UNIVERSIDAD DE MÁLAGA
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

TESIS DOCTORAL

SEGMENTACIÓN ESPACIO-TEMPORAL DE IMÁGENES MEDIANTE ESTRUCTURAS JERÁRQUICAS DE ENLACE ADAPTATIVO

AUTOR: Juan Antonio Rodríguez Fernández
Ingeniero de Telecomunicación

Málaga, 2001

MENU SALIR

D. PELEGRIN CAMACHO LOZANO, PROFESOR DEL DEPARTAMENTO DE TECNOLOGÍA ELECTRÓNICA DE LA UNIVERSIDAD DE MÁLAGA

y

D. FRANCISCO SANDOVAL HERNÁNDEZ, CATEDRÁTICO DEL DEPARTAMENTO DE TECNOLOGÍA ELECTRÓNICA DE LA UNIVERSIDAD DE MÁLAGA

CERTIFICAMOS:

Que D. Juan Antonio Rodríguez Fernández, Ingeniero de Telecomunicación, ha realizado en el Departamento de Tecnología Electrónica de la Universidad de Málaga, bajo nuestra dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulada:

"SEGMENTACIÓN ESPACIO-TEMPORAL DE IMÁGENES MEDIANTE ESTRUCTURAS JERÁRQUICAS DE ENLACE ADAPTATIVO"

Revisado el presente trabajo, estimamos que puede ser presentado al Tribunal que ha de juzgarlo.

Y para que conste a efectos de lo establecido en el Real Decreto 778/1998 regulador de los estudios de Tercer Ciclo-Doctorado, AUTORIZAMOS la presentación de esta Tesis en la Universidad de Málaga.

Málaga, 14 de Febrero de 2001

Fdo. Pelegrín Camacho Lozano
Profesor de Tecnología Electrónica

Fdo. Francisco Sandoval Hernández
Catedrático de Tecnología Electrónica

MENU SALIR

Departamento de Tecnología Electrónica
E. T. S. I. Telecomunicación
Universidad de Málaga

TESIS DOCTORAL

SEGMENTACIÓN ESPACIO-TEMPORAL DE IMÁGENES MEDIANTE ESTRUCTURAS JERÁRQUICAS DE ENLACE ADAPTATIVO

AUTOR: Juan Antonio Rodríguez Fernández
Ingeniero de Telecomunicación

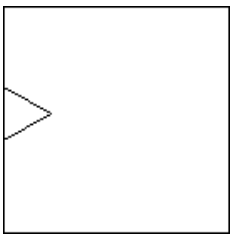
DIRECTORES:

Pelegrín Camacho Lozano
Dr. Ingeniero de Telecomunicación

Francisco Sandoval Hernández
Dr. Ingeniero de Telecomunicación

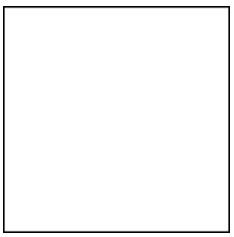
MENU SALIR

13 de febrero
¿Qué hay detrás de la ventana?



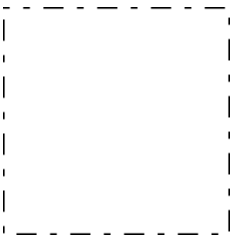
Una estrella

14 de febrero
¿Qué hay detrás de la ventana?



Una sábana extendida

15 de febrero
¿Qué hay detrás de la ventana?



(*Los detectives salvajes*, **Roberto Bolaño**, 1999)

MENU SALIR

Agradecimientos

Cuando uno aborda la ingente tarea de desarrollar una tesis, contar con el apoyo y la desinteresada ayuda necesarios puede ser la clave para convertir una difícil hazaña en un apasionante reto. Son muchas las personas que han colaborado en mayor o menor medida en esta tarea. Por ello, deseo manifestar mi gratitud más sincera a:

Marme, mi mujer, la más directa víctima de este trabajo, por saber soportarme en mis peores momentos, y animarme siempre que lo he necesitado. A ella le debo gran parte del trabajo de corrección, tan duro y a la vez tan importante.

Cristina Urdiales y Antonio Bandera, siempre incansables ante cualquier adversidad, cuyas colaboraciones han sido fundamentales en el desarrollo y redacción de esta tesis.

Pelegrín Camacho, codirector de esta tesis, aportando toda su experiencia, y prestando todo su apoyo siempre que lo he necesitado.

Francisco Sandoval, codirector y responsable del grupo de investigación en cuyo seno ha sido posible desarrollar este trabajo.

Fabián Arrebola, quién inicialmente, junto con Pelegrín Camacho, introdujo el procesamiento mediante estructuras multirresolución de fóvea desplazable y todas sus variantes.

Todos aquellos, Tonín, Piti, Migui, MariCarmen, Antonio, Bárbara, Jesús, Mariú, Matías, Maricruz, Carlos, Susana, Victor, Inma, Salvador, Rocío y Paco, que en mayor o menor medida han aportado su compañía, trabajo, consejos, experiencia o ayuda, consiguiendo que, finalmente, este proyecto llegara a buen puerto.

Todos los compañeros del Departamento de Tecnología Electrónica.

Mi familia, por darme la oportunidad de ser lo que soy.

La Comisión Interministerial de Ciencia y Tecnología (CICYT) que, a través de los proyectos TIC95-0589 y TIC98-0562, han soportado parcialmente la financiación de esta tesis.

MENU SALIR

Resumen

Esta tesis presenta una nueva técnica de segmentación espacio-temporal de imágenes para entornos reales. Los métodos existentes funcionan sólo bajo unas condiciones muy restrictivas que impiden su aplicación a un amplio conjunto de situaciones reales. El objetivo es conseguir un sistema tan resistente como sea posible que funcione para un conjunto amplio de diferentes entornos, sin imponer excesivas restricciones *a priori*. Además, el sistema debe funcionar en un ordenador personal sin necesidad de hardware específico, por lo que resulta deseable desarrollar un método cuya carga computacional asociada esté acotada por un valor razonable.

Inicialmente, se probaron varias técnicas de segmentación a partir de una única imagen, modificándolas, caso de ser necesario, para trabajar en entornos reales. Sin embargo, la información que contiene una sola imagen no es suficiente para obtener una separación de las partes que la componen. Por ello, como segundo paso, se optó por trabajar con técnicas espaciales que aportaban estabilidad a la segmentación al buscar regiones que mantenían coherencia en el espacio a lo largo de una secuencia de vídeo. Aunque los resultados mejoraron notablemente, estos métodos resultaron aún ineficaces para los casos más complejos. Finalmente, se optó por las técnicas espacio-temporales, que demostraron ser las apropiadas para secuencias reales.

Desafortunadamente, la mayor parte de las técnicas espacio-temporales existentes presentaron importantes deficiencias al ser aplicadas sobre secuencias capturadas en condiciones de trabajo reales no controladas. Esto ocurría porque dichos métodos parten de una serie de limitaciones que no se suelen cumplir bajo estas condiciones. Algunos algoritmos ofrecieron resultados aceptables a pesar de todo, pero su complejidad, traducida en un elevado tiempo de proceso, resultó excesiva para aplicaciones en tiempo real.

El método propuesto se ha desarrollado para evitar estos problemas. Para empezar, no depende de ninguna restricción y trabaja de forma jerárquica para mantener acotado el tiempo de proceso. Consiste en estabilizar adaptativamente estructuras piramidales construidas sobre fotogramas consecutivos de una secuencia para conseguir una segmentación consistente a lo largo de ésta. Cuando la estabilización ha concluido, cada nodo de una estructura cualquiera está enlazado a una región homogénea de píxeles pertenecientes al fotograma empleado para construirla, pero también a la misma región en fotogramas precedentes. Así, las regiones no sólo son coherentes en el espacio, sino también en el tiempo.

Para probar su eficiencia, la técnica desarrollada se ha incluido como parte de un sistema atencional de transmisión de vídeo, donde se controla el flujo de datos a fin de mantener una tasa constante de imágenes por segundo. El sistema se basa en la transmisión de imágenes foveales, donde sólo las áreas relevantes de la escena se presentan con una alta resolución. El método de segmentación espacio-temporal propuesto localiza dichas áreas, y la conversión de las imágenes al formato foveal realiza la compresión de la escena, de forma que el volumen total de datos transmitidos se reduce drásticamente, sin perder nivel de detalle en las regiones de interés. Si la compresión aportada no es suficiente, se aplica un criterio de envío selectivo de las distintas regiones de la escena, en función de las condiciones de retardo existentes en ese momento.

MENU SALIR

Abstract

This thesis presents a new spatiotemporal segmentation technique for real scenes. The existing methods work only under specific constrains, which make their application difficult. The goal is to achieve a system that works for a wide variety of scenarios with no constrains *a priori*. Besides, the system must run in a personal computer with no specificic hardware and the computational load must be kept as reduced as possible.

Initially, several segmentation techniques based on a single image have been tested and modified to work with real scenes. However, an unique image does not yield enough information to segment it into meaningful regions. Hence, spatial techniques relying on searching for coherent regions in a sequence of images were tested. Although results were clearly improved, these segmentation methods were still unefficient in complex cases. Finally, spatiotemporal techniques proved to be suitable for real sequences.

However, most implemented spatiotemporal techniques yielded errors when tested with sequences captured under uncontrolled conditions. This occurs because these methods are based on several constraints which are typically false under these conditions. Some algorithms provided acceptable results, but their processing time was too high.

The proposed method has been designed to avoid these problems. First, it is not based on any constraint and it works in a hierarchical way to keep a bounded processing time. The method consists of adaptively stabilizing pyramidal structures built over consecutive frames to achieve consistent segmentation results through the whole sequence. After stabilization is performed, each node of any given structure is related to an homogeneous region at its frame, but also to the same homogeneous region in the previous ones. Thus, regions are coherent in space, but also in time.

To prove its efficiency, the proposed segmentation method has been implemented as part of an attentional video transmission where data flow is controlled to keep a stable frame rate. The system relies on transmitting foveal images where only relevant areas of the scene are presented at a high resolution. The spatio-temporal segmentation method is used to locate those areas, and the compression is achieved by the foveal conversion, so that the total data volume transmitted is drastically reduced. If the compression factor succeeded is not enough, current delay conditions fix the number of rings around the foveae not to be transmitted, in order to keep a constant frame rate at reception.

MENU SALIR

Índice General

MENU SALIR

Lista de Símbolos y Acrónimos

ACFR	Centro australiano de robots de campo, del inglés *Australian Centre for Field Robotics*.
α	Parámetro estimado empíricamente que determina la velocidad del proceso de olvido exponencial.
B	Constante positiva que determina el rango de variación del umbral T en la detección de movimiento adaptativo en el espacio.
Bd	Número de subanillos dentro de cada anillo debajo de la fóvea en una geometría foveal.
$B(x, y, t)$	Fondo estimado en la posición (x, y) en el instante t.
bounding box	Caja rectangular que cubre la posición que ocupa una región determinada en un imagen.
block maching	Método de detección y estimación de movimiento mediante el estudio de la correlación de bloques de la imagen de tamaño fijo.
C	Constante positiva que determina el rango de variación del umbral empleado en la detección de movimiento adaptativo en el espacio.
CBR	*Constant Bit Rate*, velocidad constante de transferencia de secuencias de vídeo, espresada en *bits* por segundo.
CCD	Dispositivo de carga acoplada, del inglés *Charge-Coupled Device*.
${}^{t}C_l(j)$	Nodo hijo ubicado en el nivel l de la pirámide t.
${}^{t+1}C_l(j)$	Nodo hijo ubicado en el nivel l de la pirámide $t+1$.
COLAMSG	Cola de mensajes establecida entre *segment* y *servidor* con la que ambos procesos se sincronizan e intercambian información.
Clase	Entidad que representa un conjunto de píxeles de una imagen que comparten una o más características.
cliente	Proceso que recibe los paquetes enviado por el proceso *servidor* y construye la imagen multifoveal en recepción.
*controltiempo*1	Proceso que gestiona desde el *PC servidor* la medición del retardo del canal.

CMOS	*Complementary Metal – Oxide Semiconductor*, semiconductor de *metal-óxido* complementario.
CPU	*Central Processor Unit*, unidad central de proceso.
d	Factor de subdivisión o número de subanillos dentro de cada anillo.
DCT	Transformación reversible lineal, del inglés *Discrete Cosine Transform*.
d_h	Número de subanillos de cada anillo en el sentido horizontal.
d_v	Número de subanillos de cada anillo en el sentido vertical.
${}^t d_l(j)$	Desplazamiento estimado para el nodo ${}^t C_l(j)$ entre los instantes de tiempo correspondientes a las pirámides $t-1$ y t.
DSP	Procesador digital de señal, del inglés *Digital Signal Processor*.
FC	Factor de compresión que presentan las imágenes cartesiano-exponenciales.
FFT	Transformada rápida de Fourier, del inglés *Fourier Fast Transform*.
fóvea	Área de máxima resolución que presentan las geometrías de resolución variable.
GMC	Método de estimación de movimiento con modelo 2D que trabaja con la imagen global.
GMFD	Geometría Multirresolución de Fóvea Desplazable.
GMFD-MG	Geometría Multirresolución de Fóvea Desplazable con Movimiento Generalizado.
GTK	Librería gráfica de libre distribución para Linux, del inglés *Gimp ToolKit*.
HECFG44	Modelo de *frame grabber* empleado en la adquisión de imágenes.
Φ	Valor de umbral utilizado por el método de segmentación por umbralización.
i_{max}, j_{max}	Valor máximo de las coordenadas de los píxeles que componen una región.
i_{min}, j_{min}	Valor mínimo de las coordenadas de los píxeles que componen una región.
IPC	Protocolo Unix de comunicación entre procesos residentes en una misma máquina, del inglés *Inter-Process Communication*.
ISA	Bus de conexión de periféricos del ordenador personal PC, del inglés *Industry Standard Architecture*.
I_t	Derivada del valor de la intensidad luminosa.
$I(x,y)$	Nivel de gris del píxel que ocupa la posición (x,y) dentro de una imagen.
$I(x,y,t)$	Nivel de gris del píxel que ocupa la posición (x,y) en una imagen adquirida en el instante t.

$I_{dif}(x, y, t)$	Valor diferencia o máscara de movimiento de los píxeles que ocupan la posición (x, y) en los fotogramas capturados en los instantes t y $t-1$.
KP-D50	Modelo de cámara del fabricante Hitachi.
l	Nivel arbitrario de una estructura de datos 3D.
LAN	Red de área local, del inglés *Local Area Network*.
Ld	Número de subanillos dentro de cada anillo a la izquierda de la fóvea en una geometría foveal.
m	Número de anillos de resolución alrededor de la fóvea en una geometría foveal.
MAP	Criterio bayesiano, del inglés *maximum a posteriori*.
MEMCOMP1	Segmento de memoria compartida en el que el proceso *segment* almacena la imagen de resolución uniforme para su posterior procesamiento.
MEMCOMP2	Segmento de memoria compartida en el cual el proceso *controltiempo*1 almacena el retardo estimado del canal.
MHz	Megahercios.
MMX	Conjunto de instrucciones de un microprocesador específicos para el procesado de estructuras multimedia como imagen y sonido, del inglés *MultiMedia eXtension*.
modo	Esquema de transmisión elegido para cada estructura foveal, que determina el número de anillos a enviar en cada caso.
MPEG	Estándar de compresión de video introducido por el grupo de expertos de imagen y movimiento *(Motion Picture Experts Group)* y asumido como norma por las organizaciones de estandarización internacionales ISO e IEC.
ms	Milisegundo.
$M(x, y, t)$	Matriz binaria que presenta ceros en todas aquellas zonas donde no se ha detectado movimiento, llamada máscara de movimiento.
N	Número de imágenes que componen una ventana temporal.
n_c	Valor del punto i de la máscara de movimiento en el instante $t-1$.
$N.fov$	Número de estructuras multirresolución de las que consta un paquete de imágenes.
NTSC	Estándar americano de codificación de video, del inglés *National Television System Committee*.
OCR	Reconocedor óptico de carecteres, del inglés *Optical Character Recognition*.
Pirámide	Estructura jerárquica multinivel para el procesamiento de imágenes.
$\Psi_{t-1,t}$	Valor de la correlación normalizada entre dos imágenes.

PAL Estándar europeo de codificación de video, del inglés *Phase Alternating Line*.

pan-tilt-vergencia Los tres grados de libertad básicos de una estructura de soporte de dos camaras.

PC Ordenador personal, del inglés *Personal Computer*.

PCI Bus de conexión de periféricos del ordenador personal PC, del inglés *Peripheral Component Interconnect*.

POSIX Estándar de comunicaciones que incorpora el sistema operativo Linux.

RAM Memoria de acceso aleatorio, del inglés *Random Access Memory*.

Rd Número de subanillos dentro de cada anillo a la derecha de la fóvea en una geometría foveal.

rexel Celda de los niveles superiores en una estructura jerárquica de datos.

RGB Los tres campos con los que se codifica el color en señales de vídeo, a saber, rojo, verde y azul, del inglés *Red Green Blue*.

ROIs Región de interés, del inglés *Region Of Interest*.

segment Proceso que realiza la segmentación espacio-temporal de la escena y localiza las regiones de interés (ROIs) de la misma, determinando las coordenadas de las *bounding-boxes* que las contienen.

S_h Factor de desplazamiento horizontal de cada anillo de resolución con respecto a una fóvea centrada en una estructura GMFD.

SH Vector que contiene los factores de desplazamiento horizontal para cada anillo de resolución con respecto a una fóvea centrada en una estructura GMFD-MG.

S.O. Sistema Operativo.

socket Estructura empleada en las comunicaciones del protocolo TCP/IP.

servidor Proceso que recibe del proceso *segment* información sobre las posiciones que ocupan las ROIs y una imagen de resolución uniforme; gracias al proceso *controltiempo*1 dispone del valor del retardo del canal; con esta información construye un paquete y lo envía al proceso *cliente*, que reside en el PC Cliente.

S_v Factor de desplazamiento vertical de cada anillo de resolución con respecto a una fóvea centrada en una estructura GMFD.

SV Vector que contiene los factores de desplazamiento vertical para cada anillo de resolución con respecto a una fóvea centrada en una estructura GMFD-MG.

SW Software.

Sx	Dimensión horizontal de una imagen original.
$s(x, y)$	Valor del píxel que ocupa la posición (x, y) en una imagen segmentada mediante el método de umbralización.
Sy	Dimensión vertical de una imagen original.
t	Instante de tiempo.
T	Valor del umbral en la detección de movimiento adaptativo en el espacio.
TCP1	Socket abierto entre *servidor* (origen) y *cliente* (destino).
TCP2	Socket abierto entre *cliente* (origen) y *servidor* (destino).
TCP3	Socket abierto entre *controltiempo*1 (origen) y *controltiempo*2 (destino).
TCP4	Socket abierto entre *controltiempo*2 (origen) y *controltiempo*1 (destino).
TCP/IP	Protocolo de interconexión de ordenadores, del inglés *Transmission Control Protocol / Iternet Protocol*.
Td	Número de subanillos dentro de cada anillo encima de la fóvea en una geometría foveal.
$TimeStamp$	Sello de tiempo que sirve para identificar el momento en que se crea una estructura de datos determinada.
TMS320C44	Modelo de DSP del fabricante *Texas Instruments*.
$t(n_i)$	Valor del umbral para cada punto n_i del detector de movimiento adaptativo en el espacio.
U	Umbral heurístico que fija la sensibilidad de la máscara de movimiento de la escena.
U_{dif}	Umbral que determina si la diferencia entre píxeles es significativa.
USB	Bus de conexión de periféricos serie universal, del inglés *Universal Serial Bus*.
VAD	Valor Absoluto de la Diferencia entre fotogramas consecutivos.
$\vee$	Vector que expresa la velocidad de desplazamiento de cualquier píxel de una imagen 2D.
VLSI	Escala de integración muy alta en el proceso de fabricación de circuitos integrados, del inglés *Very Large Scale of Integration*.
VME	Bus de interconexión de procesadores con capacidad de procesamiento en tiempo real introducido en 1981 por Motorola, en inglés *VersaModule Eurocards*.
W	Anchura del campo de visión en píxeles.
$Waist$	Nivel de la estructura jerárquica foveal que presenta todo el campo de visión y sirve de base para la construcción de una pirámide superior.

MENU SALIR

Índice de Figuras

Índice de Tablas

MENU SALIR

Capítulo 1

Introducción

1 La visión artificial

La visión artificial es un proceso consistente en extraer información del entorno a partir de una o más imágenes de dicho entorno, usando para ello alguna técnica basada en modelos computacionales. Cualquier conjunto de datos que pueda recogerse en un formato visualizable para el ser humano puede acogerse al concepto de imagen. Algunos de los aspectos más interesantes de esta disciplina son la enorme diversidad de aplicaciones basadas en el procesamiento de imágenes y la pluralidad de ramas del conocimiento susceptibles de utilizarla, que van desde la meteorología, oceanografía o astronomía hasta la robótica o la vigilancia.

La visión artificial surge de combinar sensores de captación de imagen con técnicas de inteligencia artificial a efectos de comprender y procesar una escena concreta. La inteligencia artificial puede definirse en función de dos componentes:

- Como ciencia de lo natural, una disciplina cuyo objetivo es entender la inteligencia humana. A este respecto, es una ciencia del análisis y modelado de la naturaleza de los sistemas inteligentes.

- Como ciencia de lo artificial, es la disciplina que busca la creación de sistemas inteligentes con técnicas computacionales. Pretende crear formulaciones y modelos sobre un soporte físico concreto capaces de ofrecer comportamientos inteligentes.

En un primer momento, la visión artificial se concibió como una imitación de la visión humana. Se asumía que sería sencillo adaptar los mecanismos de ésta a una máquina que, a fin de cuentas, era capaz de efectuar cualquier operación matemática de forma mucho más

rápida. No pasó mucho tiempo antes de que esta iniciativa fracasara: aun se desconocen aspectos fundamentales del proceso visual humano y, en cualquier caso, resulta demasiado complejo para ser emulado. Según Turing, una máquina podrá ejecutar cualquier cálculo sólo cuando alguien pueda explicarlo en su lengua natural y sin ambigüedad.

1.1 El sistema de visión humano

El proceso visual humano se ha dividido tradicionalmente en cuatro etapas: i) la formación de la imagen: la luz penetra en el ojo, alcanza la retina y se fusiona para crear la imagen retinal; ii) codificación: la imagen se codifica en el camino que discurre entre la retina y el córtex visual; iii) representación: la imagen codificada se procesa mediante operaciones simples tales como detección, discriminación o reconocimiento básico; y iv) interpretación: el cerebro asocia propiedades perceptivas con sensaciones como son el color, el movimiento y la forma.

La Fig. 1.1 muestra la estructura anatómica de un ojo humano. La luz visible, en la escala de los 380 a 825 nanómetros de longitud de onda, alcanza la retina, una capa formada principalmente por tres tipos de células. El primer tipo de células, las fotorreceptoras, se divide a su vez en dos clases: los conos, asociados a la luz diurna y la percepción del color, y los bastones, asociados a la visión en la oscuridad. El segundo tipo está conectado a los axones de los fotorreceptores y actúa como interfaz con el tercer y último tipo, las células ganglionares, que a su vez se dividen en células M y P según criterios fisiológicos y funcionales. El resto de células de la retina -horizontales y amacrinas- cumple funciones de conexión con las anteriores. En el centro de la retina existe una zona denominada fóvea que contiene una alta densidad de conos distribuidos de forma radial y que presenta un punto ciego allí donde se inserta el nervio óptico.

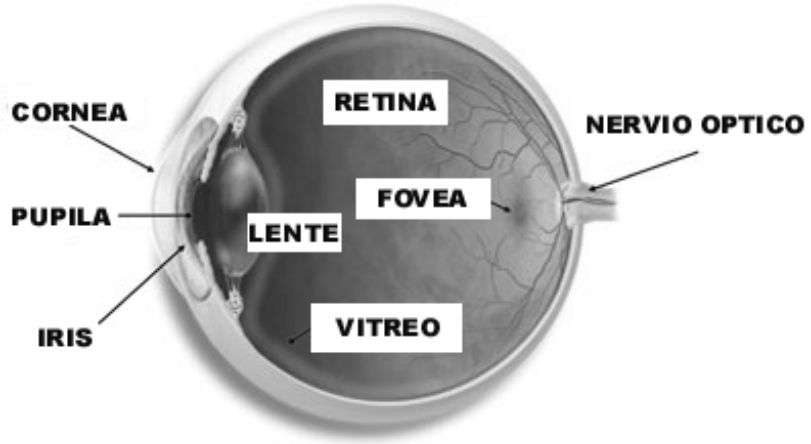


Figura 1.1: Estructura del ojo humano

A pesar de la existencia de un punto ciego y de la multitud de capilares que riegan la retina, el observador tiene la impresión de observar una escena contínua [Grossberg y Kuperstein 1986].

Las dos principales vías de visión en el cerebro se forman a partir de la proyección de la imagen hacia el córtex visual (vía geniculoestriculada) y al colículo superior (vía retinorectal). La visión consciente recae sobre la primera vía que, de ser dañada, conduce a la ceguera parcial o total salvo para la localización inconsciente de objetos. La segunda vía es particularmente importante en las tareas de atención y movimiento de los ojos. El córtex visual se divide en seis capas de células monoculares y binoculares de cuatro tipos: centro-periferia, simples, complejas e hipercomplejas. Estas células ya presentan un cierto grado de especialización, como las simples, que son sensibles a líneas orientadas, o las hipercomplejas, que responden a curvas y esquinas. Actualmente, los esquemas que emplea el cerebro para procesar la información proveniente del córtex siguen en fase de estudio: su complejidad sólo ha permitido desentramar una mínima parte de su funcionamiento.

1.2 Psicología de la percepción

Si bien el proceso de percepción visual en el ser humano no puede explicarse de forma satisfactoria, es posible analizarlo y, en cierta forma, preverlo a partir de la experiencia. Así, es posible tomar el conocimiento extraído de estos procesos como base para el diseño de un sistema de visión artificial, incluso a pesar de que no se conoce a nivel funcional la forma en que se ejecutan. Dada la enorme complejidad que entraña el procesamiento de una imagen desde el punto de vista de la neurología, la psicología ha tomado el relevo en este asunto. A efectos de distinguir el proceso puramente físico de captura de una imagen cualquiera de su posterior procesamiento, cabe definir el concepto de sensación como 'la respuesta inmediata que se produce en el cerebro causada por la excitación del ojo'. Este proceso es más sencillo que la percepción, que se puede definir como 'el resultado de fundir sensaciones a efectos de crear una representación mental útil del entorno'. La percepción engloba procesos de organización, descarte e interpretación de sensaciones y se puede entender como el proceso de creación de un significado a partir de un grupo de entradas sensoriales.

Tradicionalmente, el proceso visual se explica partiendo de dos enfoques muy diferentes: el asociacionista y la escuela de la forma o *gestalt* [Kohler 1947]. La psicología asociacionista postula que la imagen percibida por el individuo no es más que la suma de las imágenes de los distintos objetos que componen el campo de visión; mientras que la escuela de la forma afirma que el total es algo más que la suma de las partes y define lo que se conoce como propiedades

emergentes, que son cualidades del conjunto no inherentes a la suma de las partes.

Existen evidencias que apoyan el enfoque asocicionista, como el hecho de que en un primer nivel de percepción la imagen se capta como un todo o de que es la propia experiencia previa del individuo la que permite extraer información de una imagen plana para resolver volúmenes, sombras, superposiciones y distancias relativas al observador. No obstante, tal como se ha comentado en el apartado anterior, existen varios niveles de preprocesamiento anteriores al que tiene lugar en el córtex, que buscan patrones y características morfológicas diferenciales de la escena. Además, en la imagen existen propiedades emergentes inherentes a la disposición de los distintos objetos, como, por ejemplo, las sombras. Múltiples experimentos [Coren y Girgus 1978] han mostrado que, efectivamente, el ser humano tiende a percibir más de lo que realmente está observando en virtud de su experiencia. Un ejemplo clásico de ilusiones ópticas, conocido como la Escalera Imposible, puede observarse en la Fig. 1.2 y fue diseñado originalmente por Oscar Reutesvard, aunque posteriormente lo reprodujeron otros artistas más conocidos como M. C. Escher (*Ascending and Descending*) o genetistas como Lionel Penrose, que desconocían el trabajo de Reutesvard. En la imagen un grupo de individuos transita por una escalera. Sin embargo, prestando más atención a la escena, puede apreciarse que es imposible determinar cuál es el primer escalón y cuál es el último: el ascenso/descenso es cíclico. La explicación de este fenómeno reside en que el modelo de escalera presenta una discontinuidad a la derecha, pero ésta

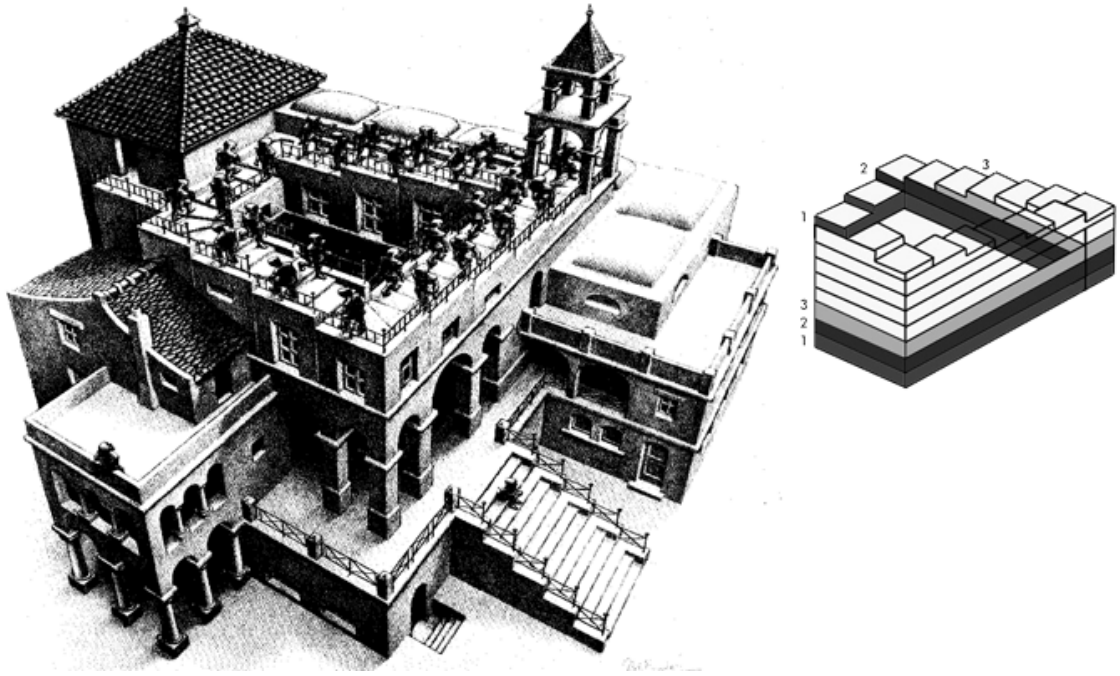


Figura 1.2: Ilusiones ópticas en 'Ascending and descending', de M.C. Escher, y su explicación física

no se aprecia porque el sistema visual humano asume que lo que ve es correcto, es decir, que los escalones no muestran discontinuidades. De esta forma, aunque el diseño es conceptualmente imposible, no interfiere con la percepción del usuario. Las conclusiones que se desprenden de experimentos como éste nos inclinan a basarnos en la teroría de la *gestalt*.

Una consecuencia directa de lo anteriormente comentado es que lo que el ser humano percibe es distinto de lo que existe en la realidad objetiva. La percepción está afectada por una serie de hipótesis previas que suelen influir en que se vea lo que se espera ver. Así, partiendo de una imagen determinada, el individuo tiende a definir una serie de entidades en ella según determinados criterios. Siguiendo las teorías de la *gestalt*, la visión tiende a organizarse de acuerdo a los siguientes principios:

- Se tienden a agrupar las percepciones. Las leyes de agrupación de Wertheimer han resistido el paso del tiempo y son las siguientes:
  - Ley de proximidad, que postula que el individuo tiende a agrupar objetos próximos.
  - Ley de similitud, que postula que el individuo tiende a agrupar objetos parecidos.
  - Ley de cierre o inclusión, que postula que el individuo tiende a rellenar huecos a efectos de percibir formas completas.
  - Ley de buena continuación, que postula que el individuo tiende a ver las líneas continuas, sin cortes.
- Se tiende a percibir que el mundo no cambia, lo que impulsa a apreciar:
  - Constancia en brillo y color.
  - Constancia en tamaño.
  - Constancia en forma.
- Se tiende a percibir figuras que destacan sobre un fondo.
- La atención es selectiva, sólo una figura se considera como tal en cada momento. La Fig. 1.3 muestra un ejemplo donde, o bien se percibe a la mujer contemplándose en el espejo o bien una calavera.
- Los estímulos que presentan ciertas características tienden a convertirse en figuras, atrayendo la atención del observador. Estas características son, por ejemplo: i) repetición; ii) intensidad; iii) variación; iv) necesidad.
- La percepción de la figura afecta a la percepción del fondo.

- La percepción depende de y es relativa a la experiencia pasada y fija el nivel de adaptación, que es el criterio por el que se decide si un objeto es grande o pequeño, está cerca o lejos y consideraciones de este tipo. Este importante hecho puede observarse en la Fig. 1.4, donde dos gemelas idénticas parecen tener un tamaño muy distinto sencillamente porque la imagen se capturó en un cuarto especialmente diseñado para dar esa impresión. El primero en construir dicho cuarto fue Adelbert Ames, Jr. en 1946, basándose en un concepto originario de Hermann Helmholtz de finales del siglo XIX.

Se puede extraer como conclusión de todos estos principios que la enorme capacidad de aprendizaje y adaptación del ser humano influye enormemente en el proceso visual y que, por tanto, difícilmente puede reproducirse en un sistema de visión artificial. No obstante, los principios por los que opera sí pueden usarse como base para desarrollar algoritmos de procesamiento que operen de forma similar al ojo humano en aplicaciones concretas, como por ejemplo las leyes de agrupación en segmentación. Por consiguiente, parece razonable intentar ajustar los objetivos de diseño de un sistema a las necesidades que la aplicación presente, en lugar de pretender imitar el funcionamiento del sistema visual humano.

1.3 Aspectos relevantes de la visión artificial

En su actual grado de desarrollo, la visión artificial puede reemplazar a la humana en una gran variedad de tareas como, por ejemplo, la inspección, el seguimiento o guiado de objetivos,

Figura 1.3: Ilusión óptica: Portada del album Retroactive, de Def Lepard, por Nels Isralson.

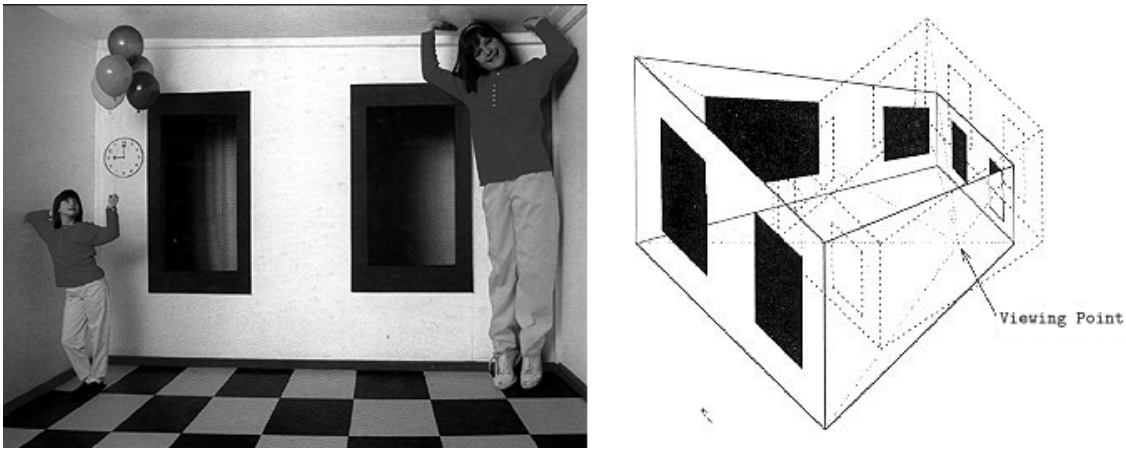


Figura 1.4: Ilusión óptica: Cuarto de Ames y su explicación.

etcétera. De hecho, es más efectiva si se requiere información muy precisa de forma rápida y/o repetitiva a partir de una imagen. Un sistema de visión estándar extrae información de una imagen a través de los siguientes pasos:

- Adquisión de la imagen mediante un dispositivo sensor adecuado y almacenamiento en una zona de memoria determinada.
- Preprocesamiento de la imagen para reducir las distorsiones geométricas implícitas al proceso de captura.
- Procesamiento de la imagen para eliminar el ruido de digitalización en la medida de lo posible, así como para mejorar determinadas características de interés.
- Extracción de las características de los objetos detectados en la escena.
- Reducción de dicho conjunto de características mediante la eliminación de aquellas que resultan superfluas u obedecen a ruido del sistema.
- Análisis de los resultados a efectos de tomar una decisión, que dependerá de la aplicación que vaya a llevarse a cabo.

La Fig. 1.5 presenta una de las posibles configuraciones de un sistema de visión, si bien pueden existir múltiples variaciones en función del campo de aplicación. Tal como puede apreciarse, la información visual del entorno es recogida por un sensor de imagen, que puede ser del tipo cámara de vídeo, ecógrafo, escáner óptico, resonancia magnética o cualquier otro dispositivo que genere una representación perceptible por el ojo humano. La señal generada se transfiere al ordenador y, en ocasiones, se procesa mediante una tarjeta capturadora. La imagen

digital resultante, que presentará como mínimo un ruido de discretización intrínseco al sistema, estará disponible en ese momento en el ordenador servidor de vídeo, cuya tarea puede incluir su procesamiento, interpretación, almacenamiento y visualización. De ser necesario, este ordenador se encargará de distribuir la imagen por cualquier tipo de dispositivo de red hacia otro u otros procesadores a los que tenga acceso, para así poder acometer un procesamiento paralelo de la misma.

Comenzando por la cámara, la luz proveniente del entorno atraviesa una lente y su intensidad es capturada por un conjunto de elementos sensores, que generan una imagen bidimensional. La distribución de estos sensores suele ser uniforme, pero puede darse el caso de que se ajusten a un campo de resolución cambiante que emule el sistema foveal comentado en el subapartado anterior. Este tipo de topologías permite combinar una alta resolución en determinadas áreas de interés de la imagen con un volumen de datos reducido si se compara con imágenes de resolución uniforme que presenten el mismo campo visual. En general, las tecnologías más clásicas para la captura de imágenes son las cámaras de tubo vidicón, basadas en tecnología analógica, y las cámaras CCD, basadas en sensores de estado sólido. Inicialmente se tendió al primer modelo, por su bajo coste y alta resolución. Más tarde, fueron superadas por las cámaras basadas en la tecnología CCD, que presentan un importante conjunto de ventajas adicionales: un menor peso, tamaño, consumo, una geometría espacial exacta en todo el campo de visión, mayor solidez y fiabilidad y un mayor ancho de banda. Actualmente, una tecnología emergente basada en la fabricación de dispositivos CMOS parece abrirse camino en el campo de la adquisición de imágenes. Esta nueva tecnología presenta aún mejores prestaciones en cuanto a consumo y coste de fabricación, así como una capacidad de integración de un mayor número de regiones sensibles por área, aunque aún presentan problemas de ruido y un rango dinámico

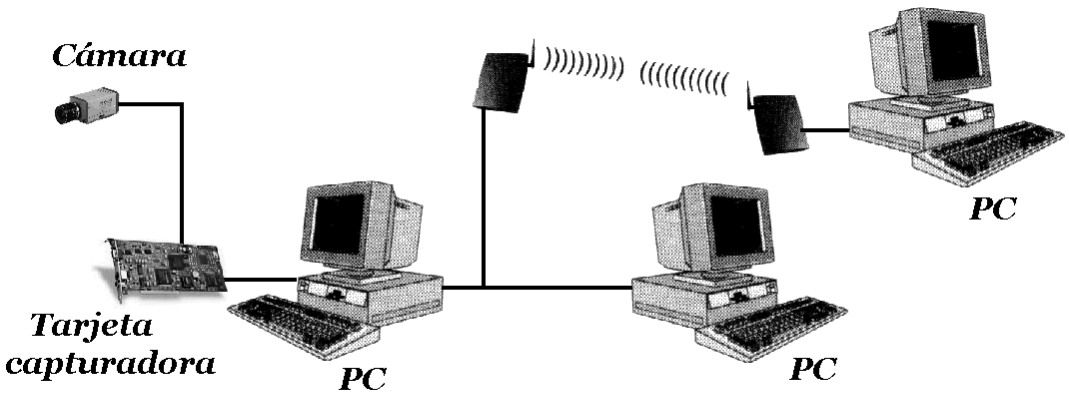


Figura 1.5: Sistema estándar de visión artificial

menor que los CCD, lo que actualmente limita su uso a aplicaciones profesionales en campos muy concretos de la fotografía digital.

La formación de imágenes planas en los sensores a partir de un entorno 3D está intrínsecamente ligada al problema de la proyección sobre el plano de la imagen mediante una transformación de perspectiva. Dejando aparte las distorsiones asociadas a las distancias focales, modelables mediante series finitas, la perspectiva no distorsiona los objetos en el plano, sino que se proyectan invertidos y escalados según una distancia focal f. Además, hay que tener en cuenta las transformaciones asociadas a la posición relativa de la cámara en el entorno, siendo las más habituales traslación, proyección, escalado y rotación. Todas estas transformaciones pueden modelarse mediante matrices y, por tanto, deshacerse para la interpretación de una imagen. No obstante, debido al ruido de discretización, el modelo corregido de un objeto cualquiera en una imagen digital puede llegar a ser considerablemente distinto del equivalente no sometido a transformaciones.

Las tarjetas capturadoras permiten adaptar las imágenes provenientes de las cámaras de vídeo al ordenador encargado de procesarlas. En el caso de las cámaras analógicas, la señal atraviesa un convertidor analógico-digital que realiza un muestreo en el espacio de los datos y lleva a cabo una cuantificación discreta del color de cada muestra. De esta forma, la mencionada imagen analógica pasa a convertirse en una matriz bidimensional, en la que la posición de cada muestra viene indicada por su fila y su columna y el valor que presenta es relativo a su color. Posteriormente, las muestras obtenidas son transferidas a la memoria del ordenador para su procesamiento, a un dispositivo de almacenamiento masivo, como un disco duro, o directamente a la tarjeta de vídeo para su visualización. Con las cámaras digitales, la tarjeta capturadora sólo se encarga de adaptar el flujo de datos proveniente de la cámara al bus del ordenador al que está conectada. En ambos casos, la tarjeta capturadora suele acoplarse al ordenador mediante uno de los siguientes tipos de bus: VME, PCI o USB. Aunque se trata de un bus en desuso, aún hoy en día pueden encontrarse tarjetas capturadoras conectadas al bus ISA, pero sólo son útiles si el tratamiento de la imagen se realiza íntegramente sobre la propia tarjeta capturadora o si se emplea un bus alternativo de comunicación con otros dispositivos procesadores. Por último cabría mencionar las cámaras con interfaz paralelo, cuya utilidad se reduce a aplicaciones concretas en las que no se exige calidad de captura y el bajo coste es uno de los requerimientos básicos.

El procesamiento de la imagen incluye técnicas tan extendidas como realce, suavizado, promediado o filtrado y sirven para preparar la imagen de cara a su posterior análisis mediante procedimientos más o menos específicos. Así, cabe destacar que el objetivo de este procesamiento

no es tanto extraer información de la imagen como actuar sobre ella para compensar defectos de iluminación, eliminar ruido y efectos espurios y llevar a cabo cualquier tipo de algoritmo que sirva para producir una imagen de mayor calidad para simplificar y facilitar posteriores etapas.

Antes de entrar en procesamientos más específicos, es necesario plantearse qué pretende conseguir el sistema y cómo llevarlo a cabo. Tal como se comentó previamente, un sistema de visión artificial no puede funcionar como su equivalente natural, ya que el ser humano es mucho más flexible y tiene una capacidad mayor. Por tanto, cuando se van a ejecutar algoritmos más complejos, es importante concretar y reducir los objetivos que se pretende conseguir. A continuación, hay que especificar los requerimientos de *hardware* y *software* que imponen dichos objetivos. Será por tanto necesario precisar conceptos como oscuro, brillante, cercano, lejano, móvil o quieto. En muchos casos, esta tarea no resultará sencilla, ya que los factores que han de ser analizados pueden ser patrones visuales muy sutiles o incluso puede no estar muy claro qué factor se debe cuantificar. Seguidamente, es necesario buscar algoritmos específicos capaces de manipular las características definidas de forma simple y eficaz. Es muy importante resaltar que no se puede extraer una determinada información de una imagen cuando dicha información no está disponible en ésta. Aunque parezca una obviedad, la segmentación mediante criterios de color de una imagen real adolece de este problema, debido a que, para llevarla a cabo, el ser humano no sólo se ciñe a dicho criterio, sino también a su experiencia y a otros muchos factores. A este respecto, las imágenes binoculares son una fuente importante de información acerca del mundo real; dan información de la profundidad y, por tanto, permiten la extracción de objetos en función de la distancia al observador. El movimiento es otra importante fuente de información visual, ya que una secuencia de imágenes contiene información acerca de la distribución tridimensional del campo de visión observado, así como del movimiento de todos los elementos presentes en él respecto a la cámara.

El último aspecto que debemos considerar reside en la implementación del sistema diseñado. Así, la elección del *hardware* dependerá de factores como la velocidad, el precio y su adecuación a las necesidades de procesamiento. No obstante, para conseguir un sistema versátil y asequible a cualquier usuario, resulta mucho más interesante una opción de bajo coste y alta flexibilidad consistente en una capturadora estándar y una CPU de propósito general. Obviamente, este tipo de CPU es capaz de llevar a cabo cualquier procesamiento si dilatamos el tiempo de ejecución lo suficiente. Sin embargo, el tiempo es un factor crítico para muchas aplicaciones. Una posible solución en estos casos consiste en adaptar los algoritmos al *hardware* disponible en lugar de operar a la inversa. Así pues, el desarrollo de algoritmos simples que puedan funcionar de forma eficiente en un sistema de propósito general incrementa su trans-

portabilidad y asequibilidad y, obviamente, reduce drásticamente el precio global del sistema. Por último, cabe mencionar que cuando no sea posible la solución anterior, otra alternativa sería diseñar algún tipo de *hardware* compatible con la CPU genérica que acelerase en parte el proceso y cuyo coste fuese más reducido que el de utilizar una CPU de propósito específico.

1.4 Panorama actual de la visión artificial

Posiblemente debido al creciente aumento de la potencia de cálculo de los ordenadores y a la disminución del precio del *hardware* en general, el conjunto de campos en los que se aplican algoritmos propios de la visión artificial ha crecido espectacularmente en los últimos años. Así, a las típicas aplicaciones médicas o de inspección industrial (Fig. 1.6) que usaban algoritmos de restauración o mejora de imágenes, se ha añadido un importante conjunto de aplicaciones que se basan más en visión activa o reconocimiento de objetos, lo que ha supuesto un importante esfuerzo de integración de los algoritmos de visión con otros más propios de la inteligencia artificial. En este apartado se hará un breve recorrido por algunas de estas aplicaciones, partiendo de las que se pueden considerar más clásicas, que se aplican sobre escenas bidimensionales, hasta llegar a las más actuales, que implican el tratamiento de escenas tridimensionales.

Entre las tareas más clásicas en las que se emplean algoritmos de visión artificial, las de mejora y restauración de imagen son posiblemente las que primero encontraron aplicación práctica. La misión de estas tareas es, básicamente, conseguir modificar las características de la imagen captada para que sea más apta para una determinada aplicación. Así, en este tipo de algoritmos se encuadran los de filtrado de imagen usados en radiología [Hall 1971] o en inspección de placas de circuito impreso [Jarvis 1980]. En la actualidad, sin embargo, la problemática del análisis de imagen plana es otra. Así, uno de los principales objetivos del procesamiento de imágenes planas ha pasado a ser, posiblemente, el del reconocimiento de formas. Este campo

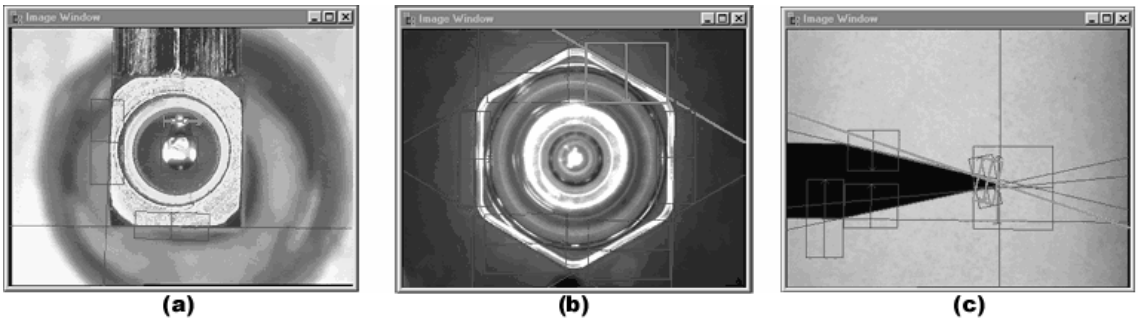


Figura 1.6: Inspección industrial: a) inspección de soldaduras; b) medida automatizada; c) corrección de ángulos en puntas de aguja

engloba directamente un importante conjunto de aplicaciones, como pueden ser el reconocimiento automático de caracteres o el análisis de cariotipos, o bien sirve de ayuda en otras, como en los sistemas de extracción de información geográfica a partir de imágenes capturadas desde el aire o el espacio [Matsumoto et al. 1981] (Fig. 1.7).

La idea de construir un sistema que sea capaz de leer de forma no supervisada se remonta a la década de 1930 [Suen et al. 1980]. Así, debido a la aparente facilidad con que parecen hacerlo los seres humanos, se podría pensar que se está ante una tarea fácil. Sin embargo, la impresionante cantidad de estilos y formas existente tanto de alfabetos automáticamente impresos como escritos a mano, dificultan enormemente la tarea. En un primer momento, los principales esfuerzos se destinaron a estandarizar tipos de fuentes impresas y formatos [Suen y Mori 1982], de manera que actualmente se pueden encontrar sistemas de OCR (*Optical character recognition*) que procesan correctamente documentos impresos. Sin embargo, en la actualidad los esfuerzos se dedican más a intentar el diseño de un algoritmo de OCR que pueda trabajar sobre espacios muestrales parcialmente estandarizados, como matrículas de vehículos de cualquier país o marcado de vagones y contenedores [Suen 1986] (Fig. 1.8), o sin ningún tipo de estandarización (caracteres escritos en cheques o cartas escritas a mano [Bellegarda et al. 1993][Connell y Jain 2001]).

Conceptos similares a los empleados por el OCR, como los descriptores de forma y tamaño, área o radio, son empleados en medicina para describir distintos objetos biológicos y han sido, tradicionalmente, extraídos manualmente de las placas fotográficas. Las técnicas de procesamiento digital pueden emplearse para aumentar la velocidad y calidad de estas medidas, generándose ahora las imágenes en formato digital mediante métodos variados (Fig. 1.9), y pueden también ser usadas para obtener otros descriptores que son difícilmente extraíbles de otra forma [Bradbury 1983]. De cualquier forma, la visión artificial ha encontrado en la medicina uno de los campos de mayor aplicación. Algunos ejemplos de esta integración son la realización de cariotipos [Charters y Graham 1999], técnicas visuales no intrusivas para detección de cáncer de mama [Bartrum y Crow 1984] [Wolberg et al. 1994] o el análisis de electroencefalogramas

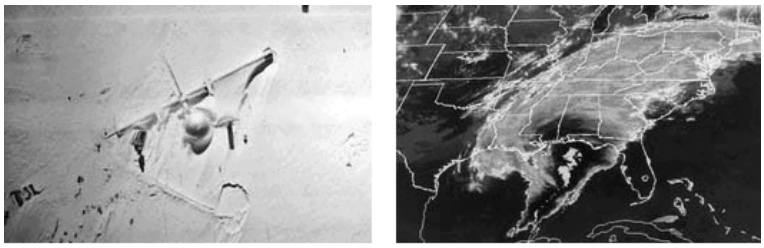


Figura 1.7: Análisis de información geográfica: a) fotografía aérea; b) imagen por satélite

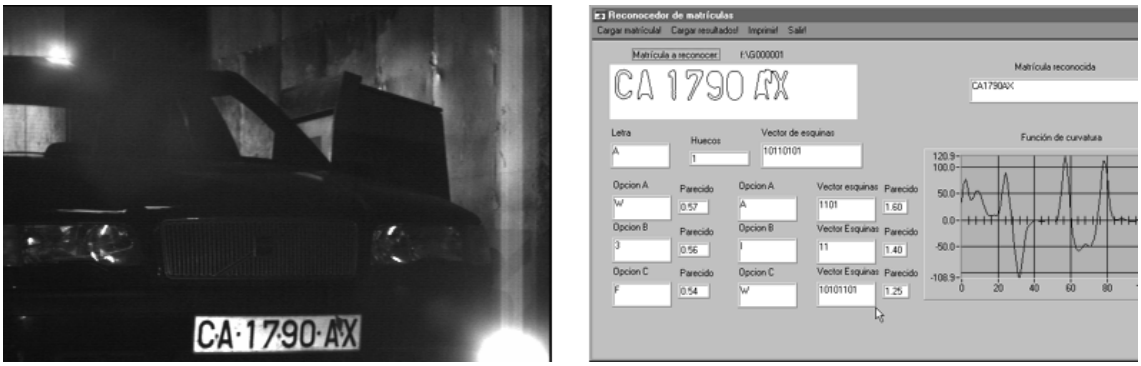


Figura 1.8: OCR para reconocimiento automático de matrículas

[Bourne et al. 1981] [Zhou y Zhou 1999].

Por otro lado, cada vez cobra más importancia dentro de la visión artificial el tratamiento de escenas tridimensionales. Las aplicaciones que precisan este tipo de procesamiento abarcan desde el modelado visual para la inspección industrial [Rosen y Nitzan 1977] [Kim et al. 1999] hasta las técnicas de tomografía [Bhattacharya y Majumder 2000], que tratan de reconstruir un objeto mediante el estudio de las imágenes planas tomadas del mismo al ser iluminado desde distintas direcciones, pasando por la reconstrucción, tanto de objetos como de entornos, empleada en robótica [Johnson y Hebert 1998] [Beauvais y Lakshmanan 2000].

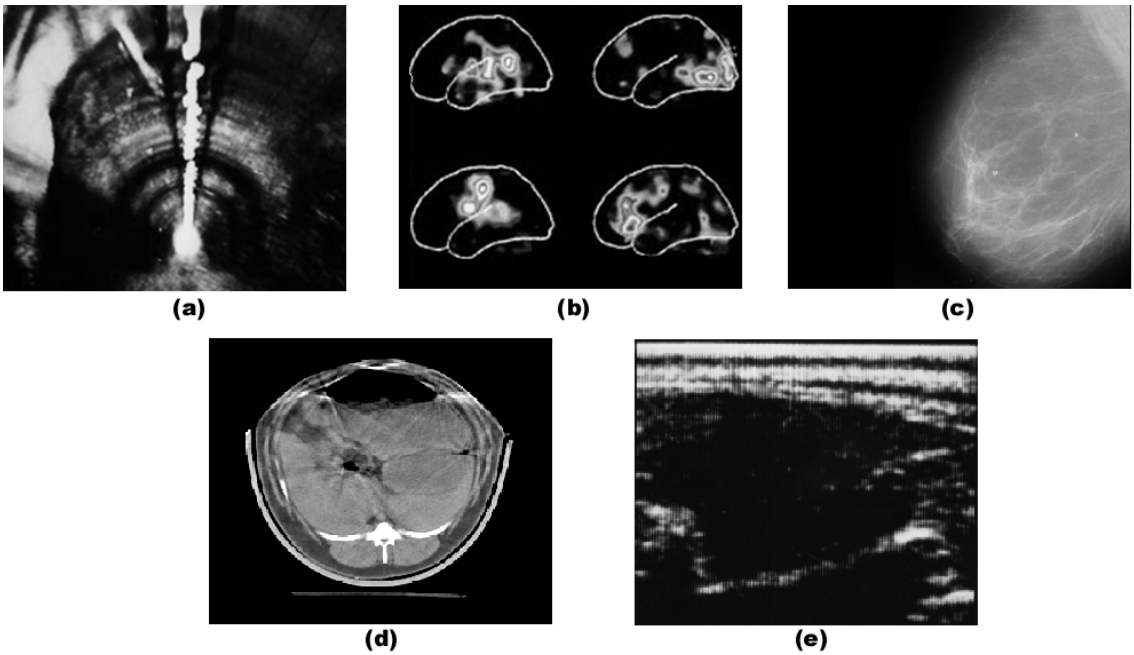


Figura 1.9: Visión en imagen médica: a) resonancia magnética; b) tomografía; c) mamografía; d) radiografía; e) ecografía

Además, al trabajar con escenas tridimensionales, el problema visual ha ganado en complejidad, surgiendo nuevos retos como el enfoque de la cámara o el uso de la visión estereoscópica [Granlund 1999], con el consiguiente problema de controlar el ángulo que forman las dos cámaras necesarias en el proceso [Rodríguez et al. 1999]. De ahí que las aplicaciones industriales suelan emplear la visión 3D sólo en caso de que ésta sea estrictamente necesaria. Sin embargo, sí que han sido ampliamente empleadas desde hace tiempo las capacidades de reconstrucción de entornos que ofrece el análisis de distintos planos bidimensionales del mismo, tanto en radiografía [Stark et al. 1981] como en reconocimiento de objetos [Gilbert 1972].

La aplicación de los algoritmos clásicos de visión en el mundo de la robótica móvil, junto con la aceptación de la premisa de que la compresión de la imagen implica también el proceso de adquisición selectiva de información tanto en tiempo como en espacio [Blake y Yuille 1992] [Aloimonos y Bandopadhay 1988], origina la denominada visión activa. En contraste con la teoría clásica de la visión artificial, basada en un proceso reconstructivo, que conduce a la creación de representaciones del entorno a niveles de abstracción cada vez mayores, la visión activa selecciona aquella información del entorno que necesita para una determinada tarea o comportamiento. Al reducir significativamente el volumen de información adquirida y procesada, estos sistemas pueden operar más fácilmente en situaciones reales [Araújo et al. 1998].

La autonomía es la principal característica de cualquier agente móvil (Fig. 1.10), e implica la capacidad de ajustarse a cambios en el entorno. Esto no quiere decir que un mismo sistema, trabajando en distintos entornos, tenga que emplear un mismo sistema perceptivo. Al igual que la evolución ha forzado a una misma especie animal a adaptarse de distinta manera en función del entorno, la estructura y algoritmos serán distintos según sea el propósito y entorno en que se deba desenvolver el agente móvil. Esta comparación es algo más que un simple ejemplo, ya que el comportamiento de los seres vivos ha sido una importante fuente de inspiración para el desarrollo de la visión activa [Aloimonos 1997].

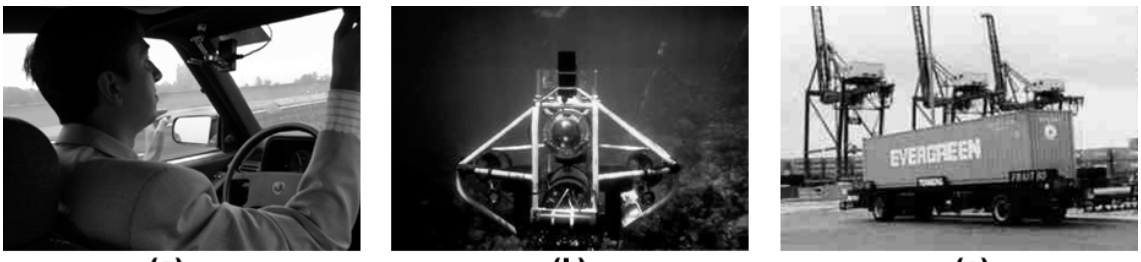


Figura 1.10: Agentes móviles autónomos guiados por visión: a) el coche ARGO; b) un robot submarino de la ACFR; c) un contenedor de la ACFR

Uno de los campos en el que se puede prever un importante avance en la aplicación de algoritmos y técnicas de visión activa es el de la videovigilancia [Howarth y Buxton 1996] [Kanade et al. 1997]. Dentro de este campo se pueden englobar actividades tan diversas como la monitorización y vigilancia del tráfico [Davis et al. 1997] o el control de la actividad humana mediante imágenes aéreas [Rao 1996]. La característica principal común a estas aplicaciones es la presencia de movimiento. En ocasiones el objetivo consiste exclusivamente en detectar movimiento en la escena, mientras que en la mayoría de los casos se requiere conocer la trayectoria de los móviles con el fin de poder realizar un seguimiento de los mismos. En todos los casos, especialmente cuando se trabaja en entornos cerrados, la imagen se recibe en forma de secuencia de vídeo, lo que supone una dificultad añadida, pues si se quiere trabajar en tiempo real, el sistema deberá adaptarse, tanto en el *software* como en el *hardware*, a una velocidad de procesamiento generalmente alta.

2 Justificación y objetivos

Uno de los puntos en común de la práctica totalidad de los sistemas basados en visión artificial es la necesidad de dividir la escena en regiones para darle un sentido. Antes de establecer esta división, la imagen no es más que un conjunto de píxeles con información de color. Después de ejecutarla, se tienen regiones con unas determinadas características, formas y tamaños que permiten extraer información acerca de la disposición del entorno de manera que se pueda analizar de forma inteligente. Este método de trabajo está de acuerdo con la teoría de percepción de la *gestalt* y suelen usarse las leyes de agrupación sugeridas por esta corriente de pensamiento para aislar regiones en una imagen digital.

Por otra parte, uno de los inconvenientes intrínsecos a la visión artificial es el enorme volumen de datos que implica la información visual. Si bien el cerebro humano dispone de tantos elementos de procesamiento que no debe preocuparse de esta problemática, los ordenadores, a pesar de su rápida evolución, tienen una capacidad de procesamiento, almacenamiento y velocidad limitada que le impone unas cotas de funcionamiento a los sistemas que los soportan. Este hecho apoya, a su vez, la necesidad de un proceso previo de segmentación que permita eliminar las áreas innecesarias de la escena para centralizar la potencia de cálculo únicamente en las zonas de interés y poder procesarlas de forma adecuada.

Por las razones anteriormente expuestas, la segmentación se ha convertido en el centro de atención de un elevado número de investigadores desde prácticamente los inicios de la visión artificial. Sin embargo, lejos de ser sencilla, la segmentación de una imagen se convierte en un

escollo casi insalvable cuando se está trabajando con imágenes reales. Este problema aparentemente tan simple para el ser humano es difícilmente trasladable a una máquina en tanto que no se puede cuantificar y sistematizar la experiencia que utiliza aquel a la hora de distinguir las regiones que componen una escena cualquiera. Lo que sí parece probado más allá de toda duda es que el ser humano no se ciñe a un único criterio, sino que aprovecha toda la información que tiene disponible en un momento dado, incluidos el conocimiento previo, la visión estereoscópica y la percepción de las imágenes en el espacio y en el tiempo. Otro factor, en absoluto despreciable, es la capacidad del hombre para trabajar como sistema realimentado, es decir, si su percepción actual no le permite extraer la información que desea, puede modificar las condiciones de captura hasta que lo consiga.

Las técnicas de segmentación han evolucionado a la par que las tecnologías de captura y procesamiento de imagen. A pesar de su enorme variedad, pueden dividirse en tres grandes grupos de acuerdo a la información con que trabajan: espaciales, temporales y espacio-temporales. El primer grupo engloba aquellas técnicas que se basan únicamente en características disponibles en el plano de la imagen, como color, brillo o textura, para descomponerla en regiones. Dentro de este grupo cabe destacar las técnicas basadas en profundidad, disponibles cuando se trabaja con visión estereoscópica, que ofrecen mejores resultados que el resto, ya que cuenta con el doble de información que cuando se procesa una única imagen. En un sentido amplio, también pueden englobarse en este grupo las técnicas de sustracción de fondo o diferencia de imágenes cuando se trabaja con secuencias, ya que no emplean criterios temporales propiamente dichos, sino información espacial a partir de varias imágenes. El segundo grupo de técnicas incluye aquellas que se basan en extraer parámetros de movimiento de los píxeles de la imagen a partir de las variaciones que éstos presentan en una secuencia de vídeo. En este caso, el objetivo es dividir la escena en regiones que se desplacen de forma homogénea, asumiendo que cada una de ellas constituirá un objeto distinto. Por último, las técnicas espacio-temporales buscan combinar las dos anteriores para conseguir resultados más robustos. Así, a una región no sólo se le exige coherencia en cuanto a su movimiento, sino también en cuanto a las características espaciales con que se esté trabajando.

El objetivo del presente trabajo es desarrollar un método de segmentación rápido, robusto y capaz de trabajar con todo tipo de secuencias reales independientemente de su complejidad o de la disposición de la escena. Para ello, se procederá a evaluar los tres grupos de técnicas de segmentación previamente mencionadas para estudiar su viabilidad. A partir de los resultados de este estudio, se va a implementar un sistema nuevo de segmentación que encaje dentro del grupo que ofrezca mayores ventajas. Dicho sistema se probará con imágenes de todo tipo para

evaluar su eficacia y, cuando sea posible, se comparará con otros sistemas conocidos en su campo.

Para implementar el sistema elegido se ha optado por el diseño modular, siguiendo las últimas tendencias en inteligencia artificial [Moreno et al. 1996] [Bianco y Cassinis 1996] [Buhnmann et al. 1995]. Este tipo de diseño ofrece las siguientes ventajas:

- Permite diseñar y probar cada módulo de forma independiente, haciéndose así más sencilla la detección y corrección de errores.
- Permite que cada uno de los módulos pueda manipularse mediante un conjunto limitado de primitivas, que posteriormente pueden manipularse para obtener comportamientos gradualmente más complejos.
- Permite de forma sencilla la integración de acciones a bajo nivel con las superiores.
- Facilita el uso de arquitecturas híbridas de control, que combinan acciones rápidas con procesos deliberados, dado que la interacción entre módulos puede llevarse a cabo de acuerdo al modelo "caja negra".

Asimismo, se ha optado por trabajar sobre un sistema operativo que permita multiproceso a efectos de optimizar el tiempo de procesamiento de los distintos algoritmos mediante su ejecución en paralelo, siempre que sea posible. Hasta la aparición de las teorías de Brooks en 1986 [Brooks 1986], la descomposición típica de los sistemas que predominaba en la comunidad científica era de tipo secuencial. Esta filosofía tiene una serie de implicaciones, entre las que cabe destacar un elevado tiempo de respuesta global del sistema, la implementación no progresiva del sistema completo y una depuración compleja. Sin embargo, tras las teorías de Brooks, el panorama general sufrió un cambio radical, y la nueva descomposición se orientó a un enfoque paralelo. Esta filosofía tiene numerosas ventajas, entre las que destacan:

- Tiempo de respuesta independiente de cada módulo.
- Procesamiento paralelo.
- Implementación progresiva del sistema.
- Depuración sencilla.

Así, y a efectos de aprovechar al máximo las ventajas que ofrece una implementación de este tipo, se ha escogido como sistema operativo el Linux, que permite proceso en paralelo de forma

sencilla. En un futuro, el uso de un *kernel* de tiempo real sobre este mismo sistema permitirá, asímismo, el control absoluto de prioridades para una sincronización precisa del sistema.

3 Organización del texto

Esta tesis se va a dividir en varios bloques que obedecen a la evolución natural del proceso de detección de regiones de interés que se ha llevado a cabo. Cada uno de los capítulos incluye una sección de resultados y conclusiones que corresponden a todas las pruebas intermedias que se han ido efectuando sobre el sistema. Así, los capítulos 2, 3 y 4 cubren las técnicas de segmentación de una imagen independiente, una secuencia en que no cambia el campo de visión y una secuencia en que éste sí podría cambiar. Todos los métodos se han probado con imágenes artificiales e imágenes reales capturadas en interiores. A continuación, y una vez seleccionado el mecanismo de segmentación más completo implementado, se presenta en el capítulo 5 el sistema de transmisión de imágenes completo, incluyendo los mecanismos de control propuestos, las geometrías de imagen utilizadas para su compresión y las distintas situaciones en que se ha probado el sistema. Los resultados de este capítulo se han comprobado directamente en situaciones reales, ya que en este caso las simulaciones no daban idea apropiada de la magnitud del problema. Por último, las conclusiones y las líneas de trabajo futuro que se espera que abra esta tesis se han incluido en un último capítulo. Cabe reseñar un anexo donde se incluye el manual de referencia de todo el *software* que se ha implementado para el desarrollo del sistema propuesto.

De acuerdo a lo expuesto, los capítulos presentan los siguientes contenidos:

- Detección de áreas de interés en un único fotograma.

  Este capítulo describe los distintos algoritmos de segmentación que pueden aplicarse a una imagen independiente de una secuencia de vídeo cualquiera. Dentro de las distintas características que pueden utilizarse para segmentación, se ha focalizado el problema en niveles de gris, ya que es la técnica más utilizada en estos casos. Una vez se han obtenido las regiones buscadas, se proponen un conjunto de criterios para fijar su importancia en relación con lo que se define como fondo de la escena. El resultado del estudio presentado muestra las carencias de los métodos expuestos para la segmentación de una imagen real compleja, evidenciando la necesidad de utilizar otros recursos para conseguir resultados apropiados.

- Detección de áreas de interés en una secuencia con un campo de visión fijo.

Este capítulo supone un paso más en las técnicas descritas en el anterior, ya que ahora se dispone de varias imágenes equiespaciadas en el tiempo. No obstante, es una simplificación del problema de estimación de movimiento, ya que en este caso el dispositivo sensor se mantiene en una posición fija. Así, se describen las técnicas más usuales para detección de movimiento en estos casos y se propone un método nuevo que cubre algunas de las deficiencias de los métodos estudiados para conseguir una segmentación robusta de movimiento en la secuencia, así como el seguimiento de las regiones de interés a lo largo de su paso por el campo de visión. En este caso, el fondo lo constituyen las partes fijas de la imagen frente a las áreas de interés, que son las correspondientes a los objetos que se desplazan en la secuencia. El método propuesto ha sido probado con secuencias reales de complejidad variable y en situaciones no controladas para comprobar su eficiencia, que ha quedado probada por los resultados bajo las hipótesis establecidas.

- Detección de áreas de interés en una secuencia con un campo de visión variable.

  Este capítulo cubre el problema de la segmentación por movimiento de una secuencia de imágenes, donde ya no se admiten restricciones en cuanto al movimiento del sensor o de los objetos dispuestos en cada instante de tiempo en el campo de visión. Así, se presentan y analizan un conjunto de métodos aplicables a estos casos y se introduce un nuevo sistema que permite realizar simultáneamente la segmentación por movimiento de la imagen y el seguimiento de las distintas regiones a lo largo de los distintos fotogramas.

- Sistema de transmisión de vídeo adaptado al estado del canal.

  En este capítulo se presenta un sistema completo de transmisión de vídeo, adaptado al estado del canal, propuesto como posible aplicación del nuevo método de segmentación descrito en el desarrollo de esta tesis. El sistema presenta un dispositivo capturador de vídeo en cuyo extremo corre el algoritmo de segmentación basado en movimiento que se presentó en el capítulo anterior. Una vez se dispone de las áreas de interés en orden de prioridad, se ejecuta un algoritmo de muestreado no uniforme que convierte cada imagen capturada en foveal, fijando el área de máxima resolución de la imagen sobre la zona de interés más prioritaria. Las imágenes foveales presentan un volumen de datos considerablemente menor que las imágenes uniformemente muestreadas y, por tanto, cargan menos el canal. Si se comprueba el estado de dicho canal cada cierto tiempo para estimar el retardo de éste, se puede mantener estable la tasa de imágenes por segundo mediante la reducción selectiva de resolución en función de la prioridad de las distintas áreas de la imagen.

- Conclusiones y trabajo futuro.

  Por último, este capítulo presenta las conclusiones extraídas de todo el trabajo desarrollado en la tesis, así como las posibles líneas de trabajo que podrían emprenderse a partir de los resultados disponibles. Entre los principales objetivos cabe destacar la implementación del sistema sobre un *kernel* de tiempo real y el traspaso a *hardware* del algoritmo de detección de movimiento, lo que permitiría elevar enormemente la velocidad del sistema.

Capítulo 2

Técnicas de segmentación de imágenes por nivel de gris

1 Introducción

El esquema más básico de percepción es aquel en que se dispone de una cámara cuya posición y campo de visión no varía en el tiempo, y de una única imagen que se mantiene fija durante todo el proceso perceptivo. En este caso, el dispositivo capturador no ha de ser necesariamente una cámara de vídeo, sino que puede tratarse de cualquier dispositivo capaz de producir una imagen, como una cámara de fotografía digital o un escáner. Es por tanto inmediato constatar que en estos casos la velocidad de proceso no es un factor decisivo, ya que no existen imágenes en cola esperando a ser procesadas en función de la velocidad de captura del dispositivo. No obstante, continúa siendo interesante utilizar algoritmos de procesado ágiles y sencillos para conseguir tiempos de proceso acotados, aún cuando no se disponga de procesadores específicos.

Cuando se requieren un amplio campo de visión y una resolución elevada, trabajar con la imagen completa puede implicar unos requisitos computacionales excesivos. En estos casos resulta más adecuado trabajar de forma activa, concentrando los recursos del sistema en las zonas de interés para de esta forma mantener la carga computacional bajo límites controlados [Rodríguez et al. 1998]. No obstante, para conseguir dicho objetivo es imprescindible llevar a cabo los procesos necesarios para la detección y delimitación de dichas zonas de interés. Posteriores procesos, computacionalmente más costosos, podrán llevarse a cabo únicamente en las zonas mencionadas, o de forma intensiva sobre ellas y menos precisa sobre el resto de la imagen.

Tradicionalmente, las áreas de interés de una imagen son aquellas que se distinguen del

fondo, entendiéndose éste como una zona homogénea respecto a un rasgo determinado o conjunto de éstos, cuya naturaleza puede variar en función de las características de la imagen de estudio. Si bien el ser humano es capaz de seleccionar de forma automática los rasgos distintivos de las áreas de interés, en el caso de la visión artificial, éstos suelen fijarse a priori, estableciendo criterios de segmentación por color, textura, distancia a la cámara, vergencia, velocidad, tamaño, forma, enfoque u otros muchos. Sin embargo, resulta extremadamente complejo elegir los criterios que permiten escoger un foco de atención determinado. La Fig. 2.1.a, por ejemplo, muestra un caso sencillo que presenta un claro foco de interés sobre un fondo homogéneo de color oscuro. En este caso, el criterio a seguir para elegir el área de interés es extremadamente simple: será todo aquello que no sea negro. Incluso si el fondo presentase un patrón homogéneo, la forma del pájaro permitiría distinguirlo de éste. Sin embargo, en la Fig. 2.1.b no hay distinción posible entre áreas de interés y fondo, ya que todas las estructuras son idénticas en las únicas características distintivas que presentan: forma y tamaño.

Es interesante notar que la percepción del ser humano está íntimamente ligada a la experiencia, lo que la hace difícil de emular mediante sistemas informáticos. Por ejemplo, podría tenderse a pensar que en la Fig. 2.1.a se ha seleccionado como fondo el color oscuro y como objeto de interés el pájaro en color claro sencillamente porque el área ocupada por el fondo es mucho mayor que la ocupada por el objeto de interés. Este criterio suele estar bastante extendido a la hora de descartar regiones potencialmente pertenecientes al fondo en visión artificial

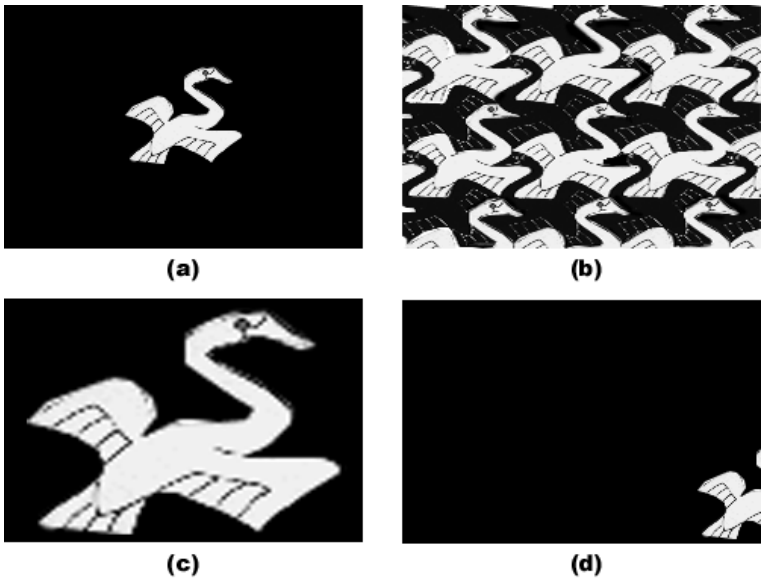


Figura 2.1: Separación fondo/objeto: a) segmentación por fondo homogéneo; b) fondo inexistente; c) segmentación por aprendizaje; d) segmentación por reconstrucción. (Imagen compuesta a partir de *Birds* de Moebius)

[Arrebola 1998]. Sin embargo, si se presta atención a la Fig. 2.1.c, el observador humano probablemente seguiría percibiendo un pájaro blanco, ahora mucho mayor, sobre fondo oscuro, a pesar de que en dicha figura no está claro cuál es el color mayoritario. Este hecho obedece obviamente a que el observador reconoce inmediatamente la silueta de un pájaro a partir del conocimiento de formas que posee y, por tanto, es capaz de substraerlo del fondo de forma eficaz. Para que una máquina pudiese trabajar de la misma forma debería segmentar la imagen por niveles de gris y, con posterioridad, tratar de reconocer los contornos de las distintas regiones en función de la base de datos que posea. En cuanto aumenta la complejidad de la imagen, esta operación, que no sólo depende del tamaño de la base de datos disponible sino también de la calidad de la segmentación, se complica sobremanera. Sin necesidad de buscar un ejemplo muy diferente, se puede apreciar en la Fig. 2.1.d que, al estar el contorno del pájaro parcialmente fuera de la imagen, serían necesarios procesos reconstructivos más inteligentes [Bandera et al. 2000b] para poder relacionarlo con el de la Fig. 2.1.c, mientras que el observador humano sigue percibiéndolo sin dificultad, ya que completa la forma que quiere ver.

La distinción de un fondo en una imagen es aún más compleja cuando la escena es real, ya que la mayoría de las características del fondo perceptibles en una única imagen dejan de ser homogéneas individualmente, y sólo un conjunto de éstas, que varía de imagen a imagen, permite al observador, en función de su experiencia, determinar donde concentra su interés, en el peor caso de acuerdo a criterios extremadamente sencillos, como por ejemplo buscar el área central de la imagen. Así, en la Fig. 2.2.a, un observador humano podría afirmar que la figura que se distingue entre la vegetación es el área de interés, básicamente porque asigna una consistencia al fondo vegetal, que ahora ya no resulta posible segmentar de acuerdo a su color, ya que presenta distintos niveles de éste. Puede tenderse a pensar que el observador podría basarse en su experiencia para unir los distintos fragmentos que forman la figura, pero si se observa la

Figura 2.2: Separación fondo/objeto: a) fondo complejo; b) fondo homogéneo.

Fig. 2.2.b es inmediato notar que la tarea es mucho más fácil eliminando el fondo que tratando de integrar regiones complejas.

De acuerdo a la teoría clásica, para fijar un área de interés sobre un fondo determinado, en el procesamiento de imágenes son necesarios dos pasos básicos interrelacionados: una primera etapa de segmentación de la imagen en regiones homogéneas, y una etapa posterior en que se descartan aquellas regiones que, de acuerdo a determinados criterios, pertenecen al fondo. Las regiones conexas restantes constituyen los focos de interés de la escena. El problema más relevante en este proceso reside en la selección de las características que van a determinar la segmentación y las reglas que permitirán el descarte del fondo, lo que resulta especialmente complicado si se trabaja con imágenes reales.

Este trabajo se centra en el procesamiento de imágenes con una única cámara, ya que la visión estéreo recibe un tratamiento muy distinto. Dada su popularidad, la segmentación de imágenes estáticas desarrollada en este capítulo se va a basar únicamente en niveles de gris, presentando las técnicas más habituales en la sección 2. Así, se intentará suplir sus deficiencias en la fase posterior de detección de objetos, que se detalla en la sección 3. La sección 4 presenta los resultados y conclusiones de esta etapa y justifica la necesidad de trabajar con otros medios, que se introducirán más adelante en los capítulos 3 y 4.

2 Segmentación de una escena

Los algoritmos de segmentación de imágenes se han dividido tradicionalmente en dos grandes grupos [Ballard y Brown 1982]: i) los que detectan bordes entre regiones en la imagen; y ii) los que tratan de encontrar regiones homogéneas. Los primeros se basan en la búsqueda de discontinuidades, incluyendo algoritmos para la detección de puntos aislados [Smith y Brady 1997], la extracción de líneas en la imagen [Canny 1983] y la búsqueda de curvas parametrizadas [Costa y Sandler 1993]; los segundos se basan en seleccionar una determinada característica distintiva de los píxeles de una imagen para dividirla en áreas homogéneas. Dado que los segundos son más robustos frente a posibles transformaciones, distorsiones y ruidos, esta sección contempla dicho grupo.

Los métodos basados en regiones se dividen a su vez en dos categorías: los métodos globales [Yanowitz y Bruckstein 1989], que se aplican sobre la imagen completa, como en el caso de los métodos morfológicos [Beucher 1990], y los métodos locales [Chiou y Hwang 1995] [Geman y Geman 1984] [Terzopoulos et al. 1987], que trabajan con porciones de ésta. También

existen métodos híbridos que emplean tanto información global como local, como los de crecimiento y división y mezclado [Pitas 1993]. Los métodos globales ofrecen mayor velocidad, dado que no deben preocuparse de detalles puntuales. En contrapartida, su precisión es menor y la resolución de las regiones resultantes es más pobre. Los métodos locales, además de ser más lentos, en ciertos casos pueden caer en trampas locales, dado que en el análisis pierden de vista la totalidad del conjunto. Los métodos híbridos, si bien generalmente son algo más lentos que los globales, ofrecen simultáneamente una visión tanto total como parcial, que les permite obtener una buena resolución sin perder de vista la globalidad de la imagen.

Tal como se ha comentado anteriormente, para proceder a la segmentación de una escena es necesario definir por un lado qué características o atributos gobernarán el proceso de segmentación, y por otro, la filosofía que se debe emplear para obtener las entidades de la imagen original. Las posibles características a utilizar en la segmentación pueden clasificarse en dos grandes grupos: i) estáticas, que son aquellas que se obtienen a partir de una única imagen; y ii) dinámicas, que son las obtenidas a partir de un conjunto de imágenes. Las características estáticas más comunes incluyen el brillo o nivel de gris, el color y la textura, a las que se les puede añadir el enfoque y la vergencia estática. En el segundo grupo se encuentran, principalmente, los parámetros obtenidos a partir del flujo óptico, como la divergencia, la traslación o el tiempo de contacto. Este grupo se contemplará en capítulos posteriores.

En general, la técnica de segmentación más extendida por su sencillez y velocidad es la segmentación por color o, en su caso, por nivel de gris. Desafortunadamente, en muchos casos esta técnica es insuficiente porque el nivel de gris del fondo no es homogéneo. Las técnicas de segmentación basadas en texturas [Patel y Stonham 1992] [Hepplewhite y Stonham 1997] permiten eliminar parcialmente este problema y distinguir el fondo independientemente de la presencia de sombras o materiales homogéneos que presentan colores variantes. No obstante, dicha técnica resulta considerablemente más lenta que la segmentación por color y sólo resuelve los casos mencionados. Una alternativa interesante se presenta cuando se dispone de las imágenes procedentes de dos cámaras. En estos casos, puede obtenerse un mapa de disparidad a partir de las posiciones relativas de ambas cámaras y estimar, por ejemplo, la distancia a éstas, con lo que es posible considerar como más interesantes las zonas más cercanas, frente a un fondo cuya distancia al observador es homogénea. Esta técnica se ha usado profusamente en aplicaciones de todo tipo, como cartografía a partir de fotografías aéreas, y su éxito radica en efectuar un control de vergencia [Rodríguez et al. 1999] para mantener el área de coincidencia de ambas imágenes en el porcentaje adecuado de acuerdo a la distancia al objeto de interés. La Fig. 2.3 muestra un ejemplo de esta técnica, donde los puntos situados en el margen superior de la imagen permiten

Figura 2.3: Paisaje estéreo

situar los ojos de la forma correcta para percibir una tercera dimensión.

Dado que se dispone de una única imagen, y por los criterios anteriormente expuestos de sencillez, velocidad y disponibilidad, se va a realizar a continuación un recorrido por distintos métodos que emplean como criterio de segmentación el nivel de gris de los píxeles de la imagen. En particular, este conjunto de métodos son la base de la mayoría de los algoritmos de segmentación más recientes, e incluyen tanto técnicas bidimensionales (umbralización, mezclado, crecimiento y división-mezclado), como técnicas jerárquicas.

2.1 Segmentación por umbralización

La forma más sencilla de segmentación por nivel de gris se conoce como umbralización y consiste en utilizar un umbral para clasificar los píxeles de acuerdo a su nivel de gris. La imagen segmentada $s(x, y)$ se obtiene usando la siguiente regla:

$$s(x,y) = \begin{cases} 1 & \text{si } I(x,y) > \phi, \\ 0 & \text{en otro caso.} \end{cases} \tag{2.1}$$

donde $I(x, y)$ es el nivel de gris de la imagen y ϕ el valor de umbral. Puede apreciarse que este método, si bien ofrece homogeneidad en las regiones si el umbral está correctamente fijado, no garantiza la conectividad entre éstas.

Para escoger un umbral ϕ adecuado suele usarse el histograma de la imagen, ya que de existir un objeto y un fondo bien diferenciados, éste presentará dos picos y ϕ se puede fijar al mínimo entre ambos. Obviamente, definiendo varios umbrales en vez de uno sólo, pueden obtenerse un mayor número de clases. La Fig. 2.4.a presenta una imagen con fondo diferenciado,

y en la Fig. 2.4.b aparece la segmentación en 4 clases que resulta de aplicar este método. El problema será determinar cuál es el número correcto de clases y dónde situar los umbrales para que la segmentación sea correcta. La Fig. 2.4.c muestra la segmentación de la Fig. 2.4.a, usando esta vez 15 umbrales equiespaciados para obtener 16 clases. A pesar de no haber fijado dichos umbrales de acuerdo a una estrategia de optimización, puede apreciarse que el fondo sigue diferenciándose correctamente.

El cálculo del mínimo del histograma para fijar el umbral no resulta sencillo cuando la imagen es ruidosa, dado que en este caso los picos no están bien definidos. Igualmente, si el fondo no es homogéneo, la segmentación no permite separarlo correctamente de los objetos.

2.2 Segmentación por mezclado

Otro ejemplo clásico es la segmentación por mezclado [Pitas 1993], que, en su variante más simple, se implementa mediante un recorrido sobre la imagen píxel a píxel. Durante este proceso, el valor de brillo de cada píxel se compara al de las regiones existentes y, de parecerse a alguna de éstas, se engloba en la región oportuna. En caso contrario, se genera una nueva región. Es necesario notar que, introduciendo las debidas modificaciones, este método puede preservar la conectividad de las regiones obtenidas, lo que no ocurría cuando se trabajaba con métodos de umbralización. El problema de este método radica en su lentitud y en que tiende a generar más clases de las necesarias cuando la imagen presenta cierta complejidad.

2.3 Segmentación por crecimiento

Para preservar la conectividad en la segmentación, el método más sencillo consiste en definir una serie de semillas, o píxeles situados en las distintas regiones de la imagen, y hacerlos crecer mientras el área sobre la que se extienden presente características homogéneas. Desafortunadamente, la segmentación por crecimiento exige, generalmente, una etapa supervisada para situar las semillas, y sólo detecta una región por semilla utilizada [Mehnert y Jackway 1997].

2.4 División y mezclado

Dentro de los algoritmos que pueden preservar la conectividad de las regiones obtenidas, un método más elaborado es el de división y mezclado [Pitas 1993] [Gonzalez y Wintz 1987]. Este algoritmo sigue una filosofía descendente ($top-down$) y parte de la premisa de que toda la imagen es homogénea. De no ser así, la imagen se divide en cuatro subimágenes. La homogeneidad de

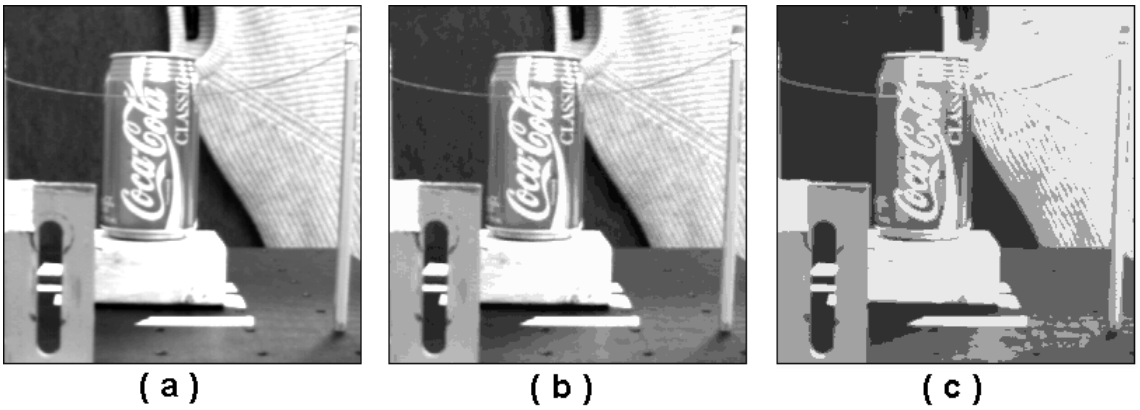


Figura 2.4: Segmentación por umbralización: a) imagen original; b) segmentación en 16 clases; y c) segmentación en 4 clases.

dichas subimágenes se estudia de forma similar y las subimágenes no homogéneas se vuelven a dividir en cuatro de forma recursiva, hasta obtener regiones homogéneas. Una vez concluido este proceso, la segmentación es claramente subóptima y, por ello, se incorpora una etapa de mezclado que consiste en unir aquellas clases en contacto cuyo valor de nivel de gris sea lo suficientemente similar. Es necesario notar que, para preservar la conectividad, se debe exigir a las clases una frontera común antes de fundirlas.

La Fig. 2.5 presenta un ejemplo de división y mezclado sobre una imagen plana en cuatro y dieciséis clases. Puede apreciarse que, en este caso, no se ha forzado la conectividad en las regiones obtenidas, obteniendo clases separadas que presentan el mismo nivel de gris.

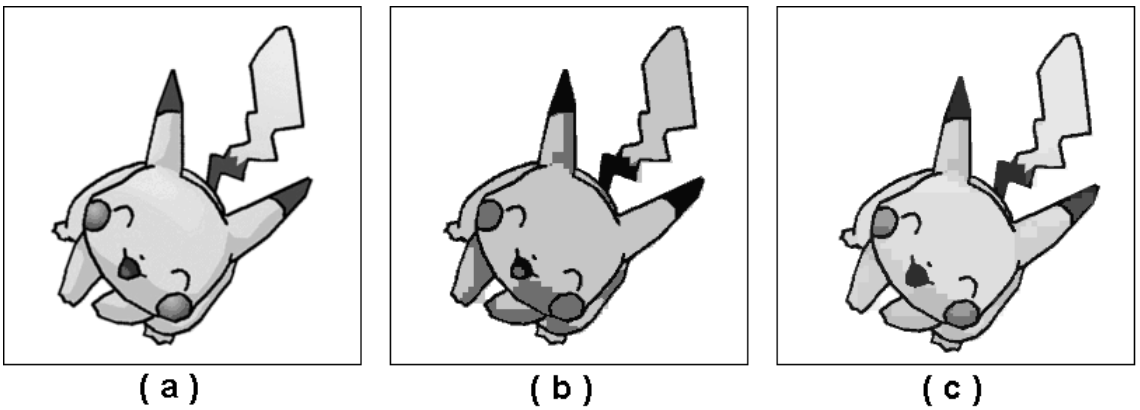


Figura 2.5: Segmentación por división y mezclado: a) imagen original; b) segmentación en 4 clases; y c) segmentación en 16 clases.

2.5 Segmentación jerárquica adaptativa

A efectos de mejorar la velocidad de los métodos previamente expuestos para aplicaciones en que el tiempo es un factor crítico, en [Tanimoto y Pavlidis 1975] se presenta una estructura jerárquica multinivel que permite establecer distintos niveles de resolución a partir de una imagen original que constituye la base de dicha estructura. Cada uno de los niveles de la mencionada estructura, denominada pirámide, presenta la mitad de resolución en cada eje sobre el inmediatamente inferior, y, por tanto, un cuarto de sus píxeles. Para generar la pirámide, se llevan a cabo los siguientes pasos:

1. Sea el nivel $l = 0$ la base de la estructura o imagen original.
2. Para generar el nivel $l + 1$, tómese cada conjunto de 2x2 píxeles del nivel l y calcúlese la media del nivel de gris de dicho conjunto. A partir de cada 2x2 píxeles se genera un nodo del nivel l que presenta el nivel de gris medio del conjunto. A efectos de relacionar cada nodo con el área del nivel inferior a partir de la cual se generó, pueden establecerse enlaces entre dicho nodo y cada uno de los 2x2 nodos utilizados. El nodo generado se conoce como padre, mientras que aquellos a los que está enlazado reciben el nombre de hijos.
3. Mientras no se alcance el nivel de trabajo deseado -que viene marcado por el número de nodos con los que el sistema pretende trabajar-, sea $l = l + 1$ y repítase el paso 2.

Se puede notar que, si bien puede construirse una pirámide hasta su cúspide, donde sólo existiría un único nodo cuyo nivel de gris sería una media de la imagen completa, generalmente el proceso de construcción suele interrumpirse antes, ya que se trabaja generalmente con pirámides truncadas.

La Fig. 2.6.a muestra una estructura de este tipo construida sobre la imagen de la Fig. 2.6.b. Las Figs. 2.6.c-g presentan sucesivos niveles de la estructura, donde se aprecia la pérdida de resolución conforme el número de nodos del nivel disminuye.

De esta forma, si los recursos del sistema no son suficientes para procesar la totalidad de los píxeles de una imagen, se puede trabajar a menor resolución y llevar a cabo el proceso de segmentación en niveles superiores, lo que permite reducir la carga computacional del proceso, si bien también se reduce la precisión de éste. La Fig. 2.7 presenta un ejemplo de este método, donde la imagen de la Fig. 2.7.a se segmenta por división y mezclado en la base (Fig. 2.7.b), en el nivel 64x64 (Fig. 2.7.c) y en el nivel 16x16 (Fig. 2.7.d). Los tiempos de proceso para los tres niveles, utilizando un Pentium 200 MMX, son de 150 seg., 2 seg. y 50 mseg. respectivamente,

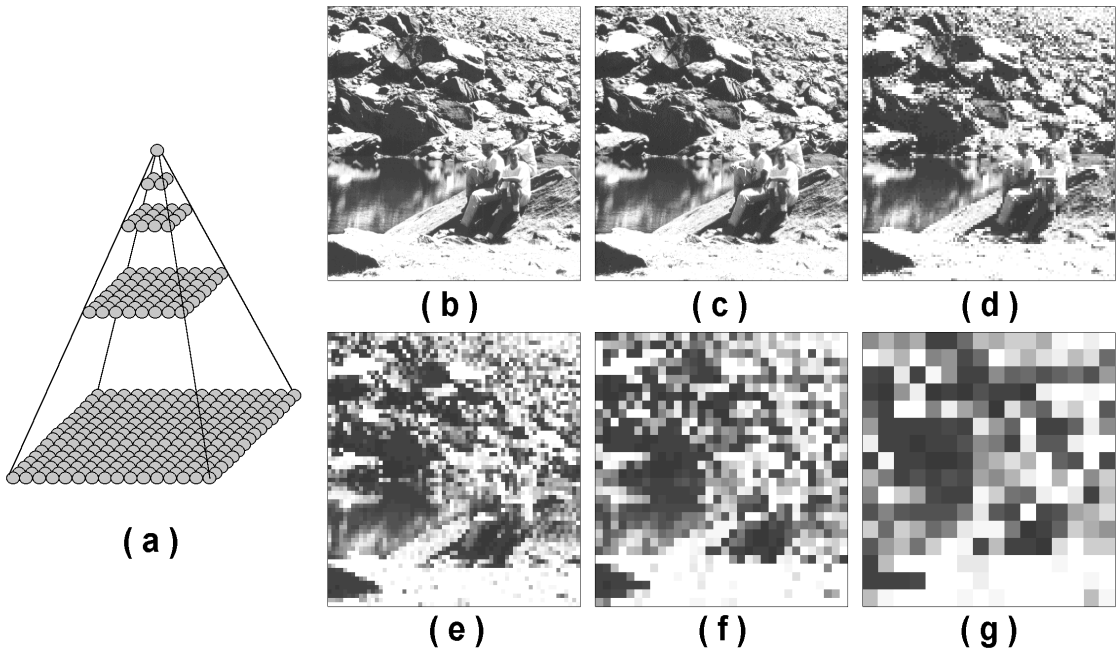


Figura 2.6: Pirámide: a) estructura piramidal; b) nivel 512x512; c) nivel 256x256; d) nivel 128x128; e) nivel 64x64; f) nivel 32x32; g) nivel 16x16.

y, aunque la disminución de tiempo es importante, puede apreciarse que va en detrimento de la calidad de forma muy evidente.

Así, la principal desventaja de este método reside en que la sucesiva pérdida de resolución en los niveles altos de la pirámide influye en un decremento de la precisión del método en función del nivel de trabajo. En muchos casos, los resultados no son adecuados para un posterior procesado, particularmente si el descarte de regiones del fondo se basa en criterios distintos al de color, como puede ser la forma.

Para aprovechar las ventajas de la estructura multinivel propuesta sin sufrir las pérdidas de resolución que implica el trabajo en niveles superiores de la pirámide, Burt propuso la técnica de reenlazado adaptativo [Burt et al. 1981a] [Burt et al. 1981b], que aprovecha la estructura de enlaces que relaciona la información entre niveles para definir de forma implícita una segmentación del entorno. El método pretende que, en lugar de definir un nodo por cada cuatro del nivel inferior independientemente de sus características, se defina un nodo por cada región homogénea de dicho nivel, a pesar de que la forma de dicha región deje de ser regular y el número de nodos que la forman pase a ser variable. Para conseguir dicho objetivo, se reasignan los enlaces entre cada dos niveles, comenzado por la base, y se recalculan los valores de gris de los padres de los niveles que se están estabilizando. Cuando dichos valores dejan de cambiar, es porque los hijos a los que están enlazados presentan un valor de gris homogéneo y muy simi-

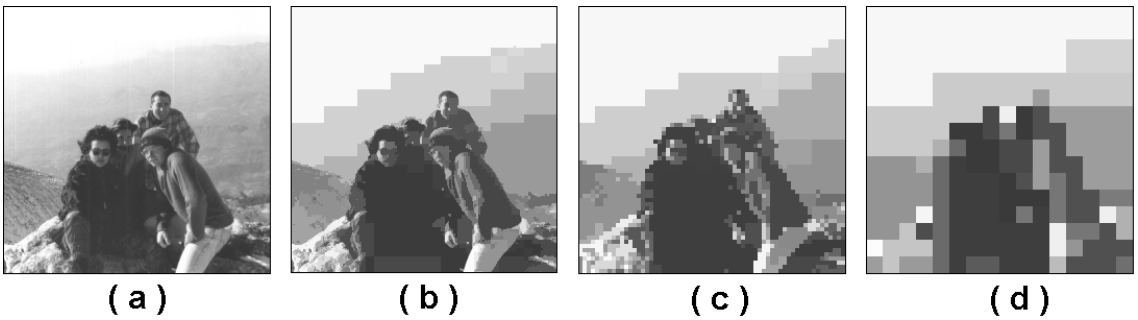


Figura 2.7: Segmentación por división y mezclado a distintos niveles de resolución: a) imagen original; b) segmentación en el nivel 256x256; c) segmentación en el nivel 64x64; d) segmentación en el nivel 16x16.

lar al del padre, ya que en caso contrario tratarían de enlazarse a otros padres cercanos. Así, cuando el proceso concluye, cualquier nodo de la estructura debe estar enlazado a una región de píxeles homogénea en la base. Es importante notar que este proceso no garantiza que el número de regiones en la base sea el correcto -de hecho, dicho número dependerá de cuántos nodos se estén estudiando en la estructura -, sólo que dichas regiones son homogéneas. En general, el procedimiento más habitual es cortar la estabilización a un nivel determinado -habitualmente 8x8 ó 16x16- y trabajar con las regiones que definen los nodos de dicho nivel, que serán tantas como nodos presente. Un paso posterior de mezclado permite obtener el número real de regiones mediante la unión de aquellas que estén en contacto y presenten un nivel de gris muy similar.

Particularizando, el proceso de segmentación por enlazado adaptativo se divide en los siguientes pasos:

1. Sea el nivel $l = 0$, que corresponde con la imagen original o base de la estructura.

2. Para estabilizar el nivel $l + 1$, tómese cada hijo del nivel l, búsquese el padre entre los 2x2 nodos del nivel $l + 1$ situados inmediatamente sobre dicho hijo y rómpase su enlace, si procede, para unirlo al padre de nivel de gris más parecido.

3. Una vez todos los hijos han sido reenlazados, recalcúlese el nivel de gris de todos los padres del nivel $l + 1$ en función de los hijos que ahora se encuentren enlazados a ellos. Nótese que, en lugar de 4, ahora cada padre puede estar enlazado a un máximo de 16 hijos.

4. Si el nivel de gris de los padres no ha cambiado, sea $l = l + 1$ y, si se desea, procédase a estabilizar el siguiente nivel retornando al paso 2. En caso contrario, repetir los pasos 2 y 3 hasta que esta condición se cumpla.

5. Una vez se ha alcanzado el nivel de trabajo, estúdiese a qué regiones de píxeles en la base están enlazados cada uno de los nodos de este nivel. Este paso es inmediato, ya que los enlaces redefinidos ligan implícitamente a cualquier nodo de cualquier nivel con una región homogénea en la base.

6. Si se desea, se puede realizar un mezclado, que funda las clases que representan regiones en contacto cuyo nivel de gris sea similar, para obtener el número correcto de dichas regiones.

Nótese que el último paso es muy rápido, ya que sólo se trabaja con un número de regiones igual al de nodos del nivel de trabajo. La Fig.2.8 muestra un ejemplo de los resultados del enlace adaptativo en la imagen de la Fig.2.8.a. Las Figs.2.8.b y c muestran el nivel 8x8 de la pirámide construida sobre dicha imagen antes y después de su estabilización, respectivamente. Mientras que las regiones en la base asociadas al nivel de la Fig.2.8.b son regulares y presentan el mismo número de nodos sin criterio de división alguno (Fig.2.8.d), las regiones enlazadas al nivel de la Fig.2.8.c son homogéneas y heredan el nivel de gris del nodo perteneciente al nivel 8x8 al que están enlazadas (Fig.2.8.e).

A pesar de esta iteratividad, la convergencia hacia una solución estable se alcanza rápidamente [Cibulskis y Dryer 1984], ofreciendo además mejor comportamiento y resultados que las técnicas convencionales de segmentación 2D [Bandera 1994] [Hird y Wilson 1989]. El principal problema de este método es que no se garantiza que la región enlazada a un determi-

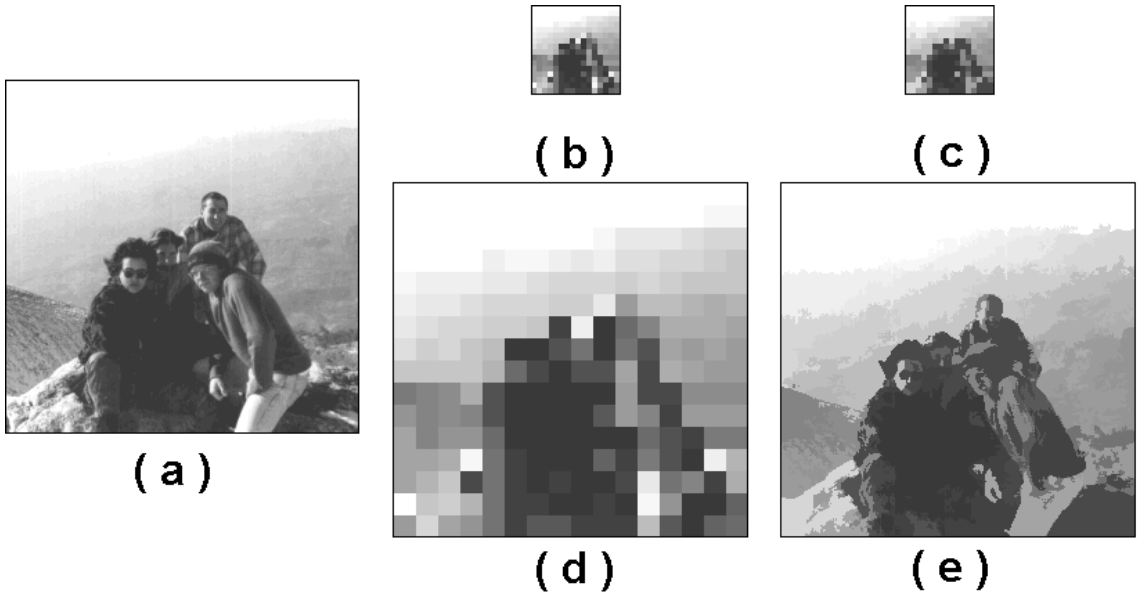


Figura 2.8: Segmentación jerárquica adaptativa: a) imagen original; b) nivel 16x16 no adaptado; c) nivel 16x16 adaptado; d) propagación del nivel b) en la base; e) propagación del nivel c) en la base.

nado nodo sea conexa, especialmente en los niveles superiores. Para paliar este efecto, puede incluirse un factor de peso que penalice el cambio de enlace cuando un nodo intenta unirse a otro físicamente alejado de él [Urdiales 1999]. No obstante, en ningún caso se podrá eliminar completamente el efecto comentado.

3 Detección de objetos

Tras separar una imagen cualquiera en un conjunto de regiones, es necesario determinar cuáles de dichas regiones constituyen realmente objetos y seleccionarlas para su posterior análisis. Así, se pueden concentrar los recursos del sistema sobre dichas zonas y obviar el procesamiento del fondo de la escena.

Las distintas técnicas de detección de objetos se diferencian básicamente por los criterios de modelado y las estrategias de trabajo. Los criterios de modelado hacen referencia a las características a que se presta atención a la hora de establecer si un determinado conjunto de píxeles es o no un objeto. Las estrategias de trabajo condicionan la filosofía que sigue el algoritmo de detección: o bien se asume que una región es un objeto, y trabajando de forma descendente se confirma o descarta la hipótesis; o bien, se trabaja de forma ascendente extrayendo información de la región bajo estudio para determinar si es o no un objeto. Habitualmente, se opta por descartar regiones que no cumplen los requisitos necesarios para ser un objeto en lugar de trabajar al contrario, ya que basta con que no se cumpla una sola de estas características para llevar a cabo el descarte, mientras que habría que evaluar un gran número de ellas para admitir una región como objeto.

3.1 Criterios de descarte

Un objeto puede caracterizarse en función de modelos descriptivos -que prestan atención a características cuantificables, como el brillo, la textura o el movimiento-, o en función de criterios proposicionales -que son un conjunto de propuestas relacionales del objeto frente a su entorno que pueden o no cumplirse en un momento dado-. Esta división se ha efectuado a partir del estudio de cómo el ser humano percibe el mundo [Johnson-Laird 1980], si bien actualmente la psicología parece decantarse por un sistema híbrido que combina ambas formas de representación.

Según se describe en [Kosslyn y Schwartz 1977], la representación descriptiva posee las siguientes características:

- Coherencia: todos los elementos de la escena representada aparecen a la vez, así como todas las relaciones entre cada elemento y el resto.
- Continuidad: generalmente, los descriptores de un objeto se mantienen constantes frente a cambios suaves y continuos de la escena.
- Analogía: el modelado descriptivo trata de conseguir una analogía total entre el objeto real y su modelo.
- Simulación: los modelos descriptivos son manipulados mediante procesos computacionalmente tan complejos que, a menudo, tienen que llevarse a cabo por medio de simulación.

Por otra parte, las características de la representación proposicional son [Palmer 1975]:

- Dispersión. Un elemento de la situación representada puede aparecer en distintas proposiciones. La coherencia en las representaciones se consigue con el uso de redes semánticas.
- Carácter discreto. Las proposiciones no son generalmente usadas para representar cambios continuos, ya que expresan la situación de la escena en un instante de tiempo determinado. Sin embargo, podrían construirse de forma que en ellas se reflejen cambios suaves y continuos.
- Abstracción. Las proposiciones son ciertas o falsas. No tienen una relación geométrica con la situación, su estructura no es análoga a la situación presente.
- Inferencia. Los modelos proposicionales se manipulan siguiendo unas 'reglas de inferencia', que permiten que nuevas proposiciones se desarrollen partiendo de las viejas.

La principal ventaja de la representación descriptiva es que el sistema no necesita ningún tipo de inteligencia específica para llevarla a cabo con éxito, mientras que una representación proposicional exige comprensión de la escena y formulación de relaciones entre las distintas entidades que la componen.

3.2 Detección de objetos sobre una estructura multirresolución

Tal como se ha comentado previamente, para acelerar el proceso de segmentación, éste se puede llevar a cabo mediante una estructura tridimensional denominada pirámide. Los sucesivos niveles de esta estructura presentan la imagen original a una resolución progresivamente menor hasta alcanzar la cima de la misma. El método de segmentación descrito se basaba en estabilizar

los enlaces entre los niveles de la estructura de forma adaptativa para que, una vez concluido el proceso, cada nodo del nivel de trabajo se encontrase enlazado a una región relativamente homogénea en la base. El algoritmo de detección de objetos que se presenta en este apartado se fundamenta en estudios previos [Arrebola 1998] [Bandera 2000] [Bandera et al. 2000a], trabaja de forma descendente y se basa tanto en criterios descriptivos como en criterios proposicionales.

En principio, se asume que cada una de las regiones enlazadas a los nodos del nivel de trabajo es un objeto distinto. Así pues, habrá que analizar hasta un total de M posibles objetos, siendo M el número de nodos del nivel de trabajo. En tanto que las técnicas relacionales implican una mayor carga computacional, parece razonable llevarlas a cabo en los niveles altos de la estructura, donde la instancia del problema es más reducida, mientras que la representación descriptiva puede llevarse a cabo en la base para mayor seguridad a la hora de determinar si una región es o no un objeto. A continuación, se presentan los dos criterios que se usan en la detección.

- Descarte por contraste.

  Dentro de las reglas proposicionales, la más sencilla que puede estudiarse se basa en la vecindad: un objeto se caracteriza por ser distinto del fondo u objetos que lo rodean. Esta proposición puede estudiarse de forma simple en niveles altos de la pirámide mediante comparación en vecindad 8 del nivel de gris de los nodos. Así, un nodo cuyo nivel de gris sea significativamente distinto de los que lo circundan se considera un objeto potencial, mientras que los que presenten niveles de gris similares a los de su entorno se descartan. Es necesario notar que cada región corresponde a un nodo distinto y que este paso asume que el fondo presenta una cierta homogeneidad. En fondos con mucha variación, apenas se descartarían regiones mediante este paso. El objetivo global del descarte por contraste es obviamente reducir el número de nodos a estudiar mediante procesos descriptivos para acelerar la detección en niveles bajos de la estructura.

- Descarte por área y dispersión.

  Después del descarte inicial de regiones, los dos procedimientos restantes se llevan a cabo en la base de la estructura y sólo sobre los objetos potenciales restantes, aumentando así la velocidad del proceso global. Estos procesos se basan en asumir que los objetos deben ser compactos y presentar un tamaño significativo, y para comprobar que cumplen estas premisas es necesario calcular las $bounding - boxes$ de las regiones candidatas. El cálculo de dichas $bounding - boxes$ es sencillo, pues basta tomar las coordenadas x e y máximas y mínimas de los píxeles que componen la región. A partir de ahí, una región se descarta

si:

- Su área está por debajo de un determinado umbral.
- La relación entre el área de la región y la de su $bounding-box$ está por debajo de un determinado umbral [Bandera 1994].

El objetivo de este proceso es incrementar la certeza de que una determinada región sea un objeto, independientemente de su nivel de gris.

Para ilustrar el funcionamiento del sistema de detección descrito, se ha aplicado el método de detección de objetos sobre la imagen de la Fig. 2.9.a. Tal como puede apreciarse, si se trabaja en el nivel 8x8 (2.9.b) se obtienen 64 clases en la base (2.9.c), que pueden relacionarse con los objetos a detectar, aunque también incluyen brillos, texturas y objetos complejos, como el cajón de madera de la parte inferior izquierda, que dificultan la detección.

La Fig. 2.10 muestra algunos resultados del proceso de descarte. Los nodos marcados con una cruz están relacionados en la base con las regiones indicadas como dispersa y no conexa en la Fig. 2.10.b. Ambas se descartan de acuerdo a los criterios expuestos anteriormente. Si se procesan de la misma forma el resto de las regiones enlazadas a los nodos del nivel 8x8, sólo quedan como potenciales objetos los nodos marcados con una circunferencia en la Fig. 2.10.a. Las regiones enlazadas a esos nodos aparecen enmarcadas por su $bounding-box$ correspondiente en la Fig. 2.10.b. Como puede observarse, corresponden efectivamente a objetos sobre el fondo de la escena.

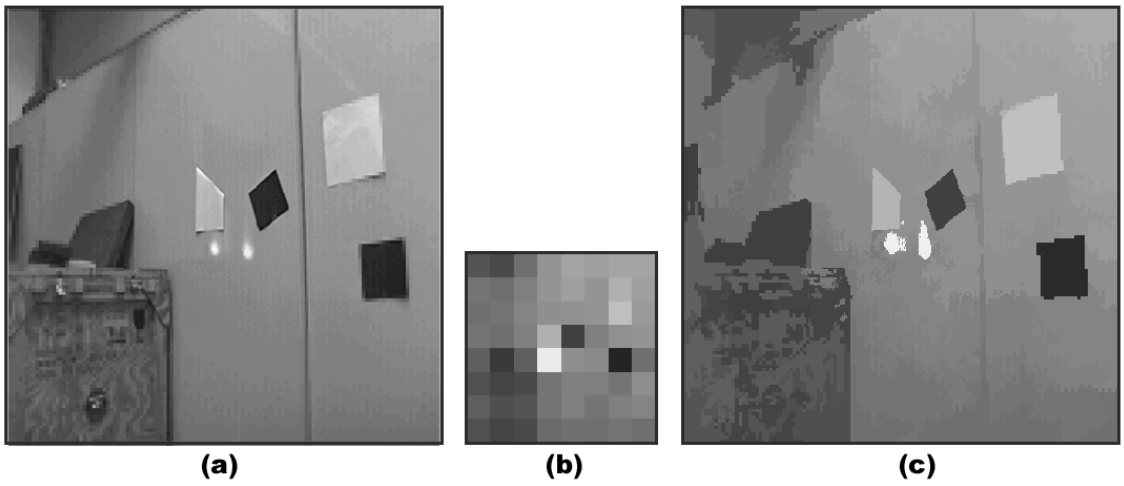


Figura 2.9: Proceso de segmentación jerárquica multirresolución: a) imagen original; b) nivel 8x8 estabilizado de la pirámide; y c) segmentación en la base.

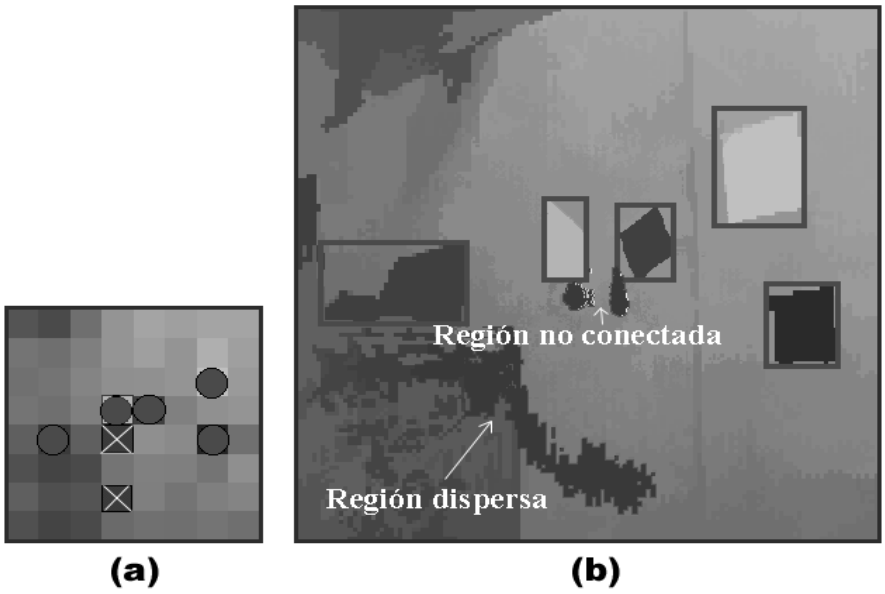


Figura 2.10: Proceso de segmentación jerárquica multirresolución: a) nivel 8x8 estabilizado de la pirámide; b) segmentación en la base y detalle de objetos detectados.

4 Resultados

A efectos de probar su validez, el algoritmo de detección de objetos a partir de una segmentación jerárquica adaptativa basada en niveles de gris, se ha implementado sobre un entorno de pruebas real con dos grados distintos de complejidad. Por una parte, se dispusieron formas de color homogéneo sobre un armario metálico. Si bien existen los problemas intrínsecos a entornos reales como las sombras, los cambios de iluminación, el ruido de captura o texturas y los reflejos de los diferentes materiales, resulta evidente que el problema de segmentación y descarte de regiones potencialmente pertenecientes al fondo se ve muy simplificado gracias a las condiciones controladas de este entorno. Por otra parte, se capturaron imágenes reales con objetos tridimensionales distribuidos al azar sobre un fondo no homogéneo. En este caso, no se impuso ninguna restricción al problema.

Las imágenes fueron capturadas mediante una cámara CCD KP-D50 soportada por un sistema Zebra de control pan-tilt-vergencia. La imagen analógica resultante se enviaba a una tarjeta capturadora TMS320C44 sobre una placa HECFG44 que, conectada mediante un bus ISA a un Pentium PC a 133 MHz, suministraba una imagen digitalizada.

La Fig. 2.11 muestra algunos de los resultados obtenidos en el primer entorno de pruebas, con distintos objetos dispuestos sobre el mencionado armario metálico. Como puede ob-

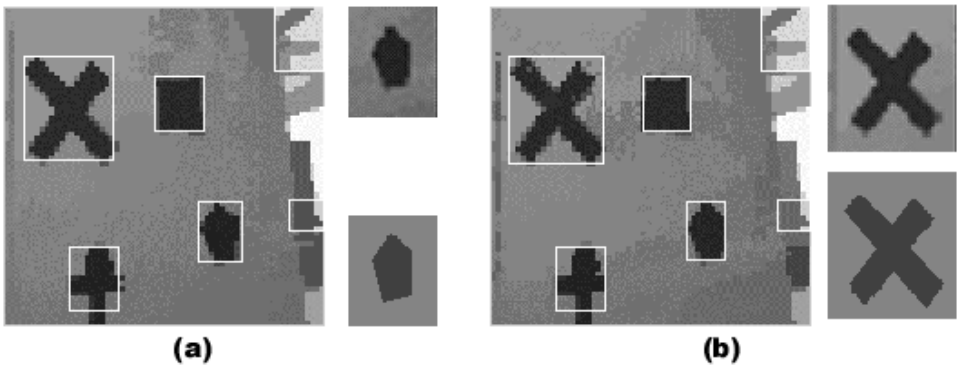


Figura 2.11: Ejemplo de detección sencilla de objetos: a) pentágono; y b) aspa.

servarse, todos los objetos dispuestos sobre la pared son correctamente detectados, quedando sus $bounding-boxes$ dibujadas sobre las imágenes de las Figs. 2.11.a y b, que corresponden a capturas realizadas sobre el mismo entorno pero en distintos instantes de tiempo. Las posiciones y tamaños de estas $bounding-boxes$ no varían significativamente de una a otra imagen, demostrándose así que el proceso es robusto frente a potenciales variaciones de las condiciones de captura como la iluminación o el ruido. No obstante, tal como se ha mencionado, esta conclusión es aplicable únicamente a las condiciones del experimento, como se mostrará más adelante. Junto a la imagen segmentada aparece, en cada caso, el detalle de un objeto seleccionado manualmente (un pentágono en la Fig. 2.11.a y un aspa en la Fig. 2.11.b), así como su versión segmentada. Puede apreciarse que, independientemente de los gradientes de color que aparecen en el fondo, debido al efecto de la luz sobre la superficie metálica, las segmentaciones son correctas en ambos casos y los descartes de regiones se producen de la forma esperada. Una excepción son las $bounding-boxes$ que se detectan junto al borde derecho de ambas imágenes, donde se acaba la superficie metálica y comienza a vislumbrarse el resto de la habitación. En estos casos, se han producido dos falsas detecciones debido a la complejidad mucho mayor de esa porción de la escena.

Las Figs. 2.12 y 2.13 muestran dos ejemplos de segmentación y descarte de regiones en una escena que presenta mayor complejidad que la anterior. En estos casos, al no existir un fondo predominante, el sistema de descarte confunde algunas porciones del mismo con potenciales objetos al basarse en los criterios de descarte establecidos. En ambas imágenes se puede apreciar cómo los objetos se funden con su entorno al estar formado el fondo por un conjunto de regiones de diversos niveles de gris. De esta forma, el proceso de detección sólo llega a identificar distintas porciones de los objetos o del fondo, sin realizar una auténtica identificación de los objetos en la escena. Así, en la Fig. 2.12.b sólo se han detectado regiones dispersas repartidas en la totalidad de la escena. Por otro lado, en la segmentación y posterior detección de los objetos

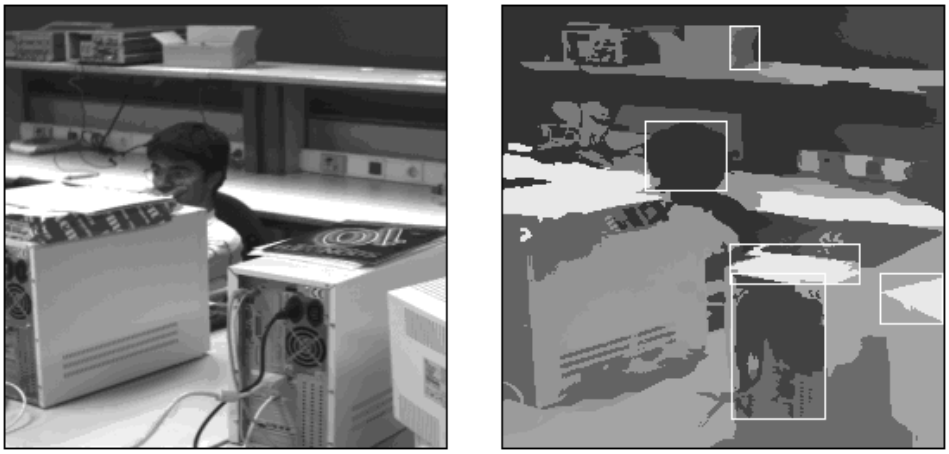


Figura 2.12: Ejemplo de segmentación compleja: a) imagen original; b) objetos potenciales.

que se presenta en la Fig. 2.13.b, el proceso de fusión de los mismos con el fondo de las regiones que se encuentran en primer plano es aún más apreciable. En este caso, las únicas regiones detectadas deberían descartarse por formar parte del fondo, mientras que el objeto más significativo (ordenador) se fragmenta y funde con un conjunto de regiones que conforman gran parte del fondo.

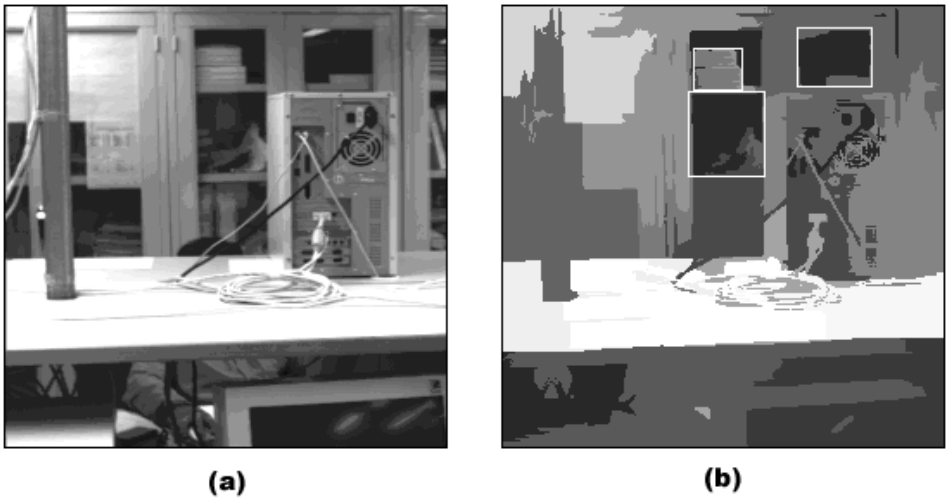


Figura 2.13: Ejemplo de segmentación compleja: a) imagen original; b) objetos potenciales.

5 Conclusiones

En este capítulo se han presentado los procesos necesarios para detectar zonas de interés en una imagen cualquiera obtenida a partir de una cámara estática. Dichas zonas de interés se seleccionan mediante segmentación de la imagen en regiones y posterior descarte de aquellas que presuntamente pertenecen al fondo, basándose en la diferencia de nivel de gris que presentan los objetos. A este respecto, se han analizado varios métodos, entre los que la segmentación jerárquica parece ofrecer las mejores prestaciones, tanto en velocidad como en estructuración de las regiones resultantes. Debido a los problemas de dispersión que resultan de segmentar imágenes reales, parece necesario introducir un factor de peso en la estabilización de la estructura de datos, para que las regiones obtenidas sean conexas en la medida de lo posible [Bandera 2000]. Este factor de peso no soluciona totalmente el problema, pero se consigue que las regiones no conexas que se encuentran enlazadas a un mismo nodo se hallen lo más cerca posible, con lo que la potencial región de interés resultante ocupará un área tan reducida como sea posible en función del tamaño del objeto que englobe. Posteriormente, se aplica un proceso de descarte que permite discriminar con algunas reglas simples cuáles de las regiones disponibles pertenecen al fondo. Por eliminación, el resto de la imagen está formado por zonas de interés.

Si bien el procedimiento es aplicable a casos simples, con imágenes reales resulta demasiado complicado extraer el fondo de forma no supervisada y mediante procedimientos lo suficientemente simples como para permitir velocidades de proceso adecuadas. Así, si bien la segmentación por niveles de gris de una imagen es rápida y sencilla, sólo ofrece resultados aceptables si el entorno presenta una serie de restricciones específicas, lo que habilita su uso únicamente bajo condiciones controladas. Por tanto, salvo que exista un fondo homogéneo, ya sea éste fácilmente separable de los objetos de interés o no, no podrá recurrirse a este tipo de técnicas. De hecho, otros métodos de segmentación de imagen como los basados en texturas tampoco han mejorado la calidad de los resultados en entornos complejos, ya que resulta un problema conceptualmente imposible el separar objetos que presentan texturas distintas, sin recurrir a criterios no extraibles de una única imagen, como puede ser la profundidad o el movimiento [Bandera 2000].

En conclusión, es necesario recurrir a procesos mucho más complejos e inteligentes que permitan un descarte adecuado de regiones en función del entorno de trabajo, o a métodos supervisados y, por tanto mucho menos versátiles, si se desea detectar automáticamente las áreas de interés de una imagen real. Además, independientemente de la técnica elegida, en ocasiones será completamente imposible llevar a cabo el proceso de segmentación y descarte,

ya que la información necesaria para ello no se encuentra en ningún caso en una única imagen. En estos casos, es necesario trabajar con señales de vídeo en lugar de imagen estática, lo que permite evaluar de forma más inmediata qué regiones son interesantes en función de su velocidad y demás factores dinámicos, tal como se presentará en los siguientes capítulos.

MENU SALIR

Capítulo 3

Segmentación espacial en el tiempo

1 Introducción

La segmentación de una imagen consiste en dividir ésta en un conjunto de regiones de características homogéneas. Por tanto, la propia definición del proceso implica un profundo análisis previo de qué características son las que determinan la identidad de una región, de forma que su estudio permita extraer, por ejemplo, los distintos objetos ubicados sobre un fondo. En el capítulo 2 se empleó como característica fundamental de cada píxel su nivel de gris, y, si bien los resultados obtenidos son correctos cuando en la imagen aparecen regiones relativamente uniformes, se pudo concluir que esta única fuente de información resulta generalmente insuficiente para el análisis de escenas reales. Para aumentar el número de características, y conseguir así un algoritmo de segmentación más robusto, de la imagen se suele extraer información acerca de la profundidad, movimiento, forma, color o textura de las distintas regiones. Algunas de estas características no se pueden extraer de una simple imagen, requiriendo disponer bien de un sistema de visión estéreo, o bien de una secuencia de imágenes.

El movimiento es una importante característica empleada por un gran número de seres vivos para extraer los objetos de interés del fondo no relevante [Gonzalez y Wintz 1987]. En un experimento llevado a cabo en la Universidad de Cornell, Gibson [Gibson et al. 1959] demostró que los seres humanos empleaban el movimiento como una de las características fundamentales para la comprensión de una determinada escena. El empleo de esta característica en la segmentación de imagen es clásico en navegación autónoma [Araújo et al. 1998], vigilancia y supervisión de tráfico [Smith y Brady 1995], simulación de objetos tridimensionales [Iiyama et al. 2000] o compresión de secuencias de vídeo [Konrad y Dufaux 1998]. En todas estas aplicaciones se requiere que cada uno de los píxeles de la imagen sea identificado y clasificado

como móvil, estático o, en algunos casos, sombra (p. ej., las tareas de vigilancia o detección de obstáculos), o que, incluso, se mida cuantitativamente dicho movimiento (p. ej., las aplicaciones de compresión de vídeo). La primera de estas tareas se conoce como detección de movimiento, mientras que la segunda se define como estimación de movimiento.

En este capítulo se analiza una propuesta de sistema de segmentación basada en el estudio del movimiento en un entorno simplificado, en el cual el fondo es estático y sólo los objetos son móviles. El capítulo se ha dividido en dos bloques básicos, que se corresponden con los procesos de detección y estimación, descritos en los apartados 2 y 3. Para cada uno de estos dos bloques se describirán tanto el algoritmo finalmente empleado, como las alternativas más significativas que se pueden encontrar actualmente. En los apartados 4 y 5 se presentan los resultados obtenidos con el presente sistema y las conclusiones extraídas.

2 Detección de movimiento

Si se compara con la tarea de estimación, la detección del movimiento es, en la mayoría de los casos, una tarea simple tanto algorítmica como computacionalmente. El objetivo es identificar qué píxeles o regiones de la imagen se han desplazado entre dos instantes de tiempo. Los métodos que se comentan en este apartado son aplicados comúnmente a imágenes con fondo estático, aunque se podrían extender a secuencias donde el fondo presente un movimiento uniforme, estimable globalmente y, por tanto, compensable.

Resulta importante destacar que el desplazamiento de los píxeles de una imagen, aunque no sea perceptible directamente, se puede obtener de los cambios de intensidad respecto a la imagen capturada en un determinado instante anterior. Sin embargo, no todo cambio de intensidad se debe asociar a un desplazamiento, ya que puede ser debido al ruido de adquisición de la cámara o a cambios de iluminación, debidos a la fuente de la misma o a la presencia de sombras. Además, un determinado desplazamiento podría no verse reflejado en ningún cambio de intensidad, o en uno excesivamente pequeño. Estos factores son los que pueden convertir esta tarea, inicialmente simple, en un problema complejo.

Los métodos de detección de movimiento se dividen en dos grandes grupos [Konrad 2000], en función de si tienen en cuenta o no una determinada hipótesis inicial acerca de la distribución del movimiento en la escena. Esta hipótesis consiste, en la mayoría de los casos, en suponer que los objetos móviles son entidades compactas, mientras que el fondo constituye otra región igualmente uniforme. Esta simple condición va a permitir eliminar gran parte del ruido introducido

en el proceso de detección por la cámara o el propio entorno.

En los siguientes subapartados se muestran algunos de los algoritmos de detección de movimiento más comunes, tanto aquellos que no asumen ningún conocimiento espacial o temporal previo sobre la escena (métodos puntuales de detección), como los que sí lo hacen (métodos locales). En el subapartado 2.3, se propone el empleo de un método mixto de detección, que inicialmente procesa la imagen usando una máscara puntual para, posteriormente, procesarla ponderando espacialmente la distribución de píxeles móviles. La principal novedad del método radica en que, además de llevar a cabo la detección de movimiento, segmenta simultáneamente la escena, dando como resultado un conjunto de objetos o regiones.

2.1 Métodos puntuales de detección de movimiento

Los métodos puntuales de detección de movimiento asumen que todo cambio de intensidad que se produce en la escena es causado por un desplazamiento. Para tratar de reforzar esta hipótesis, y así eliminar los ligeros cambios de iluminación o el ruido que se puedan introducir entre dos fotogramas consecutivos de una determinada secuencia, estos métodos emplean un valor de umbral, que determina si el cambio de intensidad de un píxel es significativo o no.

La mayoría de los métodos puntuales de detección de movimiento se basan en el empleo del valor absoluto de la diferencia (VAD) entre fotogramas consecutivos o en el empleo de un fondo de referencia. Los métodos que detectan el movimiento mediante la substracción de un fondo de referencia resultan idóneos para la identificación de objetos móviles sobre un fondo que se conoce con antelación y que no cambia con el tiempo. Sin embargo, cuando se producen cambios de fondo o de iluminación, hay que recurrir a métodos de adaptación más complejos y costosos. Estos métodos de adaptación analizan el comportamiento dinámico de las regiones de la imagen, tratando de identificar las que pertenecen al fondo en cada instante. Ya que la identificación de estas regiones se basa en el propio método de substracción, se pueden producir falsas detecciones y éstas pueden propagarse temporalmente. Por otra parte, los métodos de detección por diferencia entre fotogramas consecutivos de una secuencia se adaptan de forma rápida y automática a los cambios de iluminación, pero presentan el problema de que si un objeto deja de moverse durante un par de fotogramas consecutivos, éste se integra automáticamente en el fondo. A continuación, se presenta la filosofía básica de ambos algoritmos de detección de movimiento. Cabe reseñar que existen algoritmos híbridos [Amamoto y Matsumoto 1997][Huwer y Niemann 2000], que combinan diferencias temporales con estimación adaptativa de fondo para detectar las regiones móviles, pero que, dado que combinan conceptos propios de estos dos métodos, no serán

analizados separadamente.

2.1.1 Detección de movimiento por diferencia

Una de las aproximaciones más simples para detectar cambios entre dos fotogramas, $I(x, y, t)$ y $I(x, y, t-1)$, tomados en los instantes de tiempo t y $t-1$, respectivamente, consiste en comparar las dos imágenes píxel a píxel. La forma más simple de llevar a cabo esta comparación es usando el método de la imagen diferencia.

Si se dispone de una imagen de referencia, que contenga los componentes estáticos de la escena, y se compara con otra imagen de la misma escena pero que incluya un objeto móvil, la diferencia de las dos imágenes cancelará las componentes estacionarias, de manera que sólo las componentes dinámicas de la imagen darán lugar a píxeles no nulos en la imagen diferencia resultado del proceso de substracción. Esta imagen diferencia, $I_{dif}(x, y, t)$, entre dos fotogramas capturados en los instantes de tiempo t y $t-1$ vendrá definida por:

$$I_{dif}(x, y, t) = \begin{cases} 1 & \text{si } |I(x, y, t-1) - I(x, y, t)| > U_{dif}, \\ 0 & \text{en otro caso.} \end{cases} \tag{3.1}$$

donde U_{dif} es el umbral que determina si la diferencia entre píxeles es significativa o no, e $I(x, y, t)$ e $I(x, y, t-1)$ son las imágenes adquiridas en los instantes de tiempo t y $t-1$. Sin embargo, hay que destacar que, en este caso, ninguna de las dos imágenes tiene por qué ser la imagen de referencia ideal, que sólo contiene componentes estáticos. Esta falta de referencia será el origen de todos los problemas asociados a este método.

Por lo tanto, en la imagen $I_{dif}(x, y, t)$ se marca con valor uno el resultado del movimiento en la escena. Este método de detección de movimiento sólo resultará aplicable si las dos imágenes se capturan en condiciones de iluminación similares y si la velocidad del móvil está en consonancia con la velocidad de adquisición [Gonzalez y Wintz 1987]. En caso contrario, como se muestra en la Fig. 3.1, la imagen diferencia o máscara de movimiento presentará un comportamiento excesivamente ruidoso, difícilmente subsanable con variaciones del parámetro U_{dif}. Además, dicha máscara de movimiento mostrará, más que un objeto definido, los bordes de éste, procedentes, como se observa claramente en la Fig. 3.1.d, de los fotogramas tomados en los dos instantes de tiempo. Este hecho se relaciona con el conocido problema de la apertura, que se refiere a que si separamos demasiado en el tiempo los fotogramas empleados para determinar el movimiento, la búsqueda de una misma región en ambos fotogramas debe ampliarse a vecindades cada vez mayores, aumentando la posibilidad de que ocurran falsas identificaciones. En este caso, puede ocurrir que un único objeto esté doblemente representado en la máscara

binaria resultante del proceso de diferencia absoluta y umbralización.

2.1.2 Detección de movimiento por sustracción de fondo

A efectos de detectar la presencia de objetos en movimiento en una escena en que la cámara se mantiene fija, el método más directo se basa en la eliminación del fondo de la imagen por sustracción. Dado que la cámara no altera su posición en el tiempo, el fondo de la imagen es inmediato de obtener a partir de una única captura, que debe llevarse a cabo en ausencia de móviles. A partir de este momento, cualquier imagen es fácilmente segmentable en objetos fijos y en movimiento mediante diferencia absoluta binarizada de los niveles de gris de cada píxel en ambas imágenes. En las zonas en que no aparece cambio alguno, dicha diferencia es igual a cero, ya que los niveles de intensidad se mantienen iguales. Por el contrario, si un móvil aparece en la escena, su nivel de brillo se superpone al fondo y esta diferencia deja de ser cero, salvo cuando el nivel de gris del móvil coincide con el del fondo. Así, las áreas de interés de la escena son aquellas regiones cuya diferencia entre el fondo y la imagen en curso no es nula.

El principal problema de este método es que requiere la intervención directa de un operador para determinar el momento en que se puede realizar la captura de lo que, en adelante, se va a considerar el fondo de la imagen. Además, pese a que la cámara sea estática, el fondo puede presentar alteraciones debidas a cambios en la iluminación y a la presencia de sombras locales originadas, por ejemplo, por la aparición de nubes. Para evitar esta dependencia, la metodología clásica [Brofferio et al. 1990] [Friedman y Russell 1997] estima el fondo a partir de la media de un número determinado de imágenes, que constituyen lo que se conoce como ventana temporal, utilizando la siguiente expresión:

$$B(x,y,t) = \frac{1}{N} \sum_{t'=t-N}^{t} I(x,y,t') \tag{3.2}$$

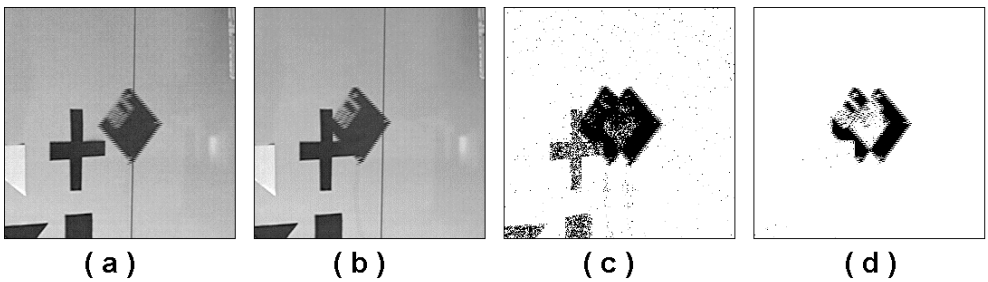


Figura 3.1: Problemas del método de detección de movimiento usando diferencia entre fotogramas: a) imagen $t-1$; b) imagen t; c) máscara binaria de movimento (U_{dif}=20); y d) máscara binaria de movimiento (U_{dif}=80).

siendo $B(x,y,t)$ el fondo estimado para la posición (x,y) en el instante t, $I(x,y,t')$ la imagen capturada en el instante t', y N el número de imágenes que componen la ventana temporal. De esta forma, el cálculo del fondo presenta dos ventajas: el ruido se suaviza al realizarse una media entre varias imágenes y los potenciales móviles de la imagen pesan menos a la hora de calcular el fondo que los objetos estáticos que puedan encontrarse en la escena. Obviamente, si un móvil fuese lo suficientemente lento como para permanecer en la misma posición durante la ventana temporal completa, quedaría automáticamente incluido en el fondo y podría inducir a posteriores errores de detección de movimiento (Fig. 3.2.a).

El problema mencionado ha sido resuelto en algunos casos adaptando N de forma dinámica a la velocidad de los móviles presentes en la escena utilizando, por ejemplo, filtros de Kalman modificados [Ridder et al. 1995]. Sin embargo, este cálculo no es sencillo y desvirtúa la baja complejidad computacional inicial del algoritmo. Debido a ello, la mayoría de los sistemas apuestan por valores de N constantes pero lo suficientemente elevados como para cubrir cualquier contingencia [Hötter et al. 1996]. Desafortunadamente, esta solución conlleva la necesidad de almacenar un número de imágenes que depende de la duración de la ventana temporal.

Una posible alternativa a aumentar la ventana temporal consiste en realizar un enmascarado dinámico sobre aquellas porciones de la imagen en las que se detecta movimiento [Huwer y Niemann 2000]. Este método consiste en etiquetar las zonas en movimiento de cada imagen para que dichas regiones no se incluyan en el proceso de promediado y, por tanto, no afecten al cálculo del fondo. Si bien de esta forma se evita que los potenciales móviles de una secuencia pasen a formar parte del fondo, aunque dichos móviles permanezcan en la misma posición durante la ventana temporal completa, el fondo no se actualiza en las áreas ocultas por los móviles (Fig. 3.2.b), lo cual puede desvirtuar el fondo estimado.

La solución a este problema la aporta la técnica de olvido exponencial [Kaup y Aach 1994], que elimina la necesidad de almacenar las imágenes de la ventana temporal calculando el fondo a sustraer mediante la siguiente expresión:

$$B(x,y,t) = (1-\alpha)\cdot B(x,y,t-1) + \alpha \cdot I(x,y,t) \tag{3.3}$$

siendo $B(x,y,t)$ el fondo estimado en el instante t, $B(x,y,t-1)$ el fondo previamente estimado, $I(x,y,t)$ la última imagen capturada y α un parámetro estimado empíricamente que determina la velocidad del proceso de olvido. En este proceso no es necesario almacenar más imágenes que el fondo y la última escena capturada y su combinación ponderada permite obtener el nuevo fondo en función de la última imagen adquirida. Si bien las necesidades de almacenamiento se minimizan, el sistema sigue adoleciendo del problema de distorsión clásico: independientemente

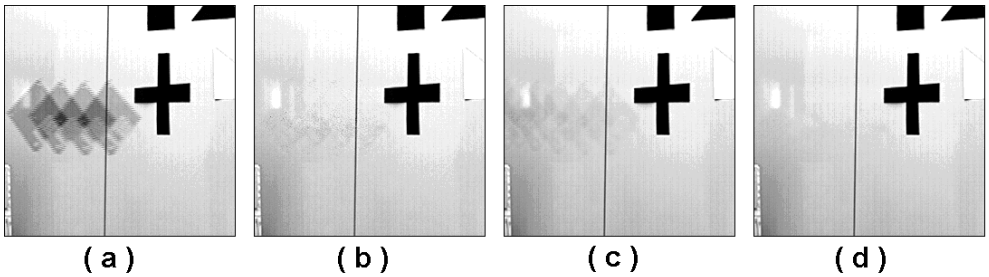


Figura 3.2: Estimación del fondo usando: a) enventanado temporal; b) enventanado temporal con enmascaramiento dinámico; c) olvido exponencial; y d) olvido exponencial con enmascaramiento dinámico.

de las oclusiones, cualquier móvil lento es susceptible de distorsionar el fondo estimado (Fig. 3.2.c). Ahora bien, si se aplica el concepto de enmascaramiento dinámico a este método, pueden combinarse sus ventajas para obtener un algoritmo capaz de adaptarse a los cambios de iluminación de la escena y al tránsito de objetos lentos frente a la cámara [Rodríguez et al. 2000a]. En este caso, ya no puede hablarse de áreas afectadas por oclusiones en tanto que ya no se define una ventana temporal que marque la duración de la secuencia de estimación. Así, la memoria de fondo se prolonga más allá de la aparición de cualquier objeto lento en la escena (Fig. 3.2.d).

En cualquier caso, y debido al proceso de enmascaramiento, los objetos móviles no se incluyen en el cálculo del fondo, que ahora obedece a la siguiente ecuación:

$$B(x,y,t) = \overline{M(x,y,t)}[(1-\alpha)\cdot B(x,y,t-1) + \alpha \cdot I(x,y,t)] + M(x,y,t)B(x,y,t-1) \tag{3.4}$$

siendo $M(x,y,t)$ la máscara de movimiento, una matriz binaria que presenta ceros en todas aquellas zonas donde no se ha detectado movimiento, es decir:

$$M(x,y,t) = \begin{cases} 1 & \text{si } |B(x,y,t-a) - I(x,y,t)| > U, \\ 0 & \text{en otro caso.} \end{cases} \tag{3.5}$$

donde $B(x,y,t-1)$ es la última estimación de fondo disponible, $I(x,y,t)$ la imagen actual y U un umbral heurístico que fija la sensibilidad de la máscara al movimiento de la escena. Como se muestra en la Fig. 3.3, el valor de U influye directamente en la detección de movimiento de la imagen, ya que si es elevado, aquellos objetos cuyo nivel de gris sea similar al del fondo no serán detectados (Fig. 3.3.c), y si es pequeño, la máscara de movimiento resulta excesivamente ruidosa (Fig. 3.3.d). No obstante, dado que existe integración en el tiempo a lo largo de la secuencia, esta dependencia no induce errores a largo plazo, ya que se puede esperar que, tarde o temprano, el móvil atravesará un área del fondo de la que será fácilmente distinguible. Además, si la máscara binaria presenta mucho ruido por la elección de un valor de umbral demasiado pequeño, se puede

incluir un proceso de filtrado posterior que, aún implicando una carga computacional adicional, reduzca considerablemente este efecto.

Este método aporta como ventaja, frente al detector basado en la diferencia entre imágenes consecutivas, su consistencia, ya que si bien este último es más rápido y simple desde un punto de vista computacional, su funcionamiento es correcto únicamente en los casos más sencillos o cuando las condiciones de captura son muy estables. Esto es debido a que cualquier cambio en la captura se refleja como un área en movimiento que, en este caso, no puede distinguirse de un fondo que no se dispone. Evidentemente, el problema del método de substracción de fondo consiste en seleccionar correctamente esta primera imagen de referencia o fondo. Por ello conviene garantizar la bondad de dicha estimación, bien generándola bajo supervisión o bien mediante promediado inicial de un alto número de imágenes, independientemente de que tras la inicialización el sistema ya no necesite ventana temporal alguna, y se empleen métodos como el del olvido exponencial dinámico [Rodríguez et al. 2000a].

2.2 Métodos locales de detección de movimiento

Los métodos puntuales de detección de movimiento se basan únicamente en la intensidad de la imagen, sin usar ningún otro tipo de información adicional, espacial o temporal, acerca de la naturaleza de los móviles. Sin embargo, el objeto móvil suele ser una región compacta del espacio, por lo que se puede asumir que si un punto de la imagen es declarado como perteneciente a un móvil, los puntos que lo rodeen tendrán mayor probabilidad de pertenecer a ese mismo móvil. Además, como ya se entreveía en el método puntual de la diferencia entre fotogramas, la secuencia temporal proporciona una información básica sobre el movimiento, que puede ser empleada para compactar la detección [Toth et al. 2000a].

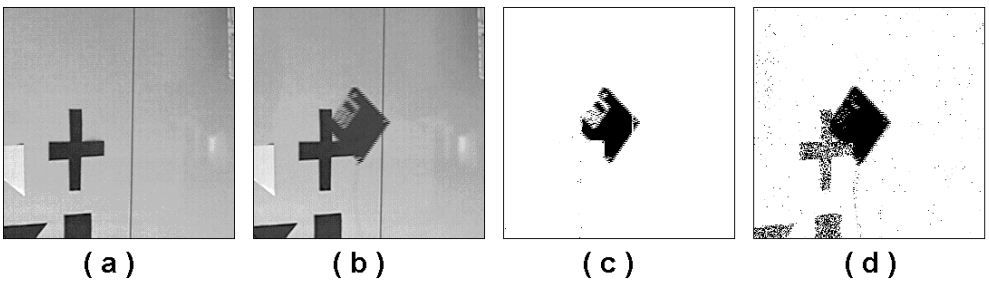


Figura 3.3: Dependencia respecto al parámetro U del proceso de detección de movimiento usando fondo: a) fondo; b) imagen; c) máscara (U=80); y d) máscara (U=20).

2.2.1 Detector de movimiento adaptativo en el espacio

Para tener en cuenta la información espacial y temporal previa, se puede combinar el detector puntual de movimiento con un modelo de campo aleatorio de Markov. Básicamente, se trata de un método local con el que se obtiene un valor de umbral distinto en cada punto de la imagen para el detector de movimiento, Ecs. (3.1) y (3.5), que será función de la probabilidad de que dicho punto pertenezca a un objeto móvil o al fondo. La idea es que si el punto en cuestión tiene mayores probabilidades de pertenecer al móvil, el umbral se reduce, con lo cual se refuerza la posibilidad de esta pertenencia, pero si el punto tiene más probabilidad de pertenecer al fondo, el valor de umbral se aumenta, reforzando así dicha pertenencia.

Para obtener este umbral adaptativo se puede usar el método propuesto por T. Aach y A. Kaup [Aach y Kaup 1995]. Este método consiste en examinar la vecindad 3x3 de cada punto de la máscara de movimiento detectada, contando el número de puntos móviles encontrados. En la práctica, este proceso se realiza en una única pasada, con lo cual la máscara de movimiento se va actualizando conforme se analiza cada punto. Dado que la imagen se recorre fila a fila, al estudiar un punto sólo se dispondrá de cuatro puntos actualizados (3 superiores y uno lateral), tomándose los otros cuatro de la máscara de movimiento anterior. El valor del umbral en cada punto será:

$$t(n_i) = T + (4 - n_i) \cdot B \tag{3.6}$$

donde n_i es el número de puntos móviles detectados en la vecindad del punto i, T el valor medio del umbral (obtenido para n_i igual a cuatro), y B una constante positiva que determina el rango de variación del umbral. En los resultados obtenidos usando esta simple medida de reforzamiento espacial (veáse la Fig. 3.4) queda patente la ganancia en compacidad de las regiones detectadas [Aach y Kaup 1995] [Konrad 2000] [Toth et al. 2000a].

La principal desventaja del detector adaptativo es, precisamente, el tener que calcular un valor de umbral distinto para cada punto de la imagen, ya que, aunque esta operación no es compleja, puede suponer un tiempo de cómputo importante dentro del total consumido por el detector de movimiento. De cualquier forma, dicho problema sería subsanable siempre que los parámetros B y T fueran constantes, empleando para ello una tabla donde a cada valor de n_i se le asignara un valor de umbral distinto, $t(n_i)$. De esta forma, la operación de cálculo de umbral se transformaría en una simple asignación, evitando la necesidad de ejecutar las operaciones de la Ec. (3.6).

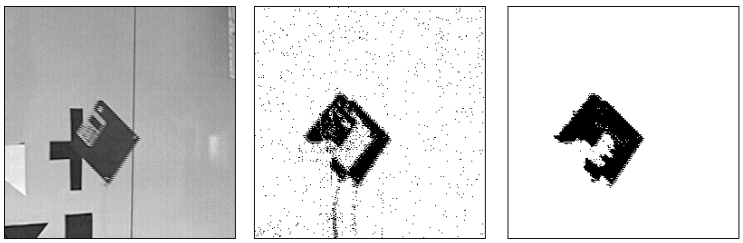


Figura 3.4: Empleo de umbral adaptativo en el espacio: a) fotograma en el instante t; b) máscara de movimiento por diferencia entre fotogramas con umbral fijo (U=15); y c) máscara de movimiento por diferencia entre fotogramas con umbral adaptativo (T=15; B=3).

2.2.2 Detector de movimiento adaptativo espacio-temporal

Para incrementar la inmunidad al ruido del método de detección, el detector puntual puede ser extendido temporalmente, empleando una técnica de enventanado como la descrita para la estimación del fondo de referencia en el subapartado 2.1. En este caso, usando la variable tiempo como tercera dimensión, se pueden definir vecindades 3D, que usan la vecindad-8 bidimensional sobre cada uno de los tres fotogramas tomados en tres instantes distintos de tiempo ($t-1$, t y $t+1$), siendo el punto en estudio el centro de la vecindad (ver Fig. 3.5) [Toth et al. 2000b].

Sin embargo, esta vecindad 3D plantea problemas, pues emplea fotogramas posteriores al que se está analizando en el instante t. Una solución alternativa válida [Geman y Geman 1984] consiste en usar sólo el valor de la máscara de movimiento en el instante $t-1$, de forma que el valor del umbral adaptativo adopta la siguiente expresión:

$$t(n_i) = T + (4 - n_i) \cdot B + C \cdot (\frac{1}{2} - n_c) \tag{3.7}$$

donde n_i es el número de puntos móviles detectados en la vecindad-8 del punto i, n_c es el valor del punto i de la máscara de movimiento en el instante $t-1$, T el valor medio del umbral (obtenido para n_i igual a cuatro y n_c igual a 0.5), y B y C dos constantes positivas que determinan el rango

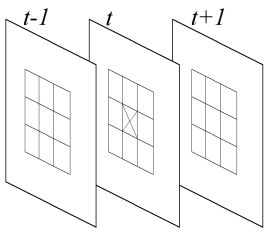


Figura 3.5: Vecindad 3D evaluada por el detector de movimiento adaptativo espacio-temporal.

de variación del umbral $t(n_i)$. Sin embargo, aunque esta mejora puede aumentar la inmunidad al ruido del detector, las ligeras ventajas que reporta no justifica el incremento de complejidad que supone su empleo.

Por último, comentar que existen otros métodos locales espacio-temporales de detección basados igualmente en estimadores estadísticos. Así, la diferencia entre fotogramas ha sido modelada como una mezcla de distribuciones gaussianas y laplacianas [Tziritas y Labit 1994], o se han usado filtros de Kalman en determinados fotogramas de referencia para tratar de adaptarse a los cambios de la secuencia [Karmann et al. 1990]. El modelo de campos aleatorios de Markov puede ser combinado con otro criterio bayesiano, el de *maximum a posteriori* (MAP) [Bouthemy y Lalande 1993]. En este caso, el modelo se complica incorporando estudios de continuidad en el tiempo y en el espacio.

El principal problema de la mayoría de los métodos locales es su tiempo de procesamiento, especialmente elevado en aquellos algoritmos estadísticos que usan funciones de coste globales [Aach y Kaup 1995] [Bischel 1994]. Para tratar de reducir este tiempo, se han empleado aproximaciones jerárquicas [Odobez y Bouthemy 1995], que no siempre resuelven correctamente el problema de la detección de móviles, como se demuestra en [Luthon et al. 1999].

2.3 Método propuesto de detección de movimiento

El método propuesto consiste en una solución híbrida, pues emplea inicialmente un detector puntual con umbral fijo y un posterior filtrado espacial que refuerza la compacidad de los objetos móviles presentes y obtiene un fondo uniforme libre de ruido [Rodríguez et al. 2000c].

El detector puntual empleado devuelve, como resultado del proceso de substracción, una imagen cuyos píxeles presentan un valor no nulo únicamente en las áreas en que se han detectado cambios frente al fondo. Obviamente, esta imagen no devuelve exactamente los móviles presentes en la escena, ya que muchos factores son origen de falsas detecciones, como sombras, luces no ambientales, apertura y cierre de puertas en interiores, o móviles cuyo color coincide total o parcialmente con el del fondo en una localización determinada. Así pues, la imagen en cuestión presentará una serie de áreas más o menos conexas, con diversos tamaños y formas distribuidos a lo largo de la imagen que, según el contexto de la escena, habrá que desestimar como potenciales objetos (Fig. 3.6.c).

A efectos de detectar las regiones válidas entre todas las áreas marcadas, se lleva a cabo un filtrado que básicamente consiste en cohesionar los móviles detectados mediante un proceso

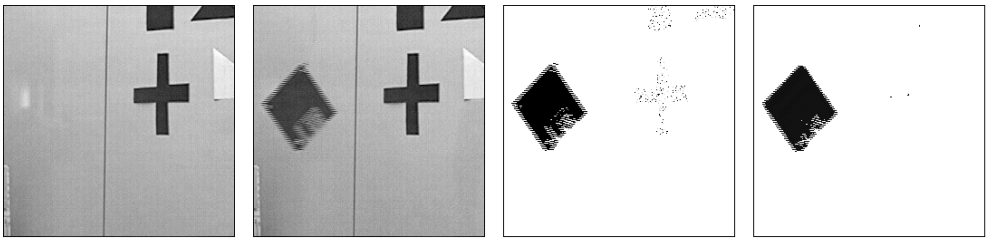


Figura 3.6: Segmentación por movimiento usando detección por fondo: a) fondo; b) imagen; c) máscara de movimiento; y d) máscara segmentada.

de etiquetado. Dicho proceso agrupa los potenciales objetos y elimina, al menos parcialmente, el ruido propio de una imagen de este tipo. Los pasos del algoritmo se detallan a continuación:

- Sobre la imagen que contiene la máscara de movimiento resultante del proceso de umbralización, se calcula, para cada píxel, el valor de la suma de todos sus vecinos en una vecindad 3x3. Si dicha suma es mayor que un umbral heurístico que fija el nivel de eliminación de ruido de la imagen, dicho píxel pertenece a un objeto. Puede notarse que este simple proceso permite eliminar puntos aislados de la imagen, que constituyen la mayor parte del ruido originado en el proceso de substracción.

- En el caso de que el píxel en estudio pertenezca a un objeto, para estudiar a cual pertenece se evalúan sus vecinos en la mencionada vecindad 3x3. Si alguno de dichos píxeles ha sido previamente etiquetado, el píxel bajo estudio recibe la misma etiqueta de éste. En el caso de que existiesen varias etiquetas en la vecindad, el píxel recibe una cualquiera de ellas y se anota la equivalencia de todas las etiquetas implicadas para posteriormente realizar la fusión de las distintas partes que componen cada objeto. Si no existiese ninguna etiqueta disponible en la vecindad, se crea una nueva.

Nótese que este proceso no es más que un barrido por las regiones compactas de la imagen en el que a cada objeto aislado se le asigna una etiqueta. Si un nuevo punto está en contacto con dos objetos etiquetados de forma distinta, automáticamente son detectados como uno solo, fundiéndose en una única entidad. De esta forma están contemplados los casos relativos a la presencia de objetos cóncavos.

La Fig. 3.6 presenta el proceso expuesto para la imagen de la Fig. 3.6.b, una vez se le extrae el fondo de la Fig. 3.6.a. Tal como puede observarse en la Fig. 3.6.c, la imagen resultado es muy ruidosa, pero el proceso de segmentación y etiquetado permite eliminar parcialmente dicho ruido y detectar los distintos objetos presentes en la imagen.

El método propuesto presenta dos importantes ventajas sobre los detectores locales analizados en el apartado 2.2. Por un lado, el umbral de detección es fijo, con lo cual se eliminan tanto los problemas de consumo de tiempo que implican su cálculo por cada punto de la imagen, como el ajuste de un conjunto de parámetros que, en ocasiones, puede ser bastante elevado. Por otro lado, el algoritmo de filtrado proporciona en este caso una máscara de movimiento segmentada. De esta forma, el paso de segmentación, previo a la estimación de movimiento, se realiza conjuntamente al filtrado de la imagen, ahorrando un tiempo considerable al sistema.

3 Estimación de movimiento

La información proporcionada por el movimiento resulta esencial para dos tipos diferentes de aplicaciones [Konrad 2000]: la compresión y el procesado de secuencias de vídeo. Esta diferenciación resulta adecuada, pues permite distinguir entre la estimación de movimiento orientada a comprimir, que debe procurar el máximo factor de compresión posible para un nivel de calidad dado, y la estimación de movimiento dedicada a servir como etapa de preproceso, que debe buscar el valor verdadero de desplazamiento de cada punto, para así poder, por ejemplo, generar nuevos fotogramas localizados entre dos fotogramas consecutivos de una secuencia (como hacen los algoritmos de conversión de NTSC a PAL).

Los métodos de estimación de movimiento se suelen clasificar en función del tipo de modelo que asignan a los objetos o regiones. De esta forma, existen métodos que modelan el objeto en tres dimensiones, asumiendo, generalmente, que el objeto es rígido [Iiyama et al. 2000] para así simplificar el cálculo. Sin embargo, la gran mayoría de los métodos usan modelos 2D, de regiones rígidas o deformables.

A continuación se describen brevemente tres métodos de estimación de movimiento con modelado 2D que resultan bastante significativos dentro del conjunto de algoritmos empleados en la actualidad, exponiendo posteriormente el método propuesto en este trabajo. Evidentemente, no están reflejadas todas las posibilidades, pues la dependencia respecto a la aplicación final hace que no resulte fácil clasificar los distintos algoritmos de estimación. Además, no se han incluido entre los métodos analizados aquellos que no trabajan con regiones dentro de la imagen, sino con la imagen global (como el método GMC que incluye la segunda versión del estándar de compresión de vídeo MPEG-4 [ISO/IEC 1996] [ISO/IEC 1998]), o que trabajan puntualmente (algoritmos de cálculo de flujo óptico [Anandan 1987] [Horn y Schunk 1981] [Lucas y Kanade 1981]).

3.1 Correspondencia de regiones

El método de correspondencia entre regiones (*block matching*) es, posiblemente, el algoritmo más intuitivo y simple de estimación de movimiento local. Básicamente, el método divide la imagen en regiones -generalmente rectangulares- y supone que todo móvil sigue, en la imagen 2D, un movimiento de traslación constante y temporalmente lineal.

Así, usando la notación de la Fig. 3.7, se cumplirá que:

$$x(m,n,t) \approx x(m-d_h, n-d_v, t-1) \quad \forall (m,n) \in S \tag{3.8}$$

El algoritmo trata, por tanto, de encontrar la correspondencia entre las regiones en los instantes $t-1$ y t, acotando el campo de búsqueda en función de un determinado radio de apertura. El método así definido resulta extremadamente costoso, pues supone correlar una región en la imagen $t-1$ con todo un conjunto de regiones en la imagen t. Para reducir el tiempo empleado en este proceso se han propuesto distintas alternativas, como la búsqueda *one-at-a-time* [Srinivasan y Rao 1985], el uso de estructuras jerárquicas [Bhaskaran y Konstantinides 1997], o la búsqueda en N pasos [Konrad 2000].

Como principales limitaciones del método cabe destacar que: i) la velocidad de los móviles está acotada en función del radio de apertura de búsqueda; ii) la descomposición de la imagen en bloques da lugar a la aparición de errores de estimación cuando un bloque contiene partes estáticas y móviles simultáneamente.

Por último, destacar que la regularidad y simplicidad de este método ha permitido su implementación en VLSI [He et al. 1995] [Baek et al. 1996], con lo cual hoy en día es prácticamente

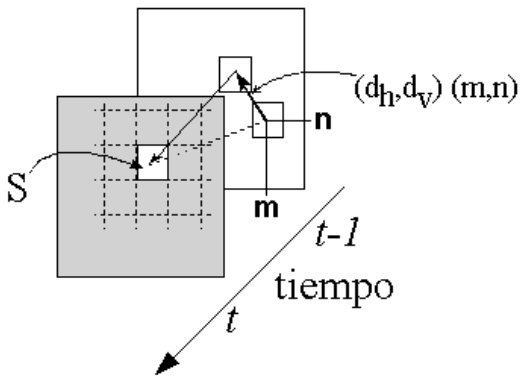


Figura 3.7: Esquema del proceso de correspondencia entre regiones.

el único estimador de movimiento usado en todos los estándares de compresión de vídeo. Un ejemplo de ello es la existencia de circuitos integrados comerciales que lo implementan, como el caso del `STi3220` del fabricante *SGS-THOMSON Microelectronics*, que realiza un $block - matching$ de búsqueda completa.

3.2 Correlación de fase

El método de correlación de fase combina la técnica de la correspondencia entre regiones, que resulta ser excesivamente costosa, con un análisis en el dominio de la frecuencia, que no puede abordar el movimiento a nivel local [Gonzalez y Wintz 1987]. El método hace uso de las ventajas de ambas aproximaciones, desarrollando un método de *block matching* que trabaja en dos pasos: en primer lugar, se seleccionan los posibles candidatos de cada región mediante un estudio global en el dominio de la frecuencia, y, posteriormente, dicha búsqueda se selecciona mediante una correlación local de regiones. En la práctica, los pasos del algoritmo son [Tekalp 1995]:

- Se dividen las imágenes I_{t-1} e I_t en grandes regiones cuadradas (p. ej., de 64x64 píxeles), calculándose la transformada rápida de Fourier (FFT) de cada región.
- Se calcula la correlación normalizada entre las dos imágenes, $\Psi_{t-1,t}$.
- Se obtiene la transformada inversa de $\Psi_{t-1,t}$, y se identifican los picos de mayor peso.
- Las coordenadas de estos picos se emplean como candidatos para realizar un *block matching* de regiones de 16x16 píxeles.

Por lo tanto, el método de la correlación de fase es básicamente un algoritmo de maximización de un criterio de acierto basado en una correlación de regiones. El método es rápido y eficiente, pero, al emplear como algoritmo de búsqueda local el de *block matching*, adolece de sus mismos errores en el modelado en cuadrícula de los objetos, y difícilmente puede tratar con rotaciones y operaciones de escalado, fruto de acercamientos o alejamientos.

3.3 Predicción usando filtros de Kalman

Los estimadores de movimiento basados en *block matching* o correlación de fase son métodos que tratan de encontrar el desplazamiento neto de todas las regiones en que dividen la imagen. Son, por tanto, métodos válidos para aplicaciones de compresión o tratamiento global de secuencias de vídeo. Existe otro tipo de situaciones en las que se trata, más que de estimar el

movimiento de toda la escena, de seguir el movimiento de un determinado móvil o conjunto de éstos [Koller et al. 1994] [Takeuchi et al. 1995]. Las aplicaciones de un estimador de movimiento de este tipo son variadas, pero principalmente se centran en supervisión y control de tráfico [Kilger 1992], y en vigilancia [Cohen y Medioni 1998].

El objetivo del estimador en este tipo de aplicaciones, una vez ha sido detectada una potencial zona de interés en una imagen determinada, consiste en localizar esta zona en el resto de las imágenes de la secuencia en tanto no abandone el campo de visión, ya que presumiblemente dicha área seguirá siendo de interés a lo largo de la mencionada secuencia. Asimismo, puede afirmarse que cuando un área de interés se detecta en varias imágenes, queda probada su validez, mientras que en caso contrario puede descartarse esta zona como producto de ruido o de cambios en las condiciones de captura. Debido a los motivos expuestos, es inmediato constatar la necesidad de un proceso de seguimiento que permita establecer una relación entre las áreas de interés detectadas en las distintas imágenes de una secuencia y que permita básicamente mantener la integridad de la información en el tiempo.

Este proceso de seguimiento, que a priori puede parecer muy simple, está lejos de ser una tarea sencilla, principalmente debido a que un móvil puede perderse en una secuencia real por multitud de circunstancias. Por ejemplo, puede darse el caso mostrado en la Fig. 3.8, donde aparecen dos móviles similares dirigiéndose al centro del campo de visión en sentidos contrarios. Al ser detectados por primera vez, un sistema automático los podría etiquetar como A y B. Posteriormente, una vez que dichos objetos se han cruzado, aparecen dos objetos, que el sistema etiquetaría como C y D, otros dos móviles detectados. Lógicamente, dichos móviles deben ser A y B, pero es necesario determinar cuál es cuál. Aunque pueda parecer que el problema se soluciona si no coexisten dos móviles en una misma imagen, puede pensarse en un caso en que un móvil avance hasta una posición determinada del campo de visión y se detenga; si entra en el campo de visión un nuevo móvil, el sistema podría confundirlo con el primero, debido a que éste aún no habría abandonado la escena.

Figura 3.8: a-c) Fotogramas de una secuencia de cruce de dos objetos móviles.

La solución a este tipo de problemas reside en confeccionar modelos que se ajusten a la cinética de los objetos en movimiento en la escena, para poder determinar dónde puede esperarse que aparezcan en imágenes posteriores a partir de su posición en un momento dado. Así, una vez se dispone de un mínimo de dos imágenes, el estudio de la posición de un objeto determinado en ambas permite estimar su velocidad, lo que permitiría a su vez calcular su posición aproximada en la imagen siguiente, en tanto se mantengan la velocidad de captura y la velocidad del móvil. Este proceso de predicción permite llevar a cabo el proceso de seguimiento mediante la búsqueda de un objeto determinado en las proximidades del lugar que se espera que ocupe en sucesivos fotogramas.

Las técnicas de predicción pueden obedecer a modelos muy variados, que abarcan desde sistemas lineales sumamente simples hasta ecuaciones matemáticas extremadamente complejas. El por qué de esta complejidad se debe a que difícilmente un móvil seguirá una trayectoria de velocidad y dirección constante (Fig. 3.9). Así, en determinados escenarios, es necesario introducir modificaciones a los predictores más simples, como es el caso de los filtros de Kalman, que suelen consistir en bucles de optimización que tienen en cuenta posibles no linealidades [Hanek y Schmitt 2000].

A efectos de mantener la carga computacional del proceso de seguimiento al mínimo y en tanto la velocidad de captura sea lo suficientemente elevada para que no se detecten cambios severos en la cinética de los objetos móviles de una escena entre una imagen y la inmediatamente siguiente, es posible usar en la mayoría de los casos modelos de predicción lineal sumamente simples que dan una idea aproximada de la posición del objeto a lo largo de una secuencia [Faugeras 1993]. Si bien la precisión de los citados modelos es insuficiente para seguir objetos

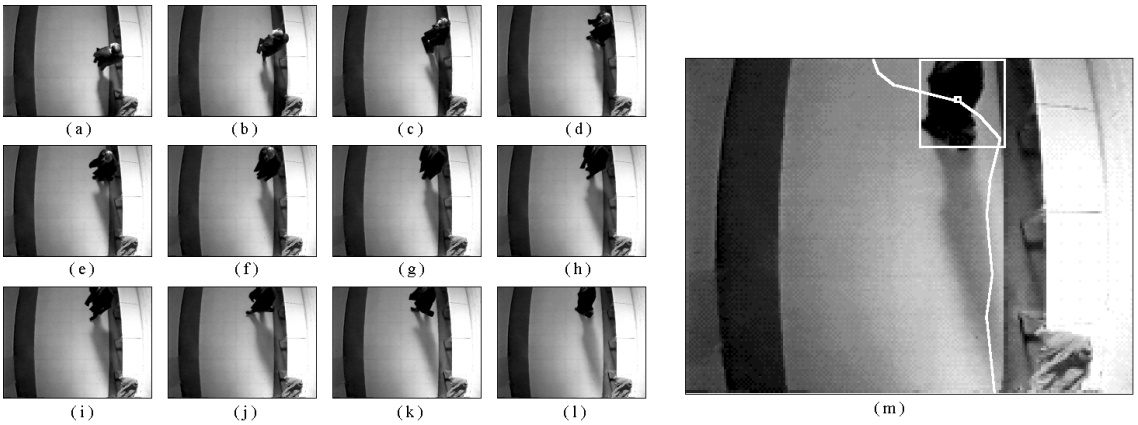


Figura 3.9: a-l) Fotogramas que muestran el desplazamiento de un determinado móvil; y m) trayectoria seguida por el móvil.

muy pequeños o muy rápidos, es posible predecir la posición del centroide de un objeto determinado, y después llevar a cabo una búsqueda rápida de centroides en la vecindad del punto predicho para corregir potenciales errores.

3.4 Método propuesto de estimación de movimiento

El modelo que se va a emplear tiende a aumentar la compacidad de las regiones de interés de una imagen, descarta errores debidos a ruido o cambios de iluminación, como sombras y otros, permitiendo mantener los recursos del sistema concentrados en dichas regiones de interés. El sistema propuesto [Rodríguez et al. 2000c] se basa en los siguientes supuestos:

- Se define como área de interés cualquier zona compacta que se distinga del fondo de la imagen y pueda ser detectada a lo largo de varias imágenes consecutivas de la secuencia. Para predecir la posición de dicha área en cada imagen es necesario dotar al sistema de memoria e inercia, lo que se consigue mediante filtros de Kalman de velocidad constante [Mohinder et al. 1993].

- Suponiendo que de una primera detección se extrae un conjunto de áreas de interés, se predice la posición de los centroides de dichas áreas en el fotograma actual y se procede a la búsqueda de cada área dentro de la $bounding-box$ que la englobaría en la siguiente imagen, si efectivamente se hubiese desplazado de acuerdo al modelo de Kalman. Posteriormente, una vez encontrada la correcta posición del móvil, se corrige la predicción previa del modelo de Kalman, atendiendo a su naturaleza predictora-correctora.

- Inicialmente, se considera que toda zona de interés encerrada en dicha $bounding - box$ pertenece al área de interés buscada, lo que significa que se admite la conservación de masa entre dos fotogramas consecutivos. Este principio es necesario porque al desplazarse, las áreas de interés tienden a crecer apropiándose de otras áreas pertenecientes al fondo cuyo nivel de brillo es similar a éstas en un momento dado. Manteniendo un tamaño acotado, hemos comprobado empíricamente que dicho efecto es parcialmente eliminado. No obstante, se permiten cambios dinámicos en la forma de los objetos entre dos fotogramas consecutivos siempre y cuando éstos no sean excesivamente significativos. Así, como se muestra en la Fig. 3.10, se admite la existencia de objetos deformables en el campo de visión, y su posición es correctamente detectada durante todo su paso a través de la escena, pero no se puede realizar el correcto seguimiento de un objeto si éste sufre una generación o pérdida espontánea y excesiva de masa. Esto ocurre en ocasiones como en las que un

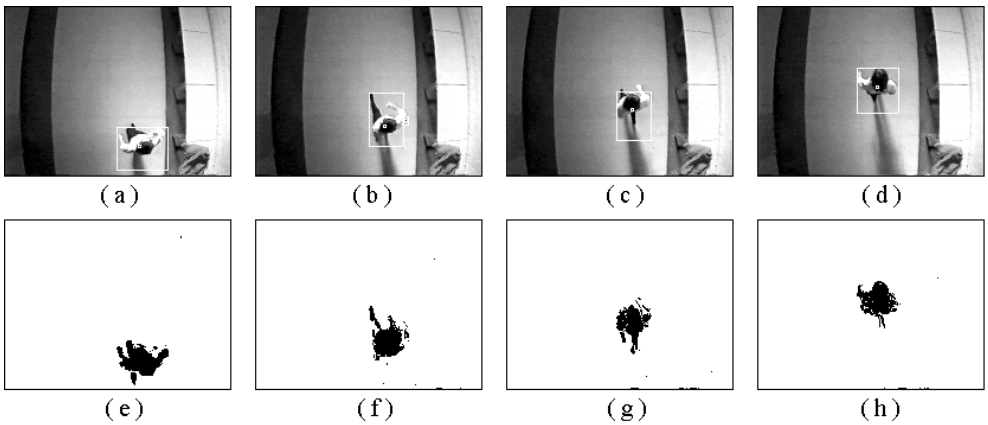


Figura 3.10: a-c) Fotogramas que muestran la capacidad de deformación permitida a los objetos móviles (Detección realizada mediante la técnica de fondo con olvido exponencial dinámico, U=30).

individuo se desplaza hasta una posición cualquiera del campo de visión, toma un objeto voluminoso perteneciente al fondo y lo mueve consigo durante el resto de su trayectoria.

El sistema desarrollado, por tanto, incluye un detector y un estimador-predictor de movimiento, de forma que realiza el seguimiento de cualquier objeto móvil que aparece en el campo de visión. El detector de movimiento implementado usa un fondo de referencia, que se refresca utilizando la técnica de olvido exponencial dinámico, y que se complementa con el filtrado de refuerzo de objetos que se proponía en el subapartado 2.3. Como se comentó en dicho epígrafe, el algoritmo de filtrado proporciona una imagen segmentada del entorno, es decir, no sólo se dispone de una división en objetos móviles y fondo, sino que estos objetos son, además, correctamente etiquetados. El estimador empleado memoriza, entre fotogramas consecutivos, la posición de cada objeto, usando esta información para predecir la posición que posiblemente ocupará en el fotograma siguiente. Además, el proceso de estimación emplea una simple regla de conservación de masa para poder discernir cuándo se producen situaciones confusas como cruces o fusiones de objetos distintos.

4 Resultados

En la Fig. 3.11 se muestra la comparación entre los resultados de seguimiento de objetos obtenidos usando un fondo con olvido exponencial dinámico (Fig. 3.11.a) o la técnica de la diferencia entre fotogramas consecutivos (Fig. 3.11.d). Como se puede observar, el empleo de un detector u otro resulta sumamente importante en este caso, estando totalmente distorsionados los resultados cuando se emplea el método de la diferencia entre fotogramas (Figs. 3.11.d-f).

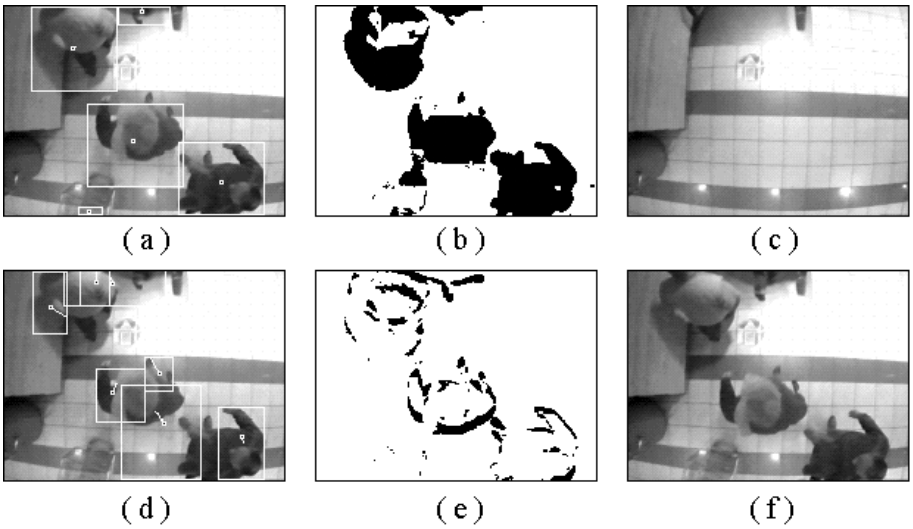


Figura 3.11: a) Detección de objetos usando la técnica de fondo con olvido exponencial dinámico (U=30), b) máscara de movimiento usada en a), c) fondo empleado para obtener b), d) detección de objetos usando diferencia entre fotogramas consecutivos, e) máscara de movimiento usada en d), y f) fondo -fotograma anterior- empleado para obtener e).

Esto se debe a que la máscara binaria resultante del proceso de umbralización de la diferencia en valor absoluto de fotogramas consecutivos es mucho menos compacta que su homónima para el caso de emplear una estimación de fondo. En este caso el efecto es especialmente notorio debido a la velocidad que presentan los móviles.

Evidentemente, el empleo de un fondo de referencia correcto con el cual comparar cada fotograma permite obtener unos resultados mucho más robustos, lo que se puede apreciar en las máscaras de movimiento extraídas usando uno u otro método. Además, permite tratar objetos móviles que, eventualmente, entren en el campo de visión, se detengan y luego prosigan su movimiento, como se muestra en la Fig. 3.12.

Sin embargo, la detección de movimiento usando el método de diferencia entre fotogramas presenta una importante ventaja frente al uso de un fondo de referencia, que es su simplicidad de cálculo. Esta simplicidad se traduce en un tiempo menor de proceso y, por ello, es capaz de procesar un número mayor de imágenes por segundo. La Tabla 3.1 muestra los tiempos máximos, medios y mínimos de procesado por fotograma empleados por un Pentium MMX 233Mhz sobre una secuencia real de 4395 fotogramas (192x144 píxeles). El proceso global se ha dividido en tres fases, cuyos tiempos de proceso se han estimado independientemente. El tiempo de "Detección" será el empleado en llevar a cabo este proceso, sin el posterior filtrado. Bajo el título "Proc. Fondo" se encuentra el tiempo destinado a la actualización del fondo, que en el caso de la detección usando diferencia de fotogramas se destina a almacenar el presente

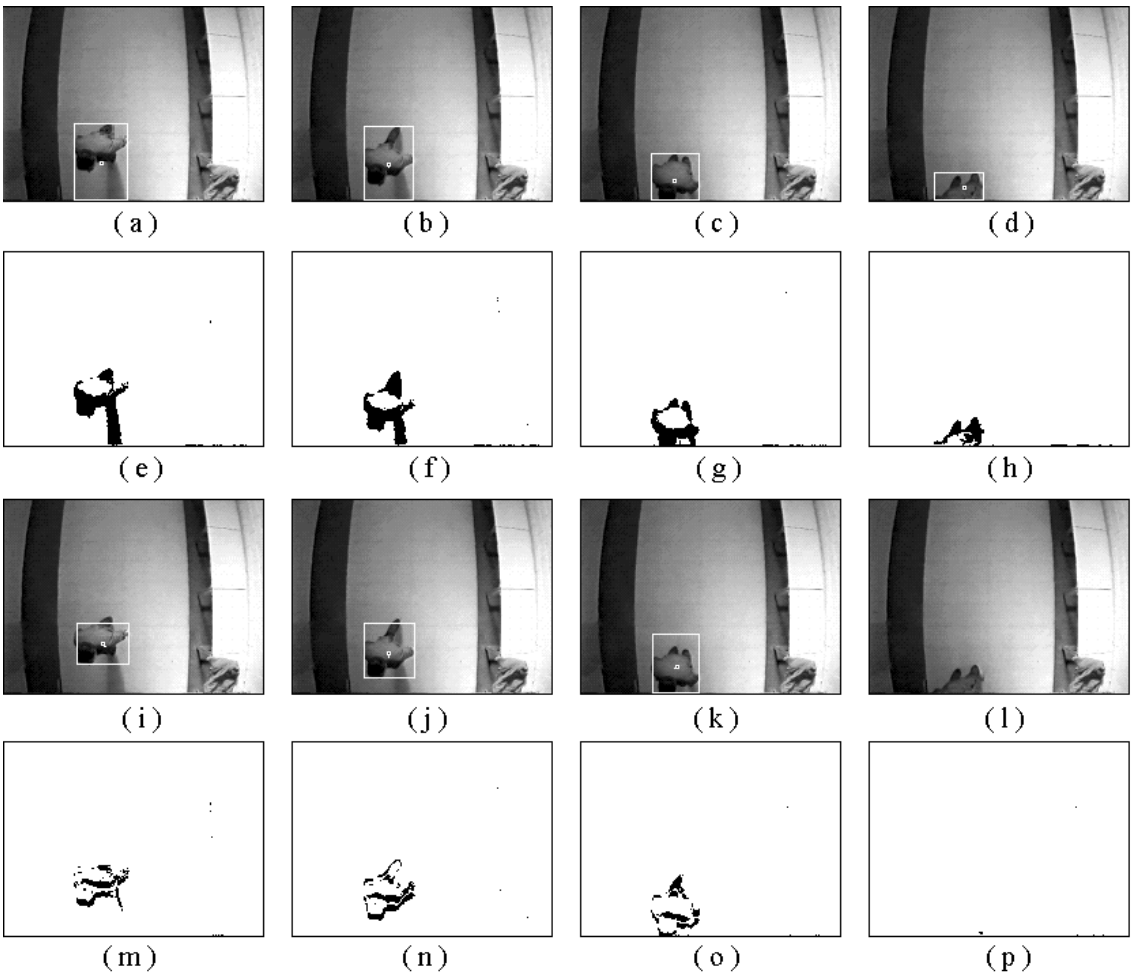


Figura 3.12: a-d) Detección de objetos usando la técnica de fondo con olvido exponencial dinámico (U=30), e-h) máscaras de movimiento usadas en a-d) respectivamente, i-l) detección de objetos usando diferencia entre fotogramas consecutivos, y m-p) máscaras de movimiento usadas en i-l) respectivamente.

fotograma en memoria. Finalmente, en "Segmentación-Estimación" se indica el tiempo que consumen los procesos de filtrado y seguimiento de objetos. Conviene destacar que el método que usa estimación de fondo emplea 37'2 ms. en todo el proceso, mientras que el de diferencia de fotogramas necesita solamente 20'4 ms. Estos tiempos se refieren siempre al peor caso de los resultados medidos, con lo cual el procesamiento completo está garantizado. Esto implica que, debido a que las secuencias usadas fueron capturadas a una velocidad de 8 fotogramas por segundo (uno cada 125 ms.), un procesador de las características mencionadas estaría en disposición de realizar la tarea de detección y seguimiento de móviles sobre un total de 3 cámaras simultáneamente con el primer método y hasta 6 con el segundo, respectivamente.

El paso de filtrado permite compactar tanto el fondo como los objetos detectados, pro-

Detección-Estimación usando fondo				
	Detección	Proc. Fondo	Segmentación-Estimación	Total
$t_{min}(ms)$	1,1	7,1	2,3	10,5
$t_{med}(ms)$	1,1	7,4	2,8	11,3
$t_{max}(ms)$	9,5	13,3	14,4	37,2
Detección-Estimación usando diferencia de fotogramas				
	Detección	Proc. Fondo	Segmentación-Estimación	Total
$t_{min}(ms)$	1,1	1,6	2,3	5,0
$t_{med}(ms)$	1,2	1,8	2,5	5,5
$t_{max}(ms)$	2,8	3,0	14,6	20,4

Tabla 3.1: Tiempos de procesamiento típicos de una secuencia de vídeo (192x144)

porcionando una máscara segmentada en la que se encuentran etiquetadas las distintas regiones. Debido al ruido que aún puede afectar a la máscara obtenida, las regiones que tengan un tamaño excesivamente pequeño son descartadas. Evidentemente, el valor del umbral o de la relación de tamaño que elimina estas regiones dependerá de la aplicación y requiere un proceso de ajuste específico supervisado.

Finalmente, el algoritmo de estimación se encarga de calcular el movimiento asociado a cada objeto, y, en este caso, de asociar cada móvil del fotograma actual con su igual en el fotograma posterior. Para poder llevar a cabo el seguimiento de cada objeto, se usa un filtro de Kalman, con una memoria de diez fotogramas. Usando este banco de datos para cada objeto, se puede establecer una correcta estimación de la trayectoria de cada móvil, resolviendo situaciones relativamente complejas, con cruces (Fig. 3.13) o fusiones parciales de móviles (Fig. 3.14).

Por último, cabe mencionar la existencia de situaciones que dan lugar a resultados anómalos. Puede ocurrir que las trayectorias de dos áreas de interés coincidan en un punto, de forma que éstas se fundan, total o parcialmente. Esta situación se muestra en la Fig. 3.15, donde, si bien los objetos móviles son inicialmente detectados de forma correcta (Fig. 3.15.a), existe un momento en que sus máscaras de movimiento se superponen y, debido a la cercanía de sus posiciones calculadas por el método predictor, el móvil marcado como C pierde parte de

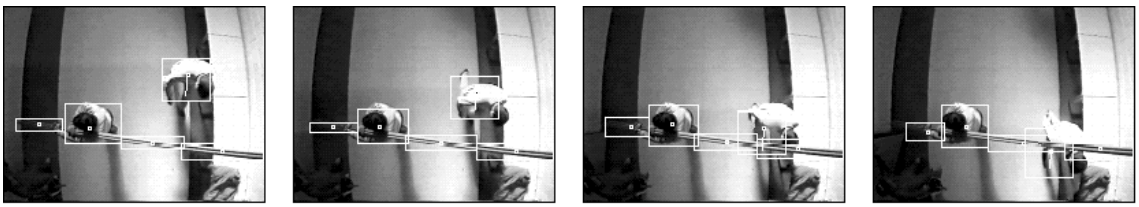


Figura 3.13: a-d) Detección de objetos móviles usando fondo con olvido exponencial dinámico.

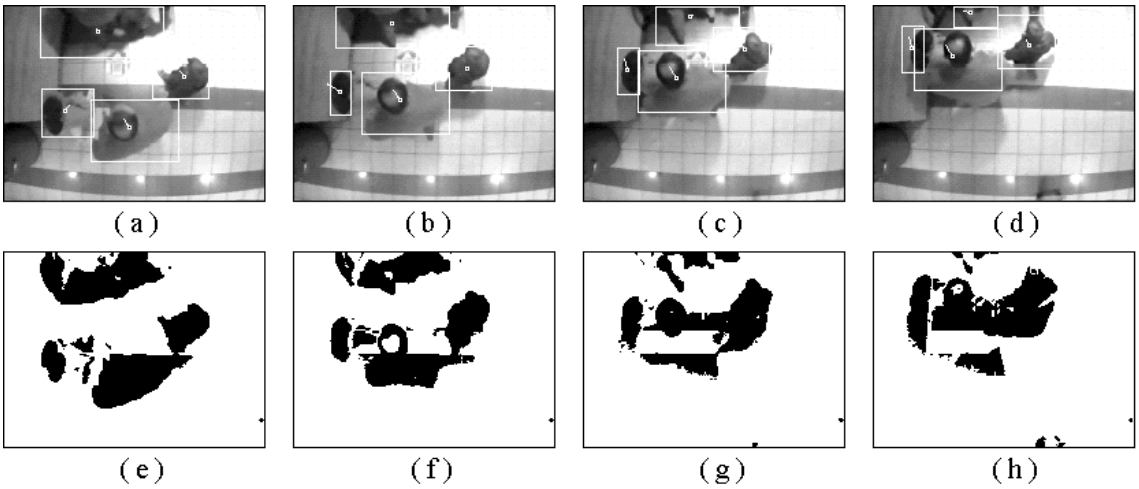


Figura 3.14: a-d) Detección de objetos usando la técnica de fondo con olvido exponencial dinámico (U=30) y e-h) máscaras de movimiento usadas en a-d) respectivamente.

su área, que es absorbida por el móvil A.

5 Conclusiones

En este apartado se han presentado brevemente algunos de los mecanismos más importantes tanto de detección como de estimación de movimiento. En este estudio, dichas técnicas se han empleado para llevar a cabo la segmentación de una secuencia de imágenes, de manera que la información espacial relativa al nivel de brillo estuviera reforzada por la de movimiento, extraída del análisis de la secuencia. Al considerar estas tareas como propias del procesado de

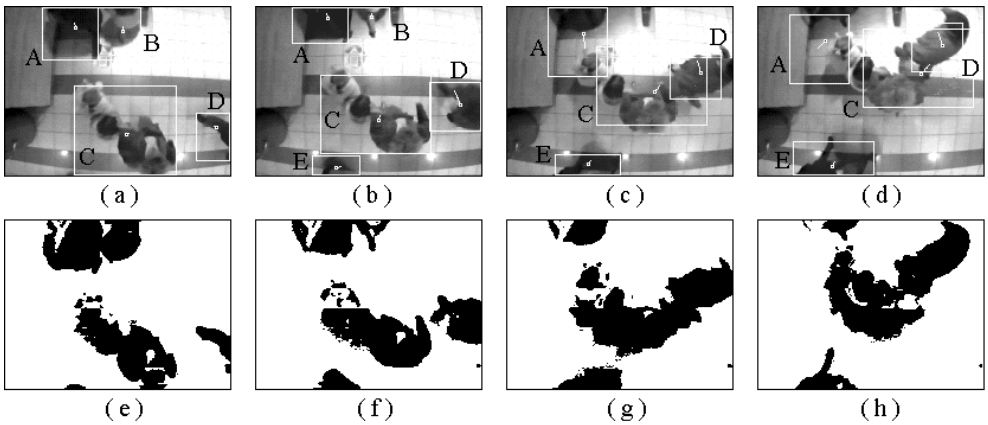


Figura 3.15: a-c) Fotogramas que muestran el mezclado de porciones pertenecientes a dos móviles distintos (Detección realizada mediante la técnica de fondo con olvido exponencial dinámico, U=30).

bajo nivel (*early vision*), no se ha abordado en este capítulo el análisis de técnicas de estimación de movimiento excesivamente costosas, como puede ser el caso de los estimadores basados en el cálculo del flujo óptico.

Entre los métodos descritos se encuentra una propuesta de estimador de movimiento, cuyas principales novedades son su rapidez y su capacidad para simultanear los procesos de reforzado espacial de los objetos detectados con el etiquetado de los mismos. Este doble procesado supone un importante ahorro de tiempo, que permite que posteriores procesos de más alto nivel puedan analizar la secuencia de imágenes para detectar más eficientemente los objetos presentes en la escena, tal y como se ha mostrado en los resultados presentados.

Finalmente, hay que matizar que el proceso de segmentación que se fundamenta en los conceptos descritos en el presente capítulo se realiza de forma independiente sobre cada fotograma de la secuencia. Esto es así porque, aunque los métodos espacio-temporales presentados incluyan información de fotogramas adquiridos en instantes de tiempo distintos, en ningún momento se lleva a cabo una segmentación simultánea de varios fotogramas. El título de este capítulo, segmentación espacial en el tiempo, pretende expresar esta idea, que básicamente consiste en que la dimensión temporal sólo permite ubicar la segmentación espacial de cada fotograma y, aunque estos datos puedan ser empleados en fotogramas posteriores, no deja de ser una información extraída por un procesado exclusivamente espacial.

Capítulo 4

Segmentación espacio-temporal

1 Introducción

Tal y como se ha comentado en capítulos previos, la segmentación de imagen real dista de ser un problema de fácil solución debido a la amplia variedad de criterios que utiliza el ser humano para distinguir y separar los objetos presentes en la escena. El capítulo 2 presentó las técnicas convencionales de segmentación, consistentes en utilizar una o varias características extraídas directamente de la escena, y dividir ésta en regiones homogéneas conforme a la selección realizada. Desafortunadamente, en el caso más general, los objetos no presentan un formato homogéneo en función de dichos rasgos cuantizables, como pueden ser el color, la textura o la intensidad. En estos casos es necesario disponer de más información para poder segmentar la escena y extraer de ella las regiones de interés. A este respecto, el capítulo 3 mostraba técnicas de segmentación espacial para secuencias de imágenes, mediante las cuales se extraía una máscara y se separaban las zonas estáticas de las formas en movimiento mediante sustracción. De esta manera, la información adicional que aporta el trabajar con varias imágenes permitía seleccionar entidades no homogéneas siempre y cuando, formando un bloque compacto, cambiaran su posición entre fotogramas sucesivos. Debe notarse que en el desarrollo de los métodos presentados en esta tesis, hasta este momento, aún no se han usado criterios temporales en un sentido estricto, sino criterios espaciales que se consolidan a lo largo del tiempo. Esta forma de trabajar con secuencias de imágenes presenta claras deficiencias, entre las que cabe destacar las siguientes:

- Es imposible distinguir entre entidades próximas que se desplazan con diferentes velocidades, tanto en valor absoluto como en dirección.
- El método no es aplicable a situaciones en que toda la escena se desplaza, aunque sea mínimamente.

Para resolver el problema en un contexto más general es necesario recurrir a descriptores de movimiento que permitan separar con eficacia áreas que se desplazan a distintas velocidades. Este capítulo se centra en la segmentación espacio-temporal de secuencias de imágenes, donde no sólo se definirán regiones de acuerdo a la homogeneidad de sus características estáticas, sino que además se les exigirá que sean coherentes respecto a sus parámetros de movimiento.

2 Técnicas de segmentación basadas en movimiento

Los sistemas de segmentación por movimiento se basan en agrupar los distintos píxeles de una imagen según sus parámetros cinéticos en lugar de sus características estáticas. La estimación robusta del movimiento de una escena a partir de una imagen digital ha supuesto un importante desafío a la visión por computador. Estas técnicas parten del campo 2D de desplazamiento, que se obtiene mediante la proyección sobre el plano de la imagen de los vectores de desplazamiento 3D de la escena. Este campo de vectores debe estimarse a partir de las variaciones espacio-temporales de los patrones de intensidad de la imagen [Horn y Weldon 1988], pero aún no se ha llegado a una definición exacta del concepto de flujo óptico. La confusión al respecto se refleja en la gran variedad de aproximaciones existentes para su estimación, así como la abundante presencia en la literatura existente sobre el tema de trabajos que afirman que su estimación debe basarse en primitivas, tales como puntos singulares distintivos o líneas, más que en características directamente extraibles de píxeles aislados. En cualquier caso, una vez estimado el movimiento de las distintas zonas que componen la escena por cualquiera de los métodos disponibles, éste ha de ser segmentado para conseguir regiones homogéneas según dicho criterio.

En general, los mayores problemas con que tropiezan los métodos más comunes de estimación de movimiento son: i) el desplazamiento entre fotogramas suele ser muy variable y algunas veces alto respecto a la velocidad de captura; ii) en escenas en las cuales aparecen múltiples objetos independientes que se desplazan simultáneamente, el movimiento global no es consistente, por lo que aparecen discontinuidades importantes en el campo de velocidad de la imagen. Adicionalmente, todos los problemas intrínsecos al proceso de adquisición de la secuencia, como los cambios de velocidad de captura, las variaciones de iluminación, o las vibraciones de la cámara, aparecen como ruido que distorsiona la estimación del movimiento.

En función de la forma en que se enfoquen los problemas mencionados, las técnicas de segmentación por movimiento se dividen en dos grandes bloques: diferenciales y cualitativas. La principal diferencia entre unas y otras radica en que las primeras se limitan a evaluar una serie de parámetros locales para extraer descriptores de movimiento, mientras que las segundas

optan por definir entidades en la escena y tratan de establecer la correspondencia de dichas entidades a lo largo de dos o más imágenes para determinar su desplazamiento entre fotogramas consecutivos. Como punto en común, ambos sistemas comparten un preprocesamiento destinado a eliminar las discontinuidades de la secuencia en la medida de lo posible, consistente en:

- Prefiltrado para suavizar la imagen y eliminar el ruido.
- Cálculo de los parámetros de la imagen.
- Integración de estas medidas por interpolación.

Es necesario indicar que la principal diferencia entre las técnicas de segmentación presentadas en este apartado y las descritas en el capítulo anterior estriba en que las segundas no usaban ninguna estimación del movimiento a la hora de segmentar, sino que analizaban la diferencia de brillo entre los píxeles de la imagen actual y los pertenecientes a un mapa de bits definido previamente. Por ello, dichas técnicas no pueden considerarse técnicas de segmentación por movimiento en el sentido estricto.

2.1 Técnicas diferenciales

Las técnicas diferenciales se basan en analizar los distintos píxeles que componen las imágenes de la secuencia de forma puntual. De este modo, tratan de estudiar parámetros locales, habitualmente variaciones diferenciales de distintos órdenes. Una vez se dispone de dichos parámetros, las distintas regiones de la imagen pueden segmentarse en función del valor de éstos. Debe notarse que si la segmentación se limita a tener en cuenta los mencionados parámetros, a pesar de la base espacio-temporal del método, la imagen quedará dividida conforme a criterios temporales exclusivamente y, por tanto, no podrá distinguirse entre entidades que se desplazan a la misma velocidad. Afortunadamente, este problema tiene fácil solución si se incluyen parámetros estáticos junto a los descriptores cinéticos en la entrada al método de segmentación final.

Las aproximaciones más clásicas se limitaban a trabajar con derivadas de primer orden para evaluar posibles traslaciones de la escena utilizando la ecuación:

$$I(x,t) = I(x - \vee t, 0) \tag{4.1}$$

donde $\vee$ es el vector que expresa la velocidad de desplazamiento de cualquier píxel de la imagen 2D. El principal inconveniente que presentan estas aproximaciones es que, dado que no analizan la imagen globalmente para obtener información de conjunto, imponen unas restricciones

bastante severas al comportamiento de los móviles, los desplazamientos de cámara, el ruido y los cambios de parámetros ambientales. La limitación más habitual en estos casos es que la intensidad de un píxel de la escena debe conservarse, presente éste movimiento o no, lo que no es usual en imágenes reales debido a cambios de iluminación y el propio ruido de adquisición del dispositivo de captura. Sólo cuando se cumple el principio de conservación de la intensidad, se puede obtener el vector de velocidad de cada píxel a partir de su gradiente. En este caso, puede derivarse la siguiente ecuación a partir de la expresión anterior:

$$\bigtriangledown I(x,t) \cdot \vee + I_t(x,t) = 0 \tag{4.2}$$

siendo I_t la derivada de la intensidad y $\bigtriangledown I(x,t) = (I_x(x,t), I_y(x,t))^T$. Esta ecuación devuelve la componente normal del movimiento de áreas del espacio con intensidad homogénea, pero, incluso asumiendo que la intensidad se conserva, puede observarse que dicha ecuación presenta dos grados de libertad, por lo que es necesario imponer nuevas restricciones. Una demostración de este método aparece en los trabajos de Horn y Schunk [Horn y Schunk 1981], que fijan el grado de libertad adicional mediante una restricción global de suavidad. En su método, la velocidad se obtiene iterativamente para cada punto a partir de dos expresiones que tienen en cuenta la intensidad puntual y la local en un vecindario. Teóricamente, 100 iteraciones por píxel son suficientes para obtener una buena estimación. Otros métodos [Lucas y Kanade 1981] usan restricciones locales, ajustando el flujo en un vecindario mediante mínimos cuadrados. En este caso, se necesitan más parámetros que en el anterior y además se trabaja sobre ventanas, calculándose las derivadas espaciales mediante operadores de tipo máscara. Es necesario recordar que cualquier sistema basado en el uso de ventanas presenta como inconvenientes la presencia de ruido si éstas son pequeñas, el incremento de los errores de estimación si éstas son grandes, y un tiempo de de proceso considerablemente elevado si su tamaño se calcula de forma adaptativa, [Koo y Jeong 2000]. Otra alternativa consiste en calcular los gradientes mediante errores modelados por gausianas [Simoncelli et al. 1991]. Estas técnicas ofrecen la ventaja de poder separar los valores poco fiables de la aproximación a partir de los autovalores de la matriz con que se trabaja, pero también suponen un gasto computacional más que considerable.

Por otra parte, los métodos de segundo grado [Nagel 1983] imponen como condición adicional a la conservación de la intensidad, la conservación de su gradiente. Estos métodos siguen basándose en desarrollos iterativos, y manejan aún más parámetros. Su principal aportación es la posibilidad de resolver algunas oclusiones de forma puntual. Desafortunadamente, las restricciones adicionales eliminan la posibilidad de rotaciones y dilataciones en la imagen y, en cualquier caso, los algoritmos son menos precisos que los de primer grado [Barron et al. 1994].

Obviamente, en todos los métodos mencionados la intensidad debe ser diferenciable,

por lo que es imprescindible un filtrado previo de la secuencia. Además, salvo que se trabaje con más de dos imágenes a la vez, es condición necesaria que la estructura que presenta la intensidad en la escena sea casi lineal, al igual que si la secuencia de que se dispone es ruidosa [Barron et al. 1994].

2.2 Técnicas cualitativas

Algunos autores sugieren que la estimación precisa del movimiento a partir de una secuencia de imágenes es, en general, imposible, debido a las diferencias inherentes entre la variación de la intensidad y el campo motriz en sí [Verri y Poggio 1987]. Por ello, la mayoría de los métodos actuales suelen optar por usar estimaciones cualitativas del movimiento en lugar de técnicas diferenciales.

Las aproximaciones cualitativas, si bien no son tan precisas como las diferenciales cuando éstas son aplicables, ofrecen mejores resultados, ya que al trabajar con vecindades de píxeles suavizan el efecto de las discontinuidades de intensidad de la escena. Además, la aplicación de técnicas cualitativas no requiere restricciones tan severas sobre la escena. Estas técnicas están basadas en definir primitivas en cada fotograma y, posteriormente, relacionarlas en fotogramas consecutivos. Algunos métodos se basan en localizar líneas rectas [Sobottka et al. 1997], puntos singulares [Smith y Brady 1997] o líneas curvas [Isard y Blake 1996], ya que éstas son primitivas que se pueden extraer de forma rápida y sencilla. Sin embargo, en imágenes complejas, la ubicación de dichas primitivas puede variar considerablemente en función de las condiciones de captura e iluminación y, en muchos casos, pueden desaparecer en determinados fotogramas, por lo que en determinadas circunstancias es necesario recurrir a métodos predictivos para identificar las primitivas entre fotograma y fotograma.

Habitualmente, suelen emplearse filtros extendidos de Kalman (EKF) para este propósito [Faugeras 1993] [Rodríguez et al. 2000b], pero estos filtros asumen una relación lineal entre las primitivas en distintos fotogramas. Si en lugar de usar una aproximación de Taylor de primer orden en el EKF se utilizan iteraciones de Newton [Hanek y Schmitt 2000] este inconveniente desaparece, pero la complejidad computacional se eleva enormemente. Esta es la causa de que muchos de estos métodos estén limitados a aplicaciones muy específicas, con escenarios controlados.

Para evitar estos inconvenientes y conseguir una mayor consistencia entre las primitivas detectadas en fotogramas consecutivos, se suelen buscar regiones homogéneas dentro de cada imagen. Desafortunadamente, esta técnica conlleva una doble problemática: i) es difícil obtener

una segmentación fiable de imágenes complejas; ii) no es evidente encontrar una relación estable entre las regiones que aparecen en frames consecutivos. En general, el seguimiento se consigue maximizando una medida de parecido, que puede ser una correlación cruzada o el cálculo de mínimos cuadrados entre regiones enventanadas. Para resolver estos problemas es necesario segmentar simultáneamente en el espacio y en el tiempo.

3 Introducción a la segmentación temporal jerárquica

Uno de los mayores inconvenientes de la segmentación temporal de secuencias de vídeo es que la mayor parte de los métodos son lentos. Adicionalmente, los cambios bruscos de intensidad suelen influir negativamente en el funcionamiento de éstos. Por ello, algunos sistemas se basan en pirámides multirresolución, que son estructuras jerárquicas de datos que, una vez construidos sobre una imagen cualquiera, la contienen en distintos niveles de detalle. El efecto más apreciable de dichas pirámides es que en sus niveles altos la imagen presenta un menor número de nodos, sujetos a un filtrado de paso bajo cada vez mayor. Sobre estas estructuras, los métodos de cálculo de flujo pueden trabajar en los niveles de menor resolución para reducir su tiempo de cómputo y los cambios bruscos de la escena y, a continuación, propagar los resultados a niveles de mayor grado de detalle. En la mayoría de los casos, es necesario un procesamiento posterior para refinar el flujo cada vez que se desciende a un nivel de resolución mayor, pero generalmente éste se suele llevar a cabo de forma inteligente, únicamente sobre determinadas áreas de píxeles, para mantener el tiempo de proceso acotado.

Una de las estructuras de datos más habituales a este respecto es la pirámide laplaciana [Burt 1983], que se construye de la siguiente forma, partiendo de un nivel 0, que es la imagen a procesar en sí:

- Construir una pirámide estándar de resolución decreciente mediante promediado o submuestreo.
- Construir una pirámide gaussiana mediante la convolución de cada uno de los píxeles de un nivel en un vecindario 5x5. Para obtener el siguiente nivel submuestrear el resultado de la convolución.
- Cada nivel l de la pirámide laplaciana se obtiene mediante la diferencia píxel a píxel entre el nivel l de la pirámide gaussiana y la pirámide construida por promediado o submuestreo.

Puede observarse que, igual que ocurría cuando se trabajaba de forma no jerárquica, en todos

los niveles de la estructura se lleva a cabo un proceso de suavizado.

El primer método que emplea este tipo de estructuras aparece en [Anandan 1987] y se basa en minimizar mediante operadores de Beaudet [Beaudet 1978] el error cuadrático medio resultante de asociar el flujo de movimiento de cada nodo del nivel de trabajo a la vecindad 3x3 de nodos del nivel inmediatamente inferior, comenzando desde el nivel de menor resolución. Una vez se ha concluido la operación en un nivel, se aplica al siguiente. Puede observarse que el método se limita a comparar las regiones que define la pirámide sobre la base a un nivel de resolución cada vez mayor. El principal problema de este sistema estriba en su enorme lentitud, así como en el hecho de que los resultados que ofrece no son siempre fiables. Para paliar, al menos en parte, estos inconvenientes, Singh [Singh 1990] propone utilizar igualmente la pirámide, sólo que en este caso los errores cuadráticos medios se calculan mediante un método distinto, obteniendo una medida de confianza a partir de la matriz de covarianza de dichos errores. Además, los desplazamientos se propagan usando restricciones en un vecindario, obteniéndose así un mejor refinado. De forma similar, Bandera [Bandera 1994] utiliza un método de correlación entre ventanas dispuestas sobre los niveles de dos pirámides construidas sobre fotogramas consecutivos para relacionar las regiones que se definen en ambos fotogramas. Dado que en niveles altos de la estructura los desplazamientos son pequeños, se puede trabajar con ventanas de dimensiones reducidas para que el tiempo de procesamiento no se dispare. Además, debido a que en niveles inferiores se aprecian más elementos relevantes y texturas, la ventana puede mantenerse con un tamaño reducido, ya que incluso en vecindarios pequeños aparece información suficiente para la caracterización.

En lugar de emplear técnicas cualitativas de comparación de regiones, otros sistemas optan por usar métodos diferenciales en los niveles de baja resolución de las mencionadas estructuras piramidales [Hartley 1985] [Bandera 1994] [Mahzoun et al. 1999]. Su único inconveniente es que las regiones deseadas no se definen de forma implícita, sino que se requiere un posterior proceso de segmentación del flujo resultante para obtenerlas [Hartley 1985]. La principal ventaja de utilizar estructuras jerárquicas en estos casos es que en los niveles más altos de la estructura tienden a cumplirse todas las restricciones que requieren este grupo de métodos. Si los métodos de refinamiento son suficientemente inteligentes, pueden obtenerse buenos resultados incluso para secuencias imposibles de procesar a resolución homogénea por implicar un tiempo de procesamiento elevado o cambios de intensidad en la secuencia. Así, por ejemplo, en [Mahzoun et al. 1999] se propone predecir el aspecto del fotograma siguiente a partir de la estimación de flujo actual propagado al siguiente nivel de resolución y minimizar iterativamente la diferencia entre el fotograma generado y el real para obtener el flujo en cada nivel.

4 Segmentación espacio-temporal mediante pirámides

Tal como se comentó en el capítulo 2, las pirámides pueden utilizarse para obtener una segmentación jerárquica mediante la estabilización adaptativa de la estructura de acuerdo a una o varias características de una imagen [Burt 1983]. Sin embargo, los métodos jerárquicos comentados en el apartado anterior no aprovechan en su totalidad las ventajas que ofrece la estructura, ya que únicamente la usan como herramienta de filtrado y reducción de resolución. Resulta mucho más interesante, desde el punto de vista del procesado jerárquico, explotar la mencionada estructura, en particular la red de enlaces que relaciona los nodos en niveles superiores con regiones de píxeles en la base. A este respecto, cabe preguntarse si un juego de enlaces similar no podría asimismo relacionar los nodos de los niveles superiores de una imagen cualquiera con regiones en la base de la imagen inmediatamente anterior en la secuencia. Una estructura de enlaces de este tipo permitiría relacionar ambas imágenes mediante los enlaces entre una y otra, siempre y cuando no se pierdan las relaciones topológicas entre los distintos niveles o, lo que es lo mismo, siempre que las estructuras estén adaptativamente estabilizadas.

En el capítulo 2 se mostró cómo una pirámide estabilizada permite relacionar cada nodo de la estructura con un área irregular de píxeles de nivel de gris homogéneo en la base. La propuesta presentada a continuación consiste en estabilizar dos pirámides de forma combinada. Así, no sólo se obtiene una relación entre cada nodo y la base de la pirámide a la que dicho nodo pertenece, sino que también está relacionado con la base de la pirámide construida y estabilizada para el fotograma anterior. Para la implementación del nuevo esquema de enlazado espacio-temporal se propone el siguiente algoritmo [Rodríguez et al. 2001b]:

1. Sea el nivel l=0.
2. Enlazar cada nodo (hijo) ${}^{t+1}C_l(j)$, ubicado en el nivel l de la pirámide $t+1$, al nodo (padre) que presente un nivel de gris más parecido al suyo propio. Este nodo-padre se elige de entre los nueve nodos situados inmediatamente sobre él en el nivel $l+1$ de la pirámide $t+1$.
3. Enlazar cada hijo ${}^{t}C_l(j)$, en el nivel l de la pirámide t, al padre que presente un nivel de gris más parecido al suyo propio. En este caso, los nueve candidatos estarán situados inmediatamente sobre la posición que ocupa el nodo ${}^{t+1}C_l(j)$ en el nivel l de la pirámide $t+1$.
4. Una vez que todos los nodos del nivel l de ambas pirámides se han reenlazado, se recalculan los valores de nivel de gris de los nodos del nivel $l+1$ en la pirámide $t+1$ como la media

del valor de los hijos que se encuentran actualmente enlazados a ellos en ambas pirámides. Puede observarse que un padre cualquiera puede tener enlazados hasta un total de 72 hijos, 36 en cada pirámide, por trabajar con una vecindad 3x3.

5. Si no aparecen cambios en el nivel de gris de los nodos del nivel $l+1$ en la pirámide $t+1$ por encima de un cierto umbral, sea $l = l+1$, volver al paso 2 hasta que l sea igual al nivel de trabajo. En caso contrario, se repiten los pasos 2, 3 y 4.

La elección de un nivel de trabajo adecuado debe llevarse a cabo cuidadosamente, ya que el número de nodos de dicho nivel fija el número de clases de la segmentación resultante. En principio, basta con garantizar que dicho número de nodos sea mayor que el número de regiones resultantes en la base, pero este problema se contemplará de forma más detallada en el apartado 6 del presente capítulo.

De forma análoga a la segmentación sobre una única imagen analizada en el capítulo 2, el proceso propuesto realiza una segmentación de forma combinada sobre dos imágenes consecutivas que quedan enlazadas a la pirámide $t+1$. Debe notarse, no obstante, que mientras la imagen t está igualmente enlazada a través de su propia pirámide a la imagen $t-1$ y así sucesivamente hasta llegar a la pirámide 0, la estructura de clases tiende a conservarse a lo largo de toda la secuencia. El proceso iterativo de segmentación garantiza que un nivel determinado no se estabiliza hasta que sus nodos están asociados a regiones de brillo homogéneo en el nivel inferior de ambas pirámides. Siguiendo la estructura de enlaces de dos pirámides es, por tanto, posible asociar dos a dos las regiones resultantes en fotogramas consecutivos y, estudiando el desplazamiento de sus centroides, determinar el movimiento que han efectuado.

Una vez se alcanza el nivel de trabajo deseado, que habitualmente se elige de un tamaño de 8x8 ó 16x16 nodos, cada nodo situado entre dicho nivel y la base se encuentra enlazado a una región homogénea de píxeles de similar nivel de gris en la base de la pirámide $t+1$, pero también en la base de la pirámide t (Fig. 4.1). Como consecuencia del proceso de estabilización, dichas regiones son la misma en ambas imágenes, ya que los nodos se enlazan al nodo más adecuado de acuerdo a su naturaleza y posición. Así, cada nodo sirve de referencia para obtener la posición de una región a través de una secuencia de imágenes, según los enlaces que mantienen con dichas imágenes. A efectos de estimar el flujo óptico en dichas regiones, bastaría con evaluar el desplazamiento de su centroide.

La Fig. 4.2 presenta un ejemplo de aplicación del método propuesto para una secuencia artificial donde un cuadrado se desplaza muy lentamente hacia la derecha a lo largo de cuatro fotogramas consecutivos, que constituyen la base de cuatro pirámides (secuencia # 1). La

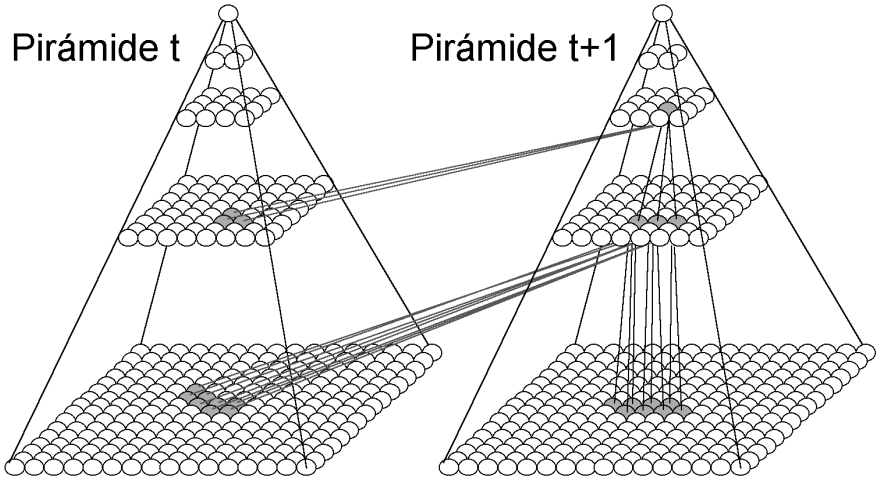


Figura 4.1: Estructuras estabilizadas de forma combinada.

pirámide **0**, situada a la izquierda de la Fig. 4.2.a, está construida sobre el primer fotograma y, por tanto, se estabiliza únicamente en el espacio, ya que no existe ninguna pirámide anterior con la que enlazarse en el tiempo. La simple observación de los niveles de las pirámides permite constatar que la división en clases en la base se lleva a cabo correctamente ya que, de no ser así, en los niveles superiores de la pirámide aparecerían nodos grises que se identificarían con las clases en que se mezclan píxeles negros y blancos. Esta pirámide **0** es utilizada a la hora de estabilizar la pirámide **1**, que aparece justo a continuación. Puede observarse que también en este caso la segmentación es correcta, pero además es necesario constatar que es correcta en ambos fotogramas, ya que ahora los nodos de la pirámide **1** están enlazados a ambas imágenes y el efecto de los nodos grises se apreciaría ante la existencia de enlaces incorrectos. Si se observan el resto de las pirámides de la secuencia, puede apreciarse que en todos los casos la segmentación se ha llevado a cabo adecuadamente. Otra forma de evaluar los resultados del proceso consiste en

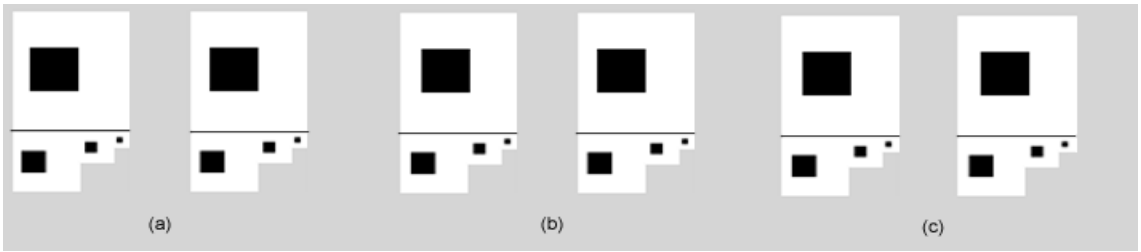


Figura 4.2: Análisis de la secuencia # 1. Base y niveles 32x32, 16x16, y 8x8 tras la estabilización combinada de : a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3.

representar el movimiento estimado de las regiones que componen la escena que, en este caso y debido a que se está trabajando en el nivel 4x4 de la pirámide, son 16. Si la forma de las regiones cambiase notoriamente entre fotogramas consecutivos, los desplazamientos serían incorrectos, ya que la posición de los centroides presentaría cambios que obedecerían a la presencia de errores de este tipo y no al desplazamiento real de las regiones. La Fig. 4.3 presenta los desplazamientos estimados de los fotogramas **0** a **1**, **1** a **2** y **2** a **3**. Tal como se observa, el único desplazamiento apreciable es el del cuadrado negro, que se mueve lenta pero progresivamente hacia la derecha de la imagen. También es conveniente observar que aparecen unos vectores de desplazamiento de menor intensidad en las áreas delantera y posterior de dicho cuadrado, que obedecen a la reagrupación de clases generada por la oclusión y el descubrimiento de las zonas del fondo por las que transita el cuadrado.

El método propuesto comparte un inconveniente con las técnicas diferenciales y cualitativas presentadas, referido al problema de la apertura. Sólo ofrece buenos resultados cuando los desplazamientos a que están sometidos los objetos de la escena son pequeños, ya que si la variación entre fotogramas es muy grande, un píxel perteneciente a una imagen en el instante t no podría enlazarse al mismo padre que su equivalente en la imagen $t+1$, ya que éste se encontraría muy alejado de él y, por tanto, en ningún caso pertenecería a la vecindad de 3x3 nodos que se analiza a la hora de escoger un nodo padre.

La Fig. 4.4 ilustra este caso, donde se aprecia que el disquete que se desplaza por la escena presenta una velocidad mucho mayor que el cuadrado del caso anterior, tal como puede apreciarse en las oclusiones y descubrimientos que aparecen de un fotograma al siguiente (secuencia # 2). Si bien la pirámide estabilizada únicamente en el espacio, que aparece a la izquierda de la Fig. 4.4.a, no está distorsionada, debido a que los nodos pueden enlazarse

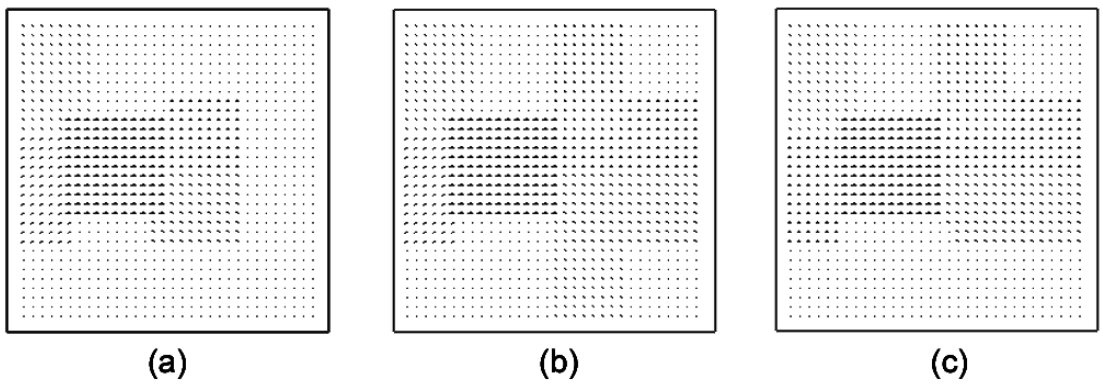


Figura 4.3: Análisis de la secuencia # 1. Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3.

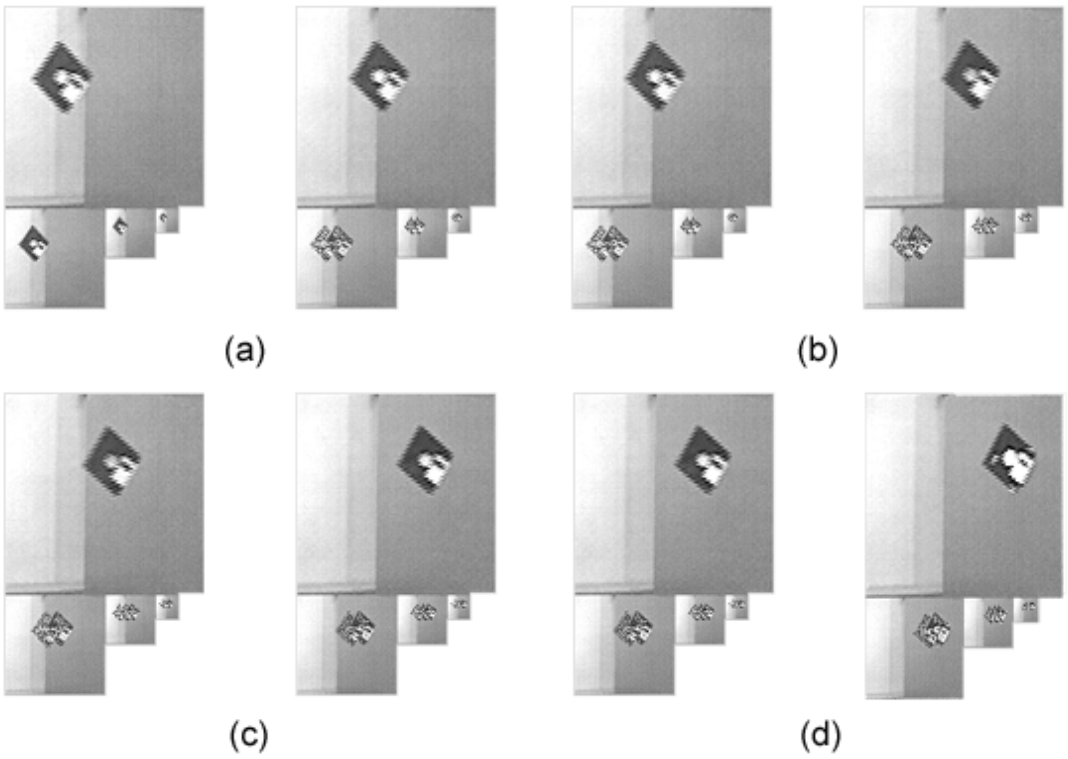


Figura 4.4: Análisis de la secuencia # 2. Base y niveles 64x64, 32x32 y 16x16 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

correctamente al padre adecuado, la excesiva separación del disquete en el fotograma **1** frente a su posición en el fotograma **0** imposibilita que los píxeles que lo componen se enlacen a los mismos padres en la pirámide **1**. Por ello los niveles de dicha pirámide aparecen distorsionados, pudiendo observarse la aparición de un disquete 'fantasma' en aquellas áreas en que se ha forzado el enlace de hijos de distintos niveles de gris a un mismo padre. Este efecto aparece porque, en el enlazado adaptativo, un hijo debe forzosamente enlazarse a un padre, independientemente de si encuentra alguno que presente un nivel de gris similar al suyo propio. Así, cuando un píxel del disquete en la pirámide **0** sólo encuentra nodos padre pertenecientes al fondo en la vecindad 3x3 de la pirámide **1** en la que busca su padre, acabará enlazándose al que presente el nivel de gris más parecido al suyo, independientemente de lo diferentes que puedan ser. El nivel de gris de dicho padre se verá muy afectado por la adición de hijos ajenos, y acabará adoptando un valor de nivel de gris intermedio para adapatarse lo mejor posible a sus hijos en ambas pirámides. En estos casos, en lugar de producirse una estabilización adecuada, se adoptará una estabilización de compromiso que trate de adecuar lo mejor posible dos imágenes que no pueden, en principio, asociarse dos a dos. Además, la Fig. 4.4 muestra que este efecto se va agravando en niveles

superiores de la imagen, conforme se van acumulando errores de fotograma a fotograma.

Podría parecer que los resultados del proceso mostrado no son tan malos, en tanto que los errores se concentran en una zona pequeña y sólo parecen influir en la aparición de estelas de movimiento, pero la realidad es que la estimación de movimiento, uno de los objetivos del proceso, es completamente incorrecta en dichas áreas. Como ejemplo, puede apreciarse en la Fig. 4.5 que la estimación de movimiento resultante, obtenida a partir de las regiones que se definen mediante las pirámides de la Fig. 4.4, no es ni remotamente aproximada al movimiento del disquete. En estos casos, la única información válida extraída es la asociada al fondo estático de la escena, cuyo desplazamiento nulo se obtiene correctamente.

Una primera solución al problema expuesto radica en aumentar la vecindad de búsqueda del padre de los nodos de la pirámide t sobre la pirámide $t+1$. Así, aunque éstos se hubiesen alejado en el espacio, cabría la posibilidad de encontrarlos. La Fig. 4.6 muestra el mismo experimento de la Fig. 4.4, pero utilizando ahora una vecindad 15x15 en lugar de una vecindad 3x3. Puede apreciarse que, si bien aparece aún cierto ruido en los bordes del objeto en movimiento debido a los efectos provocados por el descubrimiento y la oclusión originados por el propio movimiento, las distorsiones en niveles altos de las pirámides ya no son tan notables como en el caso anterior. El desplazamiento estimado para las regiones resultantes de la segmentación combinada aparecen en la Fig. 4.7 y puede observarse cómo, en este caso, los vectores de desplazamiento son bastante coherentes con el movimiento del objeto, sobre todo si se tiene en cuenta que se trabaja en un escenario real sobre un fondo no homogéneo.

A pesar de los resultados que ofrece el aumento de la vecindad de búsqueda, esta solución no es válida en la práctica por varios motivos. Primero, podría señalarse que la vecindad donde debería buscarse un padre potencial depende de la velocidad con que se desplace el objeto en

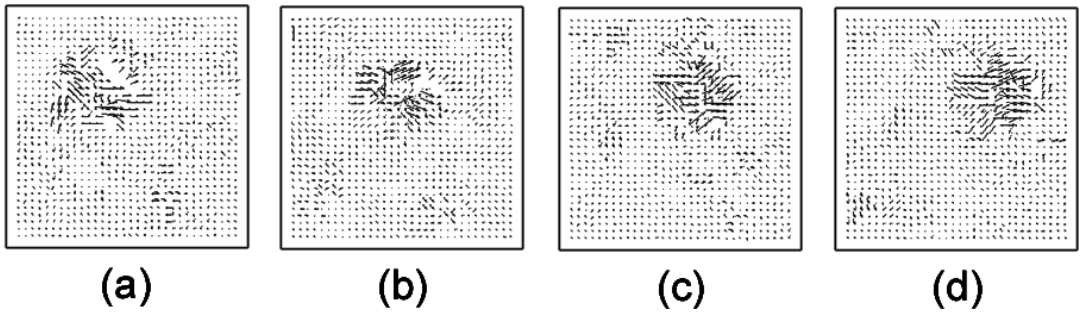


Figura 4.5: Análisis de la secuencia # 2. Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

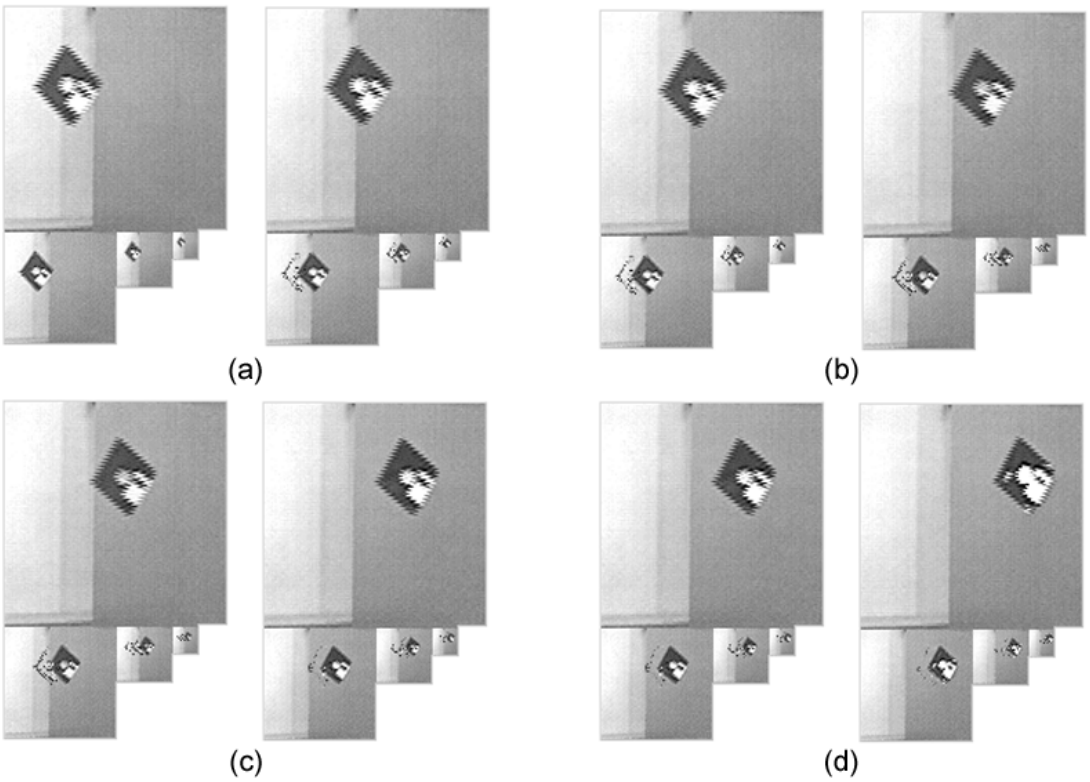


Figura 4.6: Análisis de la secuencia # 2 (vecindad 15x15). Base y niveles 64x64, 32x32 y 16x16 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

movimiento y, por tanto, no sólo cambia de una secuencia a otra sino que también cambia de un objeto a otro de la misma escena. Se podría pensar en empezar utilizando una vecindad muy grande para luego adaptarla dinámicamente a cada objeto en función de la estimación de movimiento más reciente, pero ello provocaría problemas de enlazado en las fronteras de los objetos que se desplazan a velocidades distintas. En segundo lugar, el enlazado adaptativo es un proceso iterativo que para niveles bajos de la pirámide se ejecuta sobre un gran número de nodos. Si además se trabaja con vecindades grandes, la carga computacional crece hasta niveles impracticables para el procesamiento de secuencias de vídeo. Por citar un ejemplo, en la secuencia de la Fig. 4.6 cada pirámide tardó 4534 veces más en estabilizarse que su equivalente de la Fig. 4.4. En ambos casos, los fotogramas presentaban un tamaño de 128x128 píxeles y el nivel de trabajo elegido fue de 4x4 nodos.

Antes de continuar cabe mencionar que los métodos analizados previamente, tantos las diferenciales como las cualitativas, adolecen del mismo inconveniente que el método propuesto. Es decir, la mayoría de ellos falla cuando el desplazamiento del móvil supera una velocidad

máxima relacionada con la apertura de búsqueda de cada algoritmo. En estos casos, al igual que con el método piramidal propuesto, se puede optar por ampliar el área de búsqueda de píxeles, para las técnicas diferenciales, o el área de búsqueda de las primitivas previamente extraídas de las imágenes, en el caso de las técnicas cualitativas. En ambos casos, la ampliación de éstas áreas induce el aumento de errores en los resultados al existir más candidatos inapropiados con los que realizar las comparaciones. Cabe resaltar que los métodos piramidales pueden afrontar este problema debido a su naturaleza multinivel, que permite la aplicación de los algoritmos siguiendo una estrategia *coarse to fine*, que comience obteniendo una aproximación gruesa a la solución del problema en niveles altos de la estructura y continúe puliendo dicha solución en sucesivas aplicaciones en niveles de mayor detalle. Esta característica propia de las estructuras jerárquicas de procesado justifica sobradamente su empleo como arquitectura básica de segmentación espacio-temporal.

5 Enlazado predictivo jerárquico de imágenes consecutivas

Tal y como se ha comentado en el apartado anterior, el principal problema del método expuesto estriba en la restricción que impone a la velocidad de los móviles que aparecen en la escena. Para subsanar este inconveniente, dado que se dispone de una estimación del movimiento de las distintas regiones de la imagen, éstas pueden utilizarse para definir las áreas de búsqueda de los padres potenciales en la pirámide $t+1$ para nodos pertenecientes a la pirámide t. Así, si se prevé que un nodo situado en la posición (x_0, y_0) en la pirámide t va a sufrir un desplazamiento (a, b), en lugar de buscar su padre potencial en una vecindad 3x3 sobre el nodo (x_0, y_0), éste se buscará en una vecindad 3x3 sobre $(x_0 + a, y_0 + b)$. Retomando la nomenclatura del apartado anterior, y denominando ${}^t d_l(j)$ al desplazamiento estimado para el nodo ${}^t C_l(j)$ entre los instantes de tiempo

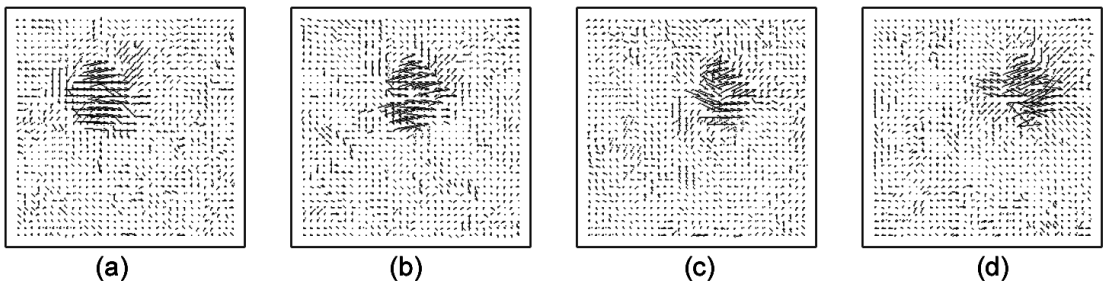


Figura 4.7: Análisis de la secuencia # 2 (vecindad 15x15). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

correspondientes a las pirámides $t-1$ y t, la Fig. 4.8 muestra cómo se definen las vecindades de búsqueda del nodo padre para los nodos ${}^{t+1}C_l(j)$ y ${}^{t}C_l(j)$.

Utilizando este sencillo sistema predictivo, se puede mantener siempre la vecindad de búsqueda de padres potenciales en un valor acotado de 3x3, independientemente del área de la imagen en estudio y de la velocidad de los móviles presentes en la escena. Con estas modificaciones, el algoritmo de segmentación pasa a constar de los siguientes pasos:

1. Sea el nivel l=0.
2. Enlazar cada hijo ${}^{t+1}C_l(j)$, del nivel l de la pirámide $t+1$, al padre que presente un nivel de gris más parecido al suyo propio. La vecindad de búsqueda la forman los nueve nodos situados inmediatamente sobre él en el nivel $l+1$ de la pirámide $t+1$ (Fig. 4.8.a).
3. Enlazar cada hijo ${}^{t}C_l(j)$, del nivel l de la pirámide t, al padre que presente un nivel de gris más parecido al suyo. La vecindad de búsqueda la forman los nueve nodos situados inmediatamente sobre el nodo ${}^{t}C_l(j) + {}^{t}d_l(j)$ en el nivel l de la pirámide $t+1$ (Fig. 4.8.b). El valor de ${}^{t}d_l(i)$ se corresponde con el desplazamiento, a nivel l, que la región enlazada al nodo ${}^{t}C_l(j)$ sufre entre los fotogramas $t-1$ y t, y se mide como la diferencia entre la posición de su centroide en ambos fotogramas. Este desplazamiento puede interpretarse como una estimación aproximada del movimiento de dicha región y, si no se dispone de estimación alguna para el nodo en particular, ${}^{t}d_l(j)$ vale cero.
4. Una vez que todos los nodos de los niveles l de ambas pirámides se han reenlazado, se recalculan los valores de gris de los nodos del nivel $l+1$ de la pirámide $t+1$. Para ello se

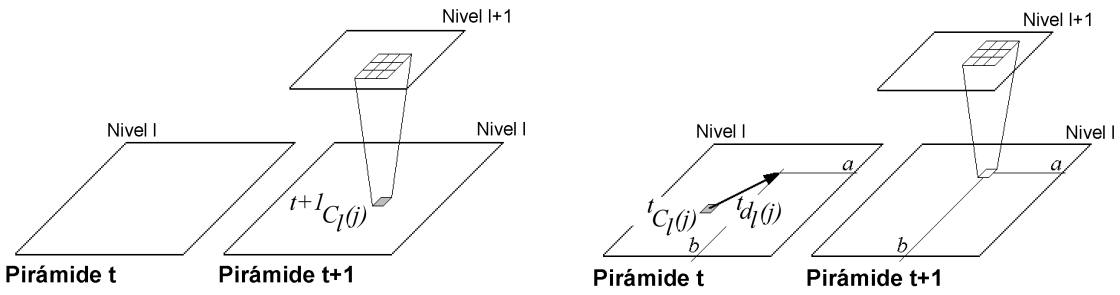


Figura 4.8: Enlazado predictivo: a) vecindad de búsqueda empleada para el enlazado de un nodo hijo de la pirámide $t+1$ con sus posibles padres en la misma pirámide; y b) vecindad de búsqueda empleada para el enlazado de un nodo hijo de la pirámide t con sus posibles padres en la pirámide $t+1$.

promedia el valor de los hijos que se encuentran actualmente enlazados a ellos en ambas pirámides.

5. Si no aparecen cambios en el valor de gris de los nodos del nivel $l+1$ de la pirámide $t+1$ por encima de un cierto umbral, sea $l = l+1$, se volverá al paso 2 hasta que l sea igual al nivel donde se desee trabajar. En caso contrario, se repiten los pasos 2, 3 y 4.

La Fig. 4.9 muestra el análisis de la secuencia # 1 que, como puede observarse, también se estabiliza correctamente por este procedimiento. En este caso, aunque no se aprecie macroscópicamente en la figura, el enlazado entre padres e hijos del objeto móvil es más coherente que en el caso anterior, ya que al realizar la búsqueda de padres con enlazado predictivo las posibilidades de acertar con el padre correcto aumentan considerablemente. En cualquier caso, esta secuencia presenta un móvil cuya velocidad de desplazamiento es lo suficientemente lenta como para que el desplazamiento del área de búsqueda añadido en la mejora que se comenta en este apartado no aporte ventajas significativas en este caso. Este aspecto se ilustra igualmente en los vectores de desplazamiento estimados (Fig. 4.10) que, aunque presentan pequeños cambios frente a los obtenidos sin la predicción de movimiento con el método de enlazado básico mostrados en la Fig. 4.3, resultan apenas perceptibles, y se concentran en aquellas zonas de movimiento que, por pertenecer a regiones de ocultamiento y descubrimiento, eliminaría un posterior proceso de umbralización.

Para comprobar, por tanto, las ventajas de este nuevo método, se ha aplicado sobre la secuencia # 2, en la que un disquete se desplazaba a una velocidad excesiva, lo que impedía que el proceso de reenlazado no predictivo funcionase correctamente. La Fig. 4.11 presenta los resultados obtenidos al estabilizar las pirámides de forma combinada usando el reenlazado predictivo. Inicialmente, cuando no se dispone de estimación alguna de movimiento, la estabilización combinada aún no se ha inicializado, con al consecuente aparición en los niveles

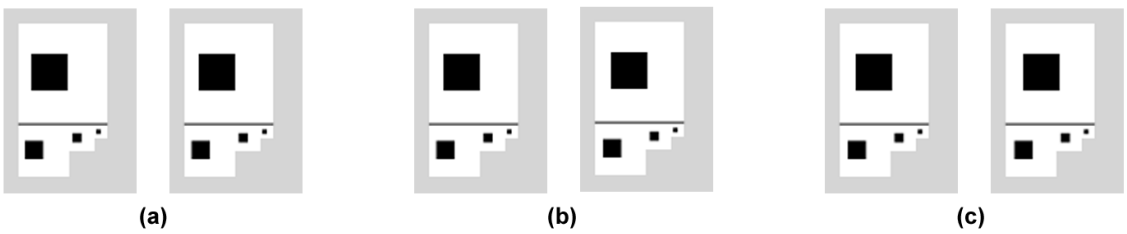


Figura 4.9: Análisis de la secuencia # 1 (enlazado predictivo). Base y niveles 32x32, 16x16 y 8x8 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3.

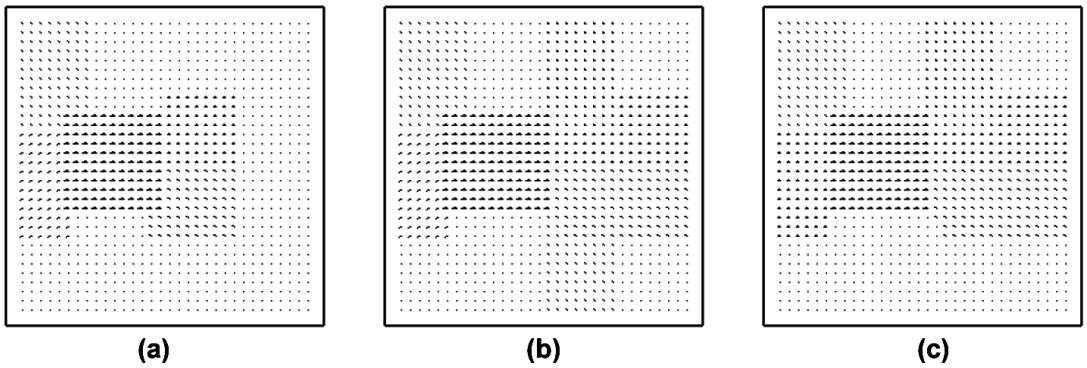


Figura 4.10: Análisis de la secuencia # 1 (enlazado predictivo). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3.

superiores de la pirámide estabilizada del anteriormente mencionado disquete 'fantasma' (Fig. 4.11.a). En los dos fotogramas siguientes (Figs. 4.11.b-c), el movimiento del objeto todavía no se ha estimado correctamente debido a esa falta inicial de datos, pero puede observarse que el disquete 'fantasma' aparece cada vez más desdibujado debido a que el número de píxeles correctamente enlazados aumenta progresivamente. Finalmente, en la Fig. 4.11.d puede observase que el efecto sólo es apreciable en unos cuantos nodos junto al borde del disquete en niveles superiores de la pirámide, que son más atribuibles a los efectos de las oclusiones y los descubrimientos producidos por el cuerpo en movimiento que a una estimación defectuosa del desplazamiento en sí.

La Fig. 4.12 muestra los vectores de desplazamiento estimados a partir de las pirámides de la Fig. 4.11, donde puede apreciarse, a nivel de regiones, la tendencia que se ha comentado sobre los niveles de las pirámides combinadas. Así, mientras que la primera estimación de flujo del disquete está muy distorsionada, llegando a patrones casi aleatorios en las distintas áreas que lo componen (Fig. 4.12.a), la segunda estimación (Fig. 4.12.b) comienza a mostrar una clara tendencia a la derecha, que se confirma en la tercera (Fig. 4.12.c). La cuarta estimación (Fig. 4.12.d) presenta una pequeña inclinación hacia la dirección noreste de la imagen, correspondiéndose con el inicio del ascenso pendular del objeto. Es importante notar que, si bien la estimación de movimiento no es en absoluto perfecta, es casi imposible conseguir precisión cuando se trabaja con imagen real. Especialmente, es de remarcar el hecho de que no se ha aplicado ningún tipo de control sobre la secuencia de prueba, por lo que la velocidad de captura puede no ser estable, las condiciones de iluminación son adversas debido al empleo de luz halógena no sincronizada con la cámara, el fondo no es homogéneo, la velocidad del móvil no es lineal y existe ruido de captura.

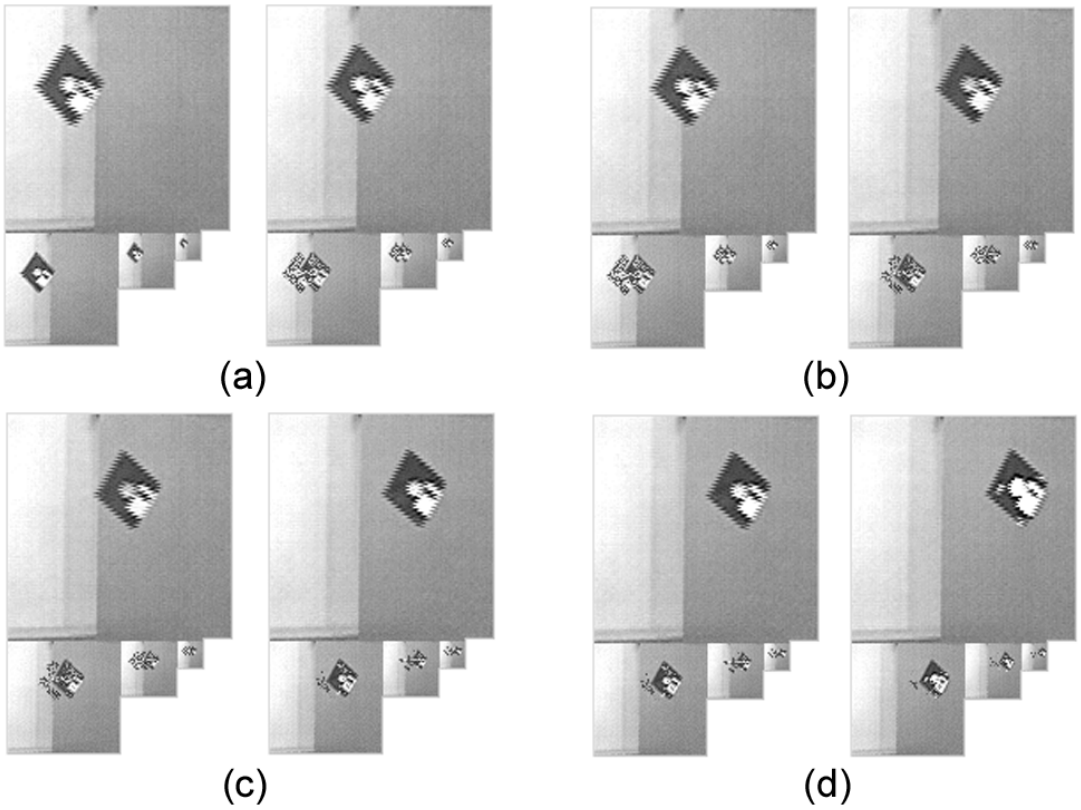


Figura 4.11: Análisis de la secuencia # 2 (enlazado predictivo). Base y niveles 64x64, 32x32 y 16x16 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

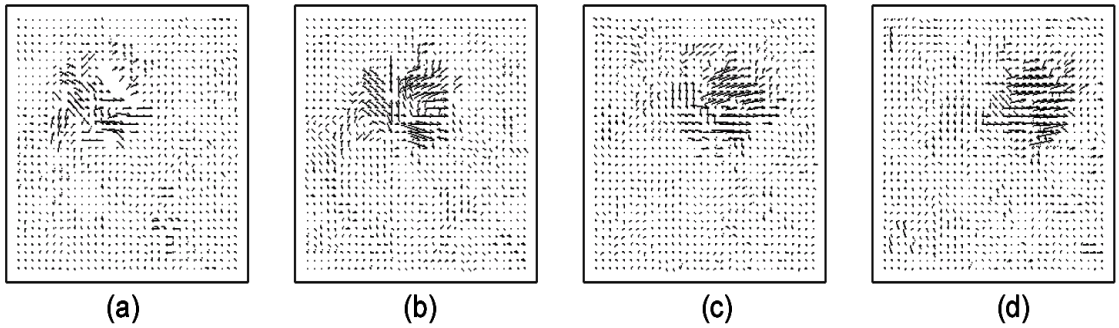


Figura 4.12: Análisis de la secuencia # 2 (enlazado predictivo). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

Para apreciar mejor el efecto de la predicción en el reenlazado adaptativo combinado de pirámides consecutivas, se ha creado una secuencia sencilla artificial no sujeta a ninguno de los problemas arriba mencionados. La secuencia # 3 presenta un cuadrado negro sobre fondo blanco homogéneo desplazándose a velocidad constante en sentido horizontal (Fig. 4.13) y diagonal (Fig. 4.15), así como en $zig-zag$ (Fig. 4.17) para probar el efecto de corrección que introduce el empleo de la predicción del movimiento en el enlace combinado adaptativo. En todos los casos puede observarse como, tras la inestabilidad derivada de la falta de una estimación inicial de movimiento, el sistema se recupera rápidamente y tiende a estabilizarse hacia una segmentación adecuada. Obviamente, aparece una pequeña distorsión en el tercer caso, cuando la dirección del movimiento cambia bruscamente y falla la predicción (Fig. 4.17.d), pero la segmentación se corrige en sólo dos fotogramas, lo que refuerza la capacidad de adaptación del método a pesar de no emplear modelos complejos de representación del movimiento.

Es interesante resaltar que el movimiento estimado no ofrece tan buenos resultados como sería deseable, ya que existe una gran cantidad de regiones pertenecientes al fondo blanco estático que presentan un desplazamiento que, si bien es menor que el del objeto en sí, no es en absoluto despreciable. Así, si bien se observa que el valor de desplazamiento del cuadrado se ha obtenido correctamente en los tres casos, los resultados globales no son plenamente satisfactorios. Esta limitación, asociada en este caso a las regiones que constituyen el fondo, se debe a que se está trabajando con un número de clases inadecuado, y su solución se analizará en el apartado siguiente.

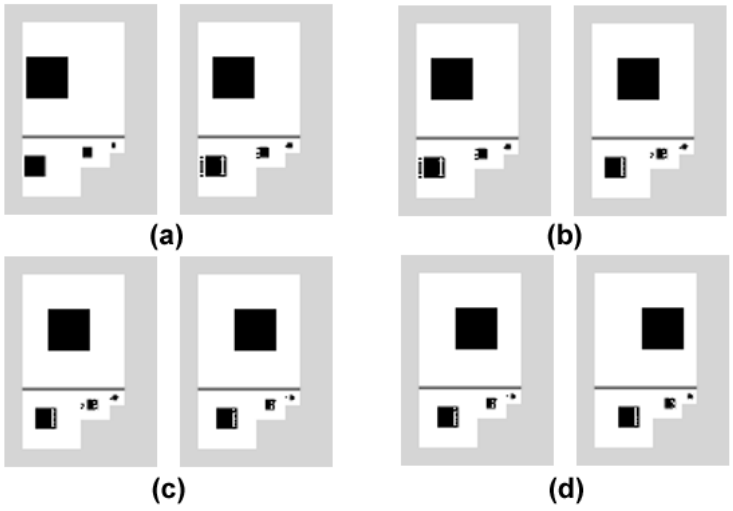


Figura 4.13: Análisis de la secuencia # 3 -movimiento horizontal- (enlazado predictivo). Base y niveles 32x32, 16x16 y 8x8 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

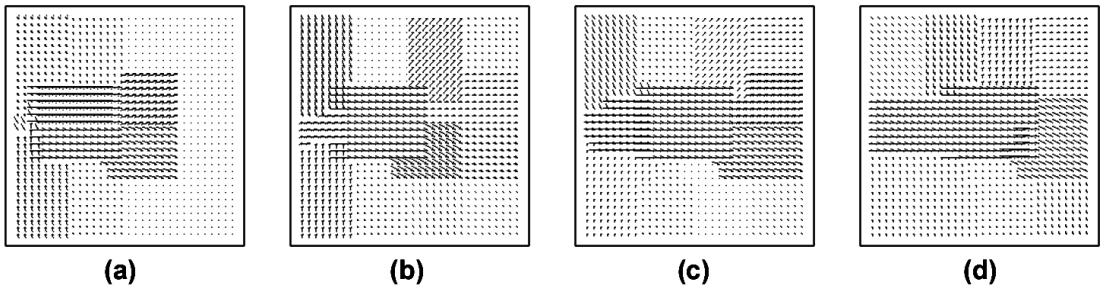


Figura 4.14: Análisis de la secuencia # 3 -movimiento horizontal- (enlazado predictivo). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

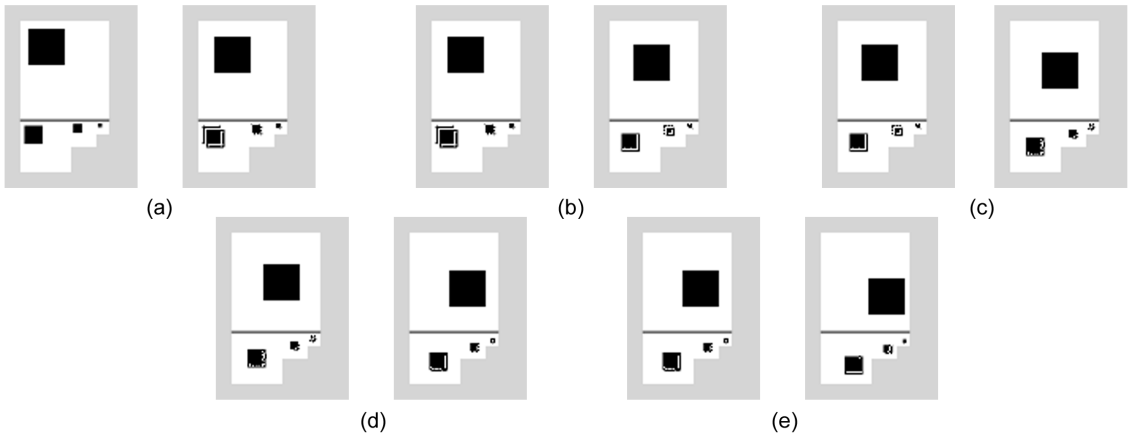


Figura 4.15: Análisis de la secuencia # 3 -movimiento diagonal- (enlazado predictivo). Base y niveles 32x32, 16x16 y 8x8 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

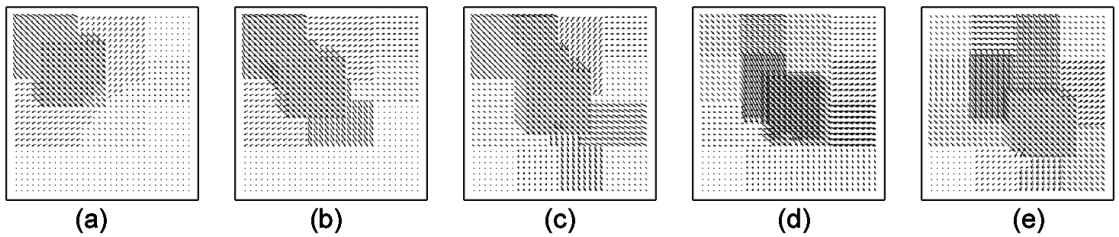


Figura 4.16: Análisis de la secuencia # 3 -movimiento diagonal- (enlazado predictivo). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

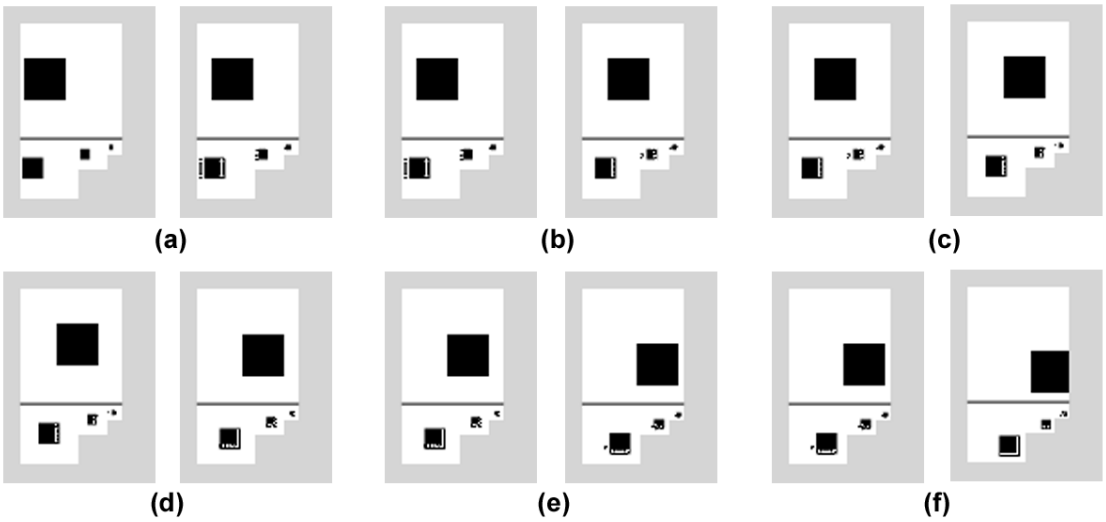


Figura 4.17: Análisis de la secuencia # 3 -movimiento en $zig-zag$- (enlazado predictivo). Base y niveles 32x32, 16x16 y 8x8 tras la estabilización combinada de: a) pirámide 0 y pirámide 1; b) pirámide 1 y pirámide 2; c) pirámide 2 y pirámide 3; d) pirámide 3 y pirámide 4.

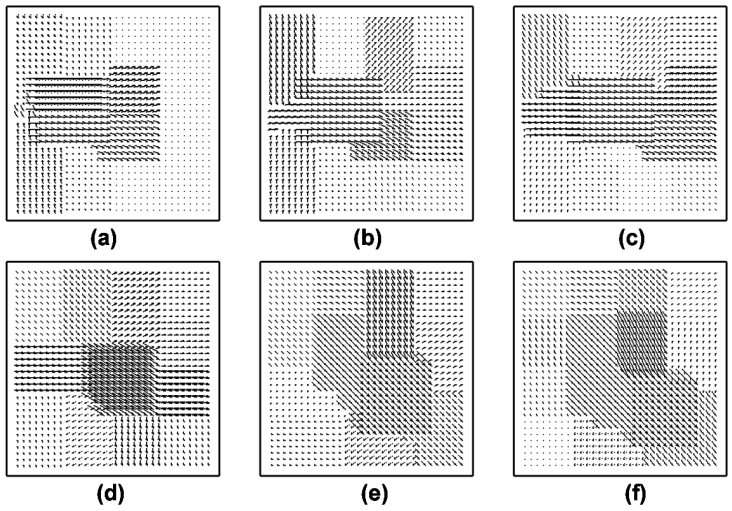


Figura 4.18: Análisis de la secuencia # 3 -movimiento en $zig-zag$- (enlazado predictivo). Vectores de desplazamiento de los píxeles de la escena entre: a) fotogramas 0 y 1; b) fotogramas 1 y 2; c) fotogramas 2 y 3; d) fotogramas 3 y 4.

6 Ajuste no supervisado de clases

Todas las estimaciones de movimiento presentadas hasta ahora han mostrado la aparición de áreas que, supuestamente, se desplazan de modo uniforme respecto al fondo de la imagen, a pesar de que no existe ningún desplazamiento en el área de la escena que ocupan. El efecto mencionado obedece a una reorganización aleatoria en la forma de las clases cuando el número de éstas es incorrecto. La segmentación jerárquica mediante pirámides fuerza una descomposición de la imagen original en tantas clases como nodos muestre el nivel de trabajo elegido. Es por ello que generalmente la imagen va a estar descompuesta en un número diferente de clases de las que realmente presenta, resultando ser el origen de la aparición de problemas para el método de segmentación y estimación de movimiento propuesto.

La Fig. 4.19 muestra los efectos de trabajar en los niveles de 2x2, 4x4, 8x8 y 16x16 nodos para una misma secuencia, presentándose los vectores de desplazamientos estimados para estas pirámides en la Fig. 4.20. Puede observarse que cuando el número de clases es pequeño (Fig. 4.19.a), el proceso de segmentación suele ofrecer malos resultados porque se fuerzan fusiones entre clases no homogéneas que no deberían en ningún caso producirse. En este ejemplo, el error es inmediatamente constatable observando la aparición de nodos de valores de nivel de gris medio, que en realidad no existen en la escena original. Este efecto puede eliminarse mediante el incremento del número de clases a obtener, es decir, trabajando en niveles con un mayor número de nodos. Por otro lado, cuando se fuerza un número elevado de clases, éstas ya no pueden identificarse con objetos reales compactos que presentan un movimiento homogéneo, sino que dichos objetos quedan fragmentados en varias clases. Este hecho no implicaría problema alguno si la segmentación ofreciese siempre los mismos resultados en cada fotograma, pero el hecho de que se puedan producir ocultaciones y descubrimientos, cambios de la iluminación y de las condiciones de captura, e incluso desplazamientos de la totalidad de la escena, origina variaciones en el proceso de enlazado que inciden en cambios notables del área y la forma de las clases. En las Figs. 4.19.b-c no se puede apreciar este efecto a simple vista, pero sí se pueden observar en los vectores de desplazamiento estimados a partir de ellas (Figs. 4.20.b-c). En dichas estimaciones aparecen áreas en movimiento que corresponden al fondo, pero que obedecen en realidad a alteraciones de la forma de las clases que lo componen y que, como consecuencia directa, inducen desplazamientos de los centroides de éstas, a pesar de ser el fondo una región homogénea única y estática. En la Fig. 4.19.d el efecto es aún más acusado, ya que parte del triángulo se ha fundido con el fondo de la imagen y aparece sesgado en la pirámide. La estimación del desplazamiento en este caso es la peor de todas (Fig. 4.20.d).

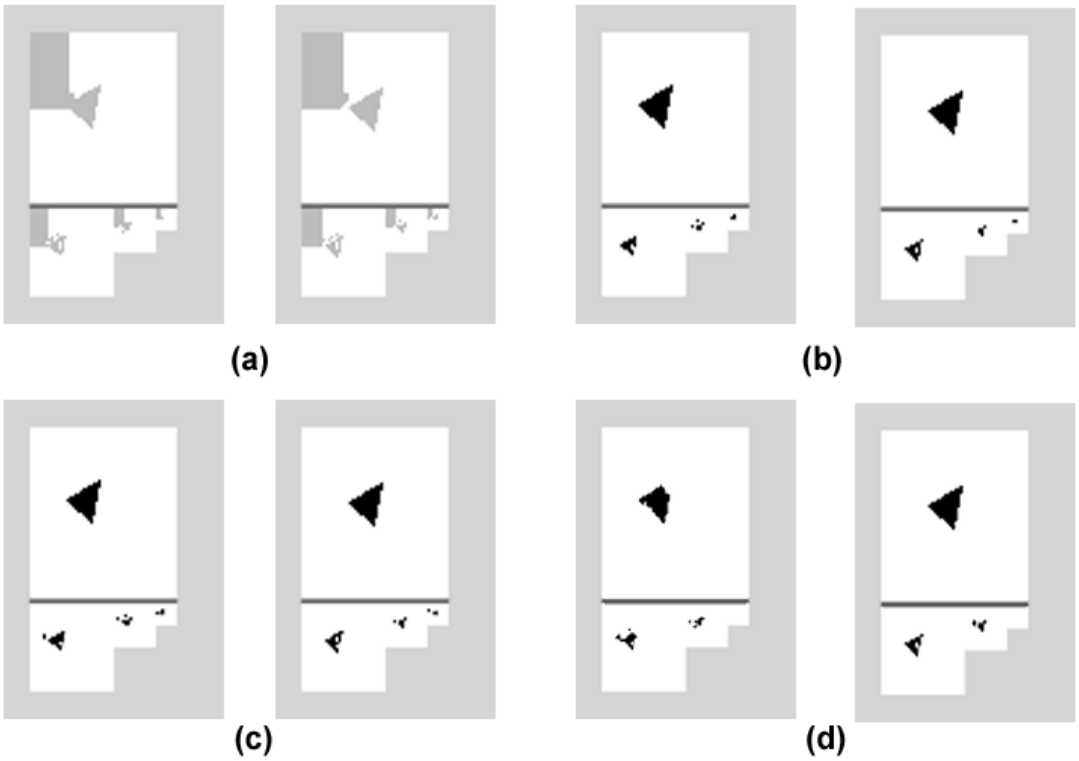


Figura 4.19: Pirámides combinadas para un nivel de trabajo de tamaño: a) 2x2; b) 4x4; c) 8x8; d) 16x16.

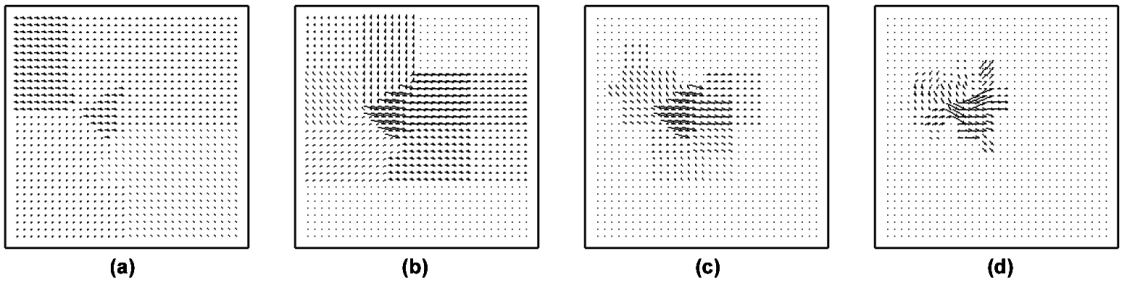


Figura 4.20: Vectores de desplazamiento estimados sin fusión de clases para el triángulo de la Fig. 4.19, usando un nivel de trabajo de tamaño: a) 2x2; b) 4x4; c) 8x8; d) 16x16.

El efecto podría, incluso, agravarse en imágenes reales donde texturas, sombras y cambios de iluminación fuerzan la aparición de clases inexistentes. Estas clases eliminan la posibilidad de que otras clases presentes en la imagen puedan aparecer, ya que el número de regiones que se pueden generar mediante el proceso de segmentación está limitado. Por ejemplo, la Fig. 4.21.a muestra la segmentación resultante de trabajar en un nivel 8x8 sobre la ya presentada secuencia del disquete en movimiento pendular (secuencia # 2). Los vectores de desplazamiento estimado a partir de dicha segmentación aparecen en la Fig. 4.21.b, donde puede observarse que la región metálica del disquete se ha perdido por completo y parece no presentar movimiento. Estudiando los distintos niveles de las pirámides segmentadas se puede descubrir que, por efecto de las sombras, las franjas de fondo aparentemente homogéneo a la izquierda y detrás del disquete se dividen en varias clases. Dado que el número de clases máximo ya se ha alcanzado, a la hora de asignar la parte metálica del disquete a una región, se fuerza que ésta se combine con aquella de las ya existentes que presenta un nivel de gris más parecido al suyo. Dado que la parte plástica del disquete es negra, el fondo presenta un color más parecido a la mencionada zona metálica y, por tanto, ésta se une a dicha clase a pesar de que ni siquiera están en contacto en la imagen original. Por otro lado, la masa de la zona metálica es casi despreciable frente al número de píxeles del fondo, con lo que el centroide de la clase combinada apenas sufre desplazamiento, por lo que a dicha área se le asocia un vector de desplazamiento nulo, como se aprecia en la Fig. 4.21.b.

Para resolver el problema de las fusiones incorrectas, pero también el de la fragmentación de objetos originada por la elección de un número de clases excesivamente elevado, se propone [Rodríguez et al. 2001a] trabajar inicialmente con un nivel que presente un número de nodos suficiente para garantizar la no aparición de fusiones indeseadas y llevar a cabo *a posteriori*

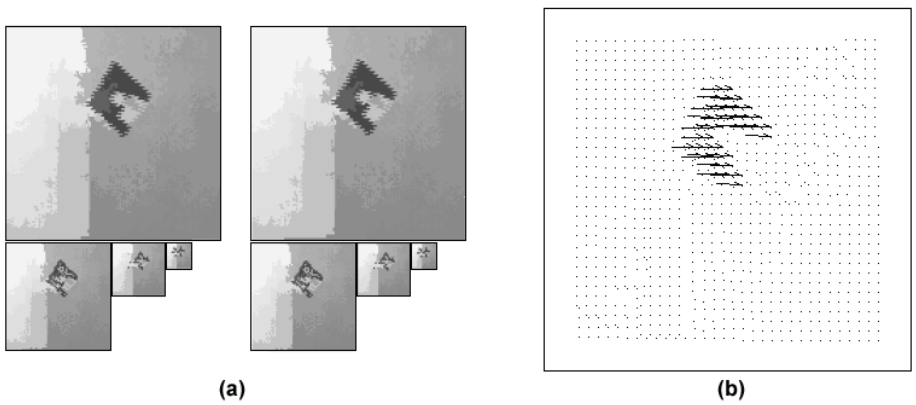


Figura 4.21: Fusión incorrecta de clases: a) segmentación en fotogramas consecutivos; b) movimiento estimado de las regiones definidas en a).

un proceso de fusión no supervisada que ajuste el número de clases a su valor real. Así, se construiría una única clase a partir de todas las que componen un objeto y, dado que la forma de éste se conserva a lo largo de la secuencia, también lo hará la forma de la clase resultante. En tanto que se dispone de una segmentación previa, esta fusión se puede llevar a cabo de una forma muy sencilla, evitando así el incremento innecesario de la carga computacional global del proceso. Los pasos que se proponen son los siguientes:

- Estimación de las $bounding - boxes$ de todas las regiones definidas por la segmentación piramidal. Estas $bounding - boxes$ son de forma rectangular, y vienen definidas por los puntos (i_{min}, j_{min}) e (i_{max}, j_{max}), que se corresponden con los valores mínimos y máximos de las coordenadas de los píxeles que componen cada región.
- Análisis de todas aquellas regiones cuyas $bounding - boxes$ se superponen. Sólo si las regiones que las forman comparten uno o más píxeles fronterizos, se permitirá la fusión de ambas clases, siempre y cuando sus niveles de gris sean similares; en caso contrario, las regiones se marcan como no aptas para la fusión.

Este proceso debe repetirse hasta que todas las $bounding - boxes$ superpuestas que queden en la imagen estén etiquetadas como no aptas para la fusión. Debe destacarse que los criterios utilizados ya suponen el descarte de un buen número de regiones y que, en aquellas que se analizan, sólo se estudian los píxeles fronterizos. Además, sólo las fronteras de las regiones que no se tocan se estudian en su totalidad. Debido a todo ello, el proceso de fusión es bastante rápido y no supone un deterioro importante del rendimiento global del sistema. La Fig. 4.22 presenta los tres casos posibles que pueden darse entre una pareja de regiones. Cuando sus $bounding - boxes$ no están superpuestas, la fusión directamente se rechaza (caso A). Si están superpuestas y existe algún píxel en contacto entre las regiones, éstas se funden (caso B). Si estando las $bounding - boxes$ superpuestas, se recorre completamente el contorno de cualquiera

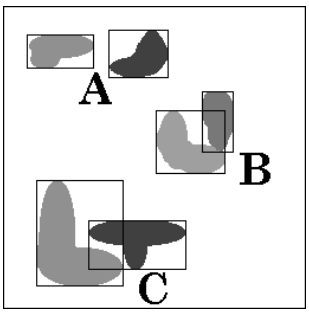


Figura 4.22: Casos de fusión posterior de clases generadas por la segmentación combinada de dos pirámides: a) no fusión; b) fusión; c) no fusión.

de las dos regiones y no se encuentra ningún píxel en contacto, las clases se etiquetan como no aptas para fusión (caso C).

La Fig. 4.23 muestra el mismo ejemplo de la Fig. 4.19, pero trabajando con el algoritmo propuesto para el ajuste no supervisado de clases. Puede apreciarse como, salvo en el primer caso en el cual el número de clases generadas es claramente insuficiente, el resto de los resultados son coherentes y presentan una estimación de movimiento correcta (Fig. 4.24). Esto es debido a que, independientemente del número de clases que devuelva la pirámide, el algoritmo de fusión las convierte en dos: el triángulo y su fondo blanco.

Finalmente, la única duda que podría surgir sobre el método propuesto estaría relacionada con el tratamiento de los móviles deformables, ya que en este caso se producen desplazamientos de los centroides, que no se deben al movimiento del objeto en sí, sino al cambio que presenta su forma. Sin embargo, salvo que estos cambios sean repentinos o excesivos, el método funciona de forma correcta sin necesidad de incluir modificación alguna. Así, la Fig. 4.25 muestra como, en una secuencia capturada con una cámara cenital, donde los móviles se desplazan alterando su forma (secuencia # 4), la segmentación se realiza correctamente y los vectores de desplazamiento (Fig. 4.26) son adecuadamente estimados a pesar de dichas deformaciones.

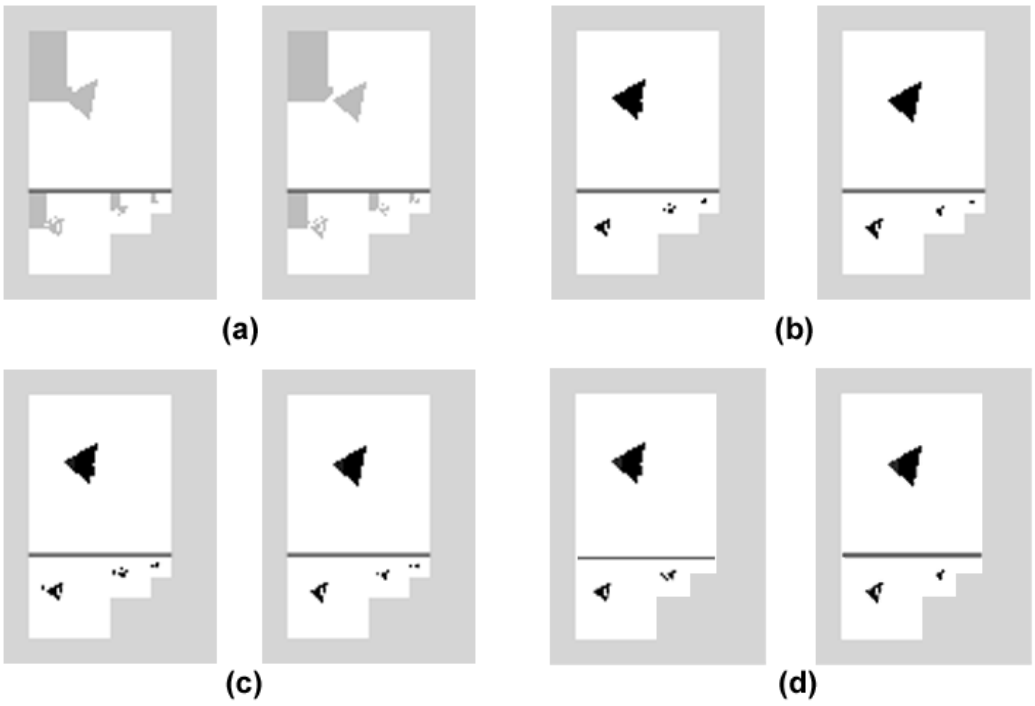


Figura 4.23: Pirámides combinadas para un nivel de trabajo de tamaño: a) 2x2; b) 4x4; c) 8x8; d) 16x16.

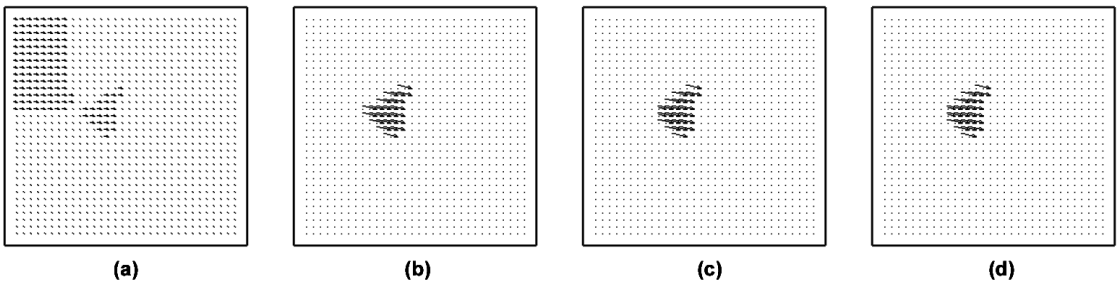


Figura 4.24: Vectores de desplazamiento estimados con fusión de clases para el triángulo de la Fig. 4.23 usando niveles de trabajo de tamaño: a) 2x2; b) 4x4; c) 8x8; d) 16x16.

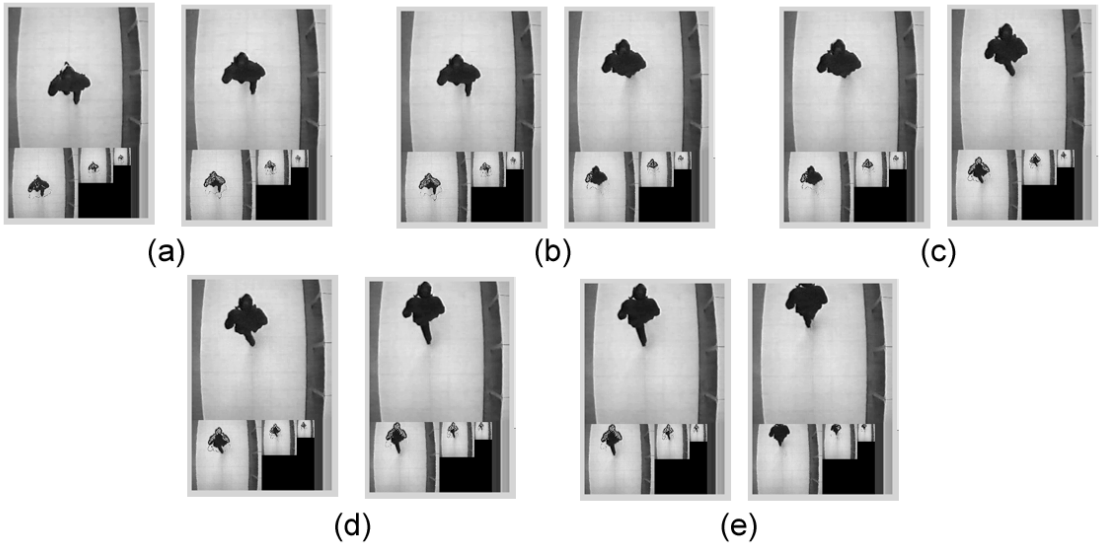


Figura 4.25: Análisis de la secuencia # 4 (enlazado predictivo). Pirámides combinadas generadas desde el nivel 8x8 entre los fotogramas: a) 1 y 2; b) 2 y 3; c) 3 y 4; d) 4 y 5; e) 5 y 6.

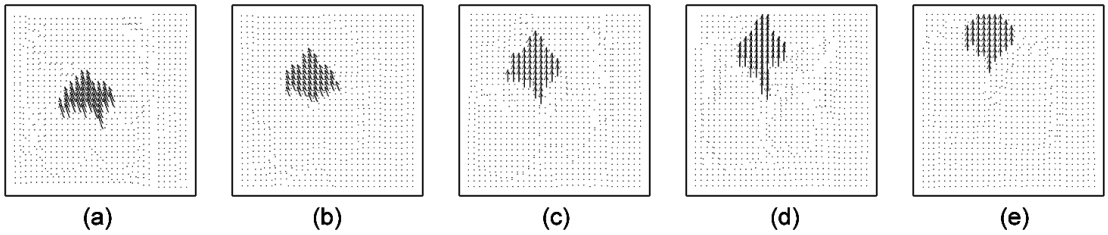


Figura 4.26: Análisis de la secuencia # 4 (enlazado predictivo). Vectores de desplazamiento estimados entre los fotogramas: a) 1 y 2; b) 2 y 3; c) 3 y 4; d) 4 y 5; e) 5 y 6.

7 Resultados

Para comprobar su funcionamiento, el método de segmentación espacio-temporal propuesto ha sido aplicado sobre varias secuencias de vídeo reales con distintos niveles de complejidad. En cada apartado se muestran tanto los resultados de las segmentación como los vectores de movimiento estimados. Cabe mencionar que, para resaltar las clases que aparecen como resultado del proceso de segmentación y posterior fusión, las pirámides segmentadas presentan el nivel de gris correspondiente a la clase a la que pertenecen en lugar del nivel de gris original. Los fotogramas de las secuencias analizadas en este apartado se han reproducido en el Apéndice B. A continuación, se presentan los resultados obtenidos con estas secuencias en orden ascendente de complejidad y se comentan aspectos interesantes acerca del comportamiento del sistema.

7.1 Captura con cámara cenital estática (Secuencia #1 Apéndice B)

Como primer ejemplo se ha elegido un caso sencillo en el cual se ha dispuesto una cámara fija en el techo de un pasillo que captura a los transeúntes que lo cruzan. El sistema está sujeto a mínimos cambios de iluminación, ya que ésta es artificial y permanece encendida durante la captura de la secuencia. Además, apenas existen efectos de perspectiva, y los móviles, a pesar de ser deformables, mantienen aproximadamente su tamaño durante su paso por el pasillo. La Fig. 4.27 muestra los resultados de la segmentación espacio-temporal de la secuencia y la Fig. 4.28 presenta los vectores de desplazamientos estimados asociados.

Como ya se ha comentado previamente, la ausencia de una estimación inicial adecuada para el primer fotograma, induce la aparición de defectos en la asignación de enlaces en los inicios de la secuencia. Sin embargo, debido a la capacidad de adaptación del método, rápidamente se corrige este efecto, y en el tercer fotograma ya se dispone de una estimación correcta. Lo más notable en esta secuencia es el hecho de que el desplazamiento del móvil se estima adecuadamente a pesar de los cambios de forma que presenta, así como que, aunque el suelo se divide en varias clases por diferencias de gris entre zonas iluminadas con diferente intensidad, las clases se identifican sin problemas de un fotograma a otro. De no ser así aparecerían vectores de desplazamiento erróneos en los píxeles correspondientes al suelo del corredor por el cambio de posición de los centroides debido a una variación brusca de tamaño.

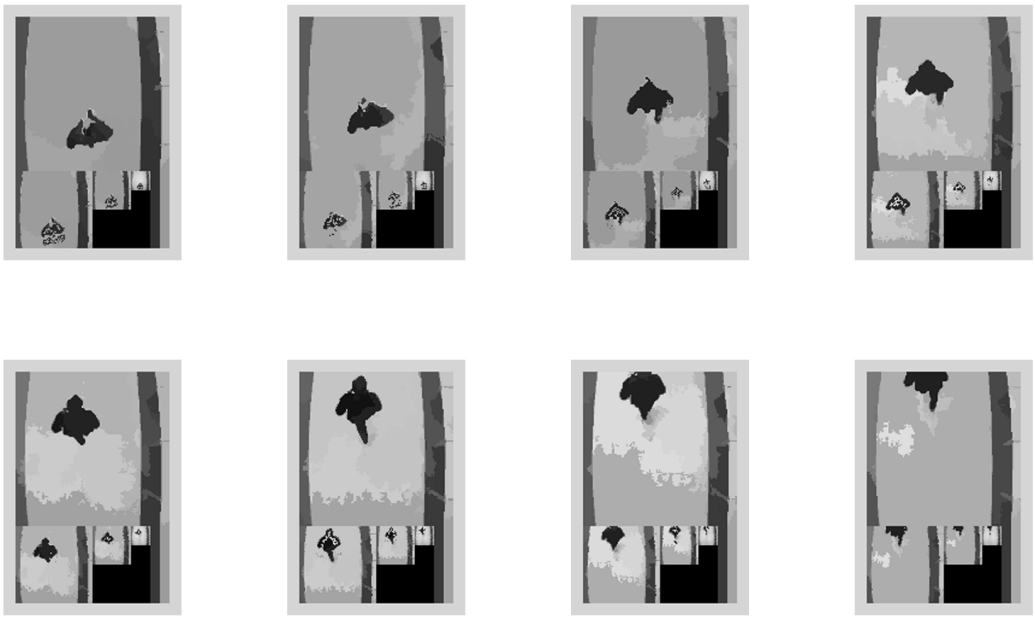


Figura 4.27: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B1: Pirámides segmentadas.

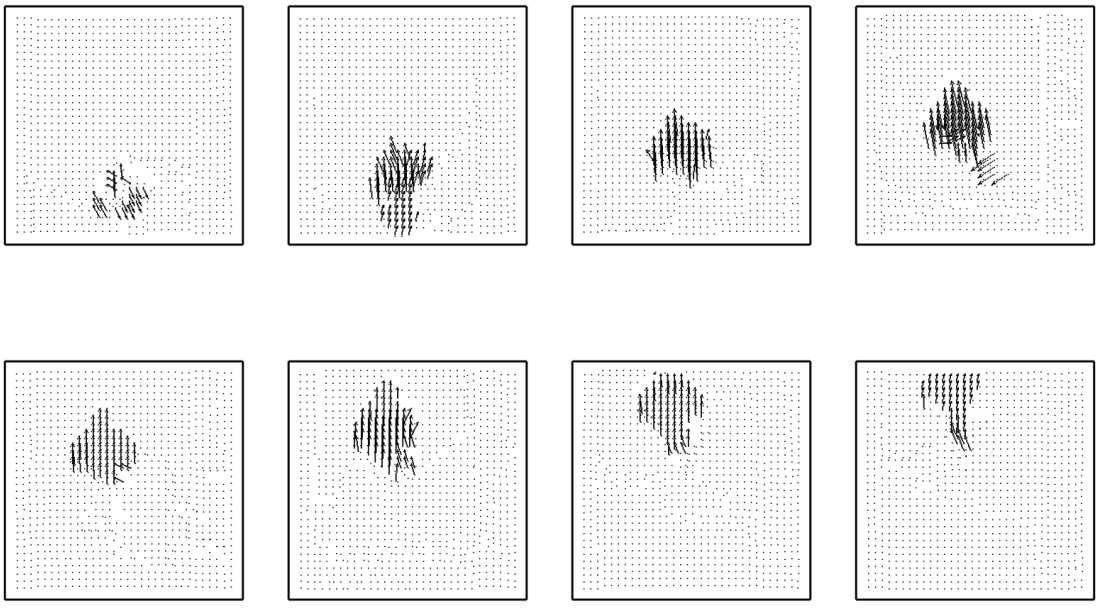


Figura 4.28: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B1: Vectores de desplazamiento estimados.

7.2 Supervisión de tráfico con cámara estática (Secuencia #2 Apéndice B)

En este experimento se capturó una secuencia de tráfico desde un puente elevado, usando una cámara compacta de vídeo soportada por un trípode. En este caso la escena es mucho más compleja que la analizada en el subapartado anterior y, además, está sometida a los cambios típicos de las condiciones de captura que se dan en exteriores, como la presencia de cambios de iluminación y la aparición de texturas complejas en los objetos. Adicionalmente, es necesario notar que la escena presenta una perspectiva importante con fuga hacia el fondo. Dicha perspectiva provoca un cambio de masa considerable en los automóviles conforme se aproximan entre fotogramas consecutivos, así como un incremento de la velocidad que presentan debido a los efectos de la proyección.

En secuencias reales complejas el método de segmentación propuesto resulta particularmente robusto, ya que no sólo mantiene una considerable estabilidad entre los resultados de segmentación en fotogramas consecutivos, sino que implícitamente relaciona las regiones resultantes a lo largo de la secuencia. Para probar este hecho, puede observarse en la Fig. 4.29 los resultados de la segmentación jerárquica adaptativa clásica aplicada sobre varios fotogramas consecutivos de la secuencia cuando las pirámides no se combinan temporalmente. En este caso se atiende únicamente a criterios espaciales, por lo que las clases no se agrupan manteniendo la estructura de la escena en el tiempo, sino que minimizan el error de agrupamiento de cada fotograma por separado.

Tal y como se comentó en el capítulo anterior, la segmentación exclusivamente espacial de imágenes reales presenta muchos problemas, ya que mínimos cambios en las condiciones de captura pueden provocar grandes cambios en los resultados. Así, puede observarse cómo en el segundo fotograma de la secuencia, debido a un cambio leve de iluminación y a las oclusiones y descubrimientos provocados por la camioneta en su desplazamiento, la región que corresponde a la carretera cambia su estructura completamente, ampliándose para incluir una clase previa que se debía a cambios de luz sobre dicha carretera. Este efecto continúa ocurriendo a lo largo de toda la secuencia, dificultando enormemente el proceso de establecer una relación entre las distintas áreas de píxeles que van constituyendo la carretera de un fotograma a otro. De forma similar, puede observarse que el arcén de la derecha presenta una división en clases casi aleatoria para adecuarse a las condiciones del resto de la escena.

La Fig. 4.30 presenta parejas de fotogramas consecutivos segmentados utilizando el método propuesto. En este caso ya no es necesario un proceso de comparación de regiones para asociarlas dos a dos, ya que están intrínsecamente relacionadas entre sí mediante la estructura

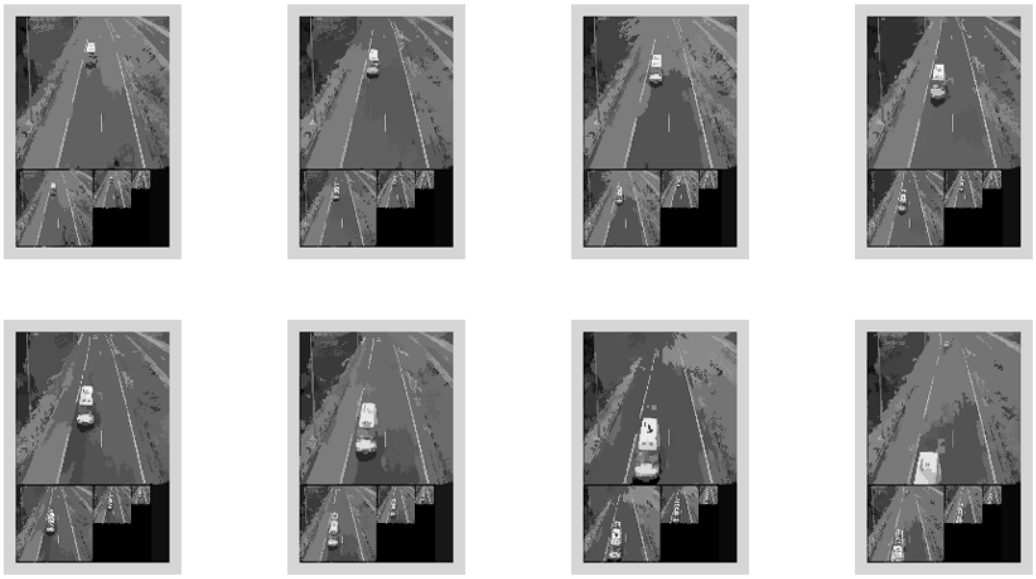


Figura 4.29: Resultados de la segmentación jerárquica espacial adaptativa de la secuencia B2: Pirámides segmentadas.

de enlaces definida. Es importante destacar que la segunda pirámide de cada juego se construye sobre el mismo fotograma de la primera del juego siguiente. Los resultados de la segmentación sobre un mismo fotograma pueden cambiar en función del fotograma con el que se está estabilizando, pudiendo ser el anterior o el posterior. Esto no supone ningún problema, ya que la estructura de enlaces previamente definida permite relacionar las nuevas clases con las anteriores y tener localizada cualquier clase que represente un móvil en la escena a través de la estructura de enlaces establecida sobre la porción completa de secuencia en la que aparece.

Lo más interesante de este segundo caso es cómo la estructura de clases, que se perdía en la segmentación espacial para cada nuevo fotograma, se mantiene entre fotogramas consecutivos enlazados combinadamente cuando se emplean criterios espacio-temporales. De este modo, la carretera es una única entidad para cada pareja de fotogramas y, a partir de ahí, es posible estimar su vector de desplazamiento con bastante precisión. Quizá los fotogramas en que más se aprecia el efecto mencionado son aquellos en que la camioneta está próxima a salir del campo de visión. En este caso su desplazamiento es máximo y la porción de imagen que ocupa también, por lo que las oclusiones y descubrimientos están más marcados que nunca. Al contrario de cuando se usaba la segmentación exclusivamente espacial, puede observarse que las clases que representan la carretera en las cuatro últimas pirámides combinadas sigue presentando la misma forma.

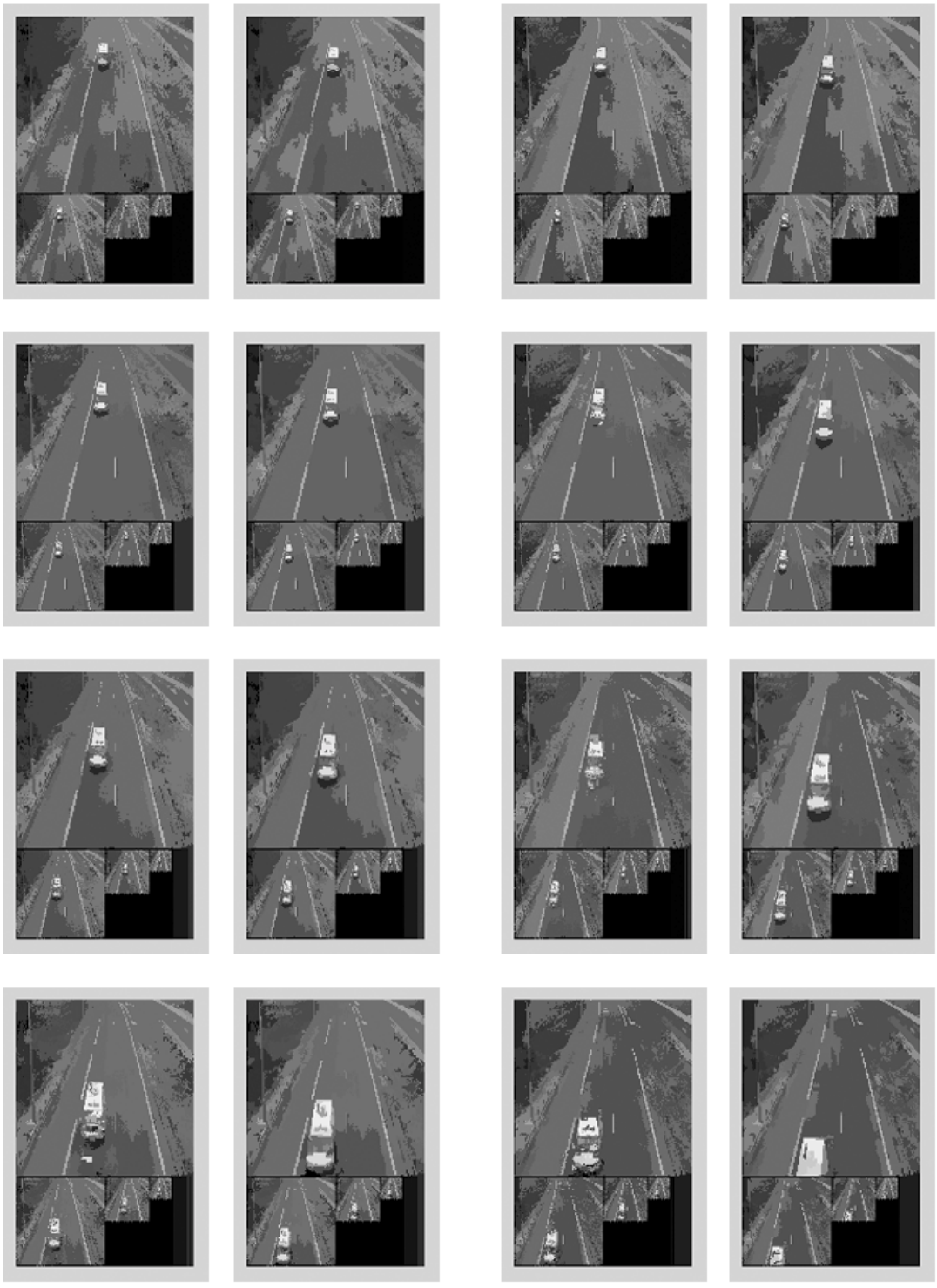


Figura 4.30: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B2: Pirámides segmentadas.

Para ratificar el buen funcionamiento del método, es interesante observar la estimación de desplazamiento de las regiones resultantes efectuada a lo largo del proceso de segmentación (Fig. 4.31). Puede apreciarse que el fondo completo aparece con desplazamiento cero, mientras que para la camioneta se estima correctamente un vector de desplazamiento que depende de su velocidad relativa respecto a la cámara. Es de señalar el hecho de que en las antepenúltima y penúltima imágenes de la secuencia aparece una región errónea desplazándose a una velocidad similar al vehículo que deteriora los resultados. Este efecto se debe al hecho de que se están usando criterios no sólo temporales, sino espacio-temporales, y que, cuando el vehículo atraviesa una zona que presenta el mismo color que su carrocería, siempre está sujeto al riesgo de fusionarse con ella y arrastrarla por la inercia de su propio movimiento. No obstante, es necesario notar que el efecto se corrige por sí solo y no arruina los resultados de segmentación posteriores.

7.3 Panorámica con perspectiva (Secuencia #3 Apéndice B)

En este caso, se capturó con una cámara analógica de vídeo una secuencia que muestra cómo un transeúnte cruza la escena diagonalmente partiendo del extremo más próximo al objetivo. El sistema estuvo sujeto en todo momento a las condiciones típicas de los procesos de filmación en exteriores, que en este caso afectaron fundamentalmente a la aparición de sombras provocadas

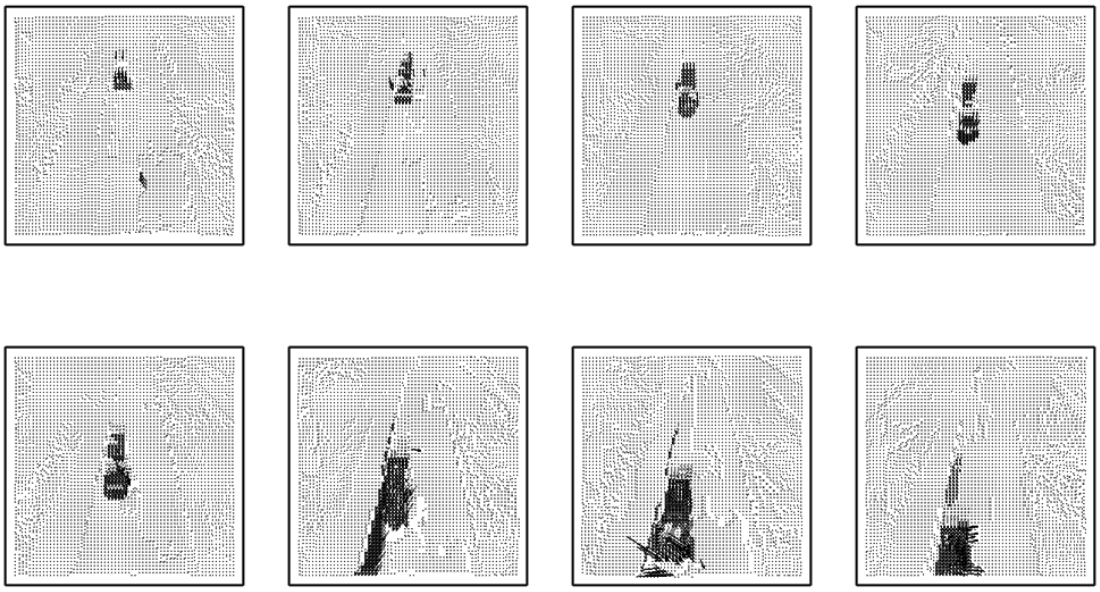


Figura 4.31: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B2: Vectores de desplazamiento estimados.

por la luz natural. La principal diferencia respecto a la secuencia anterior radica en la velocidad de los móviles, en este caso inferior. De este modo se demuestra que el sistema es robusto para un amplio rango de variaciones temporales.

Las Figs. 4.32 y 4.33 muestran la segmentación espacio-temporal de la secuencia y los vectores de desplazamiento estimados para dicha secuencia. Puede observarse como, tras la inestabilidad inicial debida a la ausencia de una estimación del desplazamiento de las regiones de la escena, el sistema comienza a estabilizarse y a generar clases equivalentes entre fotogramas a pesar del movimiento del transeúnte y del cambio de tamaño que experimenta como consecuencia de la perspectiva.

Es necesario señalar que en esta secuencia en particular aparecen una serie de vectores de desplazamiento incorrectos de baja magnitud en áreas del suelo durante la primera mitad de la secuencia. Este efecto se debe al hecho de que la sombra proyectada por el transeúnte divide literalmente dicho suelo en dos mitades, lo que impide que se integre correctamente en una única clase. Esto provoca la generación de clases de formas pseudoaleatorias, ya que su configuración depende de la posición relativa de la sombra en el fotograma siguiente. Es inmediatamente apreciable que en el momento en que la sombra deja de cortar la región de suelo estos efectos dejan de ocurrir, aunque, como es natural, al usar un método espacial basado en nivel de gris, la sombra se presenta como una región que se desplaza a lo largo de toda la secuencia. Cabe mencionar además que el método propuesto no distingue entre sombras y objetos, lo cual, a priori, no supone ninguna limitación a su utilización como método de detección y estimación de movimiento, ya que posteriores algoritmos de alto nivel pueden interpretar las distintas regiones detectadas en la escena eliminando artefactos de este tipo.

7.4 Aplicaciones de videoconferencia (Secuencia #4 Apéndice B)

En las secuencias previas se ha podido observar cómo se comporta el sistema cuando aparece un móvil sobre un fondo fijo, pero no se ha observado qué ocurre cuando dicho móvil ocupa la mayor parte de la escena. Ésta situación es la habitual en secuencias de videoconferencia, donde el interlocutor enfocado por la cámara oculta un gran porcentaje del fondo. En este caso, los movimientos de dicho interlocutor, en particular si son bruscos, inducen oclusiones y descubrimientos importantes. Además, es necesario considerar que en estos casos los efectos de luz y sombra son particularmente notorios en los pliegues de la ropa, cabello o texturas de la piel. La Fig. 4.34 presenta los resultados de segmentar la secuencia #4 del Apéndice B, mientras que la estimación de movimiento correspondiente a dicha secuencia aparece en la Fig. 4.35.

Figura 4.32: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B3: Pirámides segmentadas.

Figura 4.33: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B3: Vectores de desplazamiento estimados.

Figura 4.34: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B4: Pirámides segmentadas.

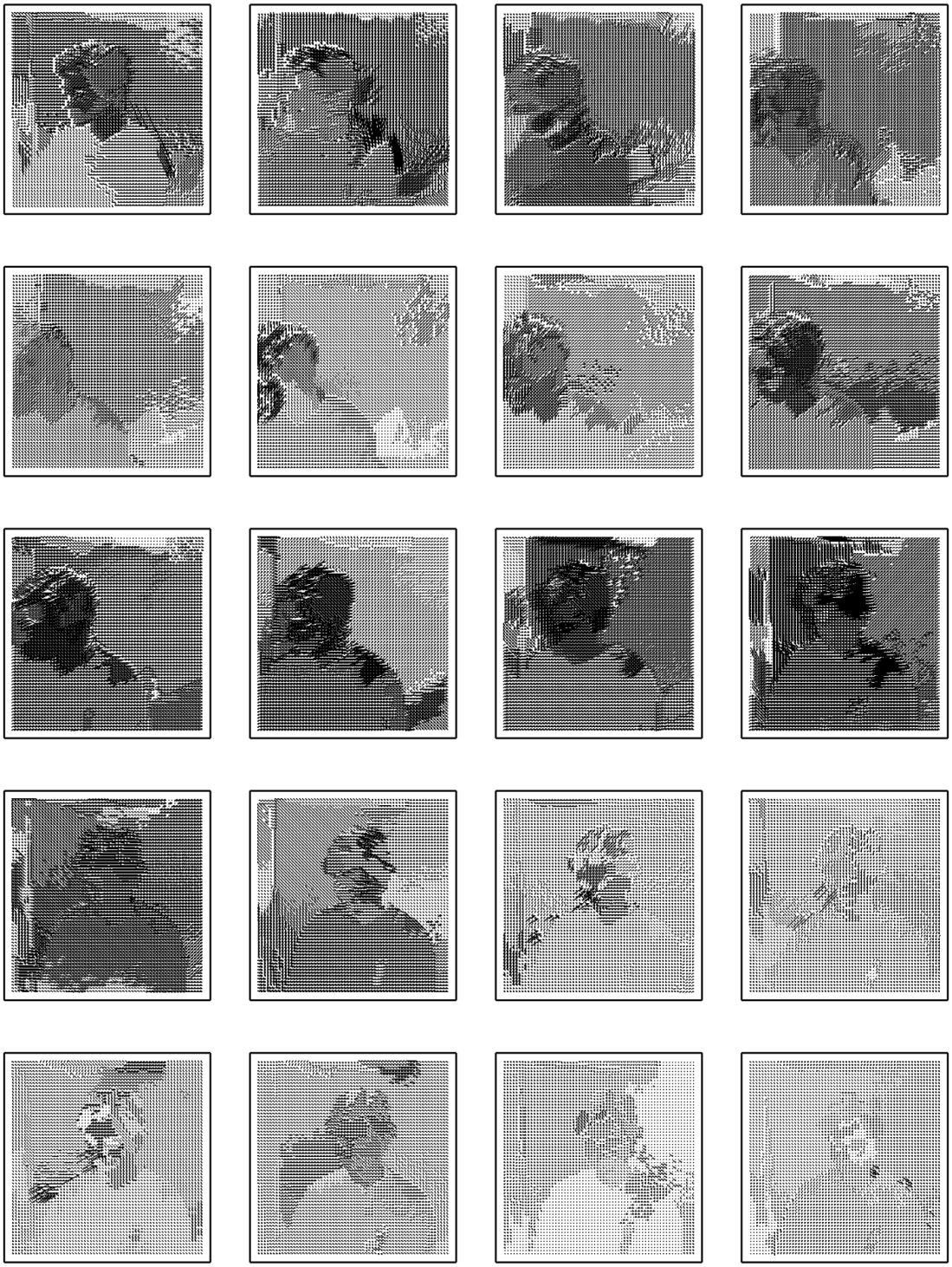


Figura 4.35: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B4: Vectores de desplazamiento estimados.

Puede observarse como la interlocutora efectúa un movimiento brusco al principio de ésta para recoger algo situado fuera de escena a la izquierda y posteriormente recupera su posición original, efectuando a continuación movimientos más suaves. Este efecto se ve reflejado en los vectores de desplazamiento obtenidos, donde sólo se percibe movimiento para los fotogramas en que la interlocutora se inclina y se incorpora. Los resultados mostrados corresponden a fotogramas posteriores a los necesarios para la inicialización del proceso, una vez que se dispone de pirámides correctamente estabilizadas, y reproducen el movimiento de forma adecuada. Por ejemplo, la primera imagen segmentada corresponde a un pequeño giro del rostro con el consecuente cambio de iluminación; los dos fotogramas a la derecha de la tercera fila también corresponden a un gesto similar, sólo que en ese caso el giro es más brusco y hacia la izquierda, tal como muestran los vectores de desplazamiento.

En esta secuencia cabe destacar algunos efectos interesantes, como el hecho de que el bastón de la lámpara situada a la izquierda de la escena se segmente como una única clase y se mantenga como una entidad, a pesar de las oclusiones ocasionadas por la interlocutora. De no ser así, presentaría vectores de desplazamiento importantes por el efecto de traslación del centroide cuando esta clase perdiera masa bruscamente. En el tercer fotograma de la figura, en que la clase se reduce bruscamente a la mitad, puede observarse la aparición de este efecto, pero en sucesivos fotogramas se corrige paulatinamente porque el método consigue compensarlo adaptándose a la nueva situación. Algo similar ocurre con el sillón sobre el que está sentada la interlocutora, que aparece a la derecha de la escena cuando ésta se inclina, o en el logotipo situado a la derecha del jersey, que no aparece hasta la mitad de la secuencia.

La segmentación es, como puede observarse, similar en todos los casos, independientemente de la posición que ocupe la interlocutora. Sin embargo, aparecen nuevas clases como consecuencia de los efectos de luz y sombra sobre determinadas áreas de la escena, aunque son coherentes en tanto en cuanto se mantienen las características de iluminación durante un par de fotogramas consecutivos, tal y como se aprecia en el rostro de la interlocutora en función de su posición relativa respecto al foco de luz. Debe notarse que estas áreas presentan, como era de esperar, unos vectores de desplazamiento muy similares, lo que permitiría agruparlas en una sola clase si no se tuviese en cuenta sus características dinámicas como criterio de segmentación.

Cabe mencionar, finalmente, que este tipo de aplicaciones en las que el objeto de interés ocupa gran parte de la escena requieren, más que una estimación del flujo de las regiones que la componen, una determinación eficiente de las diferentes clases que aparecen, aunque de esta forma dicho objeto esté representado por más de una clase. Esto se debe a que las aplicaciones orientadas a este tipo de escenas están más orientadas a la codificación de imágenes con fines

de compresión, que a la identificación y seguimiento de objetos, como es el caso de los sistemas de conteo, supervisión o vigilancia. De esta forma, aunque aparezcan efectos espúrios en la obtención de los vectores de movimiento de las regiones de la escena, el método sigue aportando una segmentación en clases más coherente que el resto de las técnicas descritas en capítulos anteriores.

7.5 Movimientos rotatorios sobre fondo estático (Secuencia #5 Apéndice B)

En este experimento se ha analizado una secuencia clásica en visión por computador, consistente en un cubo de Rubik apoyado sobre una plataforma que gira lentamente. Los resultados de la segmentación se muestran en la Fig. 4.36 y los vectores de desplazamiento estimados en la Fig. 4.37. Es necesario indicar que el movimiento que presenta el cubo es tan lento que dichos desplazamientos son apenas perceptibles, salvo para algunas regiones de la escena, ya que la unidad mínima de desplazamiento es el píxel. En este caso, la única forma de evaluar los resultados de la aplicación del método a la secuencia consiste en observar la conservación de la forma de las clases. Esto permitirá conocer si la segmentación es estable en este caso.

Se puede apreciar, por ejemplo, la gran estabilidad de las regiones que pertenecen al fondo, así como la conservación de tonos de grises entre fotogramas consecutivos. Es de notar que la zona en la que mejor se aprecian los desplazamientos es, lógicamente, el borde de la plataforma rotatoria, no sólo porque presenta mayor velocidad que el resto de la escena, sino porque su diseño distintivo presenta mayor información que otras zonas como, por ejemplo, la superficie clara de la base giratoria. Como comentario final, resulta interesante constatar el hecho de que cada uno de los cuadros de cada cara del cubo constituye una clase distinta, que puede rastrearse correctamente a lo largo de la secuencia siguiendo la estructura de enlaces definida tras la estabilización conjunta. Como este efecto es imposible de apreciar en la segmentación final, ya que cada cara posee nueve clases del mismo color, en la Fig. 4.38 se muestra en detalle la evolución de una clase a lo largo de tres fotogramas de la secuencia.

Cabe mencionar que en este caso la aplicación de métodos diferenciales es posible debido a que cumple las restricciones impuestos por estos. Sin embargo, estos métodos aportan como resultado de su aplicación exclusivamente un mapa de vectores de desplazamiento que posteriormente habría que agrupar para determinar las distintas regiones o clases que aparecen en la escena. Así, cabe destacar que el método propuesto no sólo es aplicable en aquellos casos en que dichas restricciones no se cumplen y, por tanto, las técnicas existentes fallan, sino que también aportan una solución a las situaciones en que dichas técnicas tienen aplicación.

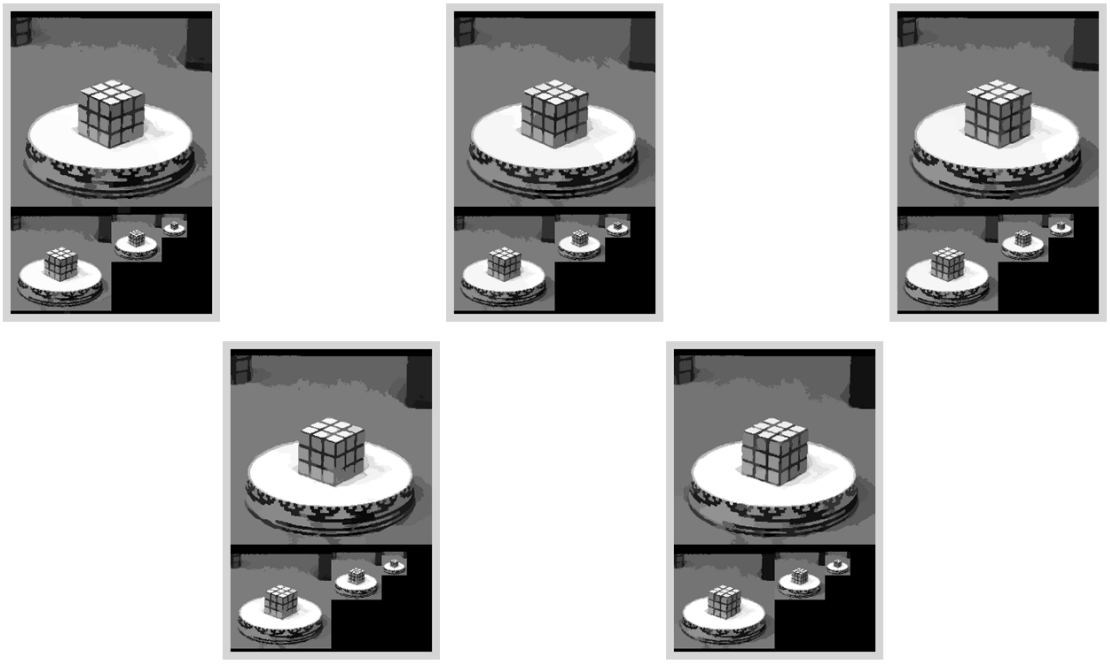


Figura 4.36: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B5: Pirámides segmentadas.

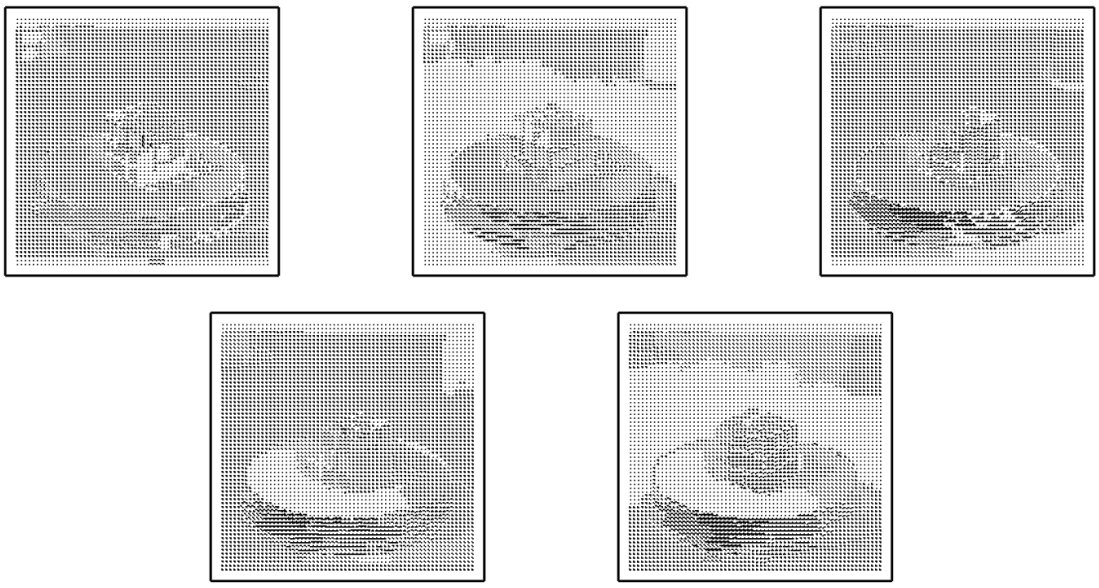


Figura 4.37: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B5: Vectores de desplazamiento estimados.

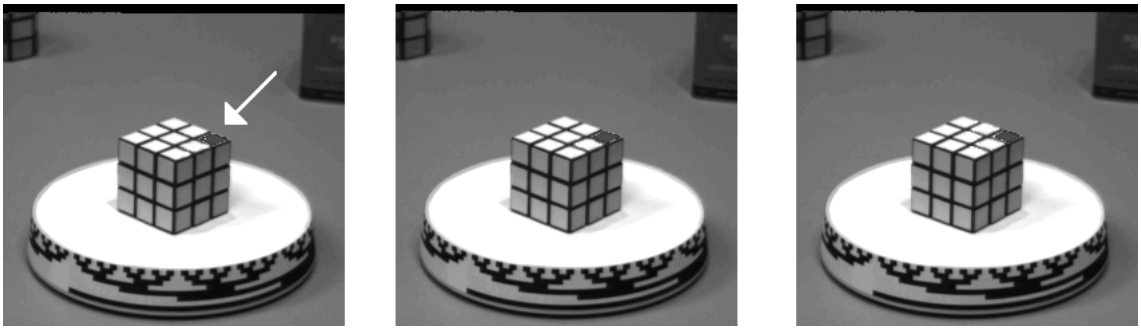


Figura 4.38: Seguimiento de una clase de la escena a lo largo de tres fotogramas de la secuencia B5.

7.6 Desplazamiento de la cámara (Secuencia #6 Apéndice B)

Este ejemplo supone un caso típico en las aplicaciones de visión artificial, en la que aumenta el grado de complejidad debido a que la cámara también puede presentar movimiento. De esta forma, a pesar de trabajar sobre un fondo estático, dicho fondo presenta un movimiento lento uniforme de desplazamiento con respecto a la cámara que es, a fin de cuentas, su sistema de referencia. La secuencia fué capturada en un entorno cerrado de laboratorio y presenta, en primer plano, parte de un monitor de PC y, como fondo, una estantería con distintos objetos. Puede apreciarse que dicho monitor nunca entra completamente en el campo de visión, sino que ocupa porciones cambiantes del mismo conforme cambia el plano de la escena. La Fig. 4.39 muestra el resultado de la segmentación espacio-temporal de la secuencia, mostrándose los vectores de desplazamiento asociados a ésta en la Fig. 4.40.

Al igual que en casos anteriores, en ocasiones es difícil observar los vectores de desplazamiento cuando se presentan en un formato tan denso como el de la Fig. 4.40. En la Fig. 4.41 se han reproducido los vectores de desplazamiento que la secuencia presenta entre los fotogramas 4 y 7 con un mayor nivel de detalle. Puede apreciarse que dichos vectores están orientados a la izquierda, como cabía esperar, ya que el movimiento de la cámara es rotatorio dextrógiro. No obstante, cuando el movimiento de la cámara provoca la aparición de nuevos objetos, que no se pueden fusionar con clases ya existentes, aparecen pequeños flujos aleatorios, debido a que no existe referencia para dichos objetos en fotogramas previos de la secuencia. Este efecto podría corregirse si se trabajase usando fotogramas posteriores además de los anteriores, pero esta medida ralentizaría el sistema y, en cualquier caso, violaría el principio de causalidad cuando se trabaja en tiempo real. Además, si se conoce el movimiento de la cámara se puede predecir este efecto asumiendo la potencial aparición de nuevas clases, de forma que no se tengan en cuenta a la hora de obtener los valores de desplazamiento de la escena.

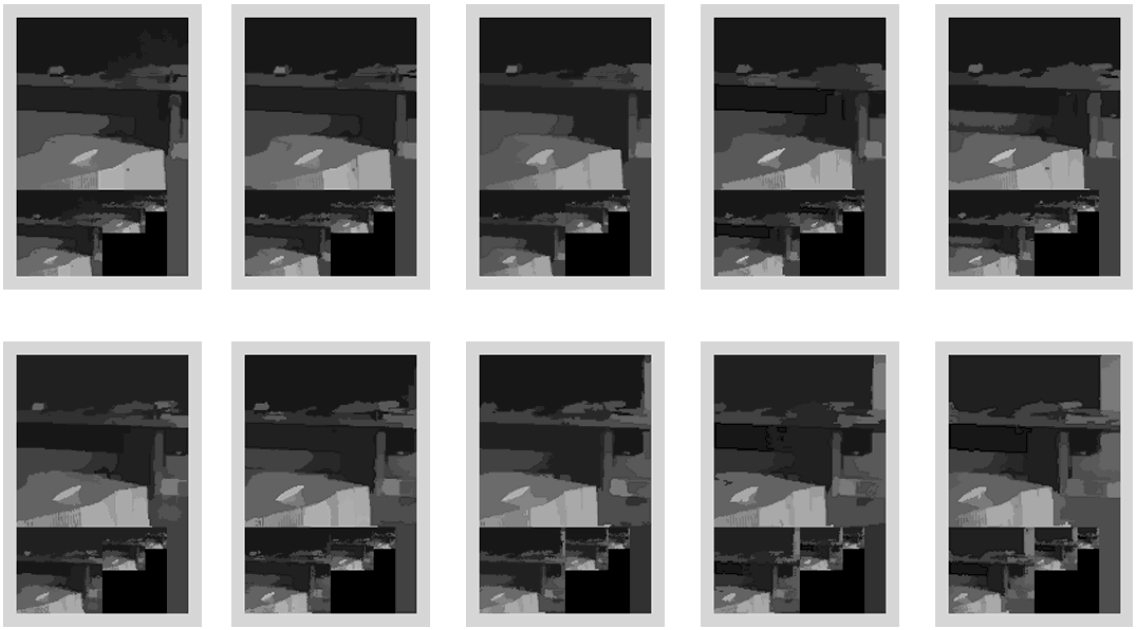


Figura 4.39: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B6: Pirámides segmentadas.

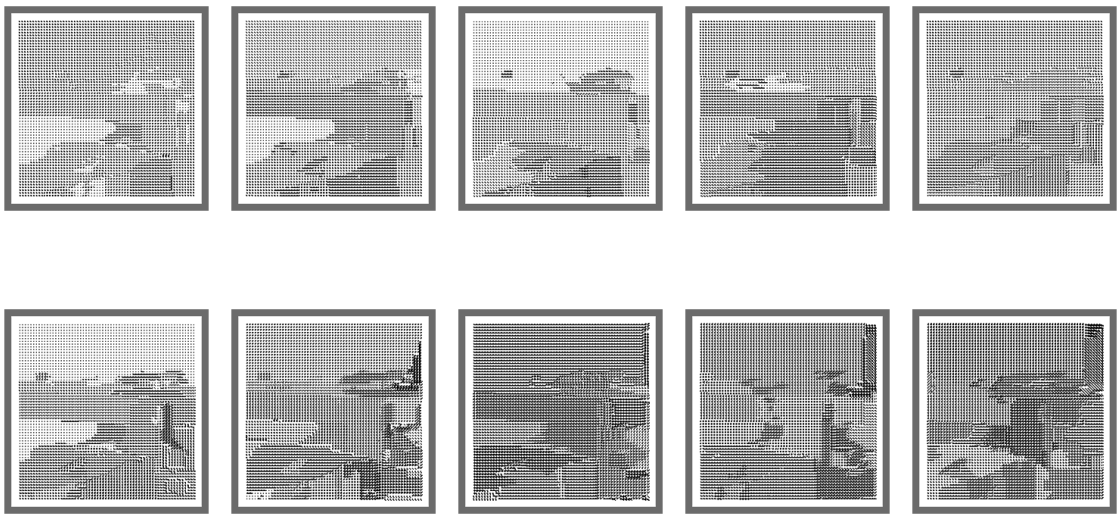


Figura 4.40: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B6: Vectores de desplazamiento estimados.

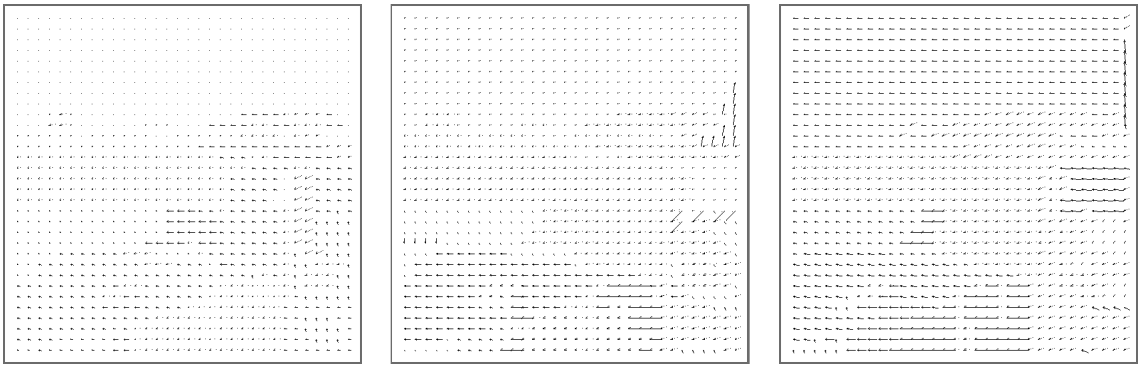


Figura 4.41: Vectores de desplazamiento estimados para la secuencia B6.

Es necesario indicar que, aunque en la escena no aparezca ningún objeto móvil, la presencia de un campo de vectores de movimiento no uniforme refleja el hecho de que los objetos que componen este escenario se encuentran distribuidos en el entorno a diferentes distancias de la cámara. Por ello, aunque la velocidad de giro angular de la cámara es aproximadamente constante, cada objeto presenta diferentes velocidades lineales una vez proyectados en el plano de la imagen los puntos que lo representan. Esto se traduce en que la estimación de movimiento distingue entre objetos en función de su distancia a la cámara. Mediante un método de calibración apropiado se podrían deducir dichas distancias y crear un mapa de profundidad de la escena. Cabe mencionar que las técnicas diferenciales también serían aplicables a este caso, segmentando *a posteriori* el campo de vectores resultante. Sin embargo, conforme la cámara se aleja de la escena, la perspectiva induce cada vez menos a la aparición de este fenómeno. Como consecuencia, cuando la distancia del objeto más cercano es muy superior a la distancia focal de la cámara, se puede asumir un modelo plano de proyección, que da lugar a un movimiento homogéneo de toda la escena. En este caso, los vectores de desplazamiento no permitirían discernir entre objetos distintos, mientras que el método propuesto de enlazado adaptativo combinado, debido a su naturaleza mixta espacio-temporal, aún aportaría una segmentación espacial atendiendo al nivel de gris de los distintos objetos presentes en la escena.

7.7 Seguimiento de objetos en movimiento (Secuencia #7 Apéndice B)

En este experimento, en lugar de girar la cámara sobre una escena que no presenta objetos en movimiento, se utilizó una secuencia en que la cámara sigue a un individuo que se desplaza. Ahora aparecen dos movimientos distintos: el del fondo de la escena y el del móvil en sí. En este caso se ha empleado una secuencia que no presenta texturas ni cambios de iluminación, lo que simplifica ligeramente las condiciones de trabajo.

La Fig. 4.42 presenta la segmentación espacio-temporal de la secuencia y la Fig. 4.43 los vectores de desplazamiento estimados. El hecho de que toda la imagen se desplace, dificulta la extracción visual de información significativa a partir de los vectores de desplazamiento, debido a la alta densidad que presentan esta última figura. No obstante, si se realiza un submuestreo de la misma, representando sólo algunos vectores de desplazamiento sobre los fotogramas, es mucho más sencillo discernir lo que está ocurriendo en ella. Así, por ejemplo, en la Fig. 4.44 se presentan en detalle los vectores de desplazamiento correspondientes a los fotogramas 2 y 3, y 3 y 4, de la secuencia segmentada en la Fig. 4.42. Puede observarse como el grueso del fondo se traslada hacia la izquierda obedeciendo al movimiento de desplazamiento hacia la derecha de la cámara. La fachada en primer plano se desplaza más rápidamente que el callejón del fondo debido a la perspectiva, tal y como puede observarse en la secuencia original (secuencia #7 del Apéndice B). Del fotograma 3 al 4, el movimiento de la cámara se traduce en la completa desaparición de uno de los cubos de basura. Este hecho queda convenientemente reflejado en la estimación de desplazamiento, que crece en dicha zona. El individuo en primer plano avanza entre los fotogramas 2 y 3, pero se detiene entre el 3 y el 4 salvo por el cambio en la posición de las piernas. Su movimiento también puede apreciarse en la figura en que se presentan los vectores estimados de desplazamiento donde, incluso, puede distinguirse entre el movimiento pendular del brazo y el avance del torso, frente a los cambios en las piernas. Nótese que en los fotogramas 3 y 4 los brazos no cambian de posición, quedando reflejado como una región de movimiento cero, al igual que ocurre con el transeúnte, debido a que su movimiento relativo frente a la cámara siempre es nulo o, en cualquier caso, menor que el del fondo.

Para observar qué ocurre ante cambios de velocidad de la cámara y, concretamente, cuando ésta se detiene, se presentan también en la Fig. 4.45 los vectores de desplazamiento estimados para los fotogramas 10 y 11, y 11 y 12, en los que el individuo procede a entrar en el edificio. Cuando éste se detiene para girar hacia el interior, el desplazamiento que presenta es aún pequeño, pero es detectado correctamente hacia el interior del edificio. El resto de la escena presenta un valor prácticamente nulo de desplazamiento, como corresponde al no existir móviles con cámara estática. En el siguiente fotograma el individuo está procediendo decididamente a entrar en el edificio, lo que se traduce en un conjunto de vectores de mayor módulo para las clases que lo representan.

Con esta secuencia se ha mostrado como el método propuesto puede manejar correctamente no sólo las situaciones en que los objetos se mueven frente a un fondo estático, sino también aquellas en que aparecen móviles sobre un fondo, también móvil.

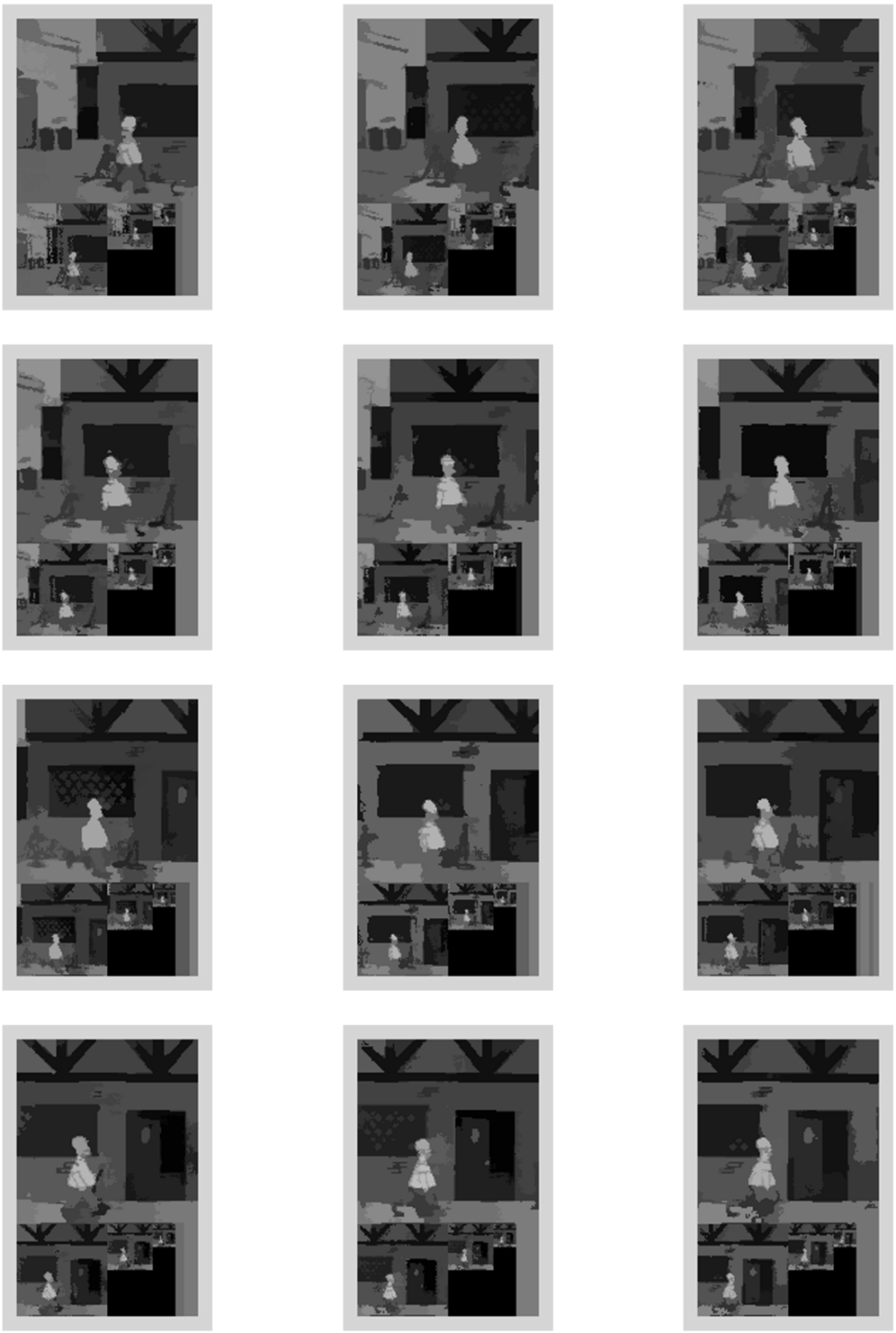


Figura 4.42: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B7: Pirámides segmentadas.

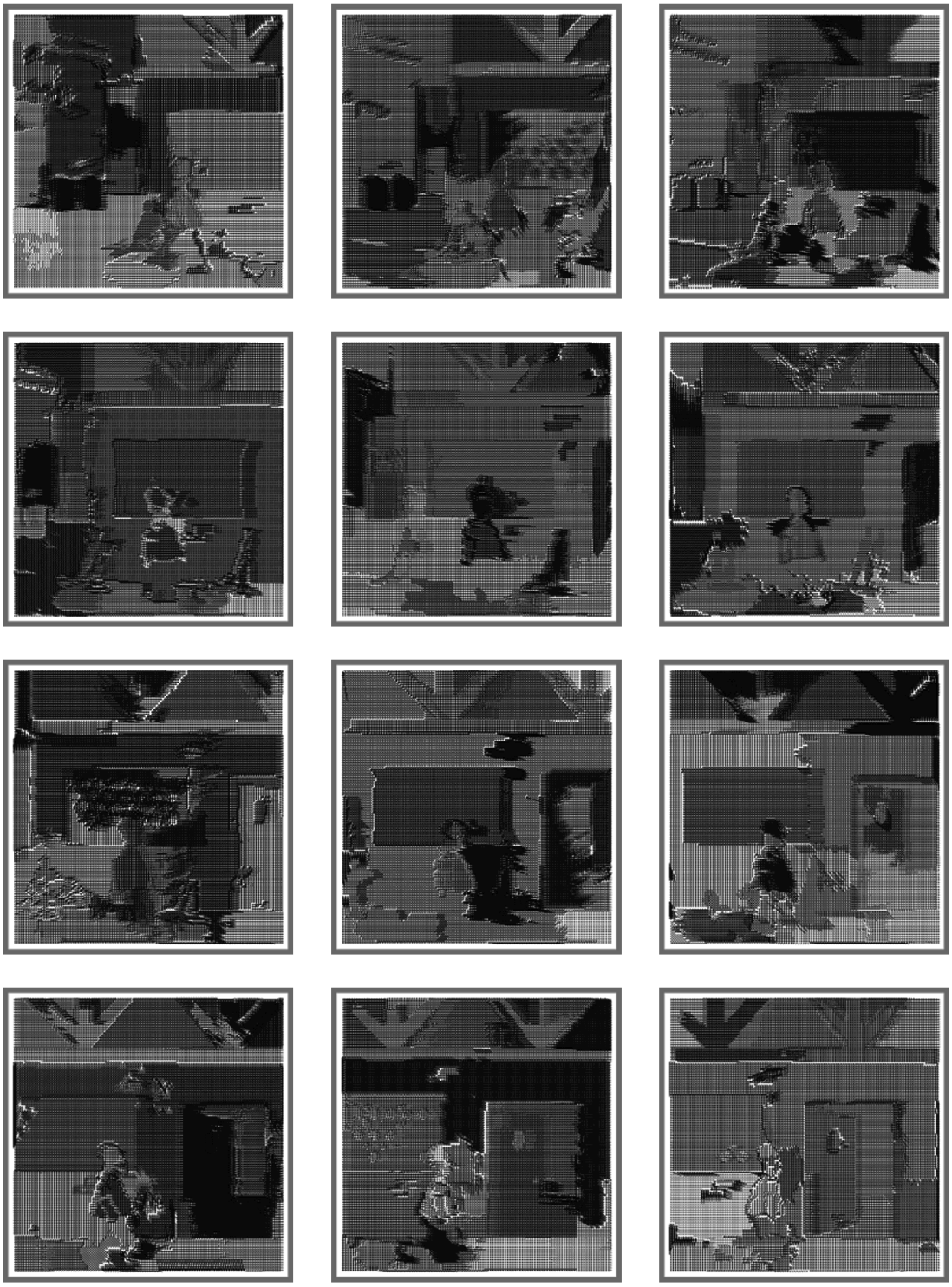


Figura 4.43: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B7: Vectores de desplazamiento estimados.

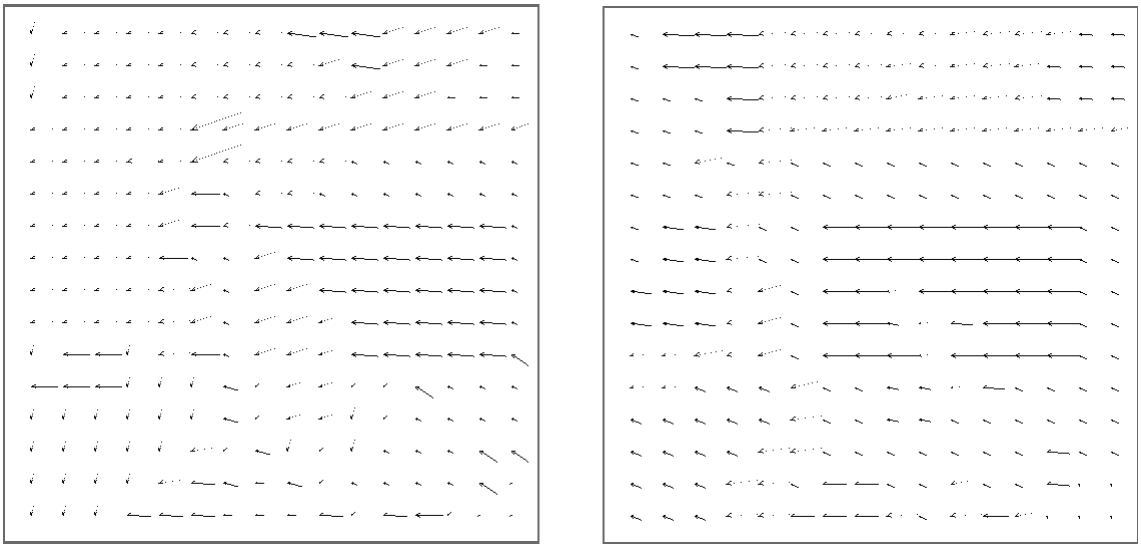


Figura 4.44: Detalle de los vectores de desplazamiento estimados para la secuencia B7 entre los fotogramas 2 y 3, y los fotogramas 3 y 4

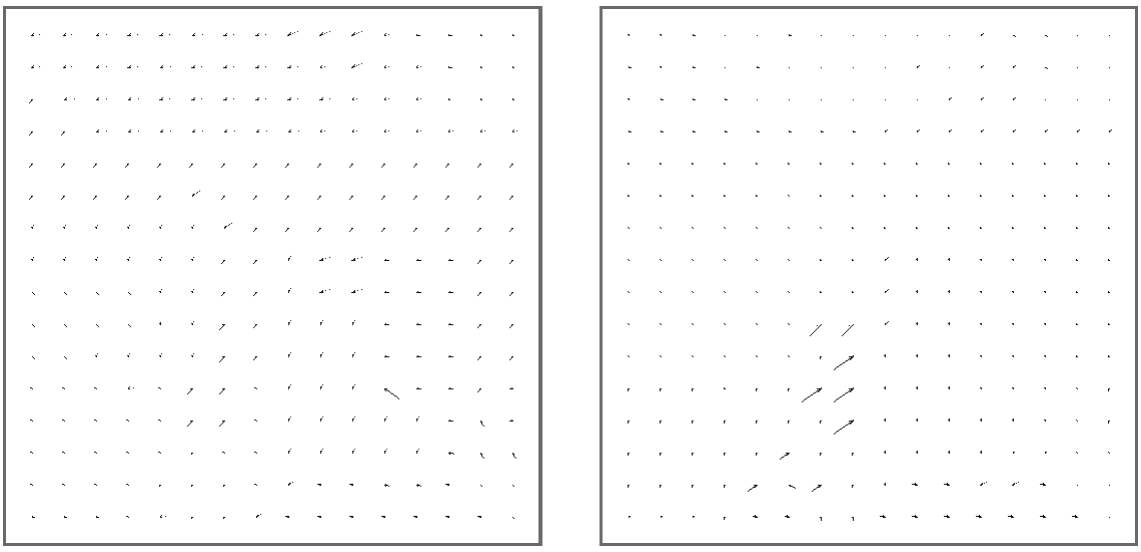


Figura 4.45: Detalle de los vectores de desplazamiento estimados para la secuencia B7 entre los fotogramas 10 y 11, y los fotogramas 11 y 12

7.8 Desplazamientos sobre escenas dinámicas (Secuencia #8 Apéndice B)

El caso más complicado de segmentación espacio-temporal es aquel en el que la cámara se desplaza sobre un fondo complejo en el que existen objetos en movimiento. Para comprobar el funcionamiento del método en las circunstancias mencionadas, se capturó una nueva secuencia de tráfico que presentaba estas características, añadiendo un grado de dificultad con la presencia de texturas complejas y sombras al caso contemplado en la secuencia anterior. Los resultados de la segmentación y los vectores de desplazamiento se muestran en las Figs. 4.46 y 4.47 respectivamente. En este caso es aún más complicado extraer información de ambas figuras debido a la enorme complejidad de la secuencia. Por ello se han dispuesto en la Fig. 4.48 el detalle de los vectores de desplazamiento estimados entre los cinco primeros fotogramas.

La principal conclusión que puede extraerse de la Fig. 4.48 es que toda la escena se desplaza hacia la izquierda, consecuente con el movimiento de la cámara, que es hacia la derecha. Este resultado, aunque parece una obviedad, es en sí un éxito dada la complejidad de la escena en estudio, que se traduce en fuertes variaciones de los niveles de gris de las regiones presentes en fotogramas consecutivos debidas a la aparición de sombras y cambios de perspectiva. Además, la continua existencia de oclusiones y descubrimientos, no sólo debidos a los cambios de posición de los móviles, sino también al propio movimiento de la cámara, puede forzar la reagrupación

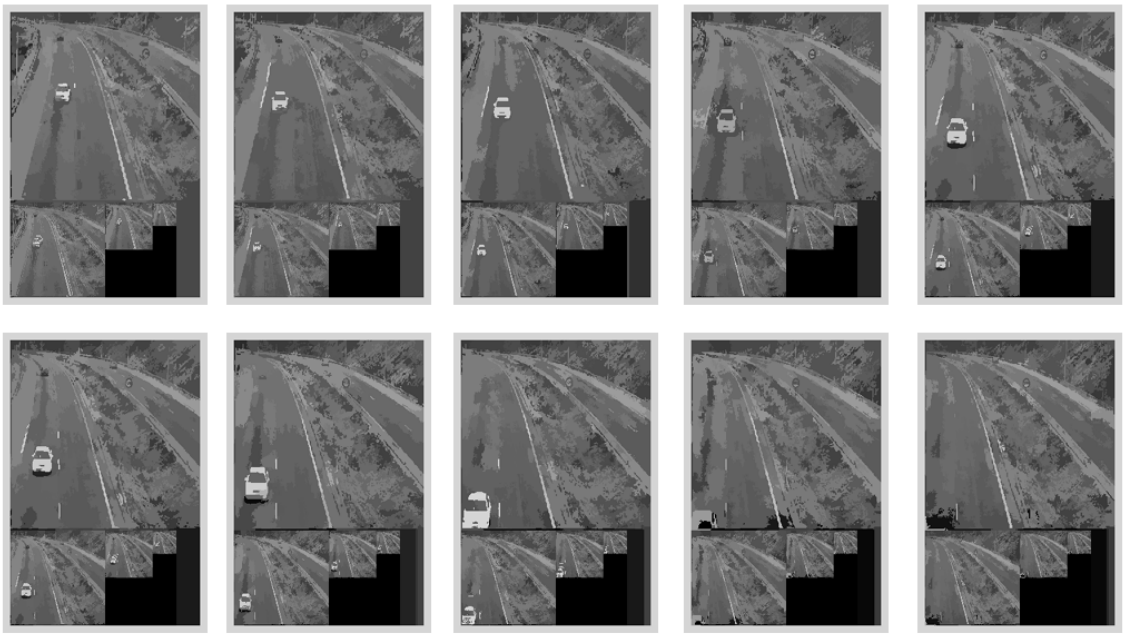


Figura 4.46: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B8: Pirámides segmentadas.

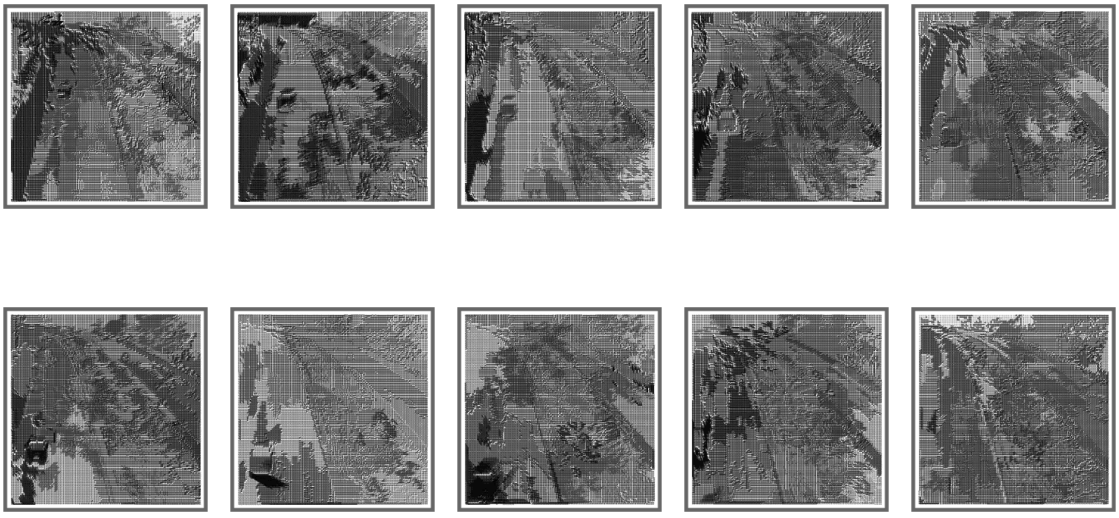


Figura 4.47: Resultados de la segmentación jerárquica espacio-temporal adaptativa de la secuencia B8: Vectores de desplazamiento estimados.

de las clases. Obsérvese, por ejemplo, cómo la estructura de clases de la maleza situada en la mediana de la carretera (Fig. 4.46) se mantiene estable, a pesar de que la descomposición podría parecer un tanto errática por las variaciones de gris de la vegetación. Lo importante en este caso es que siempre se divide en las mismas clases, lo que se manifiesta en la estimación homogénea de los vectores de desplazamiento que aparecen en la Fig. 4.48.

Los bordes de la secuencia de fotogramas suponen, en este caso, un reto para el método de segmentación, ya que el desplazamiento de la cámara induce la aparición de nuevas regiones, que suelen presentar texturas complejas. Sin embargo, dichas regiones se reestructuran de nuevo en los siguientes fotogramas conforme se dispone de más información acerca de su naturaleza. Esa es la causa de que la estimación de movimiento presente deficiencias en torno a estas zonas, ya que la única referencia que posee una clase a la hora de enlazarse al fotograma siguiente, cuando el área con la que debía hacerlo ya no se encuentra en el campo de visión, es ella misma. Así, los vectores de desplazamiento cambian de dirección hacia arriba o hacia abajo al no poder enlazarse a la zona izquierda que se pierde entre fotogramas, pero es interesante notar que el error no es aleatorio, sino que presenta una tendencia debido a la naturaleza inercial del proceso predictivo. Por otra parte, puede apreciarse que el desplazamiento del móvil se estima correctamente. Aunque el coche se desplaza hacia el sur, debido a que la cámara gira hacia la derecha, los vectores que lo representan aparecen en dirección suroeste.

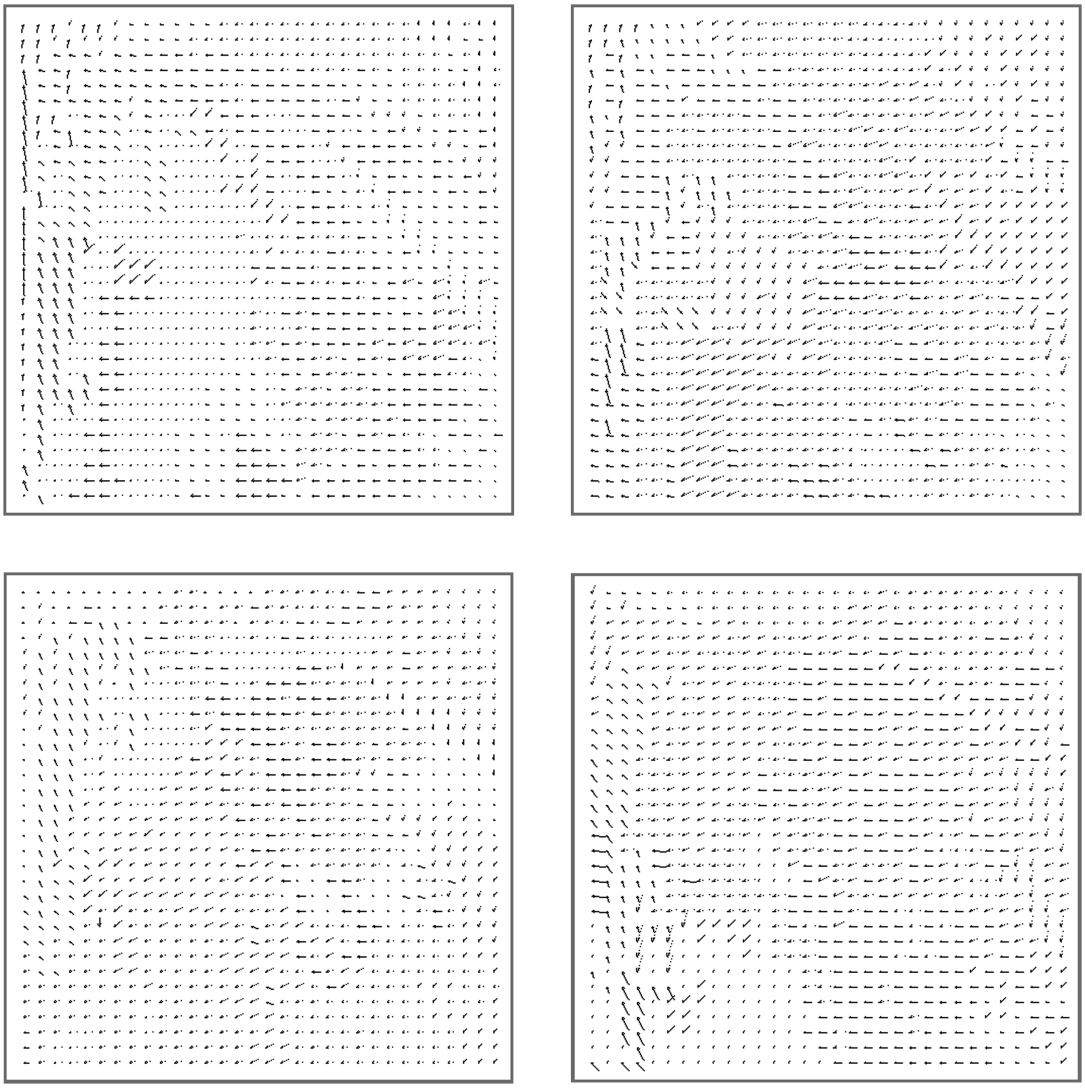


Figura 4.48: Detalle de los vectores de desplazamiento estimados para la secuencia B8

7.9 Comparación con otros métodos

El método propuesto no puede compararse en igualdad de condiciones con otras técnicas porque no sólo permite estimar el desplazamiento de las regiones que componen una imagen sino que también presenta la ventaja adicional de llevar implícita su segmentación y seguimiento a través de la estructura de enlaces [Rodríguez et al. 2001c]. Esto supone una mejora frente a la mayoría de los métodos, puesto que éstos requieren un paso final de segmentación de la escena a partir del movimiento estimado de sus píxeles. No obstante, se van a evaluar los resultados de este método frente a diversos algoritmos clásicos de estimación de movimiento comparando los vectores de desplazamiento que resultan de su aplicación.

Existen dos grupos importantes de técnicas de estimación de movimiento: las diferenciales y las cualitativas. Las primeras se basan en resolver la ecuación que describe el movimiento a partir de la variación de intensidad de los píxeles a lo largo de un conjunto de fotogramas y su principal desventaja radica en que se basan en una serie de restricciones que típicamente no se cumplen en secuencias capturadas bajo condiciones reales. Estos métodos funcionan siempre de forma muy parecida, ya que se basan en la misma formulación básica y habitualmente se dividen en métodos de primer y segundo orden, en función de las restricciones que imponen sobre la imagen pero, dado que los de segundo orden suelen ser aún más inestables que los de primero en imagen real, en esta comparación sólo se van a considerar los primeros. De entre todos ellos, dado que la variación de unos a otros es mínima, se han elegido los dos más clásicos: Horn & Schunk [Horn y Schunk 1981] y Lucas & Kanade [Lucas y Kanade 1981]. La mayor parte de los métodos diferenciales modernos se basan en estos dos, añadiendo a todo el proceso etapas previas que pretenden adaptar las imágenes originales a las restricciones impuestas por los algoritmos y evitar así su mal funcionamiento. Este preprocesado suele ir en la línea de un filtrado paso-bajo, lo que supone que a veces se pierden detalles de la escena [Mahzoun et al. 1999] [Luthon et al. 1999].

Las técnicas cualitativas eliminan este problema llevando a cabo la segmentación de dos o más imágenes consecutivas y relacionando las regiones resultantes mediante una medida de parecido. Su principal problema radica en la necesidad de conseguir una segmentación estable en cada fotograma y de establecer una medida de parecido rápida y eficaz entre regiones cuya forma y color pueden estar sujetos a variaciones. Los métodos cualitativos presentan una amplia variedad en tanto que se caracterizan por el tipo de segmentación que se aplica a cada fotograma y la medida de parecido entre regiones utilizada. Dada la enorme variedad de técnicas existentes a pesar de trabajar siempre con la misma idea, se ha optado por establecer la comparativa con

el método más representativo de este grupo. Dicho método fue desarrollado por Anandan y, si bien resulta más lento que muchos algoritmos cualitativos posteriores, también suele resultar más robusto.

En cuanto a la aplicación de las distintas técnicas, si bien el método de Anandan sólo requirió el uso de dos fotogramas para establecer los vectores de desplazamiento, al igual que el propuesto, los otros métodos necesitaron evaluar 15 fotogramas en total: el actual, siete previos y siete posteriores. El algoritmo de Anandan se ejecutó con un máximo de 15 iteraciones sobre una pirámide de 3 niveles usando una vecindad 5x5. El método de Horn y Schunk se ejecutó con un total de 100 iteraciones, y un suavizado gaussiano con $\sigma = 1.5$, lo que implica la necesidad de emplear un total de 15 fotogramas en la estabilización. El método de Lucas y Kanade también requirió el uso de 15 fotogramas (igualmente $\sigma = 1.5$) y un umbral de 1.0. Los resultados de aplicar estos métodos a la secuencia #3 del Apéndice B se muestran en la Fig. 4.49. De los resultados obtenidos se puede deducir que la mejor estimación de los vectores de desplazamiento la ofreció el método propuesto (Fig. 4.49.c), pudiendo apreciarse cómo las flechas indican correctamente la dirección de avance del móvil. Esto se debe a que el método propuesto se puede considerar una técnica mixta aplicable a cualquier tipo de escena, mientras que el método de Anandan (Fig. 4.49.c) es exclusivamente cualitativo y los otros (Fig. 4.49.b y d) son diferenciales y, por tanto, se basan en la conservación de la intensidad de los píxeles de que se componen los objetos de la imagen, circunstancia que no se da habitualmente en este tipo de escenas. Por ello los campos de movimiento resultan ruidosos y, en ocasiones, presentan vectores de dirección aleatoria en el área en movimiento. Además la velocidad del transeúnte es excesiva para los métodos de Horn y Schunk y Lucas y Kanade. Dicha velocidad roza el límite permitido por la restricción de apertura del método de Anandan, y esto se puede observar en que la porción de la escena que ocupan los vectores de movimiento es mayor que el tamaño del propio móvil, debido a la estimación errónea que se produce en algunos sectores de la periferia del mismo; este problema no aparece con el método propuesto, ya que la predicción/corrección de la posición de cada clase en fotogramas sucesivos permite reducir al mínimo estos artefactos. No obstante, el método de Anandan es el que mejores resultados ofrece, tras el propuesto, y si se cumpliese la restricción de apertura mencionada, probablemente ofrecería un resultado con mayor nivel de detalle que el nuestro. Es necesario destacar que en las zonas que no presentan texturas todos los métodos diferenciales añadieron un ruido aleatorio a la estimación, ya que carecían de elementos en los que basarla. Esto no ocurre con el método propuesto, ya que al segmentar combinadamente dos fotogramas se definen implícitamente relaciones espaciales entre ambos que permiten estimar cuantitativamente su desplazamiento. Todos los métodos, incluido el propuesto, se ven afectados por la sombra, ya que en base a los criterios empleados en el

proceso de segmentación, la sombra es un móvil perfectamente válido.

Una segunda prueba para comparar los tres métodos analizados con el propuesto se realizó empleando la secuencia #4 del Apéndice B. De nuevo, los mejores resultados pueden atribuirse al método propuesto. El método de Anandan, le sigue muy de cerca, ya que consigue caracterizar los pequeños desplazamientos del hombro de la persona situada frente a la cámara (Fig. 4.50). Sin embargo, el método propuesto ofrece una mejor estimación del fondo, ya que el de Anandan muestra un ruido aleatorio mayor, en el que incluso puede apreciarse la cuadrícula multirresolución de la pirámide Laplaciana que usa para la estimación del movimiento de la escena. En cuanto a la dirección y magnitud del movimiento, puede constatarse que el método propuesto los caracteriza adecuadamente. Una vez más, los otros dos métodos diferenciales fallan, en este caso más notoriamente en tanto que la escena está sujeta a luz artificial y, por tanto, a variaciones mucho más marcadas de luces y sombras. Obsérvese también que ambos métodos sólo presentan variaciones en las fronteras entre regiones, que es donde pueden apreciarse variaciones de textura suficientemente marcadas.

En cuanto a tiempos de proceso, todos los algoritmos se ejecutaron sobre un Pentium II a 333 MHz con 96 Mbytes de memoria RAM. Las imágenes de que constaban las secuencias empleadas tenían unas dimensiones de 256x256 píxeles, aunque también se midió el tiempo de procesamiento para versiones reducidas de las imágenes con 128x128 y 64x64 píxeles respecti-

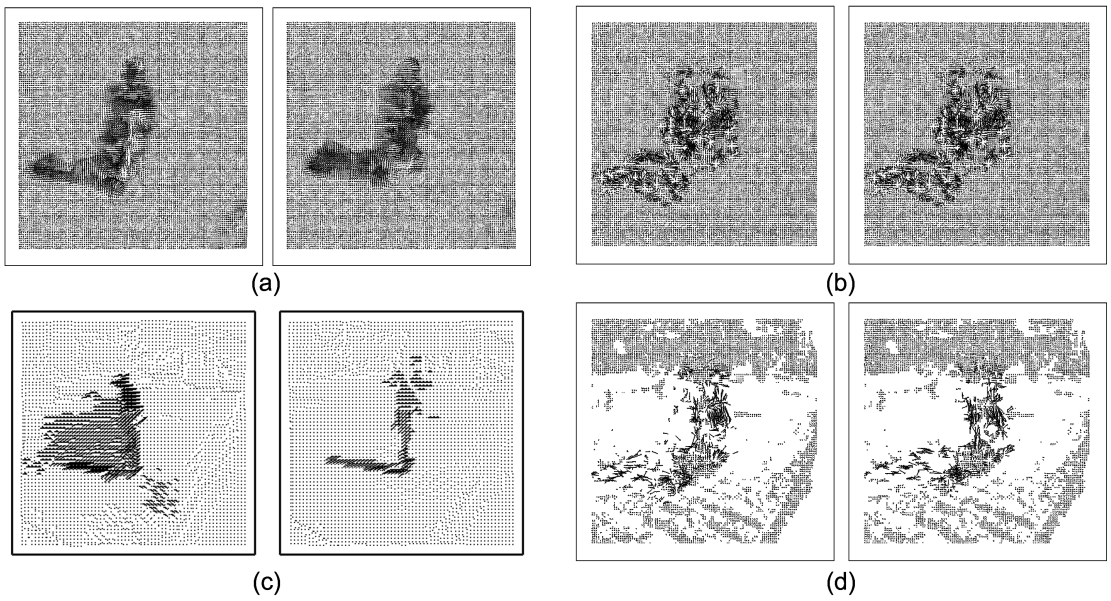


Figura 4.49: Desplazamientos estimados para la secuencia B3 mediante los métodos: a) Anandan; b) Horn y Schunk; c) propuesto; d) Lucas y Kanade.

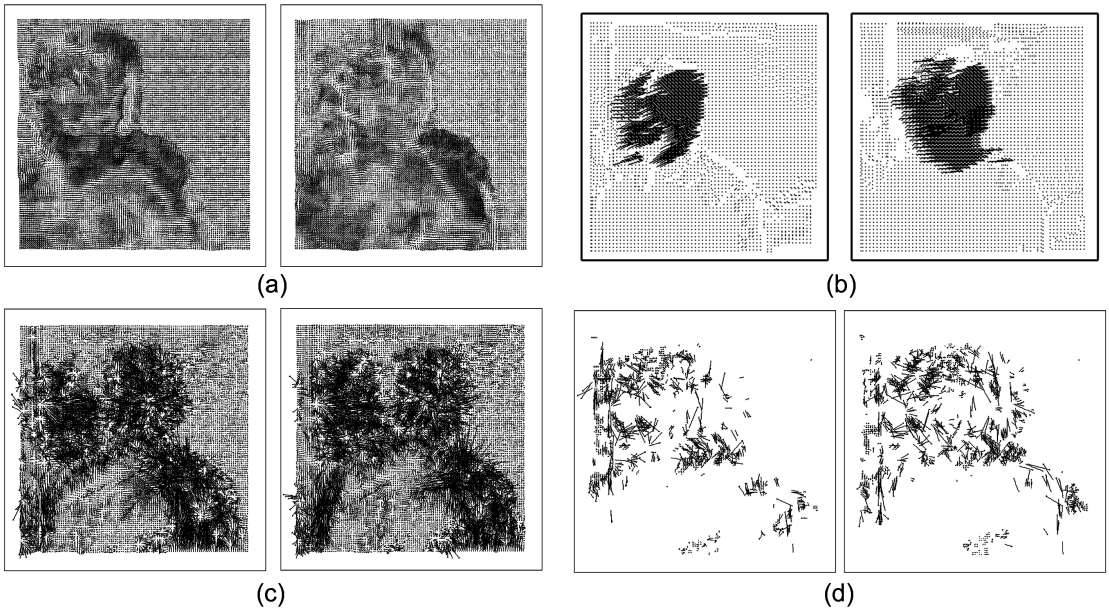


Figura 4.50: Vectores de desplazamiento estimados para la secuencia B4 mediante los métodos: a) Anandan; b) propuesto; c) Horn y Schunk; d) Lucas y Kanade.

vamente (ver Tabla 4.1). Para imágenes de 256x256 píxeles, el método de Anandan requirió un total de 206 seg. frente a los 30 seg. que requirió el método de Horn y Schunk y los 10 seg. del método de Lucas y Kanade. El método propuesto empleó únicamente 2.9 seg. En cualquier caso, si se observa la Tabla 4.1, se puede concluir a partir de los resultados obtenidos, que el método propuesto podría trabajar en tiempo real a una velocidad máxima de 13 imágenes por segundo si usamos versiones diezmadas de 64x64 píxeles de las imágenes originales, objetivo imposible de alcanzar por el resto de métodos analizados. Además, es necesario remarcar que, si bien los tres métodos analizados necesitan una etapa posterior de agrupamiento de clases en función de los vectores obtenidos, el método propuesto ya incluye este paso de segmentación en regiones gracias a su naturaleza espacio-temporal y a la forma en que se realiza el enlazado de la secuencia de imágenes. Sólo queda mencionar que el método de Anandan, para secuencias de vídeo que presenten desplazamientos del fondo o de los objetos más suaves, ofrece los mejores resultados con gran nivel de detalle. Sin embargo, la gran complejidad de este método y esta importante restricción lo convierte en un método impracticable con los recursos existentes actualmente.

	Anandan	Horn y Schunk	Lucas y Kanade	Método propuesto
$256x256(seg.)$	206	30	10	2.9
$128x128(seg.)$	51	8.4	1	0.75
$64x64(seg.)$	11	2.2	0.3	0.075

Tabla 4.1: Tiempos de procesamiento para distintos métodos de obtención de flujo óptico

8 Conclusiones

A lo largo de este capítulo se ha presentado un nuevo método de segmentación espacio-temporal que permite dividir una escena en clases en función de la cantidad de movimiento que presentan y su nivel de gris. El método, que se basa en segmentar adaptativamente de forma jerárquica y combinada fotogramas consecutivos, ha demostrado que mantiene la coherencia de la división de clases de un fotograma a otro. Este sistema se ha probado sobre una gran variedad de secuencias, presentándose los resultados de aquellas particularmente representativas. El método propuesto se ha comparado con otros métodos clásicos de estimación de desplazamiento, tanto de tipo cualitativo como diferencial. Se ha comprobado que en todos los casos los métodos cualitativos ofrecen mejores resultados que los diferenciales para secuencias de imágenes reales, lo que parece lógico en tanto que difícilmente se cumplen en secuencias reales las condiciones de trabajo sobre las que se sustentan los métodos diferenciales. La técnica cualitativa comparada puede llegar a ofrecer mejores resultados de estimación de movimiento que el método propuesto, pero siempre a costa de un incremento muy considerable del tiempo de cómputo y sólo en condiciones de trabajo más restrictivas que el nuestro. Por otra parte, el sistema de enlazado predictivo utilizado y la metodología de segmentación a partir de resultados de otras segmentaciones previas, consiguen dotar al método de una inercia que le permite ofrecer mejores resultados que el resto en ausencia de rasgos representativos a partir de los que extraer el movimiento.

La aplicación de la técnica propuesta sobre distintas secuencias ha demostrado sus ventajas como método segmentador espacio-temporal, así como su capacidad para abordar complejas situaciones que incluían: objetos móviles deformables, variaciones de iluminación, alto ruido de captura debido al uso de sensores comerciales, modelos no lineales de movimiento, tanto de los objetos como del fondo, perspectivas elevadas, movimiento de móviles sobre fondo móvil, etcétera. Además, el método es capaz de manejar velocidades altas de móviles sin pérdida de integridad en las clases y devuelve directamente un conjunto de regiones sin necesidad de posteriores segmentaciones que dividan la escena de acuerdo a los desplazamientos de las distintas partes que la componen, cosa que no ocurre con el resto de métodos analizados.

La técnica propuesta también presenta una serie de desventajas, que por otra parte comparte con el resto de los métodos existentes de detección y estimación de movimiento existentes: provoca indeseables efectos en los bordes ante el desplazamiento de la cámara y obtiene desplazamientos erráticos cuando se producen ocultaciones y descubrimientos de forma brusca. Estas deficiencias, si bien son inevitables para cualquier sistema de visión, se deben tratar mediante técnicas de más alto nivel que, mediante un refinado posterior elimine los errores intro-

ducidos. Por otro lado, el modelo predictivo que utiliza es excesivamente simple por ser lineal y no permite prever movimientos complejos, lo que en un momento dado puede ralentizar el proceso de estabilización adaptativa. Todos estos inconvenientes serán materia de estudio futuro, ya que la mayoría podrían eliminarse mediante un proceso de refinado posterior a la generación de clases propuesta.

Capítulo 5

Desarrollo de una aplicacion basada en el método de segmentación propuesto

1 Introducción

Existe una gran variedad de aplicaciones relacionadas con el procesamiento de secuencias de vídeo en las que se requiere la identificación y clasificación de regiones presentes en la escena atendiendo a criterios espaciales y de movimiento [Rodríguez et al. 2001a] [Rodríguez et al. 2001c]. Para probar la validez del método de segmentación propuesto, se ha empleado como etapa previa a un sistema de compresión de una secuencia de vídeo, con el fin de transmitirla a través de un canal de ancho de banda variable. La transmisión de señales de vídeo es necesaria en algunas aplicaciones basadas en visión artificial, bien porque el procesamiento de las imágenes se realiza de forma distribuida, bien porque el dispositivo de captura no tiene capacidad de proceso o bien, sencillamente, porque el receptor es remoto. Dado el volumen de datos que genera una secuencia de vídeo se tiende habitualmente a comprimir dicha secuencia antes de transmitirla, especialmente si no se dispone de un medio de comunicación de banda ancha. El vídeo comprimido es una de las señales de tasa variable más típicas, en la que además, la información suele ser de carácter racheado. Ello supone que, si bien durante buena parte de la transmisión el volumen de datos a transmitir se mantiene acotado, con mucha frecuencia aparecerán ráfagas impredecibles con gran cantidad de información.

Si se trabaja con aplicaciones sensibles al retardo, a efectos de garantizar una calidad de servicio determinada, la red debe ofrecer algún tipo de mecanismo de reserva de recursos que permita aumentar el ancho de banda disponible durante las ráfagas de transmisión. En caso

contrario, mientras suceden dichas ráfagas, el retardo aumenta de forma no controlada y las aplicaciones que dependen de una tasa fija de imágenes por segundo fallan. El problema de los mecanismos de reserva consiste en que no es sencillo calcular la cantidad de recursos necesarios durante los instantes de máxima transmisión, ya que su aparición depende de la naturaleza de la secuencia que se está transmitiendo. La única alternativa cuando no es posible calcular el ancho de banda necesario en cada instante consiste en sobredimensionarlo, de manera que incluso en el peor de los casos se pueda mantener un ritmo constante en la comunicación. Obviamente, esta solución supone una infrautilización de los recursos durante la mayor parte del tiempo. La situación puede empeorar si se comparte el canal con otros tipos de comunicaciones, especialmente si éstas son de ancho de banda fijo, ya que todos los problemas del canal repercutirán directamente sobre la transmisión de vídeo.

Si bien en la literatura existen multitud de técnicas que permiten manejar los recursos de forma dinámica, resultaría sumamente interesante modular la transmisión de forma que, en caso de congestión, el tráfico generado por la fuente de vídeo se reduzca para permitir mantener una tasa de imágenes fija. Dado que la naturaleza de la escena capturada no puede controlarse, podría recurrirse a técnicas de visión activa [Griffioen et al. 1995] para transmitir a velocidad constante sólo las áreas de interés, prescindiendo del resto cuando las condiciones del canal así lo requieran. Sin embargo, la transmisión selectiva de partes de la imagen impone severas limitaciones al procesado remoto de éstas y a su presentación en el extremo receptor.

Las imágenes multirresolución permiten solventar este problema mediante la representación de las distintas áreas de la imagen con niveles de resolución diferentes [Camacho et al. 1997] [Camacho et al. 1998]. Si las áreas menos interesantes se representan a baja resolución y las áreas importantes siguen presentando un alto nivel de detalle, el volumen de datos de la imagen resultante se ve considerablemente reducido. De esta forma, el flujo de vídeo puede adaptarse al estado de la red reduciendo selectivamente la resolución de la imagen: si el retardo del canal comienza a crecer, se puede recurrir a la reducción de la resolución para mantener una tasa fija de imágenes por segundo [Urdiales et al. 2000].

A efectos de aplicar los conceptos presentados en los capítulos anteriores y de probar su validez, este capítulo presenta un sistema que permite mantener en un canal sometido a distintas condiciones de retardo una tasa constante de transferencia de imágenes por segundo mediante técnicas de visión activa e imágenes multiresolución. La técnica que se describe en el presente capítulo [Rodríguez et al. 2001d] se puede englobar en los algoritmos de compresión de imagen basada en objetos, cuyos conceptos se introducen en el apartado 2. El apartado 3 se ocupa de presentar tanto la propuesta de codificación como el sistema global; el funcionamiento de los

módulos de segmentación, seguimiento y compresión se incluyen en el apartado 4; y finalmente, los apartados 5 y 6 ofrecen los resultados y las conclusiones que pueden extraerse de este trabajo.

2 Compresión de imagen basada en objetos

Las técnicas convencionales de codificación de imagen digital o de secuencias de vídeo se basan, generalmente, en dos conceptos fundamentales: el muestreo en el tiempo y el espacio, y la cuantificación del brillo o color. De esta forma, el algoritmo típico de compresión define sobre cada imagen unas determinadas vecindades espaciales (píxeles adyacentes o bloques cuadrados) o espacio-temporales (regiones conectadas entre dos o más fotogramas), o bien algún tipo de transformación reversible lineal sobre estas vecindades (como la DCT, *discrete cosine transform* [Uenohara y Kanade 1998]), y posteriormente les asigna un determinado código. El estudio estadístico de las distribuciones resultantes de estos códigos permite mejorar los algoritmos de codificación y, por tanto, la compresión de la imagen o secuencia de vídeo.

El principal problema de los esquemas clásicos de compresión es que la calidad que ofrecen puede saturarse rápidamente [Ebrahimi 2000], ya que las imágenes o secuencias de vídeo reales pueden llegar a ser muy dinámicas, lo que se traduce en una fuerte variación, tanto espacial como temporal, de sus propiedades estadísticas. Aunque resulta interesante disponer de un algoritmo de muestreo que se adapte a estas condiciones, en la práctica resulta sumamente difícil. El último problema, y no por ello menos importante, de estas estrategias de compresión es que las vecindades definidas (como el caso de los bloques cuadrados en los que se divide la escena en el estándar MPEG) no pueden describir correctamente las condiciones no estacionarias de las imágenes o secuencias de vídeo reales.

La mejora más significativa introducida en los algoritmos de compresión ha consistido en el empleo de un nuevo tipo de vecindad, el objeto o región, que se define principalmente por su contorno y textura, y que sigue un movimiento determinado a lo largo de la secuencia de vídeo como entidad compacta. Esta nueva vecindad permite resaltar unos determinados objetos del campo de visión, considerados como importantes, mientras el resto de la imagen se trata con unos parámetros de calidad inferiores. Evidentemente, no sólo habrá que segmentar la imagen en un conjunto de objetos, sino que será necesario relacionarlos a lo largo de toda la secuencia. Esta doble tarea, que supone la identificación de un objeto como una entidad que tiene una continuidad tanto espacial como temporal, ha sido el germen de la codificación de segunda generación, que constituye la base filosófica del nuevo estándar MPEG-4 [ISO/IEC 1996] [ISO/IEC 1998].

Finalmente, resaltar que esta nueva estrategia de codificación no se ha presentado como un sistema cerrado, sino que una vez definidas las entidades, éstas pueden ser tratadas individualmente. Esta forma de trabajar, que se conoce como codificación dinámica [Reusens et al. 1997], consiste en comprimir cada entidad de la forma más eficiente, y complementa el sistema de definición de vecindades mediante la individualización de las entidades que se extraen de la secuencia.

En el siguiente subapartado se establecen las bases de la codificación de vídeo de segunda generación. El esquema de codificación propuesto será desarrollado en profundidad en los restantes apartados del presente capítulo.

2.1 Codificación de vídeo de segunda generación

Hasta mediados de la década de los 80, todos los sistemas de representación o procesado de información visual tomaban como unidad el píxel. Sin embargo, en esta época, y motivados por el análisis del comportamiento del sistema de visión humano, los investigadores comenzaron a desarrollar otros métodos de representación [Kunt et al. 1985]. Estas estrategias, que no se basan en el uso del píxel como unidad básica de las imágenes, constituyen la codificación de vídeo de segunda generación, y han demostrado una mayor eficiencia de codificación respecto a las estrategias clásicas de codificación.

Evidentemente, el primer problema de las técnicas de codificación de segunda generación es la obtención de las entidades básicas de representación. Dado que la captura suele ofrecer una imagen representada por un conjunto de píxeles, habrá que desarrollar un sistema que, combinando la información de los píxeles, obtenga las nuevas entidades básicas en que se divide la imagen capturada. Existen distintos tipos de estas entidades básicas, pero las más extendidas son las que se denominan objetos. Un objeto se puede definir como un conjunto de regiones que representan una individualidad con significación dentro de la escena [Castagno et al. 1998]. Estas regiones en que se divide cada objeto serán, simplemente, porciones de la imagen con una determinada característica común. Por lo tanto, en la codificación basada en objetos, éstos sustituyen a los píxeles, y la secuencia de vídeo se dividirá en un conjunto de objetos que no pueden ser separados en elementos menores.

La codificación basada en objetos ha sido ampliamente estudiada en estos últimos años, apareciendo un importante conjunto de métodos [Torres y Kunt 1996] [Ebrahimi y Kunt 1998], algunos de ellos motivados por el hecho de que este esquema de codificación haya sido adoptado por el estándar MPEG-4. Las principales diferencias entre los distintos métodos dependen de:

- El método específico empleado para delimitar el objeto.
- El método usado para codificar la información de color, textura o nivel de gris del objeto.
- El método empleado para estimar y codificar el movimiento del objeto.
- El algoritmo de integración de la información obtenida.

3 Definición del entorno de trabajo

3.1 Estrategia de compresión propuesta

La definición formal de un método de codificación de segunda generación implica el desarrollo de todo un conjunto de métodos de caracterización que permitan identificar correctamente cada objeto [Menegaz et al. 1999]. En este trabajo, las características empleadas serán el nivel de gris y el movimiento, que permiten obtener un resultado satisfactorio [Kim y Kim 2000]. Añadir otras características, como la forma o la textura [Ebrahimi 2000], ha sido desestimado por distintas razones: la gran capacidad de deformación de los objetos elimina la forma como factor diferenciador de un objeto, y el aumento de la complejidad computacional que supone el cálculo de texturas dificulta su aplicación en tiempo real.

El empleo del nivel de gris y el movimiento se combinan en el método de segmentación de imágenes que lleva a cabo el algoritmo de enlazado piramidal espacio-temporal adaptativo descrito en el capítulo 4, por lo que no ha sido necesario establecer ningún mecanismo de fusión de ambos criterios tras extraerlos de la secuencia.

En cuanto al esquema de compresión empleado, se basa en algoritmos de generación de imágenes multirresolución. Estas estructuras se describen detalladamente en el desarrollo del Apéndice A. En este caso, al poder coexistir distintos objetos de interés en una misma imagen, se utilizan estructuras multifoveales [Camacho et al. 1998], que permiten disponer de todos estos objetos con resolución máxima.

3.2 Sistema de pruebas

El sistema de control de flujo implementado se ha probado en un entorno controlado utilizando secuencias de vídeo de imagen real capturadas en diferentes entornos. La infraestructura básica consiste en dos ordenadores personales (PCs) conectados a una red de área local (LAN) con una velocidad de transmisión media de, aproximadamente, 700 Kbytes por segundo bajo

condiciones normales de tráfico (Fig. 5.1). Una de las máquinas (servidor) posee una cámara de vídeoconferencia estática conectada a una tarjeta capturadora insertada en uno de sus *slots* PCI que captura vídeo en niveles de gris a una tasa de 10 imágenes por segundo. La secuencia de entrada presenta resolución uniforme y el servidor realiza la conversión a imagen multifoveal con resolución variable en el espacio [Camacho et al. 1998] de acuerdo a los algoritmos descritos en el Apéndice A. El segundo ordenador (cliente) recibe imágenes foveales que, de funcionar correctamente el sistema, llegan con una tasa constante (10 frames/seg.). Puede observarse que todos los algoritmos de detección de movimiento y predicción se llevan a cabo en el servidor sobre imágenes de resolución uniforme, mientras que el cliente recibe imágenes foveales y, por tanto, los algoritmos de proceso que emplee deben funcionar con resolución no uniforme.

Para medir la capacidad del enlace disponible, el PC servidor estima el retardo mediante el envío periódico de paquetes de control al PC cliente, que éste devuelve. Este simple procedimiento permite que el PC servidor tenga una referencia continua de la capacidad del canal. Este proceso de control se realiza en paralelo a la transmisión comprimida de vídeo en un sistema que trabaja sobre Linux.

A continuación se describen con mayor nivel de detalle las acciones que se realizan en ambos PCs. Estas acciones se presentarán en profundidad en el apartado 4 del presente capítulo.

3.3 Tareas del PC servidor

El PC servidor recibe una tasa de imágenes por segundo constante, procedentes de una cámara de vídeo a través de una tarjeta capturadora. Las acciones que lleva a cabo este PC son:

- Generación de la estructura piramidal asociada a la imagen de entrada.
- Enlazado espacio-temporal adaptativo entre la pirámide asociada al instante t y la asociada al instante $t-1$.

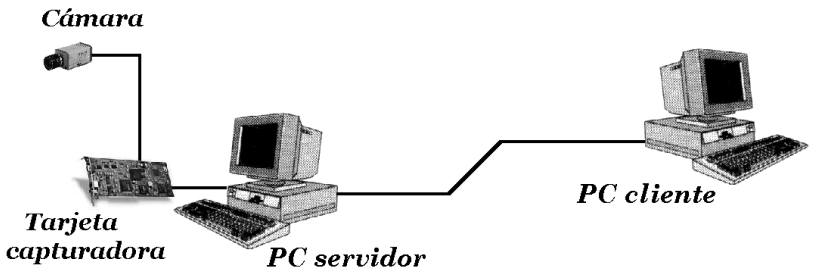


Figura 5.1: Esquema físico del sistema de compresión.

- Estimación del movimiento de las regiones de la imagen; paralelamente, se lleva a cabo una segmentación, basada en nivel de gris y movimiento, de la imagen.
- En función de la información obtenida en los pasos anteriores, se seleccionan las regiones de interés de la imagen.
- Construcción de la imagen multifoveal; dicha imagen presenta las regiones de interés en alta resolución, mientras que el resto de la imagen se engloba en anillos de menor resolución.
- Ordenación de los paquetes a transmitir; la estructura final de estos paquetes dependerá de las condiciones que se midan en el canal.

3.4 Tareas del PC cliente

El PC cliente recibirá, salvo que la capacidad del canal se reduzca drásticamente, una tasa de imágenes multifoveales por segundo igual a la generada en transmisión. En función de las condiciones del enlace, las imágenes incluirán todos los anillos de resolución o sólo parte de éstos. En principio, sólo se asegura que las fóveas son siempre transmitidas. Una vez recibidas las imágenes, el PC cliente realiza las siguientes acciones:

- Reconstrucción de la imagen multifoveal. Para ello, la información recién recibida se combina, si es necesario, con la almacenada de fotogramas previos, para generar una imagen completa de resolución variable en el espacio.
- Aplicación de los algoritmos de procesamiento basados en imagen multirresolución en el PC cliente, si procede.

4 Análisis modular del sistema

El sistema de compresión propuesto se divide en cuatro módulos importantes (Fig. 5.2). El primer módulo (segmentación-seguimiento) realiza la segmentación espacio-temporal de la secuencia de vídeo recibida, obteniendo un conjunto de regiones que presentan una continuidad tanto espacial como temporal. A continuación, el módulo de procesamiento jerárquico determina qué regiones son las más significativas de cada fotograma, y genera una imagen multifoveal de resolución variable que contiene las zonas de interés con resolución máxima, mientras que el resto de la imagen se incluye con una menor resolución. El tercer módulo (compresión) genera la secuencia de paquetes que se transmiten del PC servidor al cliente. Para determinar la

estructura de estos paquetes, este módulo usa un submódulo de control, que se comunica con el PC cliente. Finalmente, el módulo de descompresión que reside en el PC cliente realiza la reconstrucción de una imagen multirresolución a partir de los datos recibidos y, si es necesario, de los datos almacenados de recepciones anteriores.

A continuación se describen cada uno de estos módulos con mayor nivel de detalle, incidiendo tanto en los distintos submódulos que los forman, como en las comunicaciones internas entre éstos.

4.1 Segmentación-seguimiento de objetos

El módulo de segmentación y seguimiento de objetos es el más importante del sistema, y en él se realiza el proceso de segmentación espacio-temporal, que constituye el objeto de esta tesis. Dicho módulo se divide en cinco submódulos, que llevan a cabo simultáneamente las tareas de:

- Segmentación de la imagen de entrada usando para ello el nivel de gris y la posición de cada objeto, tanto espacial como temporalmente.
- Seguimiento de los objetos, de forma que se pueda identificar, en cualquier instante de tiempo, la ubicación de un determinado objeto.

Por lo tanto, este módulo es el encargado de detectar las posiciones que ocupan las entidades de alto nivel (objetos), que serán empleadas posteriormente por el resto de módulos del sistema, pero que también podrían servir de entrada a otros tipos de codificadores, como el propio MPEG-4.

Las distintas tareas que se realizan en este módulo son (Fig. 5.2):

- Generación de la estructura piramidal: este bloque es el encargado de generar la estructura piramidal que tiene por base el fotograma adquirido en el instante de tiempo t.
- Almacenamiento de la pirámide adaptativa $(t-1)$: para estimar el desplazamiento asociado a cada uno de los objetos, así como para llevar a cabo el seguimiento de éstos, se dispone en memoria de la estructura piramidal adaptativamente enlazada asociada al fotograma $t-1$.
- Enlazado espacio-temporal adaptativo: este algoritmo constituye el núcleo del módulo de segmentación-seguimiento y, por ello, del sistema; básicamente, como ya se ha comentado en el Capítulo 4, consiste en un doble enlazado de las estructuras piramidales asociadas a los instantes t y $t-1$.

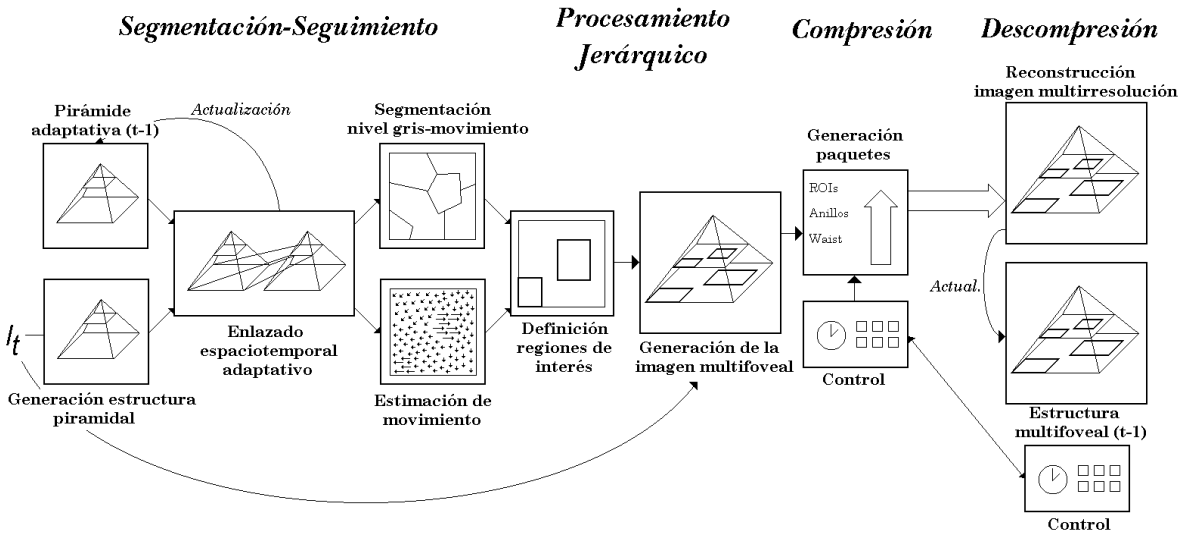


Figura 5.2: Esquema modular del sistema implementado.

- Segmentación por nivel de gris y movimiento: una vez concluido el proceso de enlazado, se tiene una segmentación de la imagen basada en el nivel de gris y el movimiento, donde cada objeto se etiqueta en función de la homogeneidad espacio-temporal.

- Estimación de movimiento: de la estructura de enlaces resultante se pueden extraer la imagen segmentada y una estimación de movimiento asociada a cada región de la imagen.

4.2 Procesamiento jerárquico

Este módulo decide qué clases pertenecen a cada objeto presente en la imagen, y construye una estructura multirresolución, en la cual aparecen las regiones de interés representadas con resolución máxima, y el resto de la escena mediante anillos de resolución decreciente. De esta forma, se obtiene una imagen multirresolución de la escena con un volumen de datos mucho menor, pero con una pérdida de información relevante mínima. Como muestra la Fig. 5.2, este módulo se divide en dos submódulos:

- Definición regiones de interés: empleando toda la información obtenida en el módulo de segmentación-seguimiento, se determina cuáles son los objetos que más información aportan a la escena (ROIs: *regions of interest*); en este caso, los objetos de interés han sido aquellos que presentan una mayor movilidad.

- Generación de la imagen multifoveal: empleando el algoritmo de generación de imagen multifoveal descrito en el apéndice A, se obtiene una estructura jerárquica en la que se

tienen las regiones asociadas a las ROIs con resolución máxima, mientras que las regiones no relevantes se representan con una menor resolución, en función de su proximidad a las ROIs.

La técnica empleada para obtener las imágenes foveales de resolución variable en el espacio se basa en las estructuras piramidales usadas en visión [Jolion y Rosenfeld 1994]. Estas estructuras presentan un nivel inferior, conocido como base de la pirámide, que es la imagen de resolución uniforme capturada, y, aplicando sucesivos promediados 4 a 1 de las celdas computadas, se obtienen los *rexels* o celdas de los niveles superiores de la estructura. De esta forma, cada nivel de la estructura piramidal presenta una cuarta parte de las celdas del nivel inmediatamente inferior, y presenta el mismo campo de visión pero con la mitad de resolución. Para reducir el volumen de datos a procesar, la imagen foveal se construye seleccionando, del nivel inferior, las regiones de interés (ROIs o fóveas) de la escena. Además, para satisfacer la demanda de un campo de visión grande que postula la visión activa, las regiones en torno a las fóveas son igualmente seleccionadas, pero a menor resolución, manteniendo de esta forma la información necesaria para determinados procesos atencionales, como la detección de posibles objetos, aunque no para otros, como el reconocimiento de formas o patrones[Camacho et al. 1998].

La estructura de una imagen multifoveal con dos fóveas y dos niveles distintos de resolución se muestra en la Fig. 5.3. Se aprecia como las fóveas mantienen la resolución de la imagen capturada, mientras que el resto de la escena se estructura en anillos de resolución decreciente que rodean a estas fóveas. La única región que conserva todo el campo de visión se conoce como *waist* o cintura de la estructura, y su resolución será la peor de las que se reflejan en la imagen multirresolución.

En la Fig. 5.4a-c se muestran algunos ejemplos de imágenes multifoveales, construidas con dos anillos de resolución más fóveas. En la Fig. 5.4.a (512x512 píxeles) se han ubicado hasta cuatro fóveas, pudiendo apreciarse como la intersección de los anillos de mejor resolución permite tener una resolución relativamente buena en toda la zona donde se ubican posibles móviles. Las Figs. 5.4b-c son de 128x128 píxeles, y se han empleado dos fóveas en cada una de ellas.

4.3 Compresión de imagen

El módulo de compresión de imagen consta de un submódulo de control y otro de generación de paquetes. El primero de estos módulos se encuentra en continua comunicación con el PC cliente, mide el retardo del canal, y con ello puede realizar el control del volumen de datos que se pueden transmitir por el canal en cada momento. El segundo submódulo genera el paquete

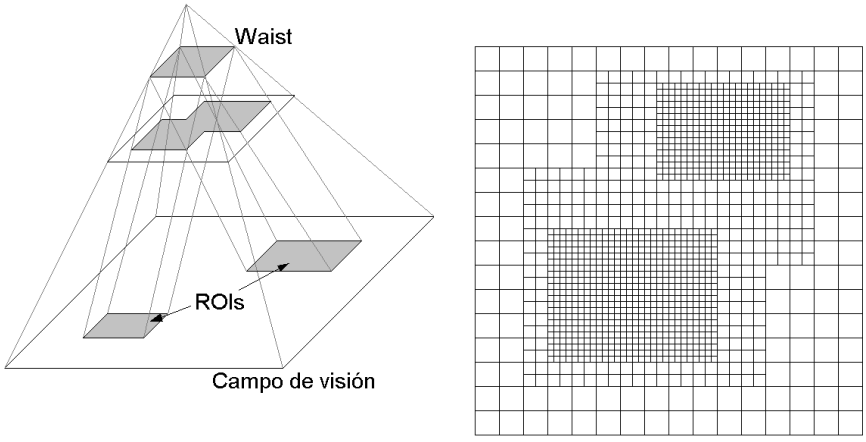


Figura 5.3: Estructura multifoveal de resolución variable.

de datos que será enviado en función de una variable interna denominada *modo*, cuyo valor es asignado por el submódulo de control. El valor de *modo* determina si el paquete enviado consta sólo de las fóveas detectadas, o si éste incluye un mayor número de anillos de resolución. En caso de que el canal tenga el suficiente ancho de banda, se transmitirá la totalidad de la imagen multirresolución.

Para medir el retardo, y así poder seleccionar el valor de *modo*, el submódulo de control envía un paquete de control al PC cliente, que éste le devuelve. El retardo medio del canal en ese momento es igual a la mitad de la diferencia entre la medición de tiempo actual y la que ha viajado en el paquete de control recibido. De esta forma es posible estimar la velocidad de transmisión del canal (Ec. (5.1)) en ese momento, y conocer el volumen de datos que puede ser

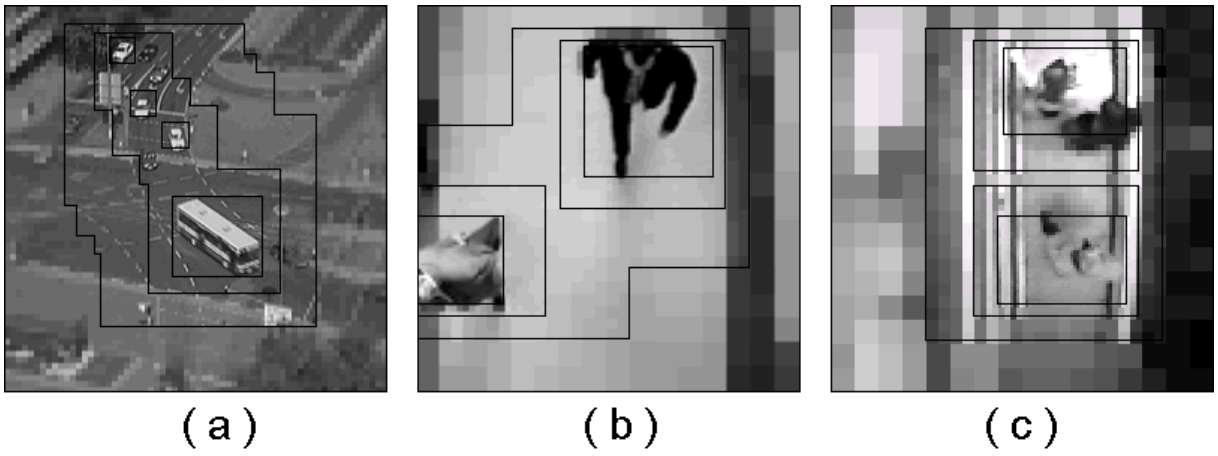


Figura 5.4: a-c) Ejemplos de imágenes multifoveales con dos fóveas y dos anillos de resolución.

transmitido si se desea mantener una tasa de imágenes por segundo constante.

$$v_{TX} = \frac{N^o \text{ de bytes del paquete de control}}{Retardo/2} = \frac{2 \cdot N^o \text{ de bytes del paquete de control}}{Retardo} \tag{5.1}$$

El paquete de control empleado en este caso tenía un tamaño constante de 50 bytes, siendo enviado cada 200 ms. Si se asume que el retardo no cambia significativamente entre envíos consecutivos, se cumplirá la siguiente ecuación:

$$v_{TX} = \frac{2 \cdot N^o \text{ de bytes del paquete de control}}{Retardo} = Tamaño \text{ de imagen} \cdot N^O \text{ imag./seg.} \tag{5.2}$$

Si las imágenes presentan un tamaño constante, el producto del retardo por el número de imágenes es igualmente constante. En la Fig. 5.5 se presenta la velocidad de transmisión para distintos valores de retardo del canal al enviar un paquete de control de 50 bytes, cuando se usan imágenes de 256x256 píxeles (65536 bytes), imágenes unifoveales centradas con fóvea de 32x32 píxeles y tres anillos de resolución distintos (3328 bytes), e imágenes unifoveales incompletas: fóvea y dos anillos (2560 bytes), fóvea y un anillo (1792 bytes) y sólo fóvea (1024 bytes). Así, si el retardo es de 2.5 ms., el número máximo de imágenes de resolución uniforme que pueden ser enviados es menor de una, y, sin embargo, se pueden enviar más de 12 imágenes multirresolución completas por segundo.

Ya que tanto el tamaño como la posición de las distintas fóveas cambia en función del

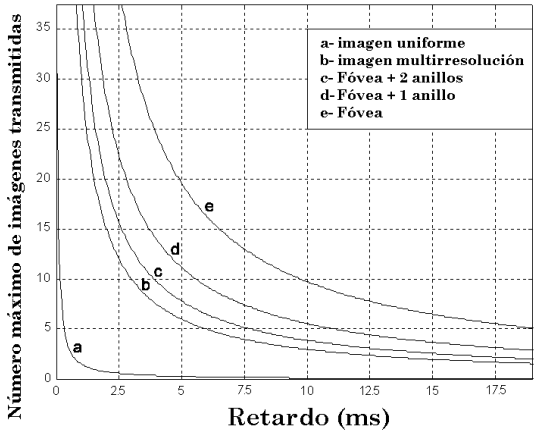


Figura 5.5: Velocidad de transmisión máxima en función del retardo para diferentes esquemas de transmisión.

desplazamiento de los objetos de interés, el volumen de datos que debe ser enviado por cada fotograma de la secuencia habrá de ser recalculado, incluso si el retardo se mantiene constante. Por otro lado, el retardo puede ser excesivamente alto incluso para la transmisión de imágenes multirresolución. En este último caso, para conseguir la tasa de imágenes por segundo requeridas en el PC cliente, el volumen de datos a enviar deberá reducirse, disminuyendo para ello el número de anillos de resolución transmitidos.

En función de la estimación del retardo, y analizando el volumen de datos que ocupan las regiones de distinta resolución, el submódulo de generación de paquetes de imágenes multirresolución generará el bloque de datos a enviar al PC cliente. La Tabla 5.1 muestra la correspondencia entre valores de modos y esquemas de transmisión empleados, mientras que en la Fig. 5.6 se presenta la estructura típica de un paquete que consta de dos fóveas, 1 anillo y $waist$. Tal y como puede apreciarse en la figura, el paquete consta de un primer parámetro, $N.fov$, que indica el número de estructuras multirresolución de las que consta el paquete. Para cada una de estas estructuras se incluye un bloque de parámetros que definen dicha estructura (Ver Apéndice A): Ld, número de subanillos dentro de cada anillo a la izquierda de la fóvea; Rd, número de subanillos dentro de cada anillo a la derecha de la fóvea; Td, número de subanillos dentro de cada anillo encima de la fóvea; Bd, número de subanillos dentro de cada anillo debajo de la fóvea; m, número de niveles de la estructura; Sx, dimensión horizontal de la imagen original; Sy, dimensión vertical de la imagen original; y $modo$, que representa el esquema de transmisión elegido para cada estructura.

4.4 Descompresión

Lógicamente, la descompresión consiste en la operación inversa a la del módulo de compresión. Para ello, se reciben los paquetes enviados desde el PC servidor, y, en función del valor de la variable $modo$, que se incluye en la cabecera de estos paquetes, se construye una imagen multirresolución que puede incluir parte de la información recibida en paquetes anteriores, si ésta no está presente en el paquete recibido.

modo	Esquema de transmisión
0	Imagen multifoveal completa
1	Fóveas más dos anillos
2	Fóveas más un anillo
3	Fóveas

Tabla 5.1: Modos de transmisión con tres anillos de resolución

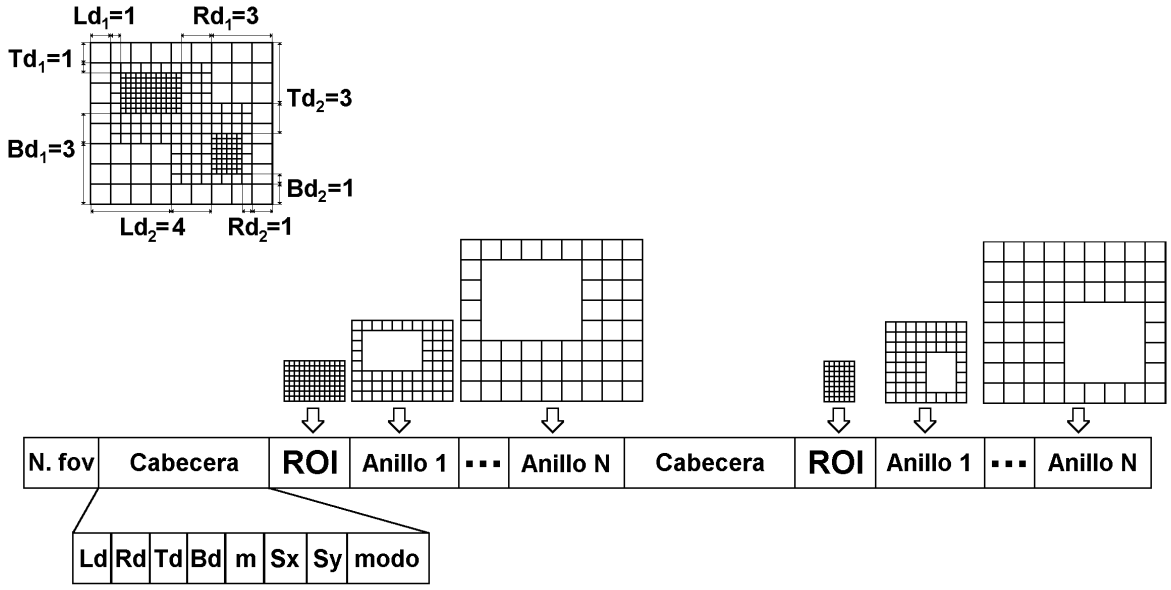


Figura 5.6: Esquema del paquete a transmitir.

Existen tres tareas distintas a realizar en esta parte del sistema:

- Reconstrucción de la imagen multirresolución: construye la imagen multirresolución usando todos los datos disponibles en el instante de tiempo t, y los que sean necesarios del instante $t-1$.

- Estructura multifoveal $t-1$: la imagen multirresolución del instante $t-1$ de mantiene almacenada, ya que, si en un momento dado el paquete recibido no incluye toda la información de la escena por la ausencia de algún anillo, puede ser necesaria para obtener una imagen completa en el instante t.

- Control: este submódulo es simplemente un repetidor, que recibe el paquete de control del PC servidor y lo devuelve; de esta forma el PC servidor puede calcular el retardo del canal.

5 Estudio de resultados

Para mostrar el funcionamiento del sistema propuesto, se ha empleado con tres escenarios distintos. El método de compresión selectiva propuesto ofrece factores de compresión elevados cuando el área ocupada por los móviles representa un porcentaje bajo del total de la escena. En estos casos puede competir directamente con el método MPEG-II que se ha convertido en

un estándar comercial. Sin embargo, la ventaja que presenta nuestro método frente a aquel se refiere a que las regiones de interés seleccionadas son transmitidas con la resolución original. El método MPEG-II es una técnica global de compresión con pérdidas y, por tanto, deteriora por igual todas las regiones de la escena, con independencia de su relevancia. Así, las imágenes obtenidas en recepción pueden contener artefactos debidos a las pérdidas inducidas por el proceso de compresión y por la naturaleza orientada a bloque de este método, que divide la escena siguiendo un un enrejillado geométrico que no respeta la posición de las áreas de interés. En algunos casos este deterioro podría imposibilitar la aplicación de algoritmos de procesamiento en el extremo de recepción. Por ejemplo, si la aplicación consistiese en la interpretación de la matrícula de un vehículo, la distorsión introducida por la compresión MPEG-II podría dificultar la fase de reconocimiento de caracteres que podría confundir caracteres deteriorados. En este caso, el sistema propuesto aportaría una solución excelente ya que en recepción se podría disponer de una copia íntegra de la matrícula en su resolución original, y el resto de la escena, con resolución decreciente.

Debido a que la ventaja que presenta el sistema propuesto frente al clásico MPEG-II es más cualitativa que cuantitativa, sus diferencias se ilustran exclusivamente para la primera de las secuencias, mientras que el estudio de las dos restantes se centra en las posibilidades de compresión y de manejo de más de una región de interés simultáneamente.

5.1 Secuencia #1

La primera secuencia que se analiza consta de 249 fotogramas, apareciendo un único móvil entre los fotogramas 50 y 101. Como se muestra en la Fig. 5.7, en la que aparece parte de los resultados de la secuencia, el sistema realiza correctamente la detección del móvil en todo momento.

Una vez realizada la operación de segmentación y seguimiento, se procede a detectar la posición, en este caso, del único móvil presente, y construir la estructura foveal asociada. En las Figs. 5.8.a-d se muestran los tamaños, en Kbytes, de cada una de las regiones de resolución en las que se divide la imagen: fóvea, primer anillo, segundo anillo, y *waist*. En la Fig. 5.9.a se presenta una de las imágenes multirresolución transmitidas. Para comparar el algoritmo propuesto con el estándar MPEG-II, las Figs. 5.9.b-c muestran las ROIs recibidas usando el esquema de compresión propuesto y el algoritmo MPEG-II respectivamente (CBR, $ConstantBitRate$, 0.4 Mbps). Se puede apreciar como el esquema MPEG-II reduce la calidad de la ROI, mientras que el esquema propuesto no lo hace.

La Fig. 5.10.a muestra el volumen de datos de la imagen multirresolución para cada

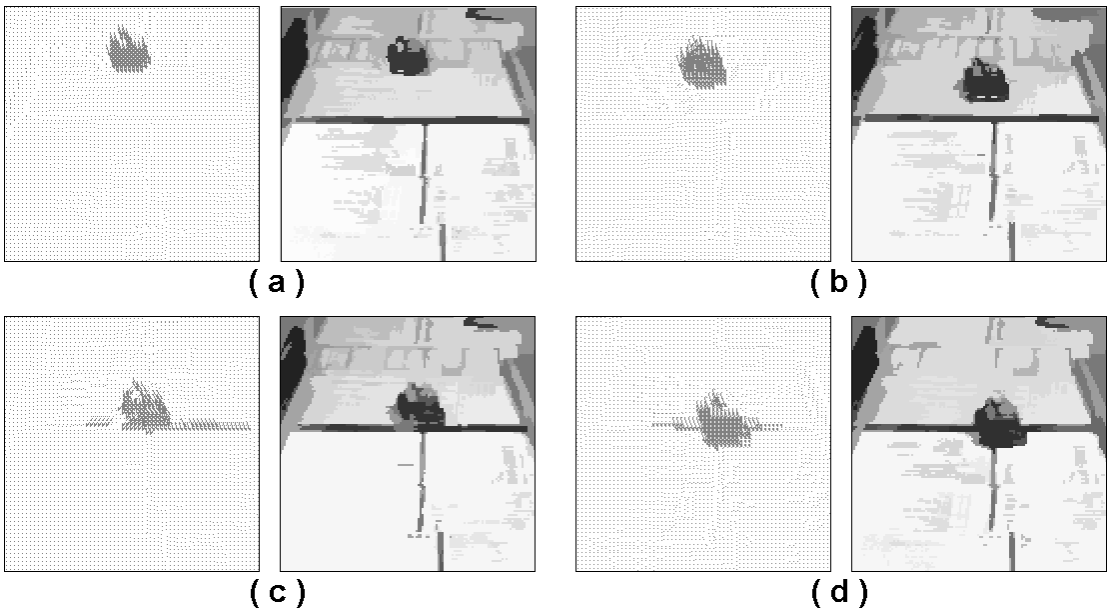


Figura 5.7: a-d) Vectores de desplazamiento detectados e imágenes segmentadas asociadas.

fotograma de la secuencia de vídeo. Mientras no existe móvil en la escena, la región de máxima resolución es sumamente pequeña y el tamaño de la imagen multirresolución es muy reducido. Cuando el móvil es detectado (a partir del fotograma #50), el área de la ROI cubre el móvil completamente y, por ello, el volumen de datos dependerá del tamaño de dicho móvil. Se puede observar como el volumen de datos supera el ancho de banda de transmisión en el intervalo en que el móvil alcanza su máximo tamaño.

Para adaptar el tamaño de los paquetes a transmitir a la estimación de retardo más reciente, los anillos de resolución exteriores dejan de transmitirse en función de dicho retardo. La Fig. 5.10.b muestra qué regiones de la imagen se incluyen en cada paquete transmitido. En cada instante, el ancho de banda disponible establece el máximo número de anillos de resolución que pueden ser transmitidos. Después de aplicar este algoritmo de control, el flujo de datos transmitidos para cada fotograma se adapta al ancho de banda disponible, como se puede observar en la Fig. 5.10.c. Finalmente, la Fig. 5.10.d muestra el factor de compresión obtenido al transmitir cada paquete, comparado con la imagen completa de resolución uniforme (256x256 píxeles). Se puede apreciar como dicho factor está siempre en el intervalo que va del 80 al 90 %.

5.2 Secuencia # 2

Esta segunda secuencia fué capturada empleando una cámara azimutal situada sobre una escalera mecánica. Al tratarse de una grabación realizada en un entorno real, se puede dar el caso de que

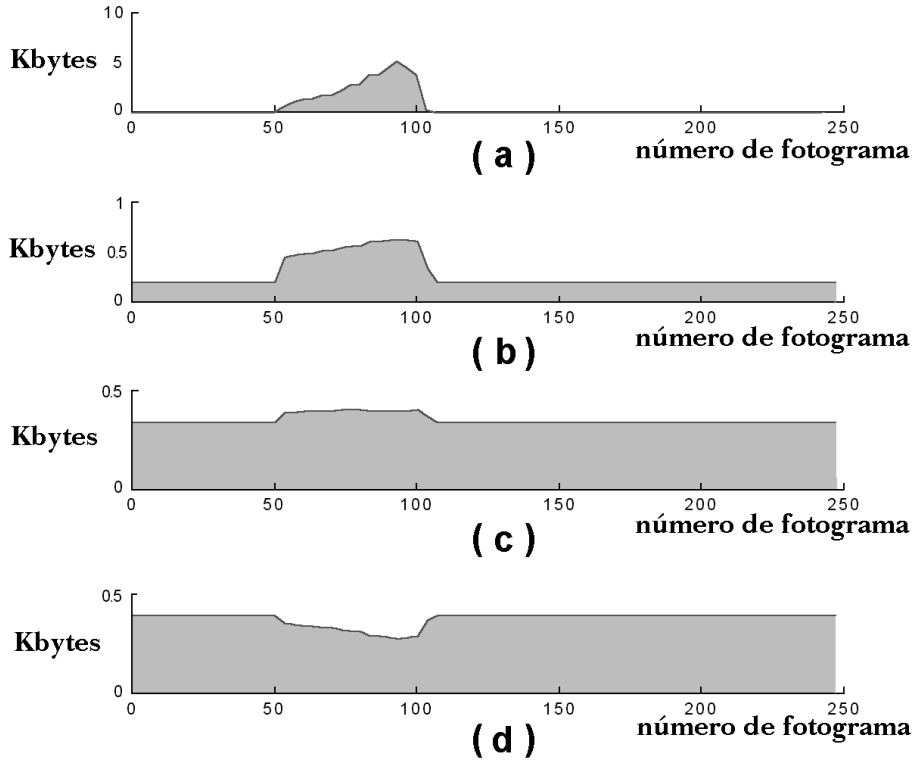


Figura 5.8: Tamaños de las distintas regiones de resolución obtenidas en el análisis de la secuencia # 1: a) tamaño de la fóvea; b) tamaño del primer anillo; c) tamaño del segundo anillo; y d) tamaño del *waist*

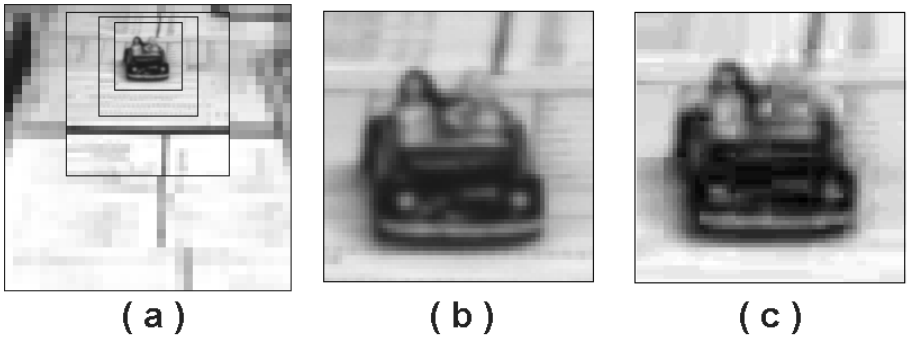


Figura 5.9: a) Imagen multirresolución en el instante t; b) aspecto de la ROI usando el algoritmo propuesto; y c) aspecto de la ROI usando MPEG-II.

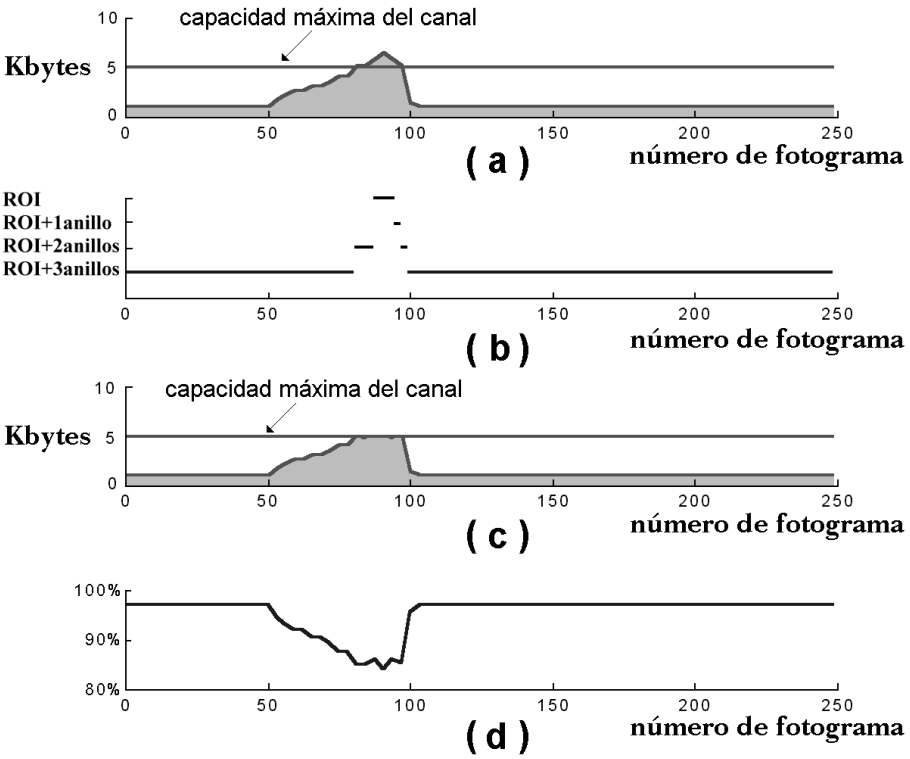


Figura 5.10: Análisis de la secuencia # 1: a) tamaño de la secuencia de vídeo multirresolución; b) esquema de transmisión; c) flujo de datos a transmitir; y d) tasa de compresión.

haya más de un móvil situado sobre dicha escalera. En el tramo de secuencia que se analiza, se ha detectado correctamente la entrada de dos móviles distintos, que aparecen simultáneamente en la escena. Esta situación requiere la transmisión simultánea de información acerca de dos regiones de interés distintas, por lo que se reducirá significativamente el factor de compresión.

En las Figs. 5.11.a-d se muestran algunos fotogramas de la secuencia, en los que se observa que, cuando están presentes los dos móviles, éstos ocupan gran parte de la escena. Se puede apreciar también como el primer móvil está formado por dos regiones de niveles de gris muy diferentes. Aunque durante el seguimiento de la secuencia estos dos objetos son generalmente detectados como uno sólo, en algunos fotogramas esporádicos estos objetos son separados, originando la detección de hasta tres móviles distintos en la escena. De cualquier forma, las regiones detectadas se corresponden con los objetos reales, con lo cual el PC cliente dispondrá de una versión válida de la escena que incluirá los objetos representativos con máxima resolución.

Esta secuencia es especialmente compleja debido a las propias características de la escena. Por ejemplo, son frecuentes los cambios de iluminación y la aparición de sombras y reflejos, que pueden distorsionar la correcta detección de los móviles. En torno al área que ocupa la escalera estos problemas son particularmente frecuentes, debido a su aspecto metálico. En el fotograma

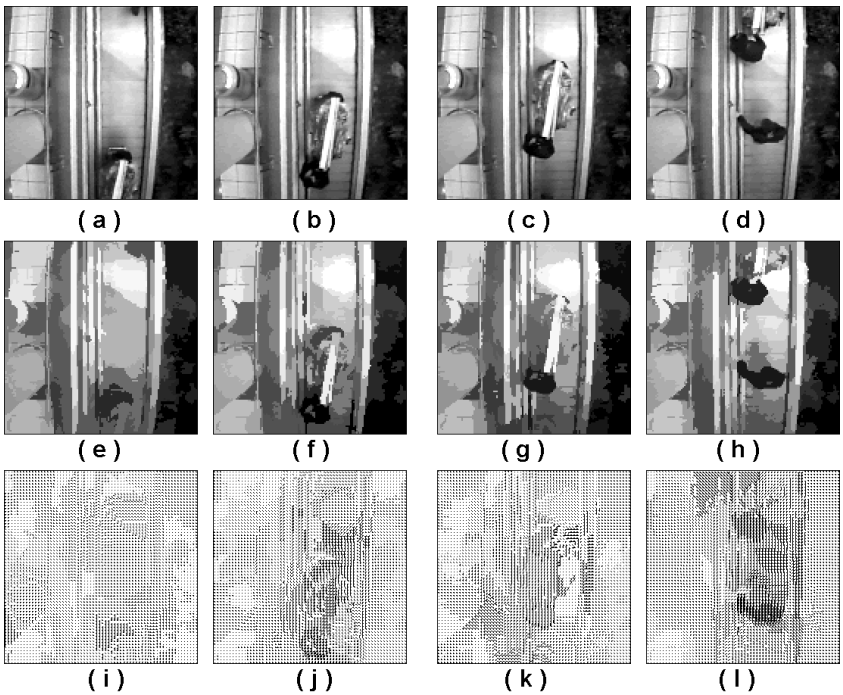


Figura 5.11: a-d) Fotogramas de la secuencia # 2; e-h) imágenes segmentadas asociadas a a-d); i-l) vectores de movimiento estimados para a-d).

que se muestra en la Fig. 5.12.a se han detectado hasta tres móviles distintos: el reflejo que incide en la escalera, la sombra que entra en escena y el móvil real. Tanto el reflejo como la sombra han supuesto cambios significativos respecto al fotograma anterior, por lo que han sido detectados como móviles que, en este caso, se solapan con el móvil real presente en la escena. El polígono multifóvea resultante se presenta en la Fig. 5.12.b. Aparte de esta situación anómala, en la mayoría de los fotogramas restantes, los procesos de detección y seguimiento de móviles se han llevado a cabo de forma correcta.

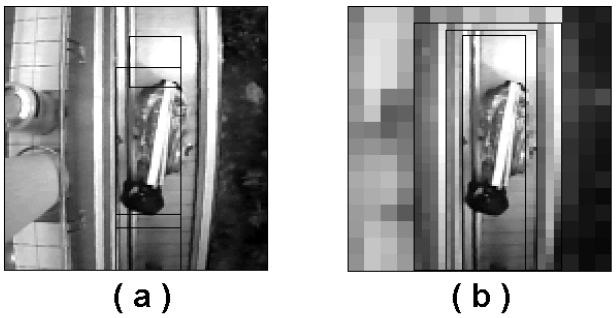


Figura 5.12: a) Fotograma de la secuencia # 2 y móviles detectados; b) polígono multifóvea generado a partir de las detecciones mostradas en a).

A continuación,, se analiza el comportamiento del sistema en sus fases de compresión y transmisión. En este caso, la secuencia tiene un tamaño de 238 fotogramas, apareciendo cada móvil en los fotogramas 20 y 123 respectivamente. En las Figs. 5.13.a-d se pueden observar los tamaños de cada una de las regiones de resolución en las que se divide la imagen. Se puede notar como, en esta secuencia, el tamaño de la región de alta resolución es relativamente mayor que en el caso analizado en el apartado 5.1, si se tiene en cuenta que en este caso las imágenes son de 128x128 píxeles, y contiene dos móviles, mientras que en la secuencia # 1 las imágenes eran de 256x256 píxeles. Evidentemente, este aumento en el tamaño ocupado por la ROI se debe tanto a la presencia de dos móviles como al mayor tamaño de éstos. En la Fig. 5.13 se puede apreciar también que el tamaño de todos los niveles aumenta cuando lo hace el número de móviles en la escena o el tamaño relativo de éstos. Este efecto podría reducirse si el sistema evitara enviar regiones redundantes contenidas simultáneamente en las distintas estructuras que se transmiten, una para cada objeto detectado. Por último, se aprecian dos picos de carga durante la transmisión de la secuencia, localizados en torno a los fotogramas 140 y 200, en los que el tamaño del volumen de datos aumenta de forma brusca. Esto se debe a que ciertos cambios de iluminación, debidos a sombras o reflejos, generan regiones que se funden con los objetos

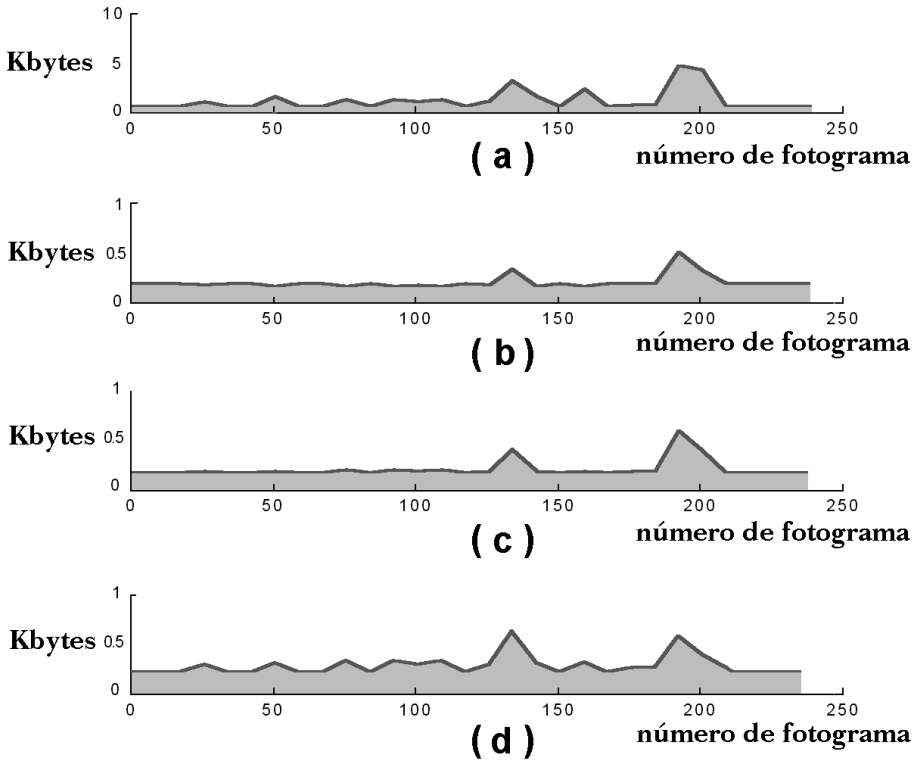


Figura 5.13: a-d) Tamaños de las regiones de resolución obtenidas en el análisis de la secuencia # 2.

detectados, aumentando su tamaño. Como se muestra en la Fig. 5.12, estos fenómenos ocurren debido a las condiciones del entorno, y dado que estos móviles irreales se funden a móviles reales, son difíciles de eliminar. De cualquier forma, como ya se comentó, estos aumentos en el tamaño de los objetos no afectan al posible procesado de imagen que se lleve a cabo en recepción.

Finalmente, en la Fig. 5.14.a se muestra el volumen del paquete transmitido para cada fotograma de la secuencia # 2. Se pueden notar los distintos tamaños que éste adopta en función del número de móviles que son detectados en la escena. Cuando ambos móviles son detectados, el área de la ROI los cubre completamente y, dado que ambos móviles pueden llegar a suponer más de un tercio de la imagen uniforme, el volumen de datos crece significativamente.

Se ha fijado artificialmente un valor de retardo en el canal, de forma que, cuando el volumen de información supera el ancho de banda estimado del canal, los anillos de resolución exteriores dejan automáticamente de transmitirse. La Fig. 5.14.b presenta qué regiones de la imagen se incluyen en cada paquete transmitido. Como se puede observar en la Fig. 5.14.c, después de aplicar este algoritmo de control, el flujo de datos transmitidos para cada fotograma se adapta al ancho de banda disponible. En la Fig. 5.14.d se muestra el factor de compresión obtenido al transmitir cada paquete, comparado con la imagen completa de resolución uniforme

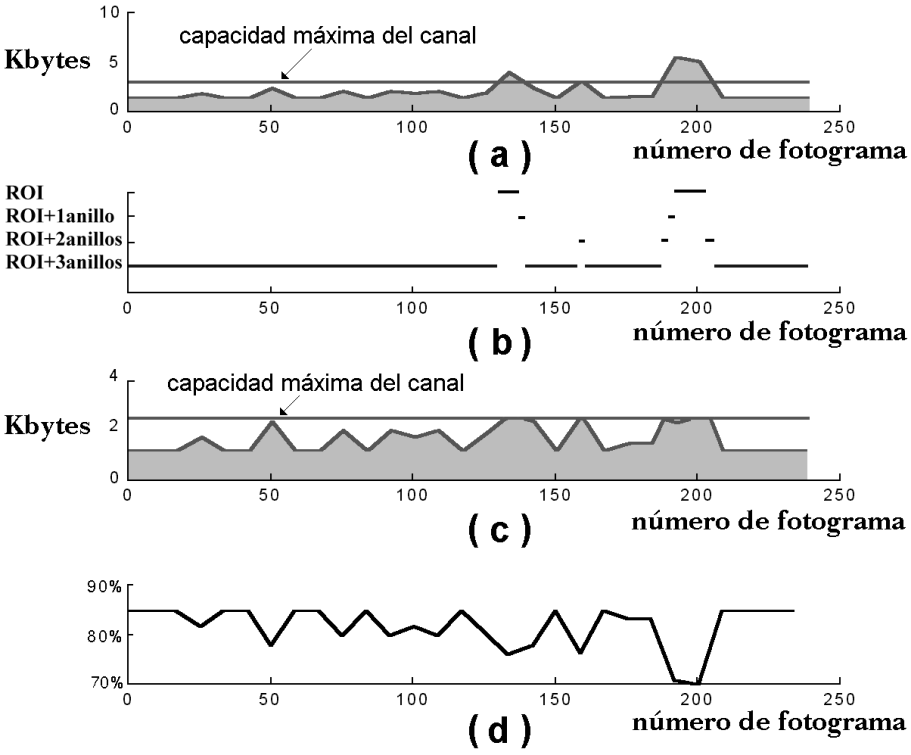


Figura 5.14: Análisis de la secuencia # 2: a) tamaño de la secuencia de vídeo multirresolución; b) esquema de transmisión; c) flujo de datos a transmitir; y d) tasa de compresión.

(128x128 píxeles). Se puede apreciar como, en este caso, dicho factor es menor que el extraído del análisis de la secuencia # 1, siendo, en cualquier caso, siempre superior al 70 %.

5.3 Secuencia # 3

La secuencia # 3 se ha rodado en interiores, empleándose tres móviles que aparecen en escena en instantes de tiempo distintos y se desplazan con velocidades igualmente diferentes. En la Fig. 5.15 se muestran ocho fotogramas equiespaciados de esta secuencia, en los que se puede apreciar el entorno en que se lleva a cabo el proceso.

El primer paso que debe realizar el sistema se lleva a cabo correctamente. En este caso, las principales fuentes de error son los cambios de iluminación, originados por el parpadeo apreciable de la luz artificial empleada y por el paso de distintas personas por el lugar en el que se realizó la grabación. Estos problemas, sin embargo, no afectan al proceso de detección de objetos de interés, aunque a veces originen falsas detecciones debidas a la aparición de sombras originadas por los obstáculos que aparecen en el entorno o agentes externos a la escena. Así, en la Fig. 5.16, que presenta las imágenes segmentadas y las estimaciones de los vectores de movimiento asociados a los fotogramas de la Fig. 5.15, se puede observar cómo los objetos de interés son fácilmente extraíbles a partir de esta información.

Una vez detectadas las regiones de interés, se procede a la construcción de la imagen multifoveal. En la Fig. 5.17 se pueden observar las imágenes multirresolución asociadas a los fotogramas de la Fig. 5.15. Se aprecia cómo en algunos fotogramas se han producido falsas detecciones. Este efecto es patente en particular en la Fig. 5.17.h, donde se han producido hasta dos falsas detecciones debidas a la aparición de sombras. De cualquier forma, estas sombras son

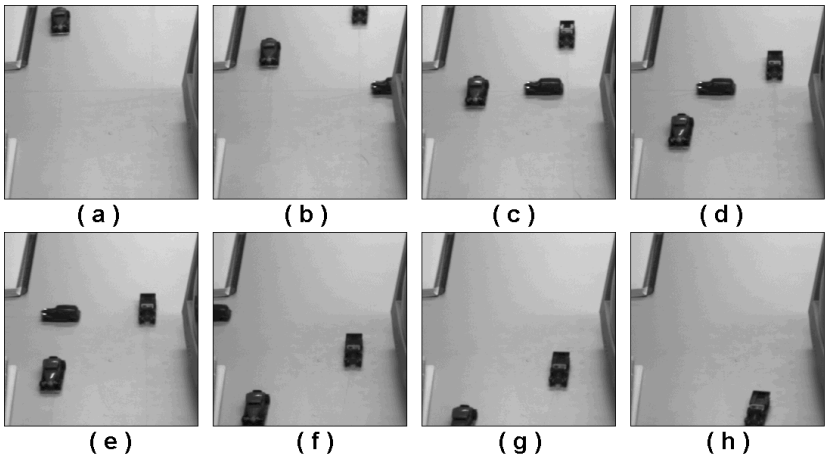


Figura 5.15: a-h) Fotogramas equiespaciados de la secuencia # 3.

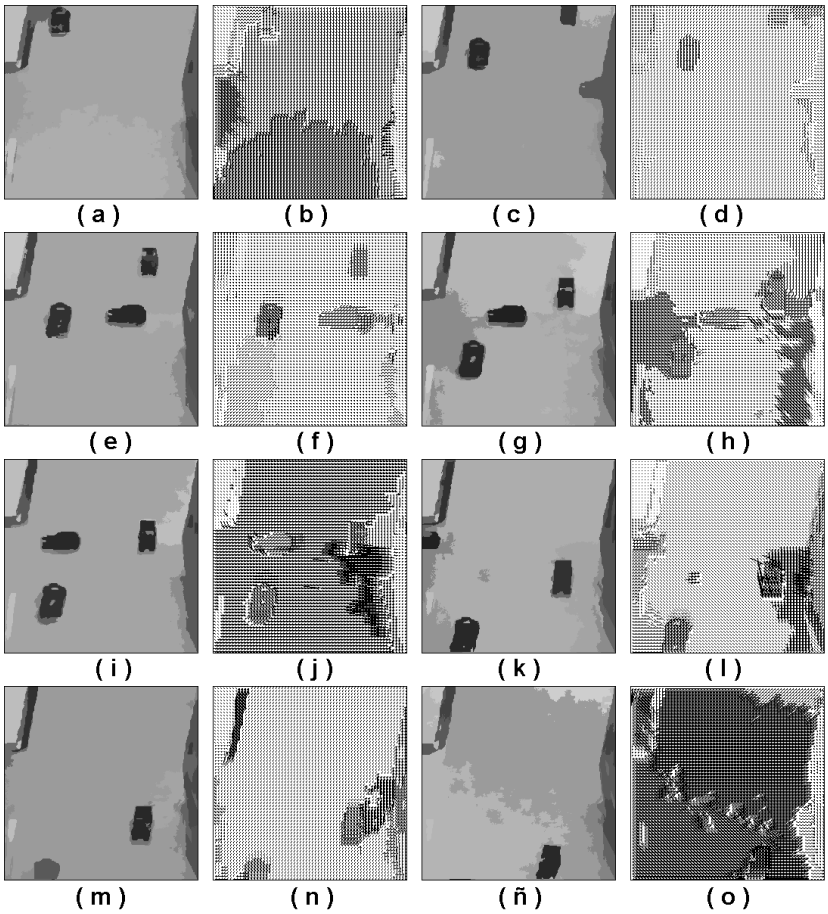


Figura 5.16: a-o) Imágenes segmentadas y vectores de movimiento estimados en los fotogramas de la secuencia # 3 mostrados en la Fig. 5.15.

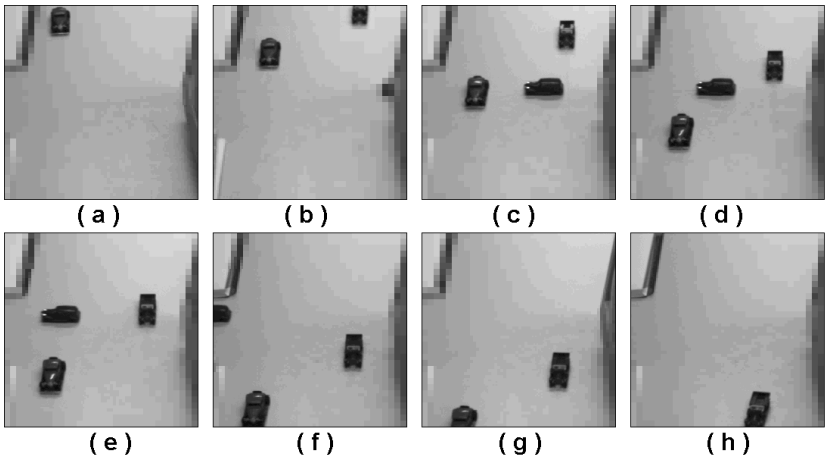


Figura 5.17: a-h) Imágenes multifoveales asociadas a los fotogramas de la secuencia # 3 mostrados en la Fig. 5.15.

producto de un cambio de iluminación global que, como se muestra en las Figs. 5.16.ñ-o, afecta a toda la imagen.

En las Figs. 5.18.a-d se muestran los tamaños de cada una de las regiones de resolución en las que se divide la imagen. En este caso, prácticamente todas las detecciones son correctas, por lo que el aumento apreciable en el tamaño de la ROI se debe a la aparición de uno o más móviles en escena.

El volumen de datos de la imagen multirresolución por fotograma se muestra en la Fig. 5.19.a. La Fig. 5.19.b muestra las regiones de la imagen incluidas en cada paquete transmitido en función de la estimación de anchura de canal. Como se puede observar en la Fig. 5.19.c, el seleccionar el volumen de datos de forma activa permite que el flujo de datos transmitidos para cada fotograma se adapte al ancho de banda disponible. Finalmente, en la Fig. 5.19.d se presenta el factor de compresión obtenido al transmitir cada paquete, comparado con la imagen completa de resolución uniforme (256x256 píxeles). En este caso, dicho factor es superior al 75%.

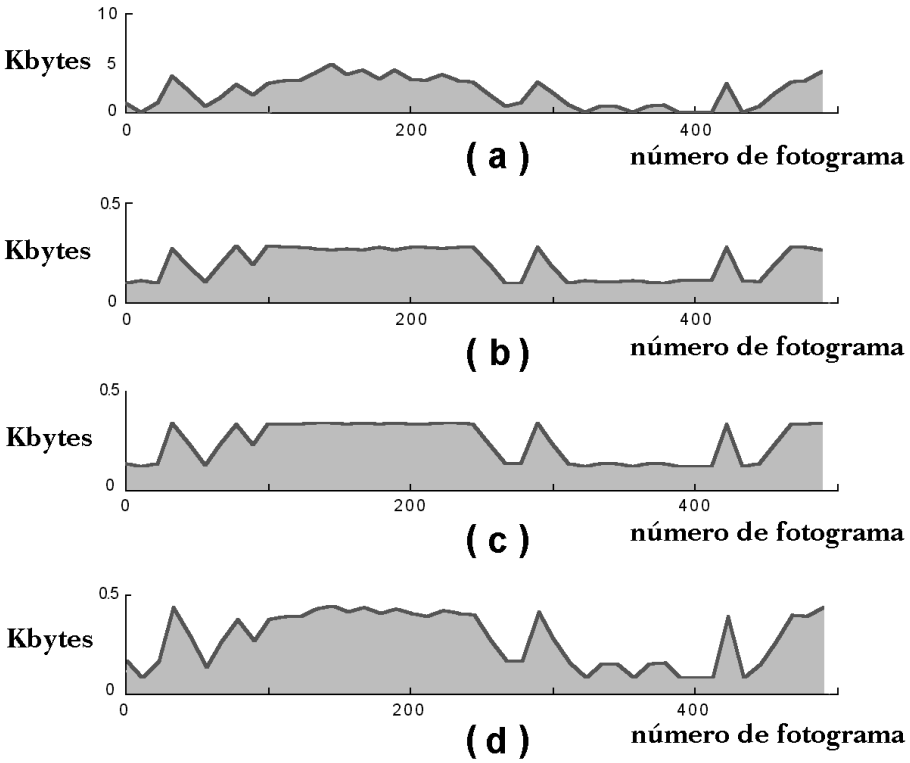


Figura 5.18: a-d) Tamaños de las regiones de resolución obtenidas en el análisis de la secuencia # 2.

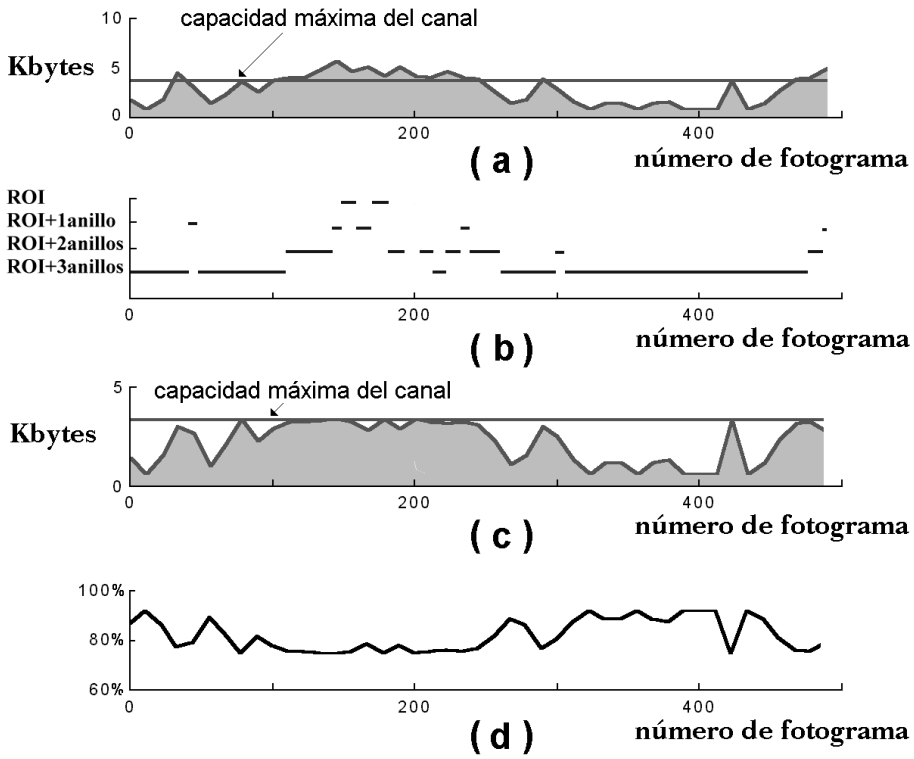


Figura 5.19: a) Tamaño de la secuencia de vídeo multirresolución; b) esquema de transmisión; c) flujo de datos a transmitir; y d) tasa de compresión.

6 Conclusiones

En este capítulo se ha aplicado el proceso de segmentación espacio-temporal propuesto en el capítulo 4 en un sistema de compresión de imagen. Evidentemente, el objetivo de dicho sistema era, más que constituir una versión definitiva de compresor, comprobar el correcto funcionamiento de la técnica de segmentación desarrollada. Por ello, los resultados obtenidos se han considerado correctos, aunque, evidentemente, el sistema en sí está abierto a numerosas modificaciones y mejoras. Entre estas mejoras, las más significativas serían: i) evitar la presencia de información redundante en la transmisión cuando hay más de un móvil en la escena; y ii) usar un algoritmo de compresión basado en técnicas predictivas para reducir el volumen de información de las distintas regiones de resolución. La primera mejora se refiere al mencionado problema que existe por la posible existencia de regiones de la escena que estén representadas por información proveniente de más de una de las estructuras que viajan por el canal cuando aparece más de un móvil. La segunda posible mejora tiene que ver con que actualmente se emplea una representación sin pérdidas para la transmisión de las ROIs, con lo cual, si el ancho máximo del canal es insuficiente, incluso para enviar sólo esta información, el sistema simplemente no transmite nada.

De cualquier forma, cabe destacar que el sistema implementado, aún sin disponer de un módulo de detección de objetos de alto nivel, es capaz de llevar a cabo esta operación y el seguimiento de distintos móviles sin apenas presentar fallos. Este hecho es especialmente destacable si se tiene en cuenta que los entornos de prueba no presentaban unas características de iluminación o captura controladas. En estas situaciones, la aparición de falsos objetos o la fusión de sombras o reflejos con los objetos reales es prácticamente inevitable si no se dispone de un módulo de alto nivel, que almacene la historia particular de cada objeto, según su trayectoria, su área o su nivel medio de gris.

Capítulo 6

Conclusiones y trabajo futuro

1 Conclusiones

En esta tesis se ha presentado un nuevo sistema de segmentación espacio-temporal para secuencias de vídeo. El objetivo principal del trabajo ha consistido en aportar nuevas ideas en este campo, enfocadas sobre todo al funcionamiento en tiempo real del mismo sobre procesadores no específicos.

El sistema se ha implementado sobre un Pentium II a 333 MHz y las secuencias de imágenes procesadas se han capturado con distintos tipos de cámara, incluyendo una cámara de videoconferencia, una cámara PULNIX digital de alta calidad y una cámara KP-D50 conectada a un *frame grabber* HECFG44. Así mismo, se ha recurrido a secuencias ya disponibles en Internet que se usan comúnmente como patrones para evaluación de distintos métodos de estimación de movimiento.

1.1 Aportaciones

El trabajo se ha dividido en bloques conforme a la evolución de las técnicas de segmentación propuestas. En cada uno de ellos se ha comentado en extensión las propuestas o mejoras implementadas, así como los resultados del proceso. Todo el desarrollo lleva a la conclusión más importante de este trabajo: las técnicas descritas de segmentación espacio-temporal de imágenes reales aportan una nueva e interesante alternativa a las ya existentes en la literatura. A continuación, se describen muy brevemente los resultados más destacables de las distintas partes de esta tesis:

- Segmentación espacial de un fotograma.

  - Estudio comparativo de técnicas clásicas de segmentación por nivel de gris.
  - Implementación del algoritmo de segmentación jerárquico por enlace adaptativo sobre imagen real.
  - Desarrollo de un conjunto de leyes de agrupación sobre estructuras jerárquicas para distinción fondo/objetos.

- Segmentación espacial en el tiempo.
  - Implementación de un método de segmentación por sustracción.
  - Desarrollo de un nuevo sistema de estimación del fondo con ventana temporal variable, capaz de recuperarse de oclusiones prolongadas.
  - Desarrollo de un conjunto de leyes de agrupación por etiquetado para distinción fondo/objetos.
  - Desarrollo de un sistema predictivo de seguimiento sencillo para la identificación de objetos a lo largo de la secuencia, capaz de recuperarse de fusiones transitorias de clases en contacto.

- Segmentación espacio-temporal.
  - Estudio comparativo de técnicas clásicas de segmentación espacio-temporales y temporales.
  - Desarrollo de una nueva estructura jerárquica para representar relaciones entre fotogramas consecutivos.
  - Desarrollo de un método de estabilización conjunto para dichas estructuras capaz de segmentar dos imágenes combinadamente.
  - Desarrollo de un método predictivo para mejorar el proceso de estabilización cuando aparecen en la escena regiones que presentan velocidades elevadas o cambios bruscos de las mismas.
  - Desarrollo de un sistema jerárquico no supervisado de fusión de clases para la correcta estimación del movimiento, usando los resultados de la segmentación propuesta.

El método de segmentación espacio-temporal propuesto se ha integrado en una aplicación real que permite probar sus ventajas a la hora de trabajar con secuencias. El sistema escogido es una aplicación de transmisión de vídeo comprimido a través de un enlace de ancho de banda reducido y variable. La tasa de compresión se selecciona en función del retardo disponible en cada momento y el tamaño de imagen se reduce mediante codificación con resolución no

uniforme y formato multifóvea reubicable de tamaño adaptativo sobre cada fotograma enviado. La segmentación espacio-temporal se utiliza en este caso para seleccionar las áreas de interés de la escena, que, siempre que el canal lo permita, se transmiten con alta resolución. El sistema se ha probado con un amplio conjunto de secuencias variadas y los resultados, si bien son mejorables adaptando específicamente el método de segmentación a la aplicación, han sido muy positivos en todos los casos.

1.2 Ventajas e inconvenientes del sistema de segmentación espacio-temporal propuesto

El método de segmentación espacio-temporal desarrollado en esta tesis presenta principalmente las siguientes ventajas:

- Robustez a cambios en las condiciones de captura.
- Robustez a cambios de iluminación en la escena.
- Posibilidad de trabajar con pocos fotogramas.
- Independencia: segmentación independiente de eventos posteriores a aquel que se está segmentando.
- Inmunidad: inmune al ruido en la secuencia.
- Posibilidad de trabajar con objetos que se desplazan rápidamente con respecto a la tasa de captura.
- Independiente a las restricciones relativas a la conservación de la intensidad o su gradiente en la escena.
- Capacidad de seguimiento de cualquier región a lo largo de la secuencia directamente a partir de la estructura de enlaces del sistema.
- Rápida recuperación de cambios bruscos de la velocidad de las regiones de la escena.
- Posibilidad de recuperación de oclusiones y descubrimientos.
- Disminución frente a otros métodos clásicos del tiempo de procesamiento de las imágenes para obtener los vectores de desplazamiento.

Es también necesario indicar las desventajas intrínsecas al sistema:

- La segmentación no es fiable hasta disponer de una estimación aproximada del movimiento de la escena, que puede obtenerse tras procesar dos o tres fotogramas.
- En ocasiones, al trabajar con un método espacio-temporal, las clases que entran en contacto y presentan el mismo nivel de gris pueden fundirse durante un tiempo.
- Por el mismo motivo, en escenas con muy poco contraste el método también puede fallar si no es capaz de discernir entre los niveles de gris de regiones contiguas.
- Las sombras, en tanto que presentan distinto color que el fondo y un movimiento homogéneo, pueden confundirse con objetos, no agregándose por tanto al fondo de la escena.
- Si los desplazamientos son del orden de unos pocos píxeles, los resultados empeoran.
- El método de predicción es extremadamente simple y, actualmente, sólo trabaja con dos fotogramas, a pesar de que se dispone de toda la historia cinética de cada región en fotogramas anteriores de la secuencia.
- Aparición de ruido en la estimación de movimiento debido a la presencia de oclusiones y descubrimientos.

2 Trabajo futuro

El trabajo futuro a realizar a partir de esta tesis estará principalmente enfocado a eliminar en la medida de lo posible los inconvenientes que presenta el sistema. A este respecto, las líneas de desarrollo actualmente previstas son las siguientes:

- Estudio de alternativas al nivel de gris como característica de un píxel cualquiera a la hora de construir la pirámide. En principio, resulta interesante utilizar un operador gradiente para caracterizar no sólo el píxel bajo estudio sino también su vecindario. Más adelante, en tanto que utilizar un vector en lugar de un escalar no cambia en absoluto la filosofía del método, puede pensarse en evaluar los tres componentes de color del píxel e incluso su microtextura.
- Desarrollo de un predictor sencillo de movimiento que tenga en cuenta la historia previa de la región. Dado que hasta ahora sólo se utilizan dos fotogramas para realizar dicha estimación, las regiones que no se desplazan de forma lineal pueden desestabilizar el sistema durante un par de fotogramas. Para evitar este efecto, y siempre que una región se mueva

de forma regular, puede estudiarse el desplazamiento de su centroide a lo largo de la secuencia para efectuar una estimación más precisa del movimiento de ésta.

- Estudio de la variación de masa de las regiones en movimiento a efectos de mitigar los errores producidos por oclusiones y desplazamientos. Si se estima la magnitud de los cambios en tamaño que puede presentar una región determinada en función de su evolución a lo largo de la secuencia, es posible predecir el efecto de oclusión o descubrimiento que tendrá sobre el fondo de la escena y corregir el desplazamiento de los centroides de las regiones implicadas a partir de dicho efecto. De esta forma sería posible eliminar el ruido inducido por él sobre la estimación de movimiento.

Adicionalmente, cabe destacar que el principal problema del método sigue siendo un elevado tiempo de proceso ya que, a pesar de mejorar los resultados de gran parte de los métodos clásicos en cuanto a velocidad por su sencillez y por el hecho de trabajar en multirresolución, sobre un PC estándar el sistema sigue siendo insuficiente para generar imágenes segmentadas, incluso para tasas de captura de videoconferencia. A este respecto cabe señalar como línea futura de desarrollo la implementación del mismo método de segmentación sobre imágenes de resolución no uniforme. Además, conviene remarcar que los métodos de segmentación jerárquica mediante enlazado adaptativo ya han sido satisfactoriamente implementados en hardware para una única imagen [Coslado et al. 1999]. Es, pues, deseable, estudiar la viabilidad de la implementación del método en hardware, ya que se prevé que sería relativamente sencillo partiendo del esquema anterior. Obviamente, de esta forma se mejoraría la eficiencia temporal del método propuesto, ampliando los campos de aplicación en los que su uso sería practicable.

MENU SALIR

Bibliografía

[Aach y Kaup 1995] Aach, T. y Kaup, A., "Bayesian algorithms for adaptive change detection in images sequences using Markov random fields", *Signal Process. Image Commun.*, **7**, pp. 147-160, 1995.

[Aloimonos y Bandopadhay 1988] Aloimonos, Y. y Bandopadhay, I.W., "Active Vision", *Int. Journal of Computer Vision*, **2**(1), pp. 333-356, 1988.

[Aloimonos 1997] Aloimonos, Y., *Visual navigation: from biological systems to unmanned ground vehicles*, Lawrence Erlbaum As.: Hillsdale, NJ-USA, 1997.

[Amamoto y Matsumoto 1997] Amamoto, N. y Matsumoto, K., "Obstruction detector by environmental adaptive background image updating", en ERTICO (Ed.) *Proc. of the 4th World Congress on Intelligent Transport Systems*, **4**, pp. 1-7, Traffic Tech. Int., Berlín-Alemania, 1997.

[Anandan 1987] Anandan, P., *Measuring visual motion from image sequences*, Ph.D. dissertation, COINS TR 87-21, Univ. of Massachusetts, Amherst, EEUU 1987.

[Anandan 1989] Anandan, P., "A computational framework and an algorithm for the measurement of visual motion", *Int. J. Computer Vision*, **2**, pp. 283-310, 1989.

[Araújo et al. 1998] Araújo, H.J., Dias, J., Batista, J. y Peixoto, P., "Active vision for autonomous systems", en A.T. Almeida & O. Kathib (Eds.) *Autonomous robotic systems*, pp. 20-49, Springer-Verlag: Berlín-Alemania, 1998.

[Arrebola 1998] Arrebola, F., *Sistema de visión basado en imágenes multirresolución de fóvea desplazable*, Tesis Doctoral, Dpto. Tecnología Electrónica, Universidad de Málaga, Málaga-España 1998.

[Baek et al. 1996] Baek, Y., Oh, H. y Lee, H., "An efficient block-matching criterion for motion estimation and its VLSI implementation", *IEEE Trans. on Consumer Electronics*, **42** (4), 885-892, 1996.

[Ballard y Brown 1982] Ballard, D. y Brown, C.M., *Computer vision*, Prentice Hall: Englewood Cliffs, NJ-USA, 1982.

[Bandera 2000] Bandera, A., *Sistema avanzado de reconocimiento autónomo de objetos*, Tesis Doctoral, Dpto. Tecnología Electrónica, Universidad de Málaga, Málaga-España 2000.

[Bandera et al. 2000a] Bandera, A., Urdiales, C., Herreros, J. L., y Sandoval, F., "Implementación de un mecanismo atencional mixto sobre un agente autonomo móvil", *Actas del XV Simposium Nacional de la Unión Científica Internacional de Radio (URSI'00)*, pp. 323-324, Zaragoza-España, 2000.

[Bandera et al. 2000b] Bandera, A., Urdiales, C., Rodríguez, J.A. y Sandoval, F. "Corner detection techniques for planar images", en S.G. Pandalai (Ed.), *Recent Research on Pattern Recognition* **1**, pp. 137-150, Transworld Research Network: Kerala-India, 2000.

[Bandera 1994] Bandera, C., *Structures and algorithms for foveal machine vision*, Amherst Systems, Technical Report, Buffalo, NY-USA, 1994.

[Bandera y Scott 1989] Bandera, C. y Scott, P., "Foveal Machine Vision Systems", *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 596-599, Cambridge-USA, 1989.

[Barron et al. 1994] Barron, J.L., Fleet, D.J. y Beauchemin, S.S., "Systems and experiment performance of optical flow techniques", *International Journal of Computer Vision*, **12** (1), pp. 43-77, 1994.

[Bartrum y Crow 1984] Bartrum, R.J. y Crow, H.C., "Transillumination light scanning to diagnose breast cancer: a feasibility study", *American Journal of Radiology*, **142**, pp. 409-414, 1984.

[Beaudet 1978] Beaudet, P.R., "Rotationally invariant image operators", *Proc. of the 4th Int. Conf. Pattern Recognition*, pp. 579-583, Tokyo-Japón, 1978.

[Beauvais y Lakshmanan 2000] Beauvais, M. y Lakshmanan, S., "CLARK: a heterogeneous sensor fusion method for finding lanes and obstacles", *Image and Vision Computing*, **18**(5), pp. 397-413, 2000.

[Bellegarda et al. 1993] Bellegarda, E.J., Bellegarda, J.R., Nahamoo, D. y Nathan, K.S., "A probabilistic framework for on-line handwritting recognition", *Proc. of the 3rd Int. Workshop on Frontiers in Handwritting Recognition*, Buffalo-EEUU, pp. 225-234, 1993.

[Beucher 1990] Beucher, S., "Segmentation tools in mathematical morphology", en P.D. Gader, Ed., *Image algebra and morphological image processing*, pp. 70-84, SPIE 1350, 1990.

[Bhaskaran y Konstantinides 1997] Bhaskaran, V. y Konstantinides, K., *Image and video compression standards: Algorithms and architectures*, Kluwer: Boston, MA-USA, 1997.

[Bhattacharya y Majumder 2000] Bhattacharya, M. y Majumder, D.D., "Registration of CT and MR images of Alzheimer's patient: a shape theoretic approach", *Pattern Recognition Letters*, **21**(6-7), pp. 531-548, 2000.

[Bianco y Cassinis 1996] Bianco, G. y Cassinis, R., "Multi-strategic approach for robot path planning", *Proc. of Eurobot'96*, pp. 108-115, Kaiserslautern-Alemania, 1996.

[Bischel 1994] Bischel, M., "Segmenting simply connected moving objects in a static scene", *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, **16**, pp. 1138-1142, 1994.

[Blake y Yuille 1992] Blake, A. y Yuille, A., *Active Vision*, MIT Press: Cambridge, 1992.

[Bourne et al. 1981] Bourne, J. R., Jagannathan, V., Hamel, B., Jansen, B.H., Ward, J.W., Hughes, J.R., y Erwin, C.W., "Evaluation of a Syntactic Pattern Recognition Approach to Quantitative Electroencephalographic Analysis", *Electroenceph. clin. Neurophysiol.*, **52**, pp. 57-64, 1981.

[Bouthemy y Lalande 1993] Bouthemy, B. y Lalande, P., "Recovery of moving object masks in an image sequence using local spatiotemporal contextual information", *Optical Engineering*, **32**, pp. 1025-1212, 1993.

[Bradbury 1983] Bradbury, S., "Commercial image analyzers and the characterization of microscope images", *J. Microsc.*, **131**, pp. 203-210, 1983.

[Brofferio et al. 1990] Brofferio, S., Carnimeo, L., Comunale, D. y Mastronardi, G., "A background updating algorithm for moving object scenes", en V. Capellini, Ed., *Time-varying image processing and moving object recognition, 2*, pp. 289-296, Elsevier Publishers B.V.: Amsterdam-Holanda, 1990.

[Brooks 1986] Brooks, A., "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, **RA-2** (1), pp. 14-23, 1986.

[Buhnmann et al. 1995] Buhnmann, J., Burgard, W., Cremers, A., Fox, D., Hoffman, T., Schneider, F., Strikos, J. y Thrun, S., "The Mobile Robot Rhino", *AI Magazine*, **16**(1), pp. 31-38, 1995.

[Burt et al. 1981a] Burt, P., Hong, T. y Rosenfeld, A., "Image smoothing based on neighbor linking", *IEEE Trans. on Systems, Man and Cybernetics*, **11** (12), pp. 769-780, 1981.

[Burt et al. 1981b] Burt, P., Hong, T. y Rosenfeld, A., "Segmentation and estimation on image region properties through cooperative hierarchical computation", *IEEE Trans. on Systems, Man and Cybernetics*, **11** (12), pp. 802-809, 1981.

[Burt 1983] Burt,P.J. y Adelson, E.H., "The Laplacian pyramid as a compact image code", *IEEE Trans. on Communications*, **31**, pp.532-540, 1983.

[Camacho et al. 1996] Camacho, P., Arrebola, F. y Sandoval, F., "Shifted fovea multiresolution geometries", *Proc. of the IEEE Int. Conf. on Image Processing*, **1**, pp. 307-310, Lausanne-Suiza, 1996.

[Camacho et al. 1997] Camacho, P., Arrebola, F. y Sandoval, F., "Adaptive fovea structures for space-variant sensors", en Alberto del Bimbo (Ed.), *Proc. of the Int. Conf. on Image Analysis and Processing, LNCS 1310*, **1**, pp. 422-429, Springer-Verlag, Berlin-Alemania, 1997.

[Camacho et al. 1998] Camacho, P., Arrebola, F., y Sandoval, F., "Multirresolution sensors with adaptive structure", *Proc. of the 24th Annual Conf. of the IEEE Industrial Electronics Society*, **2**, pp. 1230-1235, Aquisgrán-Alemania, 1998.

[Canny 1983] Canny, J. F., *Finding edges and lines in images*, Master's Thesis, MIT, Cambridge-USA, 1983.

[Capurro et al. 1997] Capurro, C., Panerai, F. y Sandini, G., "Dynamic Vergence using Log-polar Images", *Int. Journal of Computer Vision*, **24** (1), pp. 79-94, 1997.

[Castagno et al. 1998] Castagno, R., Ebrahimi, T. y Kunt, M., "Video segmentation based on multiple features for interactive multimedia applications", *IEEE Transactions on Circuits and Systems for Video Technology*, **8** (5), pp. 562-571, 1998.

[Charters y Graham 1999] Charters, G.C. y Graham, J., "Trainable grey-level models for disentangling overlapping chromosomes", *Pattern Recognition*, **32** (8), pp. 1335-1349, 1999.

[Chiou y Hwang 1995] Chiou, G.I. y Hwang, J.N. "A Neural Network-Based Stochastic Active Contour Model NNS-Snake for Contour Finding of Distinct Features", *IEEE Transactions on Image Processing*, **4**(10), pp. 1407-1416, 1995.

[Cibulskis y Dryer 1984] Cibulskis, C. y Dryer, R., "Node linking strategies in pyramids for image segmentation", en A. Rosenfeld, Ed., *Multiresolution Image Processing and Analisis*, pp. 109-120, Springer-Verlag: Berlín-Alemania, 1984.

[Cohen y Medioni 1998] Cohen, I. y Medioni, G., "Detection and tracking of objects in airborne video imagery", *IEEE Workshop on Interpretation of Visual Motion*, Santa Barbara-USA, 1998.

[Connell y Jain 2001] Connell, S.D. y Jain, A.K., "Template-based online character recognition", *Pattern Recognition*, **34**, pp. 1-14, 2001.

[Coren y Girgus 1978] Coren, S. y Girgus, J.S., *Seeing is Deceiving: The Psychology of Visual Illusions*, Lawrence Erlbaum Associates: Hillsdale, NJ-USA, 1978.

[Coslado et al. 1999] Coslado, F., Camacho, P., Gonález, M., Arrebola, F. y Sandoval, F., "VLSI implementation of a foveal polygon segmentation algorithm", *Proc. of the 10th International Conference on Image Analysis and Processing*, pp. 185-190, Venecia-Italia, 1999.

[Costa y Sandler 1993] Costa, L. y Sandler, M., "Efective detection of bar segments with Hough transform", *CVGIP: Image Understanding*, **55** (3), pp. 180-191, 1993.

[Davis et al. 1997] Davis, L., Chellapa, R., Yacoob, Y. y Zheng, Q., "Visual survillance and monitoring of human and vehicle activity", *Proc. of the DARPA Image Understanding Workshop*, pp. 19-27, New Orleans, LA-USA, 1997.

[Ebrahimi y Kunt 1998] Ebrahimi, T. y Kunt, M., "Visual data compression for multimedia applications", *Proc. IEEE*, **86**, pp. 1109-1125, 1998.

[Ebrahimi 2000] Ebrahimi, T., "Object-based video coding", in A. Bovik, Ed., *Handbook of image and video processing*, pp. 585-595, Academic Press: San Diego, CA-USA, 2000.

[Faugeras 1993] Faugeras, O., *Three-dimensional computer vision: a geometric viewpoint*, MIT Press: Cambridge, MA, 1993.

[Friedman y Russell 1997] Friedman, N. y Russell, S., "Image segmentation in video sequence: A probabilistic approach", *Proc. of the 13th Conf. on Uncertainty in Artificial Intelligence*, Providence, RI-USA, 1997.

[Geman y Geman 1984] Geman, S. y Geman, D., "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images", *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, **6**, pp. 721-741, 1984.

[Gibson et al. 1959] Gibson, E.J., Gibson, J.J., Smith, O.W. y Flock, H., "Motion parallax as a determinant of perceived depth", *Journal of Experimental Psychology*, **58**, pp. 40-51, 1959.

[Gilbert 1972] Gilbert, P., "Iterative methods for the reconstruction of three-dimensional objects from their projections", *Journal of Theoretical Biology*, **36**, pp. 105-117, 1972.

[Gonzalez y Wintz 1987] Gonzalez, R.C. y Wintz, P., *Digital image processing*, Addison-Wesley Publishing: Reading, MA-USA, 1987.

[Granlund 1999] Granlund, G.H., "The complexity of vision", *Signal Processing*, 74 (1), pp. 101-126, 1999.

[Griffioen et al. 1995] Griffioen, J., Seales, W.B. y Yavatkar, R., "Quality-of-Service Guarantees for Wireless Computer Vision", *Proc. of the Int. Soc. for Optical Engr.: Sensor Fusion and Networked Robotics VIII (SPIE'95)*, **2589**, pp. 122-133, Philadelphia, PA-USA, 1995.

[Grossberg y Kuperstein 1986] Grossberg, S. y Kuperstein, M., *Neural dynamics of adaptive sensory-motor control. Ballistic eye movements*, Amsterdam: Holland, North-Holland Elsevier, 1986.

[Hall 1971] Hall, E.L., "A survey of preprocessing and feature extraction techniques for radiographic images", *IEEE Trans. Comput.*, **20** (9), pp. 1032-1044, 1971.

[Hanek y Schmitt 2000] Hanek, R. y Schmitt, T., "Vision-Based Localization and Data Fusion in a System of Cooperating Mobile Robots", *Proc. of the 2000 IEEE/RSJ Int. Conf. on Intell. Robots and Systems*, pp. 1199-1204, Takamatsu-Japón, 2000.

[Hartley 1985] Hartley, R., "Segmentation of optical flow fields by pyramid linking", *Pattern Recognition Letters*, **3**, pp. 253-262, 1985.

[He et al. 1995] He, Z., Liou, M., Chan C. y Li, R., "Efficient architectures for the new three-step search algorithm", *Proc. of Midwest Symposium on Circuits and Systems*, **2**, pp. 1228-1235, Rio de Janeiro , Brasil, 1995.

[Hepplewhite y Stonham 1997] Hepplewhite, L. y Stonham, T. J., "N-tuple texture recognition and the zero crossing sketch", *Electronics Letters*, **33**(1), pp. 45-46, 1997.

[Hird y Wilson 1989] Hird, J.A. y Wilson, D.F., "A comparison of target detection and segmentation techniques", en Alan H. Lettington, Ed., *Optical Systems for Space and Defense*, SPIE **1191**, pp. 375-386, 1989.

[Horn y Schunk 1981] Horn, B.K.P. y Schunk, B.G., "Determining optical flow", *Artificial Intelligence*, **17**, pp. 185-204, 1981.

[Horn y Weldon 1988] Horn, B.K.P. y Weldon, Jr., E. J., "Direct methods for recovering motion", *International Journal of Computer Vision*, **2**, pp. 51-76, 1988.

[Hötter et al. 1996] Hötter, M., Mester, R. y Meyer, M., "Detection of moving objects using a robust displacement estimation including a statistical error analysis", *Proc. of the International Conference on Pattern Recognition*, pp. 249-255, Viena-Austria, 1996.

[Howarth y Buxton 1996] Howarth, R. y Buxton, H., "Visual surveillance monitoring and watching", *Proc. of the 4th Europan Conf. on Computer Vision (ECCV)*, **2**, pp. 321-334, Oxford-Inglaterra, 1996.

[Huwer y Niemann 2000] Huwer, S. y Niemann, H., "Adaptive change detection for real-time surveillance applications", *Proc. of the 3rd IEEE Int. Workshop on Visual Surveillance (VS)*, pp. 37-45, Dublin-Irlanda, 2000.

[Iiyama et al. 2000] Iiyama, M., Kameda, K. y Minoh, M., "Estimation of the Location of Joint Points of Human Body from Successive Volume Data", *Proc. of the 15th International Conference on Pattern Recognition (IAPR'00)*, **3**, pp. 699-702, Barcelona-España, 2000.

[Isard y Blake 1996] Isard, M. y Blake, A., "Contour tracking by stochastic propagation of conditional density", *Proc. of the European Conference of Computer Vision*, pp. 343-356, Cambridge, Inglaterra-UK, 1996.

[ISO/IEC 1996] ISO/IEC JTC1/SC29/WG11, "Description of MPEG-4", N1410, Octubre 1996.

[ISO/IEC 1998] ISO/IEC JTC1/SC29/WG11, "MPEG-4 overview", N2323, Julio 1998.

[Jarvis 1980] Jarvis, J.F., "A method for automating the visual inspection of printed wiring boards", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, pp. 77-82, 1980.

[Johnson y Hebert 1998] Johnson, A.E. y Hebert, M., "Surface matching for object recognition in complex three-dimensional scenes", *Image and Vision Computing*, **16** (9-10), pp. 635-651, 1998.

[Johnson-Laird 1980] Johson-Laird, P.N., "Mental models in cognitive science", *Cognitive Science*, **4** (1), pp. 71-115, 1980.

[Jolion y Rosenfeld 1994] Jolion, J.M. y Rosenfeld, A., *A pyramid framework for early vision*, Kluwer: Boston, MA-USA, 1994.

[Kanade et al. 1997] Kanade, T., Collins, R.T., Lipton, A., Anandan, P., Burt, P. y Wixson, L., "Cooperative multisensor video surveillance", *Proc. of the DARPA Image Understanding Workshop*, pp. 3-10, New Orleans, LA-USA, 1997.

[Karmann et al. 1990] Karmann, K., Brandt, A. y Gerl, R., "Moving object segmentation based on adaptive reference images", *Signal Processing: Theories and Applications*, **5**, pp. 951-954, 1990.

[Kaup y Aach 1994] Kaup, A. y Aach, T., "Efficient prediction of uncovered background in interframe coding using spatial extrapolation", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing: Image and Multidimensional Signal Processing*, **5**, pp. 501-504, Adelaide-Sudáfrica, 1994.

[Kilger 1992] Kilger, M., "A shadow handler in a video-based real-time traffic monitoring system", *IEEE Workshop on Applications of Computer Vision*, pp. 1060-1066, Palm Springs-USA, 1992.

[Kim et al. 1999] Kim, T., Cho, T., Moon, Y.S. y Park, S.H., "Visual inspection system for the classification of solder joints", *Pattern Recognition*, **32** (4), pp. 565-575, 1999.

[Kim y Kim 2000] Kim, M. y Kim, J., "Moving video object segmentation using statistical hypothesis testing", *Electronics Letters*, **36** (2), pp. 128-129, 2000.

[Kohler 1947] Kohler, W., *Gestalt psychology: an introduction to new concepts in modern psychology*, Liveright Publishing Co.: New York, 1947.

[Koller et al. 1994] Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B. y Russell, S., "Towards robust automatic traffic scene analysis in real-time", *Proc. of the 12th Int. Conf. on Pattern Recognition (ICPR'94)*, pp. 126-131, Jerusalén-Israel, 1994.

[Konrad y Dufaux 1998] Konrad, J. y Dufaux, F., "Improved global motion estimation for N3", *Tech. Rep. MPEG97/-M3096*, ISO/IEC JTC1/SC29/WG11, Feb. 1998.

[Konrad 2000] Konrad, J., "Motion detection and estimation", en Al Bovik, Ed., *Handbook of image and video processing*, pp. 207-225, Academic Press: San Diego, CA, 2000.

[Koo y Jeong 2000] Koo, H.S. y Jeong, C.S., "A Stereo Matching Algorithm Using Adaptive Window and Search Range", *Proc. of the PRICAI 2000*, pp. 803, Melbourne-Australia, 2000.

[Kosslyn y Schwartz 1977] Kosslyn, S.M. y Schwartz, S.P., "A simulation of visual imagery", *Cognitive Science*, **1** (3), pp. 265-295, 1977.

[Kunt et al. 1985] Kunt, M., Ikonomopoulos, A. y Kocher, M., "Second generation image coding techniques", *Proc. IEEE*, **73**, pp. 549-675, 1985.

[Lucas y Kanade 1981] Lucas, B. y Kanade, T., "An iterative image registration technique with an application to stereo vision", *Proc. of the DARPA Image Understanding Workshop*, pp.121-130, Washington, DC-USA, 1981.

[Luthon et al. 1999] Luthon, F., Caplier, A. y Liévin, M., "Spatiotemporal MRF approach to video segmentation: Application to motion detection and lip segmentation", *Signal Processing*, **76**, pp. 61-80, 1999.

[Mahzoun et al. 1999] Mahzoun, M.R., Kim, J., Sawazaki, S., Okazaki, K. y Tamura, S., "A scaled multigrid optical flow algorithm based on the least RMS error between real and estimated second images", *Pattern Recognition*, **32**, pp. 657-670, 1999.

[Matsumoto et al. 1981] Matsumoto, K., Naka, M. y Yamamoto, H., "A new clustering method for Landsat images using local maximums of a multidimensional histogram", *Proc. of the Symposium on Machine Processing Remotely Sensed Data*, pp. 321-326, Purdue, IN-USA, 1981.

[Mehnert y Jackway 1997] Mehnert, A. y Jackway, P., "An improved seeded region growing algorithm", *Pattern Recognition Letters*, **18**, pp. 1065-1071, 1997.

[Menegaz et al. 1999] Menegaz, G., Vaerman, V. y Thiran, J.P., "Object-based coding of volumetric medical data", *Proc. of the 6th International Conference on Image Processing (ICIP'99)*, **3**, pp. 920-924, Kobe-Japón, 1999.

[Mohinder et al. 1993] Mohinder, S., Grewal, S. y Andrews, A., *Kalman filtering: Theory and practice*, Prentice Hall: Englewoods Cliffs, NJ-USA, 1993.

[Moreno et al. 1996] Moreno, L., Salichs, M., Gachet, D., Pimentel, J., Arroyo, F. y Gonzalo, A., *Neural networks for robotic control*, Hellis Horwood Ltd.: Hertfordshire, Inglaterra-UK, 1996.

[Nagel 1983] Nagel, H.H., "Displacement vectors derived from second order intensity variations in image sequences", *Comput. Graph. Image Process*, **21**, pp. 85-117, 1983.

[Odobez y Bouthemy 1995] Odobez, J.M. y Bouthemy, P., "Robust multiresolution estimation of parametric motion models", *Visual Communication and Image Representation*, pp. 348-365, 1995.

[Palmer 1975] Palmer, S.E., "Visual perception and world knowledge: notes on a model of sensory-cognitive interaction", en D.A. Norman, D.E. Rumerlhart y el LNR Research Group, Eds., *Exploration in cognition*, W.H. Freeman: San Francisco, CA-USA, 1975.

[Patel y Stonham 1992] Patel, D. y Stonham, T. J., "Texture image classification and segmentation using RANK-order clustering", *IEEE 11th International Conference on Pattern Recognition (IAPR'92)*, **3**, pp. 92-95, The Hague-The Netherlands, 1992.

[Pitas 1993] Pitas, I., *Digital image processing algorithms*, Prentice Hall: New York, 1993.

[Rao 1996] Rao, K., "Shape description of curved 3d objects for aerial surveillance", *Proc. of the ARPA Image Understanding Workshop*, pp. 1065-1076, Palm Springs, CA-USA, 1996.

[Reusens et al. 1997] Reusens, E., Ebrahimi, T., y Kunt, M., "Dynamic coding of visual information", *IEEE Trans. Circuits Syst. Video Technol.*, **7**, pp. 489-500, 1997.

[Ridder et al. 1995] Ridder, C., Munkelt, O. y Kirchner, H., "Adaptive background estimation and foreground detection using kalman-filtering", en O. Kaynak, M. Özkan, N. Bekiroglu e I. Tunay (Eds.) *Proc. of the Int. Conf. on recent Advances in Mechatronics (ICRAM)*, pp. 193-199, Estambul-Turquía, 1995.

[Rodríguez et al. 1998] Rodríguez, J. A., Arrebola, F., Camacho, P., Sandoval, F., "Control de la mirada con imágenes multirresolución de fóvea desplazable", *Actas del XIII Simposium Nacional de la Unión Científica Internacional de Radio (URSI'98)*, pp. 295-296, Pamplona-España, 1998.

[Rodríguez et al. 1999] Rodríguez, J. A., Ruiz, A., Bandera, A. y Sandoval, F., "Cálculo del índice de disparidad con geometrías cartesiano-exponenciales", *Actas del XIV Simposium Nacional de la Unión Científica Internacional de Radio (URSI'99)*, pp. 276-277, Santiago de Compostela-España, 1999.

[Rodríguez et al. 2000a] Rodríguez, J. A., Urdiales, C., Camacho, P. y Sandoval, F., "Detección jerárquica de móviles sobre geometrías de fóvea adaptativa", *Revista Electrónica de Visión por Computador (REVC), http://www.cvc.uab.es/revc*, **3**, 2000.

[Rodríguez et al. 2000b] Rodríguez, J. A., García, A., Urdiales, C., Bandera, A. y Sandoval, F., "Detección y seguimiento de objetos en entornos dinámicos mediante estimación predictiva

del flujo óptico", *Actas del XV Simposium Nacional de la Unión Científica Internacional de Radio (URSI'00)*, pp. 495-496, Zaragoza-España, 2000.

[Rodríguez et al. 2000c] Rodríguez, J. A., Bandera, A., Urdiales, C., Arrebola, F. y Sandoval, F., *Estudio de viabilidad de un sistema de contaje de personas utilizando técnicas de visión*, Informe Interno, Dpto. de Tecnología Electrónica, Universidad de Málaga, Febrero-2000.

[Rodríguez et al. 2001a] Rodríguez, J. A., Urdiales, C., Bandera, A. y Sandoval, F., "Hierarchical object tracking by means of adaptive stabilization of 3D structures", *aceptado en SNRFAI-2001.*

[Rodríguez et al. 2001b] Rodríguez, J. A., Urdiales, C., Bandera, A. y Sandoval, F., "A multiresolution spatio-temporal segmentation technique for video sequences based on pyramidal structures", *enviado a Pattern Recognition Letters.*

[Rodríguez et al. 2001c] Rodríguez, J. A., Urdiales, C., Bandera, A. y Sandoval, F., "Object tracking in video sequences by using adaptively stabilized combined pyramids", *enviado a International Conference on Intelligent Robots and Systems, IROS'2001.*

[Rodríguez et al. 2001d] Rodríguez, J. A., Urdiales, C., Bandera, A. y Sandoval, F., "VBR video transmission by means of multifoveal geometries", *enviado a International Journal of Imaging Systems and Technology.*

[Rosen y Nitzan 1977] Rosen, C.A. y Nitzan, D., "Use of sensors in programmable automation", *Computer*, pp. 12-23, 1977.

[Simoncelli et al. 1991] Simoncelli, E.P., Adelson, E.H. y Heeger, D.J., "Probability distribution of optical flow", *Proc. of the International Conference on Computer Vision and Pattern Recognition*, pp. 310-315, Maui-Hawaii, 1991.

[Singh 1990] Singh, A., "An estimation-theoretic framework for image-flow computation", *Proc. 3rd Int. Conf. Computer Vision*, pp. 168-177, Osaka-Japón, 1990.

[Smith y Brady 1995] Smith, S.M. y Brady, J.M., "ASSET-2: Real-time motion segmentation and shape tracking", *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, **17** (8), pp. 814-820, 1995.

[Smith y Brady 1997] Smith, S.M. y Brady, J.M., "SUSAN - a new approach to low level image processing", *International Journal of Computer Vision*, **23** (1), pp. 45-78, 1997.

[Sobottka et al. 1997] Sobottka, K., Jiang, X.Y. y Bunke, H., "Spatiotemporal Segmentation of Range Image Sequences into Planar Surfaces for Collision Avoidance", *Proc. of the Int. Symposium on Automotive Technology and Automation (ISATA) Special Session on Machine Vision and Intelligent Vehicles and Autonomous Robots*, pp. 69-76, Florence-Italia, 1997.

[Srinivasan y Rao 1985] Srinivasan, R. y Rao, K., "Predictive coding based on efficient motion estimation", *IEEE Trans. Commun.*, **33**, pp. 888-896, 1985.

[Stark et al. 1981] Stark, H., Woods, J.W., Paul, I. y Hingorani, R., "Direct Fourier reconstruction in computer tomography", *IEEE Trans. Acoust. Speech Signal Proc.*, **29**, pp. 237-244, 1981.

[Suen et al. 1980] Suen, C.Y., Berthod, M. y Mori, S., "Automatic recognition of handprinted characters - the state of art", *Proc. IEEE*, **68**, pp. 469-487, 1980.

[Suen y Mori 1982] Suen, C.Y. y Mori, S., "Standardization and automatic recognition of handprinted characters", en C.Y. Suen & R. De Mori (Eds.) *Computer Analysis and Perception: Vol. 1, Visual signals*, pp. 41-53, CRC Press: Florida-USA, 1982.

[Suen 1986] Suen, C.Y., "Character recognition by computer and applications", en T.Y. Young & K. Fu (Eds.) *Handbook of pattern recognition and image processing*, pp. 569-586, Academic Pres: San Diego, CA-USA, 1986.

[Takeuchi et al. 1995] Takeuchi, Y., Wang, Z., Ohnishi, N. y Sugie, N., "Real Time Visual Tracking System Mimicking Saccadic Movements", *Proc. of the Second Asian Conference on Computer Vision*, **1**, pp.131-135, Singapore, 1995.

[Tanimoto y Pavlidis 1975] Tanimoto, S. y Pavlidis, T., "A hierarchical data structure for picture processing", *Computer Graphics and Image Processing*, **4**, pp. 104-119, 1975.

[Tekalp 1995] Tekalp, A., *Digital video processing*, Prentice Hall: Englewoods Cliffs, NJ-USA, 1995.

[Terzopoulos et al. 1987] Terzopoulos, D., Platt, J.,Barr, A. y Fleischer, K., "Elastically deformable models", *Computer Graphics*, **21** (4), pp. 205-214, 1987.

[Tistarelli y Sandini 1993] Tistarelli, M. y Sandini, G., "On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow", *IEEE Trans. Pattern Analysis and Machine Intelligence*, textbf15 (4), pp. 401-410, April 1993.

[Torres y Kunt 1996] Torres, L. y Kunt, M., *Video coding: the second generation approach*, Kluwer: Boston, MA-USA, 1996.

[Toth et al. 2000a] Toth, D., Aach, T. y Metzler, V., "Bayesian spatio-temporal motion detection under varying illumination", *Proc. of the European Signal Processing Conf. (EUSIPCO'00)*, pp. 1085-1088, Tampere-Finlandia, 2000.

[Toth et al. 2000b] Toth, D., Aach, T. y Metzler, V., "Illumination-invariant change detection", *Proc. of the 4th IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 3-7, Austin, TX-USA, 2000.

[Tziritas y Labit 1994] Tziritas, G. y Labit, C., *Motion analysis for image sequence coding*, Elsevier, The Netherlands, 1994.

[Uenohara y Kanade 1998] Uenohara, M. y Kanade, T., "Optimal approximation of uniformly rotated images: relationship between Karhunen-Loeve expansion and discrete cosine transform", *IEEE Trans. on Image Processing*, **7** (1), pp. 116-119, 1998.

[Urdiales 1999] Urdiales, C., *Arquitectura de control de movimiento y exploración para un agente autónomo*, Tesis Doctoral, Dpto. Tecnología Electrónica, Universidad de Málaga, Málaga-España, 1999.

[Urdiales et al. 2000] Urdiales, C., Rodríguez, J. A., Bandera, A., Sandoval, F., "Video flow active control by means of adaptive shifted foveal geometries", *Intelligent Robots and Computer Vision XIX: Algorithms, Techniques and Active Vision*, David P. Casasent, Editor, Proceedings of SPIE, **4197**, pp. 229-240, Boston, MA-USA, 2000.

[Verri y Poggio 1987] Verri, A. y Poggio, T., "Against quantitative optical flow", *Proc. of the 1st International Conference on Computer Vision*, pp. 171-180, Londres, Inglaterra-UK, 1987.

[Wolberg et al. 1994] Wolberg, W., Street, W., y Mangasarian, O., "Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates", *Cancer Letters*, **77**, pp. 163-171, 1994.

[Yanowitz y Bruckstein 1989] Yanowitz, S.D. y Bruckstein, A.M., "A new method for image segmentation", *Computer Vision, Graphics and Image Processing*, **46**, pp. 82-95, 1989.

[Zhou y Zhou 1999] Zhou, WD. y Zhou, J., "The development of EEG real-time monitoring system", *Shandong Journal of Biomedical Engineering*, **18**(3), pp. 22-26, 1999.

MENU SALIR

Apéndice A

Visión foveal

1 Geometrías multirresolución

El principal problema de la mayoría de los sistemas de visión artificial radica en la necesidad de disponer simultáneamente de un amplio campo de visión, una alta resolución y unos tiempos de proceso reducidos. La realidad es que una imagen de resolución uniforme de estas características presenta un volumen de datos muy elevado que hace difícil su transmisión o procesado en tiempo real. La solución a este problema puede encontrarse en el uso de imágenes de resolución no uniforme, donde sólo las áreas de interés aparecen en alta resolución, mientras que el resto del campo de visión presenta un perfil de resolución decreciente en función de su distancia a las mencionadas áreas de interés. Dado que la posición de esas áreas es desconocida a priori, es necesario emplear un algoritmo de detección adecuado para poder construir la imagen de resolución no uniforme. Este proceso de detección de áreas de interés y estructuración del campo visual supone utilizar la visión de forma activa, de manera que se dispone de un proceso selectivo de adquisición de información en función de la tarea que se va a llevar a cabo.

1.1 Topologías foveales de resolución no uniforme

Si bien inicialmente las imágenes de resolución no uniforme no tienen porqué tener un perfil estructurado, a la hora de representarlas y procesarlas resulta mucho más conveniente definir unas geometrías adecuadas. Las geometrías de resolución no uniforme más utilizadas son las denominadas geometrías foveales, que emulan la fisiología del ojo de algunos seres vivos avanzados. En estas geometrías existe un área de máxima resolución ubicada en el centro de la imagen que se denomina fóvea, y en las áreas periféricas de la imagen la resolución disminuye de forma progresiva en función de su distancia a la fóvea. Esta configuración está muy limitada por su

propia naturaleza, ya que para captar un objeto de interés a máxima resolución es necesario reposicionar el sensor de forma que el objeto quede en el centro del campo de visión, lo que si bien en un sistema biológico puede hacerse de forma inmediata, en un sistema mecánico es mucho más lento e impreciso. Sin embargo, si se combina una geometría foveal con la posibilidad de mover la fóvea sin reposicionar el sensor, se puede adquirir la información deseada de forma rápida y eficaz con un máximo nivel de detalle, independientemente de su ubicación en el campo de visión.

Tradicionalmente, las geometrías foveales se han dividido en dos configuraciones básicas, la log-polar (Fig. A.1.a) y la cartesiano exponencial (Fig. A.1.b). Los sistemas biológicos obedecen a configuraciones log-polares y hoy en día existen multitud de aplicaciones basadas en imágenes de este tipo, como pueden ser el cálculo del flujo óptico [Tistarelli y Sandini 1993] o la estimación de vergencia [Capurro et al. 1997]. Estos métodos, siendo computacionalmente costosos, pueden ejecutarse en tiempo real gracias a la reducción del volumen de datos que permiten este tipo de geometrías. La principal desventaja de las imágenes log-polares está en su estructura, que no es compatible con la mayoría de los algoritmos de proceso existentes, por lo que se requiere desarrollar herramientas específicas para casi cualquier tipo de aplicación (uso de filtros, operaciones binarias, e incluso, su propia presentación en pantalla). Las geometrías cartesiano-exponenciales ofrecen una alternativa válida a las log-polares, ya que presentan las siguientes ventajas:

- No contienen puntos ciegos en el origen de la estructura, como ocurre con las log-polares, debido a la singularidad en el origen de su sistema de referencia.
- Si se desea construir un sensor físico (en lugar de confeccionar las imágenes por $software$),

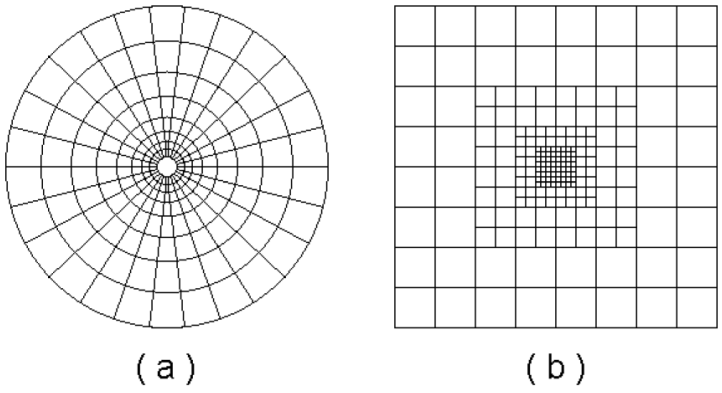


Figura A.1: Geometrías foveales: a) geometría log-polar; b) geometría cartesiano-exponencial

su implementación en VLSI es geométricamente mucho más simple que la implementación de las geometrías log-polares.

- La mayoría de los algoritmos y sistemas de proceso se han adaptado a las geometrías cartesianas, ya que las imágenes de resolución uniforme se ajustan bien a este perfil. Así, la transformación de estos algoritmos para trabajar con imágenes foveales cartesiano-exponenciales es casi inmediata.

2 Geometrías foveales cartesiano-exponenciales

Las geometrías cartesiano-exponenciales fueron propuestas originalmente por C. Bandera y P. Scott en [Bandera y Scott 1989]. En su propuesta, la rejilla tenía una distribución simétrica donde un área central de máxima resolución, denominada fóvea, estaba circundada por una serie de anillos, cada uno de los cuales presentaba a su vez resolución uniforme y progresivamente menor conforme se alejaban del centro. Esta geometría es sumamente fácil de implementar a partir de las tecnologías actuales y soporta estructuras de almacenamiento y procesamiento jerárquico. Su principal desventaja radica en la discontinuidad del perfil de disminución de la resolución, pero en la mayoría de las aplicaciones este hecho no influye negativamente en los resultados. Para definir este tipo de geometría cartesiano-exponencial son necesarios dos parámetros: i) m, el número de anillos alrededor de la fóvea; ii) d, el factor de subdivisión o número de subanillos dentro de cada anillo.

En la geometría de la Fig. A.1.b estos parámetros son igual a 3 y a 2 respectivamente. Es importante notar que existen estructuras de datos jerárquicas para el almacenamiento de estas imágenes que permiten ejecutar algoritmos de procesado clásicos sobre ellas sin necesidad de alterarlos significativamente [Arrebola 1998]. El factor de compresión (FC) que presentan las imágenes cartesiano-exponenciales es igual a:

$$FC = 4m + (1/(4 + 3m)) \tag{A.1}$$

y, tal como se aprecia en la expresión, depende únicamente de m. De esta forma, conforme aumenta el número de anillos de resolución de la imagen, el volumen de datos de la imagen disminuye. No obstante, esa disminución no es lineal, sino que a partir de un determinado valor de m dicho volumen alcanza un mínimo, y prácticamente deja de disminuir. La Fig. A.2 muestra cómo a partir de seis anillos no se consigue reducción significativa.

El principal problema de la geometría cartesiano-exponencial clásica viene dado por el hecho de que la zona de alta resolución está indefectiblemente ligada al centro de la imagen. Dado

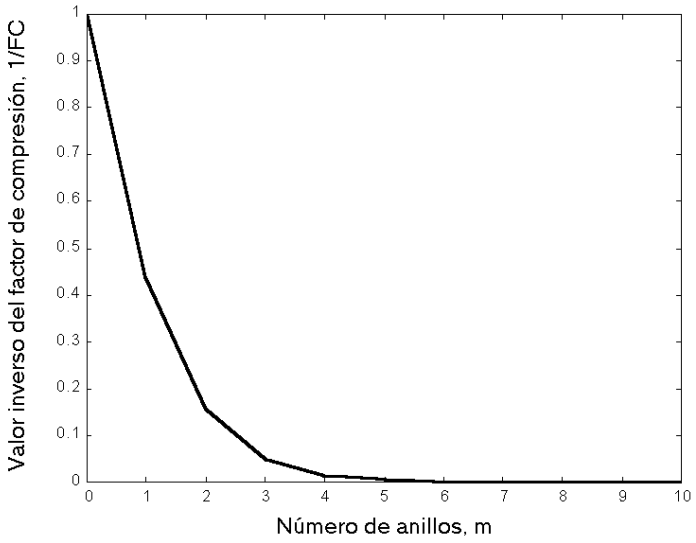


Figura A.2: Valor inverso del factor de compresión en función del número de anillos de una geometría multirresolución de fóvea centrada

que las áreas de interés de la imagen pueden estar ubicadas en cualquier posición, es necesario reposicionar la cámara para poder captarlas al máximo nivel de detalle. Esta operación requiere tiempo y puede no tener la precisión deseada, por lo que el procesado de la imagen se complica. Sin embargo, una ventaja adicional de las geometrías cartesiano-exponenciales está en que la posición del área de alta resolución es fácilmente reconfigurable. Así, en lugar de desplazar la cámara, es posible relocalizar la fóvea de forma rápida y eficaz si se desarrollan nuevas geometrías con este fin.

2.1 Geometría multiresolución de fóvea desplazable

La Fig. A.3.b muestra una geometría de fóvea desplazable (GMFD), donde puede observarse cómo la fóvea se reubica en una porción del campo de visión distinta a la zona central a la que estaba ligada en la topología anterior. Estas geometrías fueron desarrolladas en [Camacho et al. 1996] y requieren dos parámetros adicionales para su definición:

- S_h: el factor de desplazamiento horizontal de cada anillo de resolución i con respecto a una fóvea centrada.

- S_v: el factor de desplazamiento vertical de cada anillo de resolución i con respecto a una fóvea centrada.

Estos parámetros están expresados en términos de celdas de resolución homogénea, también conocidos como *rexels*.

Las GMFD presentan el mismo número de celdas, campo de visión y factor de compresión que las geometrías cartesiano-exponenciales clásicas. Como ventaja se debe destacar que ahora la fóvea no está limitada a la zona central de la escena, pero de cualquier forma, la fóvea no puede ubicarse en cualquier posición de la imagen, porque el valor máximo de S_h y S_v es igual al factor de subdivisión d de la geometría. De esta forma, sólo existen $(2d+1)^2$ posiciones permitidas. Dado que la resolución cambia en potencias de 2, puede deducirse fácilmente que el desplazamiento mínimo de la fóvea expresado en píxeles es igual a $2(2m-1)$.

2.2 Geometría multiresolución de fóvea desplazable de movimiento generalizado

La Fig. A.3.c presenta una geometría de fóvea desplazable de movimiento extendido (GMFD-MG). En este caso, la fóvea puede ubicarse en cualquier posición del campo de visión porque se permiten desplazamientos distintos entre los sucesivos anillos de resolución. Además, el desplazamiento final de la fóvea se define mediante dos vectores, SH y SV, donde cada par de elementos, SH_k y SV_k, definen el desplazamiento relativo del anillo k frente al anillo $k+1$. Las principales ventajas de la GMFD-MG frente a la GMFD son las siguientes:

- La fóvea puede ubicarse en cualquier punto del campo de visión cuyas coordenadas sean múltiplo de 2. Por tanto, el desplazamiento mínimo de la fóvea ya no depende de m y el máximo error de posicionamiento es igual a 1 píxel.
- El número de posibles posiciones de la fóvea es ahora igual a $((W-4d)/2)^2$, siendo W la anchura del campo de visión en píxeles.
- Todas las regiones de la imagen pueden ahora ser examinadas a alta resolución.

2.3 Geometría multirresolución de fóvea desplazable y tamaño adaptativo

El principal problema de las geometrías mencionadas anteriormente reside en que la fóvea es cuadrada y su dimensión depende de los parámetros que definen la geometría. Esta circunstancia limita la capacidad de cubrir objetos irregulares o de dimensiones mayores a la fóvea. En el caso de que la región de interés sea más pequeña que la fóvea, se estarán sobredimensionando los recursos empleados, al abarcarse una zona mayor de lo necesario. Si, por el contrario, la

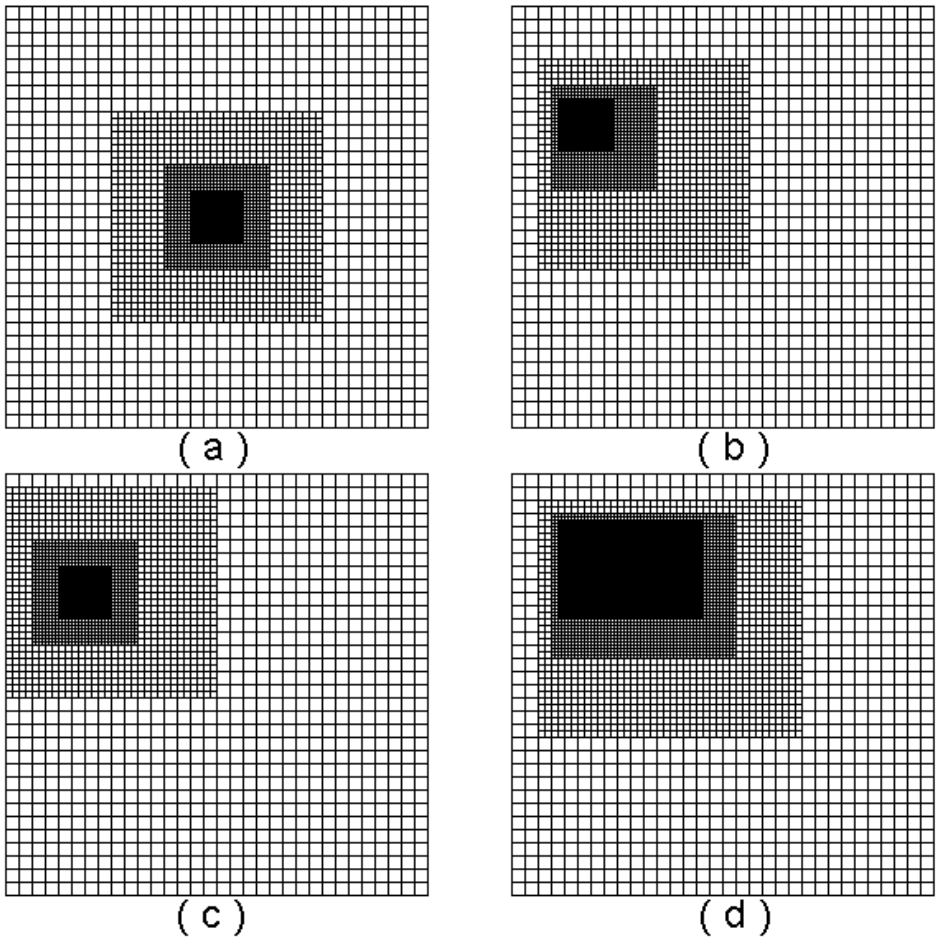


Figura A.3: Geometrías cartesiano-exponenciales: a) topología clásica; b) GMFD; c) GMFD de movimiento generalizado; y d) GMFD de fóvea de tamaño adaptativo.

región de interés es mayor que la fóvea, se requiere una serie de sucesivos reposicionamientos de ésta para observar la totalidad de la región en estudio con resolución máxima, con el inconveniente de que en ningún momento se dispone de una versión completa de la misma. Para solucionar este problema, las geometrías multirresolución de fóvea desplazable y tamaño adaptativo [Camacho et al. 1997] permiten el ajuste de la región explorada con resolución máxima en cada instante. La Fig. A.3.d muestra un ejemplo de estas geometrías. Ahora son necesarios cinco parámetros para definir esta estructura:

- m, el número de anillos de resolución.
- L_d y R_d, los factores de desplazamiento a la izquierda y a la derecha de cada anillo, respectivamente.
- T_d y B_d, los factores de desplazamiento arriba y abajo de cada anillo, respectivamente.

En la geometría presentada en la Fig. A.3.d, m vale 3, y L_d, R_d, T_d y B_d son igual a 10, 2, 2 y 12 respectivamente. La principal ventaja de esta geometría se encuentra en que cualquier objeto puede ser representado con resolución máxima independientemente de sus dimensiones y posición respecto a la cámara, siempre y cuando esté completamente contenido en la escena. Ahora, el factor de compresión depende básicamente del tamaño de la fóvea y del número de anillos, m.

2.4 Geometría multirresolución multifóvea

Las geometrías multirresolución de fóvea desplazable y tamaño adaptativo están especialmente indicadas en situaciones en las que se requiere explorar una escena posicionando de forma óptima la máxima resolución sobre la región que presenta mayor interés. Pero existen muchas aplicaciones en las que se encuentran simultáneamente en escena más de una región de interés. En estos casos, el uso de este tipo de geometrías estaría exclusivamente indicado si la aplicación concreta en que se utilizan permite realizar sucesivos reposicionamientos de la fóvea sobre las regiones de interés.

Para solventar esta limitación se han desarrollado las geometrías multirresolución multifóvea [Camacho et al. 1998]. Son básicamente iguales a las geometrías multirresolución de fóvea desplazable y tamaño adaptativo, con la diferencia de presentar más de una fóvea. Cada fóvea da lugar a una distribución particular de los distintos anillos de resolución, como muestran las Figs. A.4.a-c. La geometría multifóvea resultante es la superposición de las distintas estructuras a que dan lugar cada una de las fóveas de esta geometría, como muestra la Fig. A.4.d.

Esta geometría queda determinada por el conjunto de parámetros que definen cada una de las GMFD de tamaño adaptativo por separado, descartando la información redundante resultante de la superposición de las distintas imágenes foveales.

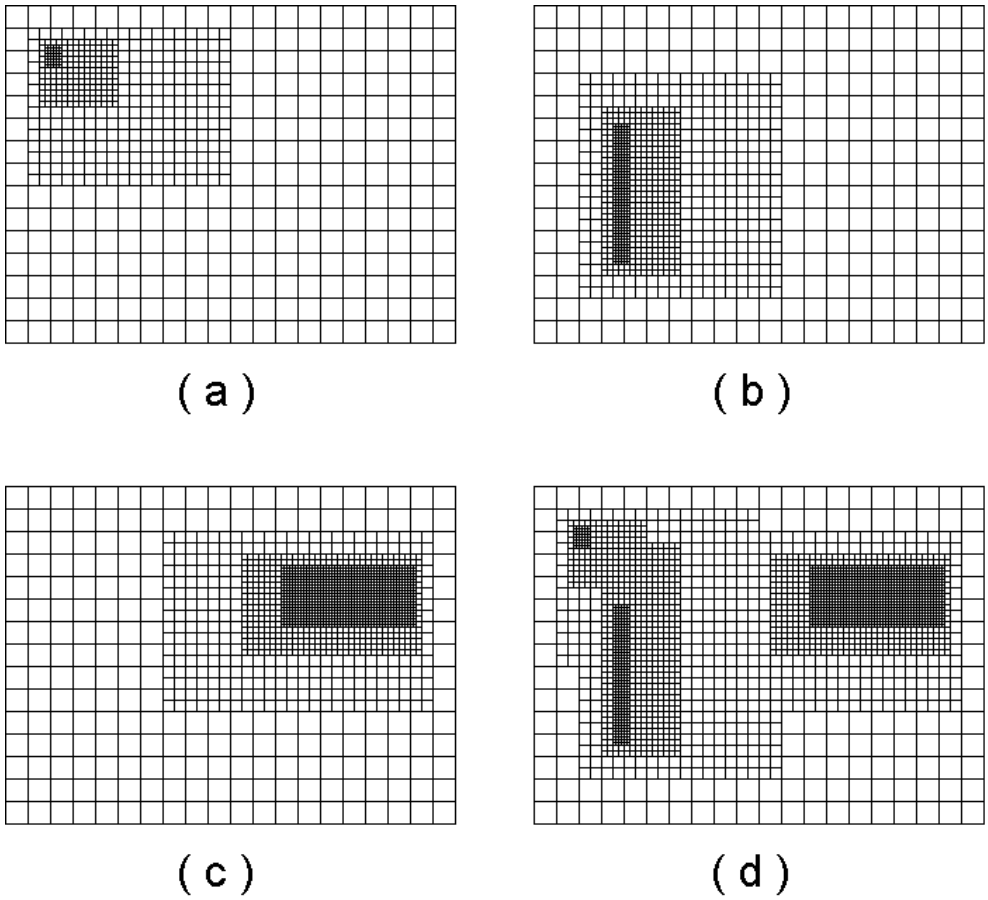


Figura A.4: Construcción de una geometría multifóvea: a) GMFD de tamaño adaptativo; b) GMFD de tamaño adaptativo; c) GMFD de tamaño adaptativo; y d) Geometría multirresolución multifóvea.

Apéndice B

Secuencias utilizadas en la segmentación espacio-temporal

En este apéndice se presentan las secuencias que se han empleado en las pruebas de segmentación espacio-temporal y se describen brevemente.

1 Captura mediante cámara acimutal fija en un pasillo

La secuencia de la Fig. B.1 fue capturada mediante una cámara de videoconferencia dispuesta en el techo de un pasillo y en perpendicular a éste, de forma que los efectos de perspectiva son mínimos debido al tamaño del campo de visión y la altura del techo. La cámara se encontraba fija y los móviles que atravesaban el pasillo no se encontraban limitados por ninguna restricción en cuanto a movimiento.

2 Captura panorámica mediante cámara fija con perspectiva I

La secuencia de la Fig. B.2 fue capturada desde un puente sobre la carretera de la escena mediante una cámara analógica, siendo digitalizada con posterioridad. En este caso, sí que aparece perspectiva en la escena, lo que puede apreciarse en el tamaño incremental de los móviles al acercarse a la cámara o en la fuga de la carretera hacia la línea del horizonte.

3 Captura panorámica mediante cámara fija con perspectiva II

La secuencia en la Fig. B.3 se capturó a nivel de suelo mediante una cámara analógica, siendo digitalizada con posterioridad. De nuevo aparece perspectiva en la escena, sólo que en este caso está más marcada debido a que los móviles pueden aparecer desde una distancia menor a la cámara.

4 Captura tipo videoconferencia

La secuencia en la Fig. B.3 presenta un plano típico de videoconferencia, donde la mayor parte del cuadro lo ocupa la interlocutora. El movimiento de ésta no se encuentra restringido en forma alguna.

5 Cubo de Rubik en rotación

En la Fig. B.5 se presenta una secuencia clásica de trabajo para estimación de movimiento, donde aparece un cubo de Rubik sobre una plataforma circular que gira muy lentamente.

6 Captura de fondo estático con cámara girando

La secuencia en la Fig. B.6 se capturó en un laboratorio, evitando la aparición de móviles en éste durante la captura, pero girando suavemente la cámara hacia la izquierda. Obviamente, este movimiento afecta de forma distinta a los diferentes planos de la imagen.

7 Captura de secuencia animada con cámara siguiendo al móvil

La Fig. B.7 presenta una secuencia de dibujos animados donde la cámara se desplaza para seguir al móvil presente en la escena. En este caso, dicho desplazamiento no tiene por qué ser homogéneo, ya que reproduce los movimientos del personaje y, por ejemplo, se detiene o frena cuando éste lo hace.

8 Captura panorámica con perspectiva y cámara móvil

La secuencia en la Fig. B.8 presenta el mismo caso de la secuencia de la Fig. B.2 con la salvedad de que en este caso la cámara se encuentra en movimiento, por lo que aumenta la complejidad del problema.

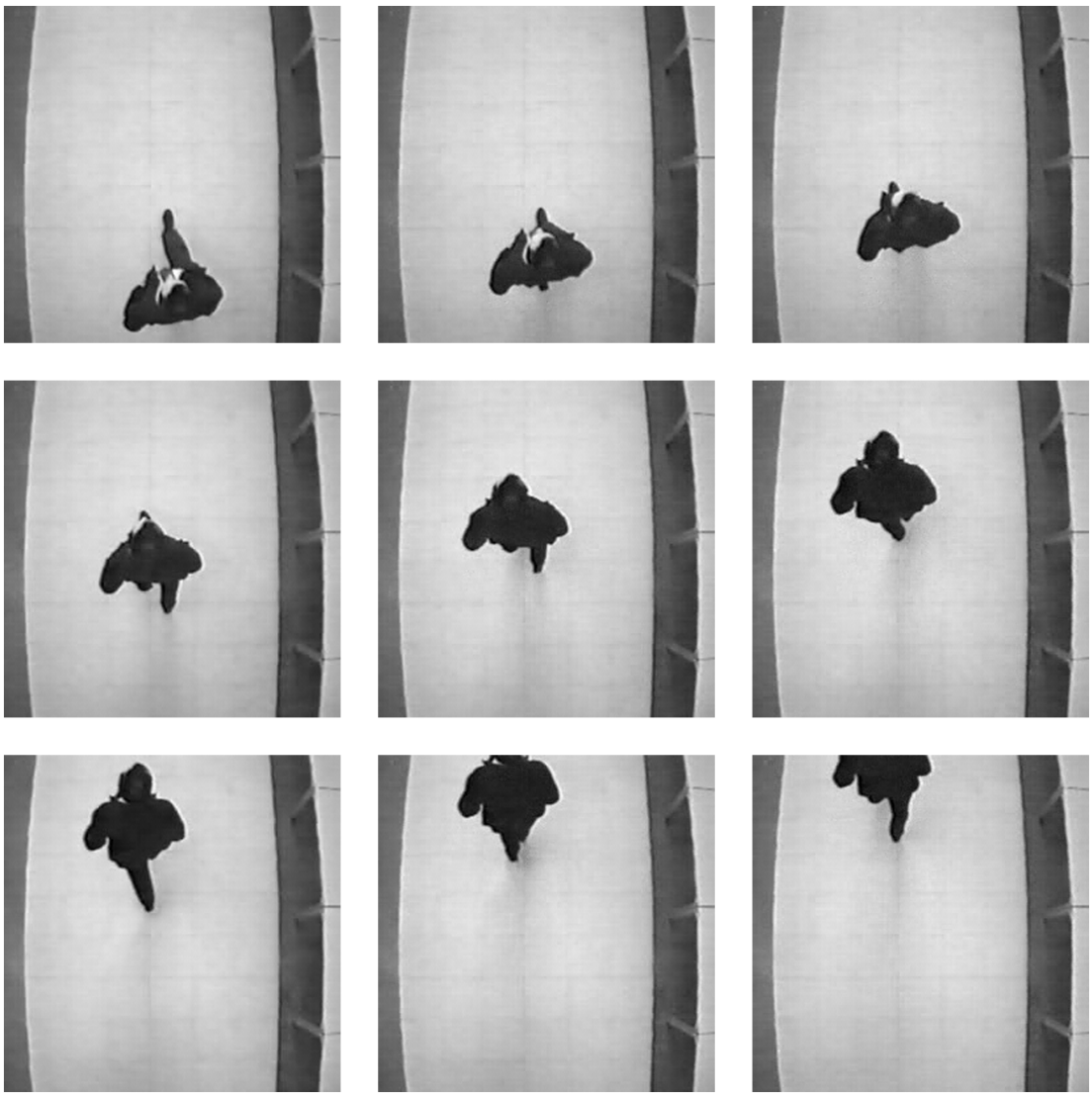


Figura B.1: Cámara acimutal fija y fondo simple.

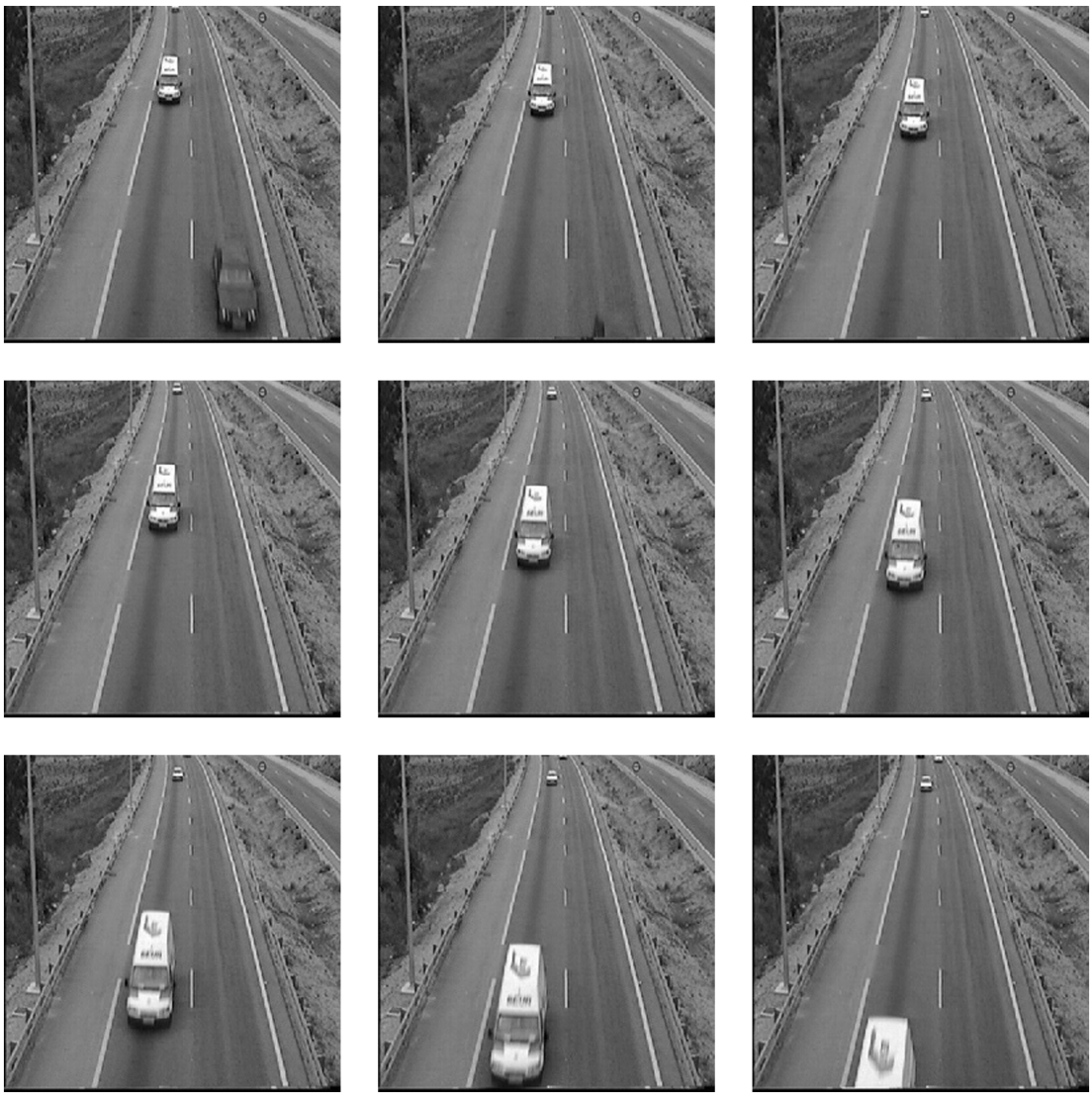


Figura B.2: Captura panorámica mediante cámara fija con perspectiva.

Figura B.3: Captura panorámica mediante cámara fija con perspectiva.

Figura B.4: Captura tipo videoconferencia.

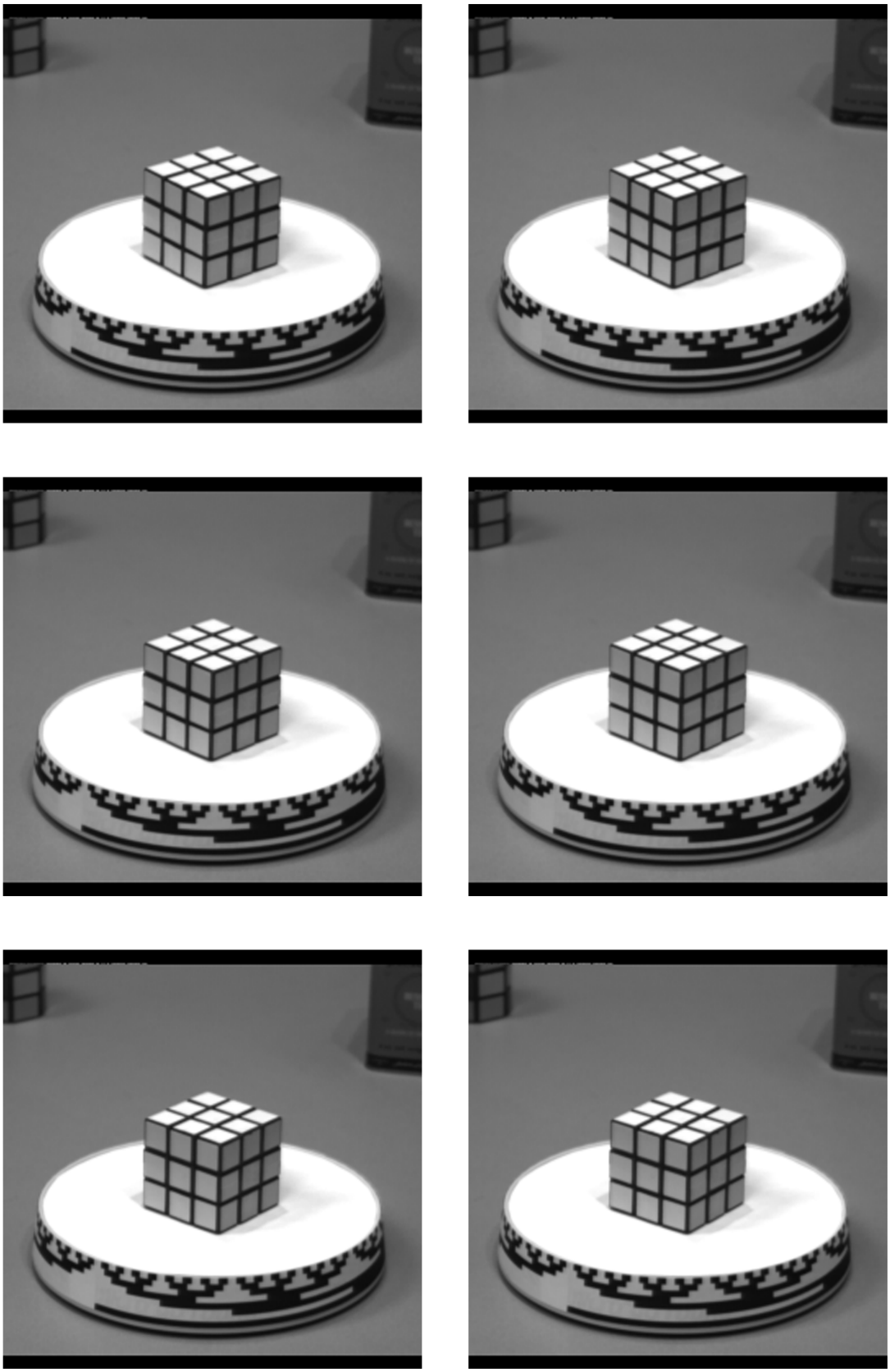


Figura B.5: Cubo de Rubik en rotación.

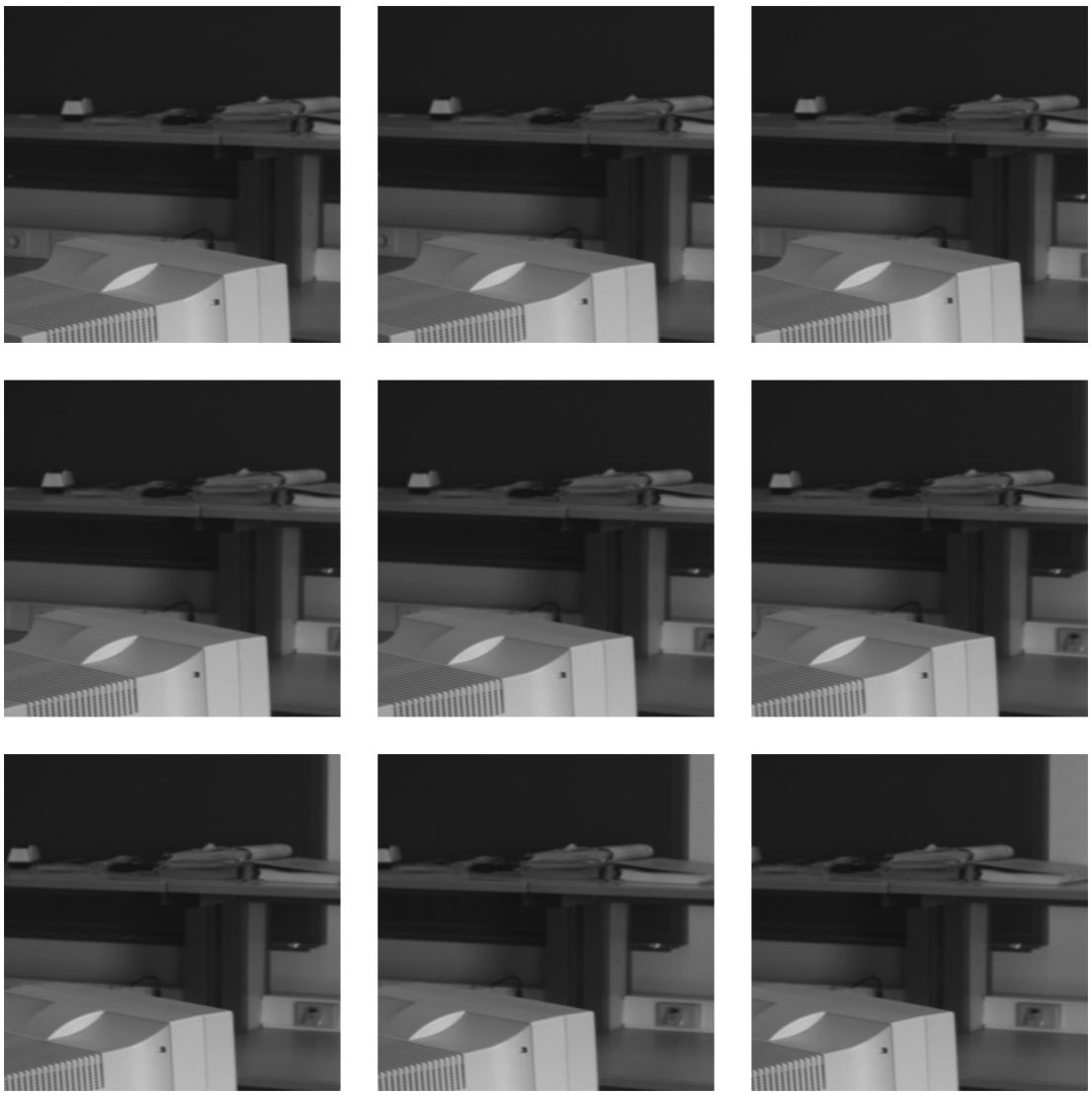


Figura B.6: Captura de fondo estático con cámara girando.

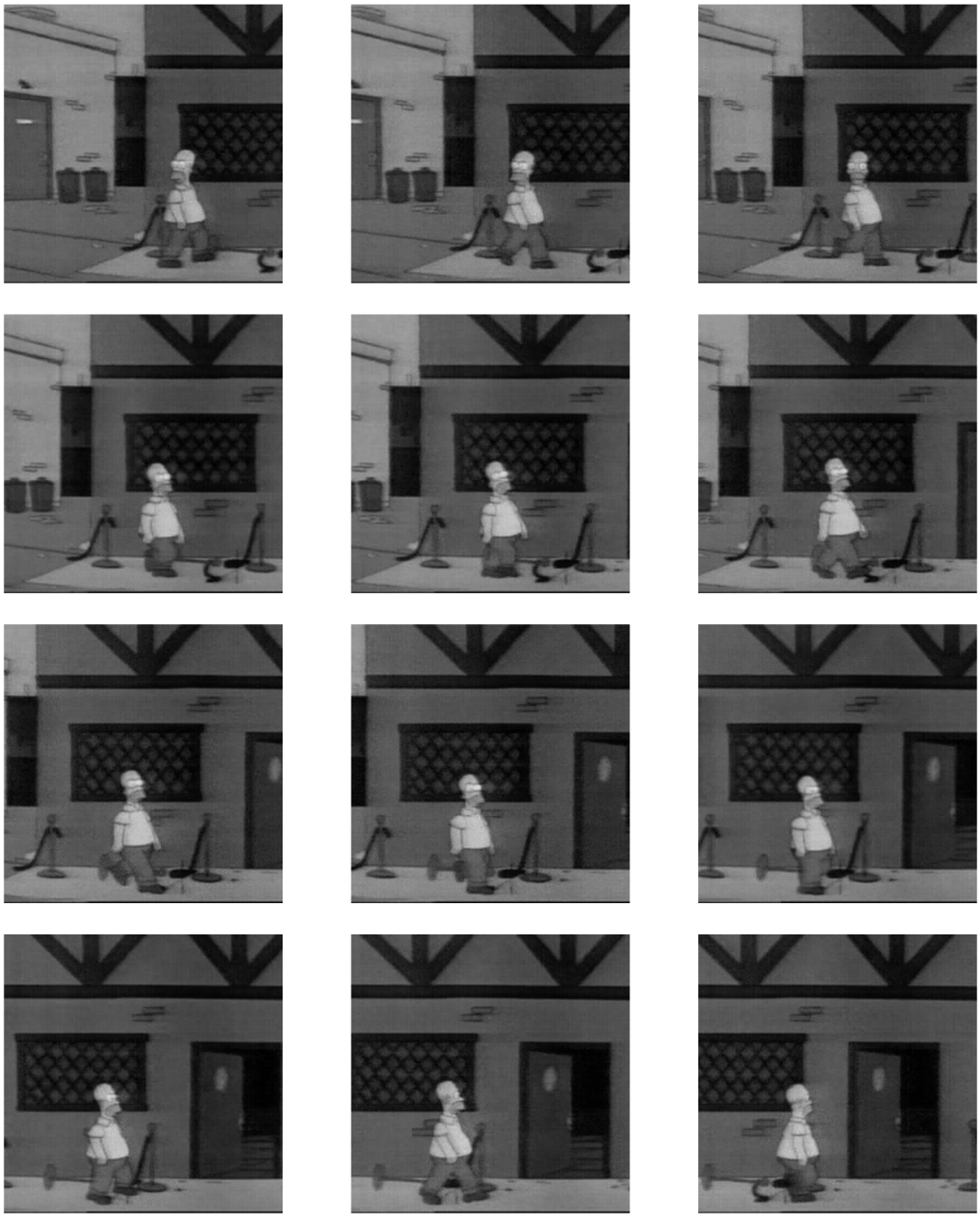


Figura B.7: Captura de secuencia animada con cámara siguiendo al móvil.

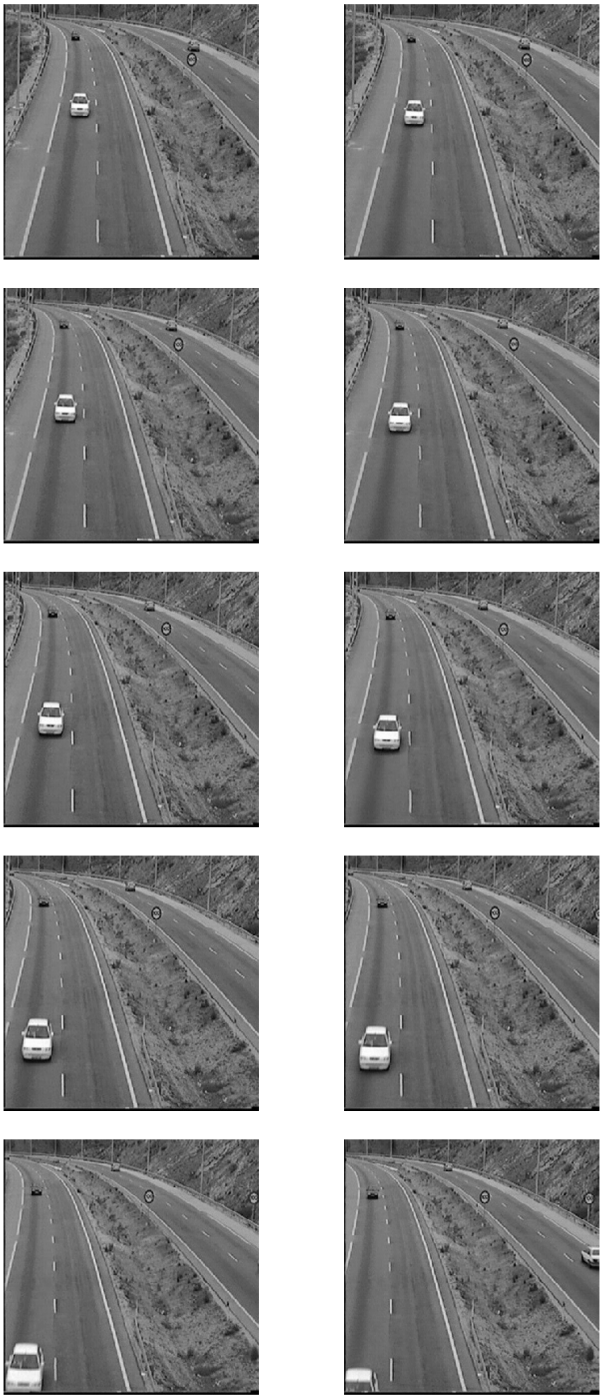


Figura B.8: Captura panorámica con perspectiva y cámara móvil.

MENU SALIR

Apéndice C

Descripción del Sw

1 Estructura general del sistema

En este apartado se describe el *software* empleado en la implementación del sistema propuesto en el Capítulo 5. Consta de un conjunto de procesos que se ejecutan de forma distribuida en dos ordenadores personales que, en adelante se denominan como PC Servidor y PC Cliente (ver Fig. C.1). Los procesos residentes en el PC Servidor son:

- **segment**: este proceso realiza la segmentación espacio-temporal de la escena y localiza las regiones de interés (ROIs) de la misma, determinando las coordenadas de las *bounding – boxes* que las contienen.
- **servidor**: recibe del proceso **segment** información sobre las posiciones que ocupan las ROIs y una imagen de resolución uniforme; gracias al proceso **controltiempo1** dispone del valor del retardo del canal; con esta información construye un paquete según el esquema descrito en la Fig. 5.6 del Capítulo 5 y lo envía al proceso **cliente**, que reside en el PC Cliente.
- **controltiempo1**: este proceso envía periódicamente (cada 2 segundos) un paquete de control al proceso **controltiempo2** y espera su retorno para, en base al tiempo transcurrido, estimar el retardo del canal, para que el proceso **servidor** pueda determinar el modo de transmisión.

Los procesos residentes en el PC Cliente son:

- **cliente**: la finalidad de este proceso consiste en, una vez recibido el paquete enviado por el proceso **servidor**, construir la imagen multifoveal completando la información recibida

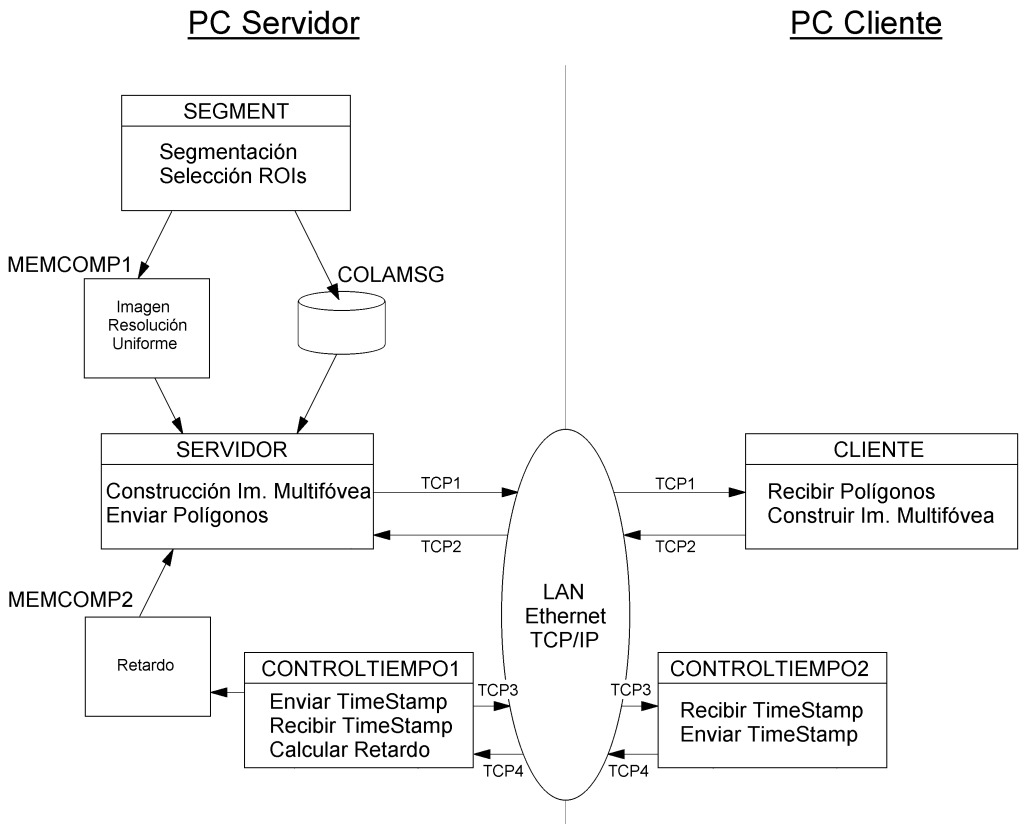


Figura C.1: Estructura general del sistema implementado

con la almacenada de recepciones anteriores.

- **controltiempo2**: este proceso recibe un paquete de control procedente del proceso **controltiempo1** y lo devuelve inmediatamente, sin aplicarle ningún tipo de procesamiento.

Para la implementación de este sistema se requería un sistema operativo (S.O.) multitarea que ofreciera mecanismos cómodos de comunicación entre procesos y entre procesadores. Además, debía disponer de librerías gráficas que permitieran el desarrollo de una interfaz cómoda y rápida. Entre los S.O. disponibles se eligió Linux porque, además de ofrecer todas las características anteriormente mencionadas, se trata de un sistema abierto, que permite la ampliación futura de algunas funcionalidades, como, por ejemplo, el empleo de un núcleo (*kernel*) de tiempo real. El apartado gráfico se desarrolló sobre la librería GTK, de libre distribución. Las comunicaciones entre procesos residentes en el mismo procesador se programaron atendiendo al estándar POSIX que incorpora el S.O. Linux. Éste incluye tres tipos de comunicación distintos (colas de mensajes, memoria compartida y semáforos) de los cuales sólo ha sido necesario emplear dos, que se describen a continuación:

- Colas de mensajes: son útiles cuando se requiere enviar pequeños paquetes de datos y permiten sincronizar procesos, ya que la recepción de mensajes de este tipo implica el bloqueo del receptor.

- Memoria compartida: se emplean para intercambiar cantidades elevadas de información o cuando dos procesos necesitan compartir información sin bloqueo.

Para la comunicación entre procesos que no residen en el mismo procesador se ha empleado la comunicación por *sockets* sobre TCP/IP. En la Fig. C.1 se aprecia la estructura del sistema y aparecen en detalle los caminos de comunicación establecidos:

- MEMCOMP1: segmento de memoria compartida en el que el proceso **segment** almacena la imagen de resolución uniforme para su posterior procesamiento; de esta forma, esta imagen también está disponible para el proceso **servidor**.

- MEMCOMP2: segmento de memoria compartida, en el cual el proceso **controltiempo1** almacena el retardo estimado del canal; el proceso **servidor** tiene acceso a esta información para determinar el *modo* de transmisión.

- COLAMSG: cola de mensajes establecida entre **segment** y **servidor** con la que ambos procesos se sincronizan e intercambian la siguiente información: tamaño de la imagen de resolución uniforme; número de ROIs en la imagen actual; y posición de cada ROI.

- TCP1: *socket* abierto entre **servidor** (origen) y **cliente** (destino) por el que se envía la siguiente información: tamaño de la imagen de resolución uniforme; nivel *waist* de la estructura multifoveal; número de ROIs en la estructura multifoveal a enviar; estructura poligonal correspondiente a cada ROI.

- TCP2: *socket* abierto entre **cliente** (origen) y **servidor** (destino) por el que el primer proceso envía una señal de conformidad (ACK) al segundo.

- TCP3: *socket* abierto entre **controltiempo1** (origen) y **controltiempo2** (destino) por el que el primero envía un paquete de control con un sello de tiempo que registra el momento en que dicho paquete es enviado.

- TCP4: *socket* abierto entre **controltiempo2** (origen) y **controltiempo1** (destino) por el que el primer proceso devuelve, sin modificar, el paquete de control al segundo.

2 Descripción de los procesos implicados

En este apartado se procede a describir los distintos procesos que intervienen en el sistema, atendiendo al esquema de la Fig. C.2.

2.1 Proceso SEGMENT

Este proceso reside en el PC Servidor y su función principal consiste en obtener las ROIs presentes en una secuencia de vídeo mediante el método de segmentación espacio-temporal propuesto.

A continuación se detallan, con una breve explicación de su funcionalidad, los principales módulos que usa:

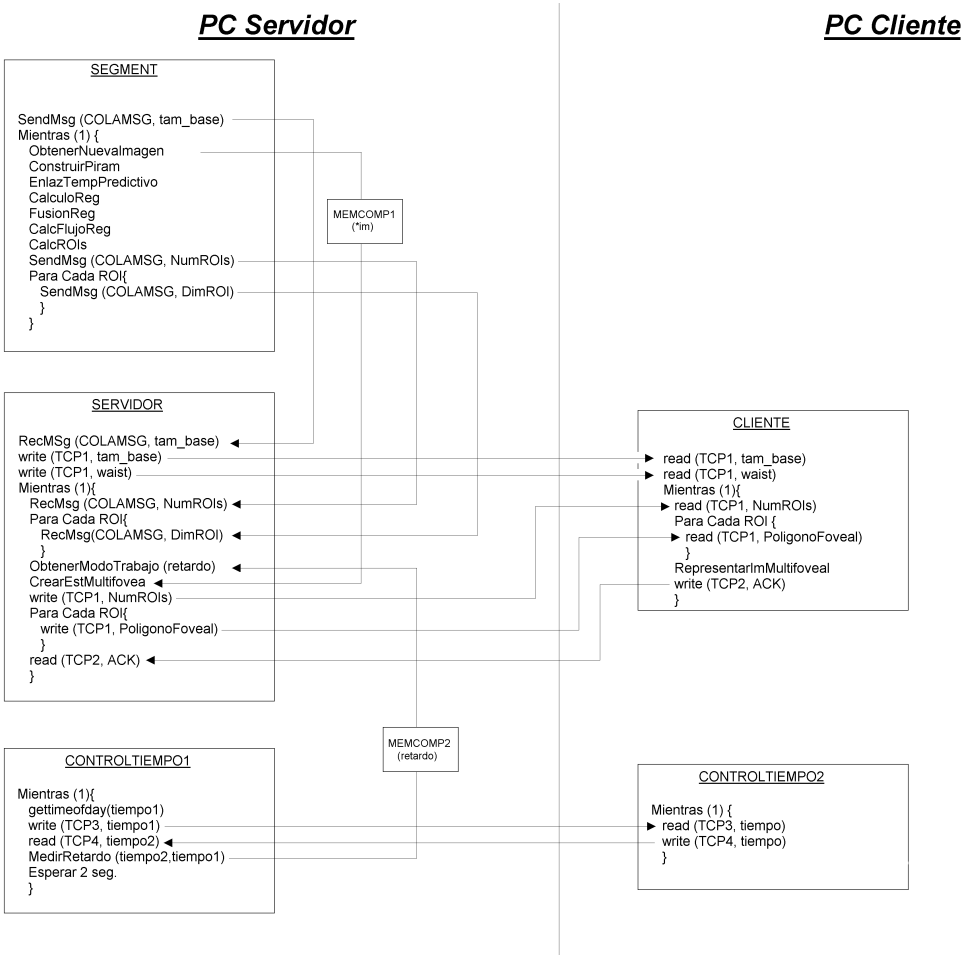


Figura C.2: Diagrama de flujo del sistema implementado

- main.c: incluye las funciones de iniciación y reserva de memoria de estructuras, creación de la interfaz gráfica, y controla el flujo principal del proceso, invocando al resto de funciones de otros módulos.

- gtkpiram.c: incluye un conjunto de funciones relacionadas con la representación de las estructuras jerárquicas empleadas.

- creapiram.c: es un conjunto de funciones relativas a la reserva de memoria para almacenar estructuras piramidales, así como funciones de construcción de estructuras de este tipo mediante sucesivos promediados de una imagen de resolución uniforme.

- linkpiram.c: este módulo contiene funciones de enlazado jerárquico adaptativo para una única imagen, así como de enlazado adaptativo temporal (entre dos pirámides creadas a partir de dos imágenes consecutivas, y de enlazado adaptativo temporal predictivo (desplazando la vecindad de búsqueda en función de los vectores de desplazamiento previamente estimados).

- fusion.c: en este módulo se encuentra el código de las funciones relacionadas con el proceso de fusión no supervisada de clases.

- flujo.c: incluye las funciones de cálculo de la posición que ocupan las clases en las bases de las pirámides $t-1$ y t y el código de los procesos de obtención de los vectores de desplazamiento de dichas clases.

- libsgmnt.a: esta librería agrupa un conjunto de funciones relacionadas con el enlazado entre celdas de un nivel (llamadas "hijos") y celdas del nivel superior (llamadas "padres").

- libFopen.a: en esta librería se encuentran las funciones relacionadas con el manejo de ficheros; permite manejar diversos formatos de imágenes y secuencias de vídeo.

- libIPC.a: esta librería contiene un conjunto de funciones relativas a las comunicaciones POSIX, concretamente las referidas a las rutinas IPC.

Entre todas las funciones implementadas en estos módulos, se describen a continuación las más significativas.

2.1.1 SendMsg.

void SendMsg(msgqid qid, long type, void *ptr, size_t size)

Función

Esta función envía a la cola de mensajes identificada por qid un mensaje de tipo type con el dato apuntado por la dirección de memoria ptr con una longitud de size bytes.

Módulo al que pertenece

libIPC.a

Algoritmo

- Construir estructura IPC de envío de mensaje.
- Enviar la estructura IPC.

Entradas

- msgqid qid: valor que identifica la cola por la que se debe enviar el mensaje.
- long type: tipo de mensaje a enviar.
- void *ptr: posición de memoria donde comienza la información a enviar.
- size_t size: número de bytes que ocupa la información que se desea enviar.

Salidas

Llama a

Llamada por

main() en main.c.

2.1.2 ConstructPyramid.

int ConstructPyramid(unsigned char *imagen, struct PYRAMID *pir)

Función

Esta función construye una estructura piramidal a partir de una imagen capturada previamente. Para ello copia la imagen original en el nivel base de la pirámide, y genera una versión promediada para cada uno de los sucesivos niveles que la componen hasta llegar al nivel de trabajo deseado.

Módulo al que pertenece

creapiram.c

Algoritmo

- Copiar la imagen original en la base de la pirámide.
- Para cada nivel por encima de la base, hasta el nivel de trabajo:
  - Construir el nivel seleccionado promediando con una vecindad 2x2 el nivel inferior.

Entradas

- unsigned char imagen: puntero de memoria que contiene la imagen original.
- PYRAMID *pir: puntero a una estructura de tipo PYRAMID en la que se ha de construir la pirámide resultante; esta estructura debe haber sido previamente iniciada mediante una llamada a la función PyramidInit().

Salidas

La pirámide construida.

Llama a

crea_et().

Llamada por

main() en main.c.

2.1.3 EnlazTempPredictivo.

void EnlazTempPredictivo(PYRAMID *pir_t1, PYRAMID *pir_t2, int vecindad, int ItMax)

Función

Esta función realiza el proceso de segmentación espacio-temporal según el esquema propuesto entre las estructuras piramidales apuntadas por pir_t1 y pir_t2.

Módulo al que pertenece

linkpiram.c

Algoritmo

- Extraer de *pir_t1 (pirámide $t-1$) el valor del vector de desplazamiento de las distintas clases existentes en la pirámide.
- Para cada nivel de la pirámide, empezando por la base, hasta el nivel de trabajo:
  - Enlazar cada celda hija del nivel actual de la pirámide apuntada por *pir_t1 con la celda padre más parecida en el nivel superior al actual de la pirámide apuntada por *pir_t2 (pirámide t) situada en una vecindad determinada sobre la proyección de la posición de dicha celda en función del desplazamiento que le corresponde según la clase a la que pertenece . Este es el proceso de estabilización de la pirámide $t-1$ con la pirámide t.
  - Enlazar cada celda hija del nivel actual de la pirámide apuntada por *pir_t2 con la celda padre más parecida situada en una vecindad determinada sobre dicha celda en el nivel superior al actual de la misma pirámide. Este es el proceso de estabilización de la pirámide t consigo misma.
  - Si se ha producido algún cambio en los enlaces, recalcular el valor de cada celda padre de la pirámide apuntada por *pir_t2 como media de los valores de todos sus hijos, tanto de los que pertenecen a su misma pirámide como de los que se encuentran en la pirámide anterior, y repetir el proceso de enlazado del nivel actual, salvo que se haya alcanzado al máximo número de iteraciones permitidas.
  - Si no se ha producido ningún cambio en los enlaces o se ha alcanzado el máximo número de iteraciones permitidas en el nivel actual, pasar al siguiente nivel.
- Una vez estabilizada la estructura piramidal, propagar desde las celdas "padre" a sus celdas "hijas", partiendo del nivel de trabajo que define el número de clases, un identificativo para

cada clase.

Entradas

- PYRAMID *pir_t1: puntero a una estructura de tipo PYRAMID que contiene la pirámide $t-1$.
- PYRAMID *pir_t2: puntero a una estructura de tipo PYRAMID que contiene la pirámide t.
- int vecindad: tamaño de la vecindad.
- int ItMax: número máximo de iteraciones permitidas en cada nivel.

Salidas

Conjunto de enlaces estabilizados en la pirámide t apuntada por *pir_t2 y segmentación por clases.

Llama a

link_et(), link_et_temp_proy(), comp_ult_temp(), comp_et_temp(), gen_clas(), gen_clas_temp().

Llamada por

main() en main.c.

2.1.4 CalculoReg.

void CalculoReg(PYRAMID *pir_ant, PYRAMID *pir_sig)

Función

Esta función calcula las dimensiones de las regiones que ocupan las clases en la base de las pirámides apuntadas por pir_ant y pir_sig.

Módulo al que pertenece

flujo.c

Algoritmo

- Para cada celda del nivel base de la pirámide apuntada por *pir_ant, anotar su pertenencia a la clase que le corresponda.
- Calcular las dimensiones de la región que ocupa cada clase en la base de la pirámide apuntada por *pir_ant.
- Para cada celda del nivel base de la pirámide apuntada por *pir_sig, anotar su pertenencia a la clase que le corresponda.
- Calcular las dimensiones de la región que ocupa cada clase en la base de la pirámide apuntada por *pir_sig.

Entradas

- PYRAMID *pir_ant: puntero a una estructura de tipo PYRAMID que contiene la pirámide $t-1$.
- PYRAMID *pir_sig: puntero a una estructura de tipo PYRAMID que contiene la pirámide t.

Salidas

Dimensiones de las regiones ocupadas por cada clase en las bases de las pirámides $t-1$ y t.

Llama a

Llamada por

main() en main.c.

2.1.5 FusionReg.

void FusionClases(PYRAMID *pir_ant, PYRAMID *pir_sig, int umbral_fusion, int *cambio_clases)

Función

Esta función realiza el proceso de fusión no supervisada de clases.

Módulo al que pertenece

fusion.c

Algoritmo

- Para cada clase existente en la pirámide apuntada por pir_sig, comprobar si su $bounding - box$ está solapada con la de alguna otra clase.
- Comprobar si hay conexión entre clases cuyas $bounding - boxes$ estén solapadas.
- Fundir en una sola las clases que estén conectadas entre sí y cuyos valores de nivel de gris difieran menos de un umbral.

Entradas

- PYRAMID *pir_ant: puntero a una estructura de tipo PYRAMID que contiene la pirámide $t-1$.
- PYRAMID *pir_sig: puntero a una estructura de tipo PYRAMID que contiene la pirámide t.
- int umbral_fusion: valor que representa la máxima diferencia en nivel de gris que pueden presentar dos clases para ser fundidas en una sola.

Salidas

- Como resultado de la aplicación de esta función las clases se han reagrupado, fundiéndose aquellas que cumplían las condiciones de fusión.
- int *cambio_clases: puntero a una variable de tipo int en la que refleja si se ha producido alguna fusión.

Llama a

NumeroClase(), TocanClases(), gen_clas(), gen_clas_temp().

Llamada por

main() en main.c.

2.1.6 CalcFlujoReg.

void CalcFlujoReg(PYRAMID *pir_ant, PYRAMID *pir_sig)

Función

Esta función realiza el cálculo de los vectores de desplazamiento de las clases.

Módulo al que pertenece

flujo.c

Algoritmo

- Para cada clase, calcular el centroide del área ocupada en la base de la pirámide apuntada por *pir_ant, y el centroide del área ocupada en la base de la pirámide apuntada por *pir_sig.
- Para cada clase, el vector de desplazamiento estimado será la diferencia entre los centroides anteriormente calculados.

Entradas

- PYRAMID *pir_ant: puntero a una estructura de tipo PYRAMID que contiene la pirámide $t-1$.
- PYRAMID *pir_sig: puntero a una estructura de tipo PYRAMID que contiene la pirámide t.

Salidas

Como resultado de la aplicación de esta función la estructura PIRAMID apuntada por el puntero *pir_sig contiene los vectores de desplazamiento de las clases.

Llama a

Llamada por

main() en main.c.

2.1.7 CalcROIs.

CalcROIs(PYRAMID *pir, int umbral_ROI, FOVEA **fov, int *numfoveas)

Función

Esta función establece las coordenadas de las $bounding - boxes$ de las ROIs presentes en la escena. Estas coordenadas determinarán la posición y el tamaño de las fóveas en la estructura multifóvea que se ha de enviar por el canal.

Módulo al que pertenece

main.c

Algoritmo

- Para cada clase, comprueba si su desplazamiento supera un umbral determinado.
- En caso afirmativo, comprueba que su compacidad (relación entre su masa y el área de su $bounding - box$) está dentro de unos límites preestablecidos.
- Una vez descartadas todas las clases que no cumplen los criterios anteriores, elije las más rápidas, hasta un número máximo de ROIs por imagen.

Entradas

- PYRAMID *pir_ant: puntero a una estructura de tipo PYRAMID que contiene la pirámide $t-1$.
- PYRAMID *pir_sig: puntero a una estructura de tipo PYRAMID que contiene la pirámide t.
- int umbral_ROI: valor mínimo que ha de tener el módulo del vector de desplazamiento de una clase para que ésta sea considerada ROI potencial.
- int *numfoveas: valor máximo de ROIs que se pueden elegir en una escena.

Salidas

- FOVEA **fov: puntero en el que se encuentra un vector con las dimensiones de las fóveas que cubren las ROIs detectadas y ordenadas según los criterios previamente descritos.
- int *numfoveas: número de ROIs detectadas en la escena.

Llama a

ClaseMasRapida()

Llamada por

main() en main.c.

2.2 Proceso SERVIDOR

Este proceso reside en el PC Servidor y su función es, una vez recibidas las dimensiones de las ROIs presentes en la escena, construir un paquete para transmitir una imagen multifóvea completa o incompleta, atendiendo a la congestión del canal.

A continuación se detallan con una breve explicación de su funcionalidad los principales módulos que usa:

- servidor.c: incluye las funciones de iniciación y reserva de memoria de estructuras, iniciación de comunicaciones con otros procesos, tanto los residentes en el mismo PC, como los que se ejecutan en el PC Cliente, y controla el flujo principal del proceso, invocando al resto de funciones de otros módulos.
- abrirsocket.c: contiene las funciones necesarias para abrir *sockets*.
- funcpol.c: incluye las funciones que se emplean para transmitir la estructura completa de un polígono foveal por un *socket*.
- imagenfoveal.c: módulo en el que se definen las funciones que presentan en una ventana GTK una imagen foveal.
- libIPC.a: esta librería contiene un conjunto de funciones relativas a las comunicaciones POSIX, concretamente aquellas que se refieren a las rutinas IPC.

Entre todas las funciones implementadas en estos módulos, se procede a describir las más significativas.

2.2.1 RecMsg.

void RecMsg(msgqid qid, long type, void ptr, size_t size)

Función

Esta función recibe de la cola de mensajes identificada por qid un mensaje de tipo type y lo copia en la dirección de memoria apuntada por ptr con una longitud de size bytes.

Módulo al que pertenece

libIPC.a

Algoritmo

- Copiar la estructura IPC de la cola de mensajes.

Entradas

- msgqid qid: valor que identifica la cola por la que se debe recoger el mensaje.
- long type: tipo de mensaje a recibir.
- void *ptr: puntero de memoria donde se debe guardar la información recibida.
- size_t size: número de bytes que ocupa la información que se desea recibir.

Salidas

Llama a

Llamada por

main() en servidor.c.

2.2.2 ObtenerModoTrabajo.

void ObtenerModoTrabajo(int retardo, int tam_base, FOVEA **fov, int numfoveas, int frame_rate, int *modo)

Función

Esta función determina el modo de trabajo (número de anillos a enviar) en función del retardo del canal, número de fotogramas a enviar por segundo, e información acerca de la imagen multifóvea a enviar.

Módulo al que pertenece

servidor.c

Algoritmo

- Encontrar el valor de *modo* tal que permita alcanzar el $frame\_rate$ deseado con las ROIs establecidas previamente.

Entradas

- int retado: tiempo que ha tardado un paquete de control de longitud conocida en atravesar el canal en camino de ida vuelta.
- int tam_base: tamaño de la imagen original.
- FOVEA **fov: puntero en el que se encuentra un vector con las dimensiones de las fóveas que cubren las ROIs detectadas y ordenadas según los criterios previamente descritos.
- int numfoveas: número de ROIs detectadas en la escena.
- int frame_rate: número de fotogramas por segundo requeridos en recepción.

Salidas

- int *modo: modo de transmisión elegido.

Llama a

Llamada por

main() en servidor.c.

2.2.3 CrearEstMultifovea.

void CrearEstMultifovea(unsigned char *imagen, FOVEA *foveae, int num_foveas, POLYGON **poligonos)

Función

Esta función determina el modo de trabajo (número de anillos a enviar) en función del retardo del canal, número de fotogramas a enviar por segundo, e información acerca de la imagen multifóvea a enviar.

Módulo al que pertenece libsgmnt.a

Algoritmo

- Para cada ROI encontrada, construir una estructura foveal.

Entradas

- unsigned char *image: puntero que apunta a la zona de memoria compartida en la que se encuentra la imagen original.
- FOVEA **foveae: puntero en el que se encuentra un vector con las dimensiones de las fóveas que cubren las ROIs detectadas y ordenadas según los criterios previamente descritos.
- int num_foveas: número de ROIs detectadas en la escena.

Salidas

- POLYGON **poligonos: puntero que apunta a un vector de estructuras foveales.

Llama a

polyadap().

Llamada por

main() en servidor.c.

2.3 Proceso CONTROLTIEMPO1

Este proceso reside en el PC Servidor y su función es, una vez recibido de vuelta el paquete de control enviado al PC Cliente, calcular el retardo del canal.

Su funcionamiento se basa en el uso de un conjunto de funciones propias del sistema, que incluye las siguientes:

- gettimeofday: recupera la hora actual.

- write: función perteneciente a la librería estándar de manejo de ficheros, que en este caso se usa para escribir sobre un *socket* previamente abierto.

- read: función perteneciente a la librería estándar de manejo de ficheros, que en este caso se usa para leer de un *socket* previamente abierto.

2.4 Proceso CLIENTE

Este proceso reside en el PC Cliente y su función es, una vez recibido el paquete que contiene la imagen multifoveal completa o incompleta, representar en pantalla la imagen multifoveal recibida, si es completa, o una imagen compuesta al completar la recibida con información almacenada de envíos anteriores.

Su funcionamiento se basa tanto en el uso de funciones del sistema, como en funciones propias. De las propias, cabe mencionar la siguiente:

2.4.1 RepresentarImMultifoveal.

void RepresentarImMultifoveal(GTKIMAGENFOVEAL *ventfov, POLYGON *polnew, POLYGON *polold, int NivelesPerdidos)

Función

Esta función dibuja en una ventana GTK una imagen multifoveal, tomando de *polnew los niveles recibidos, y de *polold los niveles perdidos.

Módulo al que pertenece

imagenfoveal.c

Algoritmo

- Dibujar los niveles perdidos de las distintas estructuras foveales (una para cada fóvea) desde *polold.
- Dibujar los niveles recibidos de las distintas estructuras foveales (una para cada fóvea) desde *polnew.

Entradas

- GTKIMAGENFOVEAL *ventfov: puntero a una estructura que contiene información necesaria para representar imágenes foveales con la librería gráfica GTK.
- POLYGON *polnew: puntero a la estructura que almacena los niveles recibidos recientemente.
- POLYGON *polold: puntero a la estructura que almacena los niveles recibidos anteriormente.
- int NivelesPerdidos: número de niveles que no han llegado en la estrustura apuntada por *polnew en la última transmisión, y que está íntimamente relacionado con el modo de transmisión.

Salidas

Imagen multifoveal representada en pantalla.

Llama a

ConstruirImagenPoligonalIncompleta(), gtk_preview_draw_row(), gtk_widget_draw ().

Llamada por

main() en servidor.c.

2.5 Proceso CONTROLTIEMPO2

Este proceso reside en el PC Cliente y su función es devolver todos los paquetes de control que recibe desde el PC Servidor. Su funcionamiento es, por tanto, muy sencillo y se basa en el uso de un conjunto de funciones propias del sistema (read y write), que ya han sido comentadas.