de la cabeza y, a partir de entonces, torso, pies y manos.

Sin embargo, una de las principales limitaciones de estas aproximaciones es la necesidad de información previa acerca de la aparición de los objetos en la escena. 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 y características de la gente (andando, alguien parado en algún sitio, alguien sentado, etc.) antes de realizar la clasificación. [37] La Figura 2.5 a modo de ejemplo los resultados obtenidos por un algoritmo de detección de personas.

## 2.5. Discriminación entre objetos robados o abandonados

En esta sección, estudiamos los diferentes enfoques de discriminación que se encuentran en la literatura para distinguir entre los objetos robados y abandonados. Como se ha explicado en las secciones anteriores, los objetos de interés son aquellas regiones de primer plano que han sido clasificados tanto como estáticos como no-personas por las etapas anteriores de análisis. La Figura 2.6 muestra el esquema de un módulo de discriminación entre abandono y robo de objetos.

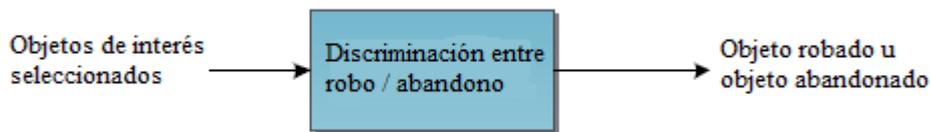


Figura 2.6: Diagrama de bloques representativo de un módulo de discriminación entre robo y abandono de objetos.

### 2.5.1. Descripción

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 algunos frames 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. [38] [39][9]

Algunos enfoques de la literatura simplifican el problema asumiendo que sólo se permiten inserciones de objetos en la escena [40][30][41]. Si bien esto puede ser una suposición válida en ciertos escenarios específicos, tales como la detección de equipaje desatendido en los aeropuertos, estas aproximaciones no tienen en cuenta los posibles elementos generados tras la aplicación de una técnica de sustracción de fondo tales como fantasmas producidos por el movimiento de elementos estáticos del escenario (o el movimiento de objetos que han mantenido estáticos durante un largo periodo de tiempo). Por lo tanto, cabe esperar que los resultados de estos enfoques no sean los suficientemente satisfactorios en las condiciones de un entorno real.

### 2.5.2. Principales técnicas

Las aproximaciones propuestas en la literatura con la capacidad de afrontar el problema de la discriminación entre robo y abandono descrito previamente se pueden categorizar en tres grupos en función del tipo de información del que hacen uso para procesar la discriminación. En primer lugar, aquellas que hacen uso de los contornos del objeto. En segundo lugar, las técnicas que hacen uso de la información de color. Por último se encuentran las técnicas que combinan ambos tipos de información obteniendo así mejores aproximaciones.

**Técnicas basadas en información de contorno:** Estas aproximaciones se basan en la realización del cálculo de la energía en los bordes de la región estática y se compara frente a la energía que posee la imagen de fondo original. [42] Esta energía se supone que es alta cuando un objeto es añadido a la escena, y



Figura 2.7: Resultados obtenidos tras aplicar un algoritmo de discriminación de robo-abandono sobre el imagen actual y el modelo de fondo de una secuencia de vídeo. De izquierda a derecha se presentan un imagen de la secuencia, el modelo de fondo generado por el algoritmo de sustracción de fondo y los objetos discriminados como abandonados por el algoritmo de discriminación utilizado.

baja cuando un objeto es eliminado. Enfoques similares se describen en [43][44]. Proponen el uso del detector de bordes de Canny dentro del cuadro delimitador del objeto estático detectado, tanto en el imagen actual como en el modelo de fondo. Si la presencia del borde es más fuerte en el imagen actual, el objeto es clasificado como abandonado, de lo contrario, el objeto es clasificado como robado.

**Técnicas basadas en información de color:** Estos enfoques emplean la información de color de las áreas definidas por los márgenes del cuadro delimitador de la región estática, tanto en el imagen actual como en el modelo de fondo [45]. Estas aproximaciones parten de la suposición de que las características de color del objeto son lo suficientemente diferentes de las de su entorno. Cuando un objeto es retirado de la escena, se espera que la porción de fondo descubierta tenga propiedades de color similares a las de sus alrededores.

**Técnicas híbridas:** Estos métodos combinan ambos tipos de información, tanto de contorno como de color, obteniendo así mejores resultados. En [26], se usan dos discriminadores (uno basado en color y otro basado en información de contorno), se construye un modelo probabilístico para cada algoritmo en cada evento (robado o abandonado). La discriminación se lleva a cabo mediante el cálculo de la probabilidad media de pertenencia a cada evento y el objeto se clasifica como el evento con mayor probabilidad media.

La Figura 2.7 muestra a modo de ejemplo los resultados obtenidos por una técnica de discriminación robo-abandono.

## 2.6. Datasets disponibles

En este apartado se hace referencia a las secuencias de test más significativas y utilizadas en la literatura. Varios datasets públicos están disponibles para la detección de objetos abandonados y robados. No obstante dichos datasets se utilizan adicionalmente en el campo de la vídeo-vigilancia para evaluar el rendimiento de diferentes módulos de procesamiento (segmentación de objetos de interés, seguimiento, detección de regiones estáticas y detección de personas). A continuación se describen los datasets públicos disponibles:

- PETS 2006 (<http://www.cvg.reading.ac.uk/PETS2006/data.html>).  
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. Este dataset esta compuesto por un total de 6 eventos de abandono de objetos en una estación de ferrocarril registrados por cuatro cámaras distintas enfocadas desde diferentes ángulos (28 vídeos en total). Los vídeos de este conjunto de datos están comprendidos entre 1 y 2 minutos de duración, con una resolución estándar PAL (768x576 píxeles, 25fps).
- PETS 2007(<http://www.cvg.reading.ac.uk/PETS2007/data.html>).  
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. Este conjunto de datos contiene 8 ejemplos de equipaje abandonado en un aeropuerto. Cada evento es registrado por cuatro cámaras diferentes. Adicionalmente, se proporciona una secuencia de entrenamiento de fondo. Las secuencias de vídeo se han registrado en un escenario denso, lleno de gente. Los vídeos están comprendidos entre 2 y 3 minutos de duración, con una resolución estándar PAL (768x576 píxeles, 25fps).
- AVSS 2007 ([http://www.eecs.qmul.ac.uk/~andrea/avss2007\\_d.html](http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html)).  
Este dataset tiene 3 secuencias de vídeo que contienen escenas de abandonado de objetos una estación de metro, con niveles tres niveles de complejidad: fácil, medio y difícil, definidos en términos de la densidad de gente en el escenario. Cada secuencia tiene alrededor de 3,5 minutos de duración, con la resolución PAL.
- 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](http://www.openvisor.org/video_videosInCategory.asp?idcategory=14)).  
 Este conjunto de datos se clasifica en diferentes categorías que incluyen eventos en escenarios interiores y exteriores (acciones humanas, monitorización de tráfico, sombras...). Además incluye un total de 9 secuencias de objetos abandonados, grabadas en un entorno interior. Estas secuencias son de baja complejidad. Los vídeos tienen alrededor de 10 segundos de duración y se proporcionan en la resolución 320x256 @ 25fps.
- CANDELA(<http://www.multitel.be/~va/candela/>).  
 Este conjunto de datos contiene 16 ejemplos de objetos abandonados dentro de un pasillo en el interior de un edificio, con las interacciones entre los diferentes propietarios de objetos. Los vídeos tienen alrededor de 30 segundos de duración, ofrecidos en una resolución de 352x288. A pesar de la sencillez del escenario, la baja resolución y el tamaño relativamente pequeño de los objetos, estas secuencias suponen un complejo reto para la segmentación de objetos de interés.
- WCAM([http://wcam.epfl.ch/seq\\_video\\_surv/seq\\_video\\_surv.html](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.
- CANTATA(<http://www.multitel.be/~va/cantata/LeftObject/>).  
 Este dataset contiene secuencias de vídeo que muestran abandono y robo de objetos en un escenario exterior. Un total de 31 secuencias de 2 minutos grabadas con dos cámaras diferentes. Algunos vídeos incluyen personas dejando los objetos en la escena (objetos abandonados) y otros vídeos muestran gente cogiendo los mismos objetos de la escena (objetos robados). Los vídeos se proporcionan en resolución estándar PAL, comprimidos en formato MPEG-4.

En la Figura 2.8 se muestran fotogramas representativos de los datasets públicos descritos en este apartado.

En la Tabla 2.4 se expone la relación entre los datasets públicos disponibles y su utilización en los artículos de la literatura trabajados. Como se puede observar, los datasets más utilizados para evaluar tanto el rendimiento de la detección de objetos abandonados y robados como el rendimiento de los diferentes módulos de procesamiento (segmentación de objetos de interés, seguimiento, detección de regiones estáticas y detección de personas) son AVSS2007, PETS2006 y CAVIAR.



Figura 2.8: Frames a modo de ejemplo de los datasets públicos disponibles en la literatura.

Artículo	AVSS2007	Binders	CANDELA	CAVIAR	CVPR2014	Dog	PETS2001	PETS2006	PETS2007
[6]					Sí				
[7]	Sí							Sí	
[8]				Sí					
[9]	Sí								
[10]	Sí							Sí	
[11]								Sí	
[12]	Sí						Sí	Sí	
[13]	Sí							Sí	
[14]				Sí				Sí	
[29]	Sí							Sí	
[15]	Sí			Sí				Sí	
[16]	Sí			Sí				Sí	
[17]								Sí	
[18]	Sí	Sí				Sí			
[20]								Sí	
[21]	Sí								
[46]								Sí	
[47]	Sí		Sí	Sí				Sí	Sí

Tabla 2.4: Relación entre los datasets públicos disponibles y su utilización en los artículos estudiados.

## 2.7. Métrica y evaluación

En este apartado se hace referencia a los parámetros de rendimiento más significativos y utilizados en la literatura, así como a los resultados obtenidos al evaluar las aproximaciones propuestas por los autores de los artículos estudiados sobre los datasets públicos disponibles en la literatura y en algunos casos, datasets privados.

### 2.7.1. Parámetros de rendimiento

A continuación se exponen las métricas utilizadas en la literatura. Para evaluar el rendimiento de las técnicas expuestas en la literatura, los autores de dichas aproximaciones hacen uso de las siguientes métricas de evaluación o escogen únicamente un conjunto de ellas.

TP (True Positive): Indica el número de detecciones realizadas correctamente. Una detección se considera correcta cuando las coordenadas que definen su localización espacial en el cuadro de vídeo durante la ventana de frames consecutivos y la etiqueta asignada por el discriminador robo-abandono coinciden con las anotadas en el ground-truth de la secuencia de vídeo.

FP (False Positive): Indica el número de detecciones realizadas incorrectamente. Una detección se considera incorrecta cuando su localización espacial, su ventana de frames consecutivos o su etiqueta no coinciden con las anotadas en el ground-truth.

TN (True Negative): Indica el número de detecciones rechazadas correctamente.

FN (False Negative): Indica el número de detecciones perdidas, es decir, aquellas que en un caso ideal deberían haber sido detectadas.

Re (Recall): Indica la tasa de detecciones correctas con respecto al número total de detecciones posibles en el caso ideal.

$$Re = TP/(TP + FN) \quad (2.1)$$

P (Precision): Indica la tasa de detecciones correctas con respecto al número total de detecciones realizadas.

$$P = TP/(TP + FP) \quad (2.2)$$

Sp (Specificity): Indica la tasa de detecciones rechazadas correctamente con respecto al número total de detecciones rechazadas.

$$Sp = TN/(TN + FP) \quad (2.3)$$

F-Measure: Indica un valor único ponderado de a partir de los valores de Precision y Recall.

$$F - Measure = (2 * Precision * Recall)/(Precision + Recall) \quad (2.4)$$

FPR (False Positive Rate): Indica la probabilidad de realizar una detección incorrecta.

$$FPR = FP/(FP + TN) \quad (2.5)$$

FNR (False Negative Rate): Indica la probabilidad de perder una detección.

$$FNR = FN/(TP + FN) \quad (2.6)$$

PWC (Percentage of Wrong Classifications): Indica el porcentaje de detecciones incorrectas y perdidas en total de manera conjunta con respecto al número total de detecciones que se esperan realizar y rechazar correctamente en el caso ideal.

Detección de regiones estáticas				
Artículo	Recall	Precision	F-Measure	Datasets utilizados
[7]	0,903	0,875	0,8887	AVSS2007, PETS2006
[8]	0,974	0,985	0,9275	CAVIAR
[9]	0,7516	0,8037	0,777	AVSS2007, privado
[10]	0,9714	0,923	0,9459	AVSS2007, PETS2006
[11]	0,32	0,585	0,4137	privado
[13]	0,805	0,7916	0,7982	AVSS2007, PETS2006, privado
[14]	1	0,3125	0,4761	AVSS2007, PETS2006, CAVIAR
[29]	1	1	1	AVSS2007, PETS2006
[17]	0,656	0,917	0,76	PETS2006
[18]	1	0,857	0,9229	AVSS2007
[19]	1	0,76	0,863	privado
[20]	-	-	-	PETS2006, privado
[21]	0,8846	0,8846	0,8363	AVSS2007, privado
[46]	0,5	0,5	0,5	PETS2006

Tabla 2.5: Comparativa de los resultados obtenidos tras la evaluación de las aproximaciones propuestas en la literatura en la etapa de detección de regiones estáticas.

$$PWC = (100 * (FP + FN)) / (FP + TN + TP + FN) \quad (2.7)$$

### 2.7.2. Resultados

En esta sección se muestra una comparativa de los resultados obtenidos de las aproximaciones pertenecientes a la literatura estudiadas durante la realización de este proyecto. Los resultados que se exponen a continuación fueron obtenidos mediante la evaluación descrita en el apartado anterior. Como se ha mencionado anteriormente en 2.6, los datasets públicos disponibles en la literatura para la detección de objetos abandonados y robados se utilizan adicionalmente en el campo de la vídeo-vigilancia para evaluar el rendimiento de diferentes módulos de procesamiento (segmentación de objetos de interés, seguimiento, detección de regiones estáticas y detección de personas). La Tabla 2.5 muestra una comparativa de los resultados obtenidos por las aproximaciones de la literatura en la etapa de detección de regiones estáticas mientras que la Tabla 2.6 muestra una comparativa de los resultados obtenidos en la etapa de discriminación robo-abandono de objetos.

Artículo	Discriminación robo-abandono			Datasets utilizados
	Recall	Precision	F-Measure	
[6]	0,7657	0,7696	0,7283	CVPR14
[12]	0,8643	-	-	PETS2001, privado
[15]	0,91	0,71	0,7976	AVSS2007, PETS2006, CAVIAR, privado
[16]	0,9642	0,8709	0,915	AVSS2007, PETS2006, CAVIAR, privado
[47]	1	1	1	PETS2006, PETS2007, CVSG, CANDELA, VISOR

Tabla 2.6: Comparativa de los resultados obtenidos tras la evaluación de las aproximaciones propuestas en la literatura en la etapa de discriminación robo-abandono.

## 2.8. Herramientas utilizadas

### 2.8.1. OpenCV

OpenCV (Open Computer Vision <http://opencv.org/>) es una librería multi-plataforma muy extendida en el ámbito del tratamiento de imágenes. Se distribuye bajo licencia BSD y es de código abierto. Funciona en Windows, Mac OSX y Linux y está programada en C/C++, Python y Java. Inicialmente fue desarrollada por el departamento Intel Research, perteneciente a Intel y en la actualidad el desarrollo y soporte de la librería lo continúa la fundación OpenCV.org.

OpenCV implementa en la actualidad más de 2500 algoritmos de procesamiento de imágenes. La librería cuenta con una gran variedad de funciones, desde funciones básicas para trabajar con una imagen hasta algoritmos de detección y seguimiento de objetos, visión artificial para robots, creación de modelos 3D o reconocimiento facial.

Se ha escogido utilizar OpenCV para la realización del proyecto por ser una librería ampliamente utilizada dentro del laboratorio, facilitando el uso e integración de los algoritmos ya existentes. Otra de las razones que han impulsado la elección, es que al ser *software* libre se distribuye de forma gratuita.

La versión que se ha utilizado de OpenCV ha sido la 3.0 por ser la última disponible al comienzo del proyecto.

### 2.8.2. VirtualDub

VirtualDub(<http://www.virtualdub.org/>) es una herramienta tanto de captura como de procesamiento de vídeo para plataformas Windows de 32 bits y de 64 bits (98 / ME / NT4 / 2000 / XP / Vista / 7), autorizado bajo la Licencia Pública General de GNU (GPL). Carece de la potencia de edición de un editor de propósito general, como Adobe Premiere, pero está optimizada para las operaciones lineales rápidas sobre una secuencia de vídeo. Tiene capacidad de procesamiento por lotes para procesar grandes cantidades de archivos y se puede ampliar con filtros de vídeo de terceros. VirtualDub

está orientado principalmente hacia el procesamiento de archivos AVI, aunque puede leer (no escribir) formatos tales como MPEG-1 o BMP.

### 2.8.3. ViPER

ViPER (The Video Performance Evaluation Resource <http://vipер-toolkit.sourceforge.net/>) es un conjunto de herramientas de secuencias de comandos y programas en Java que permiten llevar a cabo la definición de ground-truths en secuencias de vídeo, y sistemas para evaluar el rendimiento de los resultados obtenidos en comparación con los ground-truth previamente definidos. Esta herramienta está compuesta por dos módulos:

- Vipер-GT es la herramienta de anotación. Ofrece la posibilidad de anotar el ground-truth de una secuencia de vídeo frame a frame definiendo los eventos que tienen lugar en la secuencia de vídeo mediante las coordenadas que designan la posición espacial del evento en el cuadro de vídeo y categorizarlos mediante descriptores a través de una interfaz gráfica.
- Vipер-PE es la herramienta de evaluación de rendimiento. Ofrece una variedad de métricas para realizar la comparación entre archivos de anotación de detecciones en secuencias de vídeo. Con él, el usuario puede seleccionar entre varias métricas para comparar los resultados obtenidos con los datos anotados en el ground-truth. Se puede calcular la precisión y el recall fotograma a fotograma de las coordenadas que definen la posición espacial en el cuadro de vídeo de las detecciones realizadas y sus etiquetas. Adicionalmente, cuenta con un mecanismo de filtrado para evaluar únicamente los subconjuntos de datos de interés.



## Capítulo 3

### Sistema base

En este capítulo se describe el sistema de análisis de vídeo para la detección de objetos abandonados y robados desarrollado durante este proyecto. En primer lugar, se detallan las implementaciones disponibles en el Video Processing and Understanding Lab que me han sido proporcionadas al inicio de este trabajo. A continuación, en la sección 3.2 se introduce la arquitectura del sistema propuesto y se facilita una guía de configuración paso a paso. A continuación, en la sección 3.3 se especifican las técnicas de las distintas etapas del sistema de análisis de vídeo para la detección de objetos abandonados y robados integradas y se describen sus características. En la sección 3.4 se describe la implementación del sistema desarrollado y por último, en la sección 3.5 se especifican los criterios bajo los cuales se han definido los ground-truth de las secuencias de los dataset disponibles para la posterior evaluación del sistema.

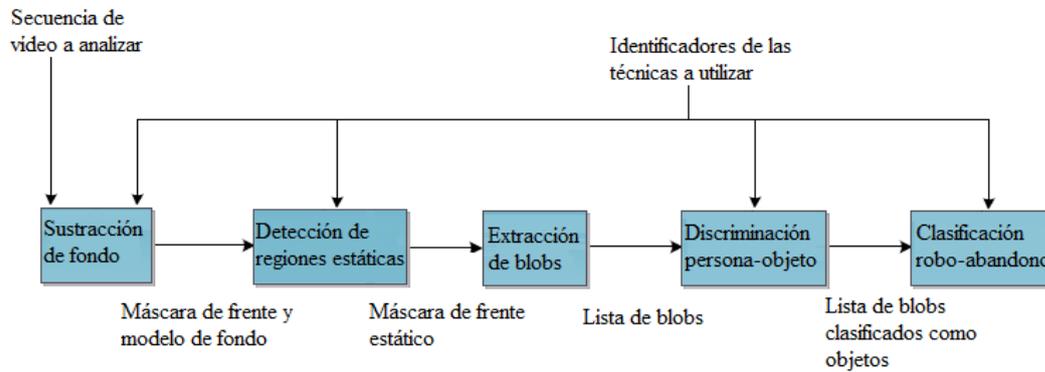


Figura 3.1: Diagrama de bloques del sistema desarrollado en este trabajo.

### 3.1. Trabajo previo

El trabajo presentado en este documento parte de un sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] implementado y proporcionado por el Video Processing and Understanding Lab en el cual se encuentran integradas las técnicas de clasificación robo/abandono descritas en el apartado 3.3.4. Adicionalmente el Video Processing and Understanding Lab ha proporcionado las técnicas de detección de regiones estáticas base del estado del arte implementadas por el laboratorio y un algoritmo de detección de regiones estáticas de primer plano en entornos densamente poblados de vídeo-vigilancia basado en la acumulación espacio-temporal de información de frente, movimiento y estructura [4, 27], desarrollado por el propio laboratorio.

A partir del material proporcionado por el Video Processing and Understanding Lab, técnicas de sustracción de fondo incluidas en la librería OpenCV 3.0, las implementaciones realizadas por los autores de las alternativas recientes de algoritmos de sustracción de fondo y detectores de personas de la literatura incluidos en la librería OpenCV 3.0 se ha desarrollado un sistema de análisis de vídeo para la detección de objetos abandonados y robados configurable que permite seleccionar las técnicas que se desean utilizar en cada etapa del sistema, entre una lista de algoritmos integrados para cada uno de los módulos, en función de las características de la secuencia de vídeo bajo análisis con el fin de obtener óptimos resultados. La Figura 3.1 muestra el diagrama de bloques del sistema descrito.

## 3.2. Arquitectura general y configuración del sistema diseñado

En esta sección se define la arquitectura general del sistema desarrollado y su configuración para un correcto funcionamiento. En el apartado 3.2.1 se muestra el esquema de la arquitectura del sistema y se desglosa su funcionalidad lógica intrínseca. El apartado 3.2.2 contiene la definición de los parámetros de configuración, una secuencia de parámetros de ejemplo y una guía paso a paso para realizar la configuración.

### 3.2.1. Arquitectura general del sistema

El sistema implementado durante la realización de este proyecto efectúa un análisis imagen a imagen de la secuencia de vídeo que recibe como parámetro de entrada. Para cada imagen capturada el módulo de segmentación de fondo genera una máscara de frente y un modelo de fondo. La máscara de frente es una imagen binaria que representa aquellos píxeles del imagen actual que difieren del modelo de fondo generado, considerándose elementos genéricos de primer plano.

A continuación la etapa de detección de regiones estáticas analiza la máscara de frente generada por la segmentación de fondo y genera una máscara de frente estático. La máscara de frente estático es una imagen binaria que representa aquellos elementos genéricos de primer plano que han permanecido estáticos, considerando estáticos aquellos elementos genéricos de primer plano que han permanecido estáticos durante un periodo superior o igual a 10 segundos. Para generar la máscara de frente estático, el modulo de detección de regiones estáticas genera una información espacio-temporal a partir de la información extraída de las sucesivas máscaras de frente analizadas.

A continuación, sobre la máscara de frente estático generada para cada imagen de la secuencia de vídeo analizada, el modulo de extracción de blobs analiza las componentes conexas de la máscara de frente estático y elabora una lista de blobs identificados inequívocamente mediante la asignación a cada blob de un ID único y las coordenadas de la posición espacial de su localización en el cuadro de vídeo.

Posteriormente, el imagen bajo estudio se analiza en la etapa de discriminación persona-objeto por un algoritmo de detección de personas. El detector de personas genera una lista con aquellos elementos de primer plano detectados como persona, nuevamente identificados inequívocamente mediante la asignación, a cada elemento de primer plano interpretado como persona por el algoritmo, de un ID único y las coordenadas de la posición espacial de su localización en el cuadro de vídeo.

Mediante la comparación de las coordenadas de la posición espacial de los elemen-

tos de la lista de blobs extraídos de la máscara de frente estático y los elementos de la lista de elementos de primer plano categorizados como persona se generan 3 nuevas listas de elementos. En primer, lugar una lista de personas no estáticas formada por aquellos elementos de la lista de personas cuyas coordenadas espaciales no se solapan con las coordenadas espaciales de ningún elemento de la lista de blobs estáticos. En segundo lugar, una lista de personas estáticas formada por aquellos elementos de la lista de personas cuyas coordenadas espaciales sí se solapan con las coordenadas espaciales de algún elemento de la lista de blobs estáticos. Y por último, una lista de blobs categorizados como objetos genéricos estáticos formada por aquellos elementos de la lista de blobs estáticos cuyas coordenadas espaciales no se solapan las coordenadas espaciales de ningún elemento de la lista de personas detectadas por el detector de personas.

Por último, la lista de blobs categorizados como objetos genéricos estáticos es procesada por el módulo de discriminación robo-abandono que clasificará cada uno de los elementos de la lista según el criterio seleccionado. La Figura 3.1 muestra el esquema de la lógica funcional del sistema desarrollado.

Adicionalmente, se ha incluido un módulo de anotación que genera un fichero de texto que contiene las detecciones realizadas por el sistema y las define unívocamente mediante la asignación a cada detección de un ID único, las coordenadas de la posición espacial en el cuadro de vídeo donde tiene lugar la detección y la duración de la misma mediante el imagen en el que se realiza la detección y el imagen en el que se finaliza.

### 3.2.2. Configuración del sistema

El sistema de análisis de vídeo para la detección de objetos abandonados y robados desarrollado en este trabajo, requiere unos parámetros de entrada que identifican la secuencia de vídeo objetivo de análisis así como las aproximaciones que se desean utilizar en las etapas de segmentación de fondo, detección de regiones estáticas, discriminación persona-objeto y clasificación robo-abandono tal y como se observa en el diagrama de bloques del sistema que aparece en la Figura 3.1. Estos parámetros de entrada deben introducirse en el siguiente orden y con el siguiente formato:

**Path de la secuencia de vídeo objetivo de análisis:** Cadena de caracteres alfanumérica que identifica la ruta en la cual el intérprete de comandos debe buscar la secuencia. Debe contener todos los directorios en los que se quiera que el intérprete realice la búsqueda, siendo el orden tomado en cuenta al momento de la búsqueda. Los nombres de los directorios no deben contener espacios.

**ID de la técnica de sustracción de fondo seleccionada:** Número entero comprendido entre el 1 y el 7 ambos inclusive. En el apartado 3.3.1 se encuentran listadas las aproximaciones de sustracción de fondo integradas en el sistema así como su identificador correspondiente.

**ID de la técnica de detección de regiones estáticas seleccionada:** Número entero comprendido entre el 1 y el 4 ambos inclusive. En el apartado 3.3.2 se encuentran listadas las aproximaciones de detección de regiones estáticas integradas en el sistema así como su identificador correspondiente.

**ID de la técnica de detección de personas seleccionada:** Número entero comprendido entre el 1 y el 4 ambos inclusive. En el apartado 3.3.3 se encuentran listadas las aproximaciones de detección de personas integradas en el sistema así como su identificador correspondiente.

**ID de la técnica de discriminación robo-abandono seleccionada:** Número entero comprendido entre el 1 y el 4 ambos inclusive. En el apartado 3.3.4 se encuentran listadas las aproximaciones de clasificación robo-abandono integradas en el sistema así como su identificador correspondiente.

**Fichero de salida:** Cadena de caracteres alfanumérica sin espacios y extensión .xml.

Los parámetros de configuración se deben introducir mediante argumentos de comandos separados entre sí por un espacio en blanco como se observa en la Figura 3.3. A continuación se muestra a modo de ejemplo una secuencia de parámetros correcta.:

```
C:/dataset/Secuencia1.avi 1 2 3 4 resultados.xml
```

Para introducir la secuencia de parámetros de configuración en el sistema, se debe seleccionar el proyecto en el explorador de soluciones del entorno de programación tal y como se observa en la Figura 3.2.

### 3.3. Algoritmos integrados en las distintas etapas del sistema

En esta sección se especifican las técnicas integradas en cada una de las etapas del sistema y se describen brevemente el fundamento y las características de cada algoritmo. En el apartado 3.3.1 se describen las técnicas de sustracción de fondo. En el apartado 3.3.2 las técnicas de detección de regiones estáticas. En el apartado 3.3.3 los detectores de personas. Por último en el apartado 3.3.4, los discriminadores robo-abandono.

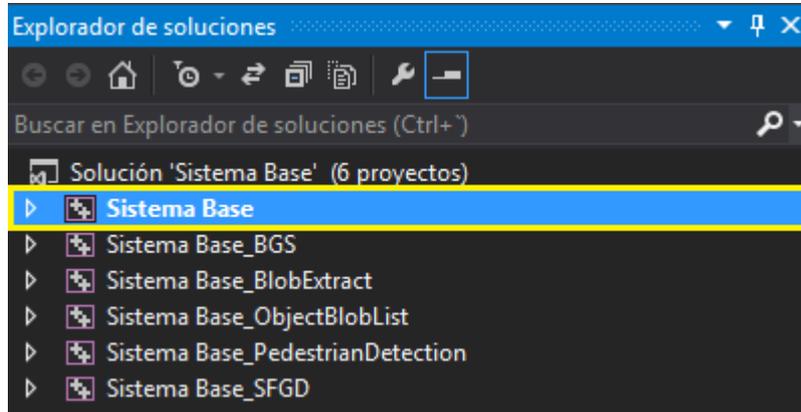


Figura 3.2: Guía de configuración del sistema. Paso 1: Seleccionar el proyecto en el explorador de soluciones

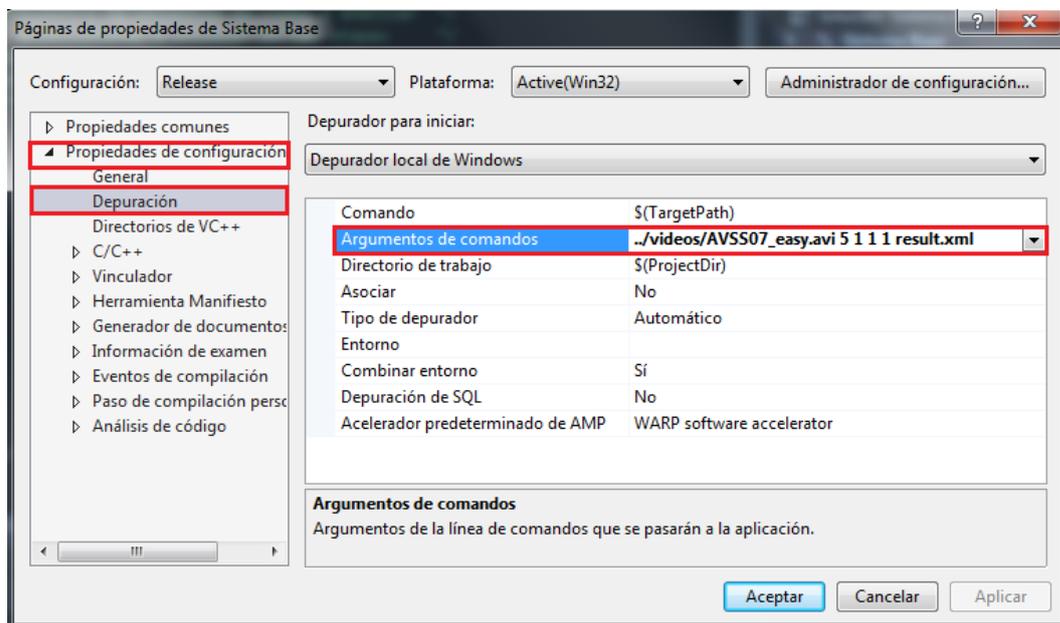


Figura 3.3: Guía de configuración del sistema. Paso 3: Introducir la secuencia de parámetros de configuración como argumento de comandos.

### 3.3.1. Técnicas de sustracción de fondo

La Tabla 3.1 muestra un resumen de las principales características de los algoritmos de sustracción de fondo integrados. La motivación por la cual se han decidido integrar estas aproximaciones ha sido que sus implementaciones se encontraban disponibles en el Video Processing and Understanding Lab y son alternativas recientes que mejoran los resultados obtenidos por los algoritmos del estado del arte en el caso de FASOM, LOBSTER, PAWCS, IMBS y SGMM mientras que en el caso de KNN y GMM, la motivación ha sido su disponibilidad en la librería OpenCV 3.0 utilizada para el desarrollo de este proyecto.

- FASOM ( Fuzzy Adaptative Self-Organizing Model): Este algoritmo, también llamado SOBS (The Self-Organizing Background Substraction) [23], implementa un enfoque de detección de objetos móviles basado en el modelo de fondo neuronal generado automáticamente por un método de auto-organización, sin conocimiento previo del modelo pero utilizando un número configurable de frames al inicio de la secuencia como frames de entrenamiento. La configuración de los parámetros del algoritmo utilizada en el sistema es la propuesta por los autores.
- LOBSTER (LOcal Binary Similarity segmenTER): Esta aproximación descrita en [48], propone un algoritmo de sustracción de fondo adaptativo, de bajo coste computacional, no paramétrico basado en las características espacio-temporales de la secuencia de vídeo que no requiere frames de entrenamiento para generar el modelo de fondo derivado del algoritmo de sustracción de fondo ViBe y una versión mejorada y modificada del descriptor de patrones binarios de similitud local LBSP. Los parámetros de configuración de este algoritmo utilizados en su integración son los propuestos por los autores.
- IMBS ( Independent Multimodal Background Substraction): Esta técnica descrita y propuesta por [49][50], plantea un algoritmo de sustracción de fondo de código abierto diseñado específicamente para trabajar con fondos acuáticos pero que puede ser utilizado sobre cualquier escenario de manera satisfactoria. Este algoritmo se basa en la discretización de la distribución de color de cada píxel aplicando un algoritmo de agrupamiento para generar el modelo de fondo. Este algoritmo requiere un tiempo definido a partir del número de frames utilizados como entrenamiento para generar un modelo de fondo. A continuación un nuevo modelo de fondo de fondo independiente del anterior es construido continuamente con una tasa de actualización constante igual al tiempo requerido para generar el primer modelo de fondo. Los parámetros de configuración

utilizados para la integración de este algoritmo de sustracción de fondo son los propuestos por los autores.

- PAWCS (Pixel-based Adaptive Word Consensus Segmenter): Este enfoque descrito en [51], plantea la definición de un diccionario de etiquetas que caracterizan la apariencia de cada píxel a partir de la información referente al color y la textura y un descriptor de LBSP. Este algoritmo de sustracción de fondo no paramétrico combina características espacio-temporales y de color de la secuencia de vídeo para generar un modelo de fondo a partir de la persistencia de las etiquetas definidas a lo largo del tiempo. Los parámetros de configuración de este algoritmo utilizados en su integración son los propuestos por los autores.
- MoG2 (Mixture of Gaussians): Este enfoque propuesto en [52][53], llamado comúnmente GMM, es el algoritmo definido en el estado del arte en el capítulo 2 en el apartado 2.2. Este algoritmo se encuentra implementado en la librería OpenCV 3.0 y con el objetivo de obtener mejores resultados se ha fijado el número de gaussianas a utilizar a 5.
- KNN (K-Nearest Neighbours ): Este enfoque propuesto por [53], es una versión mejorada del algoritmo de sustracción de fondo KDE definido en el estado del arte en el capítulo 2 en el apartado 2.2. Mientras que el algoritmo KDE utiliza un tamaño fijo de Kernel para toda la función de densidad, la técnica KNN propone el uso de un tamaño de Kernel adaptable a cada punto de estimación. En lugar de calcular un tamaño de Kernel óptimo global, se propone utilizar kernels de un tamaño mayor en las zonas con menor número de muestras y kernels de un tamaño menor en las zonas densamente pobladas. Al igual que el algoritmo KDE, cuanto mayor sea el número de frames de entrenamiento utilizado mejores serán los resultados. Este algoritmo se encuentra implementado en la librería OpenCV 3.0 y se ha utilizado su configuración por defecto.
- SGMM ( Adaptively Splitted Gaussian Mixture Model): Esta aproximación plantea una versión mejorada del algoritmo GMM propuesta y descrita en [54]. Este algoritmo calcula un valor apropiado y adaptado a las características del escenario para la inicialización del parámetro de varianza de nuevos modos y controlar la varianza de los modos existentes a fin de evitar una degeneración del modelo de fondo generado. Los parámetros de configuración de este algoritmo utilizados en su integración son los propuestos por los autores.

Técnica	ID	Requiere entrenamiento	Robustez			
			Cambios de iluminación	Sombras y reflejos	Fondos Multimodales	Oclusiones
FASOM	1	Si	Si	Si	Si	Si
LOBSTER	2	No	Si	Si	Si	Si
IMBS	3	Si	Si	Si	Si	No
PAWCS	4	No	Si	Si	Si	No
GMM	5	No	Si	Si	No	Si
KNN	6	Si	Si	Si	No	No
SGMM	7	No	Si	Si	No	Si

Tabla 3.1: Resumen de las características y robustez de los algoritmos de sustracción de fondo integrados.

### 3.3.2. Técnicas de detección de regiones estáticas

La Tabla 3.2 muestra un resumen de las principales características y parámetros de configuración de los algoritmos de detección de regiones estáticas integrados. La motivación por la cual se han decidido integrar estas aproximaciones ha sido que sus implementaciones han sido desarrolladas por el Video Processing and Understanding Lab y se consideran técnicas básicas de detección de regiones estáticas en el estado del arte en el caso de Subsampling, ACC y DBM, mientras que en el caso de History Images la motivación ha sido que su implementación ha sido desarrollada por el Video Processing and Understanding Lab y supone una mejora de las técnicas básicas de detección de regiones estáticas.

- Subsampling: Se ha fijado a 6 el número de máscaras binarias muestreadas durante el tiempo fijado como estático (como indica el autor), es decir, que se obtienen 6 máscaras binarias de frente, cuyos píxeles con valor 1 indican que pertenecen al frente de la imagen, y los píxeles con valor 0 indican que pertenecen al fondo. La Figura 3.4 muestra un diagrama representativo del funcionamiento del algoritmo.
- Acc: En este enfoque se busca detectar regiones estáticas mediante la acumulación de máscaras de frente obtenidas de manera consecutiva [55]. Su funcionamiento ha sido descrito en el apartado 2.3.
- History Images VPU DOH: Este enfoque, implementado por el Video Processing and Understanding Lab. [4] realiza un análisis espacio-temporal de tres características: frente, movimiento y estructura de cada frame de la secuencia. Cada análisis consta de dos etapas: extracción de característica y su acumulación frame a frame. De la acumulación espacio-temporal de las características,

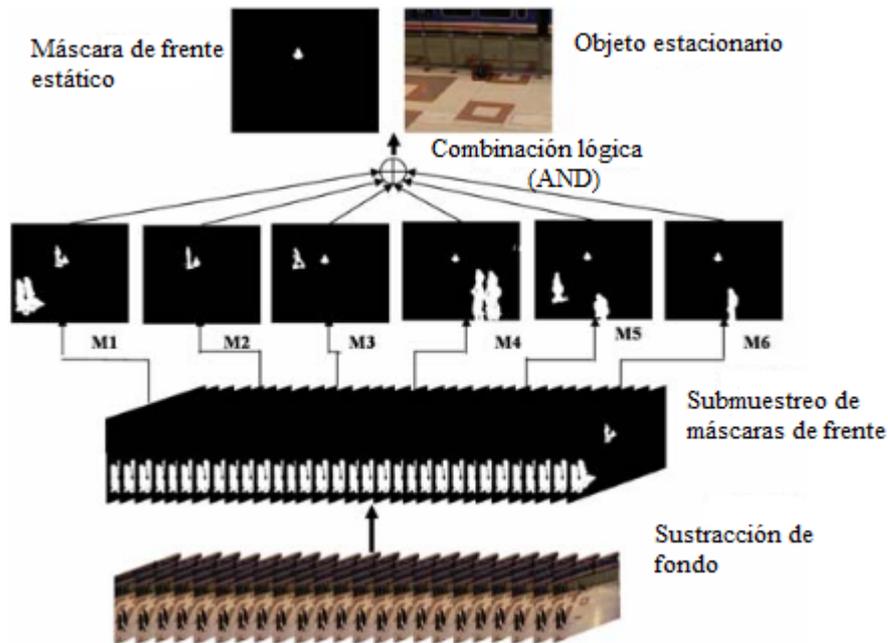


Figura 3.4: Diagrama representativo del algoritmo de detección de regiones estáticas subsampling. Extraído de [28]

se obtienen tres imágenes (History Images), que se combinan para modelar la variación de frente-movimiento-estructura a lo largo del tiempo. Con la imagen resultado se lleva a cabo una umbralización para obtener la máscara de regiones estáticas. La Figura 3.5 muestra un diagrama representativo del funcionamiento del algoritmo.

- DBM: En este método, basado en el estudio realizado en [57], se propone el uso de dos modelos de sustracción de fondo, en lugar de uno sólo como se ha visto en los anteriores métodos. La combinación de ambas máscaras binarias de frente da lugar a la máscara de frente estática tal y como describen las hipótesis descritas en el estado del arte en el apartado 2.3. La Figura 3.6 muestra un diagrama representativo del funcionamiento del algoritmo.

### 3.3.3. Técnicas de detección de personas

La Tabla 3.3 muestra un resumen de las principales características de los algoritmos de detección de personas integrados. La motivación por la cual se han decidido integrar estas aproximaciones ha sido su disponibilidad en la librería OpenCV 3.0 utilizada para el desarrollo de este proyecto.

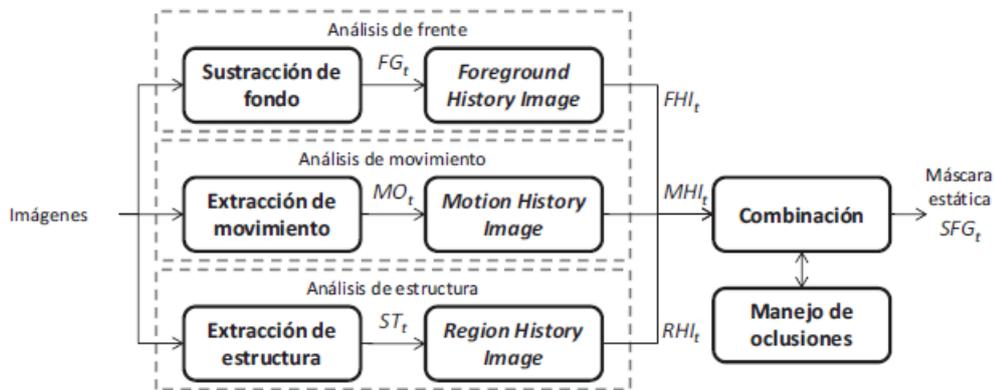


Figura 3.5: Diagrama representativo del algoritmo de detección de regiones estáticas History Images. Extraído de [56]

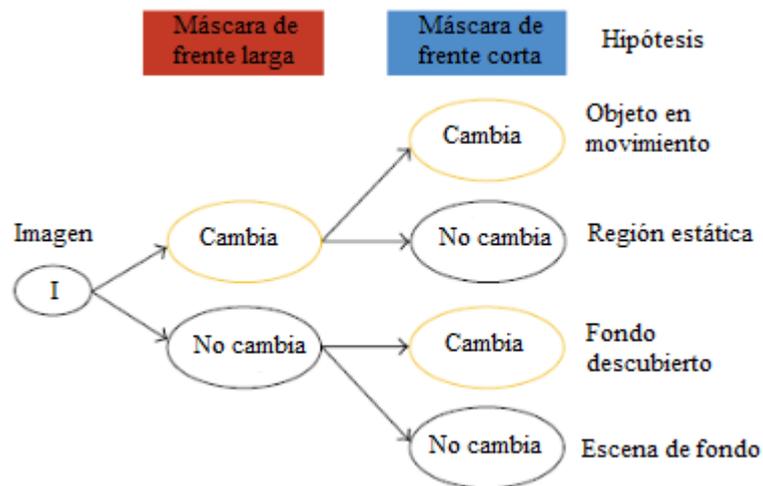


Figura 3.6: Diagrama representativo del algoritmo de detección de regiones estáticas DBM. Extraído de [28]

Técnica	ID	Robustez			Parámetros de configuración
		Inicio del modelo de fondo	Oclusiones	Cambios de iluminación	
Subsampling	1	No	Si	No	número de muestras
ACC	2	No	Si	No	umbrales de detección estática
History Images	3	No	Si	Si	umbrales de detección estática
DBM	4	Si	Si	No	umbral de detección estática

Tabla 3.2: Resumen de la robustez y parámetros de configuración de los algoritmos de detección de regiones estáticas integrados.

Técnica	ID	Tipo	Modelo de entrenamiento
HOG	1	Descriptor	Clasificador SVM con modelo de personas por defecto
Haar Upper Body	2	Clasificador en cascada	Modelo de torsos del cuerpo humano
Haar Frontal Face	3	Clasificador en cascada	Modelo de caras humanas desde un punto de vista frontal
Haar Full Body	4	Clasificador en cascada	Modelo de cuerpo humano

Tabla 3.3: Resumen de las características de los algoritmos de detección de personas integrados.

- HOG (Histogram of Oriented Gradients): El histograma de gradientes orientados es un descriptor de características utilizado en procesamiento de imágenes con el propósito de realizar la detección de objetos propuesto en [58]. Con el objetivo de realizar la detección de objetos, se provee al descriptor de un clasificador SVM (Support Vector Machine), un clasificador binario que busca un hiperplano óptimo como función de decisión, previamente entrenado. El descriptor HOG tiene algunas ventajas sobre otros descriptores, ya que opera sobre pequeñas celdas conectadas en las que se divide la imagen, obteniendo robustez a transformaciones geométricas y fométricas, a excepción de la orientación a objetos. En este caso se ha utilizado un clasificador SVM entrenado con modelos de personas en distintas posturas y realizando distintas acciones.
- Haar Cascades (Upper Body): La detección de objetos utilizando clasificadores Haar en cascada basados en características es un método de detección de objetos efectivo propuesto en [59]. Se trata de un enfoque basado en el aprendizaje o entrenamiento del detector, donde una función en cascada es entrenada con una gran cantidad de imágenes positivas y negativas. Una vez entrenado se utiliza para detectar objetos similares a los modelos entrenados en otras imágenes. En este caso se ha entrenado el detector con modelos del tren superior del cuerpo humano.
- Haar Cascades (Frontal Face): Clasificador Haar en cascada entrenado con modelos de caras desde un punto de vista frontal.
- Haar Cascades (Full Body) : Clasificador Haar en cascada entrenado con modelos del cuerpo humano completo.

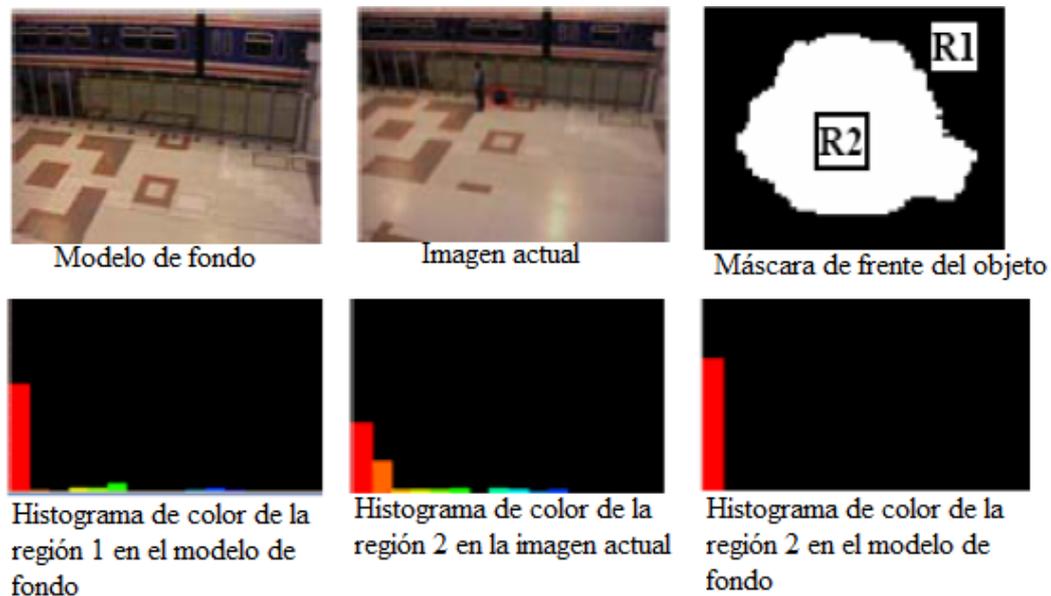


Figura 3.7: Análisis realizado por el algoritmo Color Histogram. Extraído de [60]

### 3.3.4. Técnicas de discriminación robo-abandono

La Tabla 3.4 muestra un resumen de las principales características de los algoritmos de clasificación robo/abandono integrados. La motivación por la cual se han decidido integrar estas aproximaciones ha sido que sus implementaciones han sido desarrolladas por el Video Processing and Understanding Lab y se encontraban integradas en el sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] proporcionado al inicio de este proyecto.

- Color Histogram: Este enfoque propuesto por [2], se basa en la medición de la similitud de color entre las regiones delimitadas por los objetos detectados (regiones internas y externas del recuadro que engloba el contorno del blob) tanto en el modelo de fondo como en el imagen actual. El supuesto es que, en el imagen actual, los objetos robados muestran una mayor similitud de color entre regiones internas y externas que en el modelo de fondo mientras que por el contrario, los objetos abandonados muestran una mayor similitud de color entre regiones internas y externas en el modelo de fondo que en el imagen actual. La Figura 3.7 muestra el análisis que realiza el algoritmo.
- PCC ( Pixel Color Contrast): Este detector se basa en el cálculo del contraste promedio a lo largo de los límites (contornos) de los objetos detectados [45]. El contraste promedio entre puntos dentro y fuera de la región detectada se

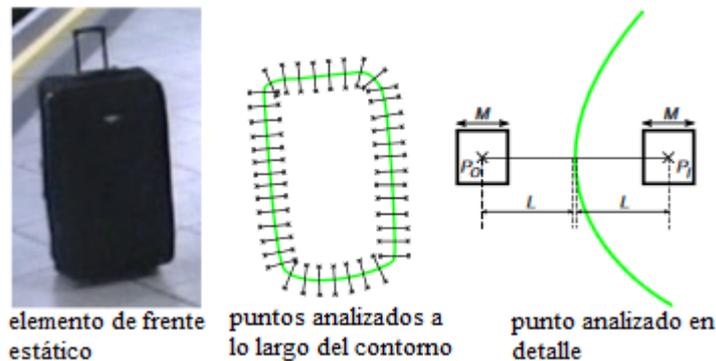


Figura 3.8: Análisis de color a ambos lados del contorno de la región estática realizado por el algoritmo. Extraído de [45]

calcula, tanto en el imagen actual como en el modelo de fondo. Si la distancia entre el contraste promedio calculado en el imagen actual y el contraste promedio calculado en el modelo de fondo supera un umbral definido, el objeto se considera abandonado, en caso contrario se considera robado. La Figura 3.8 muestra el análisis que realiza el algoritmo.

- Gradient H: Este detector se basa en el análisis de los valores del gradiente a lo largo del contorno del objeto detectado tanto en el imagen actual como en el modelo de fondo eliminando la información redundante, es decir, aquellos valores de gradiente idénticos en la imagen actual y fondo. La diferencia entre el gradiente de la imagen actual y fondo garantiza la eliminación de dicha información. El detector de gradiente alto se basa en encontrar los puntos alrededor de los objetos cuyo gradiente es máximo. Este detector dará una medida próxima a 1 cuando un objeto es abandonado y 0 cuando un objeto es robado. La figura 3.9 el análisis que realiza el algoritmo.
- Gradient L: Al contrario que el detector de gradiente alto, el detector de gradiente bajo se basa en encontrar los puntos alrededor de los objetos cuyo gradiente es menor. Este detector dará una medida próxima a 0 cuando un objeto es abandonado y 1 cuando un objeto es robado.

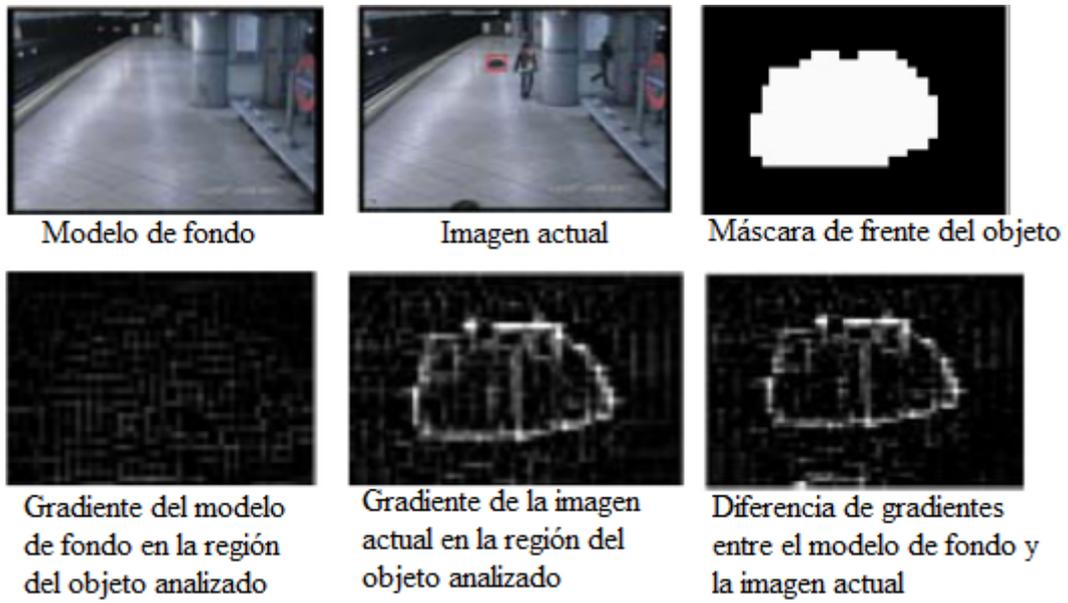


Figura 3.9: Análisis de gradientes realizado por el algoritmo. Extraído de [60]

Técnica	ID	Características
Color Histogram	1	Color
Píxel Color Contrast	2	Color, Bordes
Gradient H	3	Bordes
Gradient L	4	Bordes

Tabla 3.4: Resumen de las características de los algoritmos de clasificación robo/abandono integrados.

## 3.4. Implementación del sistema

### 3.4.1. Captura de la secuencia de vídeo-vigilancia y extracción de parámetros relevantes.

En primer lugar, mediante las instrucciones que incluye la librería OpenCV 3.0, se captura la secuencia de vídeo-vigilancia del path en el que se encuentra, siendo dicho path facilitado al sistema como primer argumento de comandos. Una vez capturada la secuencia, OpenCV 3.0 nos permite extraer tanto el framerate como el número de frames de los que está compuesta. Se fija el tiempo que un elemento de primer plano debe permanecer inmóvil en la secuencia para que el sistema lo detecte como región estática. Para el desarrollo de este proyecto se ha considerado oportuno fijarlo a 10 segundos. Por último se almacena el primer frame de la secuencia en una variable de tipo `cv::Mat`, siendo ésta una de las características de OpenCV 3.0, la versión en la que se ha desarrollado este proyecto.

### 3.4.2. Sustracción de fondo. MyBGSSelector y MyDualBGSSelector

El módulo de sustracción de fondo ha sido diseñado mediante la implementación de dos clases. Se ha tomado esta decisión debido a la necesidad de utilizar dos modelos de fondo en el caso de utilizar DBM como técnica de detección de regiones estáticas. La Tabla 3.5 muestra el esquema funcional de las clases implementadas. En la Figura 3.10 se muestra el modelo de fondo y la máscara de frente obtenidos por uno de los algoritmos integrados en el sistema. A continuación se detalla la implementación de las clases:

- Clase MyBGSSelector: Esta clase encapsula las técnicas de sustracción de fondo especificadas en el apartado 3.3.1 como clases hijas mediante la propiedad de herencia de la programación orientada a objetos. Cuando se crea el objeto de la clase MyBGSSelector, el constructor de esta clase recibe como parámetro el ID de la técnica de sustracción de fondo elegida. Los atributos de esta clase son `img_input` donde se almacena el frame a analizar, `img_output` e `img_bgmodel` donde se almacenarán la máscara de frente y el modelo de fondo generados por la técnica de sustracción de fondo seleccionada mediante su correspondiente ID y `SecuenceROI` que define la región del frame que se desea analizar, durante el desarrollo de este proyecto se ha decidido analizar el frame en su totalidad. Estos atributos han sido declarados como variables `cv::Mat`. La clase MyBGSSelector está compuesta por los siguientes métodos:

- Método `init`: Este método recibe como parámetro de entrada el frame bajo análisis. En primer lugar define el tamaño de los atributos `img_bgmodel`, `img_output` y `SecuenceROI` ajustándolos a la resolución del frame recibido como parámetro de entrada y a continuación se instancia, mediante la propiedad de polimorfismo de la programación orientada a objetos, e inicializa un objeto de la técnica de sustracción de fondo seleccionada mediante su ID en el interior del objeto creado en el constructor de la clase `MyBGSSelector`. Los parámetros de inicialización de las técnicas de sustracción de fondo utilizados son los descritos y recomendados los autores de los algoritmos.
  - Método `process`: Este método recibe como parámetro de entrada el frame a analizar y un booleano que permite decidir si se desean mostrar por pantalla el modelo de fondo y la máscara de frente generadas por la técnica de sustracción de fondo seleccionada. En primer lugar almacena en el atributo `img_input` el frame a analizar y a continuación realiza el procesamiento del frame mediante el algoritmo seleccionado invocando al método `process` interno de dicho algoritmo que recibe como parámetros `img_input`, `img_output` e `img_bgmodel`. Tras el procesamiento del frame, en caso de que así se desee, se muestran por pantalla la máscara de frente y el modelo de fondo generados por el algoritmo elegido y almacenados en los atributos `img_output` e `img_bgmodel`.
  - Método `GetForegroundImage`: Este método devuelve la máscara de frente generada por el algoritmo de sustracción de fondo seleccionado almacenada en el atributo `img_output` de tipo `cv::Mat` para utilizarla en una fase futura del sistema..
  - Método `GetBGModel`: Este método devuelve el modelo de fondo generado por el algoritmo de sustracción de fondo seleccionado almacenado en el atributo `img_bgmodel` de tipo `cv::Mat` para utilizarla en una fase futura del sistema.
  - Método `GetBackgroundRatio`: Este método devuelve la velocidad de absorción del algoritmo de sustracción de fondo seleccionado para utilizarlo en una fase futura del sistema.
- Clase `MyDualBGSSelector`: Se define esta clase mediante las propiedades de herencia y polimorfismo a partir de la clase padre `MyBGSSelector`. Esta clase sigue la misma estructura y funcionalidad que la clase `MyBGSSelector` con la peculiaridad de instanciar 2 objetos del algoritmo de sustracción de fondo

Clase	Métodos	Parámetros de entrada	Descripción
MyBGSSelector	Constructor	BGS_ID	Identifica la técnica seleccionada
	Init	Frame	Instancia e inicializa la técnica seleccionada
	process	Frame, ShowResults	Calcula el modelo de fondo y la máscara de frente
	GetForegroundImage	-	Devuelve la máscara de frente calculada
	GetBGModel	-	Devuelve el modelo de fondo generado
MyDualBGSSelector	Constructor	BGS_ID	Identifica la técnica seleccionada
	Init	Frame	Instancia e inicializa dos algoritmos a dif. tasa de actualización
	Process	Frame, ShowResults	Cálcula dos modelos de fondo y dos máscaras de frente
	GetForegroundImage	-	Devuelve vector de máscaras de frente

Tabla 3.5: Descripción de la implementación del módulo de sustracción de fondo del sistema mediante las clases MyBGSSelector y MyDualBGSSelector

seleccionado en vez de uno. En los métodos init y process se ajustan los parámetros del segundo objeto instanciado para obtener 2 algoritmos de sustracción de fondo a diferentes velocidades de absorción para poder aplicar la técnica de detección de regiones estáticas DBM. En esta clase se han añadido los atributos `img_outputLong`, `img_bgmodelLong` y `SecuencieROILong` junto a los atributos heredados de la clase MyBGSSelector siendo todos ellos del tipo `cv::Mat`. A diferencia de la clase padre, esta clase inicializa dos algoritmos en el método init, invoca a la función process interna de ambos algoritmos en el método process y devuelve un vector de `cv::Mat` en el método GetForegroundImage. Estas variaciones se llevan acabo mediante la propiedad de polimorfismo de la programación orientada a objetos.

La elección de implementar la sustracción de fondo mediante una clase u otra se hace al inicio del programa, comprobando el ID de la técnica de detección de regiones estáticas seleccionada. Se utilizará la clase MyBGSSelector excepto cuando se haya seleccionado DBM como técnica de detección de regiones estáticas, en cuyo caso se utilizará MyDualBGSSelector.

### 3.4.3. Detección de regiones estáticas. MySFGDSelector

El módulo de detección de regiones estáticas ha sido diseñado mediante la implementación de la clase MySFGDSelector. La Tabla 3.6 muestra el esquema funcional



Figura 3.10: Modelo de fondo generado y máscara de frente calculada mediante el algoritmo de sustracción de fondo LOBSTER.

de la clase implementada. En la Figura 3.11 se muestran los resultados de la ejecución secuencial de las clases `MyBGSSelector` y `MySFGDSelector` del algoritmo de sustracción de fondo GMM y el algoritmo de detección de regiones estáticas acumulador de máscaras de frente. A continuación se detalla la implementación de la clase:

Esta clase encapsula las técnicas de detección de regiones estáticas especificadas en el apartado 3.3.2. Cuando se crea el objeto de la clase `MySFGDSelector`, el constructor de esta clase recibe como parámetro el ID de la técnica de detección de regiones estáticas elegida. Los atributos de esta clase son `img_input`, `img_output` e `img_bgmodel` (de tipo `cv::Mat` todos ellos) donde se almacenan el frame a analizar y la máscara de frente y el modelo de fondo generados por la técnica de sustracción de fondo de la etapa anterior y obtenidos mediante los métodos `GetForegroundImage` y `GetBGModel` de la clase `MyBGSSelector` explicados en el apartado 3.4.2. Adicionalmente, la clase `MySFGDSelector` tiene también los atributos `framerate` y `time2static` (en segundos), inicializados en el constructor por los parámetros mencionados en el apartado 3.4.1. Puesto que el sistema diseñado realiza un análisis imagen a imagen de la secuencia, las técnicas de detección de regiones estáticas considerarán que un elemento de primer plano de la máscara de frente obtenida por segmentación de fondo es una región estática si dicho elemento permanece inmóvil y no varían sus coordenadas de la posición espacial de su localización en el cuadro de vídeo durante un número de imágenes consecutivas igual a  $\text{framerate} * \text{time2static}$ . Cabe destacar que las técnicas de detección de regiones estáticas integradas en el sistema fueron implementadas por el Video Processing and Understanding Lab en la versión 2.4.3 de OpenCV, es decir, una versión anterior a la utilizada en el desarrollo de este proyecto y que por lo tanto, fue necesario realizar las conversiones que se detallan a continuación en los métodos de la clase:

- Método `init`: Este método recibe como parámetros de entrada el frame a analizar y el objeto de tipo `MyBGSSelector` que contiene en su interior instanciado un

objeto del algoritmo de sustracción de fondo utilizado. Puesto que el frame a analizar ha sido capturado mediante las instrucciones de OpenCV 3.0 tal y como se describe en el apartado 3.4.1, este frame se almacena en una variable de tipo `cv::Mat`. Las técnicas de detección de regiones estáticas integradas fueron implementadas en la versión 2.4.3 de OpenCV y por lo tanto el tipo de variable utilizado para almacenar imágenes es `IplImage`. Este método, en primer lugar crea una variable interna `Ipl_frame` en la que almacena la información que contiene el frame bajo análisis de tipo `cv::Mat` haciendo uso de una instrucción de casting. A continuación instancia en el objeto de clase `MySFGDSelector` un objeto de la técnica de detección de regiones estáticas, identificada por el ID que recibe el constructor de `MySFGDSelector`, y lo inicializa con los parámetros `Ipl_frame`, `framerate` y `time2static`.

- Método `process`: Este método recibe como parámetros el frame a analizar, un vector de máscaras de frente y el modelo (todos ellos en formato `cv::Mat`) de fondo generados por la técnica de sustracción de fondo utilizada en la etapa anterior y obtenidos mediante los métodos `GetForegroundImage` y `GetBGModel` descritos en el apartado 3.4.2. En primer lugar, nuevamente es necesario declarar variables internas del tipo `IplImage`. Estas variables son `Ipl_frame`, `Ipl_foreground`, `Ipl_foregroundLong`, `Ipl_foregroundShort` e `Ipl_bgmodel` y en ellas se almacena la información contenida en los parámetros de entrada del método, en formato `cv::Mat`, haciendo uso de instrucciones de casting. A continuación se comprueba el número de canales de la máscara de frente generada por la técnica de sustracción de fondo, que es una imagen binaria, pero la variable que la almacena puede tener 3 canales aunque 2 de ellos no contengan información. Mediante una instrucción de conversión de formato se fuerza que las máscaras binarias tengan un único canal y finalmente se invoca la función `process` interna de la técnica de detección de regiones estáticas seleccionada, la cual genera una máscara de frente estático almacenada en una variable `StaticMask` de tipo `IplImage` que se muestra por pantalla.
- Método `GetStaticForeground`: Este método devuelve una máscara de frente estático binaria que contiene aquellas regiones detectadas como estáticas. Para facilitar la manipulación de dicha máscara, almacenada en una variable de tipo `IplImage`, es convertida a una variable de tipo `cv::Mat` mediante una instrucción de conversión. Dicha variable de tipo `cv::Mat` es la que finalmente se devuelve para su análisis en etapas futuras del sistema.

Clase	Métodos	Parámetros de entrada	Descripción
MySFGDSelector	Constructor	SFGD_ID, framerate, time2static	Identifica la técnica seleccionada
	Init	Frame, objeto MyBGSSSelector	Instancia e inicializa la técnica seleccionada
	Process	Frame, BGModel, FGMask	Calcula la máscara de frente estático
	GetStaticForeground	-	Devuelve la máscara de frente estático

Tabla 3.6: Descripción de la implementación del módulo de detección de regiones estáticas del sistema mediante la clase MySFGDSelector

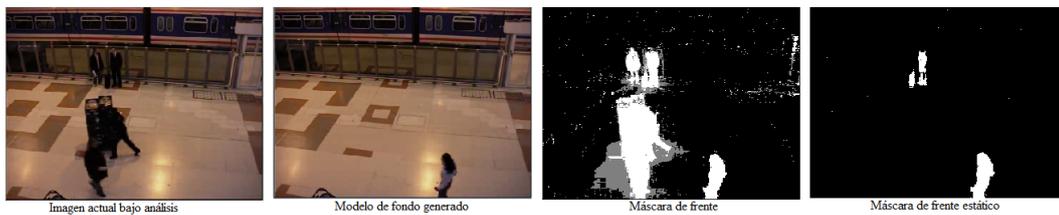
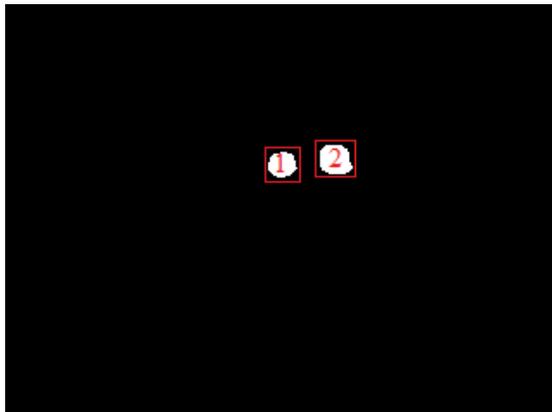


Figura 3.11: Frame actual, modelo de fondo y máscara de frente generados por el algoritmo GMM y máscara de frente estático calculada por acumulador de máscaras de frente.

#### 3.4.4. Extracción de blobs. BlobExtractor

Este módulo hace uso de la clase BlobExtractor implementada en el sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] proporcionado por el Video Processing and Understanding Lab al inicio de este proyecto. No se ha hecho ninguna modificación sobre ella, únicamente se ha integrado en el sistema desarrollado en este trabajo y se ha hecho uso de sus métodos. En primer lugar se crea un objeto de tipo BlobExtractor al inicio del programa principal, la creación de este objeto no requiere ningún parámetro de inicialización. A continuación tras procesar el frame bajo análisis en la etapa de detección de regiones estáticas, se usa el método extractBlobs que recibe como parámetro la máscara de frente estático obtenida por la técnica de detección de regiones estáticas, almacenada en una variable que devuelve el método GetStaticForeground descrito en el apartado 3.4.3. Se notifica del número de blobs extraídos por pantalla mediante el método getNumBlobs y se almacenan los blobs extraídos en un vector de cvBlobs mediante el método getBlobList, este método almacena los blobs asignando a cada uno un ID único, y las coordenadas de la posición espacial de su localización en el cuadro de vídeo. Estas coordenadas describen un rectángulo mediante las posiciones x e y que definen el vértice superior izquierdo del rectángulo y las coordenadas w y h que describen el ancho y alto. La Figura 3.12 muestra un ejemplo de los resultados obtenidos de la extracción de blobs.



Máscara de fente estático

Bloblist				
ID	x	y	w	h
1	165	102	21	21
2	197	97	23	25

Figura 3.12: Ejemplo de los resultados obtenidos tras aplicar la extracción de blobs a la máscara de frente estático.

### 3.4.5. Detección de personas. MyPEDESTRIANDETECTORSelector

El módulo de detección de personas ha sido diseñado mediante la implementación de la clase MyPEDESTRIANDETECTORSelector. La Tabla 3.7 muestra el esquema funcional de la clase implementada.

Esta clase encapsula las técnicas de detección de personas especificadas en el apartado 3.3.3. Cuando se crea el objeto de la clase MyPEDESTRIANDETECTORSelector, el constructor de esta clase recibe como parámetro el ID de la técnica de detección de personas elegida. Cabe destacar que las técnicas de detección de personas integradas en este sistema pertenecen a la librería OpenCV 3.0. Esta clase tiene un único atributo, un vector de rectángulos, found, en el que se almacenan los rectángulos que recuadran las personas detectadas en cada frame analizado. La clase MyPEDESTRIANDETECTORSelector contiene los siguientes métodos:

Método init: Este método no recibe ningún parámetro como entrada. Instancia en el objeto de clase MyPEDESTRIANDETECTORSelector un objeto de la técnica de detección de personas, identificada por el ID que recibe el constructor de MyPEDESTRIANDETECTORSelector y carga el modelo de entrenamiento oportuno.

Método process: Este método recibe como parámetro de entrada el frame bajo análisis almacenado en una variable de tipo cv::Mat. Invoca a la función interna DetectMultiScale de la técnica de detección de personas seleccionada que recibe como parámetros el frame bajo análisis, el vector de rectángulos en el que se almacenan las posiciones espaciales de la localización de las personas detectadas en el cuadro de vídeo, y un conjunto de parámetros de configuración. El con-

Clase	Métodos	Parámetros de entrada	Descripción
MyPEDESTRIAN	Constructor	PEDESTRIAN_ID	Identifica la técnica seleccionada
	Init	-	Instancia el detector seleccionado y carga el modelo de entrenamiento
DETECTORSelector	process	Frame	Almacena en el vector de rectángulos los boundingbox de las personas detectadas y devuelve el vector

Tabla 3.7: Descripción de la implementación del módulo de detección de personas del sistema mediante la clase MyPEDESTRIANDETECTORSelector

junto de parámetros de configuración de cada detector es el recomendado por la documentación de OpenCV, [http://docs.opencv.org/3.1.0/d5/d33/structcv\\_1\\_1HOGDescriptor.html#gsc.tab=0](http://docs.opencv.org/3.1.0/d5/d33/structcv_1_1HOGDescriptor.html#gsc.tab=0) para el caso del descriptor HOG, y para los clasificadores en cascada basados en características [http://docs.opencv.org/2.4/modules/objdetect/doc/cascade\\_classification.html?highlight=cascadeclassifier#cascadeclassifier](http://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html?highlight=cascadeclassifier#cascadeclassifier). A continuación se informa por pantalla del número de personas detectadas en el frame que es igual al número de elementos insertados en el vector de rectángulos found y se devuelve dicho vector, para su utilización en futuras etapas del sistema.

### 3.4.6. Personas estáticas y en movimiento. Funciones Solape y DrawStaticPeople

Los detectores integrados en el módulo de detección de personas del apartado 3.4.5 únicamente devuelven un vector de rectángulos que contiene las coordenadas de la posición espacial de la localización de las personas detectadas en el cuadro de vídeo. Por lo que se han implementado las funciones Solape y DrawStaticPeople para visualizar las detecciones realizadas. Con el fin de obtener una mayor cantidad de información visual, se ha diseñado una estrategia que recuadra en color azul aquellas personas detectadas y que aparecen en movimiento en la escena, y en color verde aquellas aquellas personas que han permanecido estáticas el número de frames necesarios como para considerarse regiones estáticas. Este objetivo es alcanzado mediante la implementación de las funciones tal y como se describe a continuación:

- Función Solape: Esta función recibe como parámetros de entrada 2 rectángulos. Un rectángulo representa las coordenadas de la posición espacial de la localización en el cuadro de vídeo de uno de los elementos del vector de rectángulos

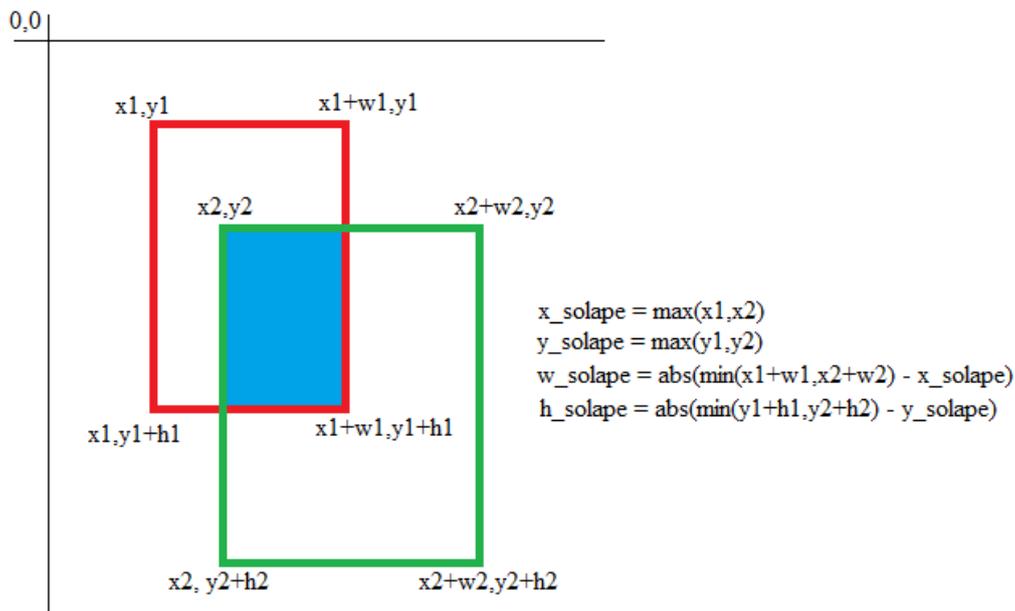


Figura 3.13: Cálculo del solape entre el boundingbox de una persona detectada y un blob extraído de la máscara de frente estática.

obtenido por el detector de personas descrito en el apartado 3.4.5 y el otro rectángulo representa las coordenadas de la posición espacial de localización en el cuadro de vídeo de uno de los elementos del vector de cvBlobs obtenido tras la extracción de blobs descrita en el apartado 3.4.4. Si el rectángulo que representa el área de la persona detectada contiene alguno de los vértices del rectángulo que representa el área de la región estática extraída, se calcula el porcentaje de área del rectángulo persona solapada por el área del rectángulo región estática y se devuelve dicho porcentaje de solapamiento. En caso contrario la función solape devuelve 0. La Figura 3.13 muestra la estrategia planteada para el cálculo de solapamiento de los rectángulos.

- Función DrawStaticPeople: Esta función recibe como parámetros la imagen bajo análisis actual almacenada en una variable cv::Mat, el vector de rectángulos obtenido en la etapa de detección de personas, el vector de cvBlobs obtenido en la etapa de extracción de blobs y un booleano que permite decidir si se desea que se muestren los resultados visualmente o no. La función DrawStaticPeople recorre el vector de rectángulos persona, para cada rectángulo persona realiza la función solape con todos los elementos del vector de regiones estáticas. Si el rectángulo persona coincide con algún rectángulo región estática en más del 60%, se considera persona estática y se recuadra en el frame bajo análisis

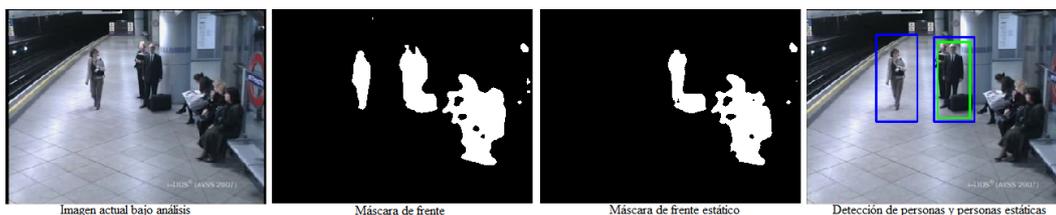


Figura 3.14: Resultados gráficos de la función DrawStaticPeople implementada.

con un rectángulo de color verde, en caso contrario se considera persona en movimiento y se recuadra en el frame bajo análisis con un rectángulo de color azul. La Figura 3.14 muestra un ejemplo de los resultados obtenidos por esta función.

### 3.4.7. Definición de una bloblist de objetos. Funciones CompareBlobs y DefineObjectBlobList

Con el objetivo de que el clasificador robo-abandono únicamente analice aquellas regiones estáticas consideradas objetos genéricos y no analice aquellas regiones estáticas que previamente han sido clasificadas como persona, se elabora una lista de blobs que únicamente contenga las regiones estáticas que no se han clasificadas como persona y que por lo tanto se consideran automáticamente como objetos genéricos. Para ello, se han integrado las clases BlobList y ObjectBlob implementadas en el sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] proporcionado por el Video Processing and Understanding Lab al inicio de este proyecto y adicionalmente se han implementado las funciones CompareBlobs y DefineObjectBlobList. Para ello, al principio del programa se crea un objeto BlobList que contiene objetos del tipo ObjectBlob, y se rellena con las funciones implementadas tras efectuar la detección de personas.

- Función CompareBlobs: Esta función, declarada como booleana, recibe como parámetros un elemento del vector de blobs obtenido tras la extracción de blobs descrita en el apartado 3.4.4 y un elemento del vector de rectángulos obtenido tras la detección de personas descrita en el apartado 3.4.5. En primer lugar define el rectángulo descrito por las coordenadas de la posición espacial de la localización del blob en el cuadro de vídeo. A continuación se invoca la función Solape descrita en el apartado 3.4.6 que calcula el porcentaje de solapamiento entre el rectángulo del blob y el rectángulo del vector obtenido tras la detección de personas. Dicho valor se almacena en el atributo Peoplelikelihood del objeto

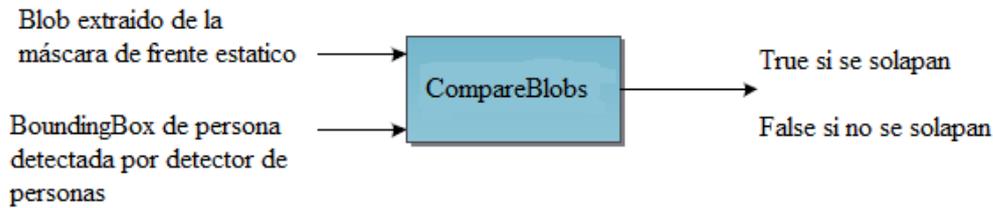


Figura 3.15: Diagrama de bloques de la función CompareBlobs implementada



Figura 3.16: Diagrama de bloques de la función DefineObjectBlobList implementada

de clase cvBlob, este atributo contiene la probabilidad de ser persona que tiene el blob analizado. Si el blob analizado tiene una probabilidad de ser persona superior al 50%, la función CompareBlobs devuelve true, en caso contrario devuelve false. La Figura 3.15 muestra el diagrama de bloques funcional de la implementación realizada.

- Función DefineObjectBlobList: Esta función recibe como parámetros la lista de blobs obtenida como resultado de la extracción de blobs, el vector de rectángulos obtenido como resultado de la detección de personas y un objeto de tipo BlobList que contiene elementos de tipo ObjectBlob. Invoca a la función CompareBlobs para comparar cada uno de los elementos de la lista de blobs con los elementos del vector de rectángulos. Si CompareBlobs devuelve true, se compara el siguiente elemento de la lista de blobs, si por el contrario devuelve false, se crea un objeto de la clase ObjectBlob, que se inicializa con las coordenadas de la posición espacial de la localización del blob en el cuadro de vídeo. La Figura 3.16 muestra el diagrama de bloques funcional de la implementación realizada.

### 3.4.8. Clasificación robo-abandono. MyCLSSFRSelector

El módulo de clasificación robo-abandono ha sido diseñado mediante la implementación de la clase MyCLSSFRSelector. La Tabla 3.8 muestra el esquema funcional de la clase implementada. La Figura 3.17 muestra los resultados de la ejecución secuencial del sistema completo utilizando el algoritmo de sustracción de fondo LOBSTER,

el algoritmo de detección estáticas acumulador de máscaras de frente, el detector de personas HOG y el clasificador robo/abandono gradiente alto.

Esta clase encapsula las técnicas de clasificación robo-abandono especificadas en el apartado 3.3.4. Cuando se crea el objeto de la clase MyCLSSFRSelector, el constructor de esta clase recibe como parámetro el ID de la técnica de detección de personas elegida. Cabe destacar que las técnicas de detección de regiones estáticas integradas en el sistema fueron implementadas en el sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] proporcionado por el Video Processing and Understanding Lab en la versión 2.4.3 de OpenCV, es decir, una versión anterior a la utilizada en el desarrollo de este proyecto y que por lo tanto, fue necesario realizar las conversiones que se detallan a continuación en los métodos de la clase:

- Método `init`: Este método no recibe ningún parámetro como entrada. Instancia en el objeto de clase MyCLSSFRSelector un objeto de la técnica de clasificación robo-abandono, identificada por el ID que recibe el constructor de MyCLSSFRSelector.
- Método `process`: Este método recibe como parámetros de entrada el frame bajo análisis almacenado en una variable de tipo `cv::Mat`, el modelo de fondo y el vector de máscaras de frente (almacenados en variables tipo `cv::Mat`) generados por la técnica de sustracción de fondo seleccionada en la primera etapa del sistema y obtenida mediante los métodos `GetBGModel` y `GetForegroundImage` descritos en el apartado 3.4.2, la máscara de frente estático (almacenada en variable tipo `cv::Mat`) generada por la técnica de detección de regiones estáticas seleccionada en la segunda etapa del sistema y obtenida mediante el método `GetStaticForeground` descrito en el apartado 3.4.3 y la lista de objetos genéricos generada por la función `DefineObjectBlobList` descrita en el apartado 3.4.7. En primer lugar, puesto que las técnicas de clasificación fueron implementadas en la versión 2.4.3 de OpenCV, este método declara las variables `Ipl_result`, inicializada a `NULL`, `Ipl_frame`, `Ipl_bkgImage`, `Ipl_ForegroundImage` e `Ipl_StaticFGImage` de tipo `IplImage`, en las que se almacena la información del frame bajo análisis, el modelo de fondo, la máscara de frente, y la máscara de frente estático respectivamente, mediante instrucciones de casting. A continuación, se elimina el ruido y se completan los huecos de las componentes conexas de la máscara de frente estático mediante operaciones de erosión y dilatación y se binariza el resultado. Por último se invoca al método `process` de la técnica de clasificación robo-abandono seleccionada en el constructor de MyCLSSFRSelector que recibe

Clase	Métodos	Parámetros de entrada	Descripción
MyCLSSFRSelector	Constructor	CLSFR_ID	Identifica la técnica seleccionada
	Init	-	Instancia e inicializa la técnica seleccionada
	Process	Frame, BGModel, FGMask, StaticFGMask, bloblist de objetos	Categoriza los objetos de la bloblist en abandono/robo según los criterios de la técnica seleccionada

Tabla 3.8: Esquema de la implementación del módulo de discriminación robo/abandono del sistema mediante la clase MyCLSSFRSelector

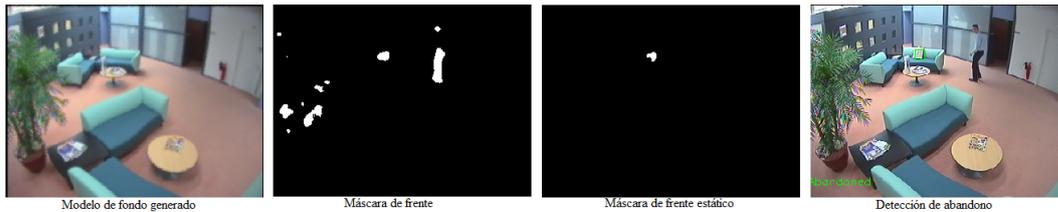


Figura 3.17: Modelo de fondo y máscara de frente generados por el algoritmo LOBS-TER, máscara de frente estático calculada por acumulador de máscaras de frente y clasificación robo/abandono mediante gradiente alto.

como parámetros las variables de tipo `IplImage` declaradas al inicio del método y la lista de objetos genéricos. Se clasifican los objetos según el criterio de la técnica seleccionada y se muestran los resultados de la clasificación de manera visual en la variable `Ipl_result`. Esta variable contiene el frame bajo análisis y muestra recuadros azules que encapsulan aquellos objetos clasificados como robados y recuadros verdes para los objetos clasificados como abandonados.

### 3.4.9. Anotación de resultados. EventController

Por último, con el objetivo de realizar una evaluación analítica de los resultados del sistema implementado, se ha integrado una versión adaptada de la clase `EventController` incluida en el sistema de análisis de vídeo para la detección de objetos abandonados y robados [26] proporcionado por el Video Processing and Understanding Lab. al inicio de este proyecto que anota las detecciones de regiones estáticas y sus clasificaciones en un fichero de extensión `.xml` recibido por argumento de comandos en la configuración del sistema como se describe en el apartado 3.2.2.

## 3.5. Definición de ground-truths

Viper-GT es una herramienta para anotar los ground-truths de las secuencias de vídeo. Las anotaciones se almacenan en archivos XML independientes que siguen la especificación de formato de archivo tipo Viper. El archivo contiene un encabezado en el que un conjunto de descriptores están predefinidos por el usuario. Estos descriptores

contienen información acerca de eventos u objetos específicos en la escena. Cada descriptor caracteriza por un identificador único, un período de tiempo (intervalo de fotogramas) y un conjunto de atributos definido por el usuario. Para el sistema desarrollado en este proyecto se han definido cinco descriptores:

**PutObject:** Este descriptor se define como referencia para los descriptores AbandonedObject y AbandonedPerson. Indica el primer frame en el que un elemento genérico de la escena, ya sea persona u objeto, permanece inmóvil por primera vez, y se mantendrá inmóvil durante un número de frames consecutivos equivalente o superior a 10 segundos de secuencia. El intervalo de fotogramas durante el que permanece activo este descriptor es desde el primer frame en el que el objeto genérico permanece inmóvil hasta que adquiere velocidad de movimiento nuevamente y abandona su posición o hasta el final de la secuencia de vídeo en caso de permanecer estático.

**AbandonedObject:** Este descriptor se corresponde con aquellos objetos de la escena que se consideran abandonados. Un objeto se considera abandonado tras permanecer inmóvil en la escena durante un número de frames consecutivos equivalente a 10 segundos de secuencia. El intervalo de fotogramas durante el que permanece activo este descriptor es desde el frame en el que se define el descriptor PutObject más un número de frames equivalente a 10 segundos de secuencia hasta el frame en el que el objeto modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo o hasta final de secuencia si permanece estático.

**StolenObject:** Este descriptor se corresponde con aquellos objetos de la escena que se consideran robados. Un objeto se considera robado cuando forma parte del modelo de fondo generado por la técnica de sustracción de fondo al inicio de la secuencia o cuando previamente ha sido anotado por el descriptor AbandonedObject y modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo. El intervalo de fotogramas durante el que permanece activo este descriptor es desde el primer frame en el que el objeto modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo hasta un número de frames superior equivalente a 10 segundos de secuencia de vídeo o final de secuencia en caso de alcanzarse el final de la misma antes de 10 segundos.

**AbandonedPerson:** Este descriptor se corresponde con aquellas personas de la escena que se consideran estáticas. Una persona se considera estática tras permanece inmóvil en la escena durante un número de frames consecutivos equivalente

a 10 segundos de secuencia. El intervalo de fotogramas durante el que permanece activo este descriptor es desde el frame en el que se define el descriptor PutObject más un número de frames equivalente a 10 segundos de secuencia hasta el frame en el que la persona modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo o hasta final de secuencia si permanece estática.

**StolenPerson:** Este descriptor se corresponde con aquellas personas de la escena que dejan de considerarse estáticas. Una persona deja de considerarse estática cuando forma parte del modelo de fondo geneado por la técnica de sustracción de fondo al inicio de la secuencia o cuando previamente ha sido anotada por el descriptor AbandonedPerson y modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo. El intervalo de fotogramas durante el que permanece activo este descriptor es desde el primer frame en el que la persona modifica las coordenadas de la posición espacial de su localización en el cuadro de vídeo hasta un número de frames superior equivalente a 10 segundos de secuencia de vídeo o final de secuencia de vídeo en caso de alcanzarse el final de la misma antes de 10 segundos.

Para la definición de los ground-truth realizada en este proyecto, se ha especificado un único atributo con el que caracterizar los descriptores. Este atributo es el boundingbox o caja delimitadora del objeto genérico definido mediante las coordenadas x,y del centroide, el ancho width y el alto height de la caja delimitadora. En la Figura 3.18 se muestra a modo de ejemplo la anotación del abandono de un objeto mediante la herramienta Viper-GT.

A continuación, se muestra parte del contenido del fichero .xml generado tras la anotación del ground-truth de una secuencia a modo de ejemplo. En la Figura 3.19 se muestran los frames en los que tienen lugar los eventos anotados en ejemplo de anotación de ground-truth.

```
<object framespan="2010:4840" id="0" name="PutObject">
  <attribute name="BoundingBox">
    <data:bbbox height="155" width="88" x="59" y="242"/>
  </attribute>
</object>
<object framespan="2260:4840" id="0" name="AbandonedObject">
  <attribute name="BoundingBox">
    <data:bbbox height="155" width="88" x="59" y="242"/>
  </attribute>
```

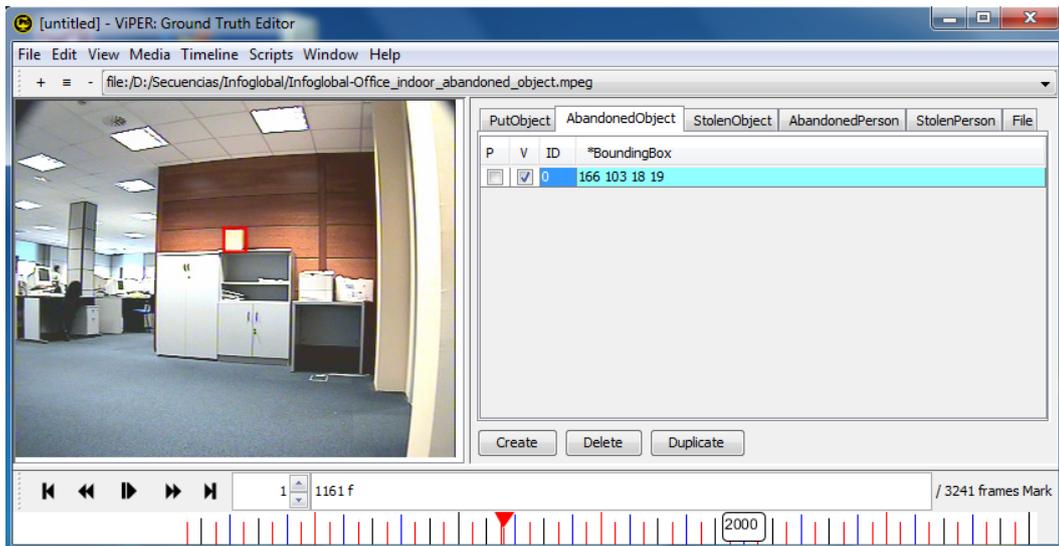


Figura 3.18: Anotación del frame exacto en el que se considera que un objeto definido por las coordenadas de su localización espacial en el cuadro de vídeo ha sido abandonado.



Figura 3.19: Frames de la secuencia en los que tienen lugar los eventos anotados.

</object>



## Capítulo 4

# Resultados experimentales

En este capítulo se muestran los resultados experimentales de dos de los módulos del sistema de análisis de vídeo para la detección de objetos abandonados y robados desarrollado en este proyecto. Se evalúan el módulo de detección de regiones estáticas y el módulo de clasificación robo-abandono. Los tests realizados se han ejecutado en un Intel Core Duo que opera a 2.20GHz y 4GB de RAM.

Este capítulo presenta las secuencias utilizadas para la evaluación del sistema en la sección 4.1, las métricas de evaluación seleccionadas para medir y comparar el rendimiento las técnicas integradas en el sistema en la sección 4.2, la comparativa de los resultados obtenidos en la evaluación de detección de regiones estáticas en la sección 4.3 y la comparativa de los resultados obtenidos en la evaluación de discriminación robo-abandono en la sección 4.4.

### 4.1. Secuencias analizadas

El sistema propuesto es evaluado con un total de 7 secuencias categorizadas en dos niveles de dificultad en función de las características de las secuencias:

La Tabla 4.1 muestra los retos ofrecidos por las secuencias de vídeo escogidas para la realización la evaluación del sistema. La Tabla 4.2 muestra el número de frames a analizar y las detecciones anotadas en el ground-truth de estas secuencias.

Secuencia	Dataset	Dificultad	Cambios de iluminación	Sombras y reflejos	Fondo Multimodal	Escenario concurrido
Office_indoor	INFOGLOBAL	Sencilla	Si	No	No	No
S4-T5-C3	PETS2006	Sencilla	No	No	No	No
AB_Easy	AVSS2007	Sencilla	Si	Si	No	Si
1.09_abandoned_object	CANDELA	Media	No	No	Si	No
1.12_abandoned_object	CANDELA	Media	Si	Si	Si	No
LeftBag_AtChair	CAVIAR	Media	Si	No	No	No
LeftBag_PickedUp	CAVIAR	Media	Si	No	No	No

Tabla 4.1: Retos presentes en las secuencias de vídeo sobre las que se realiza la evaluación

Secuencia	Dataset	Dificultad	# frames	# regiones estáticas	# objetos abandonados	# objetos robados
Office_indoor	INFOGLOBAL	Sencilla	1298	4	3	0
S4-T5-C3	PETS2006	Sencilla	3051	5	1	0
AB_Easy	AVSS2007	Sencilla	4291	7	4	3
1.09_abandoned_object	CANDELA	Media	860	1	1	1
1.12_abandoned_object	CANDELA	Media	695	1	1	1
LeftBag_AtChair	CAVIAR	Media	1117	1	1	1
LeftBag_PickedUp	CAVIAR	Media	1357	1	1	1

Tabla 4.2: Anotaciones de las secuencias de vídeo sobre las que se realiza la evaluación

## 4.2. Métricas de evaluación seleccionadas

Para evaluar el rendimiento de los algoritmos integrados en el módulo de detección de regiones estáticas y en el módulo de discriminación robo-abandono del sistema de análisis de vídeo para la detección de objetos abandonados y robados desarrollado, se ha decidido emplear las medidas estándar de Precision (P), Recall (R) y F-mesure (F):

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

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

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

donde TP, FP y FN indican detecciones correctas, incorrectas y perdidas respectivamente.



Figura 4.1: Falsos positivos detectados por errores del modelo de fondo.

Recall (R): Indica la tasa de detecciones correctas con respecto al número total de detecciones posibles en el caso ideal.

Precision (P): Indica la tasa de detecciones correctas con respecto al número total de detecciones realizadas.

F-Measure (F): Indica un valor único ponderado de a partir de los valores de Precision y Recall.

### 4.3. Evaluación de las técnicas de detección de regiones estáticas

En esta sección se han evaluado los algoritmos de detección de regiones estáticas submuestreo de máscaras de frente y acumulación de máscaras de frente descritos en el apartado 3.3.2 combinados cada uno de ellos con los algoritmos de sustracción de fondo FASOM, LOBSTER y IMBS descritos en el apartado 3.3.1.

Tras realizar una primera evaluación de los algoritmos frente al ground-truth propuesto en la que se han utilizado criterios únicamente objetivos tales como la localización (espacial y temporal) y dimensión exacta de los blobs extraídos por los algoritmos de detección de regiones estáticas se ha obtenido una evaluación no realista, ya que los resultados eran muy dispares a los presentados en el estado del arte. Esto se debe a los numerosos falsos positivos que se detectan por errores como los que se detallan a continuación.

En la Figura 4.1 se muestra el error producido en el modelo de fondo generado por el algoritmo de sustracción de fondo, este error es detectado por el algoritmo de detección de regiones estáticas generando varios falsos positivos.

En la Figura 4.2 se observa que la sombra producida por la maleta es detectada como parte de la región estática perteneciente a la maleta. Esto genera una detección de una dimensión mayor a la anotada en el ground-truth. La detección da lugar a un falso positivo al no coincidir la localización espacial exacta del blob detectado, y un falso negativo al no detectarse la región anotada en el ground-truth.

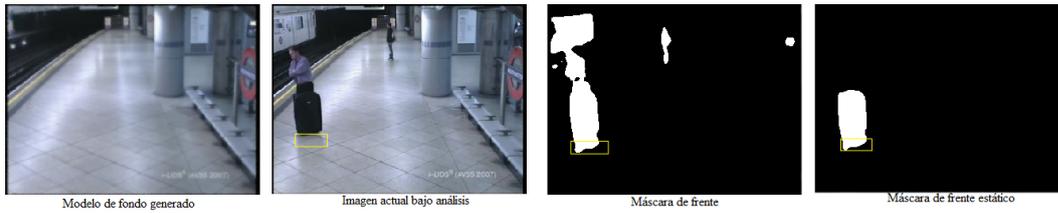


Figura 4.2: Falsos positivos detectados como regiones estáticas producidos por sombras.

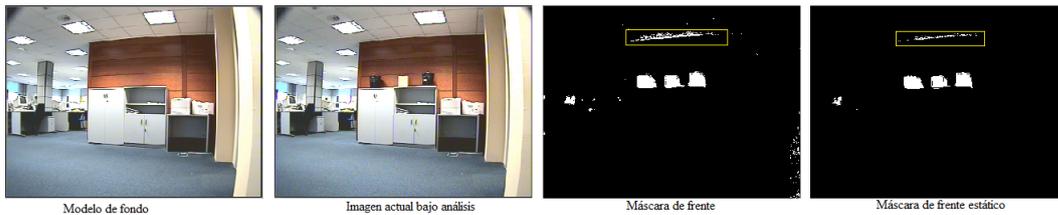


Figura 4.3: Falsos positivos detectados como regiones estáticas producidos por reflejos.

En la Figura 4.3 que los reflejos producidos por los objetos abandonados se detectan como regiones estáticas dando lugar a varios falsos positivos.

En la Figura 4.4 se observa como una persona estática y una maleta estática son detectadas como una única región estática conexa debido a la máscara de frente generada por el algoritmo de sustracción de fondo utilizado. Esto genera un falso positivo, y dos falsos negativos.

Se asume que esta diferencia reside en que las evaluaciones del estado del arte incluyen un mayor nivel semántico en la interpretación de los resultados. Por lo tanto la evaluación presentada a continuación se ha realizado, en base a los criterios del ground-truth más la interpretación del ser humano en los casos ambiguos. Aquellas regiones estáticas detectadas derivadas de un mismo error del modelo de fondo generado por algoritmo de sustracción de fondo se consideran como un único falso positivo. Las regiones estáticas detectadas que incorporan parte de la sombra producida



Figura 4.4: Falsos positivos derivados de regiones estáticas detectadas solapadas.

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	7	4	0.571	1	0.727
S4-T5-C3	6	5	0.833	1	0.9
AB_Easy	11	5	0.455	0.714	0.555
1.09_abandoned_object	1	1	1	1	1
1.12_abandoned_object	3	0	0	0	0
LeftBag_AtChair	4	1	0.25	1	0.4
LeftBag_PickedUp	4	1	0.25	1	0.4
AVERAGE			0.479	0.816	0.568

Tabla 4.3: Resultados de la detección de regiones estáticas utilizando el algoritmo Subsampling con el sustractor de fondo FASOM

por el objeto se consideran como detecciones correctas. Se ha tenido cierta flexibilidad en la detección de personas, pues éstas no permanecen completamente estáticas. Esta variación provoca cambios del blob a lo largo del tiempo, y las coordenadas de la región estática detectada no se corresponden con las coordenadas anotadas en el ground-truth. Las regiones estáticas detectadas conjuntamente derivadas de un reflejo o cambio de luminosidad se han considerado como un único falso positivo. Las regiones estáticas detectadas compuestas por 2 o más elementos del frente muy próximos consideradas como una única región estática han sido consideradas como detecciones correctas.

#### 4.3.1. Evaluación de detección de regiones estáticas mediante submuestreo de máscaras de frente

En la Tabla 4.3 se muestran los resultados obtenidos tras la evaluación de detección de regiones estáticas mediante submuestreo de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo FASOM. Aunque el algoritmo de sustracción de fondo presenta una robustez a cambios de iluminación bastante buena, algunas situaciones no se resuelven correctamente dando lugar a falsos positivos. Debido a los frames de entrenamiento necesarios para generar el modelo de fondo, se producen errores en aquellas secuencias que presentan elementos de frente en movimiento durante los primeros frames de la secuencia y que el algoritmo de detección de regiones estáticas no corrige. La tasa de actualización del modelo de fondo no es lo suficientemente rápida y las secuencias evaluadas son de corta duración por lo tanto, se producen falsos positivos derivados de la detección de *ghosts*.

En la Tabla 4.4 se muestran los resultados obtenidos tras la evaluación de detec-

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	4	4	1	1	1
S4-T5-C3	6	5	0.833	1	0.9
AB_Easy	9	6	0.667	0.857	0.75
1.09_abandoned_object	1	1	1	1	1
1.12_abandoned_object	1	0	0	0	0
LeftBag_AtChair	6	1	0.167	1	0.286
LeftBag_PickedUp	5	1	0.2	1	0.333
AVERAGE			0.552	0.836	0.609

Tabla 4.4: Resultados de la detección de regiones estáticas utilizando el algoritmo Subsampling con el sustractor de fondo IMBS

ciones de regiones estáticas mediante submuestreo de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo LOBSTER. Los errores cometidos por el algoritmo de sustracción de fondo al generar el modelo de fondo a partir del primer frame derivan en la detección de falsos positivos ya que el algoritmo de detección de regiones estáticas no los corrige. La tasa de actualización del modelo de fondo no es lo suficientemente rápida y las secuencias evaluadas son de corta duración por lo tanto, se producen falsos positivos derivados de la detección de *ghosts* de elementos considerados como parte del fondo que abandonan su posición.

En la Tabla 4.5 se muestran los resultados obtenidos tras la evaluación de detecciones de regiones estáticas mediante submuestreo de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo IMBS. Aunque el algoritmo de sustracción de fondo presenta una robustez a cambios de iluminación bastante buena, algunas situaciones no se resuelven correctamente dando lugar a falsos positivos. El tiempo necesario para generar el primer modelo de fondo, provoca que los elementos del frente que tras ser introducidos en el escenario al inicio de la secuencia permanecen estáticos sean incluidos en el primer modelo de fondo generado y ello derive en falsos negativos. La rápida velocidad de absorción del algoritmo de sustracción de fondo provoca que algunos elementos del frente que permanecen estáticos sean absorbidos por el modelo de fondo antes de ser detectados como regiones estáticas. A continuación estos elementos adquieren movimiento nuevamente lo que generará un ghost que es detectado como región estática por el submuestreo de máscaras de frente lo que deriva en falsos positivos y falsos negativos simultáneamente.

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	5	3	0.6	0.75	0.667
S4-T5-C3	3	0	0	0	0
AB_Easy	3	1	0.33	0.143	0.198
1.09_abandoned_object	0	0	0	0	0
1.12_abandoned_object	0	0	0	0	0
LeftBag_AtChair	3	1	0.33	1	0.496
LeftBag_PickedUp	2	0	0	0	0
AVERAGE			0.18	0.27	0.194

Tabla 4.5: Resultados de la detección de regiones estáticas utilizando el algoritmo Subsampling con el sustractor de fondo IMBS

#### 4.3.2. Evaluación de detección de regiones estáticas mediante acumulación de máscaras de frente

En la Tabla 4.6 se muestran los resultados obtenidos tras la evaluación de detección de regiones estáticas mediante acumulación de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo FASOM. Aunque el algoritmo de sustracción de fondo presenta una robustez a cambios de iluminación bastante buena, algunas situaciones no se resuelven correctamente dando lugar a falsos positivos. Debido a los frames de entrenamiento necesarios para generar el modelo de fondo, se producen errores en aquellas secuencias que presentan elementos de frente en movimiento durante los primeros frames de la secuencia y que el algoritmo de detección de regiones estáticas no corrige. La tasa de actualización del modelo de fondo no es lo suficientemente rápida y las secuencias evaluadas son de corta duración por lo tanto, se producen falsos positivos derivados de la detección de *ghosts* de elementos considerados como parte del fondo que abandonan su posición.

En la Tabla 4.7 se muestran los resultados obtenidos tras la evaluación de detección de regiones estáticas mediante acumulación de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo LOBSTER. Los errores cometidos por el algoritmo de sustracción de fondo al generar el modelo de fondo a partir del primer frame derivan en la detección de falsos positivos ya que el algoritmo de detección de regiones estáticas no los corrige. Este algoritmo de sustracción de fondo presenta una gran robustez frente a cambios de iluminación pero en contrapartida, no detecta correctamente aquellos elementos del frente que son demasiado pequeños lo que deriva en falsos negativos. La tasa de actualización del modelo de fondo no es lo suficientemente rápida y las secuencias evaluadas son de corta duración por lo

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	7	4	0.571	1	0.727
S4-T5-C3	4	3	0.75	0.6	0.666
AB_Easy	13	5	0.384	0.714	0.499
1.09_abandoned_object	3	1	0.333	1	0.443
1.12_abandoned_object	0	0	0	0	0
LeftBag_AtChair	4	1	0.25	1	0.4
LeftBag_PickedUp	6	1	0.167	1	0.286
AVERAGE			0.35	0.759	0.431

Tabla 4.6: Resultados de la detección de regiones estáticas utilizando el algoritmo ACC con el sustractor de fondo FASOM

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	3	3	1	0.75	0.857
S4-T5-C3	6	5	0.833	1	0.9
AB_Easy	8	6	0.75	0.857	0.799
1.09_abandoned_object	1	1	1	1	1
1.12_abandoned_object	0	0	0	0	0
LeftBag_AtChair	6	1	0.167	1	0.286
LeftBag_PickedUp	4	1	0.25	1	0.4
AVERAGE			0.571	0.801	0.606

Tabla 4.7: Resultados de la detección de regiones estáticas utilizando el algoritmo ACC con el sustractor de fondo LOBSTER

tanto, se producen falsos positivos derivados de la detección de *ghosts*.

En la Tabla 4.8 se muestran los resultados obtenidos tras la evaluación de detección de regiones estáticas mediante acumulación de máscaras de frente utilizando en la etapa previa de sustracción de fondo el algoritmo IMBS. La rápida velocidad de absorción del algoritmo de sustracción fondo, impide que el acumulador de máscaras de frente detecte las regiones estáticas durante el número necesario de máscaras de frente consecutivas. Las detecciones realizadas se deben a la oclusión parcial de elementos de frente, dichas oclusiones derivan en la detección de falsos positivos pertenecientes a las porciones de las regiones estáticas solapadas. La Figura 4.5 muestra un ejemplo de la velocidad a la que el algoritmo de sustracción de fondo incorpora los elementos del frente a su modelo de fondo antes de ser detectados como regiones estáticas.



Figura 4.5: Error producido por la configuración del sustractor de fondo IMBS y el detector de regiones estáticas acumulador de máscaras de frente. Como se observa, el modelo de fondo ha absorbido el tren antes de detectarlo como estático.

Secuencia	# detecciones realizadas	# detecciones correctas	P	R	F
Office_indoor	0	0	0	0	0
S4-T5-C3	0	0	0	0	0
AB_Easy	5	0	0	0	0
1.09_abandoned_object	0	0	0	0	0
1.12_abandoned_object	0	0	0	0	0
LeftBag_AtChair	0	0	0	0	0
LeftBag_PickedUp	0	0	0	0	0
AVERAGE			0	0	0

Tabla 4.8: Resultados de la detección de regiones estáticas utilizando el algoritmo ACC con el sustractor de fondo IMBS

### 4.3.3. Comparativa y conclusiones de los resultados obtenidos

En la Tabla 4.9 se muestran los resultados promedio de los experimentos realizados. Como se puede observar, la combinación de algoritmos que produce mejores resultados es la utilización del algoritmo LOBSTER como sustractor de fondo y el submuestreo de máscaras de frente como detector de regiones estáticas. Esta configuración será la utilizada para la evaluación de las aproximaciones de discriminación robo/abandono en la sección 4.4.

El algoritmo de sustracción de fondo FASOM utiliza un número determinado de frames al inicio de la secuencia para calcular el modelo de fondo, lo que provoca errores en el modelo para aquellas secuencias que presentan un escenario concurrido y elementos de frente en movimiento durante su inicio, lo que deriva en la detección de falsos positivos por parte de los algoritmos de detección de regiones estáticas. Aunque presenta una robustez bastante buena frente a cambios de iluminación, no es capaz de resolver determinadas situaciones y su tasa de actualización no es lo suficientemente rápida lo que da lugar a numerosos errores en escenarios complejos.

El algoritmo de sustracción de fondo IMBS, necesita un tiempo fijo para generar el primer modelo de fondo, lo que deriva en falsos negativos producidos por la absorción de elementos del frente por el primer modelo de fondo generado. Este algoritmo, pasado un tiempo definido, genera un nuevo modelo de fondo independiente del anterior, su tasa de actualización es bastante rápida por lo que aunque genere un modelo de fondo correcto, detectar las regiones estáticas resulta complicado al generarse un nuevo modelo de fondo que absorba los elementos estáticos antes de que estos sean detectados.

Al contrario que los algoritmos de sustracción de fondo anteriores, el algoritmo LOBSTER genera un modelo de fondo a partir del primer frame de la secuencia, lo que deriva en errores del modelo de fondo para aquellas secuencias que presentan elementos del primer plano en movimiento desde su inicio. La tasa de actualización de fondo no es lo suficientemente rápida como para corregir estos errores en las secuencias evaluadas puesto que son de corta duración. Este algoritmo presenta una gran robustez frente a cambios de iluminación, sombras y reflejos pero por el contrario, su baja tasa de actualización provoca falsos positivos derivados de los errores del modelo de fondo inicial y la detección de ghosts.

Aunque los resultados obtenidos por ambos algoritmos de detección de regiones estáticas son similares, se destaca que el algoritmo de submuestreo de máscaras de frente detecta con mayor facilidad las regiones estáticas correspondientes a personas frente al acumulador, ya que las personas no permanecen completamente estáticas en las secuencias, y mientras que el algoritmo de submuestreo genera la máscara de frente

SFGD	BGS	P	R	F
Subsampling	FASOM	0.479	0.816	0.568
	LOBSTER	0.552	0.836	0.609
	IMBS	0.18	0.27	0.194
ACC	FASOM	0.35	0.759	0.431
	LOBSTER	0.571	0.801	0.606
	IMBS	0	0	0

Tabla 4.9: Comparativa de los resultados obtenidos tras la evaluación de regiones estáticas

estático a partir de 6 máscaras de frente equiespaciadas en el tiempo, el acumulador requiere que la región estática no varíe durante un número determinado de máscaras de frente consecutivas. Por el contrario, por este mismo motivo, el algoritmo de submuestreo de máscaras de frente, en ocasiones mantiene la detección de regiones estáticas una vez el elemento de frente ha abandonado su posición espacial en el cuadro de vídeo derivando en la detección de falsos positivos.

#### 4.4. Evaluación de las técnicas de clasificación robo-abandono

En esta sección se han evaluado los algoritmos de discriminación robo-abandono contraste de color de píxel y gradiente alto descritos en el apartado 3.3.4 utilizando la combinación de algoritmos LOBSTER y submuestreo de máscaras de frente en las etapas de sustracción de fondo y detección de regiones estáticas al ser la configuración que mejores resultados obtiene como se ha demostrado en el apartado 4.3.3. Se ha utilizado el detector de personas histograma de gradientes descrito en el apartado 3.3.3, ya que se considera el más adecuado para las secuencias evaluadas.

Al igual que en la sección 4.3, derivado de la acumulación de errores de las etapas anteriores del sistema, tras realizar una primera evaluación de los algoritmos frente al ground-truth propuesto en la que se han utilizado criterios únicamente objetivos se ha obtenido una evaluación no realista. Por lo tanto la evaluación presentada a continuación se ha realizado, en base a los criterios del ground-truth más la interpretación del ser humano en los casos ambiguos.

Para aquellas secuencias que no presentan el evento de robo de objetos, se ha tomado la decisión de no evaluar las métricas correspondientes a la detección de robos y no han sido computadas al calcular los valores promedio de las mimas.

Secuencia	# abandonos		# robos		P-ab	R-ab	F-ab	P-robo	R-robo	F-robo	P	R	F
	detectados	correctos	detectados	correctos									
Office_indoor	3	3	3	-	1	1	1	-	-	-	0.5	1	0.666
S4-T5-C3	17	1	4	-	0.058	1	0.109	-	-	-	0.047	1	0.089
AB_Easy	13	4	5	0	0.307	1	0.469	0	0	0	0.222	0.571	0.319
1.09_abandoned_object	1	1	2	0	1	1	1	0	0	0	0.333	0.5	0.399
1.12_abandoned_object	1	0	1	0	0	0	0	0	0	0	0	0	0
LeftBag_AtChair	1	0	4	0	0	0	0	0	0	0	0	0	0
LeftBag_PickedUp	3	1	3	1	0.333	1	0.499	0.333	1	0.499	0.333	1	0.499
AVERAGE					0.385	0.714	0.439	0.066	0.2	0.099	0.205	0.581	0.281

Tabla 4.10: Resultados de la discriminación robo-abandono de objetos utilizando el algoritmo Pixel Color Contrast.

#### 4.4.1. Evaluación de clasificación robo-abandono mediante contraste de color de píxel

En la Tabla 4.10 se muestran los resultados obtenidos tras la evaluación de discriminación robo-abandono mediante el contraste de color de píxel. Los falsos negativos correspondientes a la detección de abandono de objetos son derivados de la acumulación de errores de las etapas de sustracción de fondo y detección de regiones estáticas. Por el contrario, los falsos negativos correspondientes a la detección de robo de objetos se debe a la corta duración de las secuencias evaluadas y la lenta tasa de actualización del algoritmo de fondo. El tiempo que permanecen abandonados los objetos no es suficiente para que sean absorbidos por el fondo y por lo tanto detectar como robo su extracción de la escena. Los errores producidos por el detector de personas al no categorizar correctamente aquellas personas estáticas por su apariencia en la secuencia conllevan que sus regiones estáticas sean detectadas como objetos abandonados aumentando el número de falsos positivos.

Este algoritmo presenta dificultades para categorizar objetos muy pequeños puesto que resulta complicado realizar el contraste de colores entre el interior y el exterior de la región estática, y no presenta robustez frente a oclusiones, ya que si un objeto categorizado como abandonado correctamente se ve ocluido por un elemento de frente en movimiento con tonalidades de color distintas a la del objeto, se categoriza como robo durante la oclusión para posteriormente volver a detectarse como abandonado. Cabe destacar que los falsos positivos detectados como robo también son derivados de los errores del modelo de fondo, que contiene aquellos elementos de frente presentes en el primer frame de la secuencia y posteriormente abandonan la escena.

Secuencia	# abandonos		# robos		P-ab	R-ab	F-ab	P-robo	R-robo	F-robo	P	R	F
	detectados	correctos	detectados	correctos									
Office_indoor	3	3	0	-	1	1	1	-	-	-	1	1	1
S4-T5-C3	7	1	0	-	0.142	1	0.248	-	-	-	0.142	1	0.248
AB_Easy	13	4	0	0	0.307	1	0.469	0	0	0	0.307	0.571	0.399
1.09_abandoned_object	2	1	0	0	0.5	1	0.666	0	0	0	0.5	0.5	0.5
1.12_abandoned_object	1	0	0	0	0	0	0	0	0	0	0	0	0
LeftBag_AtChair	6	1	0	0	0.166	1	0.285	0	0	0	0.166	0.5	0.249
LeftBag_PickedUp	4	1	0	0	0.25	1	0.4	0	0	0	0.25	0.5	0.333
AVERAGE					0.337	0.857	0.438	0	0	0	0.337	0.581	0.389

Tabla 4.11: Resultados de la discriminación robo-abandono de objetos utilizando el algoritmo GradientH

#### 4.4.2. Evaluación de clasificación robo-abandono mediante gradiente alto

En la Tabla 4.11 se muestran los resultados obtenidos tras la evaluación de discriminación robo-abandono mediante gradiente alto. Los falsos negativos correspondientes a la detección de abandono de objetos son derivados de la acumulación de errores de las etapas de sustracción de fondo y detección de regiones estáticas. Por el contrario, los falsos negativos correspondientes a la detección de robo de objetos se debe a la corta duración de las secuencias evaluadas y la lenta tasa de actualización del algoritmo de fondo. El tiempo que permanecen abandonados los objetos no es suficiente para que sean absorbidos por el fondo y por lo tanto detectar como robo su extracción de la escena. Los errores producidos por el detector de personas al no categorizar correctamente aquellas personas estáticas por su apariencia en la secuencia conllevan que sus regiones estáticas sean detectadas como objetos abandonados aumentando el número de falsos positivos.

Cabe destacar que este algoritmo categoriza como objetos abandonados los errores del modelo de fondo derivados de aquellos elementos de frente presentes en el primer frame de la secuencia y posteriormente abandonan la escena. Se supone que este error es provocado por adaptación del algoritmo a la versión 3.0 de OpenCV, ya que originalmente fue implementado en la versión 2.4.3 y hacía uso de una función de la librería suprimida en la nueva versión para ajustar los contornos de las regiones estáticas.

#### 4.4.3. Comparativa y conclusiones de los resultados obtenidos

En la Tabla 4.12 se muestran los resultados promedio de los experimentos realizados. Como se observa en la tabla ambos algoritmos obtienen resultados muy similares. La detección de eventos de robo y abandono es muy dependiente de los parámetros que modulan la absorción de elementos del frente estático por parte del modelo de

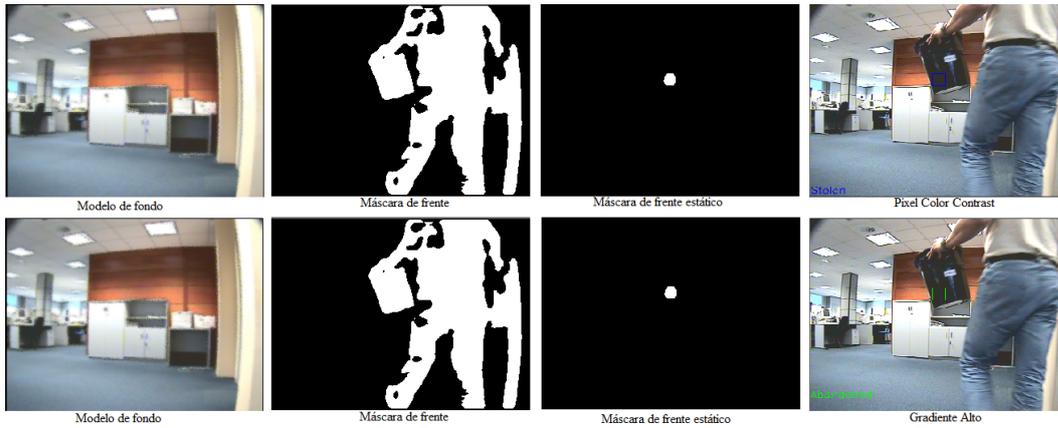


Figura 4.6: Comparación de los resultados obtenidos por ambos clasificadores frente a situaciones de oclusión.

Discriminador	P-ab	R-ab	F-ab	P-robo	R-robo	F-robo	P	R	F
PCC	0.385	0.714	0.439	0.066	0.2	0.099	0.205	0.581	0.281
GradientH	0.337	0.857	0.438	0	0	0	0.337	0.581	0.389

Tabla 4.12: Comparativa de los resultados obtenidos tras la evaluación de discriminación robo-abandono

fondo. Puesto que la discriminación robo-abandono de objetos es la última etapa del sistema de análisis de vídeo, sus resultados se ven afectados por la acumulación de errores de las etapas anteriores. Aunque los resultados de ambos algoritmos sean similares, cabe destacar que discriminador basado en el contraste de color produce falsos positivos derivados de las oclusiones sufridas por las regiones estáticas por elementos del frente en movimiento, mientras que el discriminador basado en gradientes presenta robustez frente a dichas oclusiones por lo que la elección de uno u otro se vendrá influenciada por las características de la secuencia que se desee analizar.

La Figura 4.6 muestra la comparación de ambos clasificadores frente a situaciones de oclusión de los objetos de interés.

La Figura 4.7 muestra la comparación de ambos clasificadores frente a situaciones en las que se producen errores en el modelo de fondo.

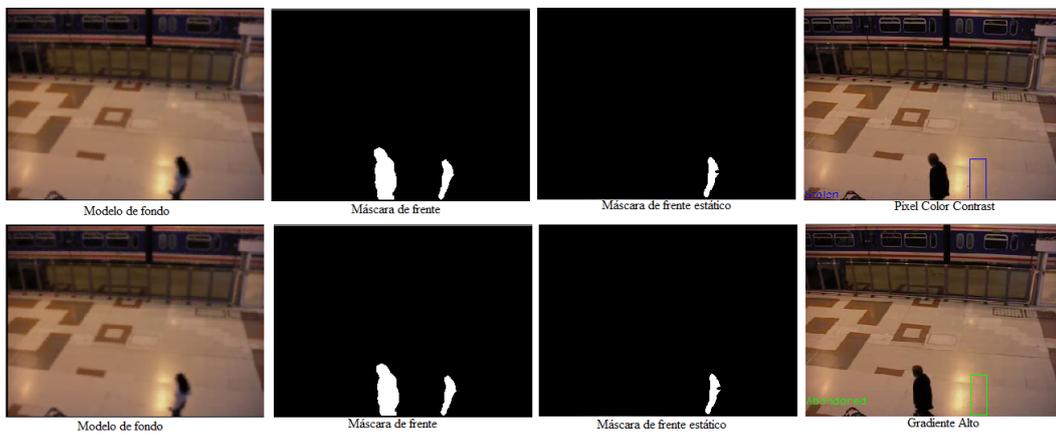


Figura 4.7: Comparación de los resultados obtenidos por ambos clasificadores frente a situaciones de en las que se producen errores de fondo



## Capítulo 5

# Conclusiones y trabajo futuro.

### 5.1. Conclusiones.

En este PFC se ha desarrollado un sistema de análisis de vídeo para la detección de objetos abandonados y robados configurable y escalable. 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 cuatro partes claramente diferenciadas: Sustracción de fondo, detección de regiones estáticas, clasificación objeto-persona y discriminación robo-abandono. Tanto para la sustracción de fondo como para la detección de regiones estáticas, se realizó una comparación y clasificación de las características que presentan las aproximaciones de la literatura.

A continuación se implementó el sistema, mediante la integración de las técnicas más relevantes de la literatura y las más actuales desarrolladas recientemente. Se implementaron las clases que encapsulan las técnicas de cada etapa así como las interfaces y funciones adicionales que permiten su ejecución de manera secuencial. Este sistema es escalable y permite la integración de nuevos algoritmos en cada una de sus etapas en un futuro. En paralelo a la implementación del sistema, se realizaron las anotaciones de los ground-truth de las secuencias de los dataset disponibles con el objetivo de evaluar distintas configuraciones del sistema en última instancia.

Por último se ha llevado a cabo la evaluación de los módulos de detección de regiones estáticas y discriminación robo-abandono sobre una serie de secuencias de distinta dificultad. La detección de eventos de robo y abandono es muy dependiente de los parámetros que modulan la absorción de elementos del frente estático por parte del modelo de fondo. Puesto que la discriminación robo-abandono de objetos es la última etapa del sistema de análisis de vídeo, sus resultados se ven afectados por la acumulación de errores de las etapas anteriores. La detección de objetos en entornos

muy poblados sigue siendo un reto de cara a futuras líneas de investigación. La etapa de clasificación de persona-objeto tiene mucho trabajo por desarrollar, y será de gran ayuda para reducir la cantidad de falsos positivos introducidos en el sistema.

## 5.2. Trabajo futuro.

Tras la implementación del sistema desarrollado en este trabajo, surgen las siguientes líneas de trabajo futuro con el objetivo de ampliar y mejorar la funcionalidad del mismo así como realizar una evaluación más exhaustiva de las técnicas integradas:

- Diseño de una interfaz gráfica para la configuración del sistema desarrollado: Los sistemas de análisis de vídeo se diseñan con el objetivo de proporcionar herramientas de análisis automáticas y ayudar al personal de seguridad. Debido a que la guía de configuración del sistema desarrollado planteada en el apartado 3.2.2 puede resultar tediosa para una persona no familiarizada con los entornos de programación se propone la implementación de una interfaz gráfica intuitiva que permita al usuario configurar el sistema sin dificultad.
- Integración del sistema desarrollado en la plataforma de análisis distribuido DiVA sobre cámaras reales: El fin último de la implementación de un sistema de análisis de vídeo para la detección de objetos abandonados y robados es formar parte de un sistema de vídeo-vigilancia y ser capaz de detectar los eventos que tengan lugar y disparar las alarmas a tiempo real.
- Configuración de las técnicas de sustracción de fondo en la clase MyDualBGS-Selector para el correcto funcionamiento de la técnica de detección de regiones estáticas DBM: Actualmente, la técnica de detección de regiones estáticas Dual Background Model únicamente funciona correctamente con el algoritmo de sustracción de fondo Gaussian Mixture Model. Las aproximaciones de sustracción de fondo integradas en el sistema, han sido configuradas con los parámetros recomendados por los autores de los algoritmos. Dual Background Model requiere dos modelos de fondo actualizados a diferentes velocidades y no se han planteado las configuraciones de los algoritmos de sustracción de fondo necesarias para obtener los modelos de fondo de absorción rápida y absorción lenta.
- Creación de nuevos dataset: Debido a que los dataset actuales de los que disponemos son insuficientes, se propone la creación de un conjunto de datos representativo de condiciones reales (reflejos/sombras, multitudes, operación continuada, etc).

- Ampliación de los experimentos realizados: Por motivos de tiempo, no se ha evaluado el módulo de detección de personas del sistema de análisis de vídeo para la detección de objetos abandonados y robados implementado. A su vez, el número de secuencias con el que se han evaluado los módulos de detección de regiones estáticas y discriminación robo-abandono es reducido y no se han evaluado todas las configuraciones posibles del sistema. Por lo que surge la necesidad de aumentar el número de experimentos realizados para realizar una evaluación más exhaustiva.







# Bibliografía

- [1] KN Plataniotis and CS Regazzoni. Visual-centric surveillance networks and services. *IEEE Signal Processing Magazine*, 22(2):12–15, 2005. [1](#)
- [2] Silvia Ferrando, Gianluca Gera, and Carlo Regazzoni. Classification of unattended and stolen objects in video-surveillance system. In *2006 IEEE International Conference on Video and Signal Based Surveillance*, pages 21–21. IEEE, 2006. [1](#), [43](#)
- [3] Jia-Ming Liang Chung-Shuo Fan and Yi-Ting Lin. A survey of intelligent video surveillance systems: History, applications and future. pages 1479–1488, 2015. [5](#)
- [4] Diego Ortego and Juan C SanMiguel. Stationary foreground detection for video-surveillance based on foreground and motion history images. In *Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on*, pages 75–80. IEEE, 2013. [9](#), [32](#), [39](#)
- [5] Álvaro Bayona, Juan C SanMiguel, and José M Martínez. Stationary foreground detection using background subtraction and temporal difference in video surveillance. In *2010 IEEE International Conference on Image Processing*, pages 4657–4660. IEEE, 2010. [9](#)
- [6] Rui Wang, Filiz Bunyak, Guna Seetharaman, and Kannappan Palaniappan. Static and moving object detection using flux tensor with split gaussian models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 414–418, 2014. [9](#), [10](#), [15](#), [16](#), [25](#), [28](#)
- [7] Grzegorz Szwoch. Extraction of stable foreground image regions for unattended luggage detection. *Multimedia Tools and Applications*, 75(2):761–786, 2016. [9](#), [10](#), [15](#), [16](#), [25](#), [27](#)
- [8] Kahlil Muchtar, Chih-Yang Lin, and Chia-Hung Yeh. Grabcut-based abandoned object detection. In *Multimedia Signal Processing (MMSP), 2014 IEEE 16th International Workshop on*, pages 1–6. IEEE, 2014. [9](#), [10](#), [15](#), [16](#), [25](#), [27](#)
- [9] Quanfu Fan, Sharath Pankanti, and Lisa Brown. Long-term object tracking for parked vehicle detection. In *Advanced Video and Signal Based Surveillance (AVSS), 2014 11th IEEE International Conference on*, pages 223–229. IEEE, 2014. [9](#), [10](#), [15](#), [16](#), [17](#), [20](#), [25](#), [27](#)
- [10] Waqas Hassan, Philip Birch, Bhargav Mitra, Nagachetan Bangalore, Rupert Young, and Chris Chatwin. Illumination invariant stationary object detection. *IET Computer Vision*, 7(1):1–8, 2013. [9](#), [10](#), [15](#), [16](#), [17](#), [25](#), [27](#)

- [11] James Ferryman, David Hogg, Jan Sochman, Ardhendu Behera, José A Rodriguez-Serrano, Simon Worgan, Longzhen Li, Valerie Leung, Murray Evans, Philippe Cornic, et al. Robust abandoned object detection integrating wide area visual surveillance and social context. *Pattern Recognition Letters*, 34(7):789–798, 2013. [9](#), [10](#), [15](#), [16](#), [17](#), [25](#), [27](#)
- [12] YingLi Tian, Andrew Senior, and Max Lu. Robust and efficient foreground analysis in complex surveillance videos. *Machine Vision and Applications*, 23(5):967–983, 2012. [9](#), [10](#), [15](#), [16](#), [17](#), [25](#), [28](#)
- [13] YingLi Tian, Rogerio Schmidt Feris, Haowei Liu, Arun Hampapur, and Ming-Ting Sun. Robust detection of abandoned and removed objects in complex surveillance videos. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(5):565–576, 2011. [9](#), [10](#), [15](#), [16](#), [17](#), [25](#), [27](#)
- [14] Rubén Heras Evangelio and Thomas Sikora. Static object detection based on a dual background model and a finite-state machine. *EURASIP Journal on Image and Video Processing*, 2011(1):1, 2010. [9](#), [10](#), [16](#), [25](#), [27](#)
- [15] Jiman Kim and Daijin Kim. Static region classification using hierarchical finite state machine. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2358–2362. IEEE, 2014. [10](#), [15](#), [16](#), [25](#), [28](#)
- [16] Jiman Kim and Bongnam Kang. Nonparametric state machine with multiple features for abnormal object classification. In *Advanced Video and Signal Based Surveillance (AVSS), 2014 11th IEEE International Conference on*, pages 199–203. IEEE, 2014. [10](#), [15](#), [16](#), [25](#), [28](#)
- [17] Jaemyun Kim, Adin Ramirez Rivera, Byungyong Ryu, Kiok Ahn, and Oksam Chae. Unattended object detection based on edge-segment distributions. In *Advanced Video and Signal Based Surveillance (AVSS), 2014 11th IEEE International Conference on*, pages 283–288. IEEE, 2014. [10](#), [15](#), [16](#), [25](#), [27](#)
- [18] Lucia Maddalena and Alfredo Petrosino. Stopped object detection by learning foreground model in videos. *IEEE transactions on neural networks and learning systems*, 24(5):723–735, 2013. [10](#), [15](#), [16](#), [17](#), [25](#), [27](#)
- [19] Csaba Beleznai, Peter Gemeiner, and Christian Zinner. Reliable left luggage detection using stereo depth and intensity cues. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 59–66, 2013. [10](#), [15](#), [16](#), [27](#)
- [20] Thi Thi Zin, Pyke Tin, Hiromitsu Hama, and Takashi Toriu. Unattended object intelligent analyzer for consumer video surveillance. *IEEE Transactions on Consumer Electronics*, 57(2):549–557, 2011. [10](#), [11](#), [15](#), [16](#), [25](#), [27](#)
- [21] Jiyang Pan, Quanfu Fan, and Sharath Pankanti. Robust abandoned object detection using region-level analysis. In *2011 18th IEEE International Conference on Image Processing*, pages 3597–3600. IEEE, 2011. [10](#), [11](#), [15](#), [16](#), [17](#), [25](#), [27](#)

- [22] Lucia Maddalena and Alfredo Petrosino. The 3dsobs+ algorithm for moving object detection. *Computer Vision and Image Understanding*, 122:65–73, 2014. 10
- [23] Lucia Maddalena and Alfredo Petrosino. The sobs algorithm: what are the limits? In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 21–26. IEEE, 2012. 10, 37
- [24] Jesús Martínez-del Rincón, J Elías Herrero-Jaraba, J Raúl Gómez, and Carlos Orrite-Urunuela. Automatic left luggage detection and tracking using multi-camera ukf. In *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS'06)*, pages 59–66, 2006. 12
- [25] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999. 12
- [26] Juan Carlos San Miguel and José M Martínez. Robust unattended and stolen object detection by fusing simple algorithms. In *Advanced Video and Signal Based Surveillance, 2008. AVSS'08. IEEE Fifth International Conference on*, pages 18–25. IEEE, 2008. 12, 21, 32, 43, 51, 55, 57, 58
- [27] Diego Ortego and Juan C SanMiguel. Multi-feature stationary foreground detection for crowded video-surveillance. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2403–2407. IEEE, 2014. 12, 32
- [28] Álvaro Bayona, Juan Carlos SanMiguel, and José M Martínez. Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques. In *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*, pages 25–30. IEEE, 2009. 12, 14, 40, 41
- [29] Kevin Lin, Shen-Chi Chen, Chu-Song Chen, Daw-Tung Lin, and Yi-Ping Hung. Left-luggage detection from finite-state-machine analysis in static-camera videos. In *ICPR*, pages 4600–4605, 2014. 15, 16, 25, 27
- [30] Jing-Ying Chang, Huei-Hung Liao, and Liang-Gee Chen. Localized detection of abandoned luggage. *EURASIP Journal on Advances in Signal Processing*, 2010(1):1, 2010. 18, 20
- [31] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):743–761, 2012. 18
- [32] Víctor Fernández-Carbajales, Miguel Ángel García, and José M Martínez. Robust people detection by fusion of evidence from multiple methods. Institute of Electrical and Electronics Engineers, 2008. 18
- [33] Mohamed Hussein, Wael Abd-Almageed, Yang Ran, and Larry Davis. Real-time human detection, tracking, and verification in uncontrolled camera motion environments. In *Fourth IEEE International Conference on Computer Vision Systems (ICVS'06)*, pages 41–41. IEEE, 2006. 18

- [34] Yu Huang and Thomas S Huang. Model-based human body tracking. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 1, pages 552–555. IEEE, 2002. 18
- [35] Roland Kehl and Luc Van Gool. Markerless tracking of complex human motions from multiple views. *Computer Vision and Image Understanding*, 104(2):190–209, 2006. 18
- [36] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. People-tracking-by-detection and people-detection-by-tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 18
- [37] Neeti A Ogale. A survey of techniques for human detection from video. *Survey, University of Maryland*, 125:133, 2006. 19
- [38] Emilio Maggio and Andrea Cavallaro. *Video tracking: theory and practice*. John Wiley & Sons, 2011. 20
- [39] Hanxuan Yang, Ling Shao, Feng Zheng, Liang Wang, and Zhan Song. Recent advances and trends in visual tracking: A review. *Neurocomputing*, 74(18):3823–3831, 2011. 20
- [40] Hui Kong, Jean-Yves Audibert, and Jean Ponce. Detecting abandoned objects with a moving camera. *IEEE Transactions on Image Processing*, 19(8):2201–2210, 2010. 20
- [41] Medha Bhargava, Chia-Chih Chen, Michael S Ryoo, and Jake K Aggarwal. Detection of object abandonment using temporal logic. *Machine Vision and Applications*, 20(5):271–281, 2009. 20
- [42] Luis Caro Campos, Juan Carlos SanMiguel, and José M Martínez. Discrimination of abandoned and stolen object based on active contours. In *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, pages 101–106. IEEE, 2011. 20
- [43] Péter L Venetianer, Zhong Zhang, Weihong Yin, and Alan J Lipton. Stationary target detection using the objectvideo surveillance system. In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 242–247. IEEE, 2007. 21
- [44] Wu-Chih Hu, Huang Deng Yuan, Wei-Hao Chen, et al. Adaptive wide field-of-view surveillance based on an ip camera on a rotational platform for automatic detection of abandoned and removed objects. *ICIC EXPRESS LETTERS 1 (1):*, 45(50), 2010. 21
- [45] Juan Carlos Sanmiguel, L Caro, and José M Martínez. Pixel-based colour contrast for abandoned and stolen object discrimination in video surveillance. *Electronics letters*, 48(2):86–87, 2012. 21, 43, 44
- [46] Chathuranga Hettiarachchi, Asitha Nanayakkara, Ayesha Dissanayaka, Charith Wijenayake, and Chathura De Silva. Abandoned object detection with logical reasoning. In *Advance Computing Conference (IACC), 2014 IEEE International*, pages 1137–1141. IEEE, 2014. 25, 27

- [47] Jack K Fitzsimons and Thomas T Lu. Markov random fields for static foreground classification in surveillance systems. In *SPIE Optical Engineering+ Applications*, pages 92171O–92171O. International Society for Optics and Photonics, 2014. 25, 28
- [48] Pierre-Luc St-Charles and Guillaume-Alexandre Bilodeau. Improving background subtraction using local binary similarity patterns. In *IEEE Winter Conference on Applications of Computer Vision*, pages 509–515. IEEE, 2014. 37
- [49] Domenico D Bloisi, Andrea Pennisi, and Luca Iocchi. Background modeling in the maritime domain. *Machine vision and applications*, 25(5):1257–1269, 2014. 37
- [50] Domenico D Bloisi, Alfonso Grillo, Andrea Pennisi, Luca Iocchi, and Claudio Passaretti. Multi-modal background model initialization. In *International Conference on Image Analysis and Processing*, pages 485–492. Springer, 2015. 37
- [51] Pierre-Luc St-Charles, Guillaume-Alexandre Bilodeau, and Robert Bergevin. A self-adjusting approach to change detection based on background word consensus. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 990–997. IEEE, 2015. 38
- [52] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE, 2004. 38
- [53] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780, 2006. 38
- [54] Ruben Heras Evangelio, Michael Patzold, Ivo Keller, and Thomas Sikora. Adaptively splitted gmm with feedback improvement for the task of background subtraction. *IEEE transactions on information forensics and security*, 9(5):863–874, 2014. 38
- [55] Sadiye Guler, Jason A Silverstein, and Ian H Pushee. Stationary objects in multiple object tracking. In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 248–253. IEEE, 2007. 39
- [56] D. Ortego. Detección de objetos estáticos de primer plano en escenarios altamente concurridos de vídeo seguridad. 2013. 41
- [57] Fatih Porikli, Yuri Ivanov, and Tetsuji Haga. Robust abandoned object detection using dual foregrounds. *EURASIP Journal on Advances in Signal Processing*, 2008(1):1–11, 2007. 40
- [58] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005. 42
- [59] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings*

*of the 2001 IEEE Computer Society Conference on*, volume 1, pages I-511. IEEE, 2001.  
42

- [60] A. Bayona. Detección de objetos abandonados/robados en secuencias de vídeo-seguridad. 2009. 43, 45

# Apéndice A

## Presupuesto

### 1. Ejecución Material

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

### 2. Gastos generales

- 16 % sobre Ejecución Material.....352 €

### 3. Beneficio Industrial

- 6 % sobre Ejecución Material.....132 €

### 4. Honorarios Proyecto

- 1800 horas a 15 € / hora ..... 27.000 €

### 5. Material fungible

- Gastos de impresión.....280 €
- Encuadernación ..... 200 €

### 6. Subtotal del presupuesto

- Subtotal Presupuesto.....32.774 €

**7. I.V.A. aplicable**

- 21 % Subtotal Presupuesto.....6.882,5 €

**8. Total presupuesto**

---

- Total Presupuesto..... 39.656,5 €

Madrid, julio 2016

El Ingeniero Jefe de Proyecto

Fdo.: Jorge Gómez Vicente

Ingeniero de Telecomunicación

## Apéndice B

# Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de integración y evaluación de sistemas de robo-abandono de objetos en vídeo. 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 entorno. 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 de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.
6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.
7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.
8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.
9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.
10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.
11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado

mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.
13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.
14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.
15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.
16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.
17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.
18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.
19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.
20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean

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

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.
22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.
23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

### **Condiciones particulares**

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

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.
2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.
4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.
5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.
6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.
7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.
8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.
9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.
10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.
11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.
12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.

