

# **DETECCIÓN DE TUMULTOS CON FILTROS DE PARTÍCULAS**

Ramón Llorca Queralt

Julio 2008

ETSETB

Directores de proyecto

Josep Ramon Casas

Cristian Cantón



# Resumen

---

A raíz de los incidentes acontecidos en los últimos años en grandes espacios públicos, se ha experimentado un importante aumento de la concienciación de cara a mejorar la seguridad en este tipo entornos. Por lo tanto, ha crecido el interés en desarrollar tecnologías capaces de crear sistemas punteros de seguridad, video vigilancia y control de operaciones en este tipo de escenarios.

Este trabajo pretende aportar una nueva manera de detección de situaciones potencialmente peligrosas, que llamaremos tumultos, acaecidas en grandes espacios públicos, concretamente en aeropuertos. Nuestro principal objetivo es poder estimar el flujo de movimiento de las personas en las rampas para posteriormente detectar cuando éste se detiene y decidir si es debido a una situación de tumulto. Esto se llevará a cabo a partir del análisis, etiquetado y estudio de numerosas secuencias tomadas desde cámaras ubicadas en el aeropuerto de Madrid, específicamente de la Terminal 4. De este modo, indicaremos cuando una situación de este tipo tenga lugar mediante algún tipo de *alarma* y así, poder prevenir los daños que ésta pueda causar.

Para llevar a cabo esta estimación, en primer lugar, deberemos desarrollar algún modo de seguimiento que nos permita extraer el flujo de movimiento. Existen, a grandes rasgos, dos tipos de enfoques con los que se aborda el problema del seguimiento. El primero se centra en descomponer la escena y efectuar asignaciones con los candidatos del siguiente fotograma a través del uso de criterios de semejanza. Por otro lado, el segundo enfoque pretende estimar la nueva posición de cada objetivo a través de la información a priori, las observaciones disponibles y un modelo dinámico del proceso. La estrategia que se desarrolla en este trabajo corresponde al segundo grupo y, concretamente, la herramienta empleada para su realización es el filtro de partículas. Gracias al aumento de la capacidad de procesado, es hoy por hoy, totalmente factible la implementación de un seguimiento basado en filtros de partículas y, en consecuencia, el uso de esta herramienta se encuentra en auge dentro del procesado de imagen. El funcionamiento de los filtros de partículas consiste en una población de partículas o muestras, que se propagan siguiendo una dinámica determinada, de manera que sobrevivan aquéllas que se encuentren en las zonas cuya verosimilitud de ubicarse donde esté el objetivo que se pretende seguir sea mayor. De este modo, iterando este funcionamiento se logra estimar la posición del objeto que se desea seguir y se realiza el seguimiento.

El diseño propuesto combina los elementos citados con contribuciones significativas logrando alcanzar mejoras notables en los resultados obtenidos. Las principales contribuciones permiten adaptar el seguimiento a un entorno con perspectiva y mejorar los resultados en un escenario aglomerado de personas.

En un futuro, la idea sería llevar a cabo su implementación en tiempo real de manera que se pudieran llegar a detectar automáticamente este tipo de situaciones, y con ello, mejorar la seguridad en las rampas de los aeropuertos o en entornos cuyas características sean similares.

# Agradecimientos

---

A toda mi familia, en especial a mis padres y a mi hermano, por aguantar este largo periodo.

Al Josep Ramon y al Cristian por orientarme y brindarme su ayuda con interés y dedicación durante todo el proyecto y por la multitud de reuniones, incluso videoconferencias, que tuvimos sin las que este trabajo no hubiera sido posible.

Al Albert, al Jaime, al Adolfo, al Marcel, al Jose y al Nico porque vieron como se gestaba la criatura desde el d5 y me ayudaron siempre que fue necesario.

A la gente de Revistes, a todos, pero en especial a Kescuin, Nico, Will, Pitifli, Mike, Dani, Román, Pedro, Judith, Jordi, Frondy, Segu, Carmona, Samoano, Arnau y Sonia, por su apoyo y ánimos sobretodo en la parte final.

A la gente del IEEE, por ocupar su despacho durante largos periodos de trabajo.

A los de siempre, que han visto como se iba formando lentamente la criatura, en especial a Pau, por su inestimable ayuda, y al resto, Guillem, Víctor, Lucas, Carlos y Sergi.

A los futboleros, desde el Telecos D, hasta el Q0 y Q33 pasando por el Monastic, que sin saberlo me sirvió de ayuda para desconectar.

A vosotras dos, por los bonitos momentos pasados.

Y al resto, gracias.

Ramón Llorca Queralt

Julio 2008

*“No entiendes realmente algo a menos que seas capaz de explicárselo a tu abuela.”*

Albert Einstein

*“Mi tío solía decir que queremos a las personas por sus cualidades  
pero que las amamos por sus defectos.”*

Hellboy

*“Alégrame el día.”*

Harry el Sucio

# ÍNDICE

---

1.- Introducción .....	8
1.1.- Escenario:.....	8
1.3.- Estado del arte: .....	13
1.4.- Organización: .....	16
2.- Seguimiento mediante filtros de partículas.....	17
2.1.- Introducción a los filtros de partículas.....	17
2.1.1.- Seguimiento Bayesiano.....	18
2.1.2.- El filtro de Kalman.....	19
2.1.3.- El filtro de partículas .....	21
2.2.- Dinámica del seguimiento:.....	28
3.- Implementación del filtro de partículas.....	30
3.1.- Extracción de características para la realización del seguimiento:.....	31
3.1.1.- Extracción de primer plano .....	32
Running Gaussian Average.....	32
Fondo adaptativo .....	34
Espacio de color .....	37
3.1.2.- Color: histograma del objeto a seguir.....	41
3.2.- Propagación de los filtros de partículas en un entorno con perspectiva: .....	43
3.2.1.- Caracterización de los filtros de partículas según el entorno .....	43
3.2.2.- Creación y destrucción de los filtros de partículas .....	45
3.2.3.- Propagación de los filtros adaptada a la perspectiva .....	47
4.- Seguimiento en un escenario aglomerado y detección de tumultos .....	50
4.1.- Adaptación al seguimiento con múltiples targets: .....	50
4.1.1.- Criterios de prevención de un mal seguimiento:.....	50
Recta de bloqueo .....	51
Bloqueo de las partículas que sobrepasan la salgan de la rampa.....	52
Alejamiento excesivo de las partículas .....	53
4.1.2.- Criterios de destrucción de los filtros de partículas: .....	55
Baja eficiencia.....	55
Dispersión explosiva.....	56
Solapamiento de filtros .....	59
Retroceso .....	60
4.2.- Clasificación y detección de tumultos:.....	62
4.2.1.- Caracterización de un tumulto.....	62
4.2.2.- Etiquetado manual de las secuencias .....	63
4.2.3.- Métodos para la detección de tumultos.....	63
Estudio de la velocidad de los filtros de partículas frame a frame: .....	64
A partir de la aglomeración de centroides de los filtros en función de su posición y velocidad: .....	69

4.2.4.- Métricas empleadas para determinar la calidad de la detección de tumultos .....	74
Definición del Ground Truth: .....	74
Detección por zonas (NT – T – NT):.....	75
Parámetros que determinarán la calidad del seguimiento:.....	75
 5.- Resultados y evaluación.....	78
5.1.- Resultados obtenidos con las distintas métricas empleadas:.....	78
5.1.1.- Tasa de detección: .....	80
5.1.2.- Tasa de falsas alarmas: .....	81
5.1.3.- Cálculo del punto de trabajo óptimo a partir del ROC:.....	82
5.3.- Resultados globales para las 100 secuencias:.....	84
 6.- Conclusiones .....	87
6.1.- Cumplimiento de los objetivos .....	87
6.2.- Futuras líneas de trabajo .....	89
 TABLA DE ILUSTRACIONES .....	90
 LISTA DE TABLAS.....	96
 BIBLIOGRAFIA.....	97
 ANEXOS .....	99
A.1.- Toy example:.....	99
A.2.- Espacio de color YCbCr y generación de histogramas con YCb e YCr:.....	106
A.3.- Caracterización de los escenarios:.....	108
A.3.1.- Escenario 1:.....	109
A.3.2.- Escenario 2:.....	112
A.3.3.- Escenario 3:.....	115
A.3.4.- Escenario 4:.....	118
A.4.- Filtro Morfológico (Post-procesado): .....	121
A.5.- Resultados obtenidos tras el cálculo de la tasa de falsas alarmas en las 100 secuencias: .....	123

# 1.- Introducción

---

## 1.1.- Escenario:

En los últimos años ha crecido el interés en desarrollar tecnologías que permitan la creación de sistemas punteros de seguridad, video vigilancia y control de operaciones en infraestructuras y espacios públicos. Es por este motivo, que desde el CDTI, un organismo adscrito al Ministerio de Industria, Turismo y Comercio, se creó, en 2005, un programa de Consorcios Estratégicos Nacionales en Investigación Técnica (CENIT), con el principal objetivo de fomentar la cooperación público privada en I+D+i mediante la financiación de proyectos conjuntos de investigación industrial.

Uno de los proyectos impulsado por este organismo recibe el nombre de  **proyecto Hesperia**, cuya principal finalidad es dar respuesta a la demanda existente de sistemas de gestión integrada de seguridad y control de operaciones que se puedan implantar de manera rentable en el mercado, algo que de momento no existe. Este proyecto está integrado por multitud de empresas y universidades entre las que se encuentra la Universidad Politécnica de Cataluña (UPC).

Dentro del proyecto Hesperia, se tratan las tecnologías que se implantarán en dos tipos de escenarios. Por un lado, las que permitirán gestionar la seguridad y las operaciones de infraestructuras públicas especialmente sensibles, como subestaciones eléctricas, de gas, depósitos de agua o estaciones de telecomunicaciones. Por otro, las que incrementarán de forma sustancial los niveles de seguridad de grandes espacios públicos, como aeropuertos, estaciones de ferrocarril, puertos, centros de ciudades especialmente en zonas peatonales, centros comerciales, etc.

Es en este último tipo de escenarios donde se encuadra este trabajo. Así pues el presente trabajo se ubica dentro de los proyectos que realizan un seguimiento de los objetos que pueden llegar a ser peligrosos, reconocimientos de caras y seguimiento de flujos de personas en espacios públicos de grandes dimensiones y con una gran afluencia de gente. Concretamente, nuestro principal objetivo es el de seguir el flujo de personas para poder detectar una situación potencialmente peligrosa para las personas con el fin de poder avisar inmediatamente cuando ésta tenga lugar. De este modo se pretende poder minimizar los daños causados a raíz de una situación de este tipo y así aumentar la seguridad en estos entornos.

El escenario en el que trabajaremos serán las terminales de pasajeros de los aeropuertos, en particular la Terminal 4 del aeropuerto de Madrid. Las secuencias empleadas en este proyecto corresponden a las grabaciones tomadas por 4 cámaras fijas de la terminal. Las cuatro cámaras muestran visiones distintas de dos rampas diferentes.

En primer lugar tenemos las cámaras PRT7A-Camera1 y PRT7A-Camera2 que están asociadas a la misma rampa pero enfocándola desde extremos opuestos. Veamos una imagen de estas cámaras para tener una idea de las características que presenta el escenario con el que trabajaremos:



**Fig. 1. 1.- Vista del escenario 1, el que muestra la cámara PRT7A-Camera1.**



**Fig. 1. 2.- Vista del escenario 4, el que muestra la cámara PRT7A-Camera2.**

En segundo lugar tenemos las cámaras PRT24B-Camera1 y PRT24B-Camera2 que también enfocan a la misma rampa desde lados opuestos. Observemos como esta rampa está en subida por lo que las imágenes que nos ofrecen ambas cámaras son realmente distintas, y como en la segunda la perspectiva de la rampa en profundidad complica su estudio:



Fig. 1. 3.- Vista del escenario 3, el que muestra la cámara PRT24B-Camera1.



Fig. 1. 4.- Vista del escenario 2, el que muestra la cámara PRT24B-Camera2.

Así pues, el escenario en el que trabajaremos consta de 3 vistas muy parecidas de una rampa, las 3 con una perspectiva claramente definida, y una cuarta donde la perspectiva en profundidad y un pivote al inicio de la rampa dificultan su estudio. Nombraremos las vistas como escenarios del 1 al 4, tal y como se indica en los pies de figura, para simplificar su nombre de cara a su estudio. Para más información sobre los escenarios ver Anexo 3.

Inicialmente trabajaremos con el escenario 1 debido a su simplicidad para posteriormente ampliar el estudio al escenario 2, el que presenta un mayor número de secuencias con tumultos, y con el que nos centraremos en determinar la calidad del algoritmo desarrollado. Finalmente estudiaremos el comportamiento del algoritmo en los escenarios 3 y 4, en los que no disponemos de ninguna secuencia con tumulto, tal y como se puede ver en la tabla 4.1 para, principalmente, comprobar la cantidad de falsas alarmas que este presenta y para determinar que mejoras son necesarias aplicar al seguimiento para obtener resultados más precisos.

El presente trabajo se ha realizado de manera que sea fácilmente adaptable a cualquier tipo de escenarios. Por lo que se ha intentado que las diferentes partes que conforman la implementación del mismo sean lo más modulares posibles, lo que nos permite realizar pequeños cambios al programa sin alterar su comportamiento. De este modo, adaptarlo a un nuevo escenario que presente características similares a los cuatro anteriores no supone un problema y aumenta la **portabilidad** de este trabajo.

## 1.2.- Objetivos:

El principal objetivo de este proyecto es **la detección de tumultos mediante el uso de los filtros de partículas**. Para poder llegar a este objetivo final de una manera adecuada se deben alcanzar unos objetivos previos. La calidad con la que éstos sean alcanzados mejorará y facilitará la posterior detección de tumultos. Veamos a continuación cuáles son estos objetivos previos:

1. Detectar objetos de primer plano: en primer lugar deberemos distinguir lo que nos interesa seguir, es decir, las personas que se mueven a lo largo de la rampa y que pueden encontrarse con una situación de tumulto. Para ello extraeremos los objetos de primer plano mediante una versión modificada del método *Running Gaussian Average*, que se explicará en el Capítulo 3. Cuanto mejor sea esta extracción de los objetos de primer plano, mejor será el seguimiento, ya que nos permitirá eliminar la mayor cantidad posible de ruido.
2. Seguimiento: una vez hemos discernido qué nos interesa seguir, pasaremos a realizar su seguimiento. Para ello emplearemos los filtros de partículas, que adaptaremos adecuadamente a las condiciones del escenario donde se apliquen para obtener el seguimiento de flujo de personas que nos permita decidir si se trata o no de un tumulto.
3. Etiquetado manual de los casos con tumultos: deberemos definir con exactitud qué consideraremos “*tumulto*”, y etiquetar las secuencias de que disponemos para extraer el *Ground Truth* que posteriormente emplearemos para determinar la calidad de la detección implementada.
4. Detección de los tumultos: a partir del seguimiento obtenido y de las secuencias deberemos definir con exactitud qué criterios emplearemos para detectarlos automáticamente y cómo podremos mejorarllos para optimizar la calidad de esta detección.



Fig. 1. 5.- Ejemplo de un tumulto en el escenario 2.

5. Evaluar la calidad del algoritmo de detección: una vez realizada la detección, pasaremos a valorar cuantitativamente la calidad algoritmo de detección mediante la definición de unas métricas. A partir de su evaluación, la tasa de detección y de falsas alarmas, determinaremos la calidad del algoritmo implementado. Para la tasa de falsas alarmas, extenderemos la evaluación a un conjunto de 100 secuencias distintas de los cuatro escenarios de que disponemos para obtener unos resultados más generales.

Las particularidades mencionadas hacen que el algoritmo a desarrollar constituya un enfoque novedoso a la detección de un problema que se podría llevar a cabo de muchas otras maneras, como por ejemplo, mediante el uso de sensores, tanto de posición como de peso, otro tipo de cámaras, ya sean móviles o cenitales, y otras posibles variantes. Este trabajo emplea una determinada herramienta dentro del procesado de imagen, el filtro de partículas, que proviene de una idea relativamente antigua pero que, debido a las limitaciones tecnológicas existentes hasta el momento, no se han podido usar por la enorme cantidad de datos que necesitan procesar. En la actualidad, gracias a los continuos avances tecnológicos, concretamente al desarrollo de los procesadores, ya es posible computar estas grandes cantidades de información en tiempo real, lo que nos permite poder trabajar con los filtros de partículas.

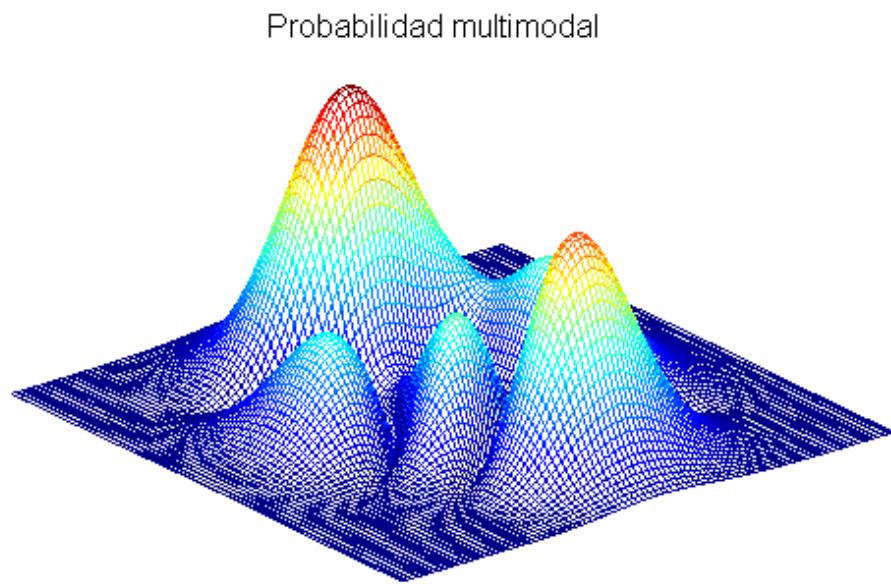
En resumen, la idea final de este trabajo es realizar un estudio preliminar de lo que podría llegar a ser **una aplicación capaz de detectar tumultos en tiempo real**, siguiendo los cinco pasos mencionados y sin tener la necesidad de realizar un procesado a posteriori que lo impediría, y por lo tanto, que pueda aplicarse en situaciones de video vigilancia y control en un futuro.

### 1.3.- Estado del arte:

El perfeccionamiento de aplicaciones que sean capaces de identificar y clasificar interacciones entre personas a partir de la información aportada por las señales obtenidas de distintos sensores multimodales pertenece a las ramas de investigación de varias disciplinas científicas, entre las que se encuentran tanto el procesado de audio como el de imagen. En este trabajo, nos centraremos en el procesado de imagen, y dentro de sus varias líneas de investigación de las que destacaríamos la detección de caras, el procesado morfológico de imágenes, la detección remota y el seguimiento de múltiples personas, nos centraremos en esta última, la cual constituirá el tema central de este trabajo.

Si el seguimiento de una persona puede llegar a ser complejo debido sus interacciones con los objetos del escenario, a los cambios de iluminación y a su comportamiento impredecible, el seguimiento multipersona incrementa su dificultad debido a los movimientos e interacciones entre individuos. La principal complejidad del seguimiento de múltiples objetivos, además de las mencionadas anteriormente, radica en el entorno de trabajo, grandes espacios con multitud de personas. Es por ello que en el presente trabajo no nos centraremos en realizar un seguimiento individual perfecto sino en obtener un seguimiento global con el que podamos **extraer el movimiento global o flujo de personas en una cinta transportadora o rampa**, lo que nos permita determinar la existencia de un tumulto.

Desde hace unos años los principales enfoques que se han empleado para realizar el seguimiento visual son dos, los que se basan en estrategias clásicas como el filtro de Kalman [1], en componentes conexas [24] o en el algoritmo *Mean Shift* o desplazamiento de la media [21, 25, 26] y los que lo hacen en algoritmos de obtención de hipótesis acerca de la ubicación del objetivo mediante la extracción de características, como pueden ser los contornos de una persona, la reconstrucción 3D o el histograma de color. El principal problema que presenta el enfoque basado en las estrategias clásicas, es que no constituye un enfoque multimodal, es decir, que la dinámica de movimiento de las personas no puede modelarse con una distribución estadística de un solo máximo (monomodal). Aún así, es posible aproximar distribuciones multimodales a partir de diversas distribuciones unimodales, pero resulta ser una aproximación poco robusta al problema de seguimiento debido a las limitaciones que éstas presentan. En cuanto al segundo tipo de enfoques, los que pretenden determinar la posición del objetivo mediante el análisis de los espacios de características, sobre todo los de color, no nos proporcionan información suficientemente robusta para la identificación de objetos, debido a los problemas que nos plantea el escenario, como la reducida calidad de las imágenes subministradas por las cámaras, los cambios de iluminación, las occlusiones debidas al escenario o incluso el hecho de que dos personas presenten un colorido similar en sus vestimentas lo que puede llegar a provocar que a efectos de color no podamos distinguirlos.



**Fig. 1. 6.- Distribución de probabilidad multimodal.**

A raíz de las limitaciones que presentan ambos enfoques, se viene desarrollando desde hace unos años una tercera estrategia para realizar el seguimiento que pretende juntar las ventajas de ambos eliminando el mayor número posible de sus limitaciones. Esta nueva estrategia, basada en los métodos de Monte Carlo, recibe el nombre de **filtros de partículas**. Entre sus múltiples aplicaciones se encuentran algoritmos que nos permiten realizar seguimientos 2D y 3D. La principal virtud de esta tecnología es que permite tratar procesos cuya dinámica puede ser modelada mediante una función de densidad de probabilidad multimodal, lo que supone una gran mejora respecto a las soluciones clásicas.

Si analizamos el entorno en el que realizaremos el seguimiento 2D, la visión de dos rampas de un aeropuerto a partir de una única rampa, podemos observar una serie de ventajas e inconvenientes que este escenario presenta:

- Disponemos de una única visión del escenario, lo que nos elimina la posibilidad de suprimir o corregir posibles occlusiones que nos ofrecería el poder trabajar con más de una visión, y además estamos expuestos a posibles vibraciones de la cámara que puede afectar negativamente a los resultados.
- El elemento de la perspectiva marca claramente el seguimiento, lo que provoca que se deba tener en cuenta la configuración espacial en los algoritmos implementados, tal y como se explica en el apartado 3.2, y que centremos el seguimiento en la rampa descendente ya que la perspectiva hace que realizar el seguimiento en la otra rampa sea difícil.

Sin embargo, a pesar del problema que nos pueda causar la **perspectiva** presente en el escenario, el hecho de realizar un seguimiento 2D simplifica el trabajo llevado a cabo, al tener como unidad básica el **píxel**, ya que al tratarse de una unidad puntual resulta sencilla de procesar a partir de una imagen 2D. Aún así, si hubiéramos dispuesto de más cámaras enfocando el escenario, se hubiera podido plantear realizar un seguimiento 3D, lo que hubiese reforzado el seguimiento al disponer de mayor información, pero supondría una complicación en el procesado, debido a que nuestra unidad básica pasaría a ser el voxel y deberíamos trabajar con modelos 3D.

## 1.4.- Organización:

La organización de este trabajo pretende seguir una estructura que permita al lector entender cuáles son los pasos necesarios para poder realizar un seguimiento con filtros de partículas y de su posterior uso para solucionar o detectar un problema determinado.

El segundo capítulo introduce al lector en los filtros de partículas. Cómo se llega a ellos a partir del seguimiento bayesiano óptimo, de las limitaciones que presenta el filtro de Kalman y cómo los filtros de partículas logran solventar estas limitaciones y permiten obtener una mejor aproximación general y robusta del seguimiento de personas. Además se explica la dinámica que emplearemos para realizar el seguimiento, en que bloques dividiremos el problema global y en qué consistirán.

El tercer capítulo explica la implementación del filtro de partículas en un caso real, es decir, cómo y por qué se tiene que adaptar al entorno en el que se aplican y qué datos se emplean para realizar el seguimiento. Asimismo se explicará la base teórica en la que se basa el método usado para la extracción de estos datos a partir de la información obtenida de las cámaras.

El cuarto capítulo detalla las modificaciones aplicadas al seguimiento para mejorar los resultados obtenidos, por qué son necesarias estas modificaciones y cómo perfeccionan el seguimiento en un escenario con múltiples *targets*. También se define qué consideraremos como tumulto y la metodología que se seguirá para su detección a partir de la información que nos subministran los filtros de partículas.

En el quinto capítulo se muestran los resultados obtenidos en la detección de tumultos en un total de 100 secuencias con y sin tumultos. Definiremos los parámetros y las métricas que emplearemos para determinar de manera objetiva la calidad del seguimiento y de la posterior detección de los tumultos.

Finalmente, el sexto y último capítulo expone de qué manera se han cumplido los objetivos inicialmente propuestos, las conclusiones obtenidas en el desarrollo del trabajo y las futuras mejoras que se podrían llevar a cabo en una posible continuación de este proyecto.

## 2.- Seguimiento mediante filtros de partículas

---

La solución al problema del seguimiento consiste en determinar la posición de uno o varios objetos en cada instante de tiempo. En nuestro caso particular, en secuencias de video, los instantes de tiempo vienen determinados por los sucesivos fotogramas. Así pues, nuestro objetivo es estimar la posición de nuestro/s objeto/s fotograma a fotograma del mejor modo posible. Hay que tener presente que nuestro objetivo final no es el de realizar un seguimiento perfecto objeto a objeto, sino el de realizar un buen seguimiento global de manera que podamos determinar el flujo de movimiento creado por los distintos objetivos, éste es el que nos permitirá detectar un tumulto.

Normalmente se agrupan los algoritmos de seguimiento visual en dos tipos, los de Representación y Localización del Objeto y los de Filtrado y Asociación de Datos [22]. El primer grupo se basa, a grandes rasgos, en la obtención de características que permitan rastrear el escenario en busca del objetivo que más se asemeje a los parámetros de búsqueda, con el pequeño inconveniente de tener que adaptarse a los cambios que experimente el objetivo a seguir. Por otro lado, el segundo grupo, de mayor complejidad en general, se basa en incorporar información a priori de la escena, típicamente basada en modelos de movimiento, y la observación de los estados para evaluar la verosimilitud de distintas hipótesis acerca de la posición del objetivo.

Es cada vez más frecuente, para los algoritmos de Filtrado y Asociación, incluir elementos de no linealidad y no Gaussianos para lograr un modelo más preciso de la dinámica del sistema físico. Hay que tener presente que el movimiento de una persona a lo largo del tiempo es modelable con una distribución de probabilidad multimodal, es decir, que no existe un movimiento preestablecido que se pueda determinar a partir de una cierta tendencia en velocidad media y varianza de dicha celeridad alrededor de la media, sino que existen varios patrones. Este es el principal motivo por el cual un modelo Gaussiano y lineal resulta ser una aproximación poco precisa del problema.

Dentro de estos algoritmos, uno de los métodos de mayor importancia es el filtro de partículas. En resumen, el filtro de partículas, que está basado en los métodos de Monte Carlo [23], utiliza un conjunto de puntos de masa, o partículas, para modelar densidades de probabilidad complejas y que constituye una generalización de los métodos basados en el filtro de Kalman, gracias al hecho de poderse aplicar en cualquier modelo de estados.

### 2.1.- Introducción a los filtros de partículas

En este apartado formularemos el problema de seguimiento y, partiendo de la solución óptima no lineal Bayesiana, revisaremos la solución clásica de Kalman para finalmente exponer una de las aproximaciones subóptimas, los Filtros de Partículas. Además, explicaremos ciertas características que presentan algunas de las variantes de los filtros de partículas y el modelo

final que emplearemos durante el trabajo realizado. Hay que tener presente que para desarrollar este apartado se ha tomado como referencia el artículo de Arulampalan, Maskell, Gordon y Clapp [1] cuya lectura es altamente recomendada para aquellos que quieran introducirse de un modo más detallado en seguimiento Bayesiano y los filtros de partículas. También se ha tenido en cuenta como referencia la presentación realizada por Carlos Segura sobre “*Bayesian Filtering*” [2].

### 2.1.1.- *Seguimiento Bayesiano*

Para definir el problema del seguimiento se debe considerar, en primer lugar, un modelo para la evolución del estado de un objetivo. Si consideramos la secuencia de estados  $\{x_k, k \in \mathbb{N}\}$ ; su evolución vendrá determinada por la siguiente **ecuación de estado**:

$$x = f_k(x_{k-1}, v_{k-1}) \quad (2.1)$$

donde  $f_k$  es una función no lineal del estado  $x_{k-1}$  y  $v_{k-1}$  es un proceso que modela el ruido. El objetivo del seguimiento es estimar recursivamente  $x_k$  empleando las medidas obtenidas a partir de una serie de observaciones ruidosas:

$$z_k = h_k(x_k, n_k) \quad (2.2)$$

Ésta es la **ecuación de medida**, donde  $h_k$  es una función, en general, no lineal y  $n_k$  un proceso estocástico que modela el ruido en la observación. En particular, buscamos estimaciones de  $x_k$  basadas en el conjunto de medidas disponibles  $z_{1:k} = \{z_i, i = 1, \dots, k\}$ , donde  $k$  denota el instante de tiempo.

Desde una perspectiva Bayesiana, el problema del seguimiento consiste en calcular recursivamente qué grado de verosimilitud tiene un estado  $x_k$  en ese mismo instante  $k$ , a partir de los datos disponibles en las observaciones realizadas hasta este momento,  $z_{1:k}$ . Es decir, es necesario estimar la densidad de probabilidad  $p(x_k|z_{1:k})$ . Se asume que la *pdf* inicial  $p(x_0|z_0) \equiv p(x_0)$  del vector de estados, también llamada densidad a priori (*prior*), es conocida (siendo  $z_0$  un conjunto de medidas u observaciones vacío). Así pues, en principio, obtendremos la densidad de probabilidad  $p(x_k|z_{1:k})$  mediante la realización de estimaciones recursivas a partir de las probabilidades anteriores al estado actual en dos pasos: **predicción y actualización**.

Suponiendo que disponemos de la *pdf* requerida en el estado  $k - 1$ ,  $p(x_{k-1}|z_{k-1})$ , el siguiente paso de predicción consiste en hallar la densidad a priori mediante la ecuación de Chapman-Kolmogorov:

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1}) p(x_{k-1}|z_{1:k-1}) dx_{k-1} \quad (2.3)$$

donde se asume que la secuencia de estado describe un proceso de Markov de orden 1, dado que consideramos  $p(x_k | x_{k-1}, z_{1:k-1}) = p(x_k | x_{k-1})$ . El modelo probabilístico de la evolución de estado  $p(x_k | x_{k-1})$  está definido por la ecuación de estado y la estadística de ruido conocida,  $v_{k-1}$ . En el contexto del seguimiento, esta densidad de probabilidad acostumbra a ser un modelo de movimiento determinado.

En el instante  $k$ , disponemos de la medida  $z_k$  que nos servirá para pasar al siguiente paso, actualizar la estadística a priori (prior) mediante la regla de Bayes:

$$p(x_k | z_{1:k-1}) = \frac{p(z_k | x_k)p(x_k | z_{1:k-1})}{p(z_k | z_{1:k-1})} \quad (2.4)$$

donde tenemos la verosimilitud multiplicada por la estadística a priori y normalizada por una constante obtenida a partir de:

$$p(z_k | z_{1:k-1}) = \int p(z_k | x_k)p(x_k | z_{1:k-1})dx_k \quad (2.5)$$

Por lo tanto, el paso de actualización depende directamente de la función de verosimilitud  $p(z_k | z_{1:k-1})$  definida en la ecuación de medida (2.2) y en la estadística de  $n_k$ . Así pues, modificaremos la densidad a priori a partir de  $z_k$  para obtener la densidad a posteriori del estado actual.

Las relaciones de recurrencia presentadas en (2.3) y (2.4) son la base de la solución óptima Bayesiana al problema del seguimiento. El problema es que estas relaciones sólo constituyen una formulación conceptual, de manera que no se pueden calcular en general de manera analítica. Aún así, existen algunos casos concretos con ciertas restricciones donde sí se pueden calcular estas soluciones, este es el caso del filtro de Kalman. Pero si el cálculo de la solución analítica es realmente intratable, es necesario recurrir a soluciones subóptimas como los filtros de partículas, para obtener una aproximación efectiva de la solución óptima Bayesiana. Hay que tener presente que además de la solución subóptima de los filtros de partículas existen otras soluciones como la aproximación basada en filtros de celdas y el filtro extendido de Kalman y que éstas están explicadas detalladamente en [1], pero como no se emplean en este trabajo no se ha considerado su explicación.

### **2.1.2.- El filtro de Kalman**

El filtro de Kalman asume que la densidad a posteriori es, en cada instante, Gaussiana y que, por lo tanto queda completamente definida por su media y su covarianza. Si  $p(x_{k-1}|z_{1:k-1})$  es Gaussiana, se puede demostrar que  $p(x_k|z_{1:k})$  también lo es. Para ello es necesario asumir que  $v_{k-1}$  y  $n_k$ , de las ecuaciones de estado y medida, presentan distribuciones Gaussianas de parámetros conocidos, por lo que, tanto  $f_k(x_{k-1}, v_{k-1})$  como  $h_k(x_k, n_k)$  son funciones lineales conocidas de sus respectivas variables ( $x_{k-1}$ ,  $v_{k-1}$ ,  $x_k$  y  $n_k$ ). De este modo,

podemos reformular las ecuaciones de estado y medida, (2.1) y (2.2) respectivamente, de la siguiente manera:

$$x_k = F_k x_{k-1} + v_{k-1} \quad (2.6)$$

$$z_k = H_k x_k + n_k \quad (2.7)$$

$F_k$  y  $H_k$  son las matrices conocidas que definen las funciones lineales. Llamaremos  $Q_{k-1}$  y  $R_k$  a la covarianza de los procesos  $v_{k-1}$  y  $n_k$ , respectivamente. Al mismo tiempo, consideraremos el que ambos procesos presentan media cero y son estadísticamente independientes. Cabe destacar que tanto las matrices de medida,  $F_k$  y  $H_k$ , como las de los parámetros de ruido,  $Q_{k-1}$  y  $R_k$ , pueden ser variantes en el tiempo. Hay que tener presente que el filtro de Kalman puede enfocarse desde un punto de vista Bayesiano a partir de las ecuaciones (2.3) y (2.4), tal y como se hará a continuación, pero que también se puede abordar mediante la minimización del error cuadrático. El filtro de Kalman puede ser enunciado como la siguiente relación recursiva:

$$p(x_{k-1}|z_{1:k-1}) = N(x_{k-1}; m_{k-1|k-1}; P_{k-1|k-1}) \quad (2.8)$$

$$p(x_k|z_{1:k-1}) = N(x_k; m_{k|k-1}; P_{k|k-1}) \quad (2.9)$$

$$p(x_k|z_{1:k}) = N(x_k; m_{k|k}; P_{k|k}) \quad (2.10)$$

Donde  $N(x; m, P)$  es la densidad Gaussiana de la variable  $x$ , media  $m$  y covarianza  $P$ . Si aplicamos las ecuaciones (2.3) y (2.4) obtendremos las ecuaciones recurrentes que conformarán el algoritmo:

$$m_{k|k-1} = F_k m_{k-1|k-1} \quad (2.11)$$

$$P_{k|k-1} = Q_{k-1} + F_k P_{k-1|k-1} F_k^H \quad (2.12)$$

$$m_{k|k} = m_{k|k-1} + K_k (z_k - H_k m_{k|k-1}) \quad (2.13)$$

$$P_{k|k} = P_{k|k-1} - K_k H_k P_{k|k-1} \quad (2.14)$$

Donde  $K_k$  es la ganancia de Kalman y tiene la siguiente expresión:

$$K_k = P_{k|k} H_k^H (H_k P_{k|k} H_k^H + R_k)^{-1} \quad (2.15)$$

El superíndice  $H$  denota el transpuesto del conjugado de la matriz, hermítica. Este algoritmo iterativo constituye la solución óptima al problema del seguimiento, siempre y cuando se cumplan las suposiciones altamente restrictivas que se han planteado inicialmente. El filtro de Kalman es el mejor algoritmo que existe siempre y cuando el entorno de trabajo sea lineal y

Gaussiano. En nuestro caso, el seguimiento multipersona, no tratamos con un problema que se pueda modelar mediante distribuciones de probabilidad lineales y Gaussianas, por lo que esta aproximación clásica no resulta efectiva y hay que recurrir a otras soluciones como el filtro de partículas.

### 2.1.3.- El filtro de partículas

Los filtros de partículas [1, 7, 8, 14, 15] pertenecen al conjunto de filtros basados en los métodos de Monte Carlo. La idea principal que persiguen es la de aproximar la densidad de probabilidad a posteriori a partir de un conjunto discreto de muestras o partículas, cada una con un peso asociado, lo que permite convertir integrales intratables en sumatorios calculables de manera bastante más sencilla. A medida que aumentamos el número de partículas empleado, mejor es la aproximación llevada a cabo y, por lo tanto, se asemeja más a la función continua descrita por la función de probabilidad a posteriori.

Formalmente, si partimos de un conjunto de medidas aleatorias,  $\{x_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ , que caracterizan la densidad a posteriori,  $p(x_{0:k}|z_{1:k})$ , donde  $\{x_{0:k}^i; i = 0, \dots, N_s\}$  es un conjunto de partículas, cada una con un peso asociado  $\{w_k^i; i = 0, \dots, N_s\}$  y  $\{x_{0:k}; x_k, j = 0, \dots, k\}$  es el conjunto de estados en cada instante  $k$ . Los pesos están normalizados de manera que  $\sum_i w_k^i = 1$ . De este modo, podemos aproximar la densidad a posteriori de la siguiente manera:

$$p(x_{0:k}|z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (2.16)$$

Así pues, hemos aproximado la densidad a posteriori de manera discreta mediante el sumatorio de distintos valores ponderados por unos pesos. Estos pesos  $w_k^i$ , se calculan a través del principio de “*Importance Sampling*” [1], el cual nos permite obtener muestras cuando la *pdf* que se quiere aproximar es muy compleja. Supongamos que  $p(x)$  es una densidad de estas características, pero que existe una  $r(x)$  evaluable tal que  $p(x) \propto r(x)$ . También disponemos de una serie de muestras  $x^i \sim q(x)$ ,  $i = 0, \dots, N_s$  calculadas a partir de la evaluación de los pesos de cada partícula en la llamada **densidad de importancia**  $q(\cdot)$ . De modo formal, podemos obtener una aproximación de la densidad  $p(\cdot)$  como:

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i) \quad (2.17)$$

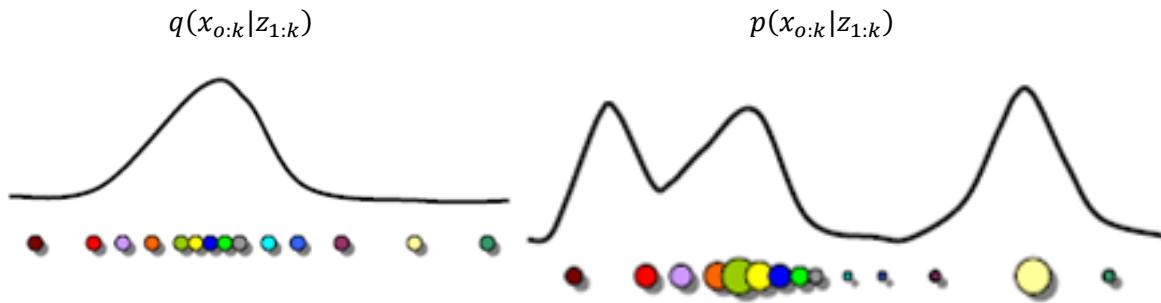
Calculando los pesos a partir de la siguiente expresión:

$$w^i \propto \frac{r(x^i)}{q(x^i)} \quad (2.18)$$

Por lo tanto, si obtenemos el conjunto de las  $N_s$  muestras a partir de la densidad de importancia  $q(\cdot)$ , en el estado  $x_{0:k}$ , entonces es posible calcular sus pesos a partir de la aproximación obtenida de  $p(x_{0:k}|z_{1:k})$  del siguiente modo:

$$w^i \propto \frac{p(x_{0:k}^i | z_{1:k})}{p(x_{0:k}^i | z_{1:k})} \quad (2.19)$$

Para entender mejor el funcionamiento del principio de “*Importance Sampling*”, observaremos las siguientes gráficas, que corresponde a las funciones  $q(\cdot)$  y  $p(\cdot)$  en un instante  $k$ :



**Fig. 2.- 7.- Ejemplo del funcionamiento del principio *Importance Sampling*.**

Tal y como hemos dicho anteriormente, como no podemos obtener muestras de  $p(\cdot)$ , primero obtendremos  $q(\cdot)$ . A partir de  $q(\cdot)$  dibujaremos las partículas según la densidad de importancia para luego evaluar sus pesos a través de la densidad a posteriori  $p(\cdot)$  y así, finalmente, obtener sus pesos definitivos. El tamaño de las partículas es directamente proporcional a su peso, es decir, cuanto mayor sea su tamaño mayor será su peso. Además cabe observar como los valores altos de los pesos coinciden con los de la *pdf*, y que los valores bajos coinciden con las colas de ésta, lo que puede llegar a provocar que desaparezcan las partículas con estos pesos.

Existe un gran número de variaciones de los filtros de partículas, realizadas para que mejoren su rendimiento en función de la aplicación en la que se usen.

Empezaremos hablando del modelo SIS, el más sencillo, para pasar a hablar del modo SIR, el que más se parece al que emplearemos durante el trabajo.

#### · *SIS PF:*

El filtro SIS (*Sequential Importance Sampling*), constituye el caso más sencillo y general de la implementación de los filtros de partículas. Su funcionamiento parte del principio de *Importance Resampling* en cuanto al modo de obtener los pesos, pero sin obviar el hecho de que estamos ante un algoritmo iterativo. Si obtenemos las muestras a partir de la propagación de las existentes en el estado anterior, nuestra densidad de importancia tendrá el siguiente valor:

$$q(x_{0:k} | z_{1:k}) = q(x_k | x_{0:k-1}, z_{1:k})q(x_{0:k-1}) \quad (2.20)$$

Ahora podemos obtener las muestras  $x_{0:k}^i \sim q(x_{0:k} | z_{1:k})$  mediante el incremento de las muestras de que disponemos  $x_{0:k-1}^i \sim q(x_{0:k-1} | z_{1:k-1})$  a partir del nuevo estado  $x_k^i \sim q(x_k | x_{0:k}, z_{1:k})$ . De este modo podemos expresar  $p(x_{0:k}, z_{1:k})$  en términos de  $p(x_{0:k-1}, z_{1:k-1}), p(z_k, x_k)$  y  $p(x_k, x_{k-1})$  de la siguiente manera:

$$p(x_{0:k}, z_{1:k}) \propto p(z_k | x_k)p(x_k | x_{k-1})p(x_{0:k-1} | z_{1:k-1}) \quad (2.21)$$

Aplicando esta *pdf*, la anterior densidad de importancia a la ecuación 2.19 y asumiendo que estamos ante un sistema de estados Markoviano de orden 1 podremos reescribir la ecuación de actualización de los pesos del filtro de partículas:

$$w_k^i \propto \frac{p(z_k | x_k^i)p(x_k^i | x_{k-1}^i)p(x_{0:k-1}^i | z_{1:k-1})}{q(x_k^i | x_{0:k-1}^i, z_{1:k})q(x_{0:k-1}^i | z_{1:k-1})} = w_{k-1}^i \frac{p(z_k | x_k^i)p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{0:k-1}^i, z_{1:k})} \quad (2.22)$$

Si finalmente consideramos que  $q(x_k^i | x_{0:k-1}^i, z_{1:k}) = q(x_k^i | x_{k-1}^i, z_{1:k})$  de manera que la densidad de importancia sólo depende de  $x_{k-1}$  y de  $z_k$ , podemos reescribir la ecuación anterior para obtener la definitiva **ecuación de los pesos del filtro de partículas**:

$$w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i)p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)} \quad (2.23)$$

Hay que tener presente que la elección de  $q(x_k^i | x_{k-1}^i, z_k)$  suele ser uno de los problemas críticos de diseño de los filtros de partículas ya que influye tanto en la distribución de las partículas como en la evaluación del valor de sus pesos. Asimismo, podemos reescribir la ecuación de la densidad  $p(\cdot)$  descrita en (2.17) como:

$$p(x_k, z_{1:k}) \approx \sum_{i=1}^{Ns} w_k^i \delta(x_k - x_k^i) \quad (2.24)$$

Los filtros de partículas que emplean esta técnica para calcular su peso sufren un problema importante, la degeneración de las partículas que lo componen, debido al incremento estocástico de la varianza de los pesos. Al realizar varias iteraciones tiende a sobrevivir una única partícula, debido a que los pesos del resto tienden a ser negligibles, casi 0. En [23] podemos encontrar una medida fiable de degeneración de las partículas. Debido a que su cálculo no puede ser realizado de manera exacta, se utiliza la siguiente aproximación:

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{Ns} (w_k^i)^2} \quad (2.25)$$

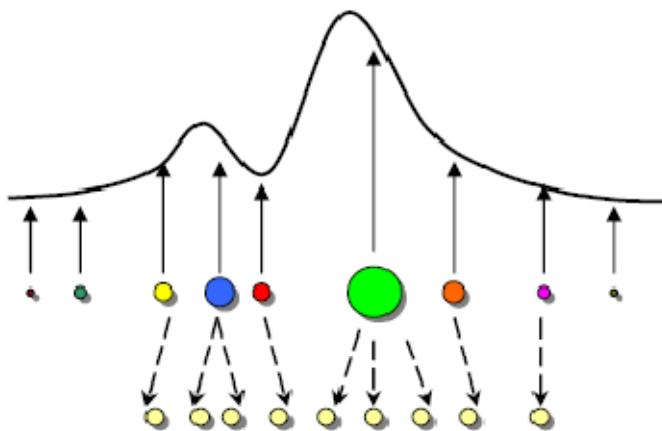
donde  $w_k^i$  son pesos normalizados. Esta medida se conoce como **tamaño efectivo de muestreo**, y va de 1 a  $N_s$ , siendo la unidad el valor que indica que sólo una partícula tiene peso. En [14] se define la **tasa de supervivencia** [14, 15] de las partículas como:

$$\alpha = \frac{N_{eff}}{N_s} \quad (2.26)$$

donde  $0 < \alpha < 1$ . Su valor indica el tanto por uno de partículas que sobreviven tras la asignación de pesos. Si este valor es muy pequeño será indicador de que el filtro se está degenerando y que, por lo tanto, no está realizando correctamente la estimación. Este parámetro se suele utilizar, con la ayuda de un umbral adecuado, para detectar la degeneración del filtro. Para combatir este problema, una de las técnicas más empleadas es el Remuestreo. Básicamente, consiste en reubicar todas las partículas y sus respectivos pesos en un nuevo conjunto donde todos los pesos sean iguales:

$$\{x_k^i, w_k^i\} \rightarrow \left\{x_k^i, \frac{1}{N_s}\right\} \quad (2.27)$$

De este modo se pretende hacer sobrevivir aquellas partículas o muestras más verosímiles, es decir, las de mayor peso, regenerándolas en  $n$  nuevas partículas por cada muestra antigua, asignándoles el mismo peso a todas ellas,  $\frac{1}{N_s}$ , el peso normalizado respecto al número de partículas empleadas. Observemos gráficamente su funcionamiento a partir de un conjunto de 9 partículas:



**Fig. 2. 8.- Funcionamiento del remuestreo, se puede ver como las partículas con un peso pequeño "mueren" mientras que las de mayor peso se remuestrean con mayor cardinalidad que las de menor.**

El uso del remuestreo soluciona en gran medida la degeneración de partículas, aún así sigue existiendo un efecto de concentración alrededor de aquellas partículas que poseen mayor peso. Lo que provoca que las partículas tiendan a agruparse en puntos muy concretos del

objetivo reduciendo así la efectividad de la estimación. Una solución a este problema es aplicar otro algoritmo de filtro de partículas, el SIR, que se verá a continuación.

#### · *SIR PF:*

El filtro SIR (*Sampling Importance Resampling*) es un caso particular del filtro SIS, en el que se escogen adecuadamente tanto la densidad de importancia como los instantes de remuestreo. Así pues, la densidad de importancia  $q(x_k|x_{k-1}, z_k)$  pasa a ser la densidad de transición  $p(x_k|x_{k-1})$  de manera que si sustituimos en (2.23) la expresión de los pesos del filtro de partículas queda del siguiente modo:

$$w_k^i \propto w_{k-1}^i p(z_k|x_k^i) \quad (2.28)$$

Además el remuestreo se realiza en cada iteración, de manera que se evita la concentración de partículas en un punto determinado a lo largo del tiempo. De este modo, gracias a esta modificación, la expresión de los pesos se simplifica de la siguiente manera:

$$w_k^i \propto p(z_k|x_k^i) \quad (2.29)$$

Debido a que tras realizar el remuestreo todos los pesos valen  $\frac{1}{N_s}$ . El único problema que se plantea el uso de este algoritmo es que al ser la densidad de importancia independiente de la observación provoca que los estados se analicen a ciegas, es decir, sin tener conocimiento previo de los mismos. Esto se traduce en que la eficiencia de este algoritmo va directamente relacionada con el número de partículas que se emplean.

#### · *Funcionamiento del Filtro de Partículas:*

Una vez analizado el algoritmo del filtro de partículas, queda por ver su funcionamiento al aplicarlo al seguimiento en general. Inicialmente explicaremos su funcionamiento para un caso sencillo, el seguimiento de una sola persona, para posteriormente ver qué modificaciones hay que realizar para poder llevar a cabo el seguimiento de múltiples objetivos.

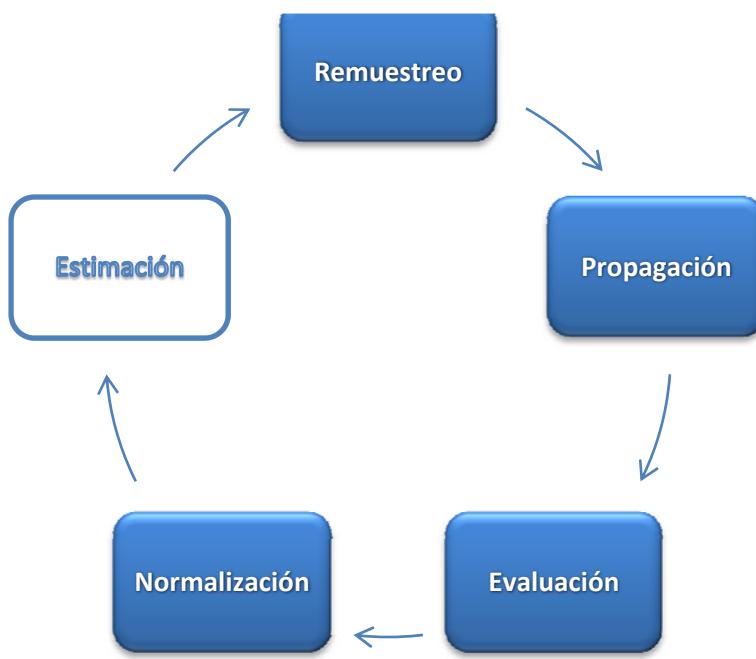
En primer lugar deberemos determinar qué características del sistema nos interesa estimar, esto es lo que constituirá el modelo del objetivo. Cada una de las partículas del filtro deberá representar estas características. En el caso del seguimiento, lo más sencillo es que una partícula represente una coordenada del espacio 2D ( $x, y$ ). De este modo, estamos formulando un filtro de partículas con un único objetivo y una única magnitud a estimar, su posición. En nuestro caso particular, como que la finalidad del seguimiento es poder estimar el flujo de la gente, una característica que también nos puede interesar estimar es la velocidad, es decir, en este caso nuestras partículas tendrían cuatro dimensiones ( $x, y, v_x$  y  $v_y$ ). Hay que tener presente que cada característica que añadamos a la partícula aumenta considerablemente el número de cálculos que realiza el algoritmo, por lo que intentar minimizar este número sería

lo ideal. Veamos a continuación cuál es la estructura iterativa que muestra el funcionamiento del filtro de partículas:

Se generan las muestras o partículas a partir de una densidad de importancia  $q(\cdot)$ , que en nuestro caso es un modelo SIR PF, que reubica las partículas mediante un modelo de movimiento al ser  $q(x_k | x_{k-1}, z_k) = p(x_k | x_{k-1})$ .

1. Se evalúan los pesos  $w_k^i$  según (2.23) con una *pdf* que modele la verosimilitud de que una región del espacio pertenezca a un objetivo. Tras calcular los pesos ya disponemos de una aproximación de la densidad a posteriori. La esperanza de esta aproximación, ecuación (2.30), es la hipótesis de la posición de la persona o de las características que queramos estimar.
2. Se realiza el remuestreo de las partículas de manera que aseguremos que mantenemos el tamaño efectivo de remuestreo (2.25) y así minimizamos el problema de la degeneración de las partículas.

La estructura anterior se puede estructurar en el siguiente diagrama de bloques donde se resume el funcionamiento del filtro de partículas:



**Fig. 2. 9.- Diagrama de bloques del filtro de partículas.**

La secuencia iterativa empieza en la **propagación**, donde se generan las partículas tal y como se ha explicado en el punto 1. A continuación se pasa a la **evaluación** de los pesos, asignando el valor correspondiente a cada partícula en función de su verosimilitud con el objetivo que se desea seguir. Seguidamente, se realiza la **normalización** de los pesos para llevar a cabo la **estimación** de las características que deseamos estimar. La estimación se realiza a partir de la ecuación (2.30) que está basada en la aproximación de la *pdf* presentada en (2.24), de este

modo se calcula la posición del filtro mediante la suma de las posiciones de las partículas ponderada por sus respectivos pesos. Finalmente, se realiza el **remuestreo** de las partículas, con la finalidad explicada en el punto 3, de manera que se vuelva a iniciar la secuencia.

$$X_k^i = E[x_k | z_{1:k}] \approx \frac{1}{N_s} \sum_{i=1}^{N_s} w_k^i x_k^i \quad (2.30)$$

Hay que tener presente que para su implementación real, el filtro deberá inicializarse, realizará el seguimiento y, finalmente será destruido, tal y como se explica en el siguiente apartado, lo explicado anteriormente es su funcionamiento teórico. Asimismo, en el capítulo 3 se explican los criterios que nos permiten evaluar la verosimilitud de una partícula con el objetivo que se desea seguir y las modificaciones que debemos aplicar al funcionamiento para que el seguimiento se realice como deseamos adaptándose perfectamente a las condiciones del entorno.

## 2.2.- Dinámica del seguimiento:

Para poder realizar un correcto seguimiento con los filtros de partículas, es necesario estructurar claramente cuál será su dinámica, es decir, cómo este se llevará a cabo. Esta dinámica constará de las distintas etapas que son necesarias debido al funcionamiento de los filtros de partículas para poder realizar un adecuado seguimiento. Hay que tener presente que el principal objetivo de nuestro seguimiento **no es el de seguir gente individualmente**, sino el de seguir el mayor número de gente posible, que se mueve en una determinada zona del escenario de trabajo, para poder así **extraer el flujo de movimiento** que nos permita obtener una idea de cómo la gente se mueve. De este modo podremos decidir cuando este flujo se bloquee o pare de manera imprevista, lo que podría deberse a una situación que clasificaremos como tumulto.

El funcionamiento básico de los sistemas de seguimiento utilizado en la actualidad sigue el siguiente patrón, que si lo adaptamos al comportamiento propio de los filtros de partículas tendrá 3 estados claramente definidos, los cuales los filtros deberán seguir durante su *vida*:



Fig. 2. 10.- Diagrama de bloques de la dinámica de seguimiento que se implementará.

1. **Inicialización:** En primer lugar deberemos decidir cuándo y dónde generaremos los filtros de partículas. Además inicializaremos su tamaño, posición e histograma.
2. **Seguimiento:** Definiremos las características que nos interesa que sigan los filtros de partículas para lograr que sólo sigan a la gente que se mueve en la rampa. También determinaremos las restricciones que debe tener el seguimiento, ya que primordialmente lo que queremos es que este se realice en una determinada zona del escenario sin que nos interfieran los movimientos que externos a esta zona de manera que el seguimiento sea lo más fluido posible. La estructura interna del seguimiento lo conforma el diagrama de bloques de la figura 2.3.
3. **Destrucción:** Finalmente los filtros de partículas deberán desaparecer, o destruirse, cuando lleguen a una zona que esté fuera de la zona crítica de estudio donde se pueda producir un tumulto.

La explicación de las etapas de inicialización y destrucción de los filtros se encuentra detallada en el apartado 3.2, mientras que la explicación de las características que seguirán los filtros se

encuentra en el apartado 3.1 y las restricciones o limitaciones que deberá tener el seguimiento en el apartado 4.1.

Hay que tener presente que estas 3 etapas están íntimamente ligadas al escenario de trabajo. En nuestro caso particular, como lo que nos interesa seguir es el flujo de personas en una rampa de un aeropuerto, situaremos la zona de inicialización de los filtros al principio de ésta, el seguimiento se realizará en la propia rampa y este finalizará una vez el filtro salga de la rampa, momento en el cual deberá ser destruido. Veamos una imagen donde se ilustran claramente estas 3 etapas:

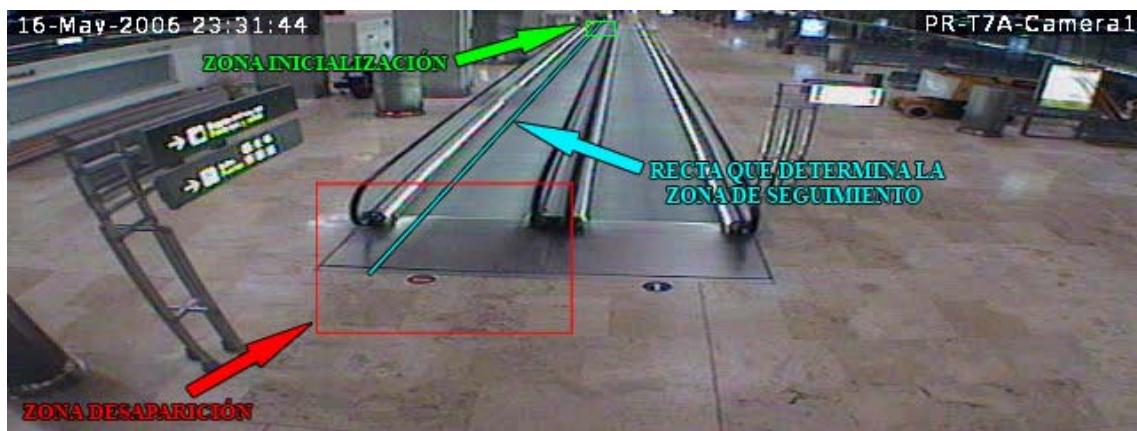


Fig. 2. 11.- Situación de las zonas de inicialización y desaparición en el escenario 1.

Finalmente, una vez obtenido un correcto funcionamiento del seguimiento, se pasará a determinar qué condiciones son las que nos dirán si nos encontramos o no en una situación clasificada como un tumulto (apartado 4.2).

### 3.- Implementación del filtro de partículas

Una vez explicada la base teórica de los filtros de partículas, qué son, en qué consisten y cómo funcionan, queda por ver cómo se implementan en un caso real 2D con perspectiva como el que nos ocupa [6, 24]. Hay que tener presente que los filtros de partículas son la herramienta que nos permite realizar el seguimiento o *tracking* dentro de nuestro sistema de detección de tumultos. Inicialmente, para familiarizarnos con su funcionamiento, hemos realizado su implementación para el seguimiento de una recta (ver Anexo 1: Toy Example). En el caso particular de este trabajo, la implementación de un sistema de detección es algo más complicado y conlleva una serie de etapas:

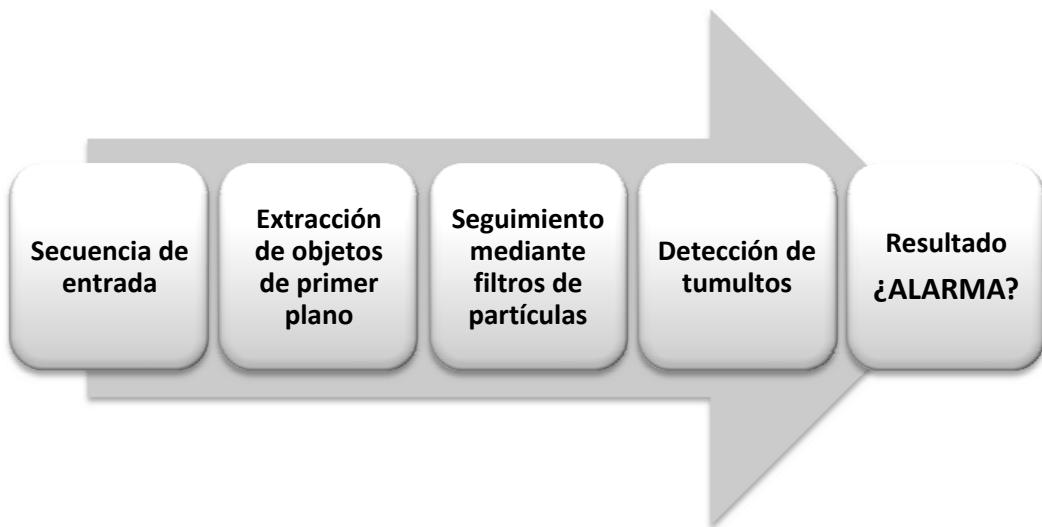


Fig. 3. 12.-Diagrama de flujo del sistema desarrollado en este proyecto.

De este modo partimos de unas secuencias de entrada, tomadas a partir de una cámara fija, con las que trabajaremos fotograma a fotograma para realizar la extracción de objetos de primer plano o *foreground*. Esto condiciona totalmente la realización del proyecto, puesto que, al tratarse de grabaciones mediante cámara fija, se puede partir de un aprendizaje del fondo de escena o *background* como punto de partida para conseguir reconocer los objetos no pertenecientes al mismo y, por lo tanto, susceptibles de detectarse como primer plano. Si la cámara estuviera en movimiento o tuviéramos más de una, la solución al problema se debería enfocar desde otro punto de vista.

A continuación, realizaremos el seguimiento utilizando los filtros de partículas que intentarán estimar la posición de los objetos de *foreground* extraídos. El funcionamiento interno de este seguimiento es el visto en el diagrama de bloques de la figura 2.4. Finalmente, realizaremos la detección de tumultos mediante el análisis del seguimiento a partir de distintos criterios que nos permitan detectar cuándo tiene lugar un tumulto, originando algún tipo de *alarma* que permita advertir de esta situación.

Los principales factores que se deben tener en cuenta en la implementación de los filtros de partículas son dos: **la evaluación de la verosimilitud y el modelo de propagación**. Inicialmente, en la sección 3.1, veremos cómo se realiza la evaluación de la verosimilitud, es decir, cuáles son las características que emplea el filtro de partículas para realizar el *tracking* y el método empleado para extraerlas de las secuencias de entrada. Seguidamente en la sección 3.2, analizaremos su modelo de propagación, el cual abarca desde el funcionamiento de su dinámica de seguimiento hasta su adaptación al entorno con perspectiva.

### 3.1.- Extracción de características para la realización del seguimiento:

La evaluación de la verosimilitud, es decir, la asignación de los pesos de las partículas del filtro se realiza mediante la ecuación (3.1). El hecho de haber elegido que el peso sea una función exponencial se debe a que, de este modo se logra discriminar mejor las partículas, es decir, provoca que se acentúe el efecto de un peso pequeño. Si el peso de una partícula es muy elevado, con este criterio, todavía lo será más.

$$w_k^i = e^{-\beta \cdot SCORE} \quad (3.1)$$

donde  $\beta$  es el parámetro que determina si nos interesa que los pesos de las partículas afecten a la estimación, de la posición real del filtro, de un modo homogéneo o que únicamente lo hagan las de mayor peso. En el trabajo emplearemos una  $\beta = 3$  tal y como se explica en el Anexo 1, donde se detallan las simulaciones realizadas para determinar el valor de  $\beta$  escogido.

La información que nos interesa que sigan los filtros de partículas viene dada por dos tipos de datos. Por un lado, los datos que nos aportan los píxeles de primer plano que contiene el filtro de partículas obtenidos tras la extracción de *foreground* (ver apartado 3.1.1). Por otro, la distancia de *Battacharyya*, una métrica que permite evaluar de forma cuantitativa la verosimilitud entre dos histogramas de color (para más detalle ver apartado 3.1.2), que obtenemos entre el histograma del color del filtro de partículas y el objeto a seguir. Esta información genera lo que llamamos SCORE y contribuye en el cálculo de los pesos de las partículas. Veamos cómo se calcula el SCORE:

$$SCORE = \lambda_c \cdot Batt + \lambda_f \cdot (1 - FG) \quad \text{donde } \lambda_c + \lambda_f = 1 \quad (3.2)$$

El SCORE va de 0 a 1, siendo 0 la verosimilitud máxima y 1 la mínima. El parámetro *Batt* indica el valor de la distancia de *Battacharyya*, éste va de 0 a 1, siendo 0 el parecido máximo y 1 el mínimo. En cambio el valor de *FG* corresponde a la cantidad de píxeles de *foreground* que contiene la partícula respecto a los que podría llegar a contener, en tanto por 1, tal y como muestra la siguiente ecuación:

$$FG = \frac{\text{Nº de píxeles de foreground de la partícula}}{\text{Nº de píxeles que puede llegar a contener la partícula}} \quad (3.3)$$

Los valores que tomen  $\lambda_c$  y  $\lambda_f$  nos permite controlar la influencia de cada término en el seguimiento, de modo que podamos modificar el tipo de *tracking* realizado según más nos convenga en función de las características del escenario. Si nos encontramos con un entorno donde resulta complicado distinguir los colores de la indumentaria de las personas pero se obtiene una buena extracción de *foreground* nos interesa que la aportación del primer plano sea mayor que la del color. Mientras que si se puede distinguir con claridad los distintos colores de las vestimentas de los objetivos a seguir, preferiremos aumentar la aportación del color frente a la del *foreground*, ya que así la calidad del seguimiento será mayor al poder distinguir mejor entre los distintos objetos que sigue.

### 3.1.1.- Extracción de primer plano

La extracción de primer plano o *foreground* [11, 16, 17, 21, 25, 26] que se ha implementado en el presente trabajo parte del sencillo modelo conocido como *Running Gaussian Average* para añadirle algunos cambios como son la modificación de la inicialización del algoritmo, del espacio de color con el que se trabaja y del valor de los parámetros de actualización, que logran mejorar los resultados obtenidos. En primer lugar, veremos el modelo *Running Gaussian Average* para pasar a ver en qué consisten las modificaciones realizadas para, finalmente, ver paso a paso el algoritmo implementado.

#### Running Gaussian Average

El método *Running Gaussian Average*, tal y como se describe en [17, 21], propone modelar el fondo o *background* mediante el análisis de cada píxel(x,y) de la imagen, tal y como muestra la Fig. 3. 14.-Explicación gráfica del modelo de 1 sola Gaussiana.. La idea consiste en representar el fondo como un modelo probabilístico que caracteriza a cada píxel con una función de densidad de probabilidad Gaussiana, *pdf*, definida por su media  $\mu$  y su varianza  $\sigma^2$ .

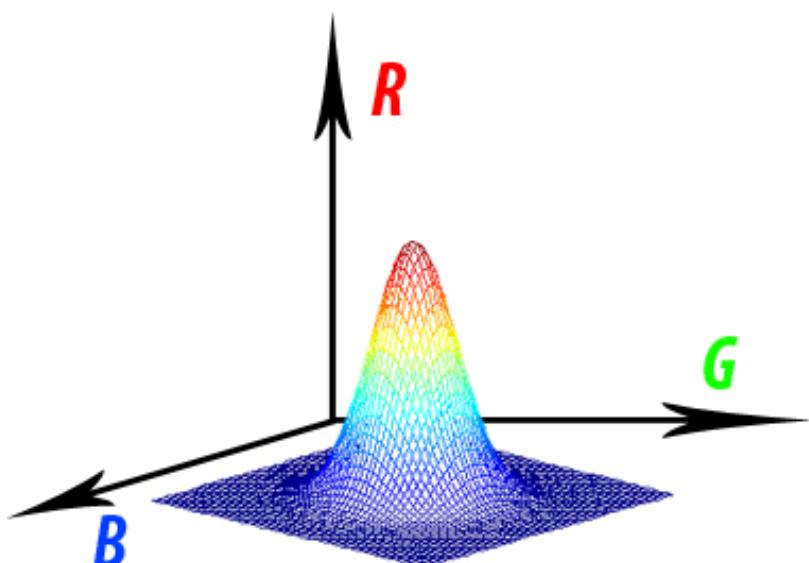


Fig. 3. 13.-Modelo de píxel como una función Gaussiana.

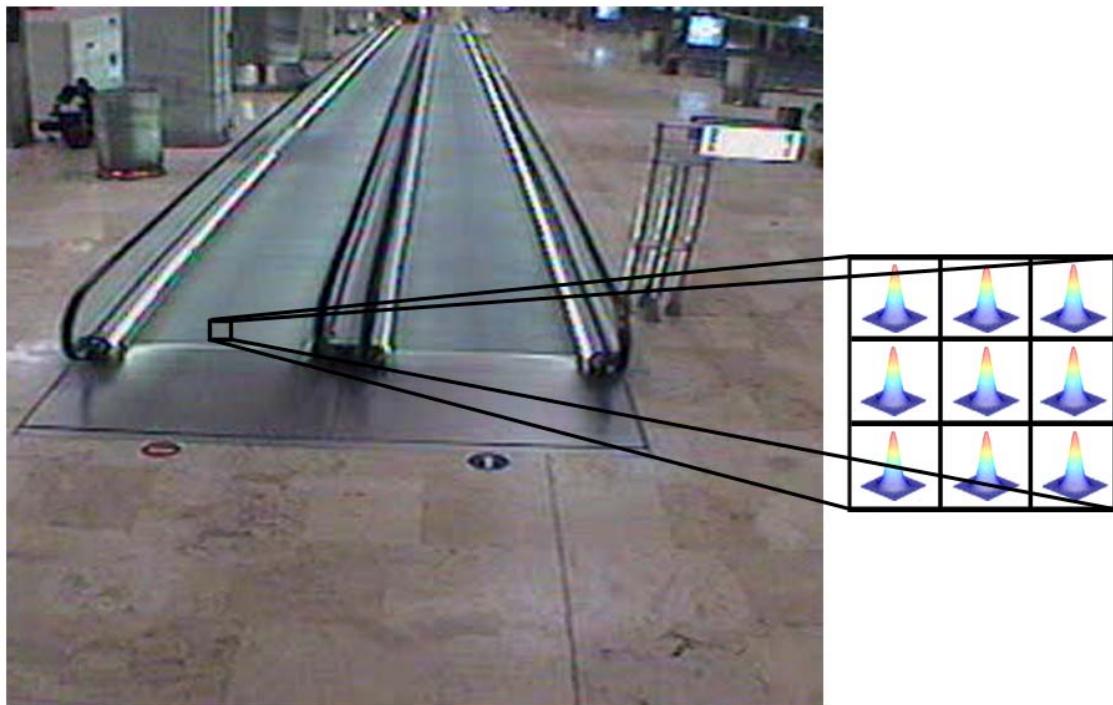
La media  $\mu_t$  y la varianza  $\sigma_t^2$  se actualizan mediante las siguientes ecuaciones:

$$\mu_t = \alpha \cdot I_t + (1 - \alpha) \cdot \mu_{t-1} \quad (3.4)$$

$$\sigma_t^2 = \alpha \cdot Distancia_t^2 + (1 - \alpha) \cdot \sigma_{t-1}^2 \quad (3.5)$$

$$Distancia_t = (I_t - \mu_t)^2 \quad (3.6)$$

donde  $I_t$  corresponde al valor del pixel en el fotograma actual,  $\mu_t$  y  $\sigma_t^2$  son la media y la varianza que caracterizan la función Gaussiana en el actual fotograma del píxel,  $\alpha$  es un peso que determina la velocidad de actualización de la Gaussiana, su valor deberá escogerse a partir del equilibrio entre estabilidad y rápida actualización. La variable  $Distancia_t$ , corresponde a la distancia Euclídea entre la media de la distribución Gaussiana y el valor del píxel en cada instante.



**Fig. 3. 14.-Explicación gráfica del modelo de 1 sola Gaussiana.**

El criterio que se toma para decidir si un píxel pertenece al *background* o bien es un píxel de primer plano viene dado por la siguiente inecuación:

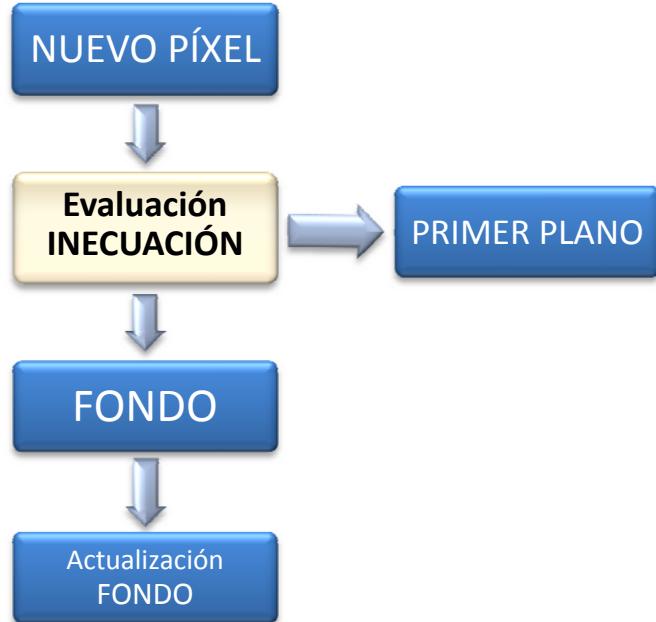
$$|I_t - \mu_t| > k * \sigma_t \quad (3.7)$$

donde  $k$  es el parámetro que fija el umbral de decisión, normalmente se emplea una  $k = 2.5$ . De este modo, **si el píxel evaluado,  $I_t$ , cumple la inecuación, este pasa a ser considerado como primer plano**. Si en cambio no la cumple, es considerado como fondo de escena.

En [17] se añade una última particularidad a este algoritmo, ésta consiste en actualizar únicamente la media y la varianza si el píxel evaluado es considerado como fondo, de este modo se introduce la siguiente ecuación para implementar esta modificación:

$$\mu_t = M \cdot \mu_{t-1} + (1 - M) \cdot (\alpha \cdot I_t + (1 - \alpha) \cdot \mu_{t-1}) \quad (3.8)$$

donde  $M = 1$  si  $I_t$  es considerado como *background* y  $M = 0$  si bien es considerado *foreground*. En resumen, el algoritmo presentado tiene la siguiente estructura:



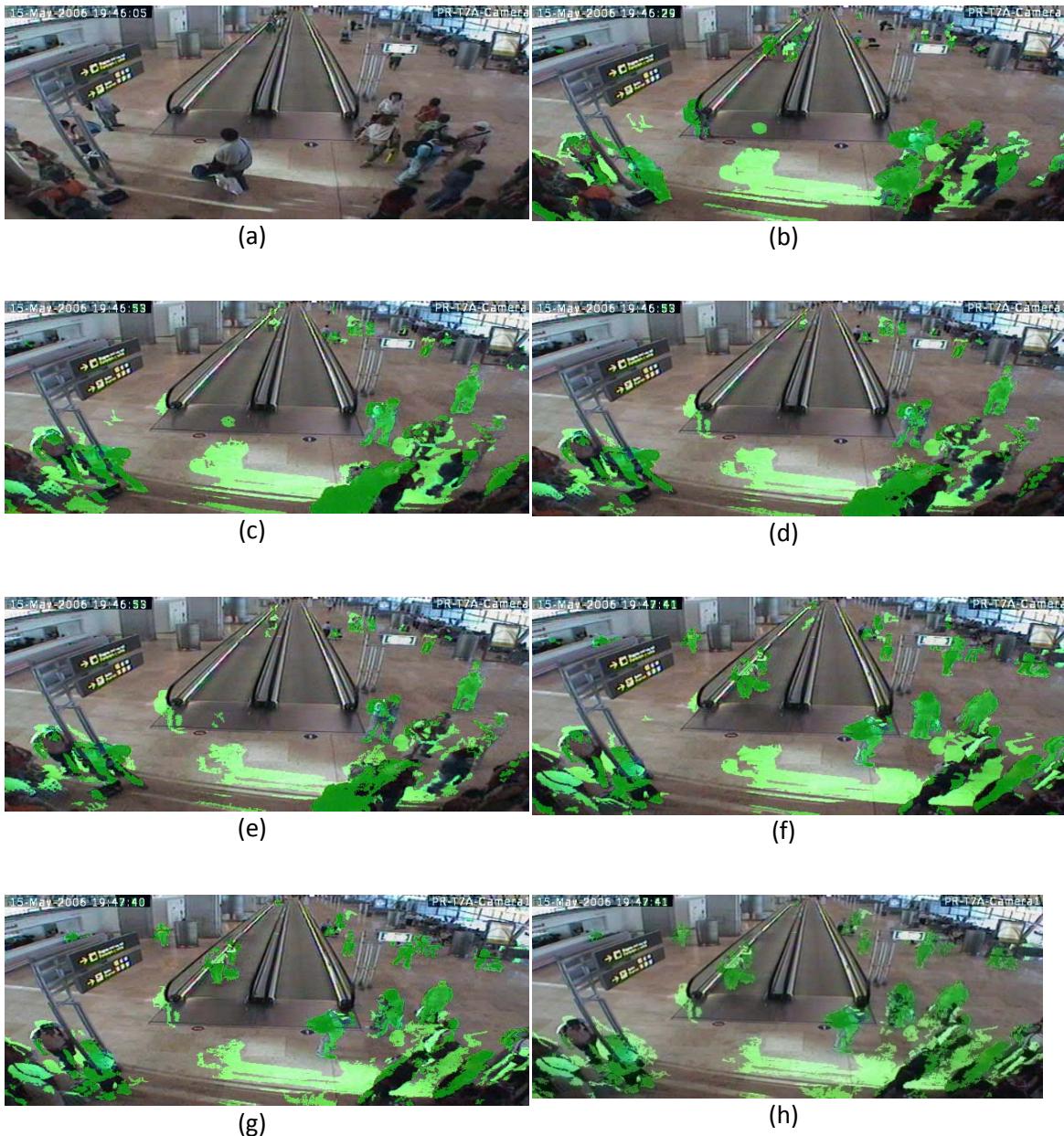
**Fig. 3. 15.-Diagrama de bloques del método *Running Gaussian Average*.**

Así pues, las principales ventajas que ofrece este algoritmo son los bajos requerimientos de memoria y coste computacional al únicamente modelar cada píxel a partir de dos valores, su media  $\mu_t$  y varianza  $\sigma^2_t$ , la flexibilidad que ofrece para poder trabajar con otros espacios de color: RGB, YUV, YCbCr, entre otros, y el poder adaptarse a distintos escenarios según interese un modelo de fondo que responda rápida o lentamente a los cambios del entorno, mediante la modificación del parámetro  $\alpha$ .

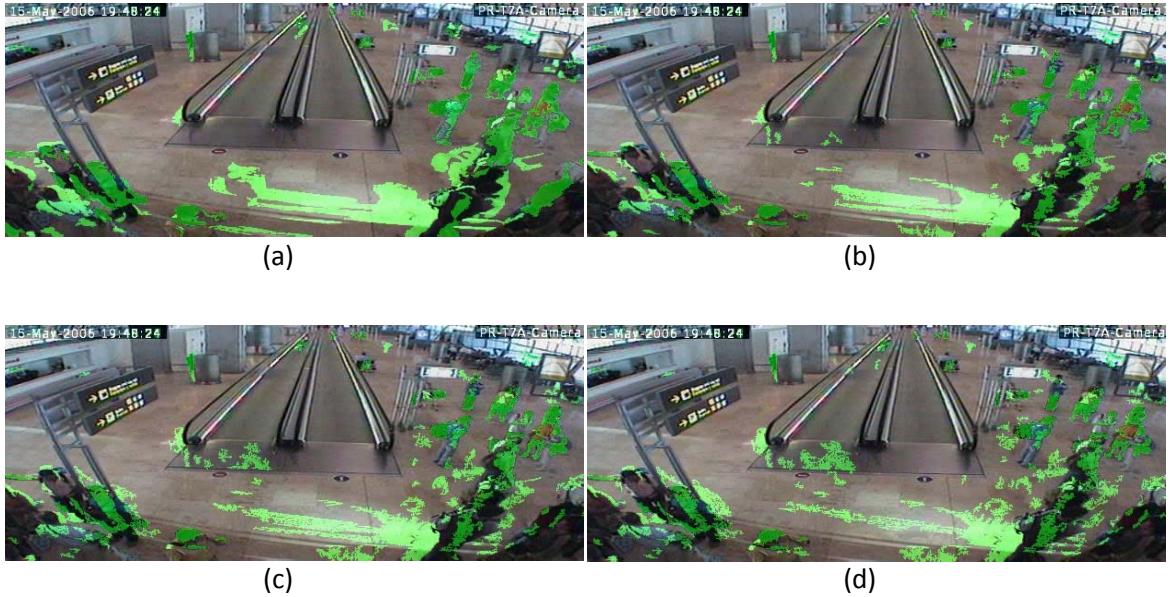
### Fondo adaptativo

Uno de las ambigüedades que presenta el método anterior es cómo inicializar la media y la varianza. Una de las maneras más usadas es inicializar la media con el primer fotograma que obtenemos de la secuencia. Esto puede llegar a suponer un problema si, por un lado, este fotograma presenta algún objeto que claramente no corresponde al fondo de la imagen, lo que provoca que ese objeto se tome, inicialmente, como parte del *background* y no se tenga en cuenta en el seguimiento, lo que puede ser un error. Por otro lado, si el coeficiente de actualización,  $\alpha$ , de la media  $\mu_t$  es muy pequeño, por ejemplo 0.01 (1%), necesitaremos, por lo

menos, 100 fotogramas sin este objeto indeseado para eliminarlo, así pues también es importante escoger adecuadamente el valor de  $\alpha$ . Veamos un ejemplo de un caso parecido al que se acaba de comentar y como varía la adaptación del fondo en función del valor de  $\alpha$  con el que se trabaje, las imágenes siguientes corresponden a la secuencia E1\_4 (para entender la nomenclatura empleada en el etiquetado de las secuencias referirse al Anexo 3):



**Fig. 3.16.-** (a) Observamos una persona al final de la rampa que provocará una mala inicialización de la media, calculada a partir de una secuencia de training de 50 frames. (b) Extracción de foreground, representada por el color verde, obtenida en el frame 51, igual para todos los distintos valores de  $\alpha$ . (c) Extracción de foreground obtenida en el frame 100 con un  $\alpha = 1\%$ . (d) Extracción de foreground obtenida en el frame 100 con un  $\alpha = 15\%$ . (e) Extracción de foreground obtenida en el frame 100 con un  $\alpha = 25\%$ . (f) Extracción de foreground obtenida en el frame 200 con un  $\alpha = 1\%$ . (g) Extracción de foreground obtenida en el frame 200 con un  $\alpha = 15\%$ . (h) Extracción de foreground obtenida en el frame 200 con un  $\alpha = 25\%$ .



**Fig. 3. 17.-**(a) Extracción de foreground obtenida en el frame 290 con un  $\alpha = 1\%$ . (b) Extracción de foreground obtenida en el frame 290 con un  $\alpha = 15\%$ . (c) Extracción de foreground obtenida en el frame 290 con un  $\alpha = 25\%$ . (d) Extracción de foreground obtenida en el frame 290 con un  $\alpha = 35\%$ , el caso extremo donde se observa como las personas que pasan por la rampa quedan casi directamente incorporada como *background*.

Así pues, cuanto mayor sea el valor de  $\alpha$ , más rápido se adaptará el fondo hasta llegar al punto de que cuando una persona que pase por la rampa se pare un pequeño intervalo de tiempo, ésta pasará a formar parte del fondo rápidamente, lo que puede suponer un problema ya que la cantidad de ruido aumenta en gran medida, tal y como se observa en la Fig. 3.17.- (d). Así pues no nos interesa ni un valor excesivamente pequeño de  $\alpha$  ni uno excesivamente grande. El valor elegido para trabajar es una  **$\alpha = 15\%$** , ya que no genera demasiado ruido y logra eliminar inicializaciones erróneas de manera bastante eficiente.

Hay que destacar que el factor de actualización de la media ha dejado de ser el mismo que el de la varianza, no se puede modificar ambos del mismo modo, ya que en el caso de la media la relación es lineal, mientras que en el caso de la varianza es cuadrática, por lo que su modificación afecta de manera muy distinta al resultado obtenido con la extracción de *foreground*. Por lo tanto, redefiniremos las ecuaciones (3.4) y (3.5) del modo siguiente:

$$\mu_t = \alpha_\mu \cdot I_t + (1 - \alpha_\mu) \cdot \mu_{t-1} \quad (3.9)$$

$$\sigma_t^2 = \alpha_V \cdot \text{Distancia}_t^2 + (1 - \alpha_V) \cdot \sigma_{t-1}^2 \quad (3.10)$$

Para finalizar, explicaremos las diferencias entre modificar el factor de actualización de la media,  $\alpha_\mu$ , y el de la varianza,  $\alpha_V$ . Por un lado, modificar  $\alpha_\mu$  implica cambiar la velocidad de actualización de la media de la Gaussiana del píxel. Si su valor es elevado, el modelo probabilístico del píxel se adapta rápidamente a las variaciones del fondo, pero a su vez, es más sensible a cambios de color del píxel y puede provocar el incremento de falsas detecciones. Si su valor es reducido, el modelo

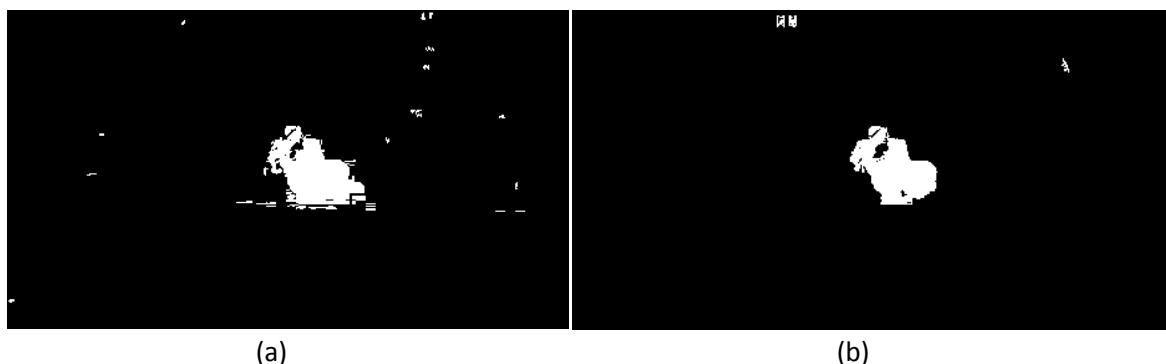
de fondo puede llegar a no actualizarse con la velocidad necesaria para hacer frente a los cambios progresivos del fondo, lo que provocaría a largo plazo, que se generen permanentes falsas detecciones en toda la imagen, debido a que el modelo de fondo de referencia está anticuado respecto al que se requiere en el fotograma actual.

Por otro lado, modificar  $\alpha_V$ , implica alterar la sensibilidad con la que se detectará si un píxel determinado pertenece al fondo de escena o es de primer plano. Aumentando su valor, el modelo de fondo se adaptará rápidamente, de manera que valores puntuales de los píxeles podrán llegar a modificar de manera importante la varianza, y por lo tanto, el umbral de detección será más sensible a pequeños cambios. Pero si lo disminuimos, ocasionará una adaptación lenta a los cambios progresivos, pudiendo no detectarlos y, por lo tanto, haciendo nuestro detector menos sensible.

### Espacio de color

El espacio de color empleado en la extracción de *foreground* es el YCbCr. La elección de este espacio de color se debe principalmente a que minimiza los errores debidos a los cambios de iluminación, brillos y sombras. Las diferencias a nivel teórico entre el espacio de color RGB y el YCbCr se pueden encontrar en el Anexo 2.

Tras haber realizado varias pruebas con el espacio de color RGB, se comprobó que éste añadía más ruido que el espacio de color YCbCr, gracias a la ya comentada reducción del efecto de los cambios de iluminación, brillos y las sombras. Esta mejora se puede apreciar en la Fig. 3.18.



**Fig. 3. 18.- Comparativa entre la extracción de foreground con distintos espacios de color. (a) Usando RGB. (b) Usando YCbCr.**

El principal motivo de la mejora obtenida al trabajar con el espacio de color YCbCr, es el hecho de que los canales Cb y Cr son más resistentes a los cambios de iluminación que los RGB. Como el cálculo de la media y la varianza de la Gaussiana de cada píxel se realiza sobre los 3 canales, al trabajar en este espacio el efecto de los problemas causados por las variaciones de la luminancia sólo nos afecta en una de las 3 componentes, la Y. De este modo conseguimos reducir los problemas causados por la luminancia.

Hay que tener presente que el algoritmo implementado no incorpora ningún tipo de sistema que elimine brillos o sombras ya que se ha considerado que con el cambio de espacio de color los resultados obtenidos ya eran lo suficientemente adecuados.

### **Método empleado:**

El algoritmo requiere un periodo de inicialización, es decir, una etapa durante la cual no se realizará la extracción de objetos de primer plano y que usará para inicializar los distintos parámetros del algoritmo, en nuestro caso, cada una de las distribuciones Gaussianas que modelan el fondo, la media  $\mu$ , y la varianza  $\sigma^2$ . Esta fase de inicialización recibe el nombre de periodo de entrenamiento o *training* ( $t < N$ ), y en el caso implementado constará de 50 frames.

El funcionamiento del algoritmo de detección de *foreground* implementado se realizará a través de los siguientes pasos:

1. **Iniciar las Gaussianas, las de las 3 componentes YCbCr. Normalmente se emplea el valor de la imagen inicial, para el caso de las medias, para posteriormente inicializarlas al valor obtenido durante el periodo de training. En cambio con la varianza se suele emplear el valor 200, aunque posteriormente también se actualiza con la calculada en el periodo de entrenamiento.**
2. **Aprender el modelo probabilístico del fondo que se empleará de referencia durante el periodo de training ( $t < N$ ):**

➤ Centrar la media de las Gaussianas al valor de la media calculada en los  $N$  frames de la secuencia de entrenamiento:

- Medias  $\mu_Y$ ,  $\mu_{Cb}$  y  $\mu_{Cr}$  mediante la siguiente ecuación:

$$\mu = \frac{1}{N} \sum_{t=1}^N \mu_t \quad (3.9)$$



**Fig. 3. 19.- Representación de la imagen media de fondo de escena calculada a partir de la secuencia de training del escenario 1.**

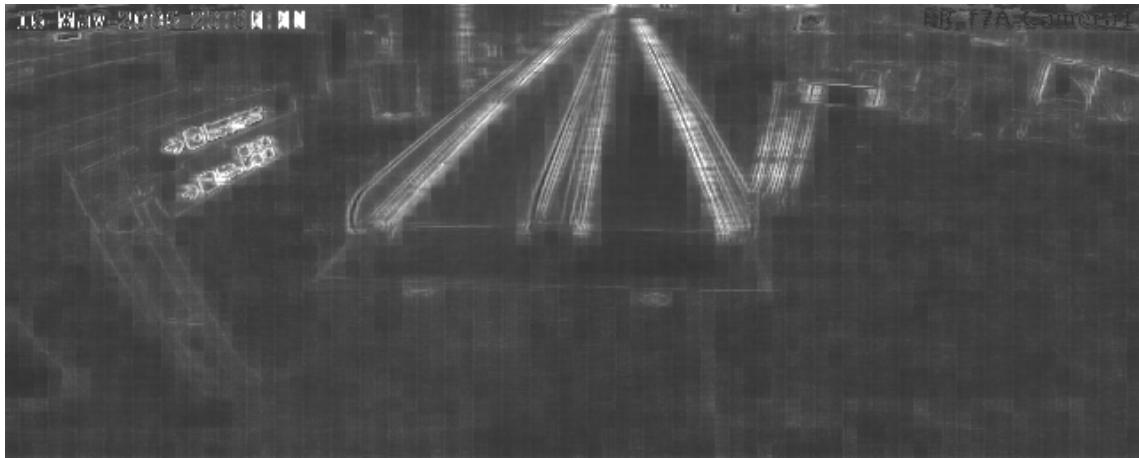
- Calcular el valor de la varianza de las Gaussianas calculada al realizar un segundo barrido del periodo de entrenamiento necesario debido al uso de la media calculada en el apartado anterior:

- Varianza  $\sigma^2$  se calcula mediante las ecuaciones:

$$Distancia^2 = (Y - \mu_y)^2 + (Cb - \mu_{Cb})^2 + (Cr - \mu_{Cr})^2 \quad (3.10)$$

$$\mu_t = \alpha_\mu * I_t + (1 - \alpha_\mu) * \mu_{t-1} \quad (3.11)$$

$$\sigma_t^2 = \alpha_V * Distancia_t^2 + (1 - \alpha_V) * \sigma_{t-1}^2 \quad (3.12)$$



**Fig. 3. 20.- Representación de la imagen varianza calculada a partir de la secuencia de training para el escenario 1.**

- Para los fotogramas siguientes al periodo de training la media y la varianza de las distribuciones Gaussianas para cada componente de YCbCr se actualizarán de la siguiente manera:

$$\mu_t = \alpha_\mu * I_t + (1 - \alpha_\mu) * \mu_{t-1} \quad (3.13)$$

$$\sigma_t^2 = \alpha_V * Distancia_t^2 + (1 - \alpha_V) * \sigma_{t-1}^2 \quad (3.14)$$

$$Distancia^2 = (Y - \mu_y)^2 + (Cb - \mu_{Cb})^2 + (Cr - \mu_{Cr})^2 \quad (3.15)$$

donde  $I_t$  es el valor del píxel en el fotograma actual y  $\mu_t$  la media que caracteriza la función Gaussiana del píxel;  $\alpha_\mu$  es un peso que define la velocidad de actualización de la media de la Gaussiana ( $\alpha_\mu = 15\%$ ),  $\alpha_V$  es un peso que define la velocidad de actualización de la varianza de la Gaussiana ( $\alpha_V = 1\%$ ),  $\sigma_t^2$  es la varianza de la Gaussiana, y la  $Distancia^2$  es la distancia Euclídea entre la media de la Gaussiana del modelo de fondo y el píxel de entrada.

### 3. Una vez finalizado el periodo de entrenamiento ( $t>N$ ), para cada fotograma:

- Para cada uno de los píxeles  $(x,y)$  del fotograma:

- Encontrar la distancia del valor del píxel  $(x,y)$  de entrada respecto a la media de la Gaussiana del modelo de fondo de referencia mediante:

$$Distancia^2 = (Y - \mu_y)^2 + (Cb - \mu_{Cb})^2 + (Cr - \mu_{Cr})^2 \quad (3.16)$$

- Comparar la distancia encontrada con el umbral de detección de píxel según la inecuación siguiente:

$$Distancia > k * \sigma_t \quad (3.17)$$

donde  $k$  es un factor multiplicador, cuya finalidad es aumentar la zona de aceptación de píxel de fondo, emplearemos una  $k = 2.5$ .

1. Si la inecuación se cumple, significa que el píxel  $(x,y)$  de entrada ha variado lo suficiente respecto al del modelo de fondo que lo caracteriza, su distribución Gaussiana, y se considera que no corresponde con los valores propios del fondo. Entonces, consideraremos el píxel de entrada como **píxel de primer plano**.
2. Si la inecuación no se cumple, se considera que el píxel de entrada  $(x,y)$ , no ha variado lo suficiente respecto del modelo de fondo que lo caracteriza, por lo que se considera que corresponde al fondo. De este modo, se toma el **píxel de entrada como fondo** y se procede a actualizar los parámetros de la media y la varianza que caracterizan la distribución Gaussiana que conforma el modelo probabilístico del píxel, para adaptarse a posibles variaciones del fondo, según las ecuaciones (3.13, 3.14 y 3.15).

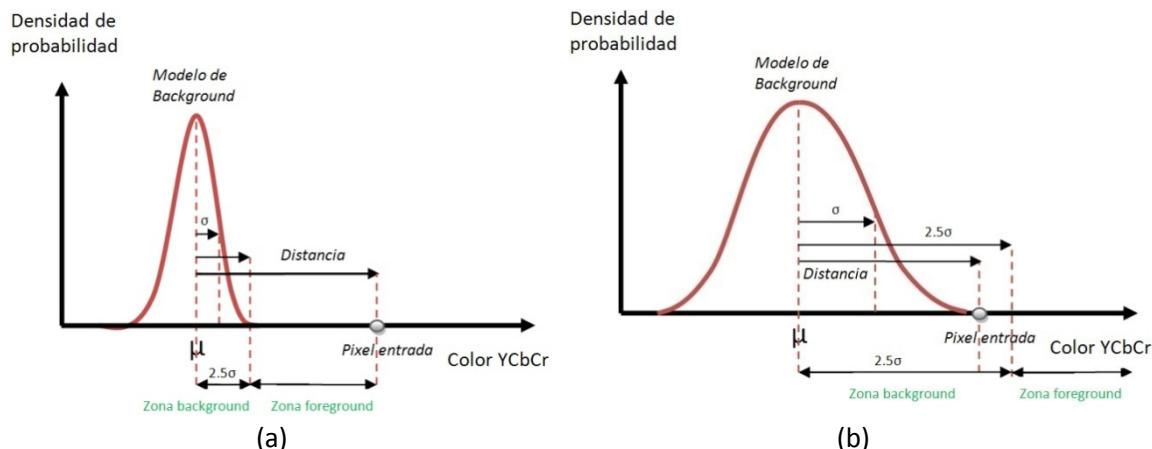


Fig. 3. 21.- (a) Píxel detectado como primer plano. (b) Píxel detectado como fondo.

- Generar la imagen binaria de salida, donde los píxeles representan:

**Negro:** si el píxel de entrada es considerado FONDO.

**Blanco:** si el píxel de entrada es considerado PRIMER PLANO.

- Finalmente, realizaremos un **tratamiento de post-procesado** para eliminar el ruido debido a las falsas detecciones que consistirá en la aplicación de un filtro morfológico de apertura, tal y como se explica en el Anexo 4.

### 3.1.2.- Color: histograma del objeto a seguir

La información de color [3, 9, 13] vendrá determinada por el histograma de color de cada persona. La finalidad de calcular un histograma para cada objeto que seguimos es lograr un seguimiento más selectivo. De este modo, el filtro de partículas deberá seguir el target cuyo histograma más se parezca al que inicialmente estaba siguiendo. El histograma se genera usando el espacio de color YCb e YCr. No se ha implementado el histograma en YCbCr debido a la complejidad que suponía trabajar con un histograma de 3 dimensiones. En cambio, se ha empleado un espacio de color basado en dos canales YCb e YCr que mantiene la información del espacio YCbCr y reduce el coste computacional. La explicación de cómo se pasa de un espacio a otro se encuentra detallada en el Anexo 2.

El histograma de color se genera correspondiendo con la zona que cubre el filtro de partículas, tal y como se puede ver en la Fig. 3.23. El histograma sólo contendrá los valores que, encontrándose dentro del filtro de partículas, sean de primer plano. Veamos un ejemplo de cómo serían los píxeles de primer plano del filtro de la Fig. 3.23 y su correspondiente histograma de color:

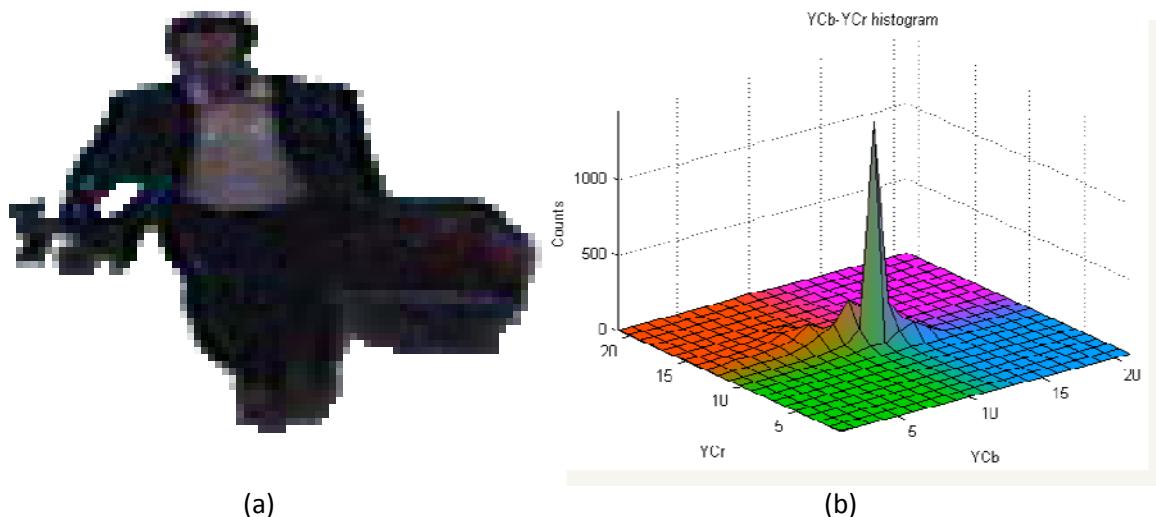


Fig. 3. 22.- (a) Una persona que podría interesar seguir. (b) Ejemplo de un histograma de color en YCb-YCr que podría corresponder a la persona de la figura (a).

Se puede observar como el objeto de la figura anterior presenta un color predominantemente grisáceo con algo de rojo y azul. Este histograma será considerado como el histograma de referencia. Compararemos los histogramas de las partículas del siguiente fotograma con el histograma de referencia mediante la distancia de Battacharyya. Hay que aclarar que el histograma de las partículas se genera a únicamente con los píxeles de *foreground* que ésta contiene.

Si bien el histograma de las partículas se calculará de nuevo cada fotograma, este no será el caso del histograma de referencia. El histograma de referencia se deberá **actualizar** con la

información obtenida en cada *frame*, de modo que sea resistente a cambios de iluminación y ruido. El método empleado para realizar esta actualización es el siguiente:

$$\text{Ref\_Hist}_k^i = \alpha \cdot \text{Ref\_Hist}_{k-1}^i + (1 - \alpha) \cdot \text{Hist}_{k-1}^i \quad (3.18)$$

donde  $\alpha$  es el factor de memoria, que controla la velocidad de actualización del histograma de referencia,  $\text{Ref\_Hist}_k^i$  es el histograma de referencia del target  $i$  en el instante  $k$ , y  $\text{Hist}_{k-1}^i$  es el histograma del target  $i$  en el instante  $k$ , es decir, el histograma asociado al filtro de partículas en el instante  $k$ , una vez realizada la estimación de su nueva posición. El valor de  $\alpha$  elegido ha sido de **0.5**. Esto se debe a que si bien un valor entre 0.1 y 0.3 conlleva una adaptación demasiado rápida, un valor entre 0.7 y 0.9 supone una actualización demasiado lenta, por lo que lo ideal es un valor de  $\alpha$  en el intervalo 0.4-0.6.

La **distancia de Battacharyya** [12, 18, 19, 20, 22] es una métrica que nos permitirá valorar de forma cuantitativa el parecido o verosimilitud entre dos histogramas. En primer lugar, se define el coeficiente de Battacharyya que se calcula mediante la siguiente ecuación:

$$\rho_k = \sum_{j=1}^{N_b} \sqrt{h_k(j)h_{k-n}(j)} \quad (3.19)$$

donde  $h_k(j)$  es el histograma que queremos evaluar en el instante  $k$ ,  $h_{k-n}(j)$  es el histograma de referencia,  $N_b$  es el número de niveles que tienen ambos histogramas. Se define la distancia de Battacharyya como el complementario de este coeficiente:

$$d_{Batt} = 1 - \rho_k \quad (3.20)$$

donde  $d_{Batt}$  puede valer como máximo 1, los histogramas no se parecen en nada, y 0, los histogramas son totalmente iguales. Este es el valor que hemos expresado como *Batt*, en la ecuación (3.2), que contribuye al cálculo del SCORE y, por lo tanto, al peso de las partículas según la ecuación (3.1).

### **3.2.- Propagación de los filtros de partículas en un entorno con perspectiva:**

La correcta adaptación de los filtros de partículas al tipo de entornos en los que trabajamos, escenarios con perspectiva, es fundamental para que el seguimiento se realice correctamente. A continuación, estudiaremos la caracterización de los filtros de partículas en este tipo de escenarios, veremos en detalle las distintas etapas de la dinámica del funcionamiento y los modelos de propagación empleados. Hay que tener presente que lo que se explicará a continuación está particularizado para el escenario 1, pero es aplicable al resto de escenarios. En el Anexo 3 se puede ver las equivalencias de las imágenes y tablas mostradas en este apartado para el resto de escenarios.

#### **3.2.1.- Caracterización de los filtros de partículas según el entorno**

La manera con la que caracterizaremos o representaremos los filtros de partículas es mediante rectángulos. Se ha elegido este tipo de figura geométrica debido a su sencillez, facilidad de caracterización a partir de una anchura y una altura, y por ser una buena aproximación de la silueta de una persona. Seguidamente se muestra un ejemplo de cómo caracterizaremos un filtro de partículas:



**Fig. 3. 23.- Muestra de un filtro de partículas caracterizado por un rectángulo.**

La principal característica del tipo de escenario con el que trabajaremos es la perspectiva. El filtro de partículas no podrá tener el mismo tamaño a lo largo de toda la rampa debido al efecto de la perspectiva. Por lo tanto, el tamaño del filtro se deberá incrementar y disminuir tal y como lo haga la perspectiva en el escenario. De este modo, el tamaño de la región de soporte que se emplea para caracterizar el filtro se modificará en función de cuál sea su posición, concretamente, el valor de su posición vertical. Tomaremos 3 instantes claves, a partir de los cuales, interpolando sus valores, generaremos una tabla con todos los valores de anchura y altura que tendrá el filtro de partículas para todos los valores verticales que pueda tener. Observemos estos 3 instantes para el caso del escenario 1:



**Fig. 3. 24.- Muestra de los 3 instantes claves a partir de los que generaremos la tabla de adaptación del tamaño del filtro de partículas.**

Un resumen de la tabla a partir de la que se realizará la adaptación del tamaño es el siguiente, cabe subrayar que el primer tamaño corresponde con el del blob inicial, que se explicará en el apartado que sigue:

X	Y	W	H
340	0	4	4
...	...	...	...
335	7	4	4
...	...	...	...
327	16	10	8
326	17	11	9
325	18	12	9
...	...	...	...
288	59	29	28
...	...	...	...
207	147	50	65
...	...	...	...
117	239	50	65

**Tabla 3. 1.- Caracterización del tamaño del filtro de partículas en el escenario 1.**

Hay que tener presente que la tabla tiene 240 valores, la anchura de la imagen en píxeles, y que todos los valores de la tabla están expresados en píxeles. En el Anexo 3 se puede ver la tabla completa, así como las del resto de escenarios.

Del mismo modo que caracterizamos los filtros, también tenemos que caracterizar las partículas que contienen los filtros. Representaremos exactamente del mismo modo las partículas que los filtros, es decir, mediante rectángulos, y la modificación de su tamaño se realizará del mismo modo. Emplearemos 100 partículas para cada filtro, de modo que la posición del filtro de partículas surgirá de la estimación de estas 100 partículas según la ecuación (2.30). Se ha escogido un número de 100 partículas debido a que tras realizar varias pruebas con otros valores se decidió que era una cantidad suficiente para obtener buenos resultados sin cargar excesivamente de información el sistema.

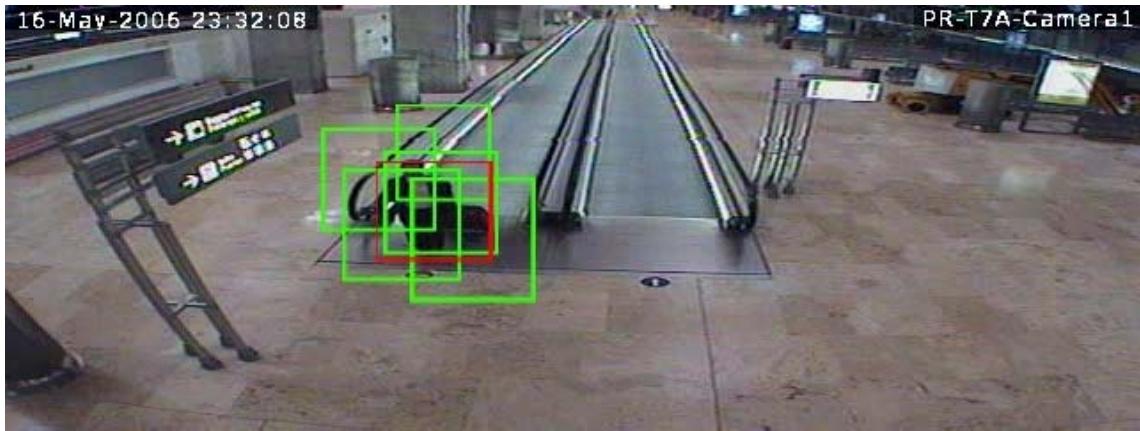


Fig. 3. 25.-Muestra del filtro de partículas anterior, rojo, estimado a partir de 5 partículas, en verde, también caracterizadas por un rectángulo.

### 3.2.2.- Creación y destrucción de los filtros de partículas

Tal y como se describió en el apartado 2.2, *Dinámica del seguimiento*, los filtros de partículas deben pasar por una serie de etapas durante su vida. Primero deben ser inicializados, para posteriormente realizar el seguimiento y finalmente morir o ser destruidos una vez hayan finalizado el seguimiento y su existencia no tenga sentido. En este apartado describiremos con detalle la primera y última de estas etapas.

En primer lugar, deberemos definir claramente cuál es la situación que provoca que se tenga que crear un filtro de partículas. Esta situación corresponde a la **inicialización** del filtro de partículas y vendrá determinada por la aparición de lo que denominaremos **blob inicial**.



Fig. 3. 26.- Imagen perteneciente al escenario 1 donde se puede ver el tamaño del blob inicial.

El blob inicial es el tamaño mínimo de píxeles de *foreground* necesario para que el filtro de partículas se cree y lo fijaremos en 16 píxeles para el escenario 1. La inicialización funcionará del siguiente modo: *si en la zona de inicialización se detecta que el foreground presente tiene el mismo tamaño o superior al del blob inicial, se iniciará la creación de un nuevo filtro de*

**partículas.** Cabe destacar, que no podrá haber más de un filtro de partículas a la vez en la zona de inicialización, por lo que primero el filtro deberá salir de ésta para que se pueda generar otro nuevo. La situación de la zona de inicialización es la que se puede ver en la siguiente imagen:



Fig. 3. 27.- Imagen que muestra el tamaño de la zona de inicialización y su ubicación para este escenario, el 1.

Es importante resaltar, que el tamaño de las imágenes que nos muestra esta cámara, y el resto, tienen un tamaño de 640x240 píxeles, y el origen se encuentra en la esquina superior izquierda. Caracterizaremos la zona de inicialización a partir de su posición y su tamaño en la imagen tal y como se puede ver en la tabla que sigue:

X (centro)	Y (centro)	W (anchura)	H (altura)
334	11	17	9

Tabla 3. 2.- Valores, en píxeles, de los parámetros necesarios para caracterizar completamente la zona de inicialización del escenario 1.

En segundo lugar, el **seguimiento** se realizará según el diagrama de bloques de la figura 2.9, con las modificaciones añadidas en el apartado 4.1.

Finalmente, decidiremos la **destrucción** o desaparición del filtro a partir del momento en que el filtro llega a la zona de desaparición o destrucción. Primeramente, caracterizaremos esta zona del mismo modo que lo hicimos para la zona de inicialización:

X (centro)	Y (centro)	W (anchura)	H (altura)
247	141	145	86

Tabla 3. 3.- Valores, en píxeles, de los parámetros necesarios para caracterizar completamente la zona de destrucción del escenario 1.

El criterio de destrucción es el siguiente: ***cuando el filtro llegue a la zona de destrucción éste será destruido si permanece en ésta durante al menos 100 frames o si habiendo entrado, decide salir y está durante 20 frames fuera de la zona de destrucción.*** Este criterio se ha establecido de este modo pensando en la detección de tumultos. La mayoría de tumultos, por no decir todos, tiene lugar en la zona de destrucción, por lo tanto no interesa que los filtros que permanezcan en ésta no sean destruidos de forma inmediata. En cambio, si salen de la zona de destrucción, como raramente ocurren tumultos fuera de ésta, nos interesa que se eliminen de manera más o menos rápida. A continuación se observa la posición de esta zona según la vista de la cámara. Se añade una zona de seguridad que se define como el espacio máximo que puede ocupar un filtro que se encuentre entre la zona de desaparición y el límite inferior de la imagen mostrada por la cámara.

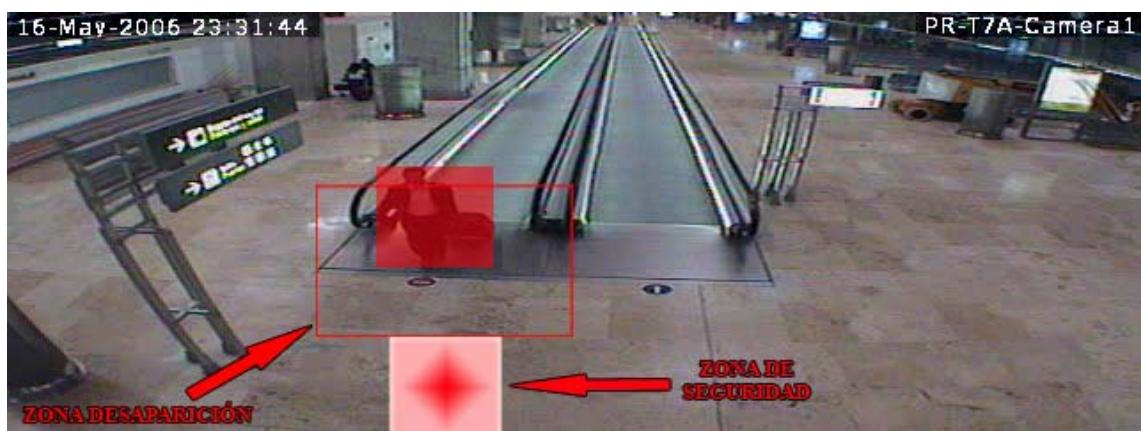


Fig. 3. 28.- Imagen que muestra el tamaño de la zona de desaparición y su situación para esta rampa en concreto.

### 3.2.3.- Propagación de los filtros adaptada a la perspectiva

La propagación de los filtros de partículas, en concreto, de las partículas que lo componen, ya que realmente el filtro no se propaga, sino que son las partículas las que lo hacen y, a raíz de esto, el filtro se propaga conforme a éstas al ser su centroide la estimación de los centroides de éstas.

El modelo de propagación elegido, es decir, la manera a partir de la cual se distribuirán las partículas en el espacio vendrá determinada mediante la adición de un ruido Gaussiano a la posición actual de las partículas. Este ruido se caracteriza por una media, la posición del instante actual de la partícula, y una varianza, que dependerá de la posición de la partícula en la imagen del mismo modo que lo hace su tamaño, tal y como se muestra en la siguiente tabla 3.4. Por lo tanto, la propagación que realizan las partículas es, respecto al espacio, de modo circular.

Fruto de la perspectiva que presenta el escenario, se ha planteado una modificación a este modelo de propagación que consiste en propagar las partículas del mismo modo como

evoluciona la rampa, al ser este el movimiento habitual de las personas, y doblando la probabilidad de que las partículas se propaguen hacia adelante según esta dirección, por lo tanto, de este modo favoreceremos a que las partículas *acierten* con la nueva ubicación del target.

X	Y	W	H	VAR
340	0	4	4	3
...	...	...	...	...
335	7	4	4	3
...	...	...	...	...
327	16	10	8	4
326	17	11	9	4
325	18	12	9	4
...	...	...	...	...
288	59	29	28	9
...	...	...	...	...
207	147	50	65	20
...	...	...	...	...
117	239	50	65	20

Tabla 3. 4.- Caracterización del tamaño y de la varianza de propagación de las partículas en el escenario 1.

Para realizar este tipo de propagación, a la que llamaremos *morphing*, deberemos obtener, en primer lugar, la ecuación de la recta que modela la rampa. En el escenario 1 esta recta se puede modelar a partir de la siguiente ecuación:

$$Y = -1.083 \cdot X + 368 \quad (3.21)$$

Una vez calculada la recta, obtendremos el valor del ángulo  $\alpha$ , el que se muestra en la siguiente imagen.

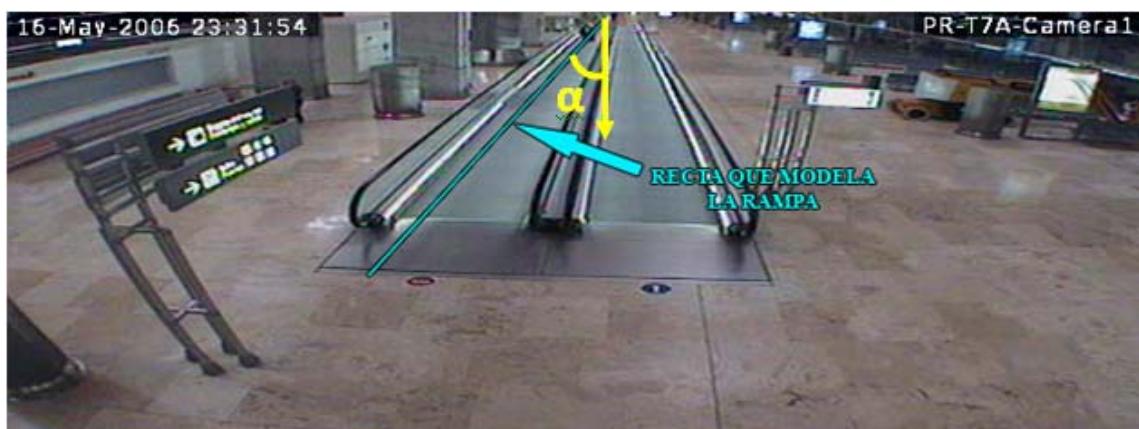


Fig. 3. 29.- Imagen de la recta con la que modelamos la rampa.

La idea es pasar de los ejes (X, Y) de la imagen original a obtener unos nuevos ejes ( $X'$ ,  $Y'$ ) a partir de su rotación usando el ángulo  $\alpha$ . Hay que tener presente que esta rotación se aplica sobre cada partícula y se realiza respecto centroide del filtro, tomando este punto como posición de referencia para efectuar la rotación. Observemos la siguiente imagen para tener una idea de esta rotación:

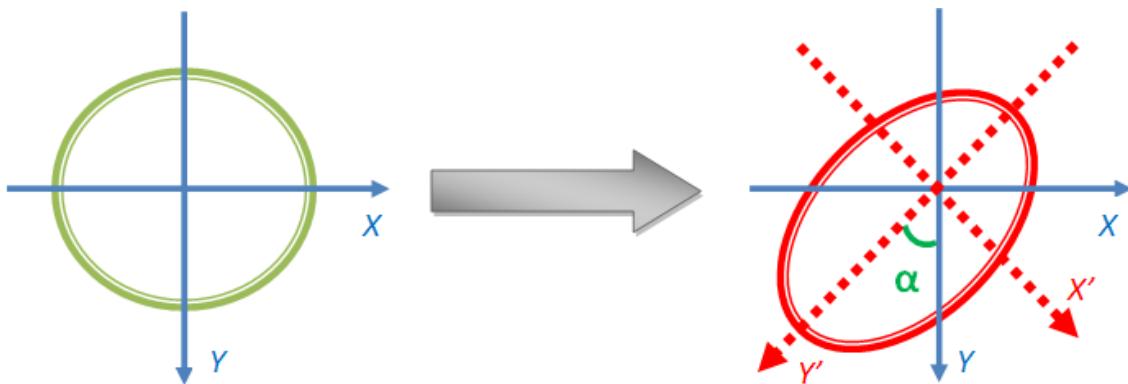


Fig. 3. 30.- Ejemplo gráfico del funcionamiento del *morphing* implementado a partir del ángulo  $\alpha$ .

El método empleado para realizar este cambio de ejes es mediante los siguientes pasos:

1. Se realiza la **traslación** de la posición actual de la partícula al punto (0,0). Esta traslación se realiza mediante la substracción de la posición del centroide del filtro.
2. **Escalamos** el valor de la posición de la partícula en función de lo que nos interese. En nuestro caso sólo nos interesa doblar el eje positivo de las Y's.
3. **Rotamos** la posición de la partícula con el ángulo  $\alpha$ , según el siguiente par de ecuaciones:

$$Y' = X \cdot \cos \alpha - Y \cdot \sin \alpha \quad (3.22)$$

$$X' = X \cdot \sin \alpha + Y \cdot \cos \alpha \quad (3.23)$$

4. Deshacemos la **traslación** realizada en el punto 1 sumando la posición del centroide del filtro a la posición de la partícula.
5. Añadimos el ruido que deseemos a la posición actual, tal y como se ha explicado antes, emplearemos un ruido Gaussiano.
6. Finalmente, si la posición obtenida tras añadir el ruido está por detrás del centro del filtro volvemos a propagar la partícula. De este modo, forzamos que las partículas tiendan a propagarse en el sentido descendente, tal y como se ve en la siguiente imagen:

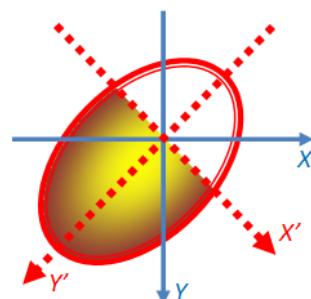


Fig. 3. 31.- Ejemplo gráfico del funcionamiento del *morphing* doblando la propagación en la zona amarilla.

# 4.- Seguimiento en un escenario aglomerado y detección de tumultos

---

En este capítulo se explicara, por un lado, como hay que adaptar el seguimiento realizado mediante los filtros de partículas para que se adapte a las condiciones del entorno y a las características que presenta el movimiento de las personas que nos interesa seguir. Por otro lado, especificaremos qué consideramos como una situación con *tumulto*, explicaremos el etiquetado de las secuencias llevado a cabo y cuáles son las estrategias que han sido implementadas para detectar este tipo de situaciones. Finalmente, se definirán las métricas empleadas para determinar la calidad del algoritmo implementado a nivel de detección de tumultos.

## 4.1.- Adaptación al seguimiento con múltiples targets:

Las medidas [4, 5, 8, 9, 10, 27] que debemos aplicar al seguimiento para que éste se realice adecuadamente se dividen en dos grandes bloques. En primer lugar, se encuentran las que evitan que el filtro de partículas *sigan* personas que no deseamos seguir, como pueden ser las que se encuentran en la rampa ascendente o fuera de ésta. En segundo lugar, se deben tomar las medidas necesarias para eliminar un filtro de partículas cuando éste esté realizando un seguimiento totalmente erróneo, su seguimiento sea redundante, es decir, ya se esté llevando a cabo por otro filtro de partículas o bien, que ya ha finalizado el seguimiento por la rampa y puede empezar a realizar un seguimiento que diste mucho del deseado y suponga un problema para la posterior detección. Cabe destacar que este tipo de situaciones no se dan en todas las secuencias, pero sí en las suficientes como para que exista la necesidad de hacer algo al respecto, es decir, implementar los criterios que se explican más adelante. A continuación analizaremos estos dos bloques y veremos en detalle las situaciones que han provocado el desarrollo de estas medidas con la intención de poder evitarlas.

### 4.1.1.- Criterios de prevención de un mal seguimiento:

Los criterios de prevención de un mal seguimiento implementados en el algoritmo son una manera de solucionar pequeños problemas que han surgido al utilizarlo en distintas secuencias. Tal y como se explicará a continuación, hay algunos que están ligados a determinadas características del escenario y otros al comportamiento de los filtros de partículas. Es importante resaltar que estos criterios logran su objetivo mediante el bloqueo de las partículas del filtro, sin destruir el filtro de partículas. De este modo, estos criterios actúan como una especie de *guía* o restricción de cara a las partículas evitando que se propaguen por donde no deben.

## Recta de bloqueo

El primer criterio de prevención surge fruto del problema que supone que por las dos rampas se muevan personas. Esto puede llegar a provocar que el filtro de partículas no distinga entre las dos rampas y realice el seguimiento por la que no debe. El principal motivo que ocasiona este comportamiento es debido a que si la cantidad de personas en la rampa ascendente es mayor que la de la descendente, las partículas tienden a seguir las ya que el peso que éstas reciben al seguirlos es mayor al disponer de más píxeles de *foreground*. Además, al tratarse de la rampa ascendente, las personas de esta rampa tienen un mayor tamaño aparente, ocasionado por el efecto de la perspectiva, que contribuye a incrementar este efecto. Veamos un ejemplo donde se aprecia esta situación y como se logra solventar gracias a la aplicación de este criterio:

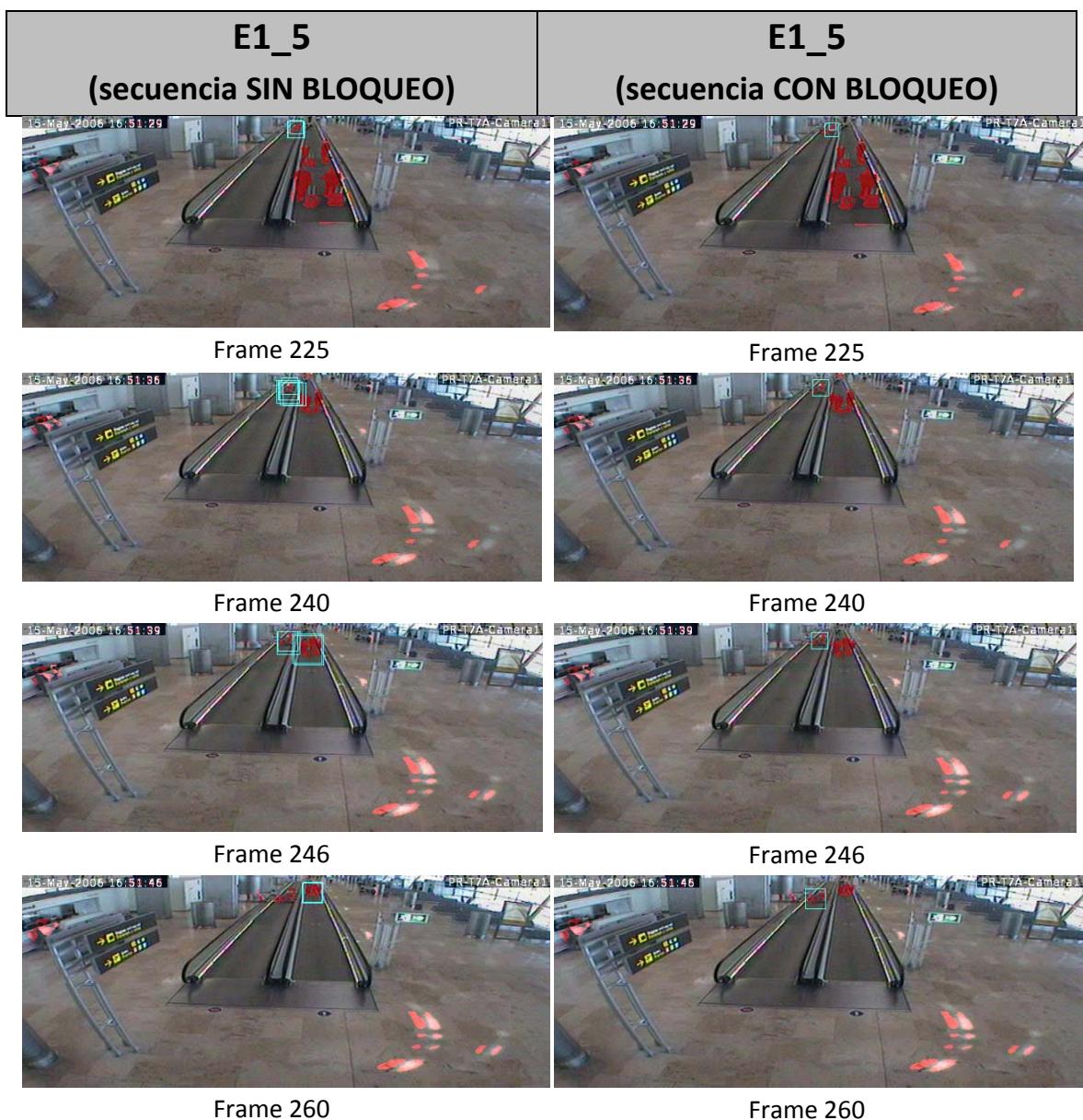


Fig. 4. 32.- Ejemplo de cómo se soluciona el problema de que el filtro siga la gente que circula por la rampa ascendente gracias al criterio de bloqueo en la secuencia E1\_5.

El criterio implementado es bastante sencillo, simplemente *definiremos una recta, que denominaremos recta de bloqueo, a partir de la cual asignaremos un peso nulo a todas las partículas que se extiendan más allá de la recta de bloqueo durante la propagación*. De este modo, eliminaremos que éstas contribuyan en la estimación de la posición del filtro de partículas, asegurando que el filtro no realice el seguimiento de las personas que se muevan por la rampa ascendente. La recta de bloqueo para el escenario 1 tiene las siguientes características y está ubicada en la posición que se observa en la siguiente figura:



**Fig. 4. 33. – Ejemplo de dónde está situada la recta de bloqueo en el escenario 1.**

Básicamente, para el resto de escenarios la posición será la misma (tal y como se puede ver en el Anexo 3), entre las dos rampas. Es importante destacar que la recta empieza en la ordenada 0, de este modo aseguramos que ninguna partícula que la sobrepase por encima se tenga en cuenta. En cuanto al final de la recta de bloqueo, este depende de cada escenario, pero se ha dibujado de manera que finalice con la rampa sin afectar al seguimiento que tenga lugar fuera de ésta.

#### Bloqueo de las partículas que sobrepasen la salgan de la rampa

Este tipo de bloqueo es necesario aplicarlo cuando por el lado exterior de la rampa descendente se encuentren personas, de manera que éstas puedan confundir al filtro de partículas durante el seguimiento. Esta situación tiene lugar sobre todo en el escenario 4, ya que es el único, de los implementados, en el que se da esta situación. La idea empleada para solucionar este tipo de situaciones es similar a la empleada con la recta de bloqueo. En este caso, partiendo de la recta que modela la rampa definiremos una distancia máxima a la que las partículas podrán estar, de manera que, cuando éstas la sobrepasen se les asignará un peso nulo, eliminando la posibilidad de que el filtro de partículas se desplace hacia ese lado de la rampa. Veamos como mejora el seguimiento si se aplica este criterio:



Fig. 4. 34.- Imagen correspondiente al frame 192 de la secuencia E4\_4 donde se puede ver la cantidad de gente que hay en el lado izquierdo y como se consigue que ésta no afecte al seguimiento.

El criterio que aplicaremos será el siguiente: *a partir de la definición de una distancia máxima a la recta que modela la rampa asignaremos un peso nulo a todas las partículas que la sobrepasen*. Esta distancia máxima se definirá respecto al eje X y corresponderá con el 150% de la anchura del filtro en esa posición.

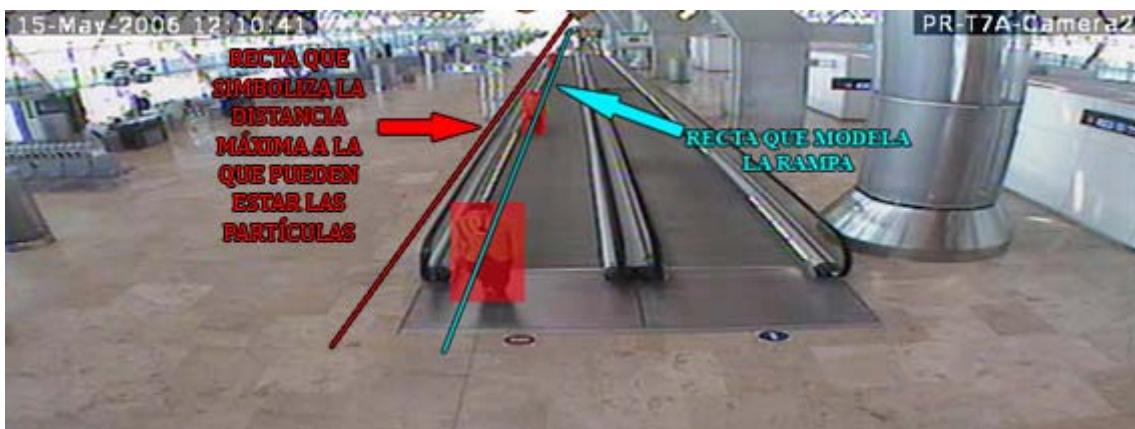


Fig. 4. 35.- En esta imagen se observa una idea de a partir de dónde las partículas reciben un peso nulo, es decir, una vez sobreponen la recta en rojo.

### Alejamiento excesivo de las partículas

En algunos casos ocurre que cuando hay dos personas más o menos próximas en la rampa, las partículas de un mismo filtro se distribuyen de manera más o menos homogénea entre estas dos, por lo que la estimación de la posición del filtro resulta estar en el centro de las dos, donde no hay nadie. Tal y como se puede ver en la imagen de la derecha:

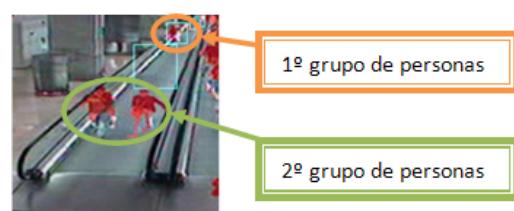
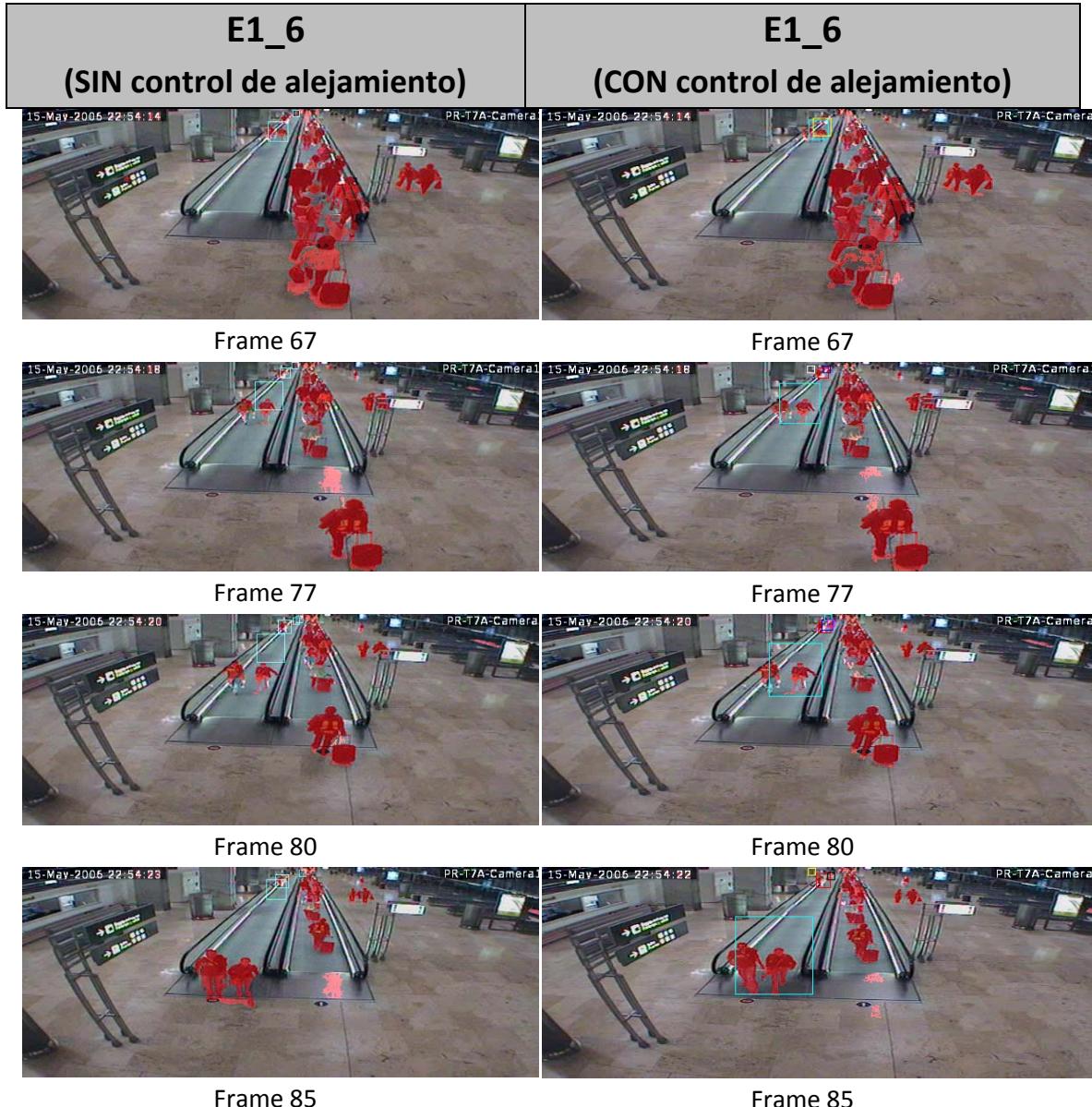


Fig. 4. 36.- Imagen donde se observa el filtro de partículas entre dos grupos de personas.

Esta situación se puede solucionar si se eliminan las partículas que se alejen demasiado del centroide del filtro, ya que se considera que el *target* u objeto a seguir tiene un tamaño concreto, por lo que las partículas no deberían alejarse demasiado de este tamaño, con lo que se logran evitar falsos seguimientos. Observemos cómo se logra solucionar este problema y, por lo tanto mejorar el seguimiento gracias a la aplicación de este criterio:



**Fig. 4. 37.- Serie de imágenes pertenecientes a la secuencia E1\_6, donde se observa claramente la mejora que supone aplicar este criterio ya que elimina la ambigüedad que genera tener dos objetivos muy próximos y permite que el filtro de partículas se centre en uno. El color rojo indica foreground.**

El criterio implementado **consiste en asignar peso nulo, eliminar, a aquellas partículas que se alejen del centroide del filtro más del 150% del tamaño del filtro.**

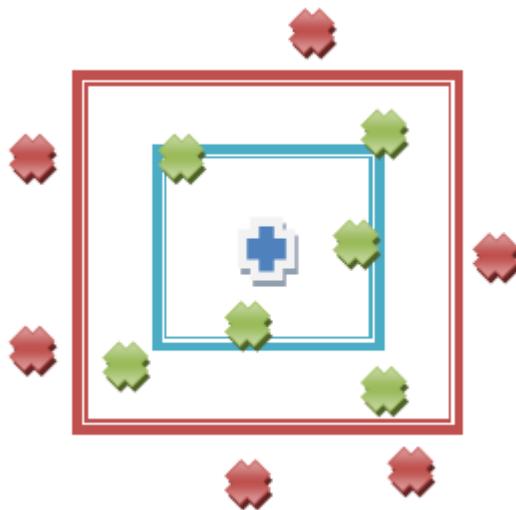


Fig. 4. 38.-Representación del funcionamiento de este criterio. El filtro de partículas tiene el centroide en la cruz azul y su tamaño es el rectángulo del mismo color. El rectángulo rojo representa el 150% del tamaño del filtro y, por lo tanto, la distancia a partir de la cual las partículas serán eliminadas. Las partículas rojas son eliminadas y las verdes sobreviven.

#### **4.1.2.- Criterios de destrucción de los filtros de partículas:**

La finalidad de los criterios de destrucción de los filtros de partículas es destruirlos cuando éstos realicen un seguimiento erróneo o se encuentren en una situación que mediante la destrucción normal, las explicadas en el apartado 3.2.2, el filtro de partículas no se vaya a destruir o tenga que transcurrir demasiado tiempo para que esto ocurra. Por lo tanto, estos criterios pretenden, en cierto modo, forzar la destrucción del filtro de partículas cuando sea necesario, es decir, tenga lugar una de las 4 situaciones que se explican a continuación.

##### **Baja eficiencia**

Tal y como se vio en el apartado 2.1.3, el referente a los filtros de partículas, el parámetro de eficiencia, ecuación (2.25), y la tasa de supervivencia, ecuación (2.26), nos permiten determinar cuando las partículas de un filtro se están degenerando, es decir, tiende a sobrevivir un pequeño número de partículas y, en el peor caso, una única partícula. Esto puede ser debido a que el filtro ya no tiene nada que seguir, debido a que lo que estaba siguiendo ha salido de la imagen y, por lo tanto, del rango visual de la cámara. Esta situación se puede ver en la siguiente secuencia de imágenes, perteneciente a la secuencia E1\_2:



Frame 242

Frame 246

**Fig. 4. 39.- En esta imagen se puede ver como el filtro de partículas sigue vivo pese a que no hay nada que seguir, la persona se encuentra fuera del rango de la cámara.**

Si este tipo de situaciones tienen lugar, el filtro debe ser destruido, ya que no nos interesa que permanezca en la imagen si no tiene sentido que lo haga. En este tipo de situaciones, el parámetro de eficiencia tiende a valer 1. Esto se debe a que como no existe información en la imagen que pueda aportar peso a las partículas, éstas tienden, debido al comportamiento a concentrarse en pocas partículas. De este modo, si **un filtro de partículas presenta una tasa de supervivencia del 10%, lo que significa que sólo el 10% del total de partículas sobrevive, o tienen peso en ese instante, el filtro será destruido**. Para garantizar que esta situación no se haya dado de modo puntual y provoque la destrucción del filtro, esperaremos a tener dos fotogramas consecutivos con este valor de la tasa de supervivencia para eliminar el filtro.

Hay que tener presente que tanto la eficiencia como la tasa de supervivencia son indicadores de que algo en el funcionamiento del filtro de partículas no está transcurriendo como debería, por lo que fijar esta condición sirve para garantizar que los filtros que no estén desarrollando el comportamiento que se espera de ellos serán destruidos.

El motivo por el cual se ha fijado el umbral de destrucción en el 10% es porque se considera, y las pruebas realizadas lo corroboran, que ya es un valor suficientemente pequeño a partir del cual el comportamiento del filtro empieza a no ser el adecuado.

### Dispersión explosiva

Esta situación tiene lugar al final del seguimiento y, por lo tanto, al final de la rampa. Además se ha de dar el caso de que al final de la rampa exista una multitud. De este modo, cuando el target que el filtro de partículas ha seguido a lo largo de la rampa llega al final de ésta y se mezcla con la multitud o desaparece de la imagen de la cámara, el filtro de partículas empieza a seguir a toda la multitud, es decir, las partículas se propagan de modo similar al de una explosión. Es por esto que llamamos a esta situación dispersión explosiva. Observemos una imagen que muestra esta situación perteneciente a la secuencia E1\_4:

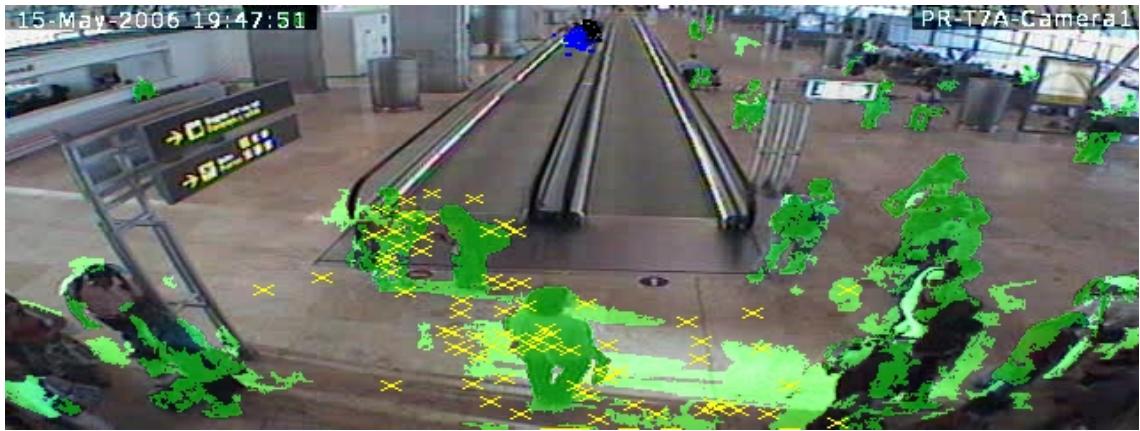


Fig. 4. 40.- Cada cruz amarilla representa el centroide de una partícula y el color verde indicada el *foreground* detectado.

Tal y como se puede ver en la imagen anterior, el filtro de color amarillo intenta seguir varios objetivos simultáneamente sin seguir ninguno bien, lo que provoca que el filtro en realidad esté en la siguiente posición:



Fig. 4. 41.- El filtro de partículas está siguiendo claramente a una persona, pero ésta no es la que debería estar siguiendo, sino la que se ha detenido al final de la rampa. El color rojo indicada el *foreground* detectado.

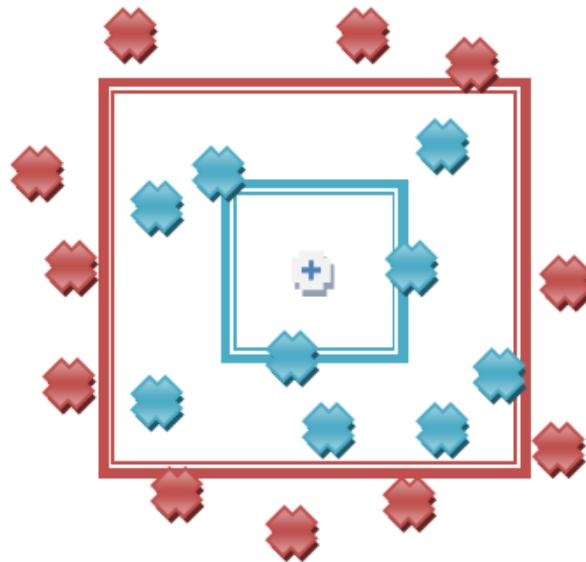
A simple vista, parece que el filtro de partículas está realizando adecuadamente el seguimiento. Aunque si observamos el seguimiento 10 frames antes:



**Fig. 4. 42.-** El filtro de partículas está siguiendo claramente a una persona en la rampa. El color rojo indicada el *foreground* detectado.

Se observa como en esta última imagen, la persona que se estaba siguiendo en la Fig. 4.41, ni siquiera aparece 10 frames antes. Esto es debido a que este individuo viene de la parte inferior de la imagen, y la dispersión explosiva que tiene lugar, conlleva que se siga a esta persona y se deje de seguir a la que correspondería. Ésta es la principal razón por la que los filtros que presenten este tipo de situaciones deben ser eliminados, ya que la estimación que está realizando es totalmente errónea.

Por lo tanto, definiremos que ***un filtro de partículas deberá ser destruido cuando un número elevado de partículas, superior al 50%, se alejen del centroide del filtro, más del 150% del tamaño del filtro en ese instante. Si esta situación ocurre en dos frames consecutivo, dicho filtro será eliminado.***



**Fig. 4. 43.-** Ejemplo visual de cómo un filtro con 20 partículas, será destruido ya que 11 partículas están fuera de la zona delimitada por la zona roja, el máximo permitido por este criterio a la que pueden estar las partículas.

## Solapamiento de filtros

Esta situación no surge del todo debido a un problema provocado por un mal seguimiento, sino que es más bien una manera de simplificar el seguimiento y facilitar la distinción de los filtros. La implementación realizada hasta el momento va generando filtros de partículas conforme van apareciendo personas en la zona de inicialización, estos realizan el seguimiento por la rampa y desaparecen al final. El inconveniente reside en que una misma persona puede ser seguida por multitud de filtros de partículas. Para evitar que esto suceda se ha implementado este criterio con el siguiente resultado:

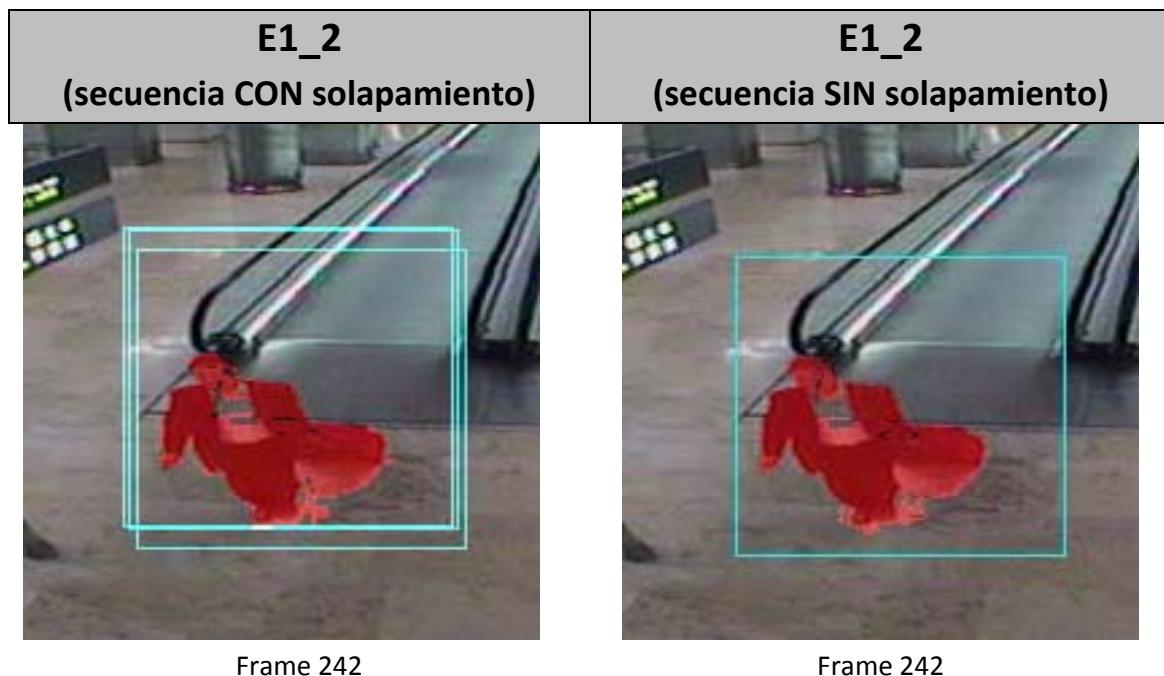
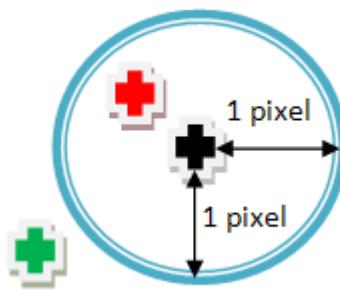


Fig. 4. 44.- Imagen ampliada del frame 242 de la secuencia E1\_2 donde se puede observar cómo, tras la aplicación de este criterio, sólo 1 filtro de partículas sigue a una persona. El color rojo indica el foreground extraído.

Tal y como se puede ver en la primera imagen, para un único objetivo algoritmo llega a generar hasta 3 filtros distintos. Lo ideal sería que un filtro siguiera a una persona, o un mismo grupo, y no varios filtros a una misma persona. Además, esto supone un gasto en coste computacional innecesario. Por lo tanto, si aplicamos este criterio logramos que el seguimiento individual se realice con un único filtro. Así pues el criterio implementado consiste en la **destrucción del filtro más antiguo si éste sigue un target que ya está siguiendo otro filtro**. Como se puede dar el caso de que dos filtros que siguen distintos targets se lleguen a solapar, si uno está parado en la rampa y el otro se mueve o porque uno va más rápido que el otro, se ha añadido un periodo de histéresis, de manera que el criterio sólo se aplica si esta situación sucede en 2 frames consecutivos. Se considerará que dos filtros están juntos cuando sus centroides disten menos de dos píxels, tanto en el eje Y como en el X.



**Fig. 4. 45.-** Cada cruz representa un centroide de un filtro de partículas distinto. El filtro rojo será eliminado al estar demasiado próximo al negro, que es el más antiguo. En cambio, el filtro verde permanecerá vivo al no estar lo suficientemente cerca del negro.

Un problema muy distinto es resolver que un mismo filtro siga a una única persona cuando hay una multitud bastante grande como sería la siguiente secuencia:



**Fig. 4. 46.-** Imagen perteneciente a la secuencia E2\_7, donde se puede ver como la multitud se logra seguir con 4 filtros de partículas.

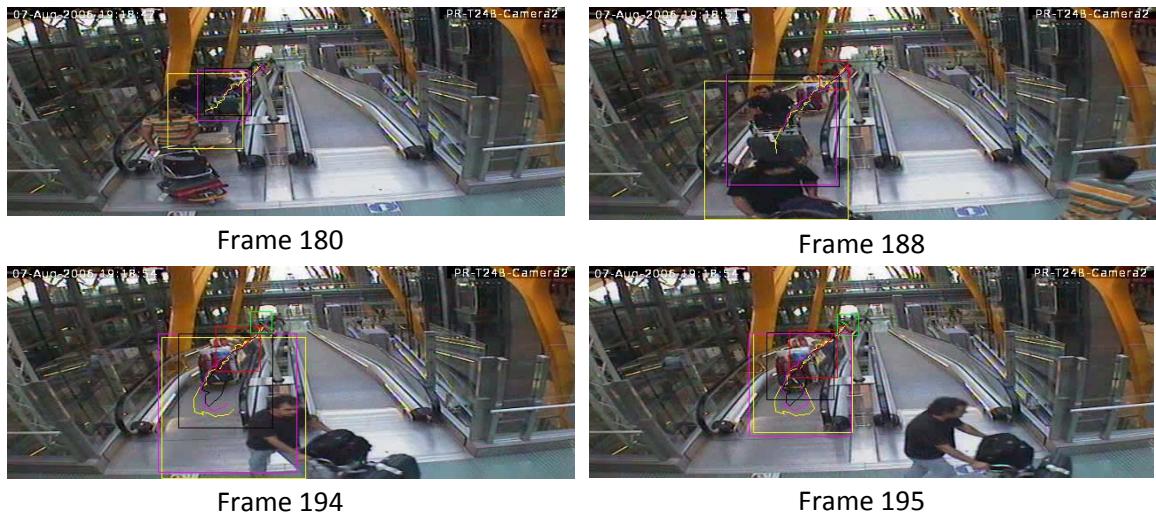
Este problema no se resolverá, ya que como se ha dicho en los objetivos del trabajo, nuestra preocupación es poder estimar el flujo de movimiento y no poder distinguir entre personas de una multitud.

## Retroceso

Existe una situación que tiene lugar repetidas veces, sobretodo en el escenario 2 debido a las características que presenta, que consiste en que el filtro de partículas vuelve hacia la zona de seguimiento una vez alcanzada la zona de destrucción. Este es el motivo por el que llamamos a este tipo de situaciones como retroceso.

Esto se debe a que cuando el filtro de partículas finaliza el seguimiento a lo largo de la rampa, si la persona que éste seguía desaparece de la imagen rápidamente, el filtro tiende a seguir a las personas que vienen por la rampa, de manera que vuelve hacia atrás. Este comportamiento

se incrementa en el caso del escenario 2 debido a que, por un lado, la imagen que nos ofrece la cámara dispone de menos espacio después de la finalización de la rampa para que pueda continuar el seguimiento y, por otro, el hecho de que la perspectiva en este escenario sea tan pronunciada provoca que el filtro de partículas tenga un tamaño considerable cuando llegue a la parte inferior de la rampa, lo que ocasiona que el filtro siga indirectamente a la gente que viene por la rampa, por lo que en el momento en el que el objetivo que seguía desaparece, el filtro empieza a seguir lo que indirectamente ya hacía. Veamos con un ejemplo esta situación:



**Fig. 4. 47.- Ejemplo que corresponde a la secuencia E2\_3, donde se observa como las trayectorias de los filtros vuelven hacia atrás tras realizar el seguimiento, un caso de una situación de retroceso.**

En la figura anterior, se observa claramente como varios filtros de partículas realizan el seguimiento correctamente de una persona con un carro durante varios fotogramas. Pero a partir del *frame* 194, los filtros que siguen a la persona con el carro, que se encuentra al final de la rampa, también siguen a la persona que viene por la rampa. En el fotograma siguiente, los filtros deciden seguir a la persona que se encuentra en la rampa, debido a que la persona que seguía ha realizado un gran desplazamiento hacia la derecha, lo que dificulta su seguimiento. Este comportamiento es totalmente indeseado y complica la estimación del flujo de movimiento que representa la gente moviéndose a través de la rampa. Por lo tanto, definiremos que ***cuando un filtro de partículas que llega a la zona de destrucción, decide volver hacia atrás, es decir, su centroide retrocede respecto al eje Y durante dos frames consecutivos, este será destruido.***

## 4.2.- Clasificación y detección de tumultos:

Llevar a cabo la detección de tumultos implica una previa definición de qué consideramos como un tumulto o una situación con tumulto. Es por ello que, en primer lugar, debemos definir detalladamente qué clasificaremos como un tumulto y, por lo tanto, es una situación con un peligro potencial. En segundo lugar, realizaremos un etiquetado general de todas las secuencias disponibles para extraer las que presenten este tipo de situaciones y realizar su análisis. En tercer lugar, definiremos los criterios empleados para su detección a partir de la información que obtenemos de los filtros de partículas mediante la información de salida que nos facilita el algoritmo implementado, en nuestro caso constará de un fichero de texto con la posición y velocidad de los filtros de partículas para cada fotograma. Finalmente, explicaremos las métricas empleadas para determinar la calidad del algoritmo implementado a nivel de detección y de falsas alarmas.

### 4.2.1.- Caracterización de un tumulto

¿Qué es un tumulto? O mejor dicho, ¿qué consideraremos como una situación con tumulto? La palabra tumulto significa confusión agitada o desorden ruidoso, es decir, **un alboroto producido por una multitud**. Veamos un ejemplo visual:



Fig. 4. 48.- Imagen que muestra lo que consideraremos una situación con tumulto.

Tal y como acabamos de ver, **un tumulto no es un parón puntual**, como el que tendría lugar si a una persona se le atasca la maleta. Un tumulto, es **una interrupción significativa en el flujo de personas**, donde se ven afectadas varias personas y además vienen más personas por la rampa que se pueden ver involucradas en éste, de modo involuntario, si no se toman medidas al respecto. Así pues, este tipo de situaciones son las que intentaremos detectar con la finalidad de generar algún tipo de alarma que pueda servir para advertir al encargado de seguridad que ésta situación está teniendo lugar. De este modo, será capaz de detener la rampa inmediatamente y así evitar los daños que puedan surgir debidos a que la rampa siga funcionando. Hay que tener presente, que si ya de por sí un tumulto es peligroso, el hecho de que la rampa siga funcionando cuando éste tiene lugar, provoca que la multitud de personas

que viene detrás se abalancen unas sobre las otras ocasionándose auténtico daño. Este es el motivo principal del interés existente en detectar este tipo de situaciones, no evitarlas, algo fuera de nuestro alcance, sino minimizar los daños causados por éstas.

#### **4.2.2.- Etiquetado manual de las secuencias**

Una vez definido con claridad lo que consideraremos un tumulto, nos queda analizar todas las secuencias de que disponemos para encontrar las que presenten un tumulto y así poder evaluar la calidad de detección del algoritmo implementado. En primer lugar, nos centraremos en localizar todas las secuencias con tumulto. En segundo lugar, como la cantidad de tumultos hallados en las secuencias disponibles es relativamente pequeña, y para evaluar la calidad del algoritmo es necesario un mayor espacio de muestras, seleccionaremos hasta 100 secuencias para poderlo determinar cuantitativamente. Estas 100 secuencias han sido etiquetadas manualmente de manera que cada secuencia con o sin tumulto nos permita evaluar distintas limitaciones y/o carencias del algoritmo. De este modo, al haber utilizado el algoritmo de seguimiento y detección en todas estas secuencias, hemos conseguido mejorar su robustez y fiabilidad.

A continuación mostramos la distribución de las 100 secuencias seleccionadas según a qué escenario corresponden y la cantidad de tumultos que éstas presentan:

ESCENARIO	E1	E2	E3	E4
Nº de secuencias	21	42	30	7
Nº de tumultos	0	9	0	0

**Tabla 4.5.- Información de la cantidad de secuencias de que disponemos por escenario y la cantidad de tumultos que éstas presentan.**

Tal y como se mencionó en el apartado 1.1, las secuencias del escenario 1, con la perspectiva claramente definida, han sido empleadas para realizar las primeras pruebas de seguimiento. Las secuencias del escenario 2, se han utilizado para determinar cuáles son los valores idóneos de los parámetros que nos permiten modificar la detección. Finalmente, el resto de secuencias sin tumultos, nos han servido para evaluar la cantidad de falsas alarmas que el algoritmo puede llegar a presentar. Cabe destacar, que el número de secuencias con tumulto es relativamente pequeño, por lo que el número de ejemplos resulta ser insuficiente para llevar a cabo un entrenamiento con el que poder generar un clasificador para la detección.

#### **4.2.3.- Métodos para la detección de tumultos**

A grandes rasgos, existen dos grandes grupos de criterios con los que podemos realizar la detección de tumultos. El primer grupo, conformado por los **criterios heurísticos**, que se basan

en la detección a partir de la observación de los datos obtenidos de los filtros de partículas, es decir, si observamos cambios significativos e inesperados en la posición o velocidad de una mayoría de los filtros, esto puede ser debido a un caso de tumulto. Por otro lado, el segundo grupo engloba los **criterios de clasificación**, que mediante el uso de las secuencias de que disponemos, realiza un aprendizaje para clasificarlas de manera que obtengamos un patrón para poder decidir si la secuencia con la que nos encontramos a continuación, y a partir de los datos que ésta nos aporte, es un caso con tumulto o no. Este criterio tiene un pequeño problema, depende demasiado de la cantidad de secuencias de que dispongamos que presenten un tumulto, lo que hace que el proceso de aprendizaje no pueda llegar a ser todo lo extenso y robusto que se desearía si no se dispone de una cantidad razonable.

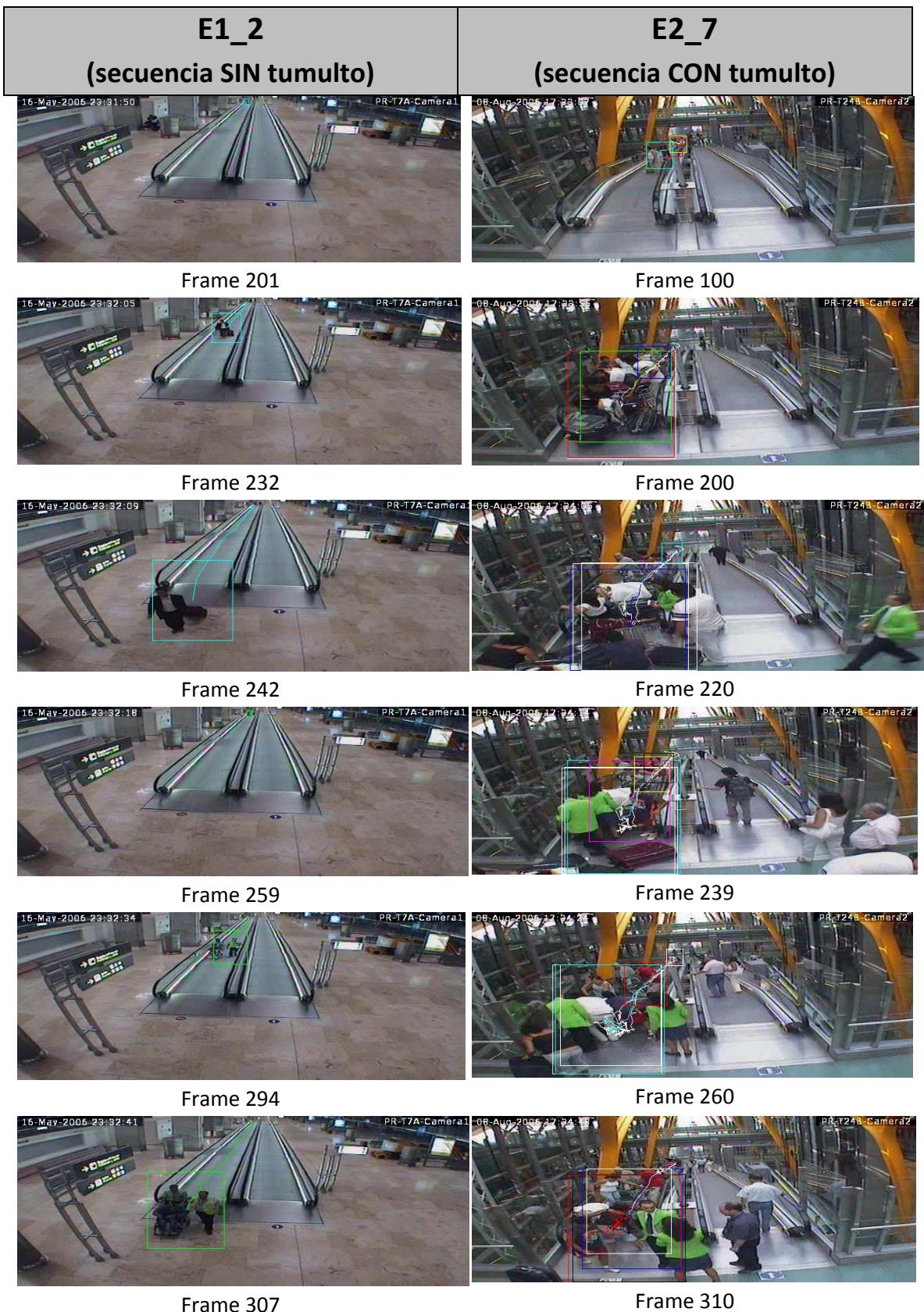
Tal y como se puede ver en el etiquetado de las secuencias mostrado en la tabla 4.1, definiremos los métodos que emplearemos para la detección de tumultos como criterios heurísticos, al no disponer de las suficientes secuencias con tumulto necesarias para poder realizar un criterio de clasificación. En los siguientes apartados expondremos dos métodos distintos que se han llevado a cabo para poder determinar si es posible la detección de tumultos con su empleo. Ambos métodos parten de la idea común de analizar el comportamiento de la velocidad de los filtros de partículas fotograma a fotograma, ya que, tal y como se vio en el apartado 1.3, el seguimiento implementado pretende **extraer el movimiento global o flujo de personas en una cinta transportadora**. De este modo, la mejor manera de determinar cuando el movimiento global o flujo se para, lo que equivaldría a una situación con tumulto, es mediante el análisis de este movimiento y, en consecuencia, de los filtros de partículas.

Con el primer método, pese a ser el más sencillo, no se logran obtener conclusiones que permitan observar con claridad una situación de tumulto, por lo que se vio la necesidad de implementar un método alternativo. A raíz de esto, se ha implementado el segundo método de detección con el que sí se logra distinguir una situación con tumulto, y resulta útil para realizar la detección. A continuación se explicarán ambos métodos.

### **Estudio de la velocidad de los filtros de partículas frame a frame:**

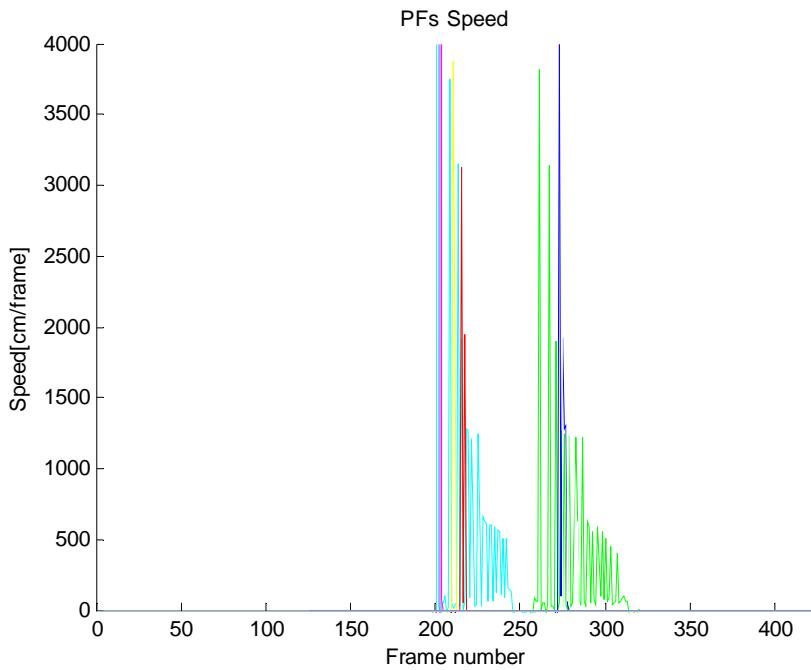
Este primer método, pretende detectar la situación de un tumulto mediante el análisis de la velocidad de los filtros de partículas fotograma a fotograma. Para llevar a cabo este análisis representaremos gráficamente la evolución de la velocidad de todos los filtros de partículas que se generan durante una secuencia para cada instante de tiempo. Pintaremos la velocidad de cada filtro con un color distinto, que irá asociado al color del filtro en la imagen, para distinguirlos con claridad.

Previamente a la representación gráfica, veamos las dos secuencias que estudiaremos, por un lado tenemos la secuencia E1\_2 (perteneciente al escenario 1), que corresponde a un seguimiento sencillo de dos grupos de personas que se mueven por la rampa en instantes distintos de tiempo. Por otro lado, estudiaremos la secuencia E2\_7 (que pertenece al escenario 2) que presenta un tumulto que empieza en el *frame* 200 y dura hasta el fotograma 420.



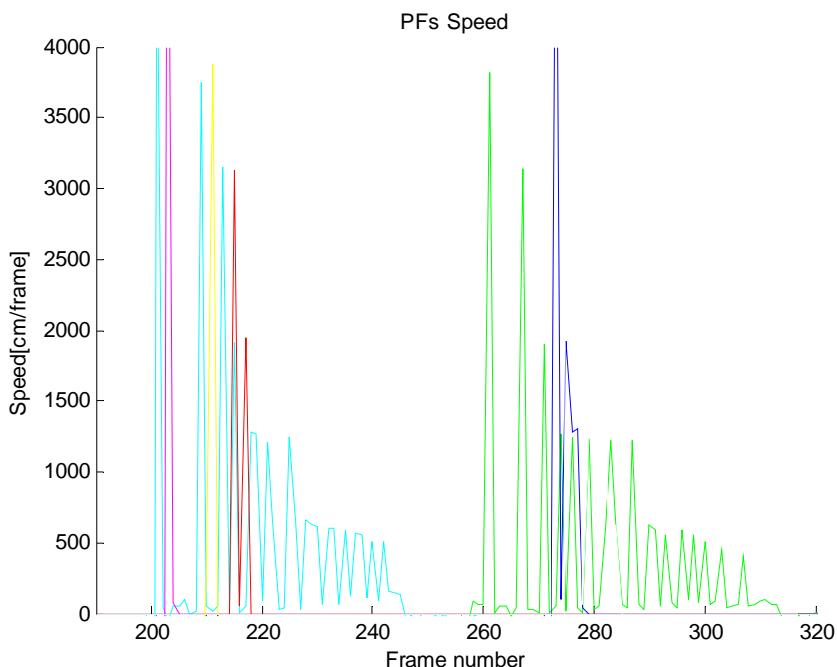
**Fig. 4. 49.- Imágenes del seguimiento con filtros de partículas de una secuencia sin tumulto y una con.**

En la siguiente figura podemos ver la representación de la velocidad de los filtros de partículas para cada *frame* en el caso de la secuencia E1\_2.



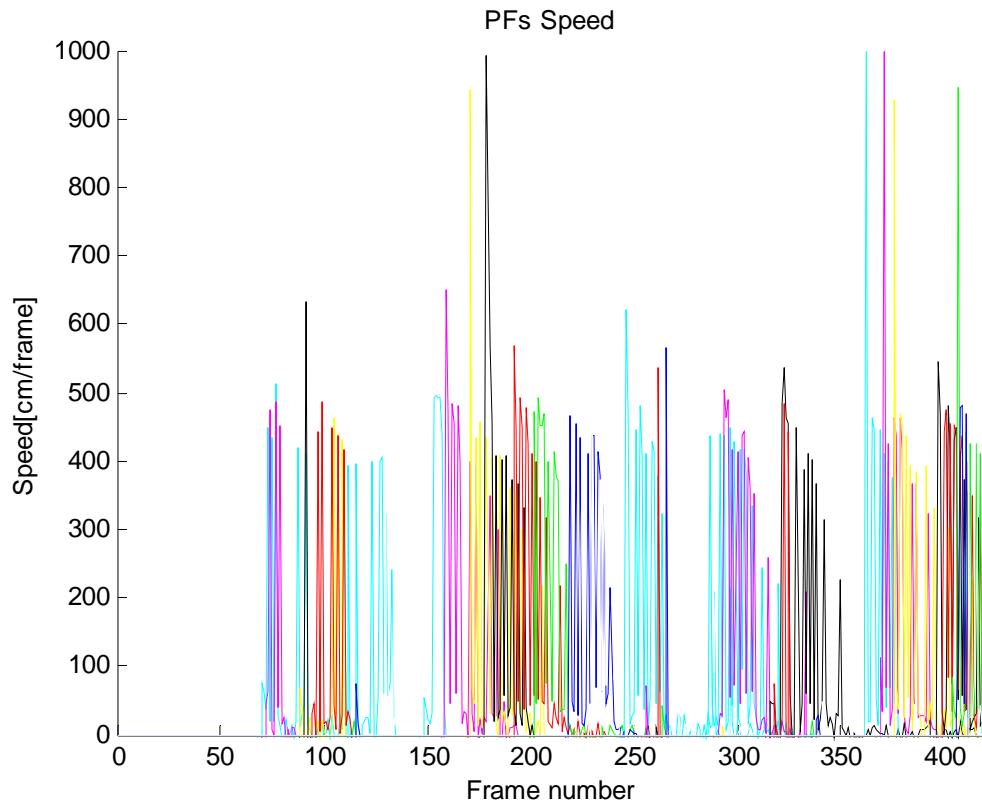
**Fig. 4. 50.- Representación de la velocidad de los filtros de partículas durante una secuencia sin tumulto, que consta de dos seguimientos independientes y sencillos.**

Para mejorar la claridad de la representación hemos aplicado un zoom a la zona donde tiene lugar el seguimiento. De este modo se puede apreciar con total claridad, en la siguiente figura, el seguimiento realizado por el filtro de color cian y el filtro de color verde, además de algunos filtros que se generan esporádicamente:



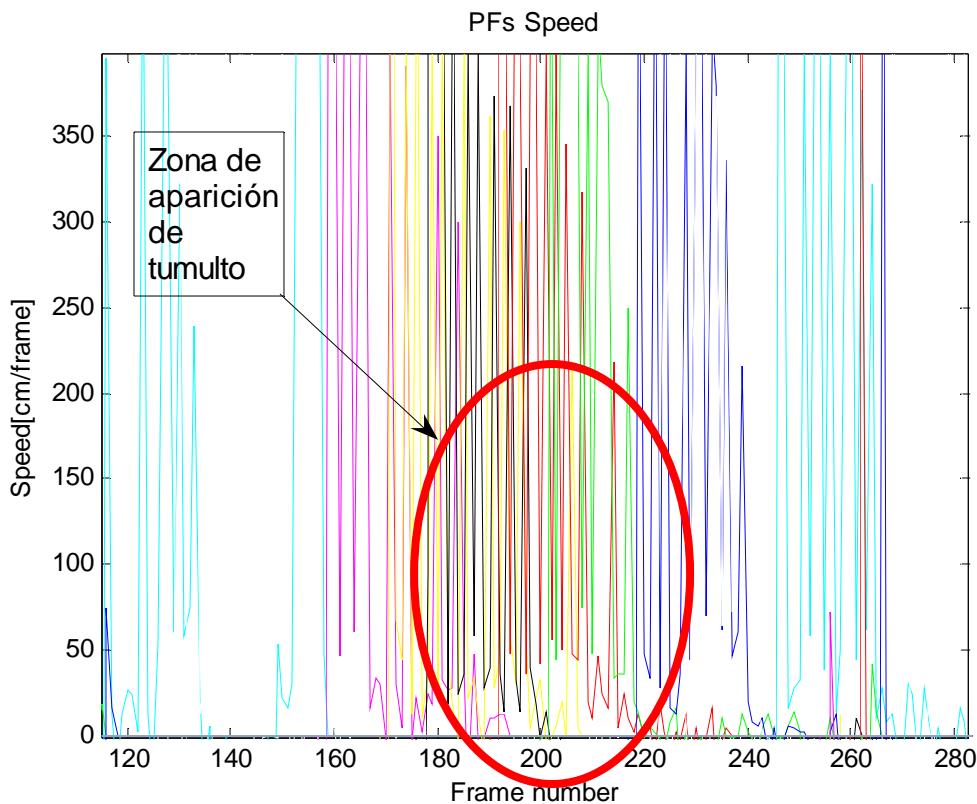
**Fig. 4. 51.- Zoom de la imagen anterior, en la que se puede apreciar con claridad los dos seguimientos realizados por los filtros de partículas.**

De momento, con el método actual, sí que se puede distinguir un seguimiento sencillo. Veamos ahora qué sucede si empleamos la misma representación para la secuencia E2\_7, bastante más compleja y con muchos más filtros de partículas:



**Fig. 4. 52.- Representación de la velocidad de los filtros de partículas durante una secuencia con tumulto.**

Tal y como cabía esperar, la representación es bastante más densa, por lo que se observan multitud de filtros de partículas de distintos colores sin determinar qué está sucediendo en cada momento. Lo que parece indicar que esta representación no nos subministra ningún tipo de información acerca de si hay o no tumulto. Para cerciorarnos de que este tipo de representación no nos aporta ningún tipo de información aparente para detectar un tumulto, ampliaremos la zona donde tiene lugar el tumulto de manera que obtendremos la siguiente representación:



**Fig. 4. 53.- Zoom de la imagen anterior destacando la zona de inicio de un tumulto.**

En conclusión, mediante esta representación no se logra distinguir un rasgo significativo que nos sugiera la existencia de un tumulto, es por ello que se ha pensado un criterio de detección alternativo, con el que sí podemos distinguir una situación con tumulto y con el que se logra realizar unas detecciones satisfactorias.



**Fig. 4. 54.- Imagen que muestra una situación de detección de un tumulto, nuestro objetivo final, con la correspondiente *alarma*, perteneciente a la secuencia E2\_7.**

## A partir de la aglomeración de centroides de los filtros en función de su posición y velocidad:

El segundo criterio implementado, parte con una idea similar al criterio anterior en lo que respecta a la monitorización de la velocidad. Pero, tal y como se ha visto, esto no es suficiente, por lo que es necesario añadir algún tipo de información adicional, concretamente la posición del filtro de partículas. Para ello definiremos un valor modificado de la posición del filtro según el eje Y, que consiste en proyectar la posición sobre la rampa de manera que obtengamos la posición real del filtro. De este modo logramos solventar el problema que nos puede causar la perspectiva provocando que parezca que dos filtros estén muy cerca uno del otro sin en realidad estarlo.

El cálculo de esta nueva Y, Y', se hará siguiendo una serie de pasos:

1. Invertiremos el valor de la Y, ya que nos interesa fijar el cero en la parte inferior de la imagen, lo que corresponde con el final de la rampa.
2. Multiplicaremos el valor de la Y por un factor de escalado K, en m/píxel, que vendrá dado por el cociente entre el tamaño vertical del filtro en esta posición, según las tablas mostradas en el anexo 3, y el tamaño genérico o altura estándar, de una persona que consideraremos de 1.8 m.

$$K = \frac{A_k^i}{\text{Altura estándar}} \quad (4.1)$$

donde  $A_k^i$  es la altura del filtro de partículas  $i$  en el instante  $k$ .

3. Calcularemos el ángulo  $\beta$  (ver Fig. 4. 55.- Explicación gráfica de cuáles son los ángulos  $\alpha$  y  $\beta$ . La idea es la misma en el resto de escenarios.) a partir del cual proyectaremos, mediante la ecuación (4.2), nuestra coordenada Y sobre el eje que representa la recta que modela la rampa para obtener el valor real de la posición del filtro sobre la rampa. El valor del ángulo  $\beta$  es sencillo de calcular conocido el ángulo  $\alpha$ , fácilmente calculable a partir de la ecuación de la recta de la rampa vista en el apartado 3.2.3.

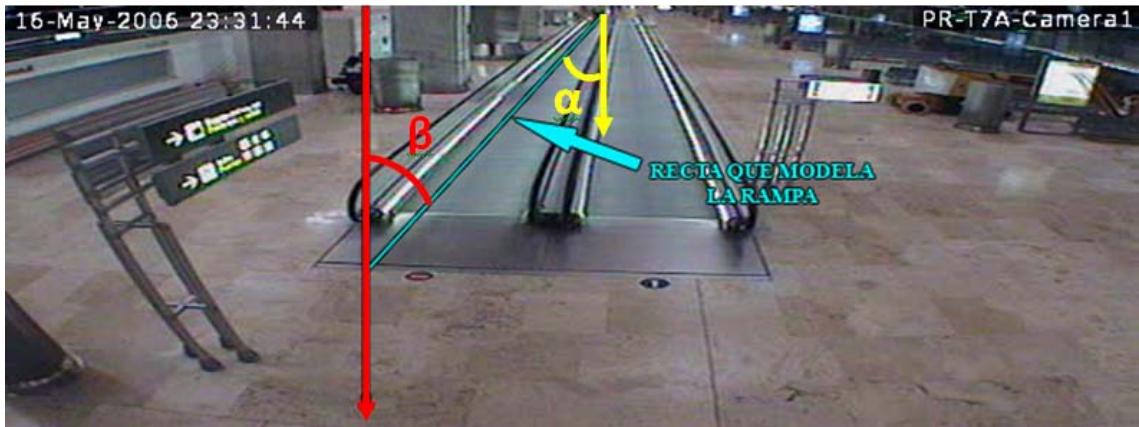
$$Y' = \frac{Y}{\cos\beta} \quad (4.2)$$

La velocidad del filtro de partículas se calculará a partir de la posición  $Y'$  con la siguiente ecuación:

$$V_t = \frac{Y'_t - Y'_{t-1}}{\Delta t} \quad (4.3)$$

donde el  $\Delta t = 2$  frames/s que es la velocidad de fotograma o *frame rate* que corresponde a las secuencias con las que trabajamos. La velocidad será expresada en m/s.

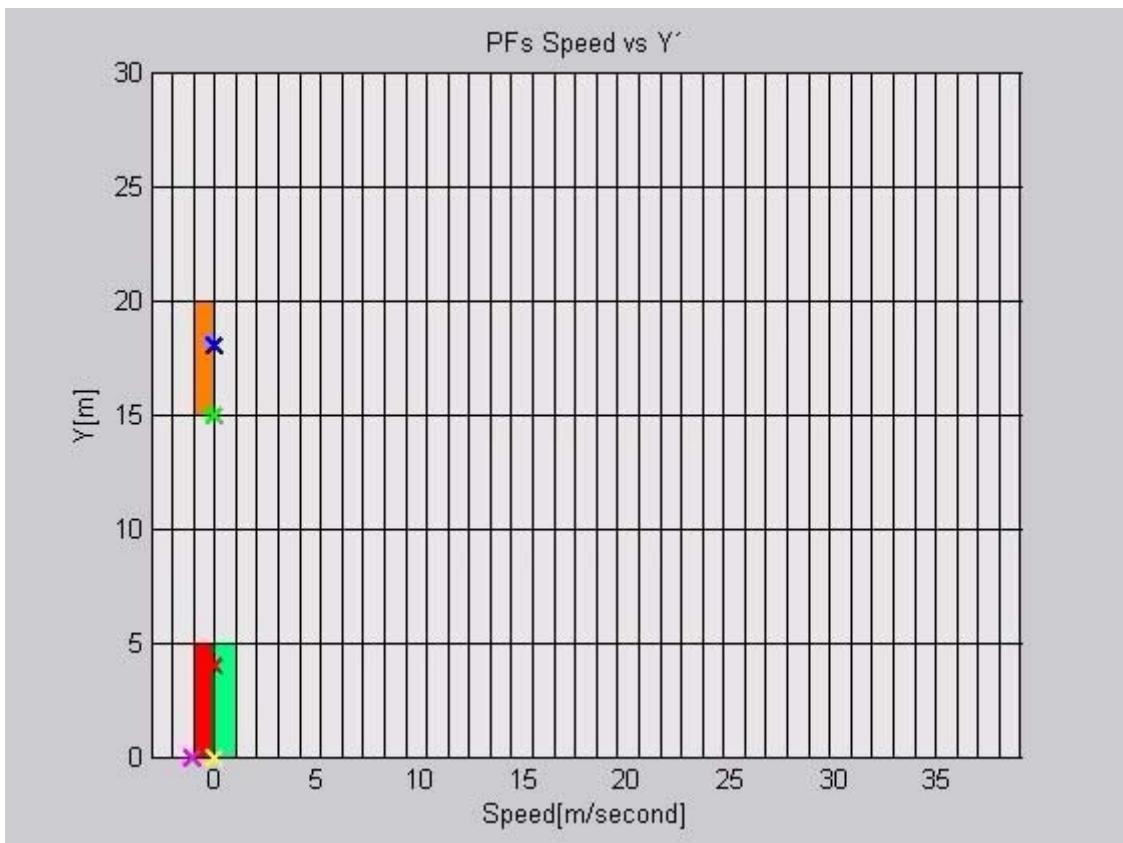
La idea principal en la que se basa este criterio consiste, básicamente, en representar en el eje de ordenadas la posición del filtro, el eje Y', tomando **el cero absoluto en la parte inferior o final de la rampa descendente**, y en el eje de abscisas situaremos el valor de la velocidad del filtro de partículas que podrá ser tanto positivo como negativo.



**Fig. 4. 55.- Explicación gráfica de cuáles son los ángulos  $\alpha$  y  $\beta$ . La idea es la misma en el resto de escenarios.**

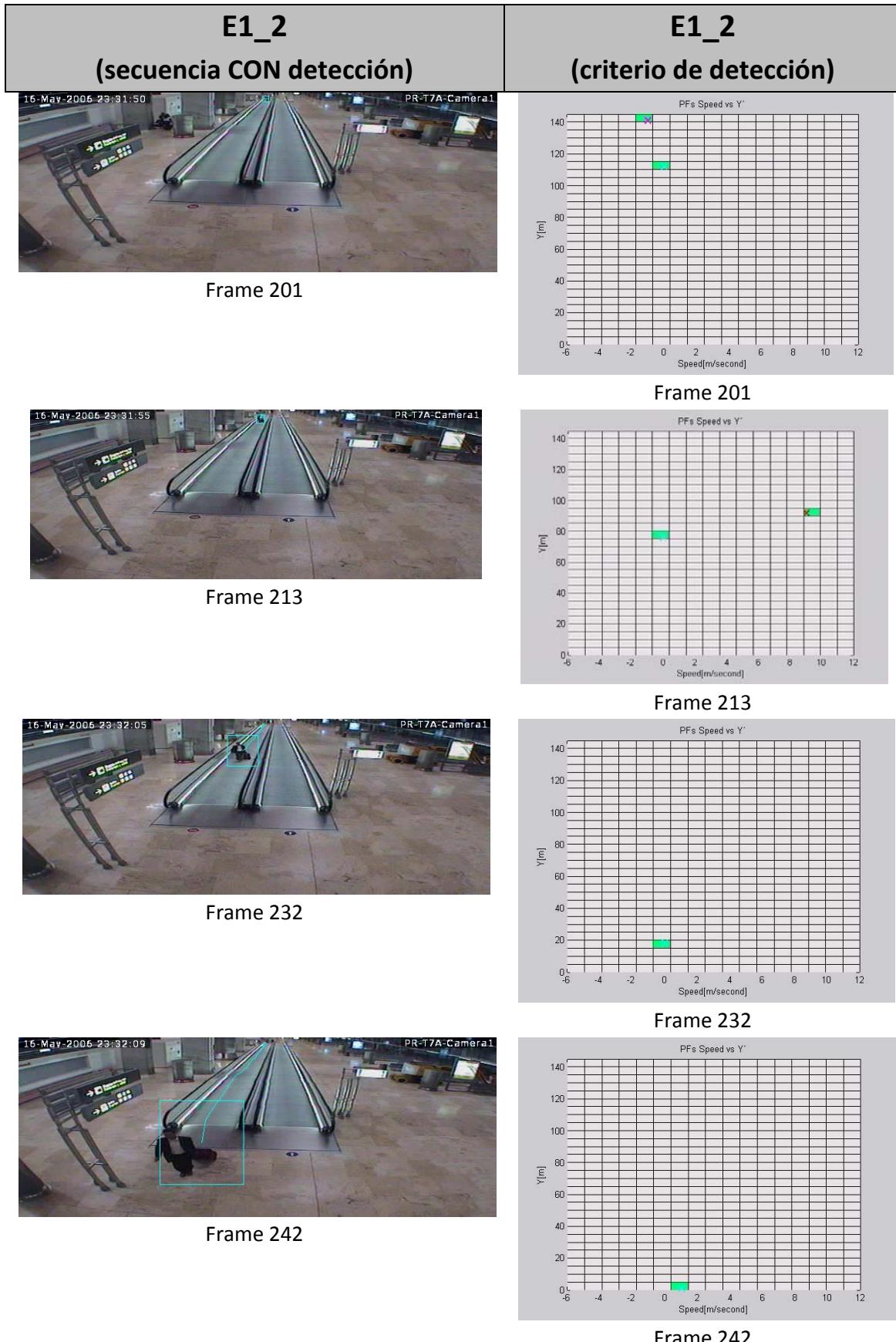
La representación de los filtros vendrá dada por una X que indicará su posición y velocidad frame a frame, es decir, conforme el filtro vaya avanzando hacia el final de la rampa y variando su velocidad la X se irá moviendo según corresponda. El principal objetivo de esta representación es **determinar la concentración de los filtros a nivel espacial y de velocidad**, de manera que dividiremos la representación en celdas o *bins* de resoluciones variables. De este modo, generaremos una especie de rejilla o *grid* encima de la cual dibujaremos los filtros de partículas, tal y como se ha explicado. Emplearemos 6 resoluciones distintas para el tamaño de las celdas del *grid*: 500x100, 500x50, 500x25, 250x100, 250x50, 250x25 y 100x50 (posición o Y' [cm] x velocidad o V [cm/s]). El motivo por el cual se ha empleado distintas resoluciones se debe a que se trata de un parámetro que modifica en gran medida los resultados obtenidos en la detección. En el apartado 5.1 se detallan los resultados conseguidos con cada resolución y la que se ha elegido en función de la calidad y de las características de la detección.

**El criterio de detección** que se emplea consiste en *contar la acumulación de filtros de partículas en cada celda, de modo que si sólo hay 1, no es un problema ya que corresponde a un seguimiento normal, si hay 2, se puede deber a que una persona se acerca a otra debido a sus distintas velocidades, pero si se empiezan a acumular 3 o más en una misma celda, puede deberse a una situación de tumulto*. A nivel gráfico, pintaremos las celdas de distintos colores según sea la cantidad de filtros de partículas que éstas contengan a modo de **indicador de peligrosidad**: verde corresponde a 1 filtro, naranja a 2 filtros y rojo a 3 o más filtros. De este modo, diremos que se ha detectado tumulto cuando se dé la situación de que en dos *frames* consecutivos, para reducir las falsas alarmas, 3 o más filtros de partículas se encuentren en una misma celda. Observemos un ejemplo de un seguimiento de múltiples personas donde se pueden apreciar claramente los 3 niveles de peligrosidad:

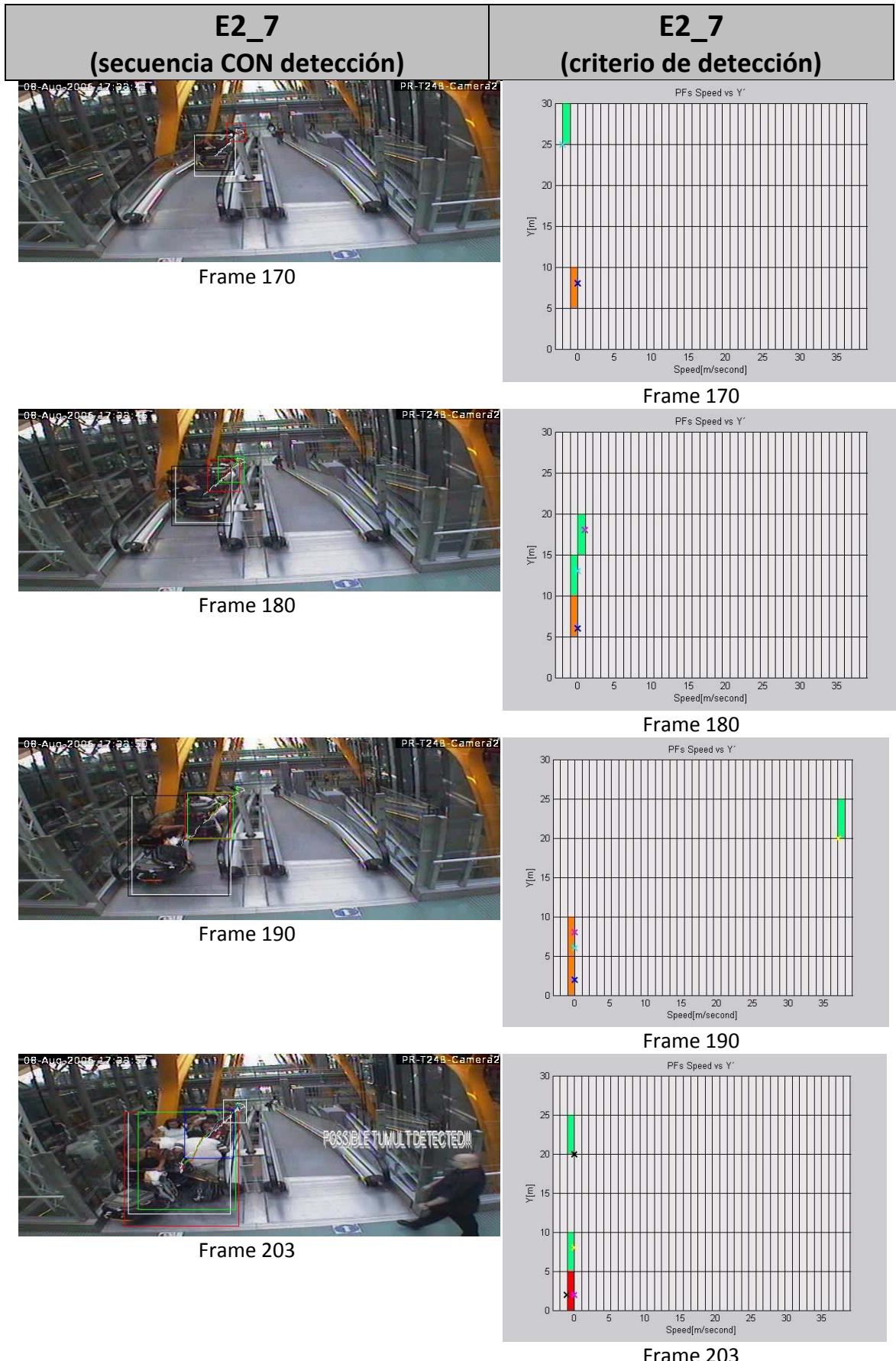


**Fig. 4. 56.- Ejemplo de detección, perteneciente a la secuencia E2\_7 y realizada con una resolución de 500x100, donde se aprecian claramente los 3 estados de colores que pueden presentar las celdas. Color verde -> 1 filtro (no hay peligro), color naranja -> 2 filtros (possible peligro) y color rojo -> 3 filtros (possible tumulto).**

Por lo tanto, ya queda explicado cuál es el criterio que emplearemos para detectar los tumultos a partir de la información que nos facilitan los filtros de partículas. El siguiente paso es ver cómo funciona realmente el criterio implementado. Para ello, analizaremos los resultados que se obtienen al aplicar este criterio en dos secuencias distintas. En primer lugar, comprobaremos como transcurre la relación entre el seguimiento y la representación con la que detectaremos el tumulto con un caso sencillo como es el que se puede ver en la Fig. 4. 57.- Ejemplo, correspondiente a la secuencia E1\_2, de un seguimiento sencillo y la detección realizada. Tal y como se observa, al haber un único objetivo, tan solo se genera un filtro de partículas y el estado de las celdas es siempre verde, es decir, sin peligro.. A continuación, veremos como la situación se complica si analizamos la detección de un tumulto en una secuencia más compleja. Finalmente, tan solo queda por determinar la calidad del criterio empleado a partir de las métricas que se explican en el siguiente apartado y del estudio de un mayor número de casos para verificar la calidad del algoritmo desarrollado.



**Fig. 4. 57.- Ejemplo, correspondiente a la secuencia E1\_2, de un seguimiento sencillo y la detección realizada. Tal y como se observa, al haber un único objetivo, tan solo se genera un filtro de partículas y el estado de las celdas es siempre verde, es decir, sin peligro.**



**Fig. 4. 58.- Ejemplo, correspondiente a la secuencia E2\_7, de un seguimiento complejo y con tumulto. Se observa, como se realiza correctamente la detección del tumulto con una resolución de 500x100.**

#### 4.2.4.- Métricas empleadas para determinar la calidad de la detección de tumultos

##### Definición del Ground Truth:

Previamente a la definición de las métricas empleadas en la detección de tumultos es necesario etiquetar manualmente lo que es el *Ground Truth*. El *Ground Truth* es una manera de etiquetar internamente una secuencia a partir de un criterio, es decir, en el caso que nos atañe, generaremos una serie de valores, para cada *frame* de la secuencia, que nos indicarán si el fotograma en cuestión presenta o no un tumulto. Por ejemplo, veamos gráficamente el *Ground Truth* de dos secuencias con tumulto, la E2\_7 y la E2\_14 para entender mejor qué representa:

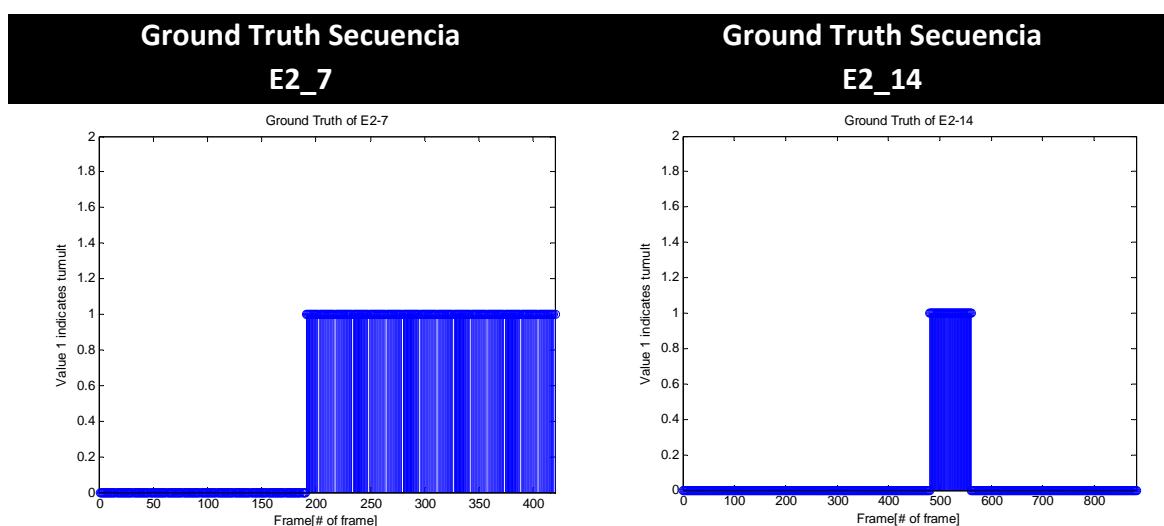


Fig. 4. 59.- *Ground Truth* de las secuencias E2\_7 y E2\_14, respectivamente.

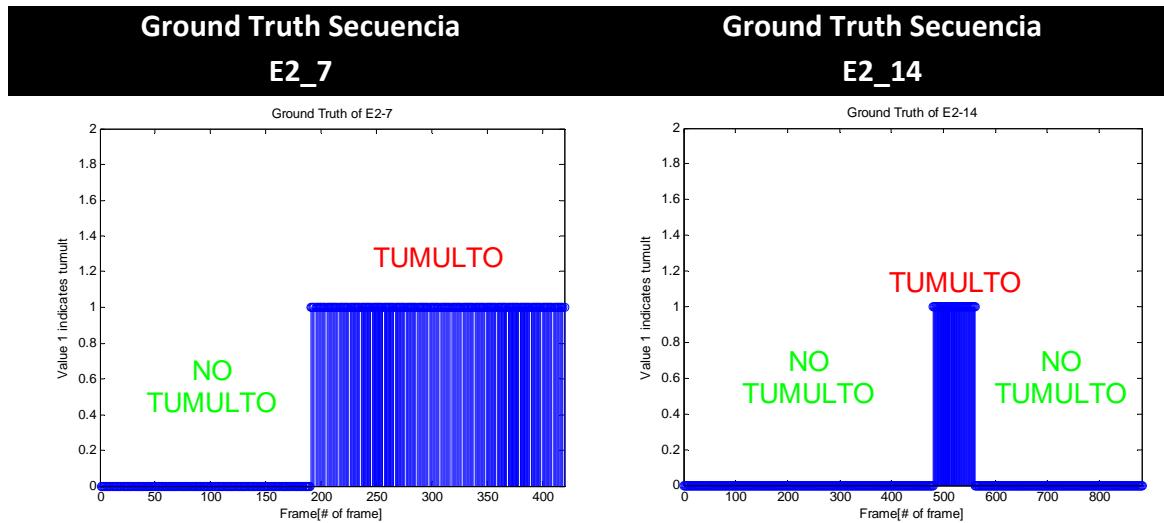
La zona azul de amplitud 1 indica que durante estos fotogramas se está produciendo una situación de tumulto. Para la secuencia E2\_7, el tumulto empieza en el *frame* 190 y finaliza en el *frame* 419, mientras que para la secuencia E2\_14, éste empieza en el *frame* 480 y acaba en el 560, es un tumulto bastante rápido. Los criterios que se han tomado para decidir cuándo empieza y cuándo acaba una situación con tumulto son los siguientes:

- **Inicio:** Un tumulto empieza cuando se observa el primer fotograma en el que se distingue un claro alboroto producido por una multitud, tal y como se muestra en la figura Fig. 4. 48.- Imagen que muestra lo que consideraremos una situación con tumulto..
- **Fin:** Un tumulto finaliza en el momento en que el fotograma que se está observando no se aprecia evidencia alguna de que una situación fuera de lo común ha tenido lugar, es decir, no se aprecia resto alguno del tumulto.

Hay que tener presente que no tiene sentido hablar de *Ground Truth*, tal y como lo hemos definido, para una secuencia sin tumulto, ya que en un caso así, todos los frames son iguales y de valor 0.

### Detección por zonas (NT - T - NT):

Una vez definido el *Ground Truth* de las secuencias con tumulto, pasaremos a dividir las secuencias en dos o tres intervalos según corresponda. Dividiremos las secuencias en intervalos según el intervalo presente tumulto (T) o no tumulto (NT). De modo que si aplicamos esta segmentación a los *Ground Truth's* de las secuencias anteriores obtenemos lo siguiente:



**Fig. 4. 60.-** Ground Truth de las secuencias E2\_7 y E2\_14 respectivamente.

En nuestro caso, todas las secuencias con las que hemos trabajado sólo presentan 1 tumulto, por lo que este tipo de segmentación tan solo puede tener un intervalo (no hay tumulto), dos (no hay tumulto + hay tumulto) o tres intervalos (no hay tumulto + hay tumulto + no hay tumulto).

### Parámetros que determinarán la calidad del seguimiento:

Los parámetros que emplearemos para determinar la calidad del seguimiento son dos: la tasa de detección y la tasa de falsas alarmas. Existen varias maneras de definirlos, normalmente su definición se realiza *frame a frame*, pero en nuestro caso y teniendo en cuenta las condiciones de seguimiento y detección, se han definido del modo que se explica a continuación.

#### Tasa de detección:

En primer lugar, definiremos la tasa de detección individual de cada secuencia. Éste valdrá 1 si existe alguna detección durante el intervalo etiquetado como tumulto, independientemente

de cuándo éste se produzca, ya sea al inicio, en medio o al final del tumulto. Por lo tanto, nuestra detección será del 100% si detectamos tumulto en un solo *frame*. Hay que recordar que el algoritmo de detección lleva implementada una histéresis, tal y como se explica en el apartado 4.2.3, por lo que la detección de un tumulto implica, implícitamente, que esta situación se dé en dos *frames* consecutivos.

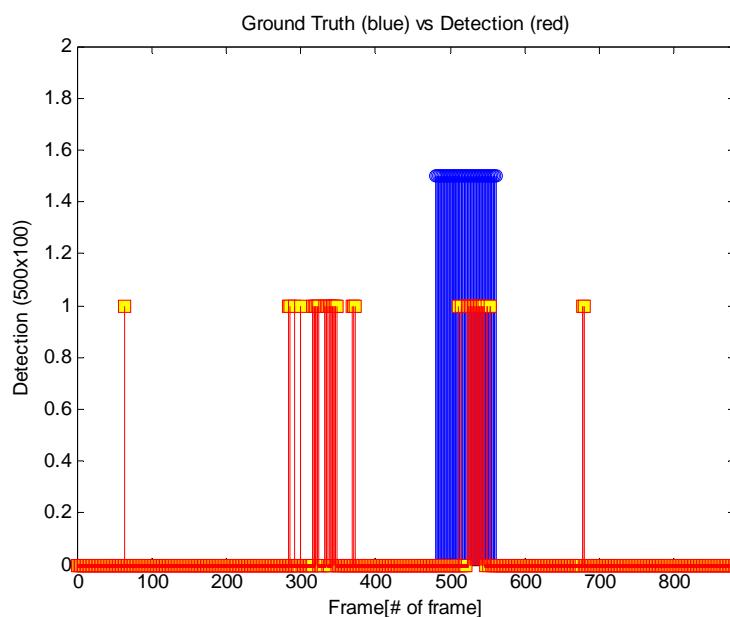
En segundo lugar, definiremos una tasa de detección global para las 9 secuencias que presentan tumulto. Este valor vendrá dado por la siguiente ecuación:

$$\text{Global Detection Rate} = \frac{\# \text{ Tumults Detected}}{\# \text{ of Tumults}} \times 100 [\%] \quad (4.4)$$

Donde el valor del tumulto detectado valdrá 0 o 1 según sea el valor de la tasa de detección particular de cada secuencia.

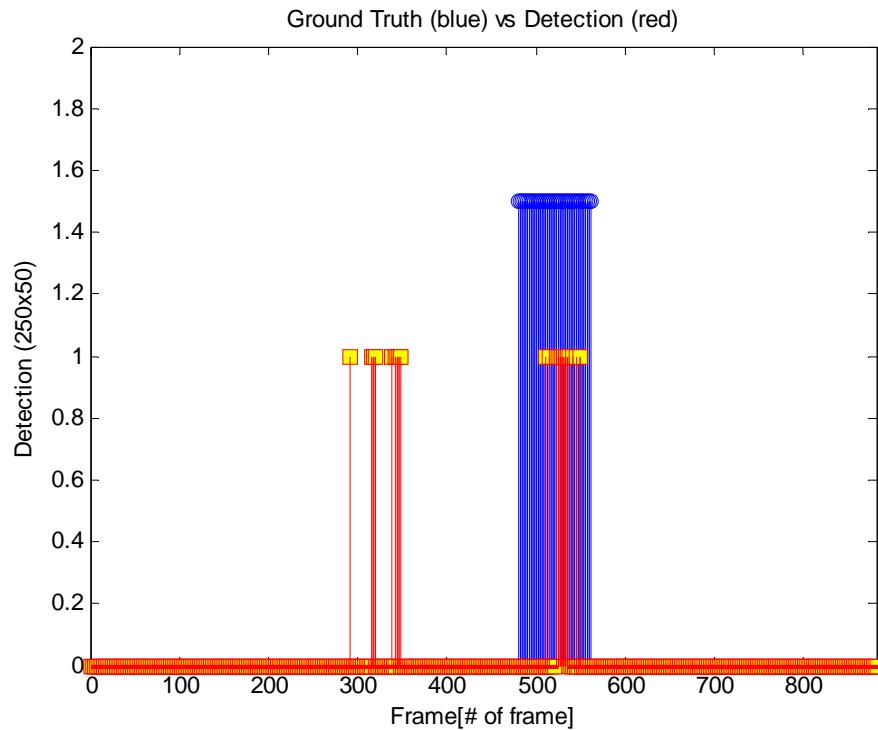
#### **Tasa de falsas alarmas:**

Del mismo modo que en el caso anterior, definiremos dos tasas de falsas alarmas, uno para el caso individual y uno para el caso global. En el caso individual, consideraremos que la secuencia presenta un falso positivo o falsa alarma si durante un intervalo de no tumulto se ha detectado uno. De modo que, tal y como sucedía antes, una única falsa detección provoca que tengamos un 100% de falsos positivos. Hay que tener en cuenta, que tal y como hemos definido esta tasa, si una secuencia tiene tres intervalos, como la secuencia E2\_14, y en los dos donde no hay tumulto, presenta dos falsas detecciones, una en cada intervalo, la tasa de falsas alarmas para esta secuencia sería de 2, o lo que es lo mismo, del 200%. Veamos como esto sucede en la detección de la secuencia E2\_14 para una resolución de 500x100, donde la detección se representa con el color rojo y el *Ground Truth* con el color azul:



**Fig. 4. 61.- La detección presentaría una tasa de detección del 100% y de falsas alarmas del 200%.**

Si observamos la detección que se obtiene de la misma secuencia, E2\_14, con una resolución de 250x50 obtenemos los siguientes resultados:



**Fig. 4.62.- En este caso, la detección se mantiene en una tasa de detección del 100%, mientras que el de falsas alarmas disminuye al 100%.**

En cuanto al cálculo de la tasa de falsas alarmas global, éste se calcula de la siguiente manera:

$$\text{Global False Positives Rate} = \frac{\# \text{ False Positives Detected}}{\# \text{ of Tumults}} \times 100 [\%] \quad (4.5)$$

Donde el número de falsas alarmas corresponde a la tasa individual de falsas alarmas de cada secuencia. Cabe destacar que esta tasa no va de 0 a 1, es decir, que el número de falsas alarmas que podemos tener es superior al de las secuencias que tenemos. Esto se debe al criterio escogido para implementar la tasa de falsas alarmas.

En conclusión, la calidad de estos dos parámetros vendrá claramente alterada por el valor de la resolución de las celdas que se utilice durante la detección.

## 5.- Resultados y evaluación

En este apartado expondremos los resultados obtenidos mediante el criterio definido en el apartado 4.2.4. A partir de estos resultados calcularemos el punto de trabajo óptimo, de este modo decidiremos que parámetros optimizan la detección de tumultos.

### 5.1.- Resultados obtenidos con las distintas métricas empleadas:

En primer lugar, explicaremos algunas de las ventajas e inconvenientes de trabajar con una resolución de las celdas grande o pequeña. Tal y como se ha explicado en el apartado 4.2.4, cuanto mayor sea la resolución implementada mayor serán tanto el número de detecciones como el número de falsas alarmas detectado. Ahora bien, está claro que lo que nos interesa es reducir al máximo el número de falsas alarmas sin que se vea afectado en exceso el número de detecciones. Sin embargo, hay que tener presente otro tipo de problema, **cuándo se realiza la detección**. El principal objetivo de nuestro algoritmo es detectar tumultos para poder evitar o reducir en mayor medida una posible situación de peligro producida a raíz de éstos. Por lo tanto, lo ideal sería realizar la detección lo antes posible, es decir, en el mismo momento en el que éste tenga lugar. Veamos pues de modo gráfico qué sucede si modificamos la resolución y cómo esto afecta a la detección:

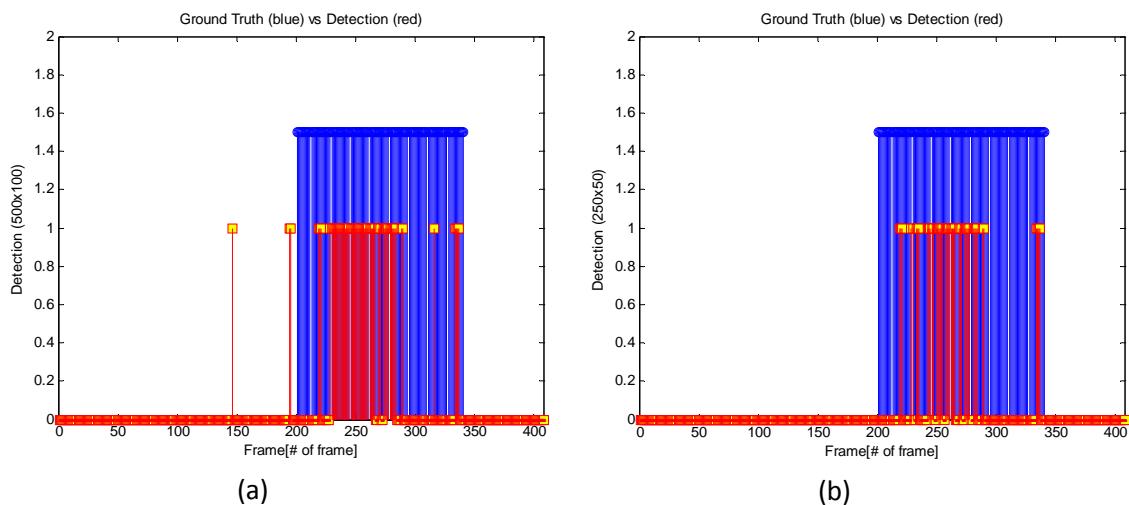
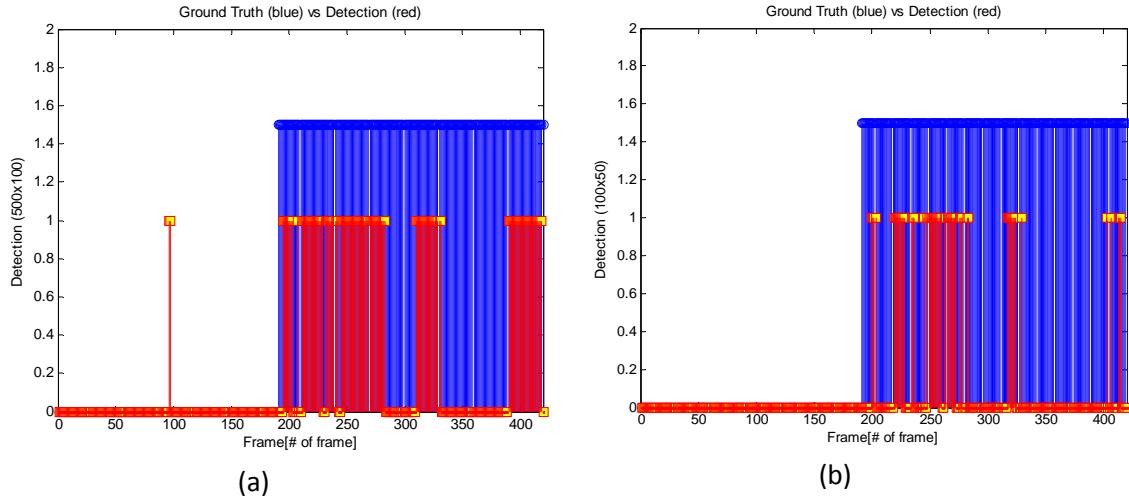


Fig. 5. 63.- Imágenes que muestran la detección (en rojo) frente al Ground Truth (en azul) para la secuencia E2\_3. (a) La resolución empleada de 500x100. (b) La resolución empleada es de 250x50.

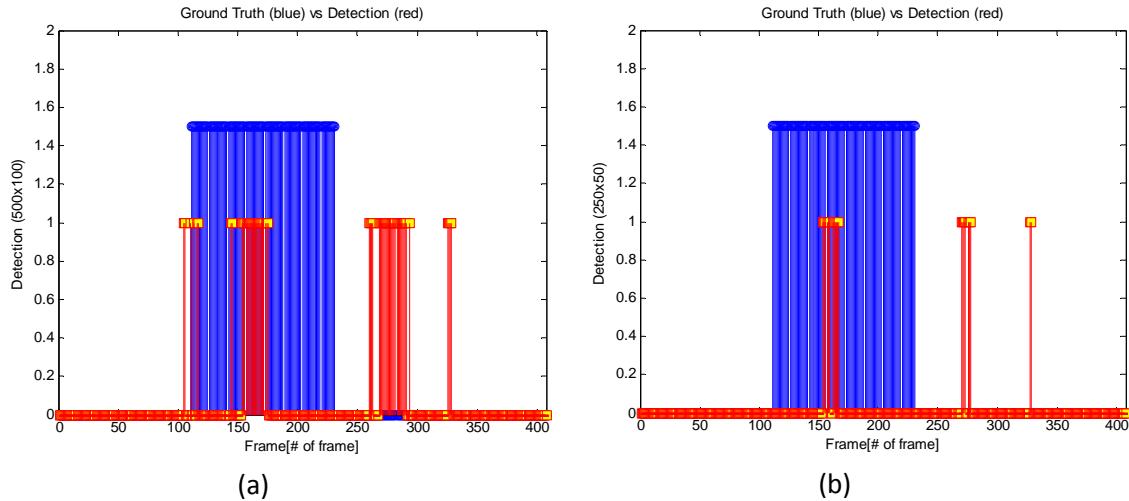
En este caso, prácticamente no se aprecia retardo alguno entre las dos detecciones, por lo que el resultado es mejor con una resolución de 250x50 que de 500x100 al eliminar todas las falsas alarmas.

Si realizamos el mismo estudio, para otra secuencia, comparando los casos extremos de la resolución, 500x100 y 100x50, se observa como tampoco nos afecta excesivamente:



**Fig. 5. 64.- Imágenes que muestran la detección (en rojo) frente al Ground Truth (en azul) para la secuencia E2\_7. (a) La resolución empleada de 500x100. (b) La resolución empleada es de 100x50.**

Pero no todo es tan sencillo, no siempre que reduzcamos el tamaño de las celdas obtendremos mejores resultados. Tal y como se puede ver en la secuencia E2\_18, reducir el tamaño del *bin* implica retrasar bastante la detección además de eliminar una falsa alarma:



**Fig. 5. 65.- Imágenes que muestran la detección (en rojo) frente al Ground Truth (en azul) para la secuencia E2\_18. (a) La resolución empleada de 500x100. (b) La resolución empleada es de 250x50.**

En resumen, la elección adecuada de la resolución no es algo trivial ni sencillo, ya que hay que tener en cuenta muchos factores y los resultados pueden variar en gran medida. Por este motivo hemos implementado un criterio, tal y como se ve en el apartado 5.1.3 para ayudarnos a decidir, teniendo en cuenta el resto de factores, cuál es la mejor resolución que se debe emplear en la detección de tumultos.

Otro aspecto a tener en cuenta, es que tal y como se ha llevado a cabo a la implementación del algoritmo, los primeros 50 fotogramas corresponden al periodo de *training* necesario para

realizar la extracción de *foreground*, esto está explicado en el apartado 3.1.1, por lo que si el tumulto tiene lugar en el *frame* 60, el algoritmo no tiene tiempo de inicializarse adecuadamente, y en consecuencia, de realizar correctamente la detección, por lo que éste es un factor a tener en cuenta. Lo ideal sería disponer de por lo menos 100 *frames* antes del tumulto para realizar la detección correctamente. Hay que tener presente que esta situación no se daría en un caso real ya que el algoritmo estaría funcionando en todo momento, únicamente se debería tener en cuenta para su inicialización.

### **5.1.1.- Tasa de detección:**

Inicialmente mostraremos los resultados de la tasa de detección para las 9 secuencias que hemos etiquetado como *tumulto* y para cada una de las distintas resoluciones empleadas para la detección.

	DETECTION RATE					
	Resolución de los bins (Y [cm] x Speed [cm/s])					
	500X100	500X50	250X100	250X50	250X25	100X50
<b>E2_2</b>	0%	0%	0%	0%	0%	0%
<b>E2_3</b>	100%	100%	100%	100%	100%	100%
<b>E2_7</b>	100%	100%	100%	100%	100%	100%
<b>E2_12</b>	100%	100%	100%	100%	0%	0%
<b>E2_14</b>	100%	100%	100%	100%	100%	100%
<b>E2_16</b>	100%	100%	0%	0%	0%	0%
<b>E2_18</b>	100%	100%	100%	100%	100%	100%
<b>E2_19</b>	100%	100%	100%	100%	0%	0%
<b>E2_20</b>	0%	0%	0%	0%	0%	0%

**Tabla 5.6.- Resultados individuales de la tasa de detección según las distintas resoluciones de detección y para las 9 secuencias con tumulto.**

Resumiendo, detectamos 4 tumultos para todas las resoluciones implementadas, el problema es que hay 3 secuencias que son detectadas dependiendo de la resolución empleada. Ahora bien, desde un punto de vista práctico, la manera de mostrar estos resultados no resulta del todo intuitiva para apreciar con claridad cuál es la resolución con la que obtenemos mejores resultados. Por ello, es necesario mostrar los resultados de manera global para cada una de las resoluciones de detección del modo siguiente:

DETECTION RATE (NT-T-NT)			
Resolución de los bins (Y [cm] x Speed [cm/s])			
GLOBAL	500	250	100
100	78%	67%	
50	78%	67%	44%
25		44%	

Tabla 5.7.- Resultados globales de la tasa de detección para las distintas resoluciones de detección.

Para decidir qué resolución es mejor es necesario observar la tasa de falsas alarmas que le corresponde ya que la detección más idónea vendrá determinada por un equilibrio entre ambos parámetros.

### 5.1.2.- Tasa de falsas alarmas:

Tal y como hemos hecho con la tasa de detección, en primer lugar, expondremos los resultados de la tasa de falsas alarmas para las 9 secuencias que hemos etiquetado como *tumulto* y para cada una de las distintas resoluciones empleadas para la detección.

FALSE POSITIVES RATE					
Resolución de los bins (Y [cm] x Speed [cm/s])					
	500X100	500X50	250X100	250X50	250X25
E2_2	100%	0%	100%	0%	0%
E2_3	100%	0%	0%	0%	0%
E2_7	100%	0%	100%	0%	0%
E2_12	100%	100%	100%	0%	0%
E2_14	200%	200%	100%	100%	100%
E2_16	100%	100%	100%	0%	0%
E2_18	200%	200%	100%	100%	0%
E2_19	100%	100%	0%	0%	0%
E2_20	0%	0%	0%	0%	0%

Tabla 5.8.- Resultados individuales de la tasa de falsas alarmas según las distintas resoluciones de detección y para las 9 secuencias con tumulto.

En segundo lugar, expresaremos los resultados anteriores de manera global para cada una de las resoluciones de detección, con el fin de que sean más sencillos y claros de entender, del modo siguiente:

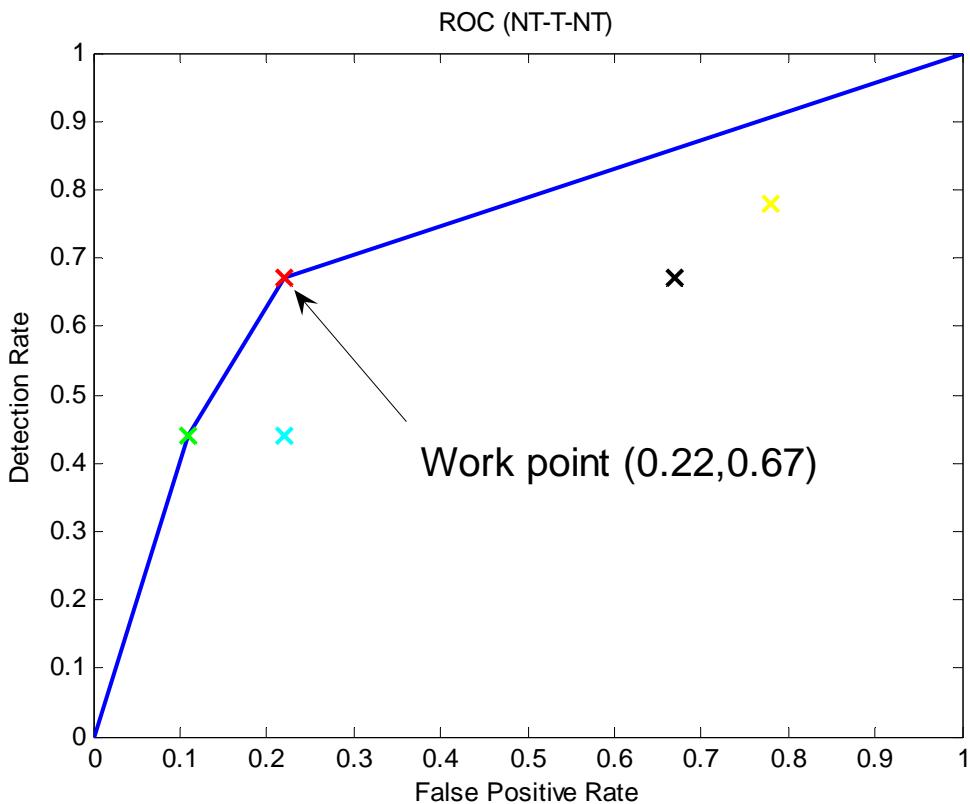
FALSE POSITIVES RATE (NT-T-NT)			
	Resolución de los bins (Y [cm] x Speed [cm/s])		
GLOBAL	500	250	100
100	111%	67%	
50	78%	22%	22%
25		11%	

**Tabla 5.9.-Resultados de la tasa de falsas alarmas para las distintas resoluciones de detección.**

En conclusión, las detecciones con las resoluciones más grandes (500x100, 500x50 y 250x100), presentan una tasa de falsas alarmas muy elevado, superior al 50%, lo que no es aceptable debido a que no nos interesa que nuestro algoritmo detecte erróneamente un tumulto una de cada dos veces. Sin embargo, con el resto de resoluciones, se aprecia una disminución considerable de las falsas alarmas, llegando a valores del 11%. A primera vista, parece razonable que la resolución que debamos emplear sea una de las 3 últimas. Para decidirnos cuál de éstas es mejor, una buena manera de hacerlo es mediante la representación gráfica del ROC, la cual nos permitirá determinar el punto de trabajo ideal para nuestro algoritmo.

### **5.1.3.- Cálculo del punto de trabajo óptimo a partir del ROC:**

Con los valores obtenidos en las tasas de los apartados 5.1.1 y 5.1.2 generaremos el ROC (*Receiver Operator Characteristic*) [28] con el que decidiremos el punto de trabajo. Este criterio es comúnmente utilizado para presentar los resultados en problemas binarios. Permite valorar la calidad de las detecciones mediante la representación gráfica de la tasa de detección (en el eje de ordenadas) y la tasa de falsas alarmas (en el eje de abscisas). Esta representación tiende a saturarse conforme la tasa de detección aumenta y, en consecuencia, la tasa de falsas alarmas. Una vez realizada la representación gráfica de los 6 puntos, uno para cada resolución de detección (aunque el de mayor resolución queda fuera de rango por lo que se ha preferido no representarlo), generaremos lo que denominaremos como *convex hull* o envoltura convexa. Se define la envoltura convexa como la intersección de todos los conjuntos convexos que contienen el conjunto de puntos X. En nuestro caso el conjunto de puntos vendrá determinado por la relación de la tasa de detección con la tasa de falsas alarmas. Definiremos **el punto de trabajo o punto óptimo** en que el algoritmo implementado se debe comportar mejor, como aquel punto perteneciente a la envoltura convexa que nos ofrezca mejores resultados.



**Fig. 5.66.- Representación de la ROC de los valores obtenidos para la tasa de detección y de falsas alarmas. El *convex hull* se representa en azul.**

De este modo, el punto a partir del que trabajaremos es el representado con la cruz roja y que corresponde a una detección con una resolución de *bin* de **500 cm** para la posición Y y de **50 cm/s** para la velocidad.

En el caso que nos ocupa, es preferible obtener una falsa alarma más que no una detección menos. Esto es debido a que el no detectar un tumulto puede provocar serios daños, mientras que una falsa alarma obligaría al encargado de la rampa a estar pendiente de si el algoritmo ha detectado correctamente o no, lo que resulta claramente menos peligroso que una no detección. Así pues los parámetros óptimos que tomamos para la detección son los siguientes:

Resolución en Y [cm]	Resolución en Velocidad [cm/s]	Tasa de detección	Tasa de falsas alarmas
<b>250</b>	<b>50</b>	<b>67%</b>	<b>22%</b>

**Tabla 5.10.- Resultados definitivos empleados en la detección de tumultos.**

### 5.3.- Resultados globales para las 100 secuencias:

Los últimos resultados que expondremos son los que reflejan la calidad del algoritmo de un modo más genérico, es decir, una vez decidida la resolución óptima de detección e implementados todos los criterios de mejora del seguimiento, un buen indicador de la calidad del algoritmo, a parte de las ya mencionadas, es el número de falsas alarmas, o tasa, que obtendríamos tras aplicarlo a 100 secuencias de escenarios y condiciones totalmente distintas. Así pues, extraeremos una tasa de falsas detecciones global aplicado a las 100 secuencias etiquetadas según la tabla 4.1.

ESCENARIO	E1	E2	E3	E4
Nº de secuencias	21	42	30	7
Tasa de falsas alarmas	5%	57%	20%	0%
Tasa total	<b>31%</b>			

**Tabla 5.11.- Resultados definitivos empleados en la detección de tumultos calculados con una resolución de 250x50.**

De modo que obtenemos un valor global de falsas alarmas del 31%, sobre un total de 100 secuencias, un valor bastante bueno para ser una primera aproximación. Hay que destacar que se ha enfatizado el análisis en secuencias del escenario 2 ya que es, debido a sus características, donde la situación de tumulto suele ocurrir con mayor frecuencia. En el Anexo 5 se desglosa esta tabla para cada una de las secuencias analizada.

Existe algún tipo de situaciones que pueden presentar las mismas características que se dan lugar en una situación de tumulto. Estas características, como ya se ha visto, consisten, a grandes rasgos, en **la coincidencia de una gran multitud de personas que se mueven por la rampa con la de un grupo estático de personas al final de ésta**. De este modo, el algoritmo las clasificará como tumultos sin, en realidad, serlo.

Un ejemplo de este tipo de situaciones se da en una de las secuencias analizada, la E2\_42, donde se observan 3 personas de mantenimiento al final de la rampa, grupo estático de personas, que permanecen un largo periodo de tiempo moviéndose sólo por el final de la rampa por razones que desconocemos, y un continuo flujo de personas a través de la rampa. Tratar de evitar la detección de este tipo de situaciones excede las pretensiones de este trabajo, por lo que, para este tipo de situaciones se debería parar la detección de tumultos, ya que provocan falsas alarmas cuando no tiene ningún tipo de sentido el estudio de estos casos.

A continuación, mostraremos algunos instantes de esta secuencia en los que se detecta tumulto:

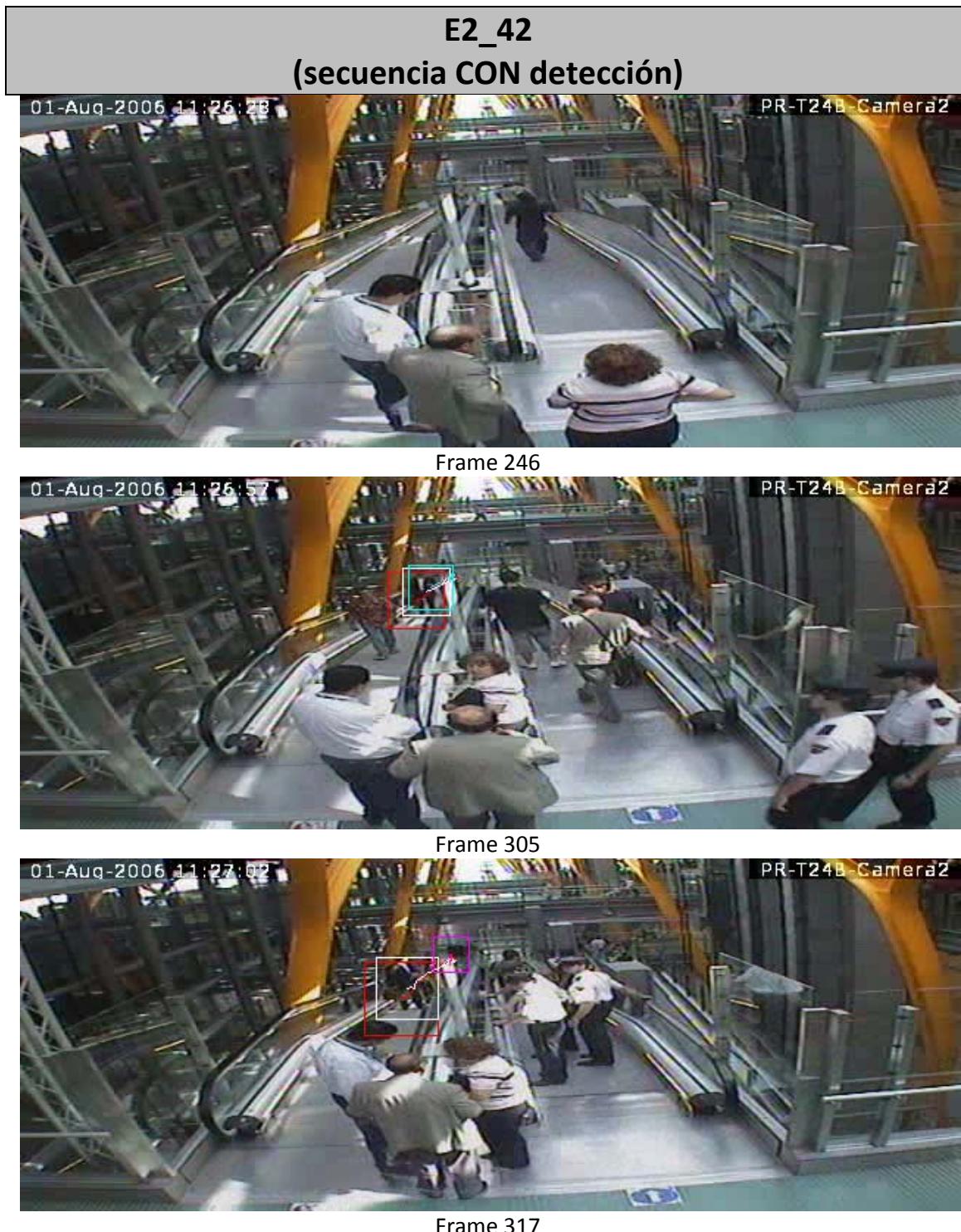


Fig. 5. 67.- En el frame 246 se observan las 3 personas que confundirán al algoritmo, provocando una falsa alarma. En el frame 305 y 307 se aprecia el inicio del seguimiento a lo largo de la rampa.

## E2\_42 (continuación) (secuencia CON detección)



Fig. 5. 68.- En el frame 328 se observa como el seguimiento llega al final de la rampa siguiendo correctamente la gente de la rampa. En cambio en el frame 331 se empieza a observar como el seguimiento se *confunde* y sigue tanto a las 3 personas como a la gente de la rampa. Finalmente cuando, en el frame 346, las personas de la rampa salen de la imagen, los filtros siguen a las 3 personas lo que provoca que la situación parezca un tumulto y, en consecuencia, se detecte como tal.

# 6.- Conclusiones

---

En este último capítulo, revisaremos los resultados obtenidos comparándolos con los objetivos establecidos al inicio del proyecto. Lo que implica revisar la contribución de la solución propuesta al problema del seguimiento y de la detección de tumultos. Finalmente, se comentarán los puntos a desarrollar en un futuro.

## 6.1.- Cumplimiento de los objetivos

El principal objetivo de este proyecto es **la detección de tumultos mediante el uso de los filtros de partículas**. La utilización de los filtros de partículas para el seguimiento de personas no supone una solución nueva, pero sí lo es el hecho de que se usen en un entorno con perspectiva y para la detección de tumultos.

La principal ventaja que ofrecen los filtros de partículas son: su escalabilidad, lo que permite añadir, de modo sencillo, espacios de características distintas para el cálculo de pesos, su capacidad para hacer el seguimiento sobre una región del espacio y no sobre un único punto y la flexibilidad que nos ofrecen para modificar pequeños detalles del seguimiento de manera que se puedan adaptar sin problemas a distintos escenarios y situaciones.

A continuación revisaremos como se han ido cumpliendo las distintas etapas que se propusieron al inicio del proyecto como camino para lograr nuestro objetivo final:

1. Detección de objetos de primer plano o extracción de foreground: pese a la sencilla detección implementada logramos obtener unos resultados bastante aceptables, gracias en gran parte, a la utilización del espacio de color (Y, Cb, Cr) con el que se logran solventar gran parte de los problemas causados por las sombras y los cambios de brillo.
2. Seguimiento de flujo de personas: se ha implementado a partir del seguimiento de las características de color y *foreground* con las que se consigue **extraer el movimiento global o flujo de personas en una cinta transportadora** en la mayoría de los casos, de modo que se logra alcanzar el principal objetivo propuesto. Las mejoras aplicadas al seguimiento logran solventar gran parte de los problemas causados por la perspectiva y la gran cantidad de personas presente en los escenarios.
3. Etiquetado manual de las secuencias: gracias al etiquetado logramos extraer una gran variedad de secuencias con distintos problemas que nos permiten mejorar la robustez y fiabilidad del algoritmo desarrollado. Asimismo se logra definir con claridad lo que se considera como un *tumulto* y, a partir del etiquetado, se llegan a localizar 9 secuencias que presentan este tipo de situaciones, las que cuales emplearemos para determinar la calidad de detección. También se realiza el etiquetado manual e individual del *Ground Truth* para estas 9 secuencias con tumulto.

4. Detección de tumultos: a partir de los dos métodos implementados se logra llegar a la obtención de un método de detección, tras descartar la idea inicialmente propuesta, con el que se obtienen resultados bastante aceptables. Pese haber descartado el primer método, su aportación para decidir la estrategia a seguir en la detección ha sido clave. Por otro lado, la detección se ha desarrollado con una serie de distintos parámetros que nos permiten adaptarla a las características de la detección, de modo que, a través de la comparación de los resultados obtenidos con distintos valores, hemos decidido cuáles nos aportan unos mejores resultados.
5. Evaluación de la calidad del algoritmo de detección: mediante la definición de la tasa de detección y de falsas alarmas se ha podido valorar de modo cuantitativo la calidad del método de detección desarrollado.

En conclusión, a partir de una elección adecuada de todos los parámetros empleados para realizar la extracción de objetos de primer plano, el seguimiento y la detección, se obtiene una tasa de **detección del 67% y de falsas alarmas del 22%**. Si ampliamos el análisis a un **total de 100 secuencias** la tasa de falsas alarmas obtenido es del **31%**.

Es importante resaltar, que el hecho de no disponer de un número importante de secuencias con tumulto ha determinado el tipo de enfoque implementado en la detección, y que ésta no se haya podido realizar mediante un clasificador de tumultos generado a partir del entrenamiento de secuencias con tumulto. Hay que tener presente que, afortunadamente, este tipo de situaciones no ocurren con frecuencia, por lo que pese a la gran cantidad de secuencias de que disponemos (cerca de 900 horas de grabaciones), las situaciones con *tumulto* no abundan. Por lo que, si se pretende enfocar la detección desde este otro punto de vista, es necesario ampliar el número de secuencias con *tumulto* y, en consecuencia, el número de secuencias donde buscar.

En resumen, para ser un primer acercamiento a la solución del problema de la detección de tumultos, los resultados obtenidos son bastante buenos y ofrecen una primera aproximación a partir de la cual seguir trabajando.

## 6.2.- Futuras líneas de trabajo

Las futuras líneas de trabajo que deja este proyecto abarcan un gran abanico de posibilidades. A continuación se expondrán algunas de estas posibilidades.

En primer lugar, se podría mejorar la extracción de *foreground* mediante el desarrollo de sistemas más robustos frente a variaciones bruscas de luminosidad y obtener una mayor fiabilidad, a partir de la combinación de la extracción de primer plano a nivel de píxel con detecciones a nivel global de la imagen de manera que ayuden a reducir las falsas detecciones. Otra opción sería implementar otro tipo de extracción de *foreground* basado en una combinación de múltiples Gaussianas o en un detector Multinivel.

En segundo, el seguimiento implementado dispone de una serie de parámetros con los que se puede jugar para mirar de obtener mejores resultados. Estudiar si la ampliación o reubicación de la zona de inicialización supondría una mejora en los resultados obtenidos. También se podría plantear la implementación de algún tipo de bloqueo entre filtros lo que evitaría tener que eliminarlos cuando se solapen y así poder garantizar que siguieran objetivos distintos.

En tercer lugar, probar otros criterios o métodos de detección de tumultos basados en el estudio de otro tipo de variables o características de los filtros de partículas.

En cuarto lugar, realizar una ampliación del número de secuencias con y sin tumulto para ver el comportamiento del algoritmo en nuevas situaciones y poderlo mejorar. De este modo, al trabajar con un mayor número de secuencias con tumulto, se podría plantear el estudio de una detección mediante un clasificador generado a partir del entrenamiento con estas secuencias. También se podría ampliar la implementación a otro tipo de escenarios lo que permitiría comprobar la portabilidad del trabajo realizado.

En quinto y último lugar, se podría plantear la modificación en la posición de las cámaras o añadir nuevas, como sería la incorporación de una cámara cenital respecto a la rampa, que podría contribuir en mejorar la robustez del algoritmo y muy posiblemente supondría un replanteamiento global de la estrategia de detección llevada a cabo.

Asimismo, la optimización el algoritmo, desde el punto de vista de programación, para que sea capaz de detectar tumultos en tiempo real es uno de los aspectos a mejorar si se pretende, en un futuro, implantar en un caso real.

Finalmente, una de las futuras líneas de trabajo consiste en poder integrar el trabajo desarrollado en cuestión de implementación de código en el lenguaje de programación C y C++, para el software de desarrollo “ImagePlus” del grupo de procesado de imagen del departamento TSC de la Universidad Politécnica de Cataluña.

# TABLA DE ILUSTRACIONES

---

Fig. 1. 1.- Vista del escenario 1, el que muestra la cámara PRT7A-Camera1.....	9
Fig. 1. 2.- Vista del escenario 4, el que muestra la cámara PRT7A-Camera2.....	9
Fig. 1. 3.- Vista del escenario 3, el que muestra la cámara PRT24B-Camera1.....	10
Fig. 1. 4.- Vista del escenario 2, el que muestra la cámara PRT24B-Camera2.....	10
Fig. 1. 5.- Ejemplo de un tumulto en el escenario 2.....	11
Fig. 1. 6.- Distribución de probabilidad multimodal.....	14
Fig. 2.- 7.- Ejemplo del funcionamiento del principio <i>Importance Sampling</i> .....	22
Fig. 2. 8.- Funcionamiento del remuestreo, se puede ver como las partículas con un peso pequeño “mueren” mientras que las de mayor peso se remuestrean con mayor cardinalidad que las de menor.....	24
Fig. 2. 9.- Diagrama de bloques del filtro de partículas.....	26
Fig. 2. 10.- Diagrama de bloques de la dinámica de seguimiento que se implementará. ....	28
Fig. 2. 11.- Situación de las zonas de inicialización y desaparición en el escenario 1.....	29
Fig. 3. 12.-Diagrama de flujo del sistema desarrollado en este proyecto.....	30
Fig. 3. 13.-Modelo de píxel como una función Gaussiana. ....	32
Fig. 3. 14.-Explicación gráfica del modelo de 1 sola Gaussiana. ....	33
Fig. 3. 15.-Diagrama de bloques del método <i>Running Gaussian Average</i> .....	34
Fig. 3. 16.- (a) Observamos una persona al final de la rampa que provocará una mala inicialización de la media, calculada a partir de una secuencia de training de 50 frames. (b) Extracción de foreground, representada por el color verde, obtenida en el frame 51, igual para todos los distintos valores de $\alpha$ . (c) Extracción de foreground obtenida en el frame 100 con un $\alpha = 1\%$ . (d) Extracción de foreground obtenida en el frame 100 con un $\alpha = 15\%$ .(e) Extracción de foreground obtenida en el frame 100 con un $\alpha = 25\%$ . (f) Extracción de foreground obtenida en el frame 200 con un $\alpha = 1\%$ . (g) Extracción de foreground obtenida en el frame 200 con un $\alpha = 15\%$ . (h) Extracción de foreground obtenida en el frame 200 con un $\alpha = 25\%$ . ....	35
Fig. 3. 17.- (a) Extracción de foreground obtenida en el frame 290 con un $\alpha = 1\%$ . (b) Extracción de foreground obtenida en el frame 290 con un $\alpha = 15\%$ . (c) Extracción de foreground obtenida en el frame 290 con un $\alpha = 25\%$ . (d) Extracción de foreground obtenida en el frame 290 con un $\alpha = 35\%$ , el caso extremo donde se observa como las personas que pasan por la rampa queda casi directamente incorporada como <i>background</i> .....	36
Fig. 3. 18.- Comparativa entre la extracción de foreground con distintos espacios de color. (a) Usando RGB. (b) Usando YCbCr. ....	37
Fig. 3. 19.- Representación de la imagen media de fondo de escena calculada a partir de la secuencia de training del escenario 1. ....	38
Fig. 3. 20.- Representación de la imagen varianza calculada a partir de la secuencia de training para el escenario 1. ....	39
Fig. 3. 21.- (a) Píxel detectado como primer plano. (b) Píxel detectado como fondo. ....	40
Fig. 3. 22.- (a) Una persona que podría interesar seguir. (b) Ejemplo de un histograma de color en YCb-YCr que podría corresponder a la persona de la figura (a). ....	41
Fig. 3. 23.- Muestra de un filtro de partículas caracterizado por un rectángulo. ....	43
Fig. 3. 24.- Muestra de los 3 instantes claves a partir de los que generaremos la tabla de adaptación del tamaño del filtro de partículas. ....	44

Fig. 3. 25.-Muestra del filtro de partículas anterior, rojo, estimado a partir de 5 partículas, en verde, también caracterizadas por un rectángulo.....	45
Fig. 3. 26.- Imagen perteneciente al escenario 1 donde se puede ver el tamaño del blob inicial. ....	45
Fig. 3. 27.- Imagen que muestra el tamaño de la zona de inicialización y su ubicación para este escenario, el 1. ....	46
Fig. 3. 28.- Imagen que muestra el tamaño de la zona de desaparición y su situación para esta rampa en concreto.....	47
Fig. 3. 29.- Imagen de la recta con la que modelamos la rampa. ....	48
Fig. 3. 30.- Ejemplo gráfico del funcionamiento del <i>morphing</i> implementado a partir del ángulo $\alpha$ . .....	49
Fig. 3. 31.- Ejemplo gráfico del funcionamiento del <i>morphing</i> doblando la propagación en la zona amarilla. ....	49
Fig. 4. 32.- Ejemplo de cómo se soluciona el problema de que el filtro siga la gente que circula por la rampa ascendente gracias al criterio de bloqueo en la secuencia E1_5. ....	51
Fig. 4. 33. – Ejemplo de dónde está situada la recta de bloqueo en el escenario 1. ....	52
Fig. 4. 34.- Imagen correspondiente al <i>frame</i> 192 de la secuencia E4_4 donde se puede ver la cantidad de gente que hay en el lado izquierdo y como se consigue que ésta no afecte al seguimiento.....	53
Fig. 4. 35.- En esta imagen se observa una idea de a partir de dónde las partículas reciben un peso nulo, es decir, una vez sobrepasan la recta en rojo. ....	53
Fig. 4. 36.- Imagen donde se observa el filtro de partículas entre dos grupos de personas. ....	53
Fig. 4. 37.- Serie de imágenes pertenecientes a la secuencia E1_6, donde se observa claramente la mejora que supone aplicar este criterio ya que elimina la ambigüedad que genera tener dos objetivos muy próximos y permite que el filtro de partículas se centre en uno. El color rojo indica foreground. ....	54
Fig. 4. 38.-Representación del funcionamiento de este criterio. El filtro de partículas tiene el centroide en la cruz azul y su tamaño es el rectángulo del mismo color. El rectángulo rojo representa el 150% del tamaño del filtro y, por lo tanto, la distancia a partir de la cual las partículas serán eliminadas. Las partículas rojas son eliminadas y las verdes sobreviven.....	55
Fig. 4. 39.- En esta imagen se puede ver como el filtro de partículas sigue vivo pese a que no hay nada que seguir, la persona se encuentra fuera del rango de la cámara. ....	56
Fig. 4. 40.- Cada cruz amarilla representa el centroide de una partícula y el color verde indicada el <i>foreground</i> detectado. ....	57
Fig. 4. 41.- El filtro de partículas está siguiendo claramente a una persona, pero ésta no es la que debería estar siguiendo, sino la que se ha detenido al final de la rampa. El color rojo indicada el <i>foreground</i> detectado.....	57
Fig. 4. 42.- El filtro de partículas está siguiendo claramente a una persona en la rampa. El color rojo indicada el <i>foreground</i> detectado. ....	58
Fig. 4. 43.- Ejemplo visual de cómo un filtro con 20 partículas, será destruido ya que 11 partículas están fuera de la zona delimitada por la zona roja, el máximo permitido por este criterio a la que pueden estar las partículas. ....	58
Fig. 4. 44.- Imagen ampliada del frame 242 de la secuencia E1_2 donde se puede observar cómo, tras la aplicación de este criterio, sólo 1 filtro de partículas sigue a una persona. El color rojo indica el <i>foreground</i> extraído. ....	59

Fig. 4. 45.- Cada cruz representa un centroide de un filtro de partículas distinto. El filtro rojo	60
será eliminado al estar demasiado próximo al negro, que es el más antiguo. En cambio, el filtro	
verde permanecerá vivo al no estar lo suficientemente cerca del negro.....	
Fig. 4. 46.- Imagen perteneciente a la secuencia E2_7, donde se puede ver como la multitud se	60
logra seguir con 4 filtros de partículas.....	
Fig. 4. 47.- Ejemplo que corresponde a la secuencia E2_3, donde se observa como las	61
trayectorias de los filtros vuelven hacia atrás tras realizar el seguimiento, un caso de una	
situación de retroceso.....	
Fig. 4. 48.- Imagen que muestra lo que consideraremos una situación con tumulto. ....	62
Fig. 4. 49.- Imágenes del seguimiento con filtros de partículas de una secuencia sin tumulto y	65
una con.....	
Fig. 4. 50.- Representación de la velocidad de los filtros de partículas durante una secuencia sin	66
tumulto, que consta de dos seguimientos independientes y sencillos. ....	
Fig. 4. 51.- Zoom de la imagen anterior, en la que se puede apreciar con claridad los dos	66
seguimientos realizados por los filtros de partículas.....	
Fig. 4. 52.- Representación de la velocidad de los filtros de partículas durante una secuencia	67
con tumulto.....	
Fig. 4. 53.- Zoom de la imagen anterior destacando la zona de inicio de un tumulto.....	68
Fig. 4. 54.- Imagen que muestra una situación de detección de un tumulto, nuestro objetivo	68
final, con la correspondiente <i>alarma</i> , perteneciente a la secuencia E2_7.....	
Fig. 4. 55.- Explicación gráfica de cuáles son los ángulos $\alpha$ y $\beta$ . La idea es la misma en el resto de	70
escenarios.....	
Fig. 4. 56.- Ejemplo de detección, perteneciente a la secuencia E2_7 y realizada con una	71
resolución de 500x100, donde se aprecian claramente los 3 estados de colores que pueden	
presentar las celdas. Color verde -> 1 filtro (no hay peligro), color naranja -> 2 filtros (posible	
peligro) y color rojo -> 3 filtros (posible tumulto).....	
Fig. 4. 57.- Ejemplo, correspondiente a la secuencia E1_2, de un seguimiento sencillo y la	72
detección realizada. Tal y como se observa, al haber un único objetivo, tan solo se genera un	
filtro de partículas y el estado de las celdas es siempre verde, es decir, sin peligro.....	
Fig. 4. 58.- Ejemplo, correspondiente a la secuencia E2_7, de un seguimiento complejo y con	73
tumulto. Se observa, como se realiza correctamente la detección del tumulto con una	
resolución de 500x100. ....	
Fig. 4. 59.- <i>Ground Truth</i> de las secuencias E2_7 y E2_14, respectivamente. ....	74
Fig. 4. 60.- <i>Ground Truth</i> de las secuencias E2_7 y E2_14 respectivamente. ....	75
Fig. 4. 61.- La detección presentaría una tasa de detección del 100% y de falsas alarmas del	76
200%.....	
Fig. 4. 62.- En este caso, la detección se mantiene en una tasa de detección del 100%, mientras	77
que el de falsas alarmas disminuye al 100%.....	
Fig. 5. 63.- Imágenes que muestran la detección (en rojo) frente al <i>Ground Truth</i> (en azul) para	78
la secuencia E2_3. (a) La resolución empleada de 500x100. (b) La resolución empleada es de	
250x50.....	
Fig. 5. 64.- Imágenes que muestran la detección (en rojo) frente al <i>Ground Truth</i> (en azul) para	79
la secuencia E2_7. (a) La resolución empleada de 500x100. (b) La resolución empleada es de	
100x50.....	

Fig. 5. 65.- Imágenes que muestran la detección (en rojo) frente al Ground Truth (en azul) para la secuencia E2_18. (a) La resolución empleada de 500x100. (b) La resolución empleada es de 250x50.....	79
Fig. 5.66.- Representación de la ROC de los valores obtenidos para la tasa de detección y de falsas alarmas. El <i>convex hull</i> se representa en azul.....	83
Fig. 5. 67.- En el frame 246 se observan las 3 personas que confundirán al algoritmo, provocando una falsa alarma. En el frame 305 y 307 se aprecia el inicio del seguimiento a lo largo de la rampa. ....	85
Fig. 5. 68.- En el frame 328 se observa como el seguimiento llega al final de la rampa siguiendo correctamente la gente de la rampa. En cambio en el frame 331 se empieza a observar como el seguimiento se <i>confunde</i> y sigue tanto a las 3 personas como a la gente de la rampa. Finalmente cuando, en el frame 346, las personas de la rampa salen de la imagen, los filtros siguen a las 3 personas lo que provoca que la situación parezca un tumulto y, en consecuencia, se detecte como tal.....	86
Fig. A.69.- (a) Filtro de partículas con una varianza alta (20), se puede apreciar el gran tamaño de la nube de partículas. (b) Si la varianza es pequeña (5), la nube de partículas es menor y se ciñe más a la posición del centroide del filtro.....	99
Fig. A.70.- (a) Estimación de la recta $y = x$ con una $\beta = 5$ , sigue las oscilaciones debidas al ruido. (b) Estimación de la recta $y = x$ con una $\beta = 1$ , se ciñe a seguir el auténtico valor de la recta sin ruido. ....	100
Fig. A.71.- (a) Estimación de la recta $y = 3x + 1$ con una $\beta = 3$ , sigue bastante bien la forma de la recta pese al ruido que ésta presenta. (b) Estimación de la recta $y = 3x + 1$ con una $\beta = 1$ , le cuesta seguir la recta pese a hacer un intento de seguir su pendiente.....	100
Fig. A.72.- (a) Estimación de la recta $y = 3x + 1$ con una $\beta = 5$ , sigue también bastante bien la forma de la recta pese al ruido que ésta presenta, algo mejor al principio pero al final parece que se pierde un poco. (b) Estimación de la recta $y = 3x + 1$ con una $\beta = 10$ , sigue bastante bien la evolución de la recta hasta que llega un momento en el que se pierde del todo. ....	101
Fig. A.73.- (a) Estimación de una sinusoide con una $\beta = 3$ , sigue bastante bien su forma pese a perder algún valor de sus picos. (b) Estimación con $\beta = 1$ , sigue bastante bien su forma pero pierde totalmente los valores de sus picos.....	101
Fig. A.74.- (a) Estimación con $\beta = 5$ , sigue muy bien su forma sin perder los valores de sus picos. (b) Estimación con $\beta = 0.1$ , se pierde totalmente al seguir la sinusoide, tanto los picos como su forma.....	102
Fig. A.75.- (a) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 3$ y varianza = 3, se pierde totalmente al inicio. (b) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 1$ y varianza = 3, se sigue perdiendo al inicio. (c) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 0.1$ y varianza = 3, también se pierde al inicio. (d) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 3$ y varianza = 15, empieza bien la estimación pero en seguida se pierde. (e) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 1$ y varianza = 15, empieza algo mejor que la anterior estimación pero se sigue perdiendo. (f) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 0.1$ y varianza = 15, empieza bien la estimación y logra seguir algo más la curva que las estimaciones anteriores.....	103
Fig. A. 76.- (a) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 3$ y varianza = 50, inicialmente sigue bien la parábola pero se pierde a mitad. (b) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 1$ y varianza = 50, similar a la anterior pero se pierde más tarde. (c) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 0.1$ y varianza = 50, mejora las dos anteriores pero sin llegar a estimarla completamente. (d) Estimación de la	

recta $y = x^2 + x + 1$ con una $\beta = 3$ y varianza = 100, se sigue perdiendo a mitad pese a llegar más arriba. (e) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 1$ y varianza = 100, llega hasta el final pese a tener cada vez más ruido. (f) Estimación de la recta $y = x^2 + x + 1$ con una $\beta = 0.1$ y varianza = 100, similar a la anterior pero con algo menos de ruido sobre todo al principio.....	104
Fig. A. 77.- Muestra de qué representan, a nivel de color, los cubos de color de ambos espacios. (a) El modelo RGB. (b) El modelo YCbCr.....	106
Fig. A. 78.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.....	109
Fig. A. 79.- Imagen que muestra el tamaño y la posición de la zona de inicialización.....	109
Fig. A. 80.- Imagen que muestra el tamaño y la posición de la zona de destrucción. ....	110
Fig. A. 81.- Imagen que muestra la recta que modela la rampa, $Y = -1.083*X + 368$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas. ....	110
Fig. A. 82.- Imagen que muestra la recta que de bloqueo, $Y = -2.96*X + 1021$ , empleada en este escenario.....	110
Fig. A. 83.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.....	112
Fig. A. 84.- Imagen que muestra el tamaño y la posición de la zona de inicialización.....	112
Fig. A. 85.- Imagen que muestra el tamaño y la posición de la zona de destrucción. ....	113
Fig. A. 86.- Imagen que muestra la recta que modela la rampa, $Y = -1.317*X + 450$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas. ....	113
Fig. A. 87.- Imagen que muestra la recta que de bloqueo, $Y = -11.63*X + 3720$ , empleada en este escenario.....	113
Fig. A. 88.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.....	115
Fig. A. 89.- Imagen que muestra el tamaño y la posición de la zona de inicialización.....	115
Fig. A. 90.- Imagen que muestra el tamaño y la posición de la zona de destrucción. ....	116
Fig. A. 91.- Imagen que muestra la recta que modela la rampa, $Y = -3.123*X + 1075$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas. ....	116
Fig. A. 92.- Imagen que muestra la recta que de bloqueo empleada, $Y = 6.731*X + 2396$ , en este escenario.....	116
Fig. A. 93.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.....	118
Fig. A. 94.- Imagen que muestra el tamaño y la posición de la zona de inicialización.....	118
Fig. A. 95.- Imagen que muestra el tamaño y la posición de la zona de destrucción. ....	119
Fig. A. 96.- Imagen que muestra la recta que modela la rampa, $Y = -2.535*X + 817$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas. ....	119
Fig. A. 97.- Imagen que muestra la recta que de bloqueo, $Y = 3.694*X - 1170$ , empleada en este escenario.....	119
Fig. A. 98.- Matriz inicial y ejemplo de una máscara genérica para la morfología matemática.	
.....	121

Fig. A. 99.- Ejemplo de los 3 tipos distintos conectividad: conectividad 4, la empleada, conectividad 6 y conectividad 8.....	122
Fig. A. 100.- (a) Extracción de foreground sin tratamiento de post-procesado. (b) Extracción de foreground con tratamiento de post-procesado de tamaño de 25 píxeles y con conectividad 4.	
.....	122

# LISTA DE TABLAS

---

Tabla 3. 1.- Caracterización del tamaño del filtro de partículas en el escenario 1 .....	44
Tabla 3. 2.- Valores, en píxeles, de los parámetros necesarios para caracterizar completamente la zona de inicialización del escenario 1.....	46
Tabla 3. 3.- Valores, en píxeles, de los parámetros necesarios para caracterizar completamente la zona de destrucción del escenario 1 .....	46
Tabla 3. 4.- Caracterización del tamaño y de la varianza de propagación de las partículas en el escenario 1 .....	48
Tabla 4.5.- Información de la cantidad de secuencias de que disponemos por escenario y la cantidad de tumultos que éstas presentan.....	63
Tabla 5.6.- Resultados individuales de la tasa de detección según las distintas resoluciones de detección y para las 9 secuencias con tumulto.....	80
Tabla 5.7.- Resultados globales de la tasa de detección para las distintas resoluciones de detección.....	81
Tabla 5.8.- Resultados individuales de la tasa de falsas alarmas según las distintas resoluciones de detección y para las 9 secuencias con tumulto.....	81
Tabla 5.9.-Resultados de la tasa de falsas alarmas para las distintas resoluciones de detección.	82
.....	
Tabla 5.10.- Resultados definitivos empleados en la detección de tumultos.....	83
Tabla 5.11.- Resultados definitivos empleados en la detección de tumultos calculados con una resolución de 250x50. ....	84
Tabla A. 12.- Esta tabla muestra la evolución del tamaño del filtro de partículas en función de su posición según Y y la varianza empleada en la propagación de las partículas. Las celdas amarillas corresponden a los instantes de la Fig. A. 81.- Imagen que muestra la recta que modela la rampa, $Y = -1.083*X + 368$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.....	111
Tabla A. 13.- Esta tabla muestra la evolución del tamaño del filtro de partículas en función de su posición según Y y la varianza empleada en la propagación de las partículas. Las celdas amarillas corresponden a los instantes de la Fig. A. 86.- Imagen que muestra la recta que modela la rampa, $Y = -1.317*X + 450$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.....	114
Tabla A. 14.- Esta tabla muestra la evolución del tamaño del filtro de partículas en función de su posición según Y y la varianza empleada en la propagación de las partículas. Las celdas amarillas corresponden a los instantes de la Fig. A. 91.- Imagen que muestra la recta que modela la rampa, $Y = -3.123*X + 1075$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas. ....	117
Tabla A. 15.- Esta tabla muestra la evolución del tamaño del filtro de partículas en función de su posición según Y y la varianza empleada en la propagación de las partículas. Las celdas amarillas corresponden a los instantes de la Fig. A. 96.- Imagen que muestra la recta que modela la rampa, $Y = -2.535*X + 817$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.....	120
Tabla A. 16.- Resultados de la tasa de falsas alarmas obtenido para cada secuencia para una resolución de 250x50. ....	123

# BIBLIOGRAFIA

---

- [1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “*A Tutorial on Particle Filters for On-Line Non-Linear/Non-Gaussian Bayesian Tracking*”, IEEE Trans. Signal Processing, vol. 50, no. 2, pp. 174-189, 2002.
- [2] Carlos Segura, “*Bayesian Filtering*”, Presentación en UPC-TALP.
- [3] C. Canton-Ferrer, Carlos Segura, J. R. Casas, M. Pardàs and Javier Hernando, “*Audiovisual head orientation estimation in multisensor scenarios*”, EURASIP Journal on Advances in Signal Processing, 2008.
- [4] A. López, C. Canton-Ferrer and J. R. Casas. “*Multi-Person 3D Tracking with Particle Filters on Voxels*”. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Honolulu (USA), April 16-20, 2007.
- [5] A. López, “*Seguimiento 3D para Múltiples Personas en Entornos Multicaminos*”, PFC ETSETB (UPC).
- [6] Stefan Karlsson, Murtaza Taj, and Andrea Cavallaro, “*Detection and Tracking of Humans and Faces*,” EURASIP Journal on Image and Video Processing, 2008.
- [7] Rudolph van der Merve, Arnaud Doucet, Nando de Freitas, Eric Wan, “*The Unscented Particle Filter*” Advances in Neural Information Processing Systems, 2001 - citeseer.ist.psu.edu.
- [8] K. Okuma, A. Talenghani, N. de Freitas, J. J. Little and D. G. Lowe, “*A Boosted Particle Filter: Multitarget Detection and Tracking*”, T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3021, pp. 28–39, 2004.
- [9] Xinyu Xu and Baoxin Li, “*Head Tracking Using Particle Filter With Intensity Gradient and Color Histogram*”, icme, pp. 888-891, 2005 IEEE International Conference on Multimedia and Expo, 2005.
- [10] J. MacCormick and A. Blake, “*A probabilistic exclusion principle for tracking multiple object*”, in Proc. Int. Conf. Comput. Vision, 1999, pp. 572–578.
- [11] Stauffer, C.; Grimson, W.E.L., “*Learning patterns of activity using real-time tracking*”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 22, Issue 8, Aug. 2000 Page(s): 747 - 757, <http://dx.doi.org/10.1109/34.868677>.
- [12] Kailath, T., “*The Divergence and Bhattacharyya Distance Measures in Signal Selection*”, IEEE Transactions on Communications, Vol.15, Iss.1, Feb 1967, Pages: 52- 60.
- [13] K. Nummiaro, E. Koller-Meier, and L. Van Gool, “*An Adaptive Color-Based Particle Filter*”, in Proc. of the 1st Workshop on Generative-Model-Based Vision, June 2002, pp. 53–60.K.
- [14] J. Deutscher, A. Blake and I. Reid, “*Articulated Body Motion Capture by Annealed Particle Filtering*”, PROC IEEE COMPUT SOC CONF COMPUT VISION PATTERN RECOGNIT, 2000 - csa.com.

- [15] John MacCaormick and Michael Isard, "Partitioned sampling, articulated objects, and interface-quality hand tracking", in Proc European Conf. Computer Vision, Vol.2, pp.3-19, 2000.
- [16] C. Wren, A. Azarbayenjani, T. Darrell, and A.P. Pentland, "Pfinder: real-time tracking of the human body" IEEE Trans. On Pattern Analysis and Machine Intell., vol. 19, nº 7, pp 780-785, 1997.
- [17] M. Piccardi. "Background subtraction techniques: a review", In Proc. of IEEE SMC 2004 International Conference on Systems, Man and Cybernetics, volume 4, pages 3099-3104, The Hague, The Netherlands, Oct 2004.
- [18] F. Aherne, N. Thacker, and P. Rockett, "The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data", Kybernetika, vol. 34, no. 4, pp. 363-368, 1998.
- [19] Ana Paula Majtey, "Sobre la Noción de Distancia en Física", Distancia Bhattacharyya, pp. 82.
- [20] Inder Jeet Taneja, "Distance Measures", Departamento de Matemática – UFSC, <http://www.mtm.ufsc.br/~taneja/book/node20.html>.
- [21] Jaime Gallego Vila, "Detección y seguimiento de objetos de primer plano", PFC ETSETB (UPC).
- [22] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, no. 5, pp. 564–577, May 2003.
- [23] N. Bergman, "Recursive Bayesian estimation: Navigation and tracking applications", Ph.D dissertation, Linköping Univ., Linköping, Sweden, 1999.
- [24] L-Q Xu. J. L. Landabaso and B Lei , "Segmentation and tracking of multiple moving objects for intelligent video analysys", BT Technology Journal Vol 22 No 3 July 2004.
- [25] D. Comaniciu, V. Ramesh and P. Meer, "Real-Time of Non-Rigid Objects using Mean Shift", IEEE Conf. Computer Vision and Pattern Recognition, South Carolina, Vol. 2, 2000
- [26] R. T. Collins, "Mean-shift Blob Tracking through Scale Space", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003.
- [27] Rossella Sblendido, "3D Multi-person tracking in multi-camera environments using color features for particle filters", PFC ETSETB (UPC).
- [28] J. Davis and M. Goadrich, "The Relationship Between Precision-Recall and ROC Curves", 2006, CONF 23, pages 233-240.

# ANEXOS

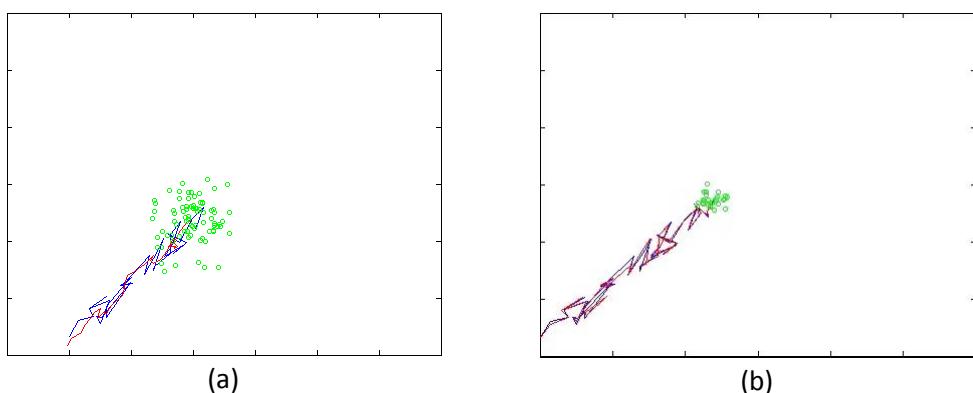
---

## A.1.- Toy example:

Una buena manera de inicializarse con una nueva tecnología o herramienta es aplicarla a una simplificación del caso real al que se pretende emplear, esto es lo que se conoce como ejemplo de prueba o *toy example*. Así pues, decidimos aplicar nuestra herramienta, los filtros de partículas, al seguimiento de una recta en 2D. Hay que tener presente que el seguimiento que se realiza durante el presente trabajo, a través de una rampa con perspectiva, no deja de ser una recta con mayor o menor inclinación. Es por este motivo que realizar un primer seguimiento sobre una simple recta nos permite determinar claramente el comportamiento de los filtros de partículas en estas condiciones y ver cómo se altera su conducta si cambiamos las condiciones del seguimiento, es decir, si en lugar de una recta queremos que sigan una curva o una sinusoida. Además existen una serie de parámetros que nos permiten adaptar los filtros de partículas al tipo de seguimiento que deseemos modelar y que, gracias a este trabajo previo, se han podido estudiar con mayor detalle del que nos hubiera permitido su estudio en el caso real.

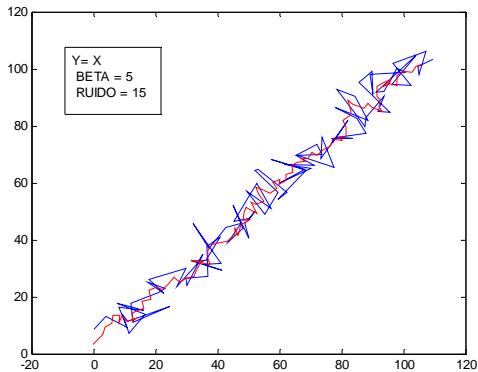
En primer lugar, veamos los parámetros que tendremos en cuenta en nuestro estudio, estos parámetros son algunos de los explicados en el apartado 2 y 3 del trabajo:

- $\beta$ : Este parámetro es el que nos determina si nos interesa que los pesos de las partículas no afecten todas de un modo homogéneo o únicamente las de mayor peso en el momento de estimar la posición real del filtro.
- **La varianza del filtro:** La varianza nos establece lo *lejos* que las partículas se pueden propagar respecto a su posición anterior, es decir, una varianza alta hará que la nube de partículas sea mayor, tal y como se ve en la figura 1.a, que la de una varianza baja, como la de la figura 1.b.

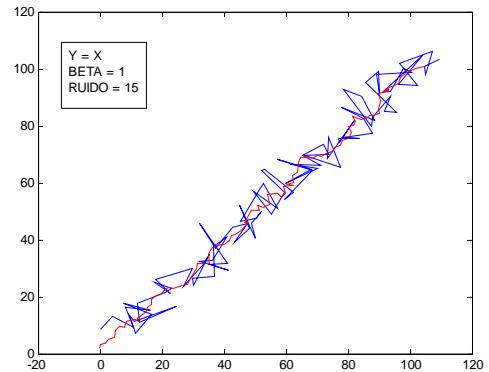


**Fig. A.69.-** (a) Filtro de partículas con una varianza alta (20), se puede apreciar el gran tamaño de la nube de partículas. (b) Si la varianza es pequeña (5), la nube de partículas es menor y se ciñe más a la posición del centroide del filtro.

En segundo lugar, si realizamos el estudio del parámetro  $\beta$  en la recta más sencilla de todas a la que le añadimos algo de ruido,  $Y = X$ , podemos observar los siguientes resultados:



(a)

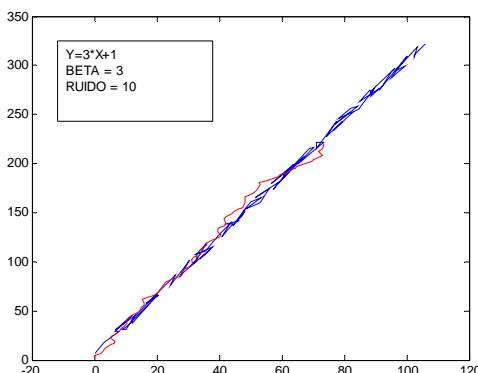


(b)

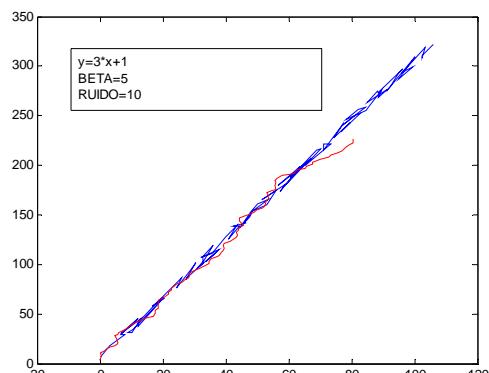
**Fig. A.70.- (a) Estimación de la recta  $y = x$  con una  $\beta = 5$ , sigue las oscilaciones debidas al ruido. (b) Estimación de la recta  $y = x$  con una  $\beta = 1$ , se ciñe a seguir el auténtico valor de la recta sin ruido.**

La recta azul es la recta que queremos estimar, es decir, la recta  $Y = X$  con ruido, mientras que la recta roja es la que resulta de la estimación realizada con los filtros de partículas. Se puede apreciar como si el valor de  $\beta$  es alto la estimación sigue con mayor precisión el ruido ya que nos predominan las partículas de mayor peso en la estimación, es decir, las que siguen mejor la recta, mientras que si el valor de  $\beta$  es cercano a 1 o inferior, la estimación del filtro de partículas tenderá a ser más parecida a la recta original sin ruido debido a que se tienen en cuenta todas las partículas por igual sin modificar su peso.

Si ampliamos el estudio a una recta algo más complicada, aumentando la pendiente y añadiendo un pequeño desplazamiento inicial respecto al origen, como por ejemplo, la recta de ecuación  $Y = 3 \cdot X + 1$ , y observamos lo que sucede con distintos valores de  $\beta$  en las siguientes 4 figuras:

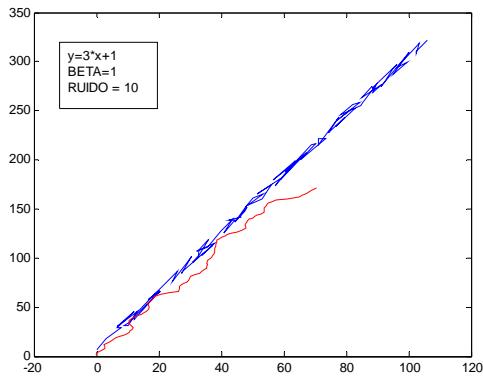


(a)

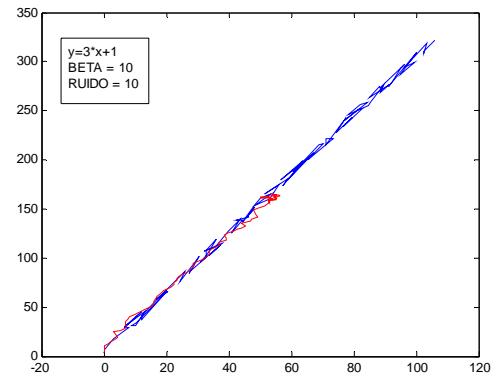


(b)

**Fig. A.71.- (a) Estimación de la recta  $y = 3x + 1$  con una  $\beta = 3$ , sigue bastante bien la forma de la recta pese al ruido que ésta presenta. (b) Estimación de la recta  $y = 3x + 1$  con una  $\beta = 5$ , le cuesta seguir la recta pese a hacer un intento de seguir su pendiente.**



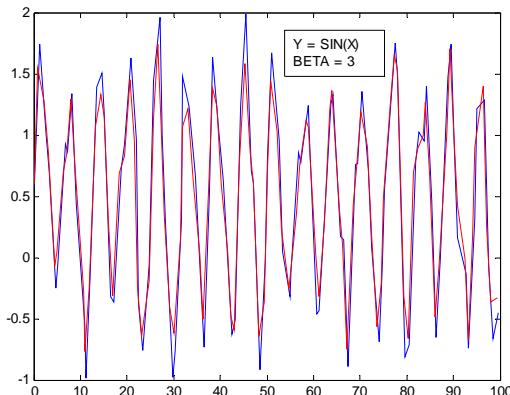
(a)



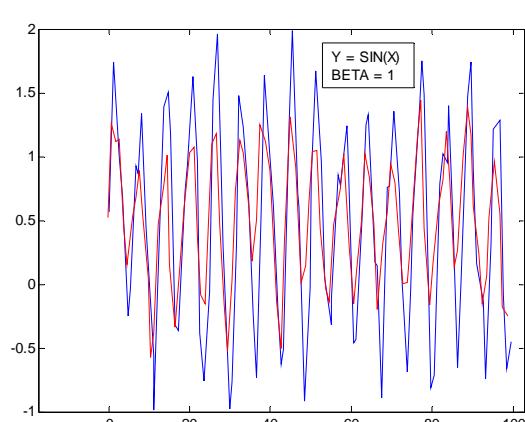
(b)

**Fig. A.72.-**(a) Estimación de la recta  $y = 3x + 1$  con una  $\beta = 5$ , sigue también bastante bien la forma de la recta pese al ruido que ésta presenta, algo mejor al principio pero al final parece que se pierde un poco. (b) Estimación de la recta  $y = 3x + 1$  con una  $\beta = 10$ , sigue bastante bien la evolución de la recta hasta que llega un momento en el que se pierde del todo.

Tras observar el comportamiento con una recta algo compleja, veamos cómo se comportan los filtros con el seguimiento de una sinusoides para poder extraer cuál es el valor de  $\beta$  que más se adecúa al caso real que nos enfrentaremos.

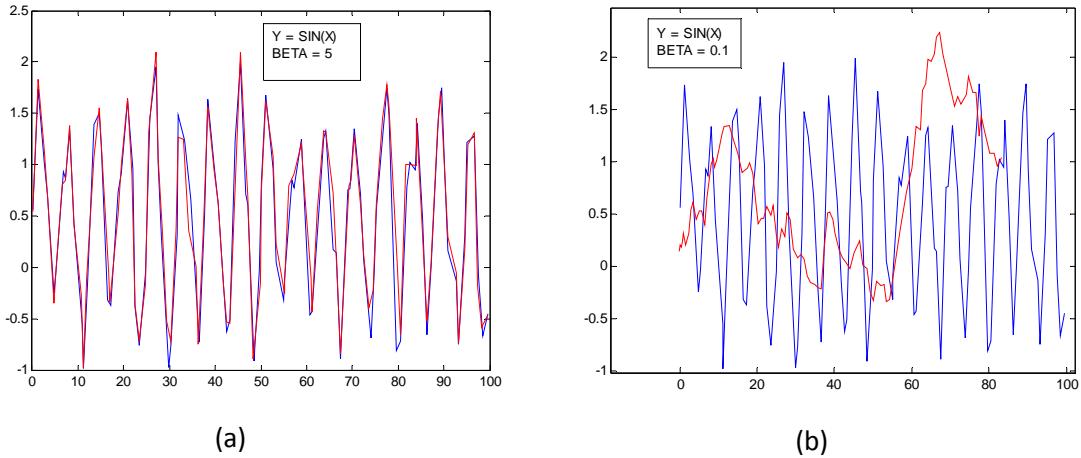


(a)



(b)

**Fig. A.73.-** (a) Estimación de una sinusoides con una  $\beta = 3$ , sigue bastante bien su forma pese a perder algún valor de sus picos. (b) Estimación con  $\beta = 1$ , sigue bastante bien su forma pero pierde totalmente los valores de sus picos.

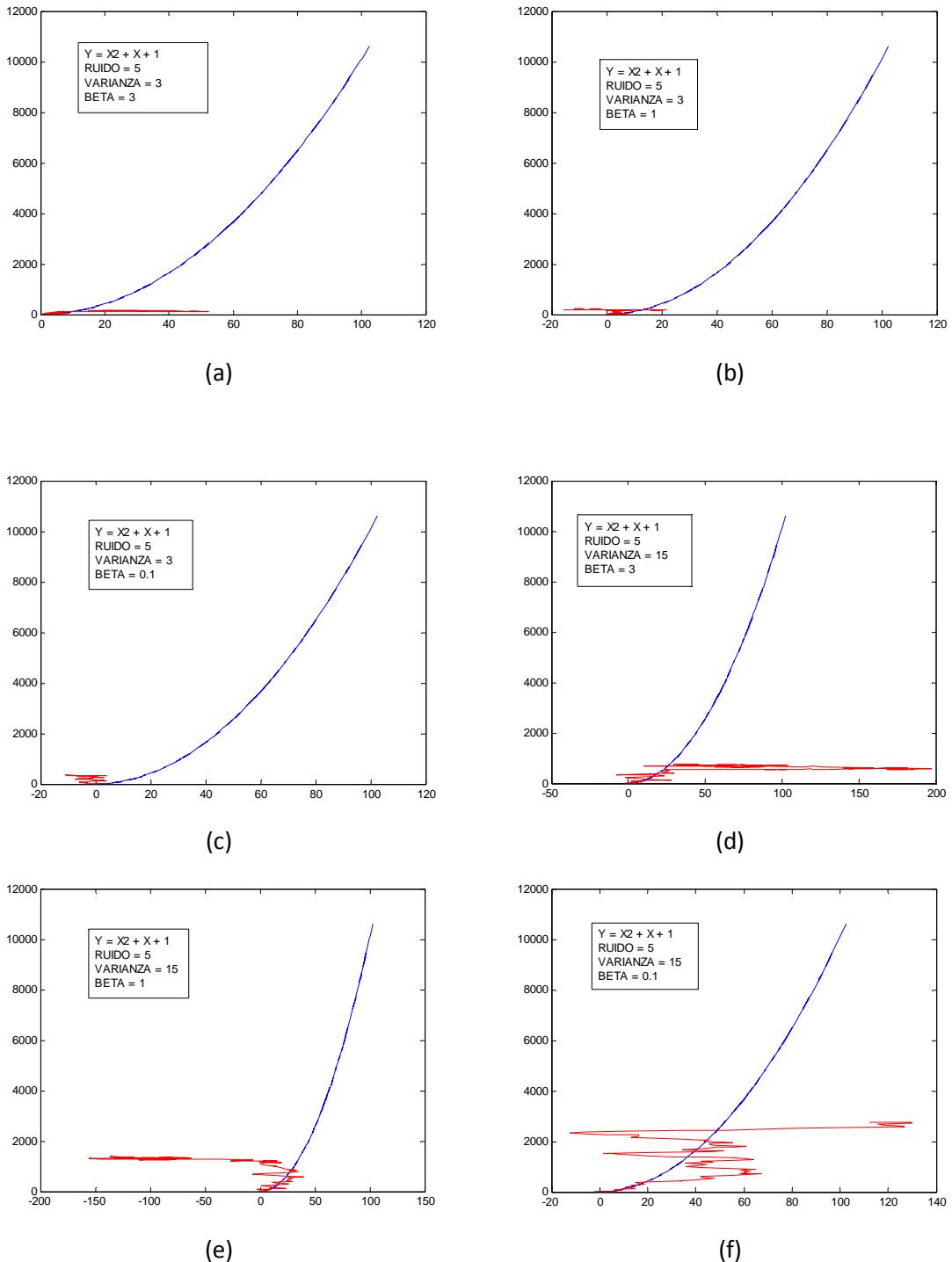


**Fig. A.74.- (a) Estimación con  $\beta = 5$ , sigue muy bien su forma sin perder los valores de sus picos. (b) Estimación con  $\beta = 0.1$ , se pierde totalmente al seguir la sinusoide, tanto los picos como su forma.**

Por un lado, podemos concluir como una  $\beta$  pequeña, de valor igual o inferior a 1, se pierde fácilmente cuando la recta presenta ruido y es incapaz de seguir una forma que está continuamente oscilando como puede ser una sinusoide. Esto es debido a que se toman demasiado en cuenta los valores de las partículas de menor peso lo que distorsiona la estimación obtenida. Por el otro, si el valor de  $\beta$  es grande, superior o igual a 5, el filtro de partículas hace un excelente seguimiento inicial pero se pierde al final del seguimiento. Al trabajar con un valor elevado de  $\beta$ , la estimación depende demasiado de las partículas de mayor peso, y por poco que éstas sean erróneas provoca que la estimación también lo sea. Así pues, un valor idóneo es aquél valor intermedio que evite que ambos efectos indeseados nos afecten en exceso, es decir, un valor:  $1 < \beta < 5$ . Se ha decidido tomar como valor de trabajo el valor de  $\beta = 3$ , debido a que en el caso del seguimiento de la recta, el resultado es mejor al obtenido con  $\beta = 5$  y este caso se asemeja más al real que en el de la sinusoide.

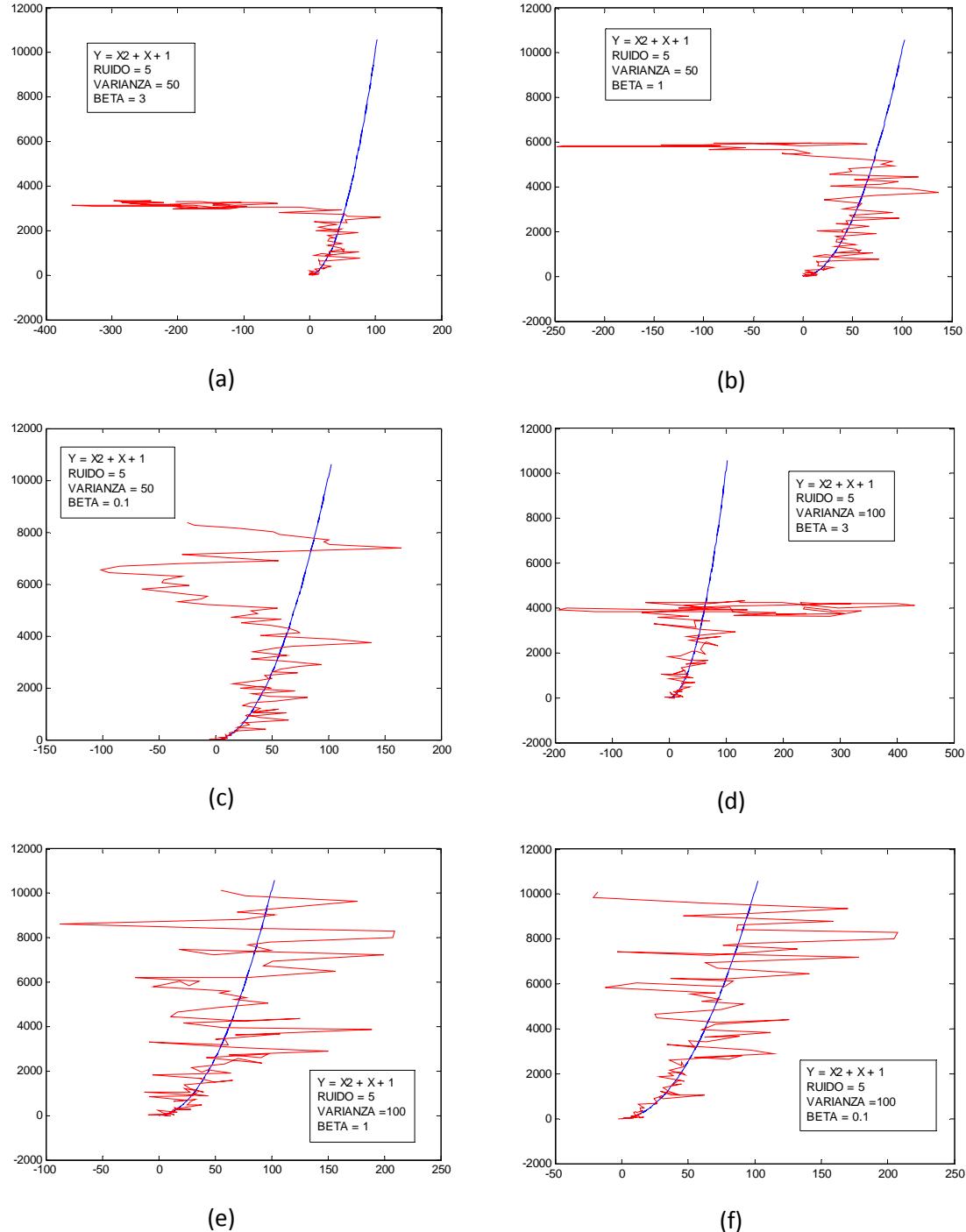
Todos los ejemplos anteriores están calculados con una varianza = 1, debido a que este parámetro no afecta demasiado en el ejemplo de la recta ya que una varianza elevada provocaría que tuviéramos un mayor número de partículas con un peso pequeño y menos partículas con un peso alto al alejarlas más cuando éstas fueran propagadas. En el ejemplo de la sinusoide sí que nos afectaría algo más porque si el valor fuese muy elevado podría llevar a que la estimación del filtro confundiera la oscilación que está siguiendo con la anterior o la siguiente, lo que provocaría malos resultados.

Principalmente, modificar el valor de la varianza nos permite estimar curvas o rectas que crezcan de manera muy rápida o presenten gran cantidad de ruido, lo que provoca que el filtro de partículas en seguida se pierda, es decir, no sea capaz de estimar adecuadamente la función. Por el contrario, si se trabaja con una varianza elevada, la nube de partículas es mayor y esto permite al filtro de partículas seguir saltos grandes en el seguimiento, lo que no sería posible con una varianza pequeña, y por lo tanto, se logra una mejor estimación de la función. Para poder apreciar este comportamiento estudiaremos el comportamiento de los filtros de partículas en el seguimiento de una parábola,  $Y = X^2 + X + 1$ , con distintas  $\beta$  y varianzas:



**Fig. A.75.-** (a) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 3$  y varianza = 3, se pierde totalmente al inicio. (b) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 1$  y varianza = 3, se sigue perdiendo al inicio. (c) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 0.1$  y varianza = 3, también se pierde al inicio. (d) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 3$  y varianza = 15, empieza bien la estimación pero en seguida se pierde. (e) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 1$  y varianza = 15, empieza algo mejor que la anterior estimación pero se sigue perdiendo. (f) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 0.1$  y varianza = 15, empieza bien la estimación y logra seguir algo más la curva que las estimaciones anteriores.

Se puede observar como si seguimos aumentando el valor de la varianza, cada vez se logra estimar mejor la parábola pese a tener mucho ruido en la estimación debido a la gran dispersión de las partículas fruto del elevado valor de ésta:



**Fig. A. 76.-** (a) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 3$  y varianza = 50, inicialmente sigue bien la parábola pero se pierde a mitad. (b) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 1$  y varianza = 50, similar a la anterior pero se pierde más tarde. (c) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 0.1$  y varianza = 50, mejora las dos anteriores pero sin llegar a estimarla completamente. (d) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 3$  y varianza = 100, se sigue perdiendo a mitad pese a llegar más arriba. (e) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 1$  y varianza = 100, llega hasta el final pese a tener cada vez más ruido. (f) Estimación de la recta  $y = x^2 + x + 1$  con una  $\beta = 0.1$  y varianza = 100, similar a la anterior pero con algo menos de ruido sobre todo al principio.

Tras estudiar el comportamiento de los filtros de partículas con distintos valores de la varianza, se puede determinar que si queremos estimar una función cuya variación inicial sea elevada, presente un salto importante, la mejor manera es tener en cuenta el mayor número de partículas y que éstas estén lo más dispersas posibles, de manera que aumentemos la probabilidad de que algunas realicen el seguimiento adecuadamente. Así pues, debemos trabajar con un valor de  $\beta$  bajo y uno de la varianza elevado, para lograr que el filtro se comporte como nos interesa.

El valor de la varianza con el que trabajaremos durante el trabajo será inicialmente de **3**. Esta elección se debe a que el seguimiento que realizaremos se asemeja al de una recta con ruido y a que el valor de este ruido, el que nos podría provocar grandes saltos en el seguimiento, es pequeño debido a la situación de la zona de inicialización de los filtros. En definitiva, el valor de la varianza vendrá determinado exclusivamente por el escenario de trabajo y de dónde esté situada la zona de inicialización siempre y cuando lo que deseemos estimar sea una recta. Es por ello que el valor de la varianza va variando conforme avanza la perspectiva y aumenta la necesidad de que las partículas se propaguen más lejos, tal y como se muestra en las tablas del anexo A.2.

Finalmente, podemos obtener las siguientes conclusiones de este trabajo previo:

1. El valor idóneo de  $\beta$  para un seguimiento rectilíneo con ruido está entre 1 y 5.
2. En general, un valor pequeño de  $\beta$  nos proporciona la estimación global de lo que se está siguiendo.
3. En general, un valor elevado de  $\beta$  nos proporciona una estimación precisa de lo que se está siguiendo.
4. Un valor elevado de la varianza sirve para estimar saltos bruscos que de otro modo no se alcanzarían a seguir.

## A.2.- Espacio de color YCbCr y generación de histogramas con YCb e YCr:

El espacio de color YCbCr que se emplea en la extracción de foreground, apartado 3.1.1, es el que normalmente se usa en sistemas de video y fotografía digital. Y es la componente de luminancia y Cb y Cr son las componentes de crominancia azul y roja, respectivamente. A veces el espacio YCbCr se abrevia como YCC.

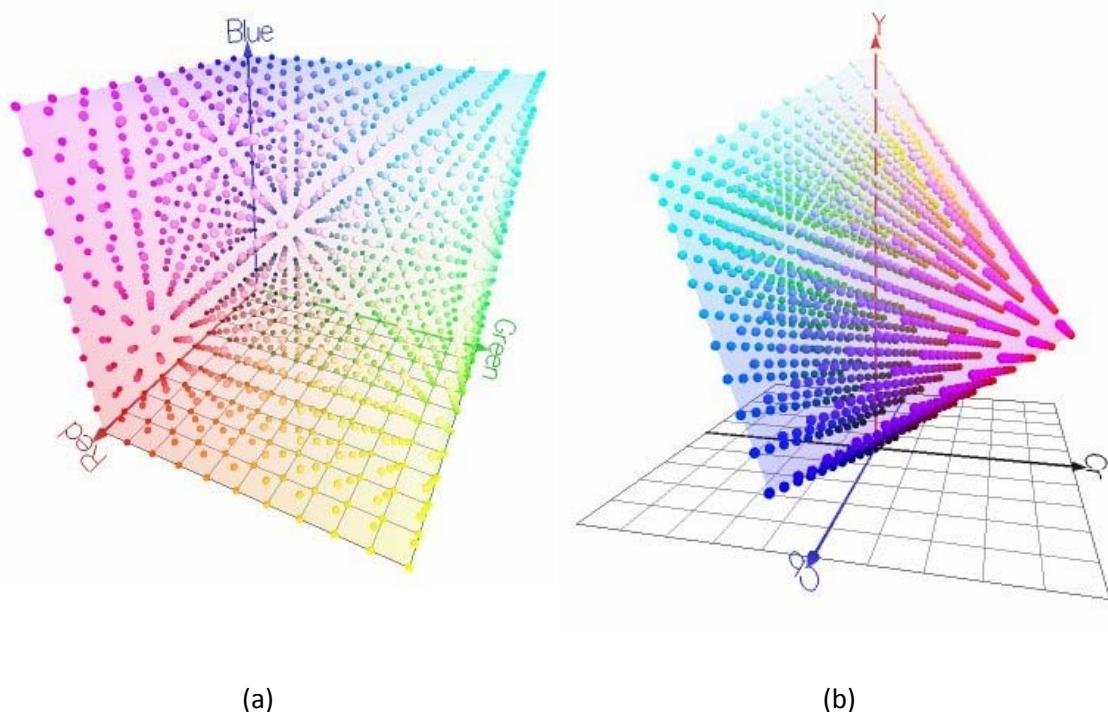
La conversión del espacio RGB a YCbCr se realiza a partir de las siguientes ecuaciones:

$$Y = 0.3 \cdot R + 0.59 \cdot G + 0.11 \cdot B \quad (1)$$

$$Cb = B - Y \quad (2)$$

$$Cr = R - Y \quad (3)$$

El cálculo de YCbCr a partir de estas ecuaciones supone una rotación y escalado del cubo RGB respecto al YCbCr, lo que conlleva que algunos puntos del cubo YCbCr no se puedan representar en el RGB (por lo menos no en el de rango nominal). Esto, a veces, puede causar dificultades en el momento de interpretar y exponer señales expresadas en YCbCr. Veamos una imagen que muestra la diferencia existente, a nivel de color y orientación, entre los cubos de los espacios RGB e YCbCr, ambos cuantificados de 0 a 1:



**Fig. A. 77.- Muestra de qué representan, a nivel de color, los cubos de color de ambos espacios. (a) El modelo RGB. (b) El modelo YCbCr.**

Por otro lado, en la construcción del histograma de color del objeto a seguir, apartado 3.1.2, se emplea el espacio de color YCb e YCr. La ventaja de trabajar con 2 canales en vez de con 3 es que reduce en una dimensión el cálculo del histograma. Para pasar de YCbCr a YCb-YCr se realizarán los siguientes pasos.

1. Partiremos de las componentes Cb y Cr cuyos valores van de [-255,255] que se han calculado a partir de las componentes RGB e Y, cuyos valores van de [0,255].
2. Reduciremos el rango de valores de Cb y Cr a [-128,128] mediante un *clipping*. El *clipping* consiste en asignar todos los valores por debajo de -128 a -128 y los que están por encima de 128 a 128. Esto se puede hacer ya que las imágenes con las que trabajamos presentan unos tonos generalmente grisáceos, lo que provoca que la gran mayoría de los valores de Cb y Cr estén próximos al 0.
3. Calcularemos las componentes YCb e YCr usando las siguientes ecuaciones:

$$YCb = Y \cdot Cb \quad (4)$$

$$YCr = Y \cdot Cr \quad (5)$$

4. Ahora el rango de valores de estas componentes puede llegar a ser [-32640, 32640], debido a que el valor máximo viene determinado por  $128 \cdot 255$ . Es necesario reducir el rango de valores, emplearemos un factor de normalización igual a 20 de manera que nos quedará [-1632, 1632].

$$YCb = \frac{YCb}{\text{Factor de normalización}} \quad (6)$$

$$YCr = \frac{YCr}{\text{Factor de normalización}} \quad (7)$$

5. Como la mayoría de los valores que presentan las imágenes no son colores puros, es decir, están lejos de los extremos, la mayoría de valores se encuentran en un rango de [-128,128], tras aplicar el factor de normalización. Por lo que, realizaremos un *clipping* para asegurarnos que estos valores se mantengan en el rango anterior.
6. Finalmente, incrementaremos el valor de ambas componentes en 128 para situar el rango de valores en [0,255].

De este modo, calcularemos los histogramas de color que tendrán dos dimensiones, YCb e YCr, que agruparemos en celdas cada 25 valores, de manera que el histograma calculado será de dimensiones 10x10.

### A.3.- Caracterización de los escenarios:

En primer lugar, cabe destacar que la nomenclatura empleada para etiquetar las secuencias y que se ha utilizado para referirse a éstas durante el presente trabajo es la siguiente:

**E[Número del escenario]\_[Número de la secuencia del escenario]**

El número del escenario corresponde a uno de los cuatro vistos en la Introducción. Mientras que el número de la secuencia del escenario se refiere simplemente al orden que se estableció durante su selección y posterior etiquetado.

En la Introducción se mencionó la **portabilidad** del algoritmo desarrollado, de su capacidad para poderse implementar en cualquier entorno que mostrara unas características similares a los que se emplean en este proyecto. Básicamente, éstas son: una cámara fija situada de frente a una rampa, que el seguimiento se desee realizar en la rampa descendente y que ésta presente un determinado tipo de perspectiva, similar a las de los escenarios con los que se ha trabajado. Para demostrar la portabilidad del algoritmo, si se desea implementar en un nuevo escenario, lo único que se debe hacer es realizar la caracterización del escenario que se explica a continuación y añadir el fichero de configuración de éste al algoritmo. De este modo, simplemente con indicar al algoritmo el fichero de configuración del escenario con el que debe trabajar y, por lo tanto el escenario de trabajo, el algoritmo es capaz de realizar el seguimiento y la detección de tumultos. Aún así, hay que tener presente que existen otras variables que puede que se deban modificar en función de cómo sean las características de este nuevo escenario y cómo éstas puedan afectar al propio algoritmo.

En resumen, para caracterizar debidamente un escenario deberemos determinar los siguientes parámetros:

- Tamaño del blob inicial
- Posición y tamaño de la zona de inicialización
- Posición y tamaño de la zona de destrucción
- Recta que modela la rampa en la que se realiza el seguimiento
- Recta de bloqueo entre las dos rampas
- Tabla completa de la relación entre el tamaño y la varianza de propagación del filtro de partículas en función de su posición respecto al eje Y.

Seguidamente se mostrarán estas características para los 4 escenarios con los que se ha trabajado en este proyecto. Algunas de estas características se han mostrado durante el presente escrito, pese a ello se ha considerado necesario volverlas a mostrar para mantener la estructura del anexo.

### **A.3.1.- Escenario 1:**

El escenario 1 se caracteriza por ser el más sencillo de los 4. La perspectiva que presenta es fácilmente modelable y las personas que se mueven por los alrededores de la rampa no dificultan el seguimiento, a excepción de las de la rampa ascendente.

#### **Tamaño del blob inicial:**



Fig. A. 78.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.

#### **Posición y tamaño de la zona de inicialización:**



Fig. A. 79.- Imagen que muestra el tamaño y la posición de la zona de inicialización.

*Posición y tamaño de la zona de destrucción:*

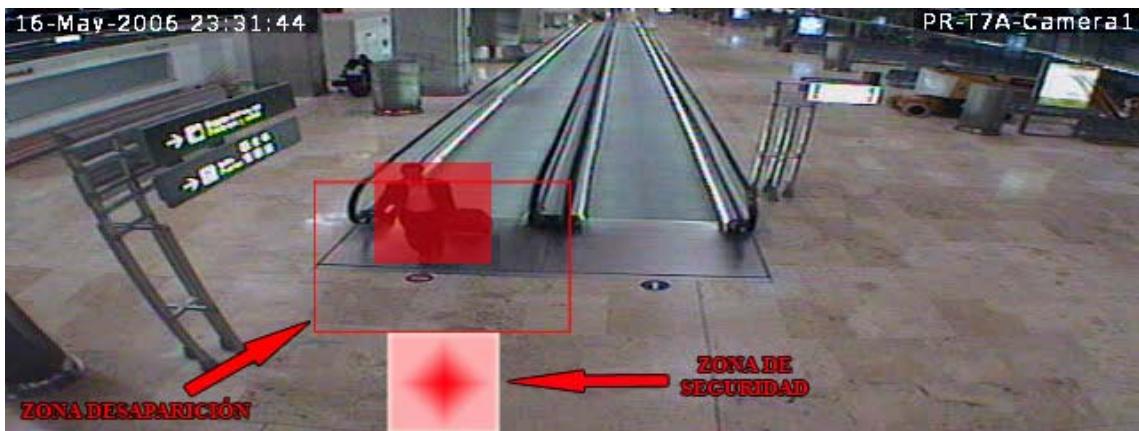


Fig. A. 80.- Imagen que muestra el tamaño y la posición de la zona de destrucción.

*Recta que modela la rampa en la que se realiza el seguimiento:*

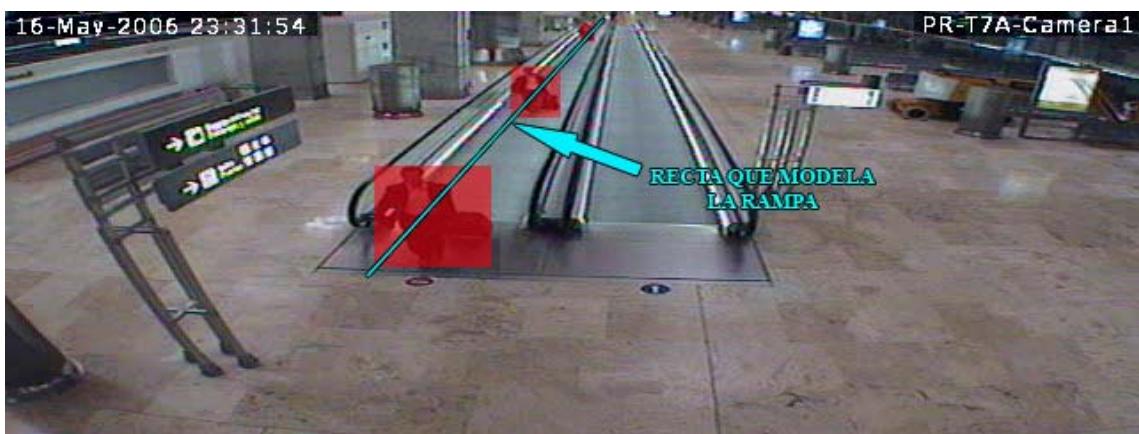


Fig. A. 81.- Imagen que muestra la recta que modela la rampa,  $Y = -1.083 \cdot X + 368$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.

*Recta de bloqueo entre las dos rampas:*



Fig. A. 82.- Imagen que muestra la recta que de bloqueo,  $Y = -2.96 \cdot X + 1021$ , empleada en este escenario.



### A.3.2.- Escenario 2:

El escenario 2 es el más complejo de los 4. Principalmente debido a la perspectiva en profundidad que presenta, que provoca que no sea fácil de modelar. Además existe un pivote entre las dos rampas que oculta una parte de la zona de inicialización. Las personas que se mueven por los alrededores de la rampa no dificultan el seguimiento, a excepción de las de la rampa ascendente. Una peculiaridad de este escenario, es que el reflejo que se obtiene en el cristal de la rampa descendente ayuda al seguimiento y permite solventar, en cierto modo, el problema del pivote.

#### Tamaño del blob inicial:



Fig. A. 83.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.

#### Posición y tamaño de la zona de inicialización:



Fig. A. 84.- Imagen que muestra el tamaño y la posición de la zona de inicialización.

*Posición y tamaño de la zona de destrucción:*



Fig. A. 85.- Imagen que muestra el tamaño y la posición de la zona de destrucción.

*Recta que modela la rampa en la que se realiza el seguimiento:*



Fig. A. 86.- Imagen que muestra la recta que modela la rampa,  $Y = -1.317 \cdot X + 450$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.

*Recta de bloqueo entre las dos rampas:*



Fig. A. 87.- Imagen que muestra la recta que de bloqueo,  $Y = -11.63 \cdot X + 3720$ , empleada en este escenario.



### A.3.3.- Escenario 3:

El escenario 3 corresponde al lado opuesto de la rampa del escenario 2. Al ser una rampa en subida, la perspectiva que presenta es sencilla y fácilmente modelable. En cuanto a las personas que se mueven por los alrededores de la rampa, éstas no dificultan el seguimiento, a excepción y como en el resto de casos, de las de la rampa ascendente.

#### Tamaño del blob inicial:



Fig. A. 88.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.

#### Posición y tamaño de la zona de inicialización:



Fig. A. 89.- Imagen que muestra el tamaño y la posición de la zona de inicialización.

*Posición y tamaño de la zona de destrucción:*

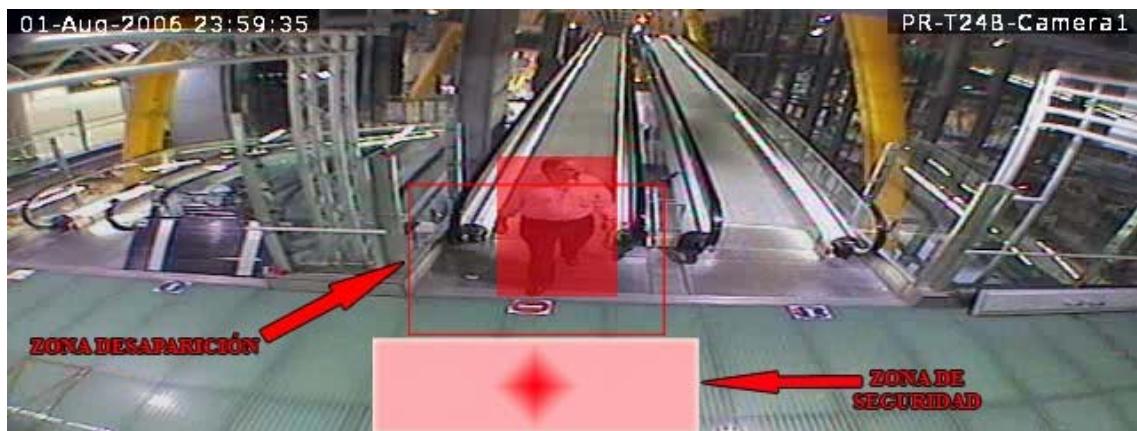


Fig. A. 90.- Imagen que muestra el tamaño y la posición de la zona de destrucción.

*Recta que modela la rampa en la que se realiza el seguimiento:*



Fig. A. 91.- Imagen que muestra la recta que modela la rampa,  $Y = -3.123 \cdot X + 1075$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.

*Recta de bloqueo entre las dos rampas:*



Fig. A. 92.- Imagen que muestra la recta que de bloqueo empleada,  $Y = 6.731 \cdot X + 2396$ , en este escenario.

**Tabla completa de la relación entre el tamaño y la varianza de propagación del filtro de partículas en función de su posición respecto al eje Y:**

X	Y	W	H	VAR
344	0	4	4	3
344	1	4	4	3
344	2	4	4	3
343	3	4	4	3
343	4	4	4	3
343	5	5	5	3
342	6	5	5	3
342	7	5	5	3
342	8	6	5	3
341	9	6	5	3
341	10	6	5	3
341	11	6	5	3
340	12	7	6	3
340	13	7	6	3
340	14	7	6	3
339	15	7	6	3
339	16	8	6	3
339	17	8	6	4
338	18	9	7	4
338	19	9	7	4
338	20	9	8	4
337	21	10	8	4
337	22	10	9	5
337	23	11	9	5
337	24	11	10	5
336	25	12	10	5
336	26	12	11	5
336	27	13	11	5
335	28	13	12	6
335	29	14	12	6
335	30	14	13	6
334	31	15	13	6
334	32	15	14	6
334	33	16	14	7
333	34	16	15	7
333	35	17	15	7
333	36	17	16	7
332	37	18	16	7
332	38	18	17	7
332	39	19	17	8
331	40	19	18	8
331	41	20	19	8
331	42	20	20	8
330	43	20	21	8
330	44	21	22	9
330	45	21	23	9
329	46	22	24	9
329	47	22	25	9
329	48	23	26	10
329	49	23	26	10
328	50	23	26	10
328	51	23	27	10
328	52	24	27	10
327	53	24	27	11
327	54	24	28	11
327	55	25	28	11
326	56	25	28	11
326	57	25	29	11
326	58	26	29	12
325	59	26	29	12
325	60	26	30	12
325	61	27	30	12
324	62	27	30	12
324	63	27	31	12
324	64	28	31	13
323	65	28	31	13
323	66	28	32	13
323	67	29	32	13
322	68	29	32	13
322	69	29	33	14
322	70	29	33	14
321	71	30	33	14
321	72	30	34	14
321	73	30	34	14
321	74	30	34	14
320	75	31	35	15
320	76	31	35	15
320	77	31	35	15
319	78	32	36	15
319	79	32	36	15
319	80	32	37	15
318	81	33	37	16
318	82	33	37	16
318	83	33	38	16
317	84	34	38	16
317	85	34	39	16
317	86	34	39	16
316	87	35	39	16
316	88	35	40	17
316	89	35	40	17
315	90	36	41	17
315	91	36	41	17
315	92	36	41	17
314	93	37	42	17
314	94	37	42	17
314	95	37	43	17
313	96	38	43	18
313	97	38	44	18
313	98	38	44	18
313	99	39	45	18
312	100	39	45	18
312	101	39	46	18
312	102	40	46	19
311	103	40	47	19
311	104	40	47	19
311	105	41	48	19
310	106	41	48	19
310	107	41	49	19
310	108	42	49	20
309	109	42	50	20
309	110	42	50	20
309	111	43	51	20
308	112	43	51	20
308	113	44	52	20
308	114	44	52	20
307	115	45	53	20
307	116	45	53	20
307	117	46	54	20
306	118	46	54	21
306	119	47	55	21
306	120	47	55	21
305	121	48	56	21
305	122	48	56	21
305	123	49	57	21
305	124	49	57	21
304	125	50	58	21
304	126	50	58	21
304	127	51	59	22
303	128	51	59	22
303	129	52	60	22
303	130	52	60	22
302	131	53	61	22
302	132	53	61	22
302	133	54	62	22
301	134	54	62	22
301	135	55	63	23
301	136	55	63	23
300	137	56	64	23
300	138	56	64	23
300	139	57	65	23
299	140	57	65	23
299	141	58	66	23
299	142	58	66	23
298	143	59	67	23
298	144	59	67	23
298	145	60	68	23
297	146	60	68	24
297	147	61	69	24
297	148	61	70	24
297	149	62	71	24
296	150	62	72	24
296	151	63	73	24
296	152	63	74	24
295	153	64	75	24
295	154	65	76	24
295	155	66	77	24
294	156	67	78	24
294	157	68	79	25
294	158	68	79	25
293	159	68	79	25

**Tabla A. 14.- Esta tabla muestra la evolución del tamaño del filtro de partículas en función de su posición según Y y la varianza empleada en la propagación de las partículas. Las celdas amarillas corresponden a los instantes de la Fig.91.**

#### **A.3.4.- Escenario 4:**

El escenario 4 corresponde al lado opuesto de la rampa del escenario 1. La perspectiva que presenta es igual que la del escenario 1 y es igualmente fácil de modelar. En este escenario las personas que se mueven por los alrededores de la rampa, en particular las del lado izquierdo, dificultan bastante el seguimiento, como también lo hacen las de la rampa ascendente.

##### **Tamaño del blob inicial:**



Fig. A. 93.- Imagen que muestra el tamaño mínimo de un blob necesario para inicializar un filtro.

##### **Posición y tamaño de la zona de inicialización:**



Fig. A. 94.- Imagen que muestra el tamaño y la posición de la zona de inicialización.

*Posición y tamaño de la zona de destrucción:*

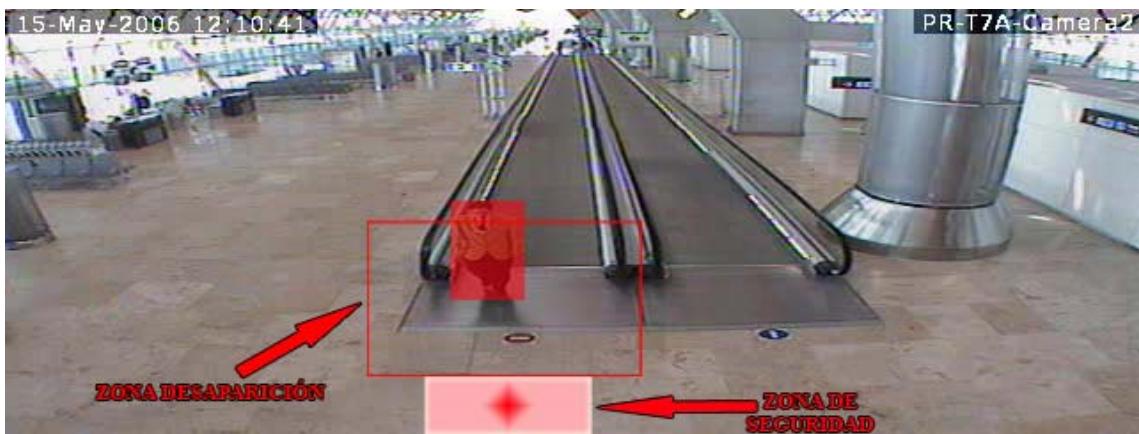


Fig. A. 95.- Imagen que muestra el tamaño y la posición de la zona de destrucción.

*Recta que modela la rampa en la que se realiza el seguimiento:*



Fig. A. 96.- Imagen que muestra la recta que modela la rampa,  $Y = -2.535 \cdot X + 817$ , y los 3 instantes tomados para la realización de la tabla que relaciona el tamaño y la varianza con la posición del filtro de partículas.

*Recta de bloqueo entre las dos rampas:*



Fig. A. 97.- Imagen que muestra la recta que de bloqueo,  $Y = 3.694 \cdot X - 1170$ , empleada en este escenario.



#### A.4.- Filtro Morfológico (Post-procesado):

Pese a que la fiabilidad de los algoritmos implementados es alta, en el análisis de las escenas aparecen píxeles, que se detectan como primer plano, no deseados. Estos píxeles deben ser eliminados en la mayor medida posible para evitar funcionamientos incorrectos. Para aumentar la fiabilidad de la detección de los algoritmos utilizados para la detección de *background* se ha utilizado morfología matemática. Al aplicar la detección de primer plano se genera una imagen binaria que modela la escena una vez se ha aplicado el algoritmo de detección de primer plano. En esta imagen los píxeles del fondo de escena se representan en negro y los píxeles de primer plano en blanco. Los filtros morfológicos utilizan operadores matemáticos (erosión y dilatación) sobre esta imagen binaria, a fin de mejorar los resultados de *foreground* obtenidos, eliminando componentes de pequeño tamaño que por sus dimensiones no pueden corresponder a objetos a detectar, sino a errores debidos al ruido o a fallos del modelo de fondo utilizado.

$X_3$	$X_2$	$X_1$
$X_4$	$X$	$X_0$
$X_5$	$X_6$	$X_7$

Fig. A. 98.- Matriz inicial y ejemplo de una máscara genérica para la morfología matemática.

La operación de erosión se utiliza para eliminar información superflua de la imagen binaria resultante de la detección de primer plano. Su aplicación consiste en una máscara booleana (ver ejemplo en Fig. 98), en la que sus elementos se inicializan a “1”. Esta máscara se solapa con todos los píxeles de la imagen binaria. Se centra la máscara en el píxel a tratar ( $X$ ), si este ha sido detectado como primer plano, se marcará como píxel de *background* si algún píxel solapado con la máscara ha sido detectado como fondo. Asimismo, en la dilatación, se marcarán como píxeles de primer plano los píxeles centrados en la máscara en los que algún píxel solapado ha sido detectado como *foreground*. La combinación de los operadores erosión y dilatación se conoce como **Apertura**, y su formulación matemática se expone a continuación:

$$y = x \ominus b \oplus b \quad (8)$$

donde  $b$  es el elemento estructurante;  $\ominus$  representa la operación de erosión;  $\oplus$  representa la operación de dilatación;  $x$  es el píxel solapado con la máscara de primer plano. **El tamaño empleado del elemento estructurante será de 5x5 pixels**, ya que un tamaño superior a éste podría empezar a eliminar zonas de *foreground* que nos pueden resultar útiles para el seguimiento. Así pues, con este valor se consigue eliminar el suficiente ruido sin destruir demasiada información útil. Además, hay que tener presente que la apertura empleada será de conectividad 4, en la Fig. A. 99 se muestran los distintos tipos de conectividad y como estos afectan el elemento estructurante.

**4-Conectividad****6-Conectividad****8-Conectividad**

	$X_2$	
$X_3$	$X_0$	$X_1$
	$X_4$	

	$X_3$	$X_2$
$X_4$	$X_0$	$X_1$
$X_5$	$X_6$	

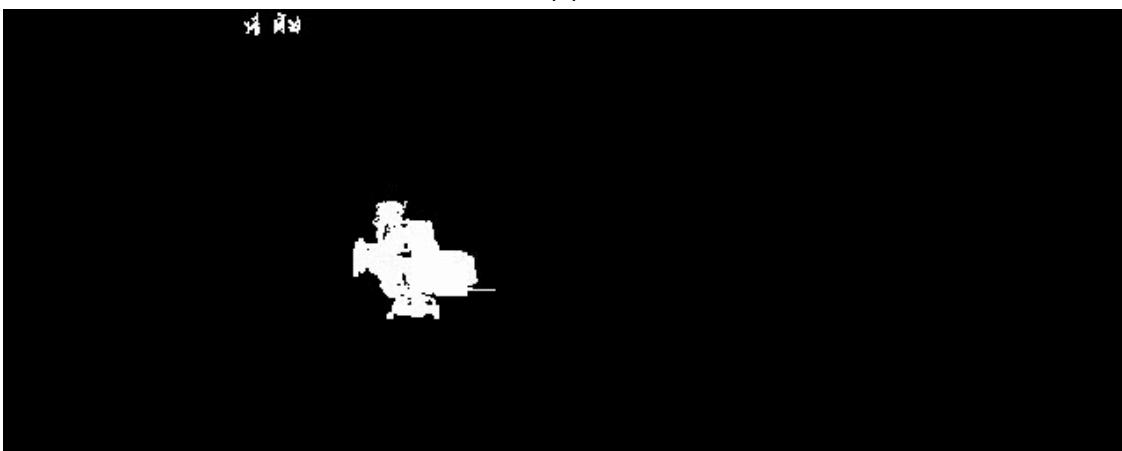
$X_4$	$X_3$	$X_2$
$X_5$	$X_0$	$X_1$
$X_6$	$X_7$	$X_8$

Fig. A. 99.- Ejemplo de los 3 tipos distintos conectividad: conectividad 4, la empleada, conectividad 6 y conectividad 8.

La Fig. 100 muestra el resultado de aplicar una apertura a una de las imágenes procesadas. Como puede observarse en la primera imagen existe una gran cantidad de ruido que podría dar lugar a errores en la detección de primer plano. La imagen a la que se ha aplicado la apertura morfológica presenta una mejora sustancial respecto a la máscara de primer plano sin modificar. Esto facilita la detección de los objetos, y evita problemas en su seguimiento.



(a)



(b)

Fig. A. 100.- (a) Extracción de foreground sin tratamiento de post-procesado. (b) Extracción de foreground con tratamiento de post-procesado de tamaño de 25 píxeles y con conectividad 4.

A.5.- Resultados obtenidos tras el cálculo de la tasa de falsas alarmas en las 100 secuencias:

Nº secuencia	E1	E2	E3	E4
1	0%	0%	0%	0%
2	0%	0%	0%	0%
3	0%	0%	0%	0%
4	0%	100%	0%	0%
5	0%	100%	100%	0%
6	0%	100%	0%	0%
7	0%	0%	0%	0%
8	0%	100%	0%	
9	0%	0%	0%	
10	0%	0%	0%	
11	0%	0%	0%	
12	0%	0%	0%	
13	0%	0%	0%	
14	100%	100%	0%	
15	0%	100%	0%	
16	0%	100%	0%	
17	0%	100%	0%	
18	0%	100%	0%	
19	0%	0%	100%	
20	0%	0%	0%	
21	0%	100%	0%	
22		100%	0%	
23		0%	100%	
24		100%	0%	
25		100%	100%	
26		0%	0%	
27		100%	100%	
28		0%	0%	
29		0%	100%	
30		0%	0%	
31		100%		
32		100%		
33		100%		
34		100%		
35		100%		
36		100%		
37		100%		
38		100%		
39		0%		
40		100%		
41		100%		
42		100%		

Tabla A. 16.- Resultados de la tasa de falsas alarmas obtenido para cada secuencia para una resolución de 250x50.