Detección de interacciones persona-objeto en secuencias de vídeo-seguridad no basada en seguimiento de objetos

Informe Técnico

SemanticVideo.TR.2010.01

Julio 2010

Rafael Martin Nieto

Juan Carlos San Miguel

José M. Martínez

Índice del documento

Resumen

En este informe se presenta un sistema de deteccion de interacciones persona-objeto en secuencias de video. Primeramente, se ha realizado un estudio del sistema base existente en el VPULab y se ha identificado la extracción de regiones estáticas como la etapa más critica de análisis observando su limitacion de precision en entornos altamente poblados. Posteriormente se ha integrado un algoritmo de deteccion de regiones estaticas que no presenta dichas limitaciones en los escenarios considerados debido a que no esta basado en seguimiento de objetos

En concreto, se trabaja sobre sistemas de video-seguridad que detectan los eventos de robo, abandono, coger y dejar objetos. La comparación de los resultados obtenidos con el sistema base y el nuevo sistema demuestra que se han mejorado las capacidades del análisis, mejorando principalmente su precisión (reduciendo la tasa de falsas alarmas).

1. Introducción

En la actualidad, la demanda de sistemas de video-vigilancia está aumentando, especialmente para su aplicación en lugares públicos con gran afluencia de gente. La gran cantidad información que se debe procesar provoca que las personas encargadas de vigilar las imágenes necesiten ayuda adicional para mejorar su eficiencia y resultar efectivos. Dicha ayuda adicional, se puede conseguir mediante sistemas que se encarguen de aportar alguna ayuda, llegando incluso a ser capaces de actuar como sistemas autónomos capaces de detectar ciertos eventos predefinidos y encargándose los propios programas de tomar las medidas pertinentes.

La creación de estos sistemas resulta compleja. El procedimiento habitual es crear un sistema "base" que comienza con la idea general y que cumple en parte los objetivos con los que ha sido creado, y mediante revisiones posteriores, se mejora su eficiencia progresivamente. En este último paso es donde se enmarca el trabajo de este informe.

Tras obtener una idea, a partir de la documentación y evaluaciones previas, sobre cómo mejorar el sistema, se diseña e implementa el nuevo algoritmo y se integra en el sistema, evaluando la mejora introducida en los resultados del sistema, de forma que se puede saber si la modificación del sistema ha sido útil o no, cualitativa y cuantitativamente.

La estructura de este informe es: La sección 2 describe el estado del arte relacionado, la seccion 3 describe el sistema base, la sección 4 describe el nuevo algoritmo integrado en el sistema para detectar regiones estáticas, la sección 5 muestra el conjunto de videos escogidos (content set) y sus características, en la sección 6 se muestran los resultados experimentales y finalmente la sección 7 presenta unas breves conclusiones del trabajo realizado.

2. Estado del arte

Los sistemas de video-vigilancia tratan de detectar los movimientos y emplazamientos de elementos que aparecen en un video en tiempo real. El objetivo principal de estos sistemas es llegar a una interpretación automática de escenas y comprender y predecir las acciones e interacciones de los
objetos observados a partir de la información adquirida por sensores.

La evolución de los sistemas de video vigilancia se suele dividir en tres generaciones[1] (sistemas CCTV, sistemas automáticos de video-vigilancia basados en CCTV y sistemas automáticos de video-vigilancia distribuidos). Los sistemas automáticos utilizan un diseño modulara para realizar las tareas asignadas[1]. Por ejemplo, para la detección de objetos se usan la "diferencia temporal" y la "substracción de fondo", para el reconocimiento y seguimiento de objetos se usan modelos en 2D (Con o sin modelos de explícitos de formas) y modelos en 3D, para el análisis de comportamientos se usan técnicas como el Dynamic

Tyme Warping (DTW), modelos dinámicos de redes probabilísticas como HMM (hidden Markov models) y redes Bayesianas.

Cuando se trata de detectar objetos abandonados o robados en una secuencia de vídeo-seguridad el primer problema consiste en detectar la región donde puede haberse producido el evento[2]. En este tipo de aplicaciones, la segmentacion fondo-frente es una etapa crítica. Los principales métodos utilizados para la segmentación fondo-frente se pueden clasificar en modelos paramétricos, entre los que destacan el método de la Gaussiana simple, el modelo de mezcla de Gaussianas (MoG) y el segmentador Gamma, y modelos no paramétricos, donde destaca el método basado en estimar la densidad del núcleo (KDE), o métodos basados en modelos ocultos de Markov (HMM). Actualmente, el modelo más utilizado para caracterizar los píxeles del fondo de la escena es el método de mezcla de Gaussianas (MoG) debido a que considera diversos factores como que afectan al fondo como posibles cambios de iluminación en la imagen, fondos multimodales, objetos moviéndose lentamente, y el ruido introducido por la cámara.

Posteriormente, la extraccion de regiones estaticas se realiza mediante el uso de una etapa de seguimiento de blobs y contadores de estacionariedad de blobs (en base a su velocidad). La principal desventaja de este tipo de aproximaciones es la dificultad de realizar dicho análisis en entornos complejos y que no se puede distinguir entre abandoned object o stolen object.

Recientemente, gran parte de la investigación en aplicaciones de vídeo-vigilancia se ha centrado en detectar regiones estáticas a partir de segmentacion fondo-frente, y tratar de realizar una clasificación correcta entre personas y objetos.

Con respecto a los eventos en los videos, rara vez aparecen en frames aislados del video, lo que provoca que un evento cualquiera se suela definir con unas características en un determinado espacio y tiempo. En [3], se presentan las características principales que permiten determinar los eventos tras la extracción de blobs: la silueta del blob, los patrones de movimiento, los modelos de persona, etc. También se ha propuesto un enfoque multi cámara que combina modelos humanos en 3D, velocidad de blobs e información contextual.

3. Sistema base

En la figura 2 se presenta un esquema de la estructura del sistema base [4]. Tras la adquisición de los distintos frames, la segmentación de movimiento está formada por el módulo de detección del frente. A continuación, el módulo de extracción de blobs analiza las regiones conexas de la máscara binaria para generar los distintos blobs detectados en dicho frame, cuyas trayectorias serán generadas por el módulo de seguimiento de blobs. Una vez se han obtenido los blobs y sus trayectorias, se extrae información de cada blob con el objetivo de realizar una clasificación entre dos clases: humanos y no humanos. Después, todos los datos generados se usan en el módulo de detección de eventos y acciones.

El sistema está diseñado para funcionar en tiempo real, como parte de un sistema de video-seguridad, lo que implica que la complejidad computacional de los algoritmos no debe ser alta.

Con respecto a la cámara, ésta debe ser estática y sin control automático de ganancia (AGC).

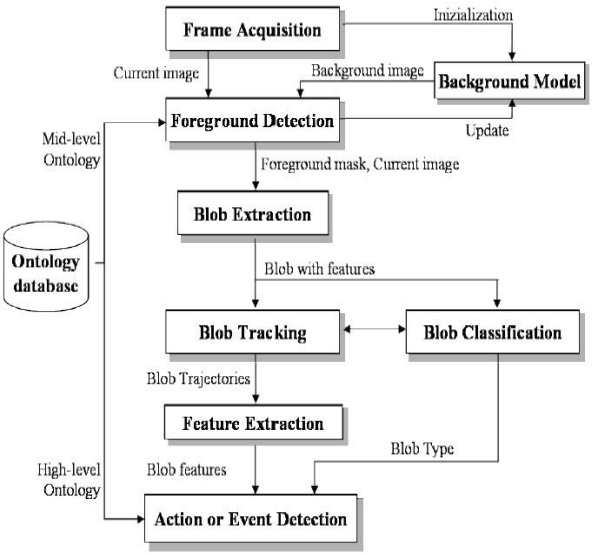

Figura 2: Sistema de video análisis

3.1 Adquisición de frames

El proceso de adquisición de frames consiste en obtener los frames (De una cámara o previamente almacenados), almacenarlos en la memoria y tomar frame a frame para analizarlos secuencialmente. El sistema trabaja con frames sin comprimir con valores RGB para cada pixel.

3.2 Detección de frente

La detección del primer plano (Foreground detection) se basa en el algoritmo de substracción de fondo. El módulo se compone de cuatro pasos: El módulo de detección de cambios, la eliminación de ruidos, la eliminación de sombras y de nuevo otra eliminación de ruidos. [4]

3.3 Extracción de blobs

El módulo de extracción de blobs se compone de dos etapas: la detección inicial de blobs, que extrae las componentes unidas mediante un análisis de componentes conexas de doble pasada, y el post procesado, que elimina las pequeñas regiones no deseadas mediante un posterior filtrado [4].

3.4 Seguimiento de blobs

El módulo de seguimiento de blobs busca las correspondencias entre los blobs detectados en los diferentes frames. Para ello se utilizan reglas basadas en color, tamaño y distancia entre los blobs detectados en frames consecutivos.

3.5 Clasificación de blobs

El objetivo del módulo de clasificación de blobs es identificar los diferentes objetos que pueden aparecer en la secuencia de video, pudiendo clasificar entre personas y objetos. Para ello se utilizan tres detectores basados en tamaño, forma y localización de la cabeza [4].

3.6 Detección de eventos

El módulo de detección de eventos se compone de dos etapas. Primero, el módulo calcula algunas propiedades necesarias para detectar los eventos. A continuación se analiza dicha información para detectar los eventos. Los eventos detectados pueden ser simples(Get object, Put object) o complejos(Stolen object, Abandoned object).

4. Integración de nuevo Algoritmo: detección de regiones estáticas

El algoritmo introducido en el sistema base se encarga de detectar regiones estáticas. Para ello, se analizan características de la señal de vídeo en diferentes instantes de tiempo (sub-muestreo). Primeramente se analiza la persistencia de una región de foreground en distintos instantes temporales. Posteriormente, se analiza el movimiento de dichas regiones en los instantes seleccionados en el muestreo anterior. El objetivo de este análisis es la eliminación de regiones de foreground siempre activas donde claramente se observa que existe movimiento y no puede existir un objeto de interés para los eventos a detectar (e.g., zonas de paso donde siempre existe foreground activo pero a su vez esta en movimiento).

El resultado de ambos análisis son dos máscaras binarias indicando las propiedades deseadas. El resultado del submuestreo de mascaras de foreground y su posterior combinación lógica (utilizando una operación AND) nos permite obtener una máscara binaria S formada por regiones estáticas a partir de un único modelo de fondo.

Para analizar el movimiento de las regiones estáticas se utiliza la técnica de “frame difference (FD)”, que consiste en restar dos imágenes entre sí, dando como resultado un estimador de movimiento dentro de dicha imagen.

4.1. Parametrización del sistema

Como parámetro principal del nuevo sistema con el algoritmo previo integrado, se considera el parámetro que define el tiempo que debe transcurrir para que un blob se considere estático.

Las dos primeras categorías del Ground Truth están compuestas de videos de poca duración, que varían desde unos segundos hasta menos de dos minutos donde los principales eventos de interés son PutObject y GetObject. Para estos casos el criterio para considerar blobs estáticos es que no se mueva en 2 segundos.

Por otro lado, para las categorías 3 y 4, que tienen una duración mayor y los principales eventos de interes son AbandonedObject y StolenObject, se ha establecido un tiempo de 20 segundos para que los blobs se consideren estáticos.

5. Content Set

Los videos del Content Set se clasifican en distintas categorías según la complejidad de los mismos. Estas categorías estarán relacionadas con las tareas a realizar por el sistema, y si bien pueden definirse algunas categorías genéricas (esto es, que aparecerán para diversos algoritmos), como la complejidad del fondo, los detalles concretos que definen a cada categoría variarán de algoritmo a algoritmo.

5.1 Clasificación de Características

En nuestro caso, los videos del Content Set se clasifican según dos aspectos relacionados con las tareas a realizar por los algoritmos de detección de eventos de interacción persona-objeto a evaluar:

- **Dificultad para extraer los objetos del fondo**: se define como la dificultad para extraer el movimiento y los objetos estacionarios en un escenario. Se relaciona, por ejemplo, con el número de los objetos, su velocidad, oclusión parcial o cambios de iluminación.

- **Complejidad del fondo**: se define como la presencia de bordes, texturas múltiples y objetos pertenecientes al fondo, como árboles moviéndose, la superficie del agua, etc.

Con estos criterios, los videos escogidos para evaluar el sistema se clasifican en las siguientes categorías:

	Complejidad	
Categoría	Extracción de fondo	Complejidad de fondo
C1	Baja	Media
C2	Baja	Alta
C3	Media	Alta
C4	Alta	Media

Tabla 1: Categorías y complejidad

Dado el tipo de eventos seleccionados, se escogió un Content Set con vídeos de distintas características y se clasificó en distintas categorias. Los videos se seleccionaron de los siguientes datasets públicos:

• PETS2006 [5]

• PETS2007 [6]

• i-LIDS dataset for AVSS2007 [7]

• VISOR [8]

• ITEA CANDELA project [9]

En la figura 1, se muestra una secuencia de cada categoría de videos, en los que se aprecia principalmente la complejidad del fondo, que coincide con la categorización de los vídeos, según lo definido en la tabla 1.

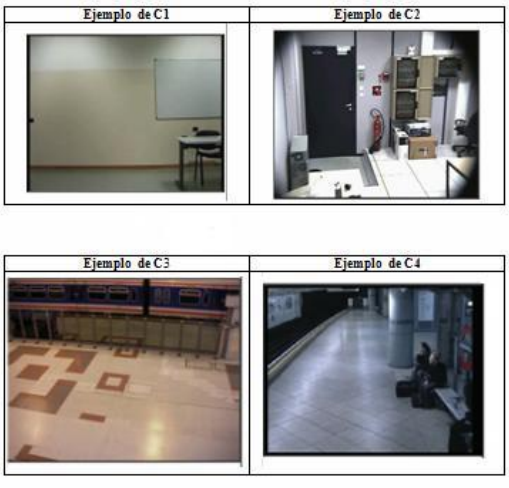

Figura 1: Ejemplo de secuencias de cada categoría

5.2. Ground Truth asociado al Content Set

Para evaluar la precision/recall del sistema base y el mejorado, se ha analizado la detección de 4 eventos:

- **PutObject**: Un Sujeto se desprende de un objeto.
- **AbandonedObject**: Tras desprenderse del objeto (esto es, producirse el evento PutObject), el sujeto se aleja del objeto, considerándose el objeto desatendido.
- **GetObject**: Un sujeto coge un objeto.
- **StolenObject**: Tras coger un objeto (esto es, producirse el evento GetObject), el sujeto abandona con el objeto el lugar bajo vigilancia.

5.3 Criterios de Anotación

En el Ground Truth de los videos, se han utilizado los siguientes criterios de anotación:

- **PutObject**: se anota el evento cuando el individuo se separa del objeto. Se tiene que ver la separación entre el objeto y el individuo.
- **AbandonedObject**: se anota el evento cuando tras un PutObject, el sujeto se aleja una distancia de aproximadamente 3 pasos o cuando el individuo sale de la imagen.
- **GetObject**: se anota el evento cuando se aprecia claramente que un objeto existente deja de estar en su posición original. La persona puede estar enfrente o detrás del objeto. al cogerlo, por lo que en algunos casos no se anota hasta que la persona se aparta (momento en el que se "ve" el evento que puede haber ocurrido antes).
- **StolenObject**: se anota el evento cuando tras un GetObject, la persona que coge el objeto sale de la imagen con dicho objeto.

5.4 Ventana de Evaluación

Respecto al intervalo de "ocurrencia" del evento, se decidió utilizar dos intervalos distintos para los eventos simples (PutObject y GetObject) y complejos (AbandonedObject y StolenObject), debido a que el algoritmo que se pretende insertar está diseñado para mejorar la detección de los eventos largos o complejos. Adicionalmente, se incluyeron dos longitudes de anotación para los intervalos de ocurrencia de los eventos simples y complejos. El objetivo es estudiar la dependencia de la evaluación de los resultados obtenidos con las longitudes de anotación. Los Se pueden diferenciar dos tipos de longitudes:

- **Longitud Corta**: Se corresponde con los frames en los que una persona que visualice el video podría considerar que se produce el evento.
- **Longitud Larga**: Se corresponde con los frames de la longitud corta, añadiendo a continuación un intervalo de frames en los que podría aparecer el evento detectado, debido al retardo computacional en el proceso de análisis del algoritmo. Se puede considerar como una longitud corta “extendida”.

Por ello, se tomó el siguiente criterio de longitudes:

Eventos	Longitud Corta (en frames)	Longitud Larga (en frames)
Sencillos	50	300
Complejos	100	500

Tabla 2: Criterio de intervalos de ocurrencia de eventos

6. Resultados experimentales

A la hora de obtener los resultados, existen diversas medidas (e.g., detecciones correctas, falsos positivos, falsos negativos), entre las cuales destacan las conocidas como Precision y Recall:

- **Precisión**: Cociente entre el número de eventos detectados correctamente y el número de candidatos que el algoritmo ha considerado como posibles eventos. Da una idea relativa de los falsos positivos (falsas alarmas) que produciría el algoritmo.
- **Recall**: Cociente entre el número de eventos detectados correctamente y el número de eventos a detectar. Da una idea relativa de la capacidad del algoritmo para detectar los eventos (alarmas perdidas).

La figura 3 ilustra de manera más clara las medidas mencionadas.

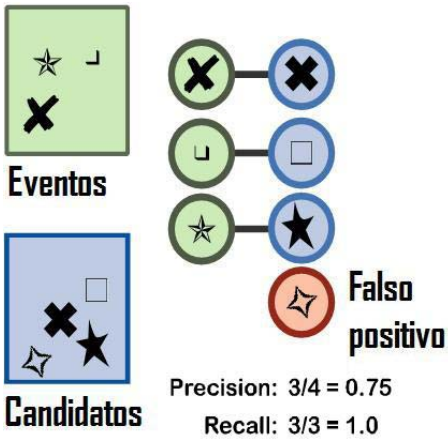

Figura 3: Ejemplo de Precision y Recall

Se ha elegido el blob tracking para mostrar ejemplos de funcionamiento ya que resulta más ilustrativo En la figura 4 se pueden ver algunos ejemplos de buen funcionamiento. En contraposición, en la figura 5 se pueden ver ejemplos de mal funcionamiento.

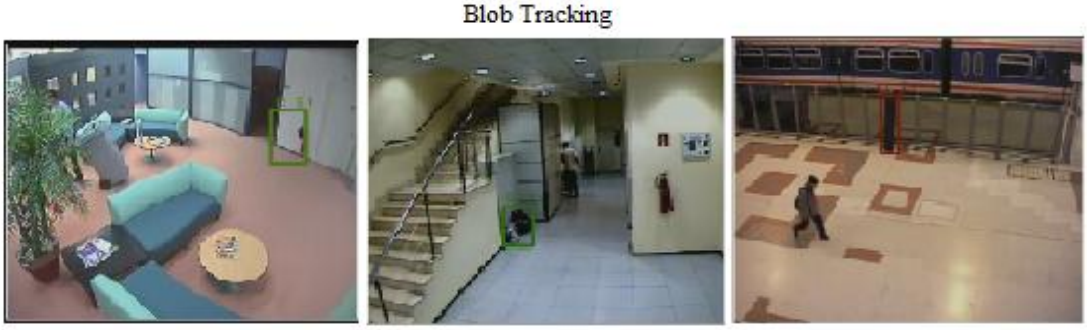

Figura 4: Ejemplos de buen funcionamiento.

Figura 5: Ejemplos de mal funcionamiento.

En las dos primeras secuencias de la figura 5, se ve como en zonas que no resultan de interés (la vía del tren, el tren, la puerta...), al modificarse, provocan falsos blobs estáticos que no resultan de interés y que pueden provocar falsos positivos. En la tercera secuencia se muestra el mal funcionamiento que se produce cuando las personas se quedan quietas en la zona grabada durante un cierto tiempo.

También cabe destacar que la sombra de la maleta provoca que su bounding box aumente de tamaño.

	Eventos cortos				Eventos largos				Todos los eventos			
	Precisión		Recall		Precisión		Recall		Precisión		Recall	
	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora
C1	100	80	68,96	30,77	0	77.78	0	77.78	52,63	78,95	66,66	42,86
C2	36,11	31,58	22,03	9,09	10.34	20	8.57	10.72	24,61	26,47	17,02	9,57
C3	20,38	56,25	7,16	13,43	21.60	41.67	12.87	83.33	20,92	50	8,97	19,18
C4	30,68	14,29	1,08	1,32	48.28	11.11	3.64	16.67	37,67	13,04	1,68	1,91

Tabla 3: Resultados para los eventos con longitud corta

	Eventos cortos				Eventos largos				Todos los eventos			
	Precisión		Recall		Precisión		Recall		Precisión		Recall	
	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora	Antes	Ahora
C1	100	80	68,96	30,77	0	77.78	0	77.78	52,63	78,95	66,66	42,85
C2	36,11	36,84	25,42	10,61	10.34	20	8.57	10.71	27,69	29,41	19,14	10,64
C3	25,24	56,25	8,87	13,43	24.69	41.67	14.71	83.33	25	50	10,72	19,18
C4	43,18	21,42	1,52	1,99	53.45	11.11	4.03	16.67	47,26	17,39	2,11	2,55

Tabla 4: Resultados para los eventos con longitud larga

7. Conclusiones

En este informe, se ha presentado un sistema para la deteccion de interacciones persona-objeto en secuencias de video y la integracion de un nuevo algoritmo para detectar regiones estáticas que permite mejorar la fiabilidad del sistema.

Como se puede observar en los resultados obtenidos, el algoritmo introducido en el sistema consigue su objetivo, mejorando los resultados frente a las capacidades detectoras del sistema sin modificar, pero solo para los eventos largos, es decir, para abandoned object y stolen object.

Frente a esta mejora, al observar los resultados para el conjunto de los 4 eventos y según la categoría, se mejoran o empeoran los resultados. Para los eventos cortos, put object y get object, los resultados son más negativos, empeorando en la mayoría de los casos.

Con estas conclusiones, cabría plantearse el considerar dos sistemas diferentes. El sistema antiguo pasaría a centrarse en detectar únicamente los eventos put object y get object, y el nuevo sistema se encargaría de detectar los eventos largos, abandoned object y stolen object. Cada uno se podría seguir desarrollando y mejorando con el método desarrollado en este informe, con el fin de mejorar las prestaciones a un nivel mayor.

Como trabajo futuro, se propone plantear la posibilidad de crear un sistema que sea capaz de conmutar entre las diferentes duraciones de los eventos, de forma automática tras obtener ciertos resultados no adecuados, como por ejemplo al observar los múltiples blobs estáticos de pequeño tamaño que se comentaron en la sección de resultados experimentales.

8. Referencias

[1] M. Valera and S.A. Velastin: Intelligent distributed surveillance systems: a review, Abril 2005.
[2] Alvaro Bayona, Juan C. SanMiguel, Jose M. Martinez: "Stationary foreground detection using background subtraction and temporal difference in video surveillance", en IEEE ICIP 2010, Hong Kong, China,, pp. 4657-4660
[3] Juan C. SanMiguel, Marcos Escudero, Jose M. Martinez and Jesus Bescos, “Real-time event detection in smart rooms", Technical Report.
[4] Juan Carlos San Miguel Avedillo:”Ontology-guided semantic video descriptions using feedback between signal processing stages”, Proyecto Fin de Master, Universidad Autónoma de Madrid, Noviembre 2008.
[5] PETS2006: Ninth IEEE International Workshop on Performance Evaluations of Tracking and Surveillance, June 2006. URL http://pets2006.net/.
[6] PETS2007: Tenth IEEE International Workshop on Performance Evaluations of Tracking and Surveillance, October 2007. URL http://pets2007.net/.
[7] i-LIDS dataset for AVSS2007: Fourth IEEE International Conference on Advanced Video and Signal based Surveillance, September 2007. URL:http://www.elec.qmul.ac.uk/sta_nfo/andrea/avss2007_d.html.
[8] VISOR: Video Surveillance Online Repository.URL: http://imagelab.ing.unimore.it/visor.
[9] ITEA CANDELA project: scenarios for abandoned object detection. URL: .